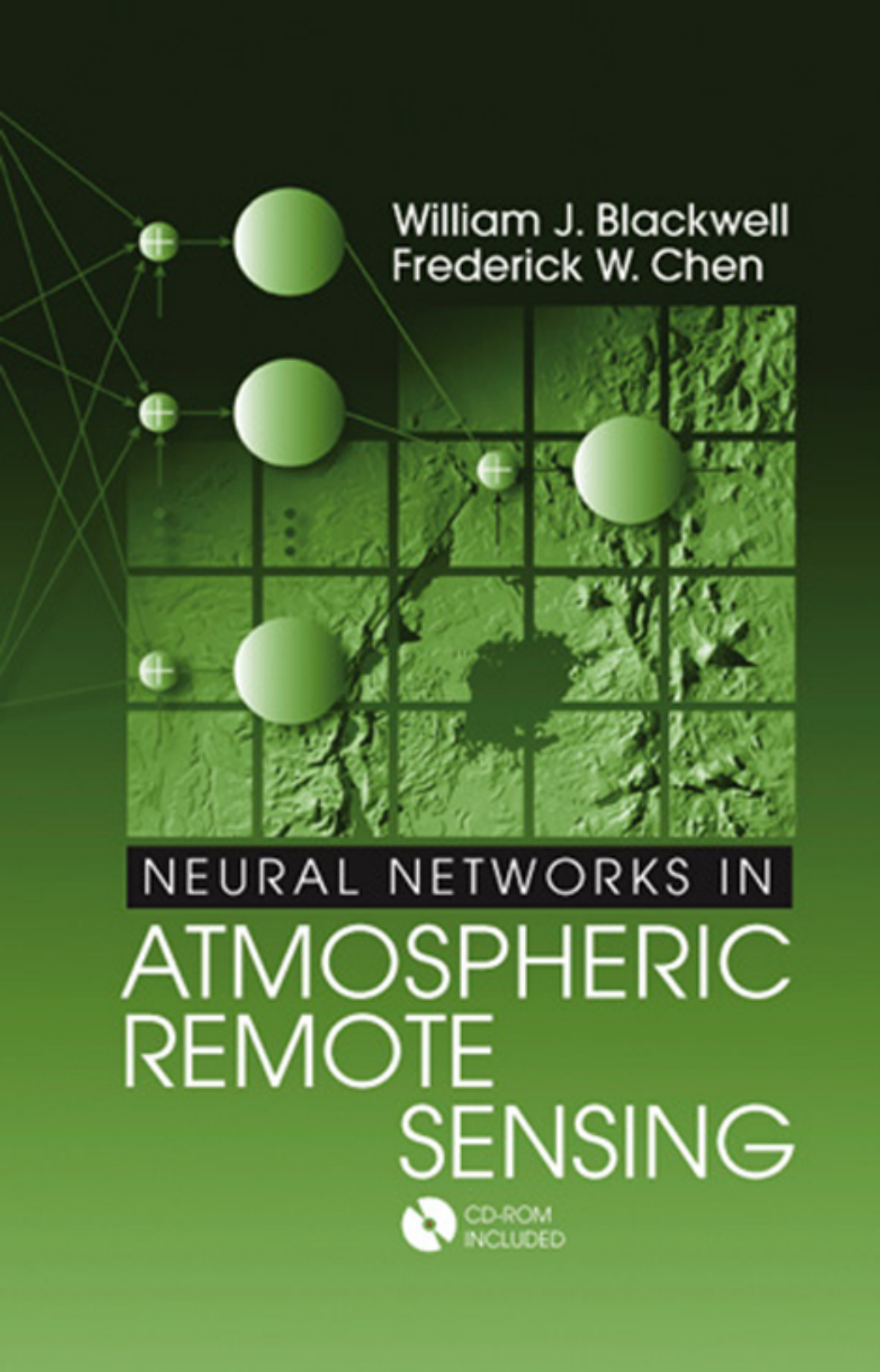

William J. Blackwell
Frederick W. Chen
NEURAL NETWORKS IN
ATMOSPHERIC
REMOTE
SENSING
CD-ROM
INCLUDED

Neural Networks in Atmospheric Remote Sensing

This is a sample library statement

Neural Networks in Atmospheric Remote Sensing

William J. Blackwell
Frederick W. Chen

Library of Congress Cataloging-in-Publication Data
A catalog record for this book is available from the U.S. Library of Congress.

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library.

ISBN-13 978-1-59693-372-9

Cover design by Yekaterina Ratner

Lincoln Laboratory
244 Wood Street
Lexington, MA 02420

This work was funded in part by the National Oceanic and Atmospheric Administration under Air Force contract FA8721-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

10 9 8 7 6 5 4 3 2 1

Disclaimer:
This eBook does not include the ancillary media that was packaged with the original printed version of the book.

To our families

Contents

Preface

This book is intended to provide a practical, applications-oriented treatment of neural network methodologies for use in atmospheric remote sensing. We focus on the retrieval of atmospheric parameters, such as the Earth's temperature and water vapor profiles and precipitation rate, but the techniques described can be applied to a wide variety of problems where function approximation is required. We use simple, largely theoretical examples to provide the reader with intuition on how performance is affected by basic neural network attributes such as model selection, initialization, and training methodology, and we then build these simple techniques into larger, "real-world" applications that are common throughout the field of atmospheric remote sensing. Many of the examples are accompanied by MATLAB™ (www.mathworks.com) software codes (available on the accompanying CD-ROM in the back of the book) that can be used as building blocks for larger and more complex problems. These codes were written using the freely available NETLAB Neural Network package and do not require any MATLAB™ "add-on" toolboxes.

The book is organized as follows. Chapter 1 presents an overview of the issues and challenges related to the retrieval of atmospheric parameters from remote measurements of atmospheric emission and scattering. Chapter 2 provides a summary of many of the relevant physical processes at the foundation of atmospheric remote sensing, including atmospheric composition, wave propagation, atmospheric absorption and scattering, radiative transfer, and spectrometer systems. Chapter 3 then follows with an overview of mathematical inversion methods commonly used in retrievals of atmospheric parameters from radiometric data, including iterative minimum variance approaches, regression, and Bayesian techniques. Constraints imposed on solutions through the use of regularization techniques are also discussed.

Chapter 4 presents theoretical background on many of the signal processing techniques commonly used in conjunction with neural network processing, including principal components analysis (PCA), Wiener filtering, periodic data representation, and blind estimation. Chapter 5 introduces multilayer perceptron neural networks and provides a general description of network topology and optimization in the broad context of machine learning. Chapter 6 presents detailed methodologies for network model selection, initialization, and training. Guidance is also provided for the use of these methodologies in practice, and common pitfalls are discussed. Chapter 7 discusses pre- and post-processing techniques that can be used to improve the effectiveness of the concomitant retrieval operators by reducing the volume of data that must be processed, and/or removing unwanted, interfering signals, such as noise, clouds, or surface variability. Chapter 8 provides practical guidance on the evaluation of network performance and discusses the important concepts of under- and overfitting, early stopping and weight decay, and network stability. The book culminates with two examples of complete neural network retrieval systems used to retrieve atmospheric parameters from passive spaceborne spectrometers. Chapter 9 presents an example of a highly nonlinear problem, retrieval of precipitation rate from passive microwave measurements, and Chapter 10 presents an example of a high-dimensional problem, retrieval of temperature and moisture profiles from combined microwave and infrared measurements. A discussion of possible future work is given in Chapter 11.

A great number of people have contributed to the work presented in this book. George Aumann, Chris Barnet, Mous Chahine, Mitch Goldberg, Tom Pagano, Bill Smith, and Joel Susskind of the AIRS Science Team have provided helpful suggestions and constructive criticisms that have helped to shape the course of much of our neural network retrieval research involving the AIRS products. Larrabee Strow and Scott Hannon provided the AIRS transmittance codes and guidance on their use. Many in the broad NPOESS community have provided valuable feedback and support, including Karen St. Germain, Degui Gu, Xu Liu, Steve Mango, and Dan Zhou. We would like to thank Laura Bickmeier, Chuck Cho, Monica Coakley, Harry Finkle, Chris Gittins, Laura Jairam, John Kerekes, Dan Mooney, Mike Pieper, Phil Rosenkranz, Chinnawat Surussavadee, and Kevin Wong for many helpful discussions. We are grateful to Dave Weitz, Vince Leslie, and Dimitris Manolakis for thoughtful comments on the manuscript and to Seth Hall for computer support. We thank Greg Berthiaume, Hsiao-hua Burke, and Roger Sudbury for their support and encouragement over the course of this project.

A special debt of gratitude is expressed to Dave Staelin for the guidance and support he has provided on all facets of this work, including preparation

of this book. His thoughtful comments have added substantially to the clarity and completeness of the presentation.

William J. Blackwell would especially like to thank Megan for putting up with his preoccupation with this project over more than a few nights and weekends and for offering many fresh and insightful perspectives.

1

Introduction

Measurements of the state and composition of the Earth's surface and atmosphere have been made using passive microwave and infrared sensors for over 50 years [1]. Applications of these remote measurements are numerous, and encompass fields ranging from meteorology, oceanography, geology, and ecology. For example, satellite measurements of atmospheric temperature are used to improve weather forecasting models, analyze climate change, and study the radiation budget of Earth [2].

Recent advances in airborne and spaceborne sounding platforms have made atmospheric measurements possible on a global scale, and advances in sensor technologies have pushed the limits of achievable spatial and temporal resolution to unprecedented levels. These performance improvements, however, are not without concomitant data processing difficulties. The vast amount of data generated by present and next generation sounding systems must be transmitted and processed in a timely manner (usually near real time), which requires processing algorithms that are both computationally efficient and robust to sensor and atmospheric anomalies (an erupting volcano, for example).

1.1 Present Challenges

A principal complication in the retrieval of geophysical parameters such as the global three-dimensional atmospheric temperature and moisture profile from satellite radiance observations is the nonlinear, non-Gaussian, and ill-posed physical and mathematical relationship between the radiance observed by a remote sensing instrument and the desired retrieved quantity. Great strides have recently been made to improve and better characterize the models that are used to capture these relationships, but these models are seldom

invertible by direct means, usually due to the complex nature of the underlying physics of the relevant geophysical processes. Common inversion approaches involve iterated numerical optimization methodologies that minimize a cost function subject to constraints imposed by a set of regularization parameters constructed so that the optimization tends toward solutions that are more "statistically probable" and/or "physically realistic." These regularization parameters are often largely subjective, and the construction of effective retrieval algorithms therefore requires a substantial component of "black art" to balance the use of the information content in the measured upwelling atmospheric radiances with the plausibility of the retrieval.

A logistical drawback to iterated, model-based inversion techniques is the computational burden required to carry out the numerical optimizations. Modern thermal infrared sensors measure spectral radiances in tens of thousands of separate wavebands (sometimes termed "hyperspectral" or even "ultraspectral") for each observed pixel. The computational complexity of the optimization routines typically scales as the square (or cube) of the number of channels, and it is rare that all of the information available in the radiance spectrum is used. The vast presence of clouds further degrades performance, and therefore a separate preprocessing stage is often employed prior to (or in concert with) numerical inversion to correct the substantial radiance errors that can be introduced due to the high opacity of cloud formations in the infrared wavelengths.

1.2 Solutions Based on Neural Networks

An alternative approach to the numerical inversion approach described above is statistical regression (parameterized function approximation), where an ensemble of input/output pairs is used to empirically derive statistical relationships between the ensembles. In the case of linear regression, second-order statistical moments (covariances) are used to compute a linear fit that minimizes the sum-squared error between the fit and the data. A linear representation is seldom sufficient to fully characterize the complex statistical relationships endemic in atmospheric data, and nonlinear regression techniques must be used. An artificial neural network is a special class of nonlinear regression operators – the mathematical structure of a neural network is chosen to afford several desirable properties, including scalability and differentiability. Patterned after the human nervous system, an artificial neural network (hereafter, simply a neural net) consists of interconnected neurons, or nodes, that implement a simple, nonlinear function of the inputs. Usually, the inputs are linearly weighted (the weights modulate each input and the biases provide an offset) and passed through an activation function

(often nonlinear). The power of neural networks, both from the standpoint of their capabilities and the derivation of their free parameters, stems from the parallel structure of the computational elements. In this book, we primarily consider feedforward connections of layers of nodes with sigmoidal (soft-limit) activation nodes. Many other variations can be used, but the feedforward variety is most common and the techniques described here are readily applied to other topologies.

The neural network approach offers several substantial advantages over iterated, model-based inversion methodologies. Once the weights and biases are derived (during the training process), the network operates very quickly and can be easily implemented in software. This simplicity and speed greatly facilitates the development and maintenance, and therefore cost, of complex geophysical retrieval systems that process high volumes of hyperspectral data. The trained neural networks are continuous and differentiable, which simplifies error propagation and therefore performance sensitivity analyses. Finally, neural networks can approximate functions with arbitrarily high degrees of nonlinearity with a sufficient number of nodes and layers. These advantages have spurred the recent use of neural network estimation algorithms for geophysical parameter retrievals [3–5]. Methods based on neural networks for data classification have also become commonplace, although we will focus on regression in this book. Many of the tips and techniques discussed, however, are directly applicable to both types of problems.

1.3 Mathematical Notation

One of the primary goals of this book is to cohesively unite the fields of statistics and estimation, mathematical inversion, machine learning, and radiative transfer in the context of the atmospheric retrieval problem. A principal challenge in this endeavor is to reconcile the often disparate sets of mathematical notation used in the literature for each field. For example, Rodger's classic treatment of the retrieval of atmospheric state variables [6] denotes the state vector to be retrieved (i.e., the *output* of the retrieval algorithm) as "$\mathbf{x}$," whereas the statistical and machine learning literature almost always reserves $\mathbf{x}$ to denote the *input* of the algorithm. Our objective has therefore been to develop a notational convention that maximizes the commonality of the notations of the various communities. To minimize confusion and ambiguity, we have also tried to choose mnemonic notations, where possible. The conventions and variable notations we have adopted are shown in Tables 1.1 and 1.2, respectively.

Table 1.1
Mathematical Notation

Variable and Operator Types	Notation
Scalars and functions with scalar output	Lowercase letters
Vectors	Uppercase letters
Matrices	Boldface uppercase letters
Vector-valued functions	Boldface lowercase letters
Expected value	$E(\cdot)$
Transpose	$(\cdot)^T$
Noisy variable	$\widetilde{(\cdot)}$
Estimate	$\widehat{(\cdot)}$

Table 1.2
Variable Names

Variable Name	Notation
Radiance measurement vector (retrieval input)	R
Atmospheric state vector (retrieval output)	S
Neural network target (or truth) vector	T
Generic input vector	X
Generic output vector	Y
Noise vector	Ψ
Weight matrix	$\mathbf{W}$
Covariance matrix	$\mathbf{C}$
Noise covariance matrix	$\mathbf{C}_{\Psi\Psi}$
Error covariance matrix	$\mathbf{C}_{\epsilon\epsilon}$
Kernel function	$k(X, X')$
Feature map	$\Phi(X)$

References

[1] D. H. Staelin. "Passive remote sensing at microwave wavelengths." *Proceedings of the IEEE*, 57(4):427–439, April 1969.

[2] G. L. Stephens. *Remote Sensing of the Lower Atmosphere*. Oxford University Press, New York, 1994.

[3] P. M. Atkinson and A. R. L. Tatnall. "Introduction to neural networks in remote sensing." *Int. J. Remote Sensing*, 18(4):699–709, 1997.

[4] V. M. Krasnopolsky and F. Chevallier. "Some neural network applications in environmental sciences. Part I: Forward and inverse problems in geophysical remote measurements." *Neural Netw.*, 16(3-4):321–334, 2003.

[5] V. M. Krasnopolsky and F. Chevallier. "Some neural network applications in environmental sciences. Part II: Advancing computational efficiency of environmental numerical models." *Neural Netw.*, 16(3-4):335–348, 2003.

[6] C. D. Rodgers. "Retrieval of atmospheric temperature and composition from remote measurements of thermal radiation." *J. Geophys. Res.*, 41(7):609–624, July 1976.

2

Physical Background of Atmospheric Remote Sensing

We begin with a broad overview of relevant physical issues in passive atmospheric sounding to provide background and context to results developed later in the book. For additional details, the reader is referred to excellent references on tropospheric remote sensing [1], atmospheric science [2, 3], atmospheric radiation [4], and electromagnetic wave propagation [5].

2.1 Overview of the Composition and Thermal Structure of the Earth's Atmosphere

The Earth's atmosphere extends over 100 km from its surface, and can roughly be categorized into four layers based on the thermal and chemical phenomena that occur within each layer. These layers are (in increasing altitude) the troposphere, the stratosphere, the mesosphere, and the thermosphere. The boundaries between each layer are usually not well-defined, but do show characteristic features. They are the tropopause, stratopause, and mesopause, respectively. The troposphere extends from the surface to an altitude of approximately 12 km (as low as 7 km near the poles and as high as 17 km near the equator) and is characterized by a steady decrease in temperature with altitude. Approximately 80% of the total mass of the atmosphere is contained in the troposphere, and almost all of the Earth's weather is created there. The troposphere is therefore the focus of most atmospheric sounding research, including the examples presented in this book. The tropopause marks the region of the atmosphere where the temperature gradually changes from decreasing with altitude to increasing with altitude, and forms a somewhat nebulous boundary layer between the troposphere and

the stratosphere. The stratosphere extends to an altitude of approximately 40 km, and is characterized by relatively high concentrations of ozone (a few parts per million). A sharp increase in temperature with altitude occurs in the stratosphere due to the absorption of ultraviolet radiation by ozone. High cirrus clouds sometimes form in the lower stratosphere, but for the most part there are no significant weather patterns in this layer, and horizontal and vertical atmospheric variability is much smaller than in the troposphere. The mesosphere extends from approximately 40 to 80 km, and is characterized by a decreasing temperature with altitude. Extremely low temperatures ($\sim -150°\mathrm{C}$) present at the top of the mesosphere sometimes allow the presence of noctilucent clouds, thought to be made of ice crystals that have formed on dust particles. The transition from the mesosphere to the thermosphere layer begins at an altitude of approximately 80 km. The thermosphere is characterized by warmer temperatures caused by the absorption of the sun's short-wave ultraviolet radiation. This radiation penetrates the upper atmosphere and causes the atmospheric particles to become positively charged. These ionized particles build up to form a series of layers, often referred to as the ionosphere.

2.1.1 Chemical Composition of the Atmosphere

The Earth's atmosphere is composed of a variety of gases. Each gas interacts characteristically with electromagnetic radiation of a given frequency. This relationship forms the physical basis by which the atmospheric temperature can be measured by observing radiation of different frequencies that has been emitted by and transmitted through the atmosphere.

The average fractional volumes of various species in the Earth's atmosphere are given in Table 2.1. Perhaps the most important gases in the atmosphere, from the point of view of their interaction with electromagnetic radiation, are water vapor, oxygen, carbon dioxide, and ozone. Oxygen and carbon dioxide are well-mixed in the atmosphere below approximately 100 km, and therefore frequencies near the resonances of these molecules are desirable for temperature sounding. The vertical distribution of ozone reaches maximum concentration near 25 km. Above 30 km, ozone is rapidly formed by photochemical reactions from oxygen so that an equilibrium is established during the daylight hours. Below this level, ozone is created more slowly and is highly variable [1]. Water vapor is perhaps the most influential atmospheric gas from the perspective of weather and climate processes. This is primarily due to its high temporal and spatial variability in the lower troposphere and its large role in energy transfer.

Table 2.1
Composition of the Earth's Atmosphere (*Source*: [1])

Molecule	Volume Fraction†	Comments
N_2	0.7808	Photochemical dissociation high in the ionosphere; mixed at lower levels
O_2	0.2095	Photochemical dissociation above 95 km; mixed at lower levels
H_2O	< 0.04	Highly variable; photodissociates above 80 km
Ar	9.34×10^{-3}	Mixed up to 110 km; diffusive separation above
CO_2	3.45×10^{-4}	Slightly variable; mixed up to 100 km; dissociated above
CH_4	1.6×10^{-6}	Mixed in troposphere; dissociated in mesosphere
N_2O	3.5×10^{-7}	Slightly variable at surface; dissociated in stratosphere and mesosphere
CO	7×10^{-8}	Variable photochemical and combustion product
O_3	$\sim 10^{-8}$	Highly variable; photochemical origin
$CFCl_3$ and CF_2Cl_2	1–2 $\times 10^{-10}$	Industrial origin; mixed in troposphere, dissociated in stratosphere

†Fraction of lower tropospheric air.

2.1.2 Vertical Distribution of Pressure and Density

The pressure and density of the Earth's atmosphere can vary substantially in the vertical dimension. It is therefore helpful to define a reference or

"standard"[1] atmosphere that is a representation of the atmosphere as a function of height only. Below an altitude of 100 km, the atmospheric pressure and density are almost always within $\pm 30\%$ of that of the standard atmosphere [1].

Atmospheric density decreases with altitude due to the Earth's gravitational field. If a condition of static equilibrium is assumed, the relationship between density and pressure as a function of altitude may be expressed by the following differential equation:

$$dp = -g\rho\, dz \tag{2.1}$$

where p and ρ are the pressure and density at altitude z measured vertically upward from the surface. The change in gravitational force with altitude is small enough over the relatively short extent of the atmosphere to be ignored. The ideal gas equation $pV = nRT$ can be used to relate the density of an ideal gas of molecular weight M_r to its temperature and pressure:

$$\rho = \frac{M_r p}{RT} \tag{2.2}$$

where R is the gas constant per mole, and T is the temperature (K). Equation (2.1) can then be expressed as

$$\frac{dp}{p} = -\frac{dz}{H} \tag{2.3}$$

which can be integrated to find the pressure p at altitude z:

$$p = p_0 \exp\left\{-\int_0^z \frac{dz}{H}\right\} \tag{2.4}$$

where p_0 is the surface pressure and $H = RT/M_r g$ is known as the *scale height*. The scale height is the increase in altitude necessary to reduce the pressure by a factor of e. In the troposphere, H typically varies between $\sim$6 km at $T = 210$ K to $\sim$8.5 km at $T = 290$ K [2].

2.1.3 Thermal Structure of the Atmosphere

The macroscopic thermal features of the atmosphere were outlined previously. We now examine features that occur on a finer vertical scale, with a focus

1. The horizontal and temporal variations of the Earth's atmosphere do vary substantially on a global and seasonal scale. A wide variety of "standard" atmospheres have been tabulated for various geographical regions and seasonal periods [3].

on the lower troposphere. The bottom 1–2 km of the atmosphere exhibits the greatest thermal variability due to strong surface interactions and diurnal variations. At some latitudes, temperature inversions exist in the lowest 2–3 km of the atmosphere. Above 3 km, there is a systematic decrease of temperature with altitude that can be characterized by an adiabatic lapse rate, as follows. Continuing the assumption of the previous section that the atmosphere is in hydrostatic equilibrium, the first law of thermodynamics can be applied to a unit "parcel" of atmospheric mass:

$$dq = c_v\, dT + p\, dV \tag{2.5}$$

where c_v is the specific heat at constant volume. Provided no heat enters or leaves the parcel (i.e., the process is adiabatic), the quantity dq is zero. Equation (2.5) can then be substituted into the differential form of the ideal gas law to yield:

$$\frac{dT}{dz} = -\frac{g}{c_p} = -\Gamma \tag{2.6}$$

where c_p is the specific heat at constant pressure and Γ is the lapse rate. Equation (2.6) shows that the change in temperature with altitude is constant, for constant c_p and g. Under typical tropospheric conditions, c_p varies slightly with altitude, and the dry adiabatic lapse rate in the troposphere is approximately 10 K/km. If the latent heat released by the condensation of rising moist air is considered, the average lapse is approximately 6.5 K/km.

2.1.4 Cloud Microphysics

Clouds affect the energy balance of the atmosphere through two mechanisms: (1) water cycle changes, including the release of latent heat through condensation and the removal of liquid water through precipitation, and (2) radiation budget changes, including the scattering, absorption, and emission of solar and terrestrial radiation. In Chapter 9, the microphysical properties of clouds (taken here to mean the size and shape of the particles and their volume concentration) will be used to characterize regions of precipitation by examining their interaction with microwave radiation. It is therefore useful to review several of the salient details of the microphysical structure of clouds and precipitation.

The microphysical properties of clouds depend highly on the size, shape, and phase of the water particles. Water droplets are typically smaller than 100 μm and are spherical [1]. The distribution of water droplet concentration (the number of droplets per volume existing in a differential radius range dr) is reasonably approximated by analytic functions. A modified Gamma

Table 2.2
Representative Drop Size Concentrations (N_0), Mean Particle Radius (r_m), and Liquid Water Content (l) for Several Cloud Types (*Source*: [6])

Cloud Type	N_0 (cm^{-3})	r_m (μm)	l (gm^{-3})
Stratus (ocean)	50	10	0.1–0.5
Stratus (land)	300–400	6	0.1–0.5
Fair-weather cumulus	300–500	4	0.3
Maritime cumulus	50	15	0.5
Cumulonimbus	70	20	2.5
Cumulus congestus	60	24	2.0
Altostratus	200–400	5	0.6

distribution is often used for this purpose. Table 2.2 gives average values of the number of particles (N_0), mean droplet radius (r_m), and cloud liquid water density (l) for a variety of clouds. Raindrops are generally nonspherical, resembling oblate spheroids with an aspect ratio (width-to-length ratio) that decreases as the drop size increases. One analytic function that is commonly used to relate raindrop size distributions to rainrate is the Marshall-Palmer distribution [7]. Ice crystals form in a wide variety of sizes and shapes. In addition to simple polyhedron forms, irregular crystals or combinations of simple shapes readily appear in nature.

2.2 Electromagnetic Wave Propagation

The thermal and compositional state of the atmosphere affects both the generation and propagation of electromagnetic (EM) waves. For now, we ignore the source of the EM waves and focus instead on their propagation through a homogeneous, lossless medium.

2.2.1 Maxwell's Equations and the Wave Equation

In a source-free, homogeneous, and isotropic medium with permittivity ϵ and permeability μ, the spatial and temporal variation of electric and magnetic

fields are related according to Maxwell's equations:

$$\nabla \times \vec{\mathbf{E}} = -\mu \frac{\partial}{\partial t} \vec{\mathbf{H}} \tag{2.7}$$

$$\nabla \times \vec{\mathbf{H}} = \epsilon \frac{\partial}{\partial t} \vec{\mathbf{E}} \tag{2.8}$$

$$\nabla \cdot \vec{\mathbf{E}} = 0 \tag{2.9}$$

$$\nabla \cdot \vec{\mathbf{H}} = 0 \tag{2.10}$$

A wave equation can be derived by taking the curl of (2.7) and substituting (2.8). After using the vector identity $\nabla \times (\nabla \times \vec{\mathbf{E}}) = \nabla(\nabla \cdot \vec{\mathbf{E}}) - \nabla^2 \vec{\mathbf{E}}$ and (2.9), we find:

$$\nabla^2 \vec{\mathbf{E}} = \mu\epsilon \frac{\partial^2}{\partial t^2} \vec{\mathbf{E}} \tag{2.11}$$

where the Laplacian operator ∇^2 in a rectangular coordinate system is

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$$

The wave equation (2.11) is a second-order partial differential equation of space and time coordinates x, y, z, and t. A simple solution to the wave equation is

$$\vec{\mathbf{E}}(\vec{\mathbf{r}}, t) = \vec{\mathbf{E}_0} \cos(2\pi\nu t \pm \vec{\mathbf{k}} \cdot \vec{\mathbf{r}}) \tag{2.12}$$

where $\vec{\mathbf{k}} = \hat{\mathbf{x}}k_x + \hat{\mathbf{y}}k_y + \hat{\mathbf{z}}k_z$ and $\vec{\mathbf{r}} = \hat{\mathbf{x}}x + \hat{\mathbf{y}}y + \hat{\mathbf{z}}z$. Equation (2.12) represents two waves propagating in opposite directions in the $\vec{\mathbf{k}}$ direction with temporal phase $2\pi\nu t$ and spatial phase $\vec{\mathbf{k}} \cdot \vec{\mathbf{r}}$. A spectrum of values of frequency (ν) found in atmospheric remote sensing systems is shown in Figure 2.1.

2.2.2 Polarization

The electric field vector of a uniform plane wave traveling in the $+z$ direction must lie in the xy-plane perpendicular to the z-axis. As time progresses, the tip of the electric field vector traces a curve in the xy-plane. It is the shape of this curve (linear, circular, or elliptical) that determines the polarization of the plane wave. If the curve is circular or elliptical, the tip may move in either a clockwise or counterclockwise direction. The interaction of electromagnetic waves with matter often depends (sometimes entirely) on the polarization state of the wave. Some remote sensing systems exploit polarization dependence to extract information about a polarized target. For example, polarimetric microwave measurements of the ocean surface reveal information about sea surface wind speed and direction due to the polarimetric signature of the resulting ocean waves.

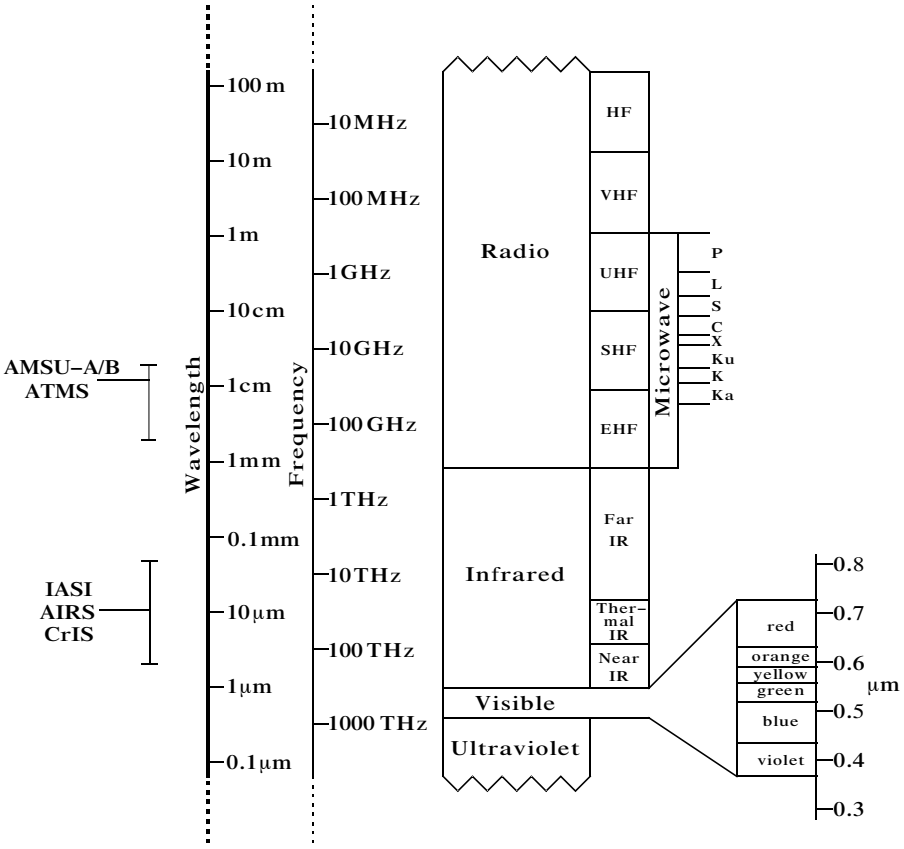


Figure 2.1 The electromagnetic spectrum. The diagram shows those parts of the electromagnetic spectrum that are important in remote sensing, together with the conventional names of the various regions of the spectrum. Also shown are wavelength regions of sensors mentioned in this book: AIRS, AMSU, IASI, CrIS, and ATMS. (*After*: [1].)

2.2.2.1 Stokes Parameters

A mathematical discussion of polarization can be facilitated by decomposing the $\vec{\mathbf{E}}$ vector into two components perpendicular to the direction of propagation, $\vec{\mathbf{k}}$, for a fixed point in space:

$$\vec{\mathbf{E}} = \hat{\mathbf{h}} E_h + \hat{\mathbf{v}} E_v = \hat{\mathbf{h}} e_h \cos(2\pi\nu - \varphi_h) + \hat{\mathbf{v}} e_v \cos(2\pi\nu - \varphi_v) \qquad (2.13)$$

where $\vec{\mathbf{k}}$, $\hat{\mathbf{h}}$, and $\hat{\mathbf{v}}$ form an orthogonal system mutually perpendicular to one another. The four Stokes parameters may then be defined as follows:

$$I = \frac{1}{\eta}(e_h^2 + e_v^2) \tag{2.14}$$

$$Q = \frac{1}{\eta}(e_h^2 - e_v^2) \tag{2.15}$$

$$U = \frac{2}{\eta}e_h e_v \cos(\varphi) \tag{2.16}$$

$$V = \frac{2}{\eta}e_h e_v \sin(\varphi) \tag{2.17}$$

where φ is the phase difference $\varphi_h - \varphi_v$ and $\eta = \sqrt{\mu/\epsilon}$ is the characteristic impedance. In practice, it is often easier to measure the Stokes parameter rather than measure e_h, e_v, and φ directly. The four Stokes parameter are related as follows:

$$I^2 = Q^2 + U^2 + V^2 \tag{2.18}$$

2.2.3 Reflection and Transmission at a Planar Boundary

Electromagnetic radiation from the sun or the cosmic background can reflect off cloud tops and the surface of the Earth. The characterization of the transmitted and reflected components of radiation is necessary to develop cloud and surface models. Consider a linearly polarized plane wave propagating in free space along direction $\vec{\mathbf{k}_i}$ that is incident upon a planar dielectric material with index of refraction $n = c\sqrt{\mu\epsilon}$ at an incidence angle of θ_i. The electric fields for the incident, reflected, and transmitted waves can be expressed as (assuming a TE wave propagating in the xz-plane):

$$\begin{aligned} \vec{\mathbf{E}_i} &= \hat{\mathbf{y}} E_i e^{-j\vec{\mathbf{k}_i}\cdot\vec{\mathbf{r}}} \\ \vec{\mathbf{E}_r} &= \hat{\mathbf{y}} \Gamma E_i e^{-j\vec{\mathbf{k}_r}\cdot\vec{\mathbf{r}}} \\ \vec{\mathbf{E}_t} &= \hat{\mathbf{y}} \mathcal{T} E_i e^{-j\vec{\mathbf{k}_t}\cdot\vec{\mathbf{r}}} \end{aligned} \tag{2.19}$$

where Γ and $\mathcal{T}$ are the complex reflection and transmission coefficients, respectively. The tangential components of the net electric field must vanish at the boundary, requiring the tangential components of all three $\vec{\mathbf{k}}$ vectors to be equal along the boundary. The tangential components of the $\vec{\mathbf{k}}$ vectors can be expressed in terms of the angles of incidence, reflection, and transmission to yield

$$k_i \sin\theta_i = k_r \sin\theta_r = k_t \sin\theta_t \tag{2.20}$$

where $k_i = k_r = \omega\sqrt{\mu_i \epsilon_i}$ is the magnitude of the propagation vectors $\vec{\mathbf{k}_i}$ and $\vec{\mathbf{k}_r}$. The magnitude of the transmitted wave vector is $k_t = \omega\sqrt{\mu_t \epsilon_t}$, which is in general not equal to k_i. Substitution into (2.20) gives the reflection law

$$\theta_r = \theta_i \tag{2.21}$$

and Snell's law

$$\frac{\sin\theta_t}{\sin\theta_i} = \frac{k_i}{k_t} = \frac{n_i}{n_t} \tag{2.22}$$

Given $\vec{\mathbf{k}_r}$ and $\vec{\mathbf{k}_t}$, the complex reflection and transmission coefficients can be found by supplementing the boundary condition for continuity of the electric field with a similar equation for the tangential magnetic field. For TE waves,

$$\Gamma_{TE} = \frac{\eta_t \cos\theta_i - \eta_i \cos\theta_t}{\eta_t \cos\theta_i + \eta_i \cos\theta_t} \tag{2.23}$$

$$\mathcal{T}_{TE} = \frac{2\eta_t \cos\theta_i}{\eta_t \cos\theta_i + \eta_i \cos\theta_t} \tag{2.24}$$

and for TM waves,

$$\Gamma_{TM} = \frac{\eta_i \cos\theta_i - \eta_t \cos\theta_t}{\eta_i \cos\theta_i + \eta_t \cos\theta_t} \tag{2.25}$$

$$\mathcal{T}_{TM} = \frac{2\eta_i \cos\theta_i}{\eta_i \cos\theta_i + \eta_t \cos\theta_t} \tag{2.26}$$

As an important consequence of the preceding equations, unpolarized radiation incident upon a planar dielectric surface can become partially or totally polarized on reflection. For example, a portion of the unpolarized microwave radiation emitted by the atmosphere is reflected by the ocean surface and another portion is absorbed and re-emitted by the ocean surface. However, the TE and TM components of the emitted radiation are different when viewed from oblique angles, a characteristic that can be used to discriminate surface water from rainfall [1].

2.3 Absorption of Electromagnetic Waves by Atmospheric Gases

A knowledge of the mechanisms of electromagnetic radiation interaction with matter, as well as some of the fundamental properties of matter itself, is necessary to infer and interpret information about the atmosphere. In the following two subsections, the interactions are described on a microscopic (molecular) and macroscopic (particle) level.

2.3.1 Mechanisms of Molecular Absorption

The total internal energy of an isolated molecule consists of three types of energy states,

$$\mathcal{E} = \mathcal{E}_e + \mathcal{E}_v + \mathcal{E}_r \tag{2.27}$$

where $\mathcal{E}_e$ = electronic energy, $\mathcal{E}_v$ = vibrational energy, and $\mathcal{E}_r$ = rotational energy. Rotational energy is associated with rotational motions of the atoms of the molecule about its center of mass, and vibrational energy is associated with vibrational motions of the atoms about their equilibrium positions. Radiation is absorbed or emitted when a transition takes place from one energy state to another. The frequency (ν) of the absorbed (or emitted) photon is given by the Bohr frequency condition,

$$\nu = \frac{\mathcal{E}_h - \mathcal{E}_l}{h} \tag{2.28}$$

where h is Planck's constant and $\mathcal{E}_h$ and $\mathcal{E}_l$ are the internal energies of the higher and lower molecular states, respectively. The absorption spectrum due to a single transition is called an absorption line. Absorption by molecules in the mid- and near-infrared occur by vibration (although a mixture of vibrations and rotations are usually induced at these frequencies). In the microwave and far-infrared, rotational transitions are the dominant mechanism of energy transfer.

2.3.2 Line Shapes

Based on (2.28), the absorption (or emission) spectrum of an isolated, unperturbed, stationary molecule consists of sharply defined frequency lines corresponding to transitions between quantized energy levels of the molecule. Atmospheric gases, however, consist of molecules that are in constant motion, interacting and colliding with one another. These disturbances cause the absorption lines to broaden. The two most important sources of line broadening are Doppler (thermal) broadening and pressure (collision) broadening, which is dominant for most frequencies up to an altitude of approximately 40 km [8].

2.3.3 Absorption Coefficients and Transmission Functions

Line shape $f(\nu - \nu_0)$, line position (ν_0), and line strength (S) mathematically define the absorption coefficient:

$$\kappa_\nu = S\, f(\nu - \nu_0) \tag{2.29}$$

The line strength of a specific atmospheric gas is governed by the number of absorbing molecules of that gas per unit volume, the temperature of the gas, and the molecular parameters associated with that transition.

Absorption of radiation by gases in the Earth's atmosphere is described in terms of transmission functions (or simply, transmittance). Lambert's law states that the change in radiance intensity along a path ds is proportional to the amount of matter along the path:

$$dR_\nu = -\kappa_\nu R_\nu \, ds \tag{2.30}$$

where κ_ν is the volume absorption coefficient. Integration of Lambert's law along the path connected by s_1 and s_2 yields

$$R_\nu(s_2) = \mathcal{T}_\nu(s_1, s_2) R_\nu(s_1) \tag{2.31}$$

where $\mathcal{T}_\nu(s_1, s_2)$ is the monochromatic transmittance defined as

$$\mathcal{T}_\nu(s_1, s_2) = e^{-\int_{s_1}^{s_2} \kappa_\nu \, ds} \tag{2.32}$$

The optical path (or thickness)[2] between s_1 and s_2 is defined as

$$\tau_\nu(s_1, s_2) = \int_{s_1}^{s_2} \kappa_\nu \, ds \tag{2.33}$$

The absorption coefficient, transmittance, and optical path form the mathematical basis for the subject of Section 2.5.2: radiative transfer. In practice, these quantities are not monochromatic, but band-averaged over some spectral response function of the instrument.

2.3.4 The Atmospheric Absorption Spectra

The atmospheric absorption spectrum for microwave frequencies is shown in Figure 2.2. Notable features include the water vapor absorption lines centered at 22.235, 183.31, and 325.15 GHz (lines at 380.20 and 448.00 GHz are difficult to identify on the plot) and oxygen absorption lines near 60, 118.75, 368.50, 424.76, and 487.25 GHz. The atmospheric absorption spectrum for infrared wavelengths between 3 and 15 μm is shown in Figure 2.3. Notable features include the water vapor absorption lines near 6–7 μm, ozone absorption lines near 10 μm, and carbon dioxide absorption lines near 4.3–4.6 μm and 13–15 μm.

2. The related quantities optical depth and opacity will be defined later.

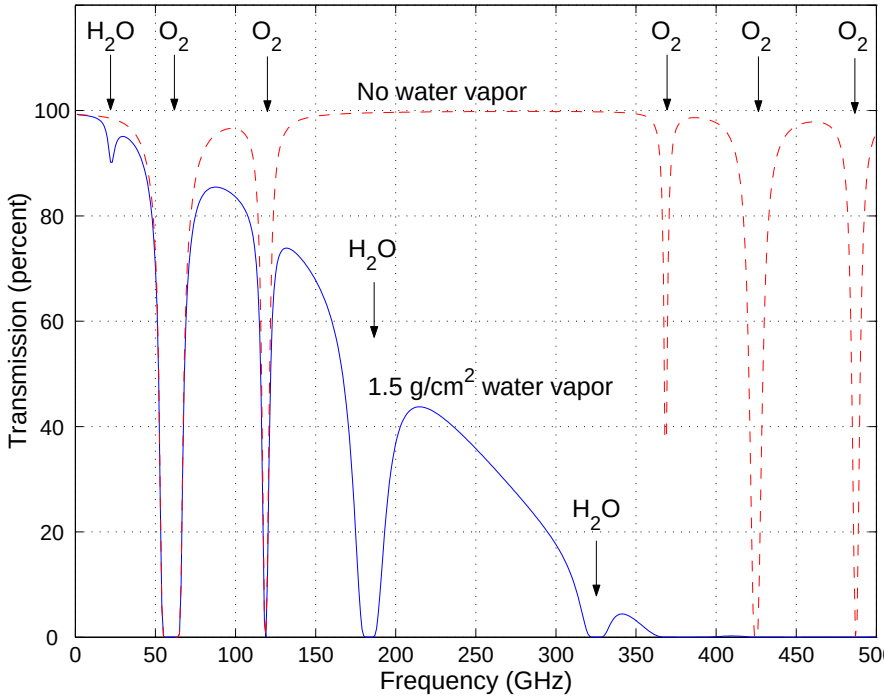


Figure 2.2 The microwave absorption spectrum. Two calculations for the percent transmission (nadir view) using the 1976 Standard Atmosphere are shown, one assuming no water vapor and one assuming 1.5 g/cm^2.

2.4 Scattering of Electromagnetic Waves by Atmospheric Particles

In addition to the molecular absorption mechanisms discussed earlier, electromagnetic waves are also scattered and absorbed by much larger particles often found in the atmosphere, such as cloud water droplets, raindrops, or even dust. The scattering of electromagnetic waves upon interaction with atmospheric particles provides a tool that can be used to help retrieve many microphysical parameters related to clouds and precipitation.

2.4.1 Mie Scattering

A suspended particle of geometrical cross-section A will absorb a fraction of incident power and will also scatter incident power in all directions. The ratio of absorbed power P_a (W) to incident power density S (W/m^2) is known as

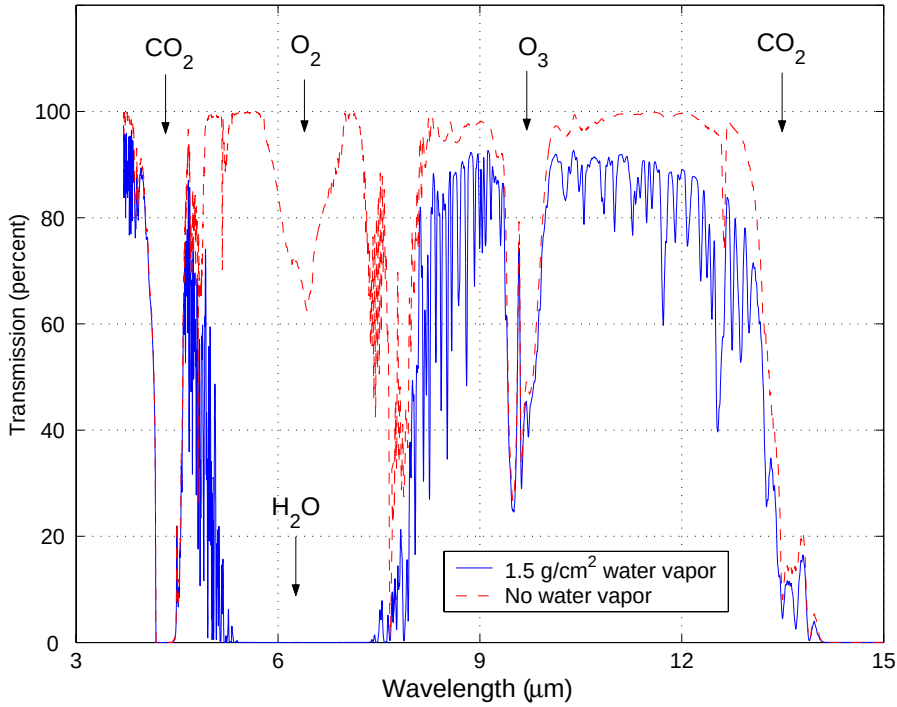


Figure 2.3 The thermal infrared absorption spectrum. Two calculations for the percent transmission (nadir view) using the 1976 Standard Atmosphere are shown, one assuming no water vapor and one assuming 1.5 g/cm^2.

the absorption cross-section

$$C_a = \frac{P_a}{S} \tag{2.34}$$

and the ratio of C_a to the physical cross-section A is known as the efficiency factor Q_a. For a spherical particle of radius r, $A = \pi r^2$ and therefore

$$Q_a = \frac{C_a}{\pi r^2} \tag{2.35}$$

Analogous quantities for scattering, that is, the scattering cross-section C_s and the scattering efficiency Q_s, are defined as

$$C_s = \frac{P_s}{S} \tag{2.36}$$

$$Q_s = \frac{C_s}{\pi r^2} \tag{2.37}$$

The total power lost due to absorption and scattering (together known as the extinction) is $P_a + P_s$ and the resulting extinction cross-section C_e and efficiency Q_e are

$$C_e = C_a + C_s \tag{2.38a}$$

$$Q_e = Q_a + Q_s \tag{2.38b}$$

The solution for the scattering and absorption of electromagnetic waves in free space by a dielectric sphere of radius r was formulated by Mie in terms of the "size parameter"

$$\chi = \frac{2\pi r}{\lambda} \tag{2.39}$$

and

$$n = \sqrt{\epsilon_c} \tag{2.40}$$

where λ is the wavelength of the incident wave, n is the complex refractive index of the particle and ϵ_c is the corresponding complex dielectric constant. Mie's expressions for the scattering and extinction efficiencies of the sphere are given by

$$Q_s(n,\chi) = \frac{2}{\chi^2}\sum_{m=1}^{\infty}(2m+1)(|a_m|^2 + |b_m|^2) \tag{2.41a}$$

$$Q_e(n,\chi) = \frac{2}{\chi^2}\sum_{m=1}^{\infty}(2m+1)\mathrm{Re}\{a_m + b_m\} \tag{2.41b}$$

where a_m and b_m are known as the Mie coefficients

$$a_m = -\frac{j_m(n\chi)[\chi j_m(\chi)]' - j_m(\chi)[n\chi j_m(n\chi)]'}{j_m(n\chi)[\chi h_m(\chi)]' - h_m(\chi)[n\chi j_m(n\chi)]'} \tag{2.42a}$$

$$b_m = -\frac{j_m(\chi)[n\chi j_m(n\chi)]' - n^2 j_m(n\chi)[\chi j_m(\chi)]'}{h_m(\chi)[n\chi j_m(n\chi)]' - n^2 j_m(n\chi)[\chi h_m(\chi)]'} \tag{2.42b}$$

where $j_m(\cdot)$ and $h_m(\cdot)$ are the spherical Bessel and Hankel functions of the first kind, and the $(\cdot)'$ operator denotes the complex conjugation.

2.4.2 The Rayleigh Approximation

The Mie expressions for Q_s and Q_e can be approximated with negligible error if the particle size is much smaller than the wavelength of the incident wave ($|n\chi| \ll 1$). The Rayleigh approximation is obtained by retaining only the most significant terms in the series expansion:

$$Q_s = \frac{8}{3}\chi^4 |K|^2 \tag{2.43}$$

$$Q_e = 4\chi \text{Im}\{-K\} + \frac{8}{3}\chi^4 |K|^2 \tag{2.44}$$

and

$$Q_a = 4\chi \text{Im}\{-K\} \tag{2.45}$$

where K is a complex quantity defined in terms of the complex index of refraction n

$$K = \frac{n^2 - 1}{n^2 + 2} = \frac{\epsilon_c - 1}{\epsilon_c + 2} \tag{2.46}$$

Note that in the Rayleigh limit the scattering efficiency scales as the fourth power of frequency, whereas the absorption efficiency scales linearly with frequency, for a fixed particle size and a frequency-independent index of refraction. For water, the index of refraction is frequency-dependent, and the absorption efficiency scales as frequency squared (for frequencies below 100 GHz or so) when this dependence is included.

2.4.3 Comparison of Scattering and Absorption by Hydrometeors

Figure 2.4 shows scattering and absorption contributions of water spheres, both in the liquid and ice phases. Deirmendjian's recursive procedure [9] was used to calculate the Mie coefficients; 80 terms were used to approximate the series. For liquid droplets, absorption is dominant in the Rayleigh region and scattering is dominant in the Mie region. For ice, scattering is dominant for all but the lowest microwave frequencies. The frequency dependence of scattering and absorption can be used to retrieve information about the particle size distributions of clouds, and the related quantity, rainrate. The distribution and type of hydrometeors found in typical clouds vary widely, and monodispersive models are inadequate. More complicated modeling is beyond the scope of this book; an excellent discussion can be found in [10].

2.5 Radiative Transfer in a Nonscattering Planar-Stratified Atmosphere

A sensor high above the Earth's surface receives emission from the Earth and its atmosphere, along with any reflected components of solar and cosmic background radiation. Measurements of this emission allow the retrieval of many atmospheric parameters, including the temperature and water vapor profile, the amount of cloud liquid water, rainrates, and sea surface temperatures.

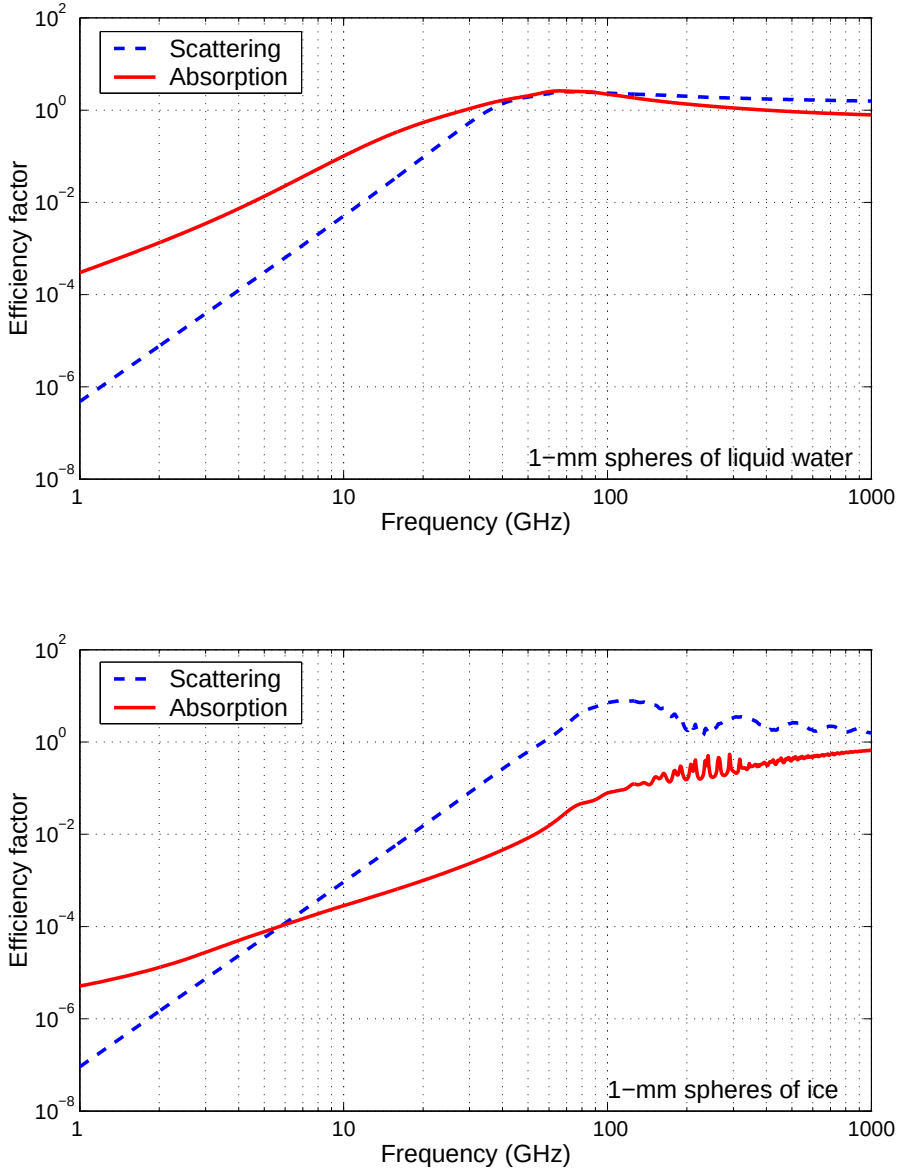


Figure 2.4 Scattering and absorption efficiency for water spheres with 1-mm radius. Liquid water spheres (273 K) are shown in the top plot and ice spheres (266 K) are shown in the bottom plot.

2.5.1 Equilibrium Radiation: Planck and Kirchhoff's Laws

The solution to the simple differential equation presented by Lambert's law (2.30) is referred to as Beer's law

$$R_\nu(s'') = R_\nu(s')e^{-\int_{s'}^{s''} \kappa_\nu(s)\,ds} \tag{2.47}$$

In addition to absorption of radiation by the gas contained within path s' to s'', Kirchhoff's law states that if in thermal equilibrium, the gas also emits radiation in an amount proportional to the absorption coefficient κ_ν:

$$R_\nu^{\text{emission}} = \kappa_\nu \mathcal{J}_\nu(T) \tag{2.48}$$

where $\mathcal{J}_\nu(T)$ is the radiation intensity produced (at each of two orthogonal polarizations) by a blackbody at temperature T and frequency ν:

$$\mathcal{J}_\nu(T) = \frac{h\nu^3}{c^2}\frac{1}{e^{h\nu/kT}-1} \ \text{W}\cdot\text{m}^{-2}\cdot\text{ster}^{-1}\cdot\text{Hz}^{-1} \tag{2.49}$$

The Planck equation exhibits a nonlinear relationship between intensity and temperature. The degree of the nonlinearity is frequency-dependent, and is shown in Figure 2.5. The nonlinearity is most severe at the higher frequencies (shorter wavelengths) and almost nonexistent at the microwave frequencies. The approximation of the Planck radiance by the linear Taylor series term is called the Rayleigh-Jeans (RJ) approximation, and the microwave brightness temperature is defined as the scaled intensity:

$$B_\nu = \frac{c^2}{2\nu^2 k}R_\nu \tag{2.50}$$

Note that if a radiometer is calibrated against a blackbody and all departures from the Rayleigh-Jeans law are ignored, brightness temperature is effectively redefined as

$$B_\nu = \frac{c^2}{2\nu^2 k}R_\nu + \frac{hv}{2k} \tag{2.51}$$

and accuracy is better than 0.1 K for frequencies up to 300 GHz and terrestrial temperatures. When extremely cold temperatures are encountered (e.g., the cosmic background) corrections to the RJ approximation are needed.

2.5.2 Radiative Transfer Due to Emission and Absorption

The net change in radiation along ds due to the combination of emission and absorption is

$$dR_\nu = dR_\nu^{\text{emission}} + dR_\nu^{\text{absorption}} \tag{2.52}$$

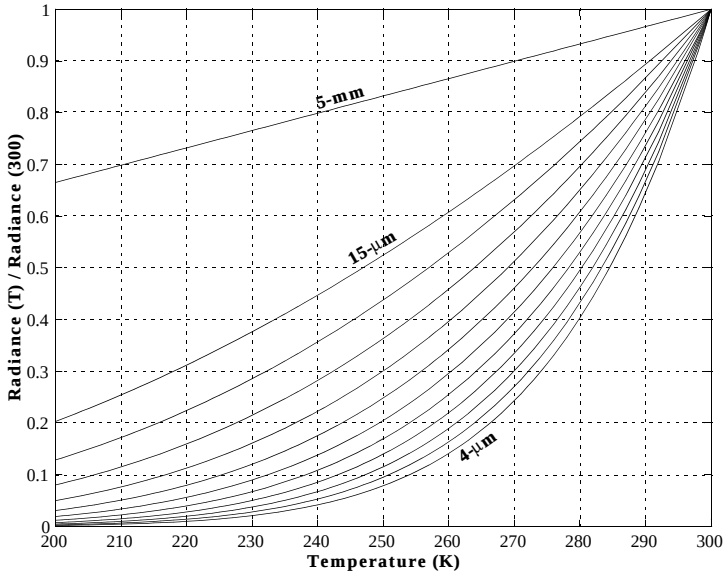


Figure 2.5 Nonlinearity of the Planck function as a function of wavelength.

Substitution of (2.30) and (2.48) into (2.52) yields the Schwartzchild equation

$$\frac{dR_\nu}{ds} = -\kappa_\nu[R_\nu - \mathcal{J}_\nu(T)] \tag{2.53}$$

which mathematically describes how radiation is transferred from one layer to another layer as a result of absorption and emission. The intensity of radiation leaving the path is therefore a function of both the absorber along the path and the temperature along the path. Passive (emission-based) sounding of constituent concentration and temperature is based upon this principle.

2.5.3 Integral Form of the Radiative Transfer Equation

Differentiation of (2.33) gives

$$d\tau_\nu(s) = -\kappa_\nu(s)\, ds \tag{2.54}$$

where we adopt the convention that τ increases from zero downward from the top of the atmosphere to a maximum value τ^* (the opacity of the atmosphere)

at the surface. Multiplying both sides of (2.53) by $e^{-\tau_\nu(s)}$ and combining terms gives

$$\frac{dR_\nu e^{-\tau_\nu(s)}}{d\tau_\nu} = -\mathcal{J}_\nu e^{-\tau_\nu(s)} \tag{2.55}$$

which upon integration from path s' to s'' yields

$$R_\nu(s'')e^{-\tau_\nu(s'')} - R_\nu(s')e^{-\tau_\nu(s')} = \int_{\tau(s'')}^{\tau(s')} \mathcal{J}_\nu(s)e^{-\tau_\nu(s)}\, d\tau(s) \tag{2.56}$$

Equation (2.56) can be rearranged into the integral form of the radiative transfer equation as follows:

$$R_\nu(s'') = R_\nu(s')e^{-[\tau_\nu(s')-\tau_\nu(s'')]} + \int_{s'}^{s''} \mathcal{J}_\nu(s)e^{-[\tau_\nu(s)-\tau_\nu(s'')]}\, d\tau_\nu(s) \tag{2.57}$$

The equivalent relation in terms of the absorption coefficient κ_ν is

$$R_\nu(s'') = R_\nu(s')e^{-\int_{s'}^{s''} \kappa_\nu(s)ds} + \int_{s'}^{s''} \kappa_\nu(s)\mathcal{J}_\nu(s)e^{-\int_{s}^{s''} \kappa_\nu(\sigma)\, d\sigma}\, ds \tag{2.58}$$

The angular properties of emission have thus far been neglected, but can easily be included for the case of a horizontally homogeneous vertically stratified atmosphere by noting that an angular tilt of θ results in an increase in the path length by a factor of $\sec\theta$ (see Figure 2.6). Optical depth is related to optical path as follows:

$$\tau(s) = \tau(z)\sec(\theta) \tag{2.59}$$

After including the angular terms, the final form of the radiative transfer equation describing the radiation intensity observed at altitude L and viewing angle θ can be formulated by including reflected atmospheric and cosmic contributions and the radiance emitted by the surface:

$$\begin{aligned} R_\nu(L) &= \int_0^L \kappa_\nu(z)\mathcal{J}_\nu[T(z)]e^{-\int_z^L \sec\theta\kappa_\nu(z')\, dz'} \sec\theta\, dz \\ &+ \rho_\nu e^{-\tau^* \sec\theta} \int_0^L \kappa_\nu(z)\mathcal{J}_\nu[T(z)]e^{-\int_0^z \sec\theta\kappa_\nu(z')\, dz'} \sec\theta\, dz \\ &+ \rho_\nu e^{-2\tau^* \sec\theta}\mathcal{J}_\nu(T_c) \\ &+ \varepsilon_\nu e^{-\tau^* \sec\theta}\mathcal{J}_\nu(T_s) \end{aligned} \tag{2.60}$$

where ε_ν is the surface emissivity, ρ_ν is the surface reflectivity, T_s is the surface temperature, and T_c is the cosmic background temperature (2.736 ± 0.017 K).

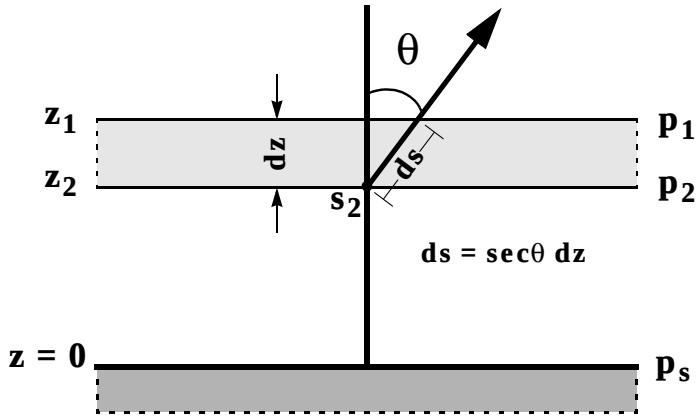


Figure 2.6 Geometry of the planar-stratified atmospheric radiative transfer equation.

2.5.4 Weighting Function

The first term in (2.60) can be recast in terms of the transmittance function $\mathcal{T}_\nu$(z):

$$R_\nu(L) = \int_0^L \mathcal{J}_\nu[T(z)] \left(\frac{d\mathcal{T}_\nu(z)}{dz} \right) dz \tag{2.61}$$

The derivative of the transmittance function with respect to altitude is often called the weighting function

$$W_\nu(z) \triangleq \frac{d\mathcal{T}_\nu(z)}{dz} \tag{2.62}$$

and gives the relative contribution of the radiance emanating from each altitude. Note that the Planck radiances are weighted, not the temperature profile. It is sometimes useful to define a temperature weighting function, where the temperature profile is weighted directly. One approach is to express the radiance intensity $R_\nu(L)$ in terms of a "blackbody-equivalent" brightness temperature $T_{B,\nu}(L)$ (the temperature of a blackbody that produces a radiance equivalent to $R_\nu(L)$ – note that $T_{B,\nu}(L) \neq B_\nu$) and linearize about a nominal temperature profile $T_0(z)$ and corresponding radiance $R_{0,\nu}(L)$.

2.5.4.1 Temperature Weighting Function

For a particular frequency, the blackbody-equivalent radiance may be written as follows:

$$T_{B,\nu}(L) = \mathcal{J}_\nu^{-1}(W_\nu(\mathcal{J}_\nu(T_z))) \tag{2.63}$$

where $\mathcal{J}_\nu(\cdot)$ is the Planck function, $W_\nu(\cdot)$ is the integration against the weighting function, and $\mathcal{J}_\nu^{-1}(\cdot)$ is the inverse Planck function. The first-order Taylor series approximation of $\mathcal{J}_\nu^{-1}(W_\nu(\mathcal{J}_\nu(\cdot)))$ is then

$$\begin{aligned} R_\nu(L) &= R_{0,\nu}(L) + \frac{d\mathcal{J}_\nu^{-1}}{dW_\nu}\frac{dW_\nu}{d\mathcal{J}_\nu}\frac{d\mathcal{J}_\nu}{dT}[T(z) - T_0(z)] & (2.64)\\ &= W_{T,\nu}(z)[T(z) - T_0(z)] + R_{0,\nu}(L) & (2.65)\end{aligned}$$

where $W_{T,\nu}(z)$ is defined as the temperature weighting function:

$$W_{T,\nu}(z) \triangleq \left.\frac{d\mathcal{J}_\nu^{-1}}{dW_\nu}\frac{dW_\nu}{d\mathcal{J}_\nu}\frac{d\mathcal{J}_\nu}{dT}\right|_{T_0(z)} \tag{2.66}$$

The Planck radiance function can be linearized about some nominal temperature profile T_0, and a temperature weighting function (sometimes called an incremental weighting function) can be defined:

$$R_\nu(L) = \int_0^L W_\nu(z)\mathcal{J}_\nu[T_0(z)]dz + \int_0^L [T(z) - T_0(z)]W_{T,\nu}(z)dz \tag{2.67}$$

where the temperature weighting function is defined as

$$W_{T,\nu}(z) = \frac{d\mathcal{J}_\nu[T_0(z)]}{dT}W_\nu(z) = \frac{hc\nu}{k}\frac{\mathcal{J}_\nu[T_0(z)]}{T_0^2(z)}W_\nu(z) \tag{2.68}$$

The difference between the Planck weighting function and the temperature weighting function can be significant for short-wavelength channels, as shown in Figure 2.7. The temperature weighting functions are sharper and peak lower in the atmosphere. The RMS errors (in units of blackbody-equivalent brightness temperature) resulting from the use of the first-order approximation given by (2.67) (and assuming the weighting functions are independent of atmospheric parameters) over a representative set of atmospheric profiles $T(z)$ (with $T_0(z) = E[T(z)]$) are shown in Figure 2.8 for the channel set of the NASA Atmospheric Infrared Sounder (AIRS) launched on the Aqua satellite in 2002 [11]. The two dominant sources of error are the nonlinearity of the Planck function (most evident in the short-wavelength channels) and the nonlinearity of the atmospheric transmittance (most evident

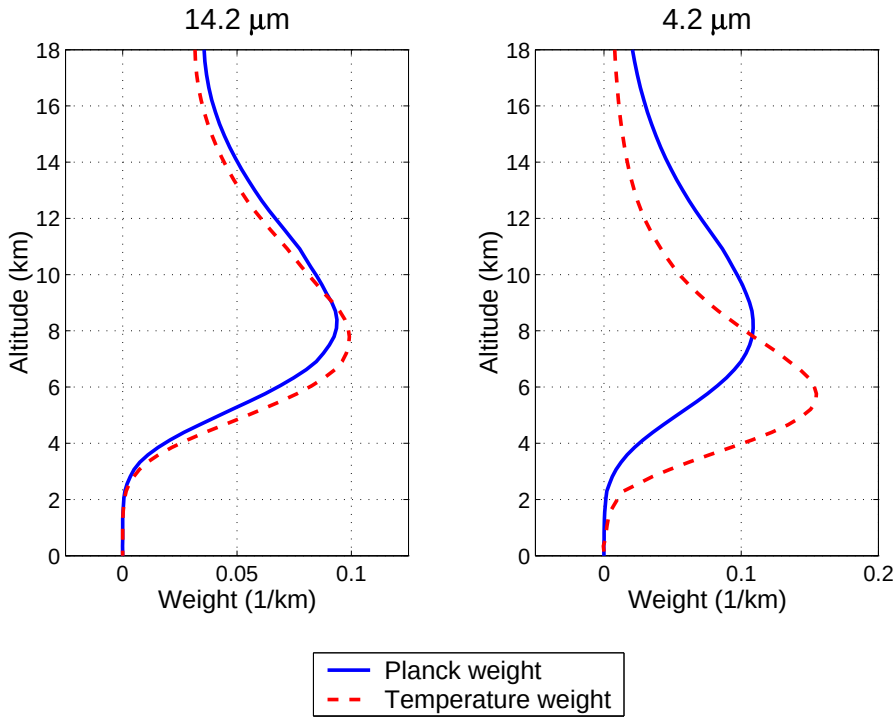


Figure 2.7 The Planck radiance weighting function and the temperature weighting function for two infrared channels.

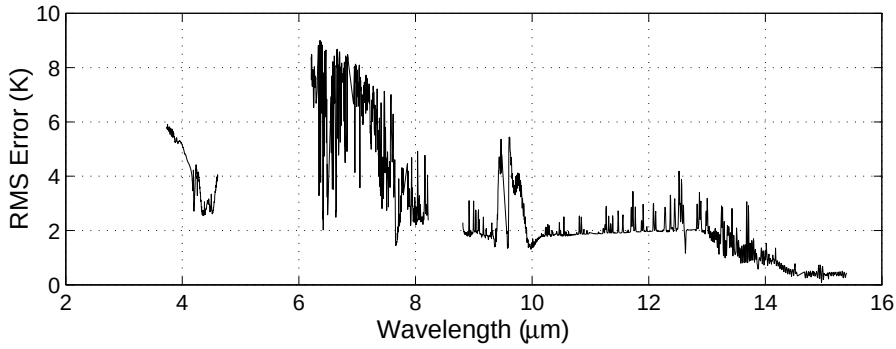


Figure 2.8 AIRS RMS radiance error due to first-order Planck approximation.

in the water vapor channels). Temperature weighting functions are almost never used directly to evaluate the radiative transfer equation because of the large errors introduced by the exclusion of nonlinearities. Nevertheless, the temperature weighting functions provide a useful characterization of the way different layers of the atmosphere at different temperatures contribute to the radiation emitted at the top of the atmosphere.

Returning to the special case of microwave frequencies and terrestrial temperatures (i.e., the Rayleigh-Jeans regime), (2.60) can be expressed in terms of the physical temperature profile $T(z)$ and the brightness temperature B_ν:

$$\begin{aligned} B_\nu(L) &= \int_0^L \kappa_\nu(z)T(z)e^{-\int_z^L \sec\theta\kappa_\nu(z')\,dz'} \sec\theta\,dz \\ &+ \rho_\nu e^{-\tau^* \sec\theta} \int_0^L \kappa_\nu(z)T(z)e^{-\int_0^z \sec\theta\kappa_\nu(z')\,dz'} \sec\theta\,dz \\ &+ \rho_\nu e^{-2\tau^* \sec\theta}\widetilde{T}_c \\ &+ \varepsilon_\nu e^{-\tau^* \sec\theta}T_s \end{aligned} \tag{2.69}$$

where $\widetilde{T}_c$ is the corrected cosmic background temperature

$$\widetilde{T}_c = \frac{h\nu}{2k}\frac{(e^{h\nu/kT_c}+1)}{(e^{h\nu/kT_c}-1)} \tag{2.70}$$

Note that in the microwave case, $W_{T,\nu}(z) = W_\nu(z)$.

2.6 Passive Spectrometer Systems

Measurement of the Earth's radiation at a spectral resolution high enough to study details of molecular absorption bands is achieved using spectrometer sensor systems. The terms "multispectral," "hyperspectral," and more recently "ultraspectral" have been used to denote spectrometer systems that measure radiance in tens, hundreds (or sometimes a few thousand), and thousands of spectral channels, respectively. The chapter concludes with a brief discussion of various spectrometer technologies, with a focus on performance advantages, disadvantages, and engineering trade-offs that must be considered when selecting an instrument to carry out a given remote sensing task. The concepts presented here are rudimentary; for more details, the reader is referred to [12–14].

2.6.1 Optical Spectrometers

For the purpose of an elementary discussion, optical spectrometers can be grouped into three system types: prism dispersion, diffraction grating, and radiation interference. The systems differ primarily in the mechanisms used to separate incident radiation into discrete spectral components.

2.6.1.1 Prism Dispersion Systems

A prism spectrometer produces radiance spectra by passing the incident radiation through a prism. The prism causes the radiation to disperse (bend) by a frequency-dependent angle. The degree to which the radiation is dispersed is determined by the refractive index of the prism. The spectra are usually detected either by sweeping the dispersed radiation across a fixed detector (for example, a photomultiplier), or sweeping the detector across the radiation field. The spectral resolution achieved by prism spectrometers is relatively coarse, and they are therefore used typically in imaging systems. The calibration of prism systems is also problematic because of the frequency dependence of the index of refraction of the prism.

2.6.1.2 Diffraction Grating Systems

A diffraction grating disperses radiation into spectra through angular-dependent interference patterns that result when radiation is passed through a dense array of small slits. Diffraction grating systems operate by either transmitting radiation through or reflecting radiation from a series of closely spaced parallel lines etched on plastic film (for transmission) or a metallic surface (for reflection). Transmission gratings generally perform poorly in comparison with reflection gratings, which are used in high-performance space spectrometers. The spectral resolving power of grating spectrometers typically exceeds that of prism spectrometers by an order of magnitude, at the expense of increased instrumentation complexity. The AIRS instrument, which is discussed in Chapter 10, is a diffraction grating spectrometer.

2.6.1.3 Interferometer Systems

The interferometer spectrometer operates quite differently than the prism or grating spectrometer in that interference effects instead of dispersion effects are used to separate spectra. One of the simplest types of interferometers is the Michelson interferometer, which splits incoming radiation into two beams of unequal length by a partially silvered plate (beam splitter) and later recombines the beams with a known path difference. The path difference

can be varied uniformly by moving a mirror at a constant speed, causing the two beams to move in and out of phase at the detector. The intensity of the resulting waveform (termed the interferogram) is related to the spectral intensity of the incident radiation by the Fourier transform. The interferogram is the autocorrelation function of the optical signal. The performance of the interferometer spectrometer relative to the grating spectrometer depends on a number of factors, including the nature (e.g., mechanical and electrical) and origin (e.g., photon and thermal) of system noise. Examples of interferometer sounding systems include the NPOESS Aircraft Sounder Testbed–Infrared (NAST-I) [15], the Cross-track Infrared Sounder (CrIS) [16], and the Infrared Atmospheric Sounding Interferometer (IASI) [17].

2.6.2 Microwave Spectrometers

Microwave and optical spectrometer systems are conceptually similar. Perhaps the most pernicious source of error in microwave spectrometer systems is the instability of the receiver, and the primary difference among microwave systems is the way in which receiver sensitivity is compromised for receiver stability. Three types of microwave spectrometers[3] are now discussed.

2.6.2.1 Total Power Spectrometer

The simplest type of microwave spectrometer measures the power of incident radiation over a collection of bandwidths B_n, integrated over a time τ. The RMS sensitivity of the measurement at any given channel is a function of the receiver noise (T_R, expressed in units of temperature), the incident radiation (T_A, expressed in units of temperature), the bandwidth (B_n, Hz), and the integration time[4] (τ, sec):

$$\Delta T_{rms} = \frac{T_R + T_A}{\sqrt{B_n \tau}} \tag{2.71}$$

Equation (2.71) assumes that the receiver gain is perfectly stable. Fluctuations in receiver gain reduce the system sensitivity as follows:

$$\Delta T_{rms} = (T_R + T_A)\sqrt{\frac{1}{B_n \tau} + \left(\frac{\Delta G}{G}\right)^2} \tag{2.72}$$

3. The term "radiometer" is used when incident electromagnetic power is measured across a given frequency band. The term "spectrometer" is used when power across several frequency bands (or channels) is measured.

4. It is assumed for the purposes of this discussion that the detector signal is convolved with a boxcar of length τ. Other averaging kernels may be used, with trade-offs between sensitivity and memory effects.

where $\Delta G/G$ is the fractional receiver gain drift. It is not uncommon for the gain drift component to dominate the noise expressed in (2.72). Examples of total-power microwave spectrometer sounding systems include the NPOESS Aircraft Sounder Testbed–Microwave (NAST-M) [18], the Advanced Microwave Sounding Unit (AMSU) [19], and the Advanced Technology Microwave Sounder (ATMS) [20].

2.6.2.2 Dicke Spectrometer

The Dicke spectrometer is essentially a total-power spectrometer with two additional features: (1) a switch used to modulate the receiver input signal, and (2) a synchronous detector, placed between the detector and integrator. The modulation consists of periodically switching the receiver input between the antenna and a reference source (T_{ref}) at a rate higher than the highest significant spectral component of the gain variation. If the noise temperature of the reference source is close to the antenna temperature T_A, the system sensitivity of the Dicke spectrometer becomes

$$\Delta T_{rms} = \frac{2(T_R + T_A)}{\sqrt{B_n \tau}} \tag{2.73}$$

2.6.2.3 Correlation Spectrometer

Another possible method of stabilizing a receiver system involves the correlation of signals. Two separate receivers are used in a correlation spectrometer, and the resulting output voltages are multiplied and detected. The average value of a product of two independent noise temperatures is zero, and because only correlated noise voltages yield a DC output, receiver gain instabilities will not affect the sensitivity of the correlation spectrometer. The sensitivity of the correlation spectrometer is a factor of $\sqrt{2}$ better than the Dicke spectrometer. However, two separate receivers are needed.

2.7 Summary

The Earth's atmosphere and its interaction with electromagnetic radiation has been examined on microscopic (molecular absorption) and macroscopic (particle extinction) levels. If the atmosphere is assumed to be nonscattering, horizontally homogeneous, and vertically stratified, straightforward relations can be derived for the radiation intensity observed by a downward-viewing satellite or aircraft sensor. The frequency dependence of scattering, absorption, and the Planck radiance offers various advantages for atmospheric profile sounding in the presence of clouds (see Table 2.3). Various instrument

Table 2.3
Comparison of Certain Characteristics of the 4.3-μm, 15.0-μm, and 5.0-mm Spectral Regions. Detector Noise RMS: 0.15 K (IR) and 0.7 K (MW) (*Source*: [21])

		4.3-μm	15.0-μm	5.0-mm
ENERGY	200 K	1.25	5,000	1
(Relative Planck radiance)	300 K	200	15,000	1
TEMPERATURE SENSITIVITY	200 K	1	10	4
(Relative to detector noise)	300 K	20	6	1
CLOUD TRANSMISSION	Water	6%	1%	96%
	Ice	1%	1%	99.98%

technologies present performance advantages and disadvantages that must be considered when implementing a remote sounding system.

References

[1] G. L. Stephens. *Remote Sensing of the Lower Atmosphere*. Oxford University Press, New York, 1994.

[2] J. T. Houghton. *The Physics of Atmospheres*. Cambridge University Press, Cambridge, U. K., 1986.

[3] J. M Wallace and P. V. Hobbs. *Atmospheric Science: An Introductory Survey*. Elsevier, New York, second edition, 2006.

[4] K. N. Liou. *An Introduction to Atmospheric Radiation*. Academic Press, Orlando, Florida, 1980.

[5] D. H. Staelin, A. W. Morgenthaler, and J. A. Kong. *Electromagnetic Waves*. Prentice Hall, Upper Saddle River, New Jersey, 1994.

[6] B. J. Mason. *The Physics of Clouds*. Oxford University Press, Oxford, U. K., 1971.

[7] J. S. Marshall and W. Palmer. "The distribution of raindrops with size." *Journal of the Atmosphere*, 5:165–166, 1948.

[8] C. Elachi. *Introduction to the Physics and Techniques of Remote Sensing*. Wiley, New York, 1987.

[9] D. Deirmendjian. *Electromagnetic Scattering on Spherical Polydispersions*. American Elsevier Publishing Co., Inc., New York, 1969.

[10] A. J. Gasiewski. "Microwave radiative transfer in hydrometeors." *Atmospheric Remote Sensing by Microwave Radiometry*, M. A. Janssen, Ed., Chapter 3, Wiley, New York, 1993.

[11] H. H. Aumann, et al. "AIRS/AMSU/HSB on the Aqua mission: Design, science objectives, data products, and processing systems." *IEEE Trans. Geosci. Remote Sens.*, 41(2):253–264, February 2003.

[12] H. S. Chen. *Space Remote Sensing Systems: An Introduction*. Academic Press, New York, 1985.

[13] J. D. Kraus. *Radio Astronomy*. Cygnus-Quaser Books, Powell, Ohio, second edition, 1986.

[14] M. A. Janssen. *Atmospheric Remote Sensing by Microwave Radiometry*. Wiley, New York, 1993.

[15] D. Cousins and M. J. Gazarik. *NAST Interferometer Design and Characterization: Final Report*. Project Report NOAA-26, MIT Lincoln Laboratory, July 1999.

[16] H. J. Bloom. "The Cross-track Infrared Sounder (CrIS): A sensor for operational meteorological remote sensing." *IEEE International Geoscience and Remote Sensing Symposium*, 3:1341–1343, July, 2001.

[17] G. Chalon, F. Cayla, and D. Diebel. "IASI: An advanced sounder for operational meteorology." *Proceedings of the 52nd Congress of IAF*, pages 1–5, October 2001.

[18] W. J. Blackwell, J. W. Barrett, F. W. Chen, R. V. Leslie, P. W. Rosenkranz, M. J. Schwartz,

and D. H. Staelin. “NPOESS aircraft sounder testbed-microwave (NAST-M): Instrument description and initial flight results.” *IEEE Trans. Geosci. Remote Sens.*, 39(11):2444–2453, November 2001.

[19] B. H. Lambrigtsen. “Calibration of the AIRS microwave instruments.” *IEEE Trans. Geosci. Remote Sens.*, 41(2):369–378, February 2003.

[20] C. Muth, P. S. Lee, J. C. Shiue, and W. A. Webb. “Advanced technology microwave sounder on NPOESS and NPP.” *IEEE International Geoscience and Remote Sensing Symposium*, 4:2454–2458 Vol. 4, September 2004.

[21] W. L. Smith. “Satellite techniques for observing the temperature structure of the atmosphere.” *Bulletin of the American Meteorological Society*, 53(11):1074–1082, November 1972.

3

An Overview of Inversion Problems in Atmospheric Remote Sensing

In this book, we focus on the retrieval of geophysical state parameters (for example, the atmospheric temperature profile) from radiometric measurements observed in a number of spectral bands, and this retrieval almost always requires mathematical inversion of some form of a physical model with vector-valued inputs and outputs, often termed the "forward model." In the simplest case, this equation can be cast in matrix form by discretizing the relevant parameters (atmospheric profiles are inherently continuous quantities, for example) and ignoring nonlinear terms. Even in this simple case, the system of linear equations may be overdetermined, in which case no solution exists, or undetermined, in which case an infinite number of solutions exist. Problems of this type are often classified as ill-posed, and additional assumptions or constraints must be introduced to allow unique solutions to be obtained [1]. For example, an assumption that the desired solution is close in the Euclidean sense to some a priori value might be used to allow the linear system of equations to be solved with least-squares techniques [2], or a constraint might be imposed that the solution must be sufficiently smooth by including a term related to the second derivative of the solution in the cost function to be minimized [3, 4]. Mathematical regularization techniques like those discussed above are used to increase the stability of the solutions to ill-posed problems.

Inversion problems in atmospheric remote sensing and the relationships of the variables involved are seldom linear, Gaussian, or well-posed. For these reasons, sophisticated methodologies must be used to derive a useful solution. The strategies employed can be categorized into three mutually exclusive

and collectively exhaustive categories that we will term *physical methods*, *statistical dependence methods*, and *hybrid methods*. Physical approaches essentially propagate a first guess of the atmospheric state through a forward model (for example, the radiative transfer equation (2.60)) and use iterative, numerical procedures to match the modeled (i.e., simulated) measurements to the actual observations by updating the guess at each iteration. Statistical regularization is often used (but is not required) to introduce a tendency of the optimization towards a likely value, for example. It is for this reason that we differentiate between "statistical dependence methods" and "statistical methods," as a physical method that uses statistical regularization is also a statistical method. Statistical dependence methods explicitly use (or empirically derive) a statistical relationship between the observations (i.e., the independent variables) and the geophysical state parameters (i.e., the dependent variables). No physical models are required in a statistical dependence method. Finally, hybrid methods use both physical models and statistical dependence to derive a solution to an inverse problem. For example, a forward model can be used to generate an ensemble of simulated observations and geophysical state parameters which then can be used with a statistical dependence method to carry out the inversion. We now present each of these three categories of inversion methodologies in detail.

3.1 Mathematical Notation

For the following analyses, we assume that a noisy observation of a random radiance vector $\widetilde{R}$ is related to some atmospheric state vector S through a forward model $\mathbf{f}(\cdot)$ as follows

$$\widetilde{R} = \mathbf{f}(S) + \Psi = R + \Psi \tag{3.1}$$

where Ψ is a random noise vector (that may depend on S), and R is the "noise-free" radiance observation. The retrieval seeks to estimate the state vector S given an observation of $\widetilde{R}$, where we use $\widehat{S}(\widetilde{R})$ to denote the estimate of S given an observation of $\widetilde{R}$.

3.2 Optimality

Almost all inversion techniques are designed to optimize something. There are many choices of suitable mathematical parameters to optimize: sum-squared error, probabilistic likelihood, resolution, and signal-to-noise ratio, to name a few. In the context of practical remote sensing inversion algorithms, there are two key points that must be stressed. First, it is difficult to guarantee optimality

with respect to even the most simple metrics due to the assumptions that must be satisfied by the retrieval system. We will therefore be careful to distinguish between a theoretically optimal retrieval algorithm and one that is employed in practice, where it is impossible to guarantee that all the necessary assumptions are universally satisfied. Second, a useful atmospheric retrieval algorithm must be robust with respect to a variety of metrics, some of which may even behave in direct opposition (for example, resolution and signal-to-noise ratio). We therefore in practice usually choose a theoretically optimal algorithm (for some chosen metric, usually sum-squared error) and evaluate performance for a wide variety of other metrics to ensure that the algorithm performs well even in pathological, but meteorologically important, cases that may not be well-represented in global statistics. For example, atmospheric profile retrieval algorithms are often evaluated on a global perspective using sum-squared error with respect to a comprehensive set of "ground truth," typically radiosondes or numerical model fields. It is also illuminating to examine performance on a case-by-case basis to verify that interesting atmospheric phenomenology is captured with the necessary fidelity.

3.3 Methods That Exploit Statistical Dependence

We begin with a discussion of techniques that directly utilize the joint probability distribution function (pdf) of $\widetilde{R}$ and S, $P(\widetilde{R}, S)$, or statistics based on this pdf, such as the cross-covariance.

3.3.1 The Bayesian Approach

The Bayesian approach to estimation involves the incorporation of a priori knowledge about the state vector S with knowledge gained by measuring $\widetilde{R}$. Mathematically, this knowledge is formulated in terms of five related probability density functions (pdfs):

$P(S)$	The prior (i.e., before the measurement) pdf of state S
$P(\widetilde{R})$	The prior pdf of the measurement $\widetilde{R}$
$P(\widetilde{R}, S)$	The joint prior pdf of $\widetilde{R}$ and S
$P(\widetilde{R}\|S)$	The conditional pdf of $\widetilde{R}$ given state S
$P(S\|\widetilde{R})$	The conditional pdf of S after measurement $\widetilde{R}$. This is the quantity of interest for the solution of the estimation problem.

Bayes' theorem relates the conditional probabilities as follows:

$$P(S|\widetilde{R}) = \frac{P(\widetilde{R}|S)P(S)}{P(\widetilde{R})} \tag{3.2}$$

Therefore, the Bayesian framework allows probabilities to be assigned to possible choices of $\widehat{S}(\widetilde{R})$ given knowledge of the joint and conditional probabilities of $\widetilde{R}$ and S. A reasonable choice for $\widehat{S}(\widetilde{R})$ is the value of S for which $P(S|\widetilde{R})$ is the largest (known as the maximum a posteriori, or MAP, estimator, and sometimes called the *maximum likelihood estimator*).

3.3.1.1 Bayes' Least-Squares Estimator

An alternative to the MAP estimator is the estimator $\mathbf{g}(\cdot)$ that minimizes some suitable cost criterion, C:

$$\widehat{S}(\cdot) = \underset{\mathbf{g}(\cdot)}{\arg\min}\ C(S, \mathbf{g}(\widetilde{R})) \tag{3.3}$$

The sum-squared error (SSE) cost criterion

$$C = E[(S - \widehat{S})^T(S - \widehat{S})] \tag{3.4}$$

is commonly chosen for this purpose and results in the following estimator, sometimes called the Bayes' least-squares (BLS) estimator:

$$\widehat{S}(\widetilde{R}) = E[S|\widetilde{R}] \tag{3.5}$$

The BLS and MAP estimators may be identical under some circumstances, for example, if S and $\widetilde{R}$ are jointly Gaussian.

3.3.1.2 Bayes' Linear Least-Squares Estimator

The BLS estimator has two disadvantages: it is often a nonlinear function of $\widetilde{R}$, and it requires a complete statistical representation of the relationship between $\widetilde{R}$ and S, which is rarely available in practice. If we constrain the estimator $\mathbf{g}(\cdot)$ in (3.3) to be linear, the resulting estimator depends only on a second-order characterization of the statistical relationship between $\widetilde{R}$ and S. This estimator is the linear least-squares estimator (LLSE):

$$\widehat{S}(\widetilde{R}) = \mathbf{C}_{S\widetilde{R}}\mathbf{C}_{\widetilde{R}\widetilde{R}}^{-1}\widetilde{R} \equiv \mathbf{L}_{S\widetilde{R}}\widetilde{R} \tag{3.6}$$

with error covariance

$$\mathbf{C}_{\epsilon\epsilon} = \mathbf{C}_{SS} - \mathbf{C}_{S\widetilde{R}}\mathbf{C}^{-1}_{\widetilde{R}\widetilde{R}}\mathbf{C}^{T}_{S\widetilde{R}} \tag{3.7}$$

We have assumed without loss of generality that S and $\widetilde{R}$ are zero-mean and have used $\mathbf{C}_{XY}$ to denote the cross-covariance of X and Y. The LLS and BLS estimators are identical when $\widetilde{R}$ and S are jointly Gaussian.

3.3.2 Linear and Nonlinear Regression Methods

Direct application of the BLS and LLS estimators is often precluded in practical applications because the needed expected values (in the case of BLS) and covariance matrices (in the case of LLS) both depend on the joint probability distribution functions of S and $\widetilde{R}$, and these pdfs are very difficult to calculate directly. A much more convenient approach is to first estimate the needed statistical parameters from the available sample data and then derive the estimator from these sample statistics. We now present two examples of this approach, where statistical relationships are derived empirically from sample data, taken here to mean an ensemble of N pairs of S and $\widetilde{R}$:

$$(S_1, \widetilde{R}_1), \ldots, (S_N, \widetilde{R}_N) \in \mathcal{S} \times \mathcal{R} \tag{3.8}$$

This ensemble could be assembled, for example, from colocated satellite radiance measurements and radiosonde observations of temperature as a function of altitude. Linear regression operators can be calculated directly from "sample covariances" that are empirically derived from the sample data. Nonlinear regression operators generalize this approach to allow more complicated mathematical relationships between S and $\widetilde{R}$ to be represented. A parameterized, nonlinear function (for example, a polynomial) is often used to fit the sample data. The parameters can be chosen using a numerical optimization technique that minimizes a cost function usually involving the Euclidean distance between the actual data and the derived fit.

3.3.2.1 Linear Regression

A given set of N observations of P parameters can be arranged into a $P \times N$ matrix, $\mathbf{X}$. The sample mean, M_X, is a column vector where each element is the average of each row of $\mathbf{X}$. The sample covariance $\widehat{\mathbf{C}}_{XX}$ is given by

$$\widehat{\mathbf{C}}_{XX} = \frac{\overline{\mathbf{X}}^T\overline{\mathbf{X}}}{N-1} \tag{3.9}$$

where $\overline{\mathbf{X}}$ is calculated by removing the sample mean from $\mathbf{X}$.

Given a matrix of noisy radiance observations, $\widetilde{\mathbf{R}}$, where each column corresponds to an observation and each row corresponds to a spectral channel, and a matrix of geophysical observations, $\mathbf{S}$, where each column corresponds to an observation and each row corresponds to a vertical level, for example, the linear regression estimate is

$$\widehat{S}(\widetilde{R}) = M_S + \widehat{\mathbf{C}}_{S\widetilde{R}}\widehat{\mathbf{C}}^{-1}_{\widetilde{R}\widetilde{R}}(\widetilde{R} - M_{\widetilde{R}}) \tag{3.10}$$

where $\widehat{\mathbf{C}}_{S\widetilde{R}}$ and $\widehat{\mathbf{C}}_{\widetilde{R}\widetilde{R}}$ are the sample covariance matrices. If the additive random noise term in (3.1) is zero mean and uncorrelated with the radiances, R, and the geophysical state vector, S, then (3.10) can be expressed in a simplified form as follows:

$$\widehat{S}(\widetilde{R}) = M_S + \widehat{\mathbf{C}}_{SR}(\widehat{\mathbf{C}}_{RR} + \mathbf{C}_{\Psi\Psi})^{-1}(\widetilde{R} - M_R) \tag{3.11}$$

where $\mathbf{C}_{\Psi\Psi}$ is the noise covariance. If the noise covariance is known a priori, it can be used directly in (3.11), otherwise, it can be estimated using techniques presented in Chapter 4. Note that the covariance matrix can serve as a form of mathematical regularization when $\widehat{\mathbf{C}}_{RR}$ is near singular. This can occur frequently with hyperspectral/ultraspectral measurements due to the very high degree of correlation among some of the channels.

It is interesting, and maybe not obvious, that linear regression can provide optimal estimates even in cases where the observations are nonlinearly related to the variables to be estimated. We illustrate this with a simple example. Suppose we wish to retrieve a scalar variable s from two scalar observations r_1 and r_2 and these variables are nonlinearly related as follows:

$$r_1 = a_1 + b_1 s + c_1 s^2 \tag{3.12}$$

$$r_2 = a_2 + b_2 s + c_2 s^2 \tag{3.13}$$

A linear combination of r_1 and r_2 recovers s exactly:

$$\widehat{s} = \alpha + \beta r_1 + \gamma r_2 \tag{3.14}$$

where

$$\alpha = \frac{c_1 a_2 - c_2 a_1}{c_1 b_2 - c_2 b_1} \tag{3.15}$$

$$\beta = \frac{c_2}{c_2 b_1 - c_1 b_2} \tag{3.16}$$

$$\gamma = \frac{c_1}{c_1 b_2 - c_2 b_1} \tag{3.17}$$

While trivial in the case presented here, this example provides insight into recent work ([5, 6], for example) demonstrating that linear estimators perform very well when used to retrieve temperature and water vapor from hyperspectral infrared observations, where the relationships are quite nonlinear due to the Planck function and the dependence of atmospheric absorption on water vapor content, but there are a large number of spectral channels to allow the nonlinear dependence to be "unraveled" by appropriate linear combinations. We will revisit this example from a different perspective in Section 5.1.3.

3.3.2.2 Nonlinear Parametric Regression

The linear form afforded by (3.11) is simple and convenient to apply in practice, but can lead to substantial inaccuracies in cases where the relationship between R and S is nonlinear. The linear regression framework presented above can be readily expanded by including nonlinear functions of R as inputs to the linear regression. For example, simple polynomial terms can be constructed and the linear regression operator can be used to optimize the coefficients that modulate these terms. Polynomial regression is an example of a parameterized method, where the model structure and complexity is predetermined and only the free parameters in the model need to be derived. Other parameterized, nonlinear functions can also be constructed, and the parameters can be obtained using numerical optimization techniques.

3.3.2.3 Nonlinear Nonparametric Regression

Nonparametric models differ from parametric models in that the model structure is not specified a priori but is instead determined from data during the training process. The term nonparametric is not meant to imply that such models completely lack parameters but that the number and nature of the parameters are flexible and not fixed in advance. A neural network can be broadly categorized as a special case of nonparameterized nonlinear regression and will be discussed in detail in Chapter 5.

We end this section with a temperature retrieval example using both linear and polynomial regression techniques. The NOAA88b radiosonde set contains approximately 7,500 global atmospheric profiles of temperature, water vapor, and ozone. A radiative transfer package was used to simulate microwave sounding observations in 100 spectral bands near the 118.75-GHz oxygen line. Figure 3.1 shows the temperature retrieval RMS errors as a function of altitude for linear regression and polynomial regression. The a priori error (not shown) is approximately 10K throughout the troposphere. Although both the linear and nonlinear regressions have substantially reduced

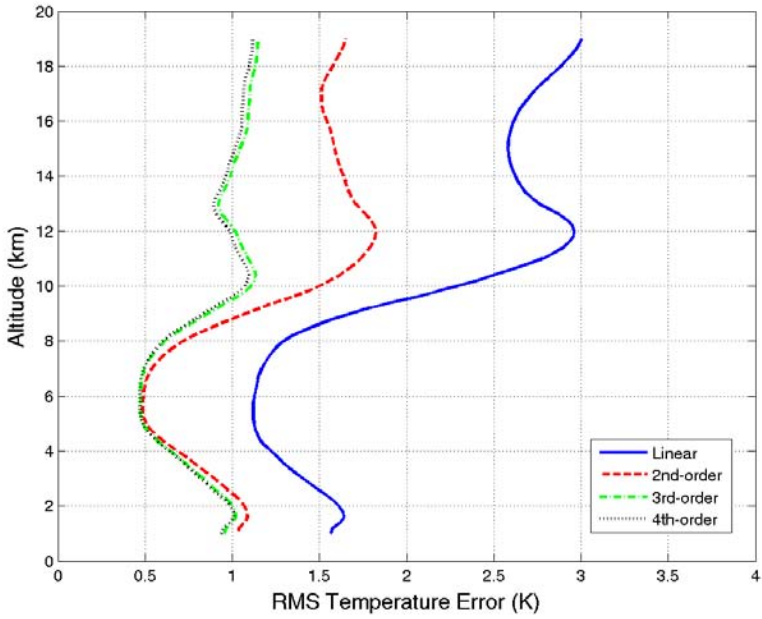


Figure 3.1 RMS temperature retrieval error for linear regression and nonlinear (polynomial) regression. A hypothetical microwave sounder with 100 channels near the 118.75-GHz oxygen line was used in the retrieval simulation.

the a priori error, the polynomial containing at least third-order terms clearly improves the accuracy of the retrieval.

3.3.2.4 Ridge Regression

It may be desirable to constrain the magnitude of the coefficients in the regression operator, $\mathbf{L}_{S\widetilde{R}}$. This is a form of regularization that stabilizes the solution if the covariance matrix is nearly singular. The modified cost function is

$$C = E[(S - \widehat{S})^T (S - \widehat{S})] + \gamma \text{trace}\{\mathbf{L}^T \mathbf{L}\} \tag{3.18}$$

and the solution is

$$\widehat{S}(\widetilde{R}) = \mathbf{C}_{S\widetilde{R}} \left(\mathbf{C}_{\widetilde{R}\widetilde{R}} + \gamma \mathbf{I} \right)^{-1} \widetilde{R} \tag{3.19}$$

This modification to the standard linear regression operator is termed *ridge regression* [7].

3.4 Physical Inversion Methods

The previous estimators are based entirely on the statistical relationship between R and S. Other methods use knowledge of the forward model $\mathbf{f}(\cdot)$ together with whatever limited statistical characterization of R and S is available. If we return to (3.1) and make the assumptions that the error and the a priori state distributions are Gaussian, the probability distributions in the numerator of (3.2) are proportional to the following terms:

$$P(\widetilde{R}|S) \sim \exp\left\{-\frac{1}{2}(\widetilde{R}-R)^T\mathbf{C}_{\Psi\Psi}^{-1}(\widetilde{R}-R)\right\} \tag{3.20}$$

$$P(S) \sim \exp\left\{-\frac{1}{2}(S-S_a)^T\mathbf{C}_{SS}^{-1}(S-S_a)\right\} \tag{3.21}$$

where S_a is the a priori state vector. The denominator in (3.2), $P(\widetilde{R})$, is often a normalizing factor in practice and can be neglected [8]. The most likely value of $P(S|\widetilde{R})$ is therefore the maximum of the product of (3.20) and (3.21), or equivalently, the maximum of the sum of their natural logarithms. This maximization is equivalent to minimizing:

$$\xi_{\text{min}} = (\widetilde{R}-R)^T\mathbf{C}_{\Psi\Psi}^{-1}(\widetilde{R}-R) + (S-S_a)^T\mathbf{C}_{SS}^{-1}(S-S_a) \tag{3.22}$$

Most physical retrieval approaches attempt to minimize a cost function similar in form to that given in (3.22), that is,

$$\widehat{S}(\cdot) = \underset{S}{\arg\min}\ \xi_{\text{min}} \tag{3.23}$$

although many variations of this cost function could be used in practice. For example, it might be advantageous to minimize a weighted sum of the two terms in (3.22). Other physical approaches seek to minimize a quite different cost function, for example, the vertical resolution of the retrieval [9]. We will not discuss these methods here – the interested reader is referred to Twomey [1] and Rodgers [8] for detailed treatment of these topics.

3.4.1 The Linear Case

The solution to (3.23) can be found analytically only under certain circumstances. The earlier assumption of Gaussianity must hold, and the relationship between $\widetilde{R}$ and S must be linear:

$$\widetilde{R} = \mathbf{W}S + \Psi \tag{3.24}$$

where $\mathbf{W}$ is sometimes called the weighting function matrix. It can be shown [8] that the solution in this case can be expressed by two equivalent relations:

$$\widehat{S}_m(\widetilde{R}) = S_a + (\mathbf{W}^T\mathbf{C}_{\Psi\Psi}^{-1}\mathbf{W} + \mathbf{C}_{SS}^{-1})^{-1}\mathbf{W}^T\mathbf{C}_{\Psi\Psi}^{-1}(\widetilde{R} - \mathbf{W}S_a) \quad (3.25)$$

$$\widehat{S}_n(\widetilde{R}) = S_a + \mathbf{C}_{SS}\mathbf{W}^T(\mathbf{W}\mathbf{C}_{SS}\mathbf{W}^T + \mathbf{C}_{\Psi\Psi})^{-1}(\widetilde{R} - \mathbf{W}S_a) \quad (3.26)$$

where the subscript on $\widehat{S}$ (m or n) denotes the order of the matrix to be inverted. Note the similarity of the n-form solution with the linear regression estimate given by (3.11).

3.4.1.1 The Minimum-Information Retrieval

The minimum-information retrieval picks the $\widehat{S}$ which is "closest" in the least-squares sense to S_a and satisfies

$$(\widetilde{R} - R)^T(\widetilde{R} - R) = M\sigma^2 \quad (3.27)$$

where $M\sigma^2$ is a scalar quantity related to the measurement error. In the linear case, the minimum-information solution in the presence of noise with covariance $\mathbf{C}_{\Psi\Psi}$ is then

$$\widehat{S}(\widetilde{R}) = S_a + \mathbf{W}^T(\mathbf{W}\mathbf{W}^T + \beta\mathbf{C}_{\Psi\Psi})^{-1}(\widetilde{R} - \mathbf{W}S_a) \quad (3.28)$$

where β is some constant (in units of 1/K^2). Note that the minimum-information retrieval is the n-form solution given in (3.26) with the a priori state covariance replaced by the identity matrix.

Figure 3.2 shows the performance of three operators used to retrieve the temperature profile ($S = T$) from simulated Advanced Microwave Sounding Unit (AMSU) radiances in clear-air: the minimum-information retrieval (3.28), the linear model with known $\mathbf{C}_{SS}$ (3.26), and the linear regression estimator (3.11). The graph demonstrates the significant impact of a priori statistics on the retrieval performance.

3.4.2 The Nonlinear Case

The minimization of (3.23) often requires numerical methods. We apply the canonical approach and set the derivative of the cost function (3.22) to zero and numerically find a root of the resulting equation. The derivative of the cost function is calculated as follows:

$$-[\nabla_S \mathbf{f}(S)]^T\mathbf{C}_{\Psi\Psi}^{-1}[\widetilde{R} - \mathbf{f}(S)] + \mathbf{C}_{SS}^{-1}(S - S_a) \quad (3.29)$$

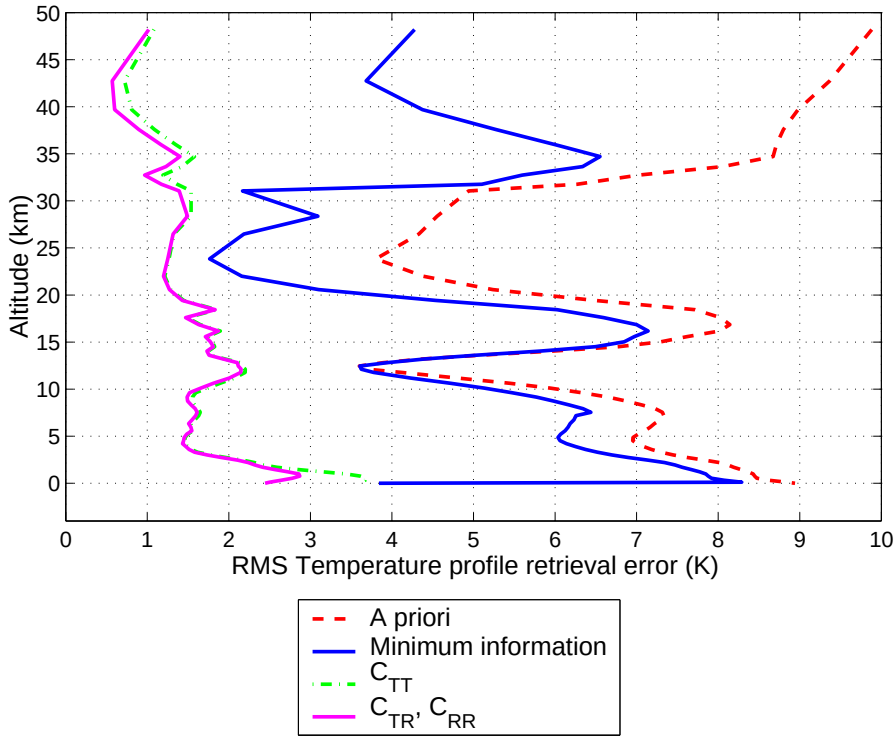


Figure 3.2 Comparison of temperature retrieval techniques. The minimum information retrieval (with $\beta = 1\text{K}^{-2}$) only uses information contained in the weighting function matrix. The optimal linear model retrieval uses the weighting function matrix and $\mathbf{C}_{TT}$ only. The direct multiple regression retrieval uses statistical characterizations of $\mathbf{C}_{RR}$ and $\mathbf{C}_{TT}$.

Newton's method iteratively solves $\mathbf{h}(S) = 0$ by stepping from the current guess (S_i) to the next guess (S_{i+1}) according to:

$$S_{i+1} = S_i - [\nabla_S \mathbf{h}(S_i)]^{-1}\mathbf{h}(S_i) \tag{3.30}$$

Substitution of (3.29) into (3.30) yields the update step needed to minimize the cost function:

$$\begin{aligned} S_{i+1} \quad = \quad & S_i + \left\{\mathbf{C}_{SS}^{-1} + \mathbf{K}_i^T\mathbf{C}_{\Psi\Psi}^{-1}\mathbf{K}_i - [\nabla_S \mathbf{K}_i]^T\mathbf{C}_{\Psi\Psi}^{-1}[\widetilde{R} - \mathbf{f}(S_i)]\right\}^{-1} \\ & \times \left\{\mathbf{K}_i^T\mathbf{C}_{\Psi\Psi}^{-1}[\widetilde{R} - \mathbf{f}(S_i)] - \mathbf{C}_{SS}^{-1}(S_i - S_a)\right\} \end{aligned} \tag{3.31}$$

where the matrix $\mathbf{K}_i$ contains the derivatives of the forward model with respect to the state vector, that is, $\mathbf{K}_i = \nabla_S \mathbf{f}(S_i)$. The implementation of this

method is complicated in practice by the calculation of the Hessian matrix, $\nabla_S \mathbf{K}_i$, which is computationally intensive to evaluate. It is often reasonable to neglect the Hessian term in moderately linear problems [10], and the update step becomes:

$$\begin{aligned} S_{i+1} &= S_i + \left\{\mathbf{C}_{SS}^{-1} + \mathbf{K}_i^T \mathbf{C}_{\Psi\Psi}^{-1} \mathbf{K}_i\right\}^{-1} \\ &\quad \times \left\{\mathbf{K}_i^T \mathbf{C}_{\Psi\Psi}^{-1} [\widetilde{R} - \mathbf{f}(S_i)] - \mathbf{C}_{SS}^{-1}(S_i - S_a)\right\} \end{aligned} \tag{3.32}$$

Alternatively, the update can be expressed relative to the a priori state, as follows:

$$\begin{aligned} S_{i+1} &= S_a + \mathbf{C}_{SS} \mathbf{K}_i^T (\mathbf{K}_i \mathbf{C}_{SS} \mathbf{K}_i^T + \mathbf{C}_{\Psi\Psi})^{-1} \\ &\quad \times \left\{\widetilde{R} - \mathbf{f}(S_i) + \mathbf{K}_i (S_i - S_a)\right\} \end{aligned} \tag{3.33}$$

This equation will be used in a retrieval example presented in Chapter 10, at which point we also discuss convergence criteria and computational efficiency.

3.5 Hybrid Inversion Methods

Some knowledge about the relationship between $\widetilde{R}$ and S must be available to allow the formulation of a suitable estimate $\widehat{S}(\widetilde{R})$, and we have seen that statistical dependence methods exploit statistical relationships and physical methods exploit physical relationships. It may be possible to improve retrieval accuracy and/or efficiency by using both statistical and physical knowledge of the state parameters and the measured radiances.

3.5.1 Improved Retrieval Accuracy

The physical inversion approaches previously discussed are less effective as the relevant processes deviate from linearity and Gaussianity, and some atmospheric processes present formidable retrieval challenges. Most notable are clouds and precipitation, which can be highly variable, dynamic, nonlinear, and non-Gaussian. Recent work on the retrieval of precipitation rate from passive microwave measurements [11–17] has demonstrated the utility of microphysical precipitation models. The complexity and nonlinearity of these models, however, has precluded their direct use in retrieval algorithms. One fruitful approach has been to generate training data using the microphysical precipitation model and subsequently derive the relevant statistical relationships with a nonlinear regression (neural network) method.

3.5.2 Improved Retrieval Efficiency

One advantage of decoupling the physical model from the retrieval algorithm is an increase in computational efficiency. Many physical models, especially those that must represent complicated, nonlinear processes in many spectral bands, are computationally prohibitive for retrieval systems that must operate in near-real-time. An alternative approach is to use the forward model "off-line" from the retrieval to generate a database of radiance–state pairs. The statistical relationships are also derived off-line, and optimal estimators can be calculated. The operational retrieval then can implement these estimators.

3.6 Error Analysis

Once a suitable retrieval operator has been constructed, it is necessary to assess the performance of the estimates. Many reasonable metrics could be examined, including the sensitivity of the retrieval to sensor noise, the resolution of the retrieval, the degree to which the retrieval system (i.e., the retrieval algorithm and the sensor) is "blind" to the atmospheric parameters of interest, and so forth. These metrics, and others, can be readily calculated from the retrieval operators in many cases. Perturbation analysis can be used when complexity and/or nonlinearity preclude simple, direct techniques.

3.6.1 Analytical Analysis

We begin with the linear case, where the state estimate takes the form

$$\widehat{S}(\widetilde{R}) = \mathbf{D}\widetilde{R} \tag{3.34}$$

where we have encapsulated the offset term into $\mathbf{D}$ by adding a new element to $\widetilde{R}$ and setting this element to one. The $\mathbf{D}$ matrix may be derived in many ways, for example, using (3.11), (3.26), or (3.28). Under the assumption of a linear forward model, (3.24) can be substituted into (3.34), and we obtain:

$$\widehat{S}(\widetilde{R}) = \mathbf{D}\mathbf{W}S + \mathbf{D}\Psi \tag{3.35}$$

and we observe that the estimated state vector is a linear function of the true state function. In the ideal, linear case, this decomposition yields the two basic contributions of retrieval imperfection: smoothing, represented by the averaging kernel, $\mathbf{A} = \mathbf{D}\mathbf{W}$, and retrieval noise, represented by $\mathbf{D}\Psi$. In practice, the forward model does not perfectly capture reality and is usually nonlinear. These complications can be handled by including additional forward model error terms and linearizing the forward model about a suitable

operating point [8]. This results in (at least) a third contribution to retrieval imperfection, in the form of a retrieval bias. The relative contributions of smoothing error and retrieval noise to the total retrieval error covariance can be readily calculated:

$$\mathbf{C}_{\text{smooth}} = (\mathbf{A} - \mathbf{I})\mathbf{C}_{SS}(\mathbf{A} - \mathbf{I})^T \tag{3.36}$$

and

$$\mathbf{C}_{\text{noise}} = \mathbf{D}\mathbf{C}_{\Psi\Psi}\mathbf{D}^T \tag{3.37}$$

where we have assumed that the measurement noise is uncorrelated with the atmospheric state. The direct calculation of the error contributions due to forward model errors is difficult using analytical methods unless assumptions and approximations are used. Numerical techniques such as Monte Carlo analysis can be helpful in such circumstances.

3.6.2 Perturbation Analysis

3.6.2.1 Atmospheric Smoothing

In the linear case, the derivative of the estimated state with respect to the true state is simply the averaging kernel:

$$\frac{\partial \widehat{S}}{\partial S} = \mathbf{A} \tag{3.38}$$

Analysis of more complicated cases is facilitated by decomposing this derivative into the product of the forward model Jacobian and the retrieval Jacobian:

$$\frac{\partial \widehat{S}}{\partial S} = \frac{\partial R}{\partial S}\frac{\partial \widehat{S}}{\partial R} \tag{3.39}$$

The forward model and the retrieval operator can then be linearized about some operating point, and these derivatives can be calculated – numerical techniques can be used, if necessary. We will see in Chapter 8 that neural network estimators are particularly amenable to perturbation analyses because the Jacobians are very easy to compute.

3.6.2.2 Retrieval Noise

The contribution of retrieval noise to the total retrieval error can be approximated using a second-order propagation of errors approach, as follows:

$$\mathbf{C}_{\text{noise}} \approx \frac{\partial \widehat{S}}{\partial R}\mathbf{C}_{\Psi\Psi}\left(\frac{\partial \widehat{S}}{\partial R}\right)^T \tag{3.40}$$

3.7 Summary

The set of equations relating an atmospheric state S to the observed radiances $\widetilde{R}$ is seldom directly invertible. The inversion is often ill-posed, where an infinite number of solutions exist, or ill-conditioned, where small perturbations of the radiance values lead to very large perturbations of the solution. Regularization techniques in the form of constraints that penalize deviations of the solution from an a priori state, for example, are used to improve the stability of the retrieval. Relationships between S and $\widetilde{R}$ can be derived using physical or statistical means, or both. In some cases, retrieval error components can be isolated and attributed to biases, atmospheric smoothing, sensor noise, and so forth.

References

[1] S. Twomey. *Introduction to the Mathematics of Inversion in Remote Sensing and Indirect Measurements*. Elsevier Scientific Publishing Company, New York, 1977.

[2] G. Strang. *Linear Algebra and Its Applications*. Academic Press, New York, 1980.

[3] A. Tikhonov. "On the solution of incorrectly stated problems and a method of regularization." *Dokl. Acad. Nauk SSSR*, 151:501–504, 1963.

[4] A. N. Tikhonov, A. V. Goncharsky, V. V. Stepanov, and A. G. Yagola. *Numerical Methods for the Solution of Ill-Posed Problems*. Kluwer, Boston, Massachusetts, 1995.

[5] M. D. Goldberg, Y. Qu, L. M. McMillin, W. Wolff, L. Zhou, and M. Divakarla. "AIRS near-real-time products and algorithms in support of operational numerical weather prediction." *IEEE Trans. Geosci. Remote Sens.*, 41(2):379–389, February 2003.

[6] E. Weisz, H. L. Huang, J. Li, E. Borbas, K. Baggett, P. K. Thapliyal, and L. Guan. "International MODIS and AIRS processing package: AIRS products and applications." *J. App. Rem. Sens.*, 1:1–23, July 2007.

[7] A. E. Hoerl. "Application of ridge analysis to regression problems." *Chemical Engineering Progress*, 58:54–59, 1962.

[8] C. D. Rodgers. *Inverse Methods for Atmospheric Sounding*. World Scientific, New York, 2000.

[9] G. E. Backus and J. F. Gilbert. "Uniqueness in the inversion of inaccurate gross earth data." *Phil. Trans. Roy. Soc. London*, 266:123–192, 1970.

[10] C. D. Rodgers. "Retrieval of atmospheric temperature and composition from remote measurements of thermal radiation." *J. Geophys. Res.*, 41(7):609–624, July 1976.

[11] C. Surussavadee and D. H. Staelin. "Comparison of AMSU millimeter-wave satellite observations, MM5/TBSCAT predicted radiances, and electromagnetic models for hydrometeors." *IEEE Trans. Geosci. Remote Sens.*, 44(10):2667–2678, October 2006.

[12] C. Surussavadee and D. H. Staelin. "Precipitation retrieval accuracies for geo-microwave sounders." *IEEE Trans. Geosci. Remote Sens.*, 45(10):3150–3159, October 2007.

[13] C. Surussavadee and D. H. Staelin. "Millimeter-wave precipitation retrievals and observed-versus-simulated radiance distributions: Sensitivity to assumptions." *J. Atmos. Sci.*, 64(11):3808–3826, November 2007.

[14] C. Surussavadee and D. H. Staelin. "Global millimeter-wave precipitation retrievals trained with a cloud-resolving numerical weather prediction model, Part I: Retrieval design." *IEEE Trans. Geosci. Remote Sens.*, 46(1):99–108, January 2008.

[15] C. Surussavadee and D. H. Staelin. "Global millimeter-wave precipitation retrievals trained with a cloud-resolving numerical weather prediction model, Part II: Performance evaluation." *IEEE Trans. Geosci. Remote Sens.*, 46(1):109–118, January 2008.

[16] R. V. Leslie, L. J. Bickmeier, W. J. Blackwell, F. W. Chen, and L. Jairam. "Improved simulation methodology for retrieval of convective precipitation from spaceborne

passive microwave measurements." *IEEE International Geoscience and Remote Sensing Symposium Proceedings*, July 2008.

[17] R. V. Leslie, W. J. Blackwell, L. J. Bickmeier, and L. G. Jairam. "Improved simulation methodology for retrieval of convective precipitation from spaceborne passive microwave measurements." *SPIE Asia-Pacific Remote Sensing Symposium*, November 2008.

4

Signal Processing and Data Representation

Before proceeding with the application of mathematical inversion techniques to geophysical retrieval problems, we first examine the statistical nature of the parameters involved. Bellman has noted a "curse of dimensionality" [1] in the estimation of nonlinear variables in high-dimensional input spaces, as the complexity of the inversion methodology increases rapidly with the number of inputs. It may be possible to substantially simplify the inversion methodology by exploiting statistical properties of the data to be represented. This in turn can have several beneficial consequences. Reducing the number of free parameters in the inversion usually results in a more stable, better-conditioned retrieval operator that is less sensitive to data artifacts, such as sensor noise or interfering atmospheric or surface features. Computational efficiency is also improved by reducing retrieval complexity. In the case of statistical dependence methods, a smaller training ensemble is generally required for simple retrieval operators with fewer free parameters.

With these considerations in mind, we therefore find it desirable to apply linear operators when possible and nonlinear operators when needed. This serves to keep the retrieval simple, stable, computationally efficient, and relatively easy to characterize. One framework that can be used to separate the problem into linear and nonlinear components is as follows:

$$\widehat{S}(\widetilde{R}) = \mathbf{l}_{\text{post}}(\mathbf{n}(\mathbf{l}_{\text{pre}}(\widetilde{R}))) \tag{4.1}$$

where $\mathbf{l}_{\text{pre}}(\cdot)$ is a linear preprocessing function, $\mathbf{l}_{\text{post}}(\cdot)$ is a linear post-processing function, and $\mathbf{n}(\cdot)$ is a nonlinear function. For the remainder of this book, we will assume a neural network will be used for $\mathbf{n}(\cdot)$. In this chapter, we develop tools that can be used to assess the information content in the

radiance measurements and atmospheric state observations. Careful analysis of this information content reveals how linear functions can be combined with neural networks to realize a relatively simple, yet powerful, retrieval system.

4.1 Analysis of the Information Content of Hyperspectral Data

The information content of a measurement can be defined in a number of ways. Two common scalar metrics that are used to measure the information contained in a measurement are the Shannon information content [2] and the number of degrees of freedom present in the signal [3], both of which are related to the eigenvalues of data covariance matrices.

4.1.1 Shannon Information Content

The Shannon definition of information content depends on the entropy of the underlying probability density functions (pdfs) that characterize the measurement. The entropy of a continuous pdf $P(\widetilde{R})$ can be defined as

$$H(P) = -\int P(\widetilde{R})\log\left[P(\widetilde{R})\right]\, d\widetilde{R} \tag{4.2}$$

The base of the logarithm is usually taken to be 2, in which case the units of entropy are bits, or e, in which case the units of entropy are nats. The information content of a measurement in the Shannon sense can be defined as the reduction of entropy upon making a measurement of $\widetilde{R}$:

$$I(R,\widetilde{R}) = H[P(R)] - H[P(R|\widetilde{R})] \tag{4.3}$$

or equivalently

$$I(R,\widetilde{R}) = H[P(\widetilde{R})] - H[P(\widetilde{R}|R)] \tag{4.4}$$

For example, the entropy of $\widetilde{R}$ can be calculated assuming a multivariate Gaussian distribution

$$P(\widetilde{R}) = \frac{1}{(2\pi)^{N/2}|\mathbf{C}_{\widetilde{R}\widetilde{R}}|^{1/2}} \exp\left\{-\frac{1}{2}\widetilde{R}^T\mathbf{C}^{-1}_{\widetilde{R}\widetilde{R}}\widetilde{R}\right\} \tag{4.5}$$

where $\mathbf{C}_{\widetilde{R}\widetilde{R}}$ is the covariance of $\widetilde{R}$:

$$\begin{aligned} H[P(\widetilde{R})] &= \sum_{i=1}^{N} \log(2\pi e\lambda_i)^{1/2} && (4.6)\\ &= N\log(2\pi e)^{1/2} + \frac{1}{2}\log\left(\prod_{i=1}^{N}\lambda_i\right) && (4.7)\\ &= N\log(2\pi e)^{1/2} + \frac{1}{2}\log|\mathbf{C}_{\widetilde{R}\widetilde{R}}| && (4.8)\\ &= c_1 + \frac{1}{2}\log|\mathbf{C}_{RR} + \mathbf{C}_{\Psi\Psi}| && (4.9) \end{aligned}$$

It can be shown that the volume of an ellipsoid describing a surface of constant probability is proportional to the square root of the product of the eigenvalues $\{\lambda_i\}$ of $\mathbf{C}_{\widetilde{R}\widetilde{R}}$ [4]. Therefore, the entropy of the pdf is related to the volume inside a surface of constant probability. When a measurement is made, this "volume of uncertainty" decreases. The information content is a measure of the factor by which it decreases, a generalization of the scalar concept of signal-to-noise ratio [3].

The information content of a measurement of $\widetilde{R}$ can be calculated under the assumption of Gaussianity. The covariance of $\widetilde{R}$ before the measurement is $\mathbf{C}_{RR} + \mathbf{C}_{\Psi\Psi}$ while the covariance of $\widetilde{R}$ after the measurement is $\mathbf{C}_{\Psi\Psi}$. These values are used in (4.4):

$$\begin{aligned} I(R,\widetilde{R}) &= H[P(\widetilde{R})] - H[P(\widetilde{R}|R)]\\ &= \frac{1}{2}\log|\mathbf{C}_{RR} + \mathbf{C}_{\Psi\Psi}| - \frac{1}{2}\log|\mathbf{C}_{\Psi\Psi}| && (4.10)\\ &= \frac{1}{2}\log|\mathbf{C}_{\Psi\Psi}^{-1}(\mathbf{C}_{RR} + \mathbf{C}_{\Psi\Psi})| && (4.11)\\ &= \frac{1}{2}\log|\mathbf{C}_{\Psi\Psi}^{-1/2}\mathbf{C}_{RR}\mathbf{C}_{\Psi\Psi}^{-1/2} + \mathbf{I}| && (4.12)\\ &= \frac{1}{2}\log|\widetilde{\mathbf{C}}_{RR} + \mathbf{I}| && (4.13) \end{aligned}$$

where

$$\widetilde{\mathbf{C}}_{RR} \triangleq \mathbf{C}_{\Psi\Psi}^{-1/2}\mathbf{C}_{RR}\mathbf{C}_{\Psi\Psi}^{-1/2} \qquad (4.14)$$

is the *whitened* covariance matrix describing R. Note that (4.13) can be easily calculated from the eigenvalues of $\widetilde{\mathbf{C}}_{RR}$

$$I(R,\widetilde{R}) = \frac{1}{2}\sum_i \log(1+\lambda_i) \qquad (4.15)$$

4.1.2 Degrees of Freedom

Another measure of information contained within a measurement is the number of degrees of freedom (DOF), where a degree of freedom can be loosely defined as an independent component of $\widetilde{R}$ that contains some information about R and the uncertainty of this information is smaller than the measurement error of the component. For example, if $\widetilde{R}$ is prewhitened and projected onto the eigenvectors of $\widetilde{\mathbf{C}}_{RR} + \mathbf{I}$, the eigenvalues of $\widetilde{\mathbf{C}}_{RR}$ give the signal-to-noise ratio (SNR) of each uncorrelated component. It is intuitive that the number of degrees of freedom should be the number of components with SNR greater than or about equal to one. Furthermore, it is convenient to make a distinction between a degree of freedom due to signal (DOF_s) and a degree of freedom due to noise (DOF_n). The previous description assumed implicitly that the degree of freedom was due to signal. If there are N elements of R, we require

$$\text{DOF}_s + \text{DOF}_n = N \tag{4.16}$$

Rodgers [3] suggests the following definitions of DOF_s and DOF_n:

$$\begin{aligned} \text{DOF}_s &\triangleq \text{tr}(\widetilde{\mathbf{C}}_{RR}[\widetilde{\mathbf{C}}_{RR} + \mathbf{I}]^{-1}) \\ &= \sum_i \frac{\lambda_i}{(1+\lambda_i)} \end{aligned} \tag{4.17}$$

$$\begin{aligned} \text{DOF}_n &\triangleq \text{tr}([\widetilde{\mathbf{C}}_{RR} + \mathbf{I}]^{-1}) \\ &= \sum_i \frac{1}{(1+\lambda_i)} \end{aligned} \tag{4.18}$$

These definitions do not necessarily yield integer values; for example, a component with SNR $= 1$ would contribute $\frac{1}{2}$ to DOF_s and $\frac{1}{2}$ to DOF_n. Equation (4.16) is satisfied, however. As a final comment, it is interesting to note the similarity between (4.15) and (4.17), both of which depend only on the eigenvalues of $\widetilde{\mathbf{C}}_{RR}$.

4.1.2.1 An Example

We now present an example to illustrate how estimates of the degrees of freedom can be used to eliminate redundant spectral information and remove sensor noise. A radiative transfer package [5] was used to simulate a 1,000-channel hyperspectral infrared sounder measuring upwelling thermal emission at the top of the atmosphere. The channels sample the carbon dioxide and water vapor lines shown in Figure 2.3 from approximately 4-μm to 15-μm wavelengths. Random noise was added to the simulated measurements.

The top panel of Figure 4.1 shows the number of degrees of freedom for the Shannon and Rodgers conventions; both show that cumulative information reaches an asymptote after approximately 50 eigenvalues. The implication is that the radiance spectra can be compressed by a factor of 20 with negligible information loss by simply setting to zero eigenvalues with order higher than 50. Note also that the high-order eigenvalues are dominated by random noise, as the DOF_n curve increases linearly after the 50th eigenvalue. Therefore, in addition to the more compact representation of the radiance information afforded by the first 50 eigenvalues, the noise component has been reduced substantially. The spectral filter will be discussed in more detail in Section 4.2.

The bottom panel of Figure 4.1 shows the information content (that is, the horizontal asymptote of the solid and dashed curves in the top panel of Figure 4.1) as a function of the number of spectral channels. For example, the top panel shows an asymptote in Shannon information of approximately 52 nats. The bottom panel shows a corresponding value in Shannon information of 52 nats for a 1,000-channel sounder. The bottom curve in the bottom panel shows the degrees of freedom due to signal for a vector subspace spanned by eigenvectors principally related to temperature profile variation (see Section 4.2.4). The roll-off of this curve with increasing number of channels suggests that there is little marginal information to be recovered beyond a few thousand channels that is correlated with the temperature profile. This result has been reported by other investigators [6].

4.2 Principal Components Analysis (PCA)

We now demonstrate how the information content in the radiance spectrum can be recovered through the application of a linear transform of minimal rank. A random vector (of atmospheric radiance intensity observations at N frequencies, for example)

$$R \triangleq \begin{bmatrix} R_{\nu_1} \\ R_{\nu_2} \\ \vdots \\ R_{\nu_N} \end{bmatrix} \tag{4.19}$$

can be decomposed into a vector $\mathcal{I}_r$ of r statistically independent components (where $1 \leq r \leq N$)

$$\mathcal{I}_r \triangleq \begin{bmatrix} \mathcal{I}_1 \\ \mathcal{I}_2 \\ \vdots \\ \mathcal{I}_r \end{bmatrix} \triangleq \begin{bmatrix} f_1(R) \\ f_2(R) \\ \vdots \\ f_r(R) \end{bmatrix} \triangleq \mathbf{f}_r(R) \tag{4.20}$$

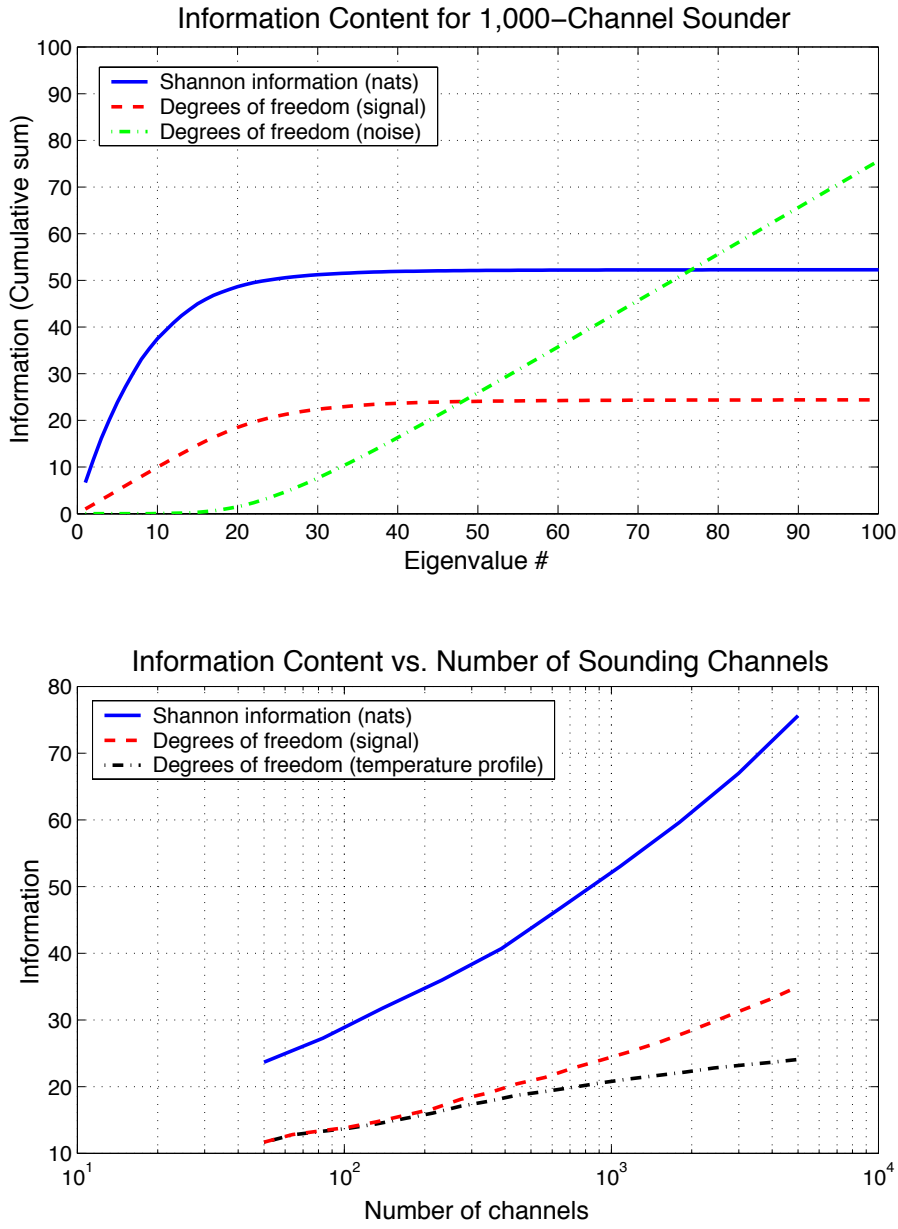


Figure 4.1 Information content analysis of a hypothetical hyperspectral IR (4–15 μm) sounder in clear air. The top graph shows the information content of a 1,000-channel sounder using the Shannon and DOF metrics as a function of eigenvalue number. The bottom graph shows the relationship between information content and the number of sounding channels distributed from 4–15 μm. Also shown is the number of degrees of freedom in the signal due to the temperature profile (see Section 4.2.4).

where $\mathbf{f} : \mathbb{R}^N \rightarrow \mathbb{R}^r$ is a continuous (usually nonlinear) function. The radiance vector R may be reconstructed from the independent components $\mathcal{I}_r$ (possibly with some distortion) as follows:

$$\widehat{R}_r \triangleq \begin{bmatrix} g_1(\mathcal{I}_r) \\ g_2(\mathcal{I}_r) \\ \vdots \\ g_N(\mathcal{I}_r) \end{bmatrix} \triangleq \mathbf{g}_r(\mathcal{I}_r) \tag{4.21}$$

where $\mathbf{g} : \mathbb{R}^r \rightarrow \mathbb{R}^N$ is a continuous (usually nonlinear) function. The vector-valued functions $\mathbf{f}_r(\cdot)$ and $\mathbf{g}_r(\cdot)$ are usually chosen to minimize some scalar-valued cost function

$$C(R - \widehat{R}_r) \tag{4.22}$$

over $1 \leq r \leq N$. Note that $\widehat{R}_r = R$ for $r = N$, and possibly for $r < N$ if the elements of R are statistically dependent. The functions $\mathbf{f}_r(\cdot)$ and $\mathbf{g}_r(\cdot)$ and the statistical moments of $\mathcal{I}_r$ provide a measure of the statistical structure of R.

If the cost function to be minimized is the expected value of the sum of the squares of the error of $R - \widehat{R}_r$, that is,

$$C(\cdot) = E\left[(R - \widehat{R}_r)^T (R - \widehat{R}_r)\right] \tag{4.23}$$

then the elements of $\mathcal{I}_r$ are called *principal components* of R.

4.2.1 Nonlinear PCA

Generally, $\mathbf{f}_r(\cdot)$ and $\mathbf{g}_r(\cdot)$ are nonlinear and cannot be found analytically. Several methods have been proposed for finding $\mathbf{f}_r(\cdot)$ and $\mathbf{g}_r(\cdot)$ using feedforward neural networks given an ensemble of observations of R [7, 8]. An autoassociative feedforward neural network was used in [9] to find $\mathbf{f}_r(\cdot)$ and $\mathbf{g}_r(\cdot)$ for remote sounding data.

4.2.2 Linear PCA

We now consider a special case where $\mathbf{f}_r(\cdot)$ and $\mathbf{g}_r(\cdot)$ are constrained to be linear functions:

$$\mathbf{f}_r(R) = \mathbf{F}R \tag{4.24a}$$

$$\mathbf{g}_r(\mathcal{I}_r) = \mathbf{G}\mathcal{I}_r \tag{4.24b}$$

where $\mathbf{F}$ is an $r \times N$ matrix and $\mathbf{G}$ is an $N \times r$ matrix ($r \leq N$). Equation (4.21) becomes

$$\widehat{R}_r = \mathbf{G}\mathbf{F}R \tag{4.25}$$

and the minimization to be carried out in terms of the cost function given by (4.23) is

$$\{\mathbf{F}, \mathbf{G}\} = \underset{\mathbf{F}, \mathbf{G}}{\arg\min}\ E\left[(R - \mathbf{GF}R)^T(R - \mathbf{GF}R)\right] \tag{4.26}$$

We begin by assuming that $\mathbf{G}$ is orthonormal. If this is not the case, a QR decomposition can be performed on $\mathbf{G}$ (its columns are linearly independent) and the non-orthonormal part can be included in $\mathbf{F}$. The joint minimization posed in (4.26) is separable, and $\mathbf{F}$ can be determined for a fixed choice of $\mathbf{G}$, and vice versa. For a fixed $\mathbf{G}$, R can be decomposed into two orthogonal components:

$$\begin{aligned} R &= (\mathbf{I} - \mathbf{G}\mathbf{G}^T)R + \mathbf{G}\mathbf{G}^T R \\ &= R_\perp + R_\parallel \end{aligned} \tag{4.27}$$

where $\mathbf{I}$ is the identity matrix. Replacing R in (4.26) with (4.28) yields

$$\begin{aligned} & E\left[\left(R_\perp + R_\parallel - \mathbf{GF}R\right)^T\left(R_\perp + R_\parallel - \mathbf{GF}R\right)\right] \\ = \ & E\left[R_\perp^T R_\perp\right] + E\left[\left(R_\parallel - \mathbf{GF}R\right)^T\left(R_\parallel - \mathbf{GF}R\right)\right] \end{aligned} \tag{4.28}$$

where we use the fact that $R_\perp$ is orthogonal to $R_\parallel - \mathbf{GF}R$:

$$\begin{aligned} & E\left[R^T(\mathbf{I} - \mathbf{G}\mathbf{G}^T)(\mathbf{G}\mathbf{G}^T R - \mathbf{GF}R)\right] \\ = \ & E\left[R^T\mathbf{G}\mathbf{G}^T R - R^T\mathbf{G}\mathbf{G}^T R - R^T\mathbf{GF}R + R^T\mathbf{GF}R\right] = 0 \end{aligned} \tag{4.29}$$

For a given $\mathbf{G}$, $E\left[R_\perp^T R_\perp\right]$ does not depend on $\mathbf{F}$, and (4.26) reduces to

$$\mathbf{F} = \underset{\mathbf{F}}{\arg\min}\ E\left[\left(\mathbf{G}\mathbf{G}^T R - \mathbf{GF}R\right)^T\left(\mathbf{G}\mathbf{G}^T R - \mathbf{GF}R\right)\right] \tag{4.30}$$

Upon inspection of (4.30) it is obvious that the choice of $\mathbf{F}$ that minimizes the cost function is

$$\mathbf{F} = \mathbf{G}^T \tag{4.31}$$

Substituting $\mathbf{F} = \mathbf{G}^T$ into (4.26) we obtain

$$\mathbf{G} = \underset{\mathbf{G}}{\arg\min}\ E\left[\left(R - \mathbf{G}\mathbf{G}^T R\right)^T\left(R - \mathbf{G}\mathbf{G}^T R\right)\right] \tag{4.32}$$

where the cost function can be simplified as follows:

$$E\left[\left(R-\mathbf{G}\mathbf{G}^TR\right)^T\left(R-\mathbf{G}\mathbf{G}^TR\right)\right] = E\left[R^TR\right]-E\left[R^T\mathbf{G}\mathbf{G}^TR\right] \tag{4.33}$$

$$= \text{tr}\left(\mathbf{C}_{RR}\right)-\text{tr}\left(\mathbf{G}^T\mathbf{C}_{RR}\mathbf{G}\right) \tag{4.34}$$

where $\mathbf{C}_{RR}$ is the data covariance matrix $E[RR^T]$. Note that the first term in (4.34) does not depend on $\mathbf{G}$, and the minimization in (4.32) is equivalent to the following maximization

$$\mathbf{G} = \underset{\mathbf{G}}{\arg\max}\ \text{tr}\left(\mathbf{G}^T\mathbf{C}_{RR}\mathbf{G}\right) = \underset{\mathbf{G}}{\arg\max}\ \sum_{i=1}^{r} G_i^T\mathbf{C}_{RR}G_i \tag{4.35}$$

where G_i is the ith column of $\mathbf{G}$. The maximization carried out in (4.35) is the well-known quadratic maximization with unit-length constraint problem, which is solved by choosing the G_is to be the r eigenvectors of $\mathbf{C}_{RR}$ with the r largest corresponding eigenvalues

$$\mathbf{G} = [Q_1 \,|\, Q_2 \,|\, \cdots \,|\, Q_r] \tag{4.36}$$

4.2.3 Principal Components Transforms

The *principal components transform* (PCT) is a linear, orthonormal operator $\mathbf{Q}_r^T$ that projects a noisy m-dimensional radiance vector, $\widetilde{R} = R + \Psi$, into an r-dimensional ($r \leq m$) subspace. The additive noise vector Ψ is assumed to be uncorrelated with the radiance vector R, and is characterized by the noise covariance matrix $\mathbf{C}_{\Psi\Psi}$. The "principal components" of $\widetilde{R}$ (i.e., $\widetilde{P} = \mathbf{Q}_r^T\widetilde{R}$) have two desirable properties: (1) the components are statistically uncorrelated, and (2) the reduced-rank reconstruction error[1]

$$c_1(\cdot) = E[(\widehat{\widetilde{R}}_r - \widetilde{R})^T(\widehat{\widetilde{R}}_r - \widetilde{R})] \tag{4.37}$$

where $\widehat{\widetilde{R}}_r \triangleq \mathbf{G}_r\widetilde{R}$ for some linear operator $\mathbf{G}_r$ with rank r, is minimized when $\mathbf{G}_r = \mathbf{Q}_r\mathbf{Q}_r^T$. The rows of $\mathbf{Q}_r^T$ contain the r most significant (ordered by descending eigenvalue) eigenvectors of the noisy data covariance matrix $\mathbf{C}_{\widetilde{R}\widetilde{R}} = \mathbf{C}_{RR} + \mathbf{C}_{\Psi\Psi}$.

1. Note the distinction between $\widehat{R}$, which denotes the estimate of a *noise-free* radiance and $\widehat{\widetilde{R}}$, which denotes the estimate of an observed radiance, which may contain noise.

4.2.3.1 The Noise-Adjusted PCT

Cost criteria other than (4.37) are often more suitable for typical hyperspectral compression applications. For example, it might be desirable to reconstruct the noise-free radiances and filter the noise. The cost equation thus becomes

$$c_2(\cdot) = E[(\widehat{R}_r - R)^T(\widehat{R}_r - R)] \tag{4.38}$$

where $\widehat{R}_r \triangleq \mathbf{H}_r\widetilde{R}$ for some linear operator $\mathbf{H}_r$ with rank r. The noise-adjusted principal components (NAPC) transform [10], where

$$\mathbf{H}_r = \mathbf{C}_{\Psi\Psi}^{1/2}\mathbf{W}_r\mathbf{W}_r^T\mathbf{C}_{\Psi\Psi}^{-1/2} \tag{4.39}$$

and $\mathbf{W}_r^T$ contains the r most-significant eigenvectors of the whitened noisy covariance matrix $\mathbf{C}_{\widetilde{W}\widetilde{W}} = \mathbf{C}_{\Psi\Psi}^{-1/2}(\mathbf{C}_{RR} + \mathbf{C}_{\Psi\Psi})\mathbf{C}_{\Psi\Psi}^{-1/2}$, maximizes the signal-to-noise ratio of each component, and is superior to the PC transform for most noise-filtering applications where the noise statistics are known a priori.

4.2.3.2 Normalized PCT

An alternative to the NAPC transform is the NPC transform, where each element of R is normalized by its standard deviation. The noise-adjusted PCT (NAPCT) is then applied to the normalized R. This transform is often used if the noise statistics are unknown.

4.2.3.3 Blind NAPCT

If the statistics of the noise are unknown, it may be possible to estimate them, and subsequently apply the NAPCT. This approach is called *blind* processing, and attempts to extract properties of $\mathbf{m}(S)$ and Ψ [from (3.1)], usually under the assumption that $\mathbf{m}(S)$ is a linear "mixing matrix" and Ψ is a Gaussian random vector with uncorrelated elements. One example of a blind NAPCT is the blind noise-adjusted principal components transform (BAPCT) [11], which uses the iterated order-noise (ION) algorithm [12] to estimate the noise statistics.

4.2.4 The Projected PC Transform

It is often unnecessary to require that the principal components be uncorrelated, and linear operators can be derived that offer improved performance over PC transforms for minimizing cost functions such as (4.38). It can be

shown using a derivation similar to that presented in Section 4.2.2 that the optimal linear operator with rank r that minimizes (4.38) is

$$\mathbf{L}_r = \mathbf{E}_r \mathbf{E}_r^T \mathbf{C}_{RR} (\mathbf{C}_{RR} + \mathbf{C}_{\Psi\Psi})^{-1} \tag{4.40}$$

where $\mathbf{E}_r = [E_1 \,|\, E_2 \,|\, \cdots \,|\, E_r]$ are the r most-significant eigenvectors of $\mathbf{C}_{RR}(\mathbf{C}_{RR} + \mathbf{C}_{\Psi\Psi})^{-1}\mathbf{C}_{RR}$. Examination of (4.40) reveals that the Wiener-filtered radiances are projected onto the r-dimensional subspace spanned by $\mathbf{E}_r$. It is this projection that motivates the name "projected principal components." This method is very similar to the concept of canonical correlations [13] introduced by Hotelling over 70 years ago [14]. An orthonormal basis for this r-dimensional subspace of the original m-dimensional radiance vector space $\mathcal{R}$ is given by the r most-significant right eigenvectors, $\mathbf{V}_r$, of the reduced-rank linear regression matrix, $\mathbf{L}_r$, given in (4.40). We then define the projected principal components of $\widetilde{R}$ as

$$\widetilde{P} = \mathbf{V}_r^T \widetilde{R} \tag{4.41}$$

Note that the elements of $\widetilde{P}$ are correlated, as $\mathbf{V}_r^T(\mathbf{C}_{RR} + \mathbf{C}_{\Psi\Psi})\mathbf{V}_r$ is not a diagonal matrix.

Another useful application of the PPC transform is the compression of spectral radiance information that is correlated with a geophysical parameter, such as the temperature profile. The r-rank linear operator that captures the most radiance information that is correlated to the temperature profile is similar to (4.40) and is given here:

$$\mathbf{L}_r = \mathbf{E}_r \mathbf{E}_r^T \mathbf{C}_{TR} (\mathbf{C}_{RR} + \mathbf{C}_{\Psi\Psi})^{-1} \tag{4.42}$$

where $\mathbf{E}_r = [E_1 \,|\, E_2 \,|\, \cdots \,|\, E_r]$ are the r most-significant eigenvectors of $\mathbf{C}_{TR}(\mathbf{C}_{RR} + \mathbf{C}_{\Psi\Psi})^{-1}\mathbf{C}_{RT}$, and $\mathbf{C}_{TR}$ is the cross-covariance of the temperature profile and the spectral radiance.

Figure 4.2 compares the performance of the PC, NAPC, and PPC transforms for three specific decompositions/reconstructions of a simulated 1,000-channel radiance vector:

1. Noisy radiance:
$$\widehat{\widetilde{R}} = \mathbf{Q}_r \mathbf{Q}_r^T \widetilde{R} \tag{4.43}$$
2. Signal portion of noisy radiance:
$$\widehat{R} = \mathbf{Q}_r \mathbf{Q}_r^T \widetilde{R} \tag{4.44}$$
3. Temperature profile retrieval:
$$\widehat{T} = \mathbf{C}_{T\widetilde{R}} \mathbf{Q}_r \left[\mathbf{Q}_r^T \mathbf{C}_{\widetilde{R}\widetilde{R}} \mathbf{Q}_r\right]^{-1} \mathbf{Q}_r^T \widetilde{R} \tag{4.45}$$

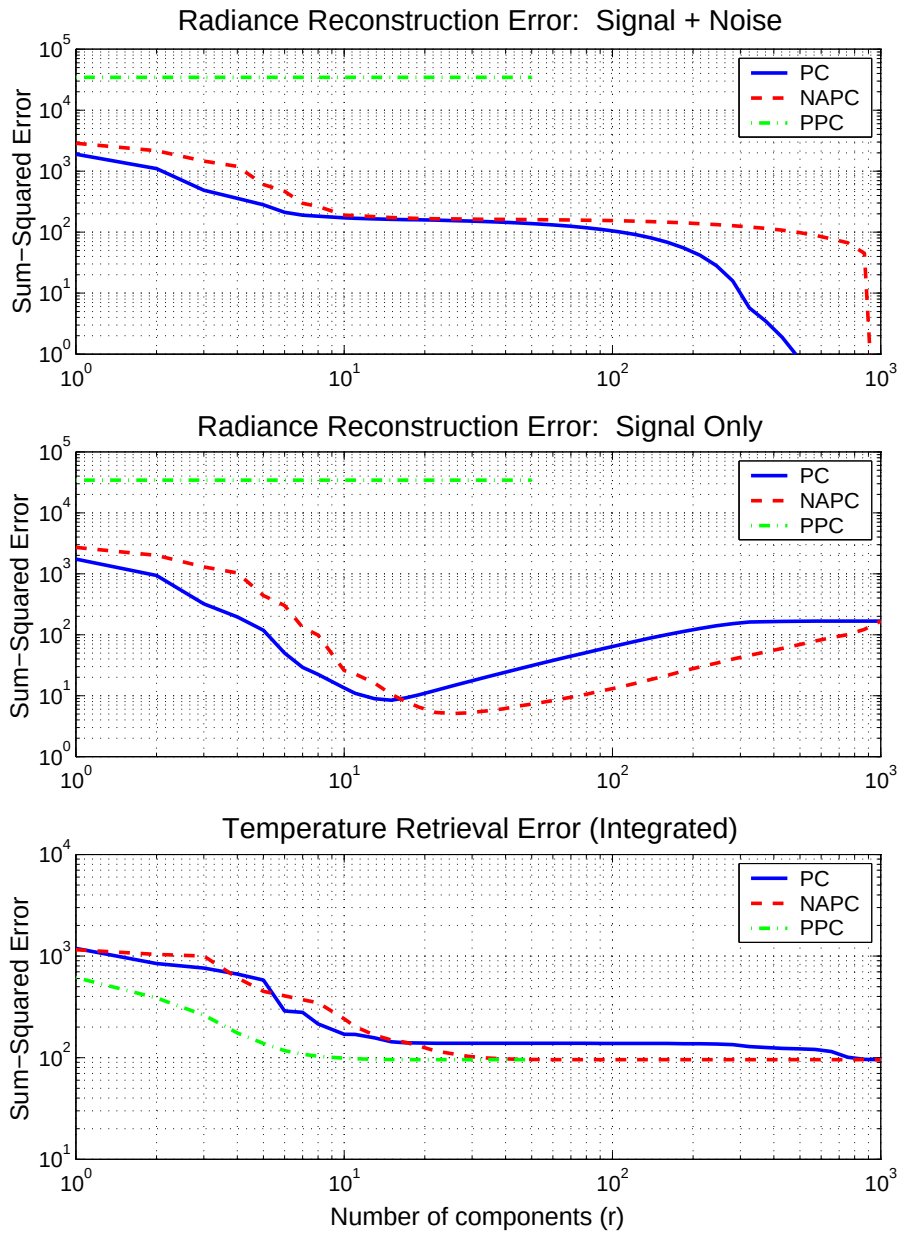


Figure 4.2 Performance comparisons of the PC, NAPC, and PPC transforms for a hypothetical 1,000-channel infrared (4–15 μm) sounder. The first plot shows the distortion introduced by representing a noisy radiance vector with r components. The second plot shows the distortion of the signal portion of the radiance. The third plot shows the integrated sum-squared error of the temperature profile estimated using r components.

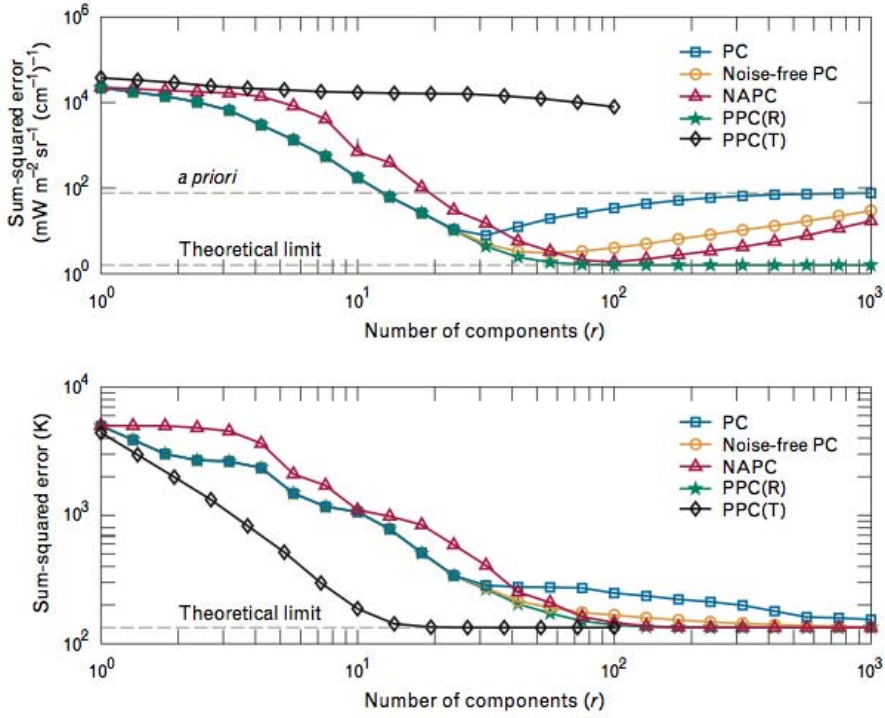


Figure 4.3 Performance comparisons of the principal components, in which the components are derived from both noisy and noise-free radiances, noise-adjusted principal components (NAPC), and projected principal components (PPC) transforms for a hypothetical 1,750-channel infrared (4μm to 15μm) sounder. Two projected principal components transforms were considered, PPC(R) and PPC(T), which are, respectively: (1) maximum representation of noise-free radiance energy, and (2) maximum representation of temperature profile energy. The upper plot shows the sum-squared error of the reduced-rank reconstruction of the noise-free spectral radiances. The lower plot shows the temperature-profile retrieval error (trace of the error covariance matrix) obtained by using linear regression with r components. The appropriate PPC transforms are more efficient, using fewer components to reach the theoretical limits.

4.2.5 Evaluation of Radiance Compression Performance Using Two Different Metrics

The compression performance of each of the PC transforms discussed previously was evaluated using two performance metrics. First, we seek the transform that yields the best (in the sum-squared sense) reconstruction of

the noise-free radiance spectrum given a noisy spectrum. Thus, we seek the optimal reduced-rank linear filter. The second performance metric is quite different and is based on the temperature retrieval performance in the following way. A radiance spectrum is first compressed using each of the PC transforms for a given number of coefficients. The resulting coefficients are then used in a linear regression to estimate the temperature profile.

The results that follow were obtained using simulated, clear-air radiance intensity spectra from an AIRS-like sounder. Approximately 7,500 1,750-channel radiance vectors were generated with spectral coverage from approximately 4 μm to 15 μm using the NOAA88b radiosonde set. The simulated intensities were expressed in spectral radiance units (mW m^{-2} sr^{-1}(cm^{-1})$^{-1}$).

4.2.5.1 PC Filtering

Figure 4.3 shows the sum-squared radiance distortion (4.37) as a function of the number of components used in the various PC decomposition techniques. The a priori level indicates the sum-squared error due to sensor noise. Results from two variants of the PC transform are plotted, where the first variant (the "PC" curve) uses eigenvectors of $\mathbf{C}_{\widetilde{R}\widetilde{R}}$ as the transform basis vectors, and the second variant (the "noise-free PC" curve) uses eigenvectors of $\mathbf{C}_{RR}$ as the transform basis vectors. It is shown in Figure 4.3 that the PPC reconstruction of noise-free radiances (PPC[R]) yields lower distortion than both the PC and NAPC transforms for any number of components (r). It is noteworthy that the "PC" and "noise-free PC" curves never reach the theoretically optimal level, defined by the full-rank Wiener filter. Furthermore, the PPC distortion curves decrease monotonically with coefficient number, while all the PC distortion curves exhibit a local minimum, after which the distortion increases with coefficient number as noisy, high-order terms are included. The noise in the high-order PPC terms is effectively zeroed out, because it is uncorrelated with the spectral radiances.

4.2.5.2 PC Regression

The PC coefficients derived in the previous example are now used in a linear regression to estimate the temperature profile. Shown in Figure 4.3 is the temperature profile error (integrated over all altitude levels) as a function of the number of coefficients used in the linear regression, for each of the PC transforms. To reach the theoretically optimal value achieved by linear regression with all channels requires approximately 20 PPC coefficients, 200 NAPC coefficients, and 1,000 PC coefficients. Thus, the PPC transform results in a factor of ten improvement over the NAPC transform when compressing temperature-correlated radiances (20 versus 200 coefficients required), and

approximately a factor of 100 improvement over the original spectral radiance vector (20 versus 1,750). Results for the moisture profile are similar, although more coefficients (typically 35 versus 25 for temperature) are needed because of the higher degree of nonlinearity in the underlying physical relationship between atmospheric moisture and the observed spectral radiance. This substantial compression enables the use of relatively small (and thus very stable and fast) neural network estimators to retrieve the desired geophysical parameters.

It is interesting to consider the two variants of the PPC transform shown in Figure 4.3, namely PPC(R), where the basis for the noise-free radiance subspace is desired, and PPC(T), where the basis for only the temperature profile information is desired. As shown in Figure 4.3, the PPC(T) transform poorly represents the noise-free radiance space, because there is substantial information that is uncorrelated with temperature (and thus ignored by the PPC(T) transform) but correlated with the noise-free radiance. Conversely, the PPC(R) transform offers a significantly less compact representation of temperature profile information (see Figure 4.3), because the transform is representing information that is not correlated with temperature and thus superfluous when retrieving the temperature profile.

4.3 Representation of Nonlinear Features

We have demonstrated that hyperspectral remote sensing data can be highly correlated, and we have introduced linear techniques that can be used to represent the information content in more statistically compact form. There are many other features in atmospheric remote sensing data that can be exploited, either to achieve a more efficient representation of information or perhaps to highlight and/or isolate a particular feature of interest. Some atmospheric characteristics are nonlinear functions of time, spectral wavelength, or state parameter, and it is useful in these cases to expand the pre- and post-processing operators to include nonlinear functions, and neural networks can be used effectively for this purpose.

Periodic variability is an example of a nonlinear feature in atmospheric remote sensing data that can be represented prior to the estimation of a state parameter. Some familiar examples of cyclical variations include that of temperature with time of day and season. It may be possible to develop robust algorithms for estimating atmospheric state without considering cyclical variations. However, given the size of contributions of cyclical variables, algorithms that account for cyclical variations may offer a significant advantage over those that do not. We will discuss in Chapter 7 pre- and post-processing operators that work in concert with the neural network estimators

to represent cyclical variability and improve performance while allowing the network architecture to be simplified.

4.4 Summary

We have described the mathematical nature of atmospheric remote sensing data and presented methods that are useful for characterization and improved representation of the information contained in the measurements. The number of degrees of freedom in the signal gives a metric for the minimum number of linear basis functions that must be used to represent the information content. Principal components transforms provide a framework for deriving optimal linear basis functions and filtering unwanted interfering signals. We presented examples of random noise removal in this chapter, but many other applications are also possible (for example, see [15] for a discussion of the representation of cloud effects). Periodic signals can also be exploited to improve the accuracy and efficiency of the retrieval system.

References

[1] R. Bellman. *Adaptive Control Processes: A Guided Tour*. Princeton University Press, Princeton, New Jersey, 1961.

[2] C. E. Shannon. "A mathematical theory of communication." *Bell System Technical Journal*, 27:379–423, 1948.

[3] C. D. Rodgers. *Inverse Methods for Atmospheric Sounding*. World Scientific, New York, 2000.

[4] G. Strang. *Linear Algebra and Its Applications*. Academic Press, New York, 1980.

[5] L. Strow, S. Hannon, and S. Desouza-Machado. "An overview of the AIRS radiative transfer model." *IEEE Trans. Geosci. Remote Sens.*, 41(2), February 2003.

[6] C. D. Rodgers. "Information content and optimisation of high spectral resolution measurements." *SPIE, Optical Spectroscopic Techniques and Instrumentation for Atmospheric and Space Research II*, 2830:136–147, 1996.

[7] M. A. Kramer. "Nonlinear principal component analysis using autoassociative neural networks." *AIChE*, 37(2):233–243, 1991.

[8] S. Tan and M. Mavrovouniotis. "Reducing data dimensionality through optimizing neural-network inputs." *AIChE*, 41(6):1471–1480, 1995.

[9] A. J. Slone. *Improved Remote Sensing Data Analysis Using Neural Networks*. Master's thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, September 1995.

[10] J. B. Lee, A. S. Woodyatt, and M. Berman. "Enhancement of high spectral resolution remote-sensing data by a noise-adjusted principal components transform." *IEEE Trans. Geosci. Remote Sens.*, 28:295–304, May 1990.

[11] J. H. Lee. *Blind Noise Estimation and Compensation for Improved Characterization of Multivariate Processes*. Ph.D. thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, 2000.

[12] J. Lee and D. H. Staelin. "Iterative signal-order and noise estimation for multivariate data." *IEE Electronics Letters*, 37(2):134–135, 2001.

[13] D. S. Wilks. *Statistical Methods in the Atmospheric Sciences*. Elsevier, New York, second edition, 2006.

[14] H. Hotelling. "The most predictable criterion." *Journal of Educational Psychology*, 26:139–142, 1935.

[15] W. J. Blackwell. "Neural network retrievals of atmospheric temperature and moisture profiles from high-resolution infrared and microwave sounding data." *Signal and Image Processing for Remote Sensing*, C. C. Chen, Ed., Chapter 11, Taylor and Francis, Boca Raton, Florida, 2006.

5

Introduction to Multilayer Perceptron Neural Networks

Artificial neural networks, or neural nets, are computational structures inspired by biological networks of densely connected neurons, each of which is capable only of simple computations. Just as biological neural networks are capable of learning from their environment, neural nets are able to learn from the presentation of training data, as the free parameters (weights and biases) are adaptively tuned to fit the training data. Neural nets can be used to learn and compute functions for which the analytical relationships between inputs and outputs are unknown and/or computationally complex and are therefore useful for pattern recognition, classification, and function approximation. Neural nets are particularly appealing for the inversion of atmospheric remote sensing data, where relationships are commonly nonlinear and non-Gaussian, and the physical processes may not be well understood.

In this book, we focus on feedforward multilayer perceptron (FFMLP) neural networks due to their simplicity, flexibility, and ease of use. Multilayer neural networks most often consist of an input layer, one or more nonlinear hidden layers, and a linear output layer. The mathematical functions implemented by FFMLP nets are continuous and differentiable, which greatly simplifies training and error analysis. But perhaps the most useful attribute of neural nets is their scalability; a network with a sufficient number of weights and biases is capable of approximating a bounded, continuous function to an arbitrary level of precision over a finite domain [1]. Therefore, neural networks can be used as universal function approximators.

The discussion of nonlinear regression in Section 3.3.2.2 illustrated that a linear combination of simple, nonlinear functions (polynomials) could be used to construct powerful nonlinear estimators. However, no guidance was

provided for the selection of these functions or the determination of how many should be used. Neural networks are constructed in a similar fashion: simple computational elements (nodes) are connected in a way that allows complex functions to be synthesized by adding nodes to the network. The distinctive features of neural networks in this context are the "activation function" applied by each of the nodes, the number of nodes, a detailed description of the node connections in the network (the topology), and the algorithm used to derive the weights and biases (training). We provide a general discussion of these features in this chapter and present additional guidance on network training and performance evaluation in Chapters 6 and 8, respectively.

5.1 A Brief Overview of Machine Learning

Machine learning algorithms attempt to identify patterns and interrelationships among variables in a data set, usually by the use of some form of inductive generalization. The field of machine learning is vast and interdisciplinary, encompassing fields from biology, mathematics, computer science, and engineering, and we will therefore provide only a cursory review of the salient issues and considerations. The interested reader is encouraged to consult excellent references [2–5] for further information.

5.1.1 Supervised and Unsupervised Learning

Learning algorithms extract mathematical features and characteristics from a set of training data, and the "learning" is often enabled either by some kind of reinforcement or competition. Learning can be either supervised or unsupervised. Supervised learning uses pairs of data arranged as inputs and targets. Each input has associated with it a target, and the learning algorithm infers the relationship between the inputs and the targets as the training proceeds. Multilayer perceptron networks and support vector machines are examples of supervised learning. Unsupervised learning methods do not require input–target pairs; the algorithm itself decides what target is best for a given input and organizes accordingly. Kohonen self-organizing feature maps and Hopfield networks are examples of unsupervised learning [6, 7]. Semisupervised approaches are also possible, where both types of learning are used within the same algorithm.

5.1.2 Classification and Regression

Machine learning problems typically fall into one of two categories: classification, where the input vector is assigned membership in one of a

number of finite groups, or regression (function approximation), where a multidimensional "fitting function" is found that maps the input vector to an output vector that closely approximates the target. In the geophysical remote sensing context, machine-based classifiers can be used for a variety of purposes, including classifying surface types, monitoring of agriculture, forestry, and ecology, and exploring for minerals and petroleum. These and similar attributes are best measured with imaging sensors.

This book focuses on determination of atmospheric parameters, where classification operators are of less utility, although the classification of cloud and precipitation types are two notable exceptions. We therefore emphasize regression methods of machine learning in atmospheric remote sensing. Examples include the determination of atmospheric temperature and water vapor profiles from measurements of upwelling atmospheric emission in the thermal infrared, retrieval of wind speed and direction from passive polarimetric microwave observations of oceans, and estimation of precipitation intensity from millimeter-wave measurements in opaque spectral bands.

5.1.3 Kernel Methods

A significant part of machine learning involves the reorganization of the input space into a "feature space" that facilitates the extraction of the necessary characteristics from the input variables. The mathematical rationale for this remapping stems largely from Cover's theorem on the separability of patterns [8], which essentially says that data cast nonlinearly into a high-dimensional feature space is more likely to be linearly separable there than in a lower-dimensional space. Recall that in Section 3.3.2.1 we observed that linear regression effectively estimates the atmospheric temperature profile from hyperspectral infrared sounding measurements, even though the physical relationship between the temperature and the observations is nonlinear due to the Planck function and atmospheric absorption mechanisms. The high level of performance achieved by linear regression for this nonlinear problem can be explained in part by Cover's theorem – the temperature profile is effectively recast in a high-dimensional nonlinear space by the radiative transfer equation (2.60).

As an example, suppose we are given a training set of n_{train} input vectors X_i of dimension d_x and target vectors T_i of dimension d_t, respectively:

$$(X_1, T_1), \ldots, (X_n, T_n) \in \mathbb{R}^{d_x} \times \mathbb{R}^{d_t} \tag{5.1}$$

Given an unseen (that is, not a member of the training set) input X, we want to choose the estimate Y such that (X, Y) is in some sense similar to the

relationships exhibited by the training data. We will find it useful to define a nonlinear mapping of X into a *feature space* of potentially much higher dimension, $\mathcal{F} \in \mathbb{R}^{d_f}$, where d_f (the dimension of the feature space) is much greater than d_x. We will use Φ to denote this mapping:

$$X \mapsto \Phi(X) \tag{5.2}$$

and refer to Φ as the *feature map*. Although this nonlinear mapping into a high dimensional space adds complexity to the numerical optimization, we can simplify the problem by defining a similarity measure using a *kernel* that generates simple dot products of the feature space:

$$k(X, X') = \langle \Phi(X), \Phi(X') \rangle \tag{5.3}$$

where we have used $\langle \cdot \rangle$ to denote the dot (inner) product. The advantage of using such a kernel as a similarity measure is that it allows the construction of algorithms in dot product spaces [9], and for a given learning problem we now consider the same algorithm in $\mathcal{F}$ instead of $\mathbb{R}^{d_x}$, that is, we now work with the sample

$$(\Phi(X_1), T_1), \ldots, (\Phi(X_n), T_n) \in \mathbb{R}^{d_f} \times \mathbb{R}^{d_t} \tag{5.4}$$

The use of kernels defined on dot product spaces also allows optimality to be guaranteed under some conditions. Once a suitable kernel and feature map have been chosen to satisfy (5.3) (all positive definite kernels qualify), the objective then becomes to find a simple classification or regression in $\mathcal{F}$.

We will see that support vector machines and neural networks with a single hidden layer are special cases of kernel methods. Common kernel functions are Gaussian "radial basis functions," sigmoids, and polynomials, although sigmoidal kernel functions do not strictly satisfy (5.3) [10].

5.1.4 Support Vector Machines

Numerical methods are often needed to solve nonlinear classification and regression problems. Support vector machines are constructed so that the resulting equations to be optimized have two desirable properties. First, the optimization problem is *convex*, thus guaranteeing a unique solution that can be found using standard quadratic programming techniques. Second, only a small subset of the training data is active in adaptively tuning the weights (although all the training data is used during training); the vectors comprising the subset of training data (found as part of the optimization process) that determine the optimal parameters are called the *support vectors*. Support vector regression techniques have recently been used to solve a variety of

geophysical inversion problems (see [11–13], for example), and we now review the basic methodology.

Support vector machines perform function approximation through the application of linear regression on a chosen feature mapping, $\Phi(X)$:

$$f(X) = \langle W, \Phi(X) \rangle + b \tag{5.5}$$

where W and b are coefficients to be estimated during training. The cost function to be minimized in the case of support vector machines is quite unique. First, only deviations of the outputs from the targets that exceed a magnitude of ϵ are penalized, and then only linearly. Second, a *flatness* criterion is imposed such that the coefficients W are chosen to have minimum Euclidean length (2-norm). The relative contribution of these two criteria to the cost function is governed by a trade-off parameter λ. Mathematically, the cost function can be expressed as

$$c(W) = \lambda \sum_{i=1}^{n_{\text{train}}} |f(X_i) - T_i|_\epsilon + \frac{1}{2}\|W\|^2 \tag{5.6}$$

where Vapnik's ϵ-insensitive loss function [2] is

$$|f(X_i) - T_i|_\epsilon = \begin{cases} |f(X) - T| - \epsilon & \text{for } |f(X_i) - T_i| \geq \epsilon \\ 0 & \text{otherwise} \end{cases} \tag{5.7}$$

and the summation is taken over all n_{train} elements in the training set. Lagrange multipliers α and α^* can be used to cast the minimizing function in terms of a kernel function satisfying (5.3) as described previously:

$$f(X, \alpha, \alpha^*) = \sum_{i=1}^{n_{\text{train}}} (\alpha_i - \alpha_i^*) k(X_i, X) + b \tag{5.8}$$

with $\alpha_i \alpha_i^* = 0$ and $\alpha_i, \alpha_i^* \geq 0$ for all i. The coefficients α_i, α_i^* are obtained by maximizing the following:

$$-\frac{1}{2} \sum_{i,j=1}^{n_{\text{train}}} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) k(X_i, X_j) - \epsilon \sum_{i=1}^{n_{\text{train}}} (\alpha_i + \alpha_i^*) + \sum_{i=1}^{n_{\text{train}}} T_i(\alpha_i - \alpha_i^*) \tag{5.9}$$

subject to

$$\sum_{i=1}^{n_{\text{train}}} (\alpha_i - \alpha_i^*) = 0 \text{ and } \alpha_i, \alpha_i^* \in [0, \lambda] \tag{5.10}$$

The Karush-Kuhn-Tucker conditions state that at the point of the solution the product between dual variables and constraints has to vanish [10]. Therefore, α_i, α_i^* vanish for all samples for which the magnitude of the error deviation is less than ϵ, and we have a sparse expansion of W in terms of X_i (i.e., we do not need all X_i to describe W). The examples that come with non-vanishing coefficients are the support vectors.

An advantage of the support vector machine methodology is that once ϵ, λ, and $k(X, X')$ have been defined, the algorithm proceeds to find the optimal mapping. There is no need to specify algorithm *topology* (for example, the number of support vectors and how they are used to compute the outputs) – this is determined by the algorithm as part of the training process.

5.1.5 Feedforward Neural Networks

Feedforward neural networks propagate the inputs (the *input layer*) through a set of computational nodes arranged in layers to calculate the network outputs. The *output layer* is the final layer of the neural network and usually contains linear elements. The layers between the input layer and the output layer are called *hidden layers* and usually contain nonlinear elements. This network topology is depicted graphically in Figure 5.1. The various types of feedforward neural networks differ primarily in the nonlinear functions (the so-called *activation functions*) that are used in the hidden layer nodes and the training algorithms that are used to optimize the free parameters of the network. In general, the connections shown in Figure 5.1 need not be fully populated: some optimization strategies start with a large number of hidden nodes and "prune" the network by eliminating connections, and possibly nodes, as training progresses. In this book, we will not consider "recurrent" networks with feedback, where the output from a forward layer is used as an input in a previous layer (see the lower dashed line in Figure 5.1). We will also require that outputs from a given layer are used as inputs only in the next layer; "layer skipping," as shown with the upper dashed line in Figure 5.1 is not considered.

5.1.5.1 Multilayer Perceptron Networks

The perceptron is the basic structural element of feedforward multilayer perceptron networks. The inputs to a perceptron are weighted, summed over the n inputs, translated, and passed through an activation function. The perceptron is shown graphically in Figure 5.2, and the transfer function can be written as follows:

$$y = f\left(\sum_{i=1}^{n} w_i x_i + b\right) \tag{5.11}$$

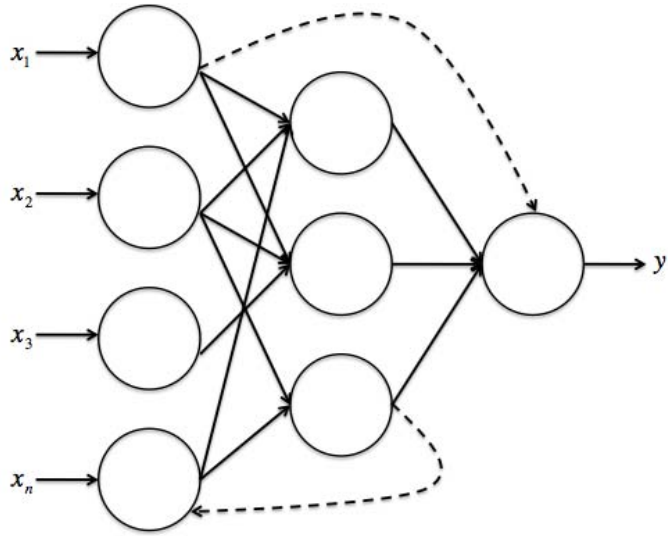


Figure 5.1 The general structure of a multilayer feedforward neural network is shown, including forward connections between successive layers. The two dashed lines indicate connections that, while possible in general multilayer networks, are not considered in this book.

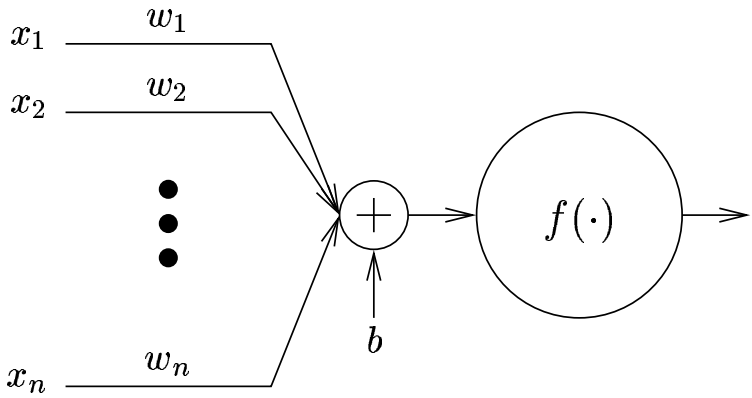


Figure 5.2 The perceptron weights and sums the inputs, applies a bias, and applies a nonlinear activation function.

where x_i is the ith input, w_i is the weight associated with the ith input, b is the bias, $f(\cdot)$ is the activation function of the perceptron, and y is the output.

The activation functions are generally chosen to be strictly increasing, smooth (continuous first derivative), and asymptotic. Perceptrons with sigmoidal (soft limit) activation functions are commonly used in the hidden layer(s), and the identity function is used in the output layer. The logistic function

$$f(x) = \frac{1}{1 + e^{-x}} \tag{5.12}$$

with first derivative $f'(x) = f(x) - f^2(x)$ can be used as a sigmoidal activation function. However, a multilayer perceptron trained with the backpropagation algorithm may, in general, learn faster when the activation function is antisymmetric, that is, $f(-x) = -f(x)$. The logistic function is not antisymmetric, but can be made antisymmetric by a simple scaling and shifting, resulting in the hyperbolic tangent function

$$f(x) = \tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{5.13}$$

with first derivative $f'(x) = 1 - f^2(x)$. These linear and hyperbolic tangent activation functions are shown in Figure 5.3. The simple form of sigmoidal function and its derivative allows fast and accurate calculation of the gradients needed to optimize selection of the weights and biases and carry out second-order error analysis. These topics will be covered in Chapters 6 and 8, respectively.

5.1.5.2 Radial Basis Function Networks

Multilayer perceptron networks, while powerful, often have complicated error surfaces and therefore higher likelihoods of suboptimal training and instability. Radial basis function networks use simple activation functions that tend to be localized. This simplicity and localization reduces the complexity of the error surfaces, but many nodes are needed to represent features that are active over large regions of the input space

Radial basis functions are based on the distance metric

$$r = \|X - X_i\| \tag{5.14}$$

and are therefore applicable to a wide range of problems in machine learning ranging from pattern recognition, function approximation, interpolation, and mixture modeling. Common radial basis functions (with width parameter σ) include:

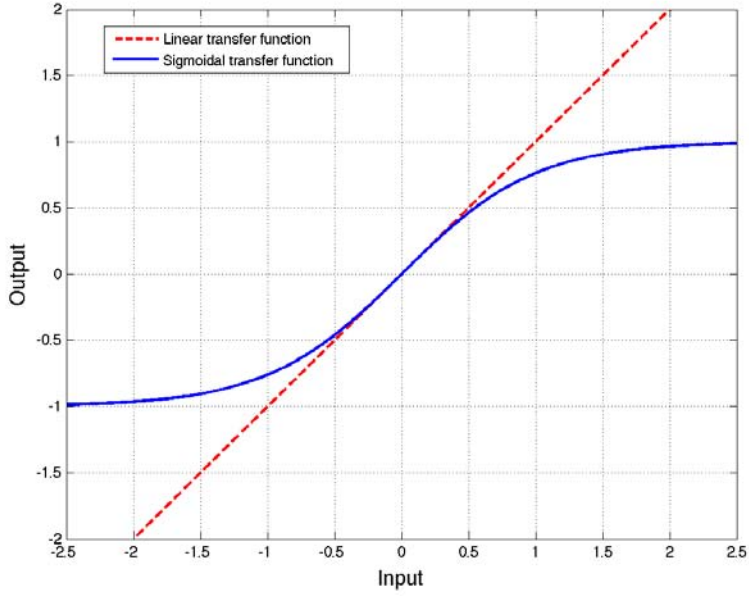


Figure 5.3 Two common neural net transfer functions are shown: hyperbolic tangent (solid curve) and linear (dashed curve).

Multiquadric:	$\phi(r) = \sqrt{r^2 + \sigma^2}$
Inverse multiquadric:	$\phi(r) = 1/\sqrt{r^2 + \sigma^2}$
Gaussian:	$\phi(r) = \mathrm{e}^{(-r^2/2\sigma^2)}$

and more sophisticated functions can be readily constructed by replacing the Euclidean distance metric given in (5.14) by the Mahalanobis distance metric:

$$r_m = \sqrt{(X - X_i)^T \mathbf{C}_{XX}^{-1} (X - X_i)} \tag{5.15}$$

Most radial basis functions are quasi-orthogonal, that is, the product of two basis functions, whose centers are far away from each other with respect to their widths, is almost zero. If we collect the scalar basis functions ϕ_i (each with a center X_i and width σ_i) into a vector basis function $\Phi(X)$, we can estimate the target function as follows:

$$Y = \mathbf{f}(X) = \mathbf{W}\Phi(X) \tag{5.16}$$

where each row of the weighting matrix $\mathbf{W}$ assigns a linear combination of the basis functions to an output. Given m basis functions and n dimensions in

the output vector, the size of the weighting matrix $\mathbf{W}$ is $n \times m$. The training algorithm determines $\mathbf{W}$, X_i, and σ_i by minimizing a cost function, usually a form of sum-squared error of the network outputs relative to the targets:

$$C(\cdot) = \|Y - T\|^2 \tag{5.17}$$

Radial basis function networks are commonly trained in two stages. Unsupervised methods are used to determine the basis function parameters, and fast, linear supervised methods are used to optimize the output weights [14, 15].

In the remainder of the book, we consider feedforward multilayer perceptron networks exclusively due to the balance of capability, flexibility, and simplicity that they provide. Care must be taken, however, in the selection of network topology, network training, and evaluation of performance to ensure optimal performance. We discuss these topics presently and in the next few chapters.

5.2 Feedforward Multilayer Perceptron Neural Networks

Perceptrons can be combined to form a multilayer network. In this type of network, individual perceptrons are arranged in layers, and the perceptrons in each layer all use the same transfer function. The inputs to the network are fed to every node of the first layer, and the outputs of each layer (except the output layer) are fed to every node of the next layer.

An example of a two-layer network (that is, one hidden layer and one output layer) is shown in Figure 5.4. In Figure 5.4, x_i is the ith input, n is the number of inputs, w_{ij} is the weight associated with the connection from the ith input to the jth node in the hidden layer, b_i is the bias of the ith node, m is the number of nodes in the hidden layer, $f(\cdot)$ is the transfer function of the perceptrons in the hidden layer, v_i is the weight between the ith node and the output node, c is the bias of the output node, $g(\cdot)$ is the transfer function of the output node, and y is the output. We can then relate the network output to the inputs as follows:

$$y = g\left(\sum_{j=1}^{m} v_j f\left(\sum_{i=1}^{n} w_{ij} x_i + b_j\right) + c\right) \tag{5.18}$$

5.2.1 Network Topology

The range of functions that can be approximated by a neural network is determined by its topology and the transfer functions of each layer.

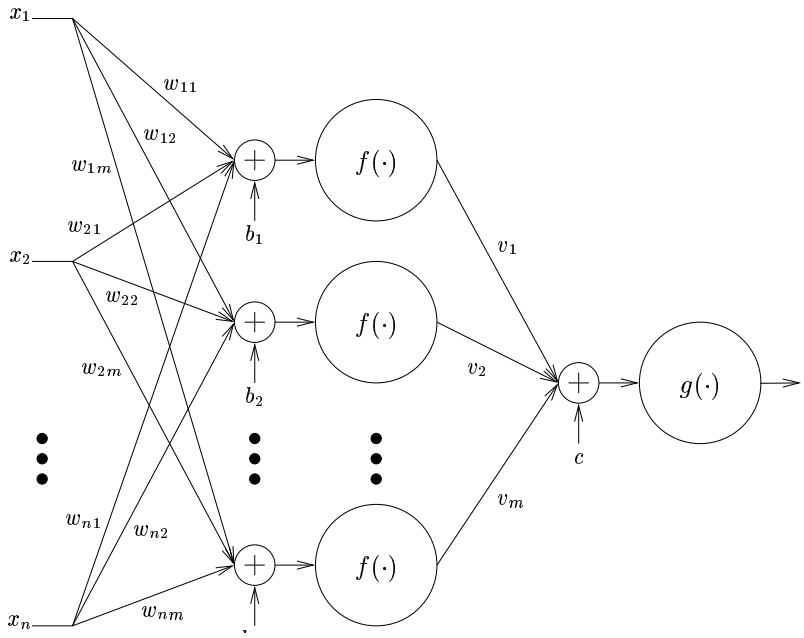


Figure 5.4 A feedforward neural network with one hidden layer and one output node.

Moderately nonlinear functions are usually well approximated by a single hidden layer of nodes and a linear output layer. Two or more hidden layers can be used to represent functions with severe nonlinearities. In the atmospheric remote sensing context, temperature profile retrievals from microwave and infrared sounding data are moderately nonlinear, but retrievals involving water vapor, clouds, and precipitation are sufficiently nonlinear to require multiple hidden layers. Simple examples at the end of this chapter illustrate the selection of topological parameters and provide background for forthcoming discussions of real-world problems.

We use the following nomenclature for network topology. N_{IN}–N_{H1}–N_{H2}–N_{out} refers to the number of nodes in the input layer, first and second hidden layers, and the output layer, respectively. For example, a "4–5–3–2 network" denotes four inputs, a first hidden layer with five nodes, a second hidden layer with three nodes, and an output layer with two nodes.

5.2.2 Network Training

The process of deriving the network weights and biases to best fit the ensemble of input and target vectors is called training. The components of network training involve assembly of the data set, selection of network topology, network initialization, and optimization of weights and biases (including regularization, if necessary). Once the network is trained, it is imperative that performance evaluation and error analysis techniques are used to ensure the network generalizes well (that is, produces a reasonable output for an unseen input) and is relatively insensitive to data artifacts, which may include sensor noise or interfering geophysical signals. We cover each of these topics in detail in the next several chapters – here we introduce the key concepts and point out the challenges and pitfalls to be addressed by the forthcoming discussion.

5.2.2.1 Initialization

Numerical optimization methods are often initialized to appropriate starting values from which optimization proceeds. Initialization for neural network training is especially important because the error surfaces are often complex. The general objective when initializing the weight and bias values is to maximally span the search space and exercise all of the available information in the input and target data. This initialization is typically carried out by assigning random values to the weights and biases. Substantial improvements to training time and resistance to local minima can be achieved by selecting the initial weight and bias values so that the active regions of all node transfer functions are utilized when training begins. The Nguyen-Widrow initialization method follows this approach and is discussed in Chapter 6.

5.2.2.2 Backpropagation Learning

After initialization, the weights and biases are tuned to best represent the relationships present in the training set. The sigmoidal activation functions are continuous and differentiable and are thus amenable to optimization algorithms based on gradient descent. Backpropagation learning is one such algorithm. The simplest implementation of backpropagation updates the network weights and biases in the direction in which the cost function decreases most rapidly, the negative of the gradient. The backpropagation algorithm calculates updates efficiently by propagating the errors back through the network (thus the name “backpropagation”). The updates can be calculated incrementally for each input–target pair or by applying all of the available training data at once using “batch” training. In this book the

Levenberg-Marquardt backpropagation algorithm is used. Several variations of this algorithm are discussed in Chapter 6.

5.2.2.3 Regularization

Regularization techniques can improve network *stability*, which is the *generalization* and *interference immunity* of a network. Network generalization can be enhanced in two ways: (1) increasing the size of the training data set and ensuring that it is both *global* and *comprehensive*, and (2) using regularization techniques to prevent the network from overfitting the training data. Common regularization techniques to combat overfitting include early stopping, weight decay, and weight pruning.

The second component of stability is the immunity of the network to interfering signals, such as sensor noise or cloud contamination. Interference immunity can be improved by judicious choices of preprocessing (discussed in Chapter 7). Regularization approaches can also be used. For example, random noise can be added to the input vectors at each training epoch (an *epoch* is a single iteration of the training algorithm) to model and therefore help desensitize the network to sensor noise.

5.3 Simple Examples

In this section, we explore the range of functions that can be approximated with the neural network structure described in the previous sections. The first two examples involve only single perceptrons, and the remaining examples involve multilayer networks, all of which are trained using backpropagation techniques. For this section, we are interested only in the range of functions possible for a given topology.

5.3.1 Single-Input Networks

5.3.1.1 Example: A Single Perceptron with One Input

We begin with a single hyperbolic tangent sigmoid perceptron with a single input x. This node has a weight that connects it with the input and also has a bias b. Its output y is then:

$$y = \tanh(wx + b) \tag{5.19}$$

When $w = 1$ and $b = 0$, $y = \tanh x$ as shown in Figure 5.5. As w varies, the slope and domain of the "active" region of the curve vary (Figure 5.5(a)). As b varies, the curve of Figure 5.5(b) shifts to the left or right.

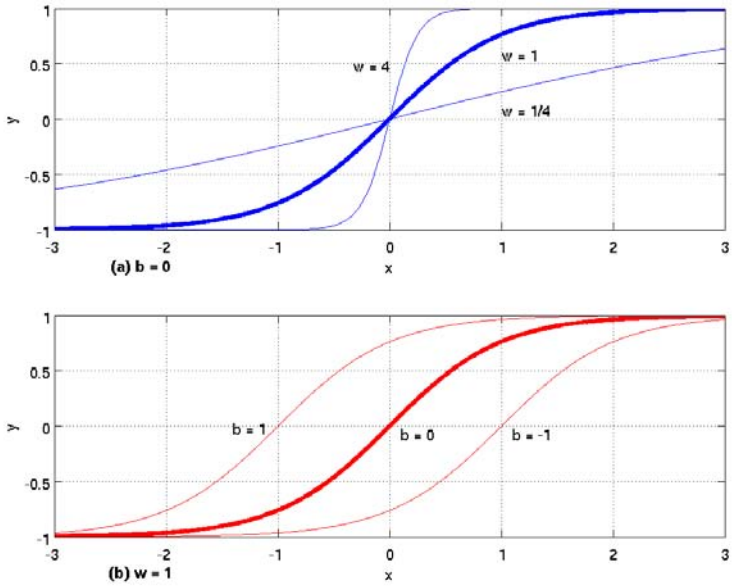


Figure 5.5 The functional behavior of a one-input perceptron is shown for a variety of parameter values. (a) b = 0, and (b) w = 1.

5.3.1.2 Example: A Single Perceptron with Two Inputs

We now expand the previous example to include a second input. The inputs x_1 and x_2 are weighted by w_1 and w_2, respectively, and a bias b is added before the transfer function is applied, as follows:

$$y = \tanh(w_1 x_1 + w_2 x_2 + b) \tag{5.20}$$

For Figure 5.6, $w_1 = 1$, $w_2 = 2$, and $b = -2$. Essentially, this surface consists of two parallel half-planes joined by a smooth transition region. w_1 and w_2 determine the direction in which the transition region extends and b determines how far away from (0,0) the transition region is.

Alternatively, the output for a perceptron with an arbitrary number of inputs can be written in terms of a dot product of a vector of inputs X and one of weights W:

$$y = \tanh(W \cdot X + b) \tag{5.21}$$

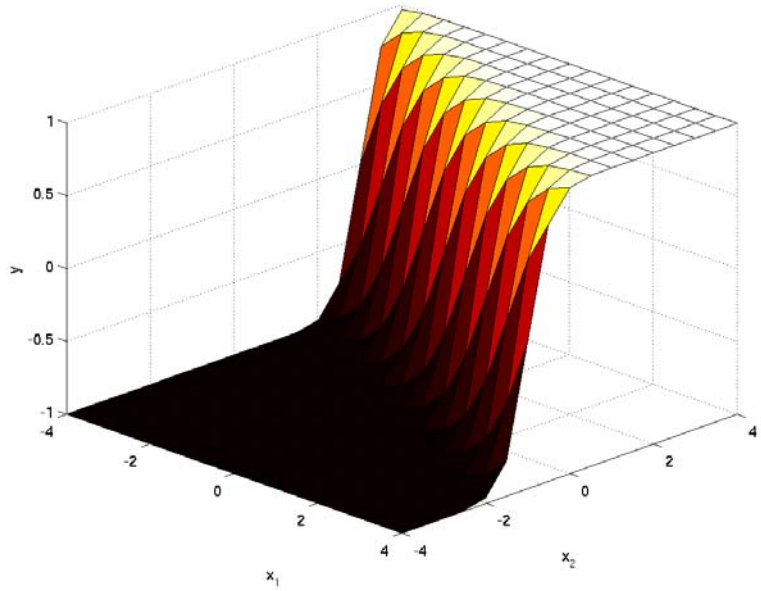


Figure 5.6 The output of a two-input perceptron showing two parallel half-planes joined by a smooth transition region. $w_1 = 1$, $w_2 = 2$, and $b = -2$.

The shape of this surface essentially is like two n-planes joined by a linearly shaped transition region, where n is the number of inputs.

5.3.1.3 Example: Signum Function

The signum function is a single-variable function whose value is -1 for negative arguments, 1 for positive arguments, and 0 for a zero-valued argument. A 1-1-1 network approximates the signum function very well, as shown in Figure 5.7. This is not surprising since the function $\tanh(wx)$ increasingly resembles the signum function as w goes to infinity. In fact, a single hyperbolic tangent node by itself can approximate this function. However, for general step functions (i.e., ones with arbitrary transition point and arbitrary values on both sides of the transition point), a second node with a linear transfer function is necessary to multiply the signum function by a scalar and to add a bias.

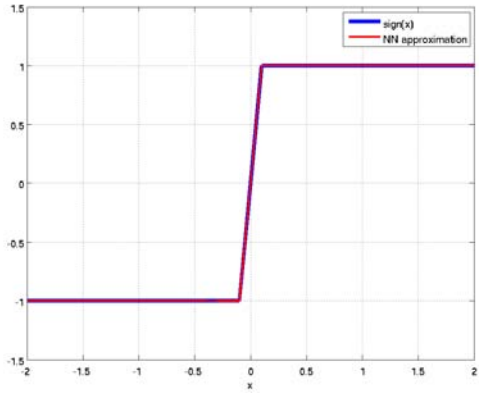


Figure 5.7 The signum function is plotted with a thick, dark line and the corresponding 1-1-1 neural network approximation is plotted with a thin, light line. The agreement is very good, as the two lines lie on top of one another.

5.3.1.4 Example: Absolute Value Function

Figure 5.8 illustrates the absolute value function. First, we consider a 1-1-1 network for which the output of the node in the first layer is $\tanh(wx + b)$ where w is the weight and b is the bias. This function increases monotonically for $w > 0$ (Figure 5.5) and decreases monotonically for $w < 0$. It was earlier noted that the linear node in the second layer could only multiply the output of the first layer by a scalar and add a bias. The second layer is not able to use the output of the first layer to produce any function that increases for some domain of arguments but decreases for other domains. Therefore, it is impossible for a 1-1-1 network to approximate the absolute value function. At best, the 1-1-1 network is able to approximate one side of the absolute value function. This is clearly seen in Figure 5.8 where the 1-1-1 network approximates well for values of $x < -1$ and then saturates for $x > -1$.

Non-monotonic functions like the absolute value function must be produced using two intermediate functions – one that monotonically increases and another that monotonically decreases. This can be achieved by adding a second node to the first layer so that one produces an increasing function while the other produces a decreasing function. A 1-2-1 network approximates $|x|$ fairly accurately (Figure 5.8); one node approximates the left side of the function and the other approximates the right side, as illustrated in Figure 5.9. Even with two hidden nodes, the region between $x = -0.2$ and

$x = 0.2$ presents a small but significant problem. One method to improve the performance in such problematic areas involves signal post-processing which will be discussed in Chapter 7. A 1-2-1 network can actually approximate the square of this function to very good accuracy, and with a square-root post-processing step can approximate the absolute value function to within an error smaller than what is achievable by training it to directly estimate the absolute value function.

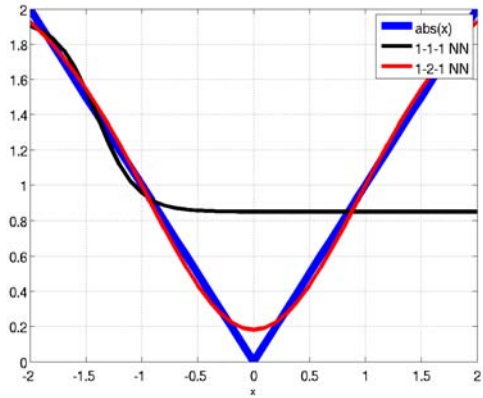


Figure 5.8 The absolute value function (thick, dark line), a 1-1-1 neural network approximation (thin, dark line), and a 1-2-1 neural network approximation (thin, light line).

5.3.1.5 Example: Cubic Function

There are two changes in direction in the function $x^3 - 2x$ for $|x| \leq 2$, and one might expect that three hidden nodes will be needed to represent this function. As shown in Figure 5.10, a 1-1-1 network is unable to learn the direction changes in this cubic function since it generally yields a positively sloped function that does not approximate the function well anywhere except around three points. A 1-2-1 network learns the function reasonably well for $|x| > 1.4$, but does not learn the area where there are changes in direction. A third hidden node provides the network with the mathematical flexibility it needs to learn the changes in direction. The weighted outputs of the hidden nodes are shown in Figure 5.11. One node learns the left side, one learns the right, and one learns the downward middle portion.

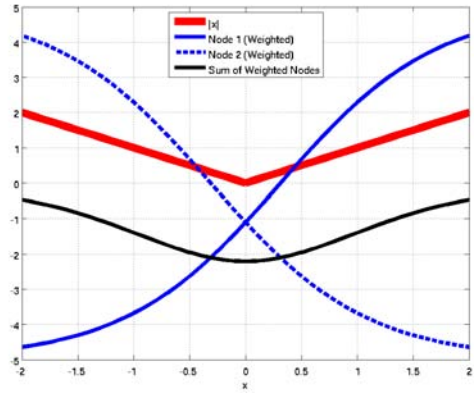


Figure 5.9 The outputs of the hidden nodes of a 1-2-1 neural network trained to approximate the absolute value function.

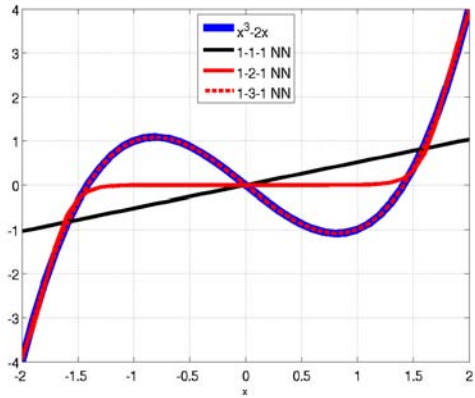


Figure 5.10 The cubic function $x^3 - 2x$ (thick, dark line), and approximations by 1-1-1 (thin, dark line), 1-2-1 (thin, light line), and 1-3-1 (dashed line, overlaid on the thick, dark line) neural networks.

5.3.1.6 Example: Sine Wave

In this example, we train a neural network to approximate four periods of a sine wave. Based on the preceding examples, a 1-9-1 neural network with nine hidden nodes should be capable of approximating the eight changes of

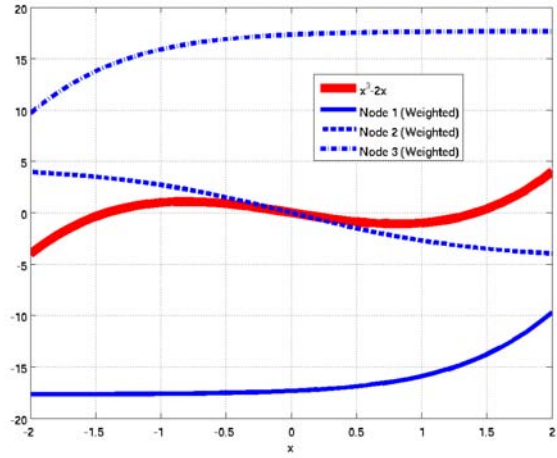


Figure 5.11 The outputs of the hidden nodes of the 1-3-1 neural network trained to approximate $x^3 - 2x$.

direction in this function, however, it is interesting to find that only five hidden nodes suffice. Figure 5.12 presents the results of training neural networks with from one to five hidden nodes. As the number of hidden nodes increased, the neural networks were able to approximate larger portions of the sine wave. Figure 5.13 shows the weighted outputs from each of the hidden nodes that feed the output node and reveals how the neural network learns the sine wave. One node handles all of the positively sloped regions, and each of the other nodes handles a negatively sloped region. The curves shown in Figure 5.13 demonstrate that the neural network nodes are each "active," but with differing relative contributions, over the entire input space. Understanding this simple behavior is a key step toward understanding how neural networks can be constructed to estimate more complicated, perhaps multidimensional, functions. The functions encountered in practice are seldom as simple to visualize as this sine wave, but the localized behavior of more complicated functions can often be examined similarly.

5.3.1.7 General Observations on Single-Variable Functions

The preceding examples show that the number of nodes needed to approximate a function depends to a large degree on the number of changes in direction. For the examples involving the absolute value functions and

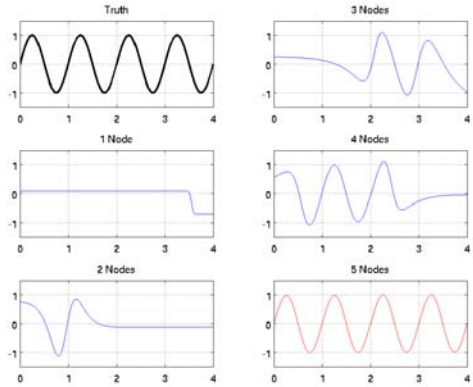


Figure 5.12 The upper left panel shows four periods of a sine wave to be approximated by a neural network. The middle left panel shows the representation obtained using one node, and the remaining panels show the effect of adding nodes one at a time. The lower right panel shows that the original sine wave can be estimated to very good accuracy with five nodes.

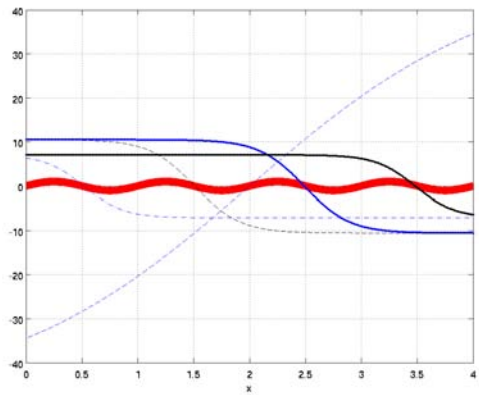


Figure 5.13 The relative contributions of the five hidden nodes to the four-period sine wave are shown. The thick, dark curve is the sine wave to be represented.

polynomial functions, the number of nodes required was equal to one plus the number of changes in direction. For polynomial functions, the number

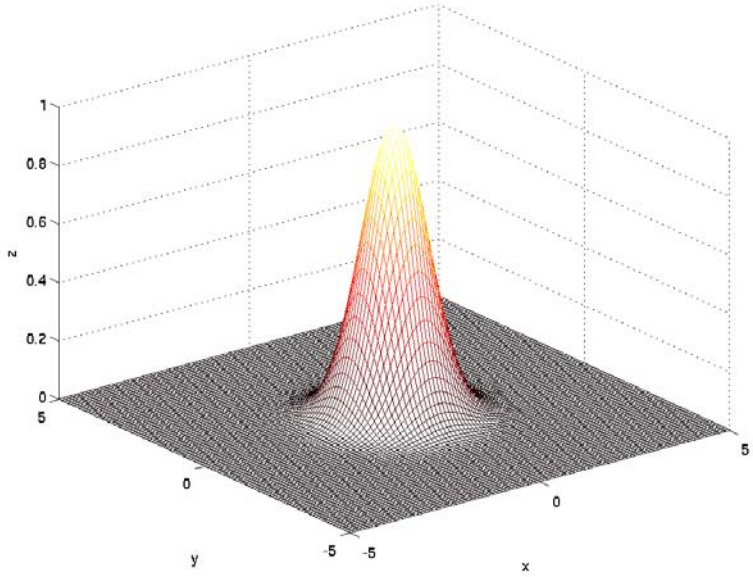


Figure 5.14 A circularly symmetric function.

of direction changes equals, at most, the degree of the polynomial. The sine wave example showed that the number of nodes could lie between the number of unique slopes in the function and one plus the total number of direction changes.

5.3.2 Two-Input Networks

We conclude the chapter with an example involving neural network estimation of the bivariate Gaussian probability density shown in Figure 5.14. This two-dimensional function is separable, that is, it is a product of a Gaussian pdf in one dimension multiplied by a Gaussian pdf in a second dimension. A 1-D normal pdf can be approximated with a 1-2-1 neural network. Therefore, a layer of four nodes is capable of approximating two 1-D Gaussian pdfs, one for each dimension. A second layer of three nodes is capable of multiplying two numbers, so a 2-4-3-1 network should be capable of learning a 2-D Gaussian function. Indeed, it has been determined that a 2-3-2-1 network is the simplest network capable of approximating a two-dimensional Gaussian.

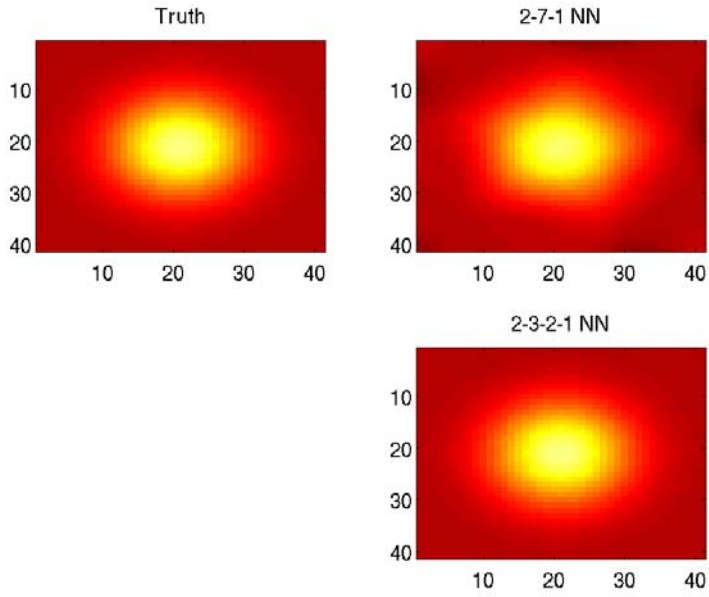


Figure 5.15 Approximations of the Gaussian pdf by 2-layer and 3-layer networks.

A two-layer network needs 17 hidden nodes to achieve an RMS error below approximately 0.1. This results in a total of 69 weights and biases to be optimized. In contrast, comparable performance is achieved using a three-layer network with three nodes in the first layer and two in the second, with a total of 20 weights and biases. Another problem with two-layer networks is that they cannot create circular contours in the functions they estimate, but approximate them instead as polygonal contours. Figure 5.15 shows an attempt by a 2-7-1 network to learn the Gaussian function, and the resulting image exhibits a pentagonal contour.

5.4 Summary

Machine learning algorithms have been widely applied to classification and regression problems in a variety of fields. Support vector machines and radial basis function networks offer solutions with guaranteed optimality under certain conditions, while multilayer perceptron networks can approximate a wide variety of complicated functions over large domains with relatively few nodes. We learned that it is possible to use a priori information about

the functional relationship between inputs and targets to predict the network topology needed to adequately learn these dependencies, and the topologies for simpler functions led to predictions of topologies for more complicated functions.

5.5 Exercises

1. Verify the minimal topologies for the following functions of a single variable x:
   (a) $f(x) = -1$ for $-2 \leq x \leq 0$, and 1 for $0 \leq x \leq 2$ (1–1–1)
   (b) $f(x) = |x|$ for $|x| \leq 2$ (1–2–1)
   (c) $f(x) = e^x$ for $|x| \leq 2$ (1–1–1)
   (d) $f(x) = x^2$ for $|x| \leq 2$ (1–2–1)
   (e) $f(x) = x^3 - 2x$ for $|x| \leq 2$ (1–3–1)
   (f) $f(x) = \sin(2\pi x)$ for $0 \leq x \leq 4$ (1–5–1)
2. For each of the following functions predict the minimal topology, verify by experiment, and determine the range of inputs for which each node contributes to the shape of the function:
   (a) $f(x) = (x-4)(x-3)(x+1)(x+2)$ for $-3 \leq x \leq 5$
   (b) $f(x) = x^5$ for $-1.5 \leq x \leq 1.5$ (hint: the minimal topology has fewer than five nodes in the first layer)
   (c) $f(x) = x^4$ for $-1.5 \leq x \leq 1.5$ (hint: the minimal topology has fewer than four nodes in the first layer)
   (d) $f(x) = \sin(2\pi x)$ for $0 \leq x \leq 7$ (seven periods of a sine wave)
   (e) $f(x) = \sin(2\pi x)$ for $0 \leq x \leq 10$ (ten periods of a sine wave)
3. Determine the minimal topology needed to multiply three non-negative numbers less than 5.
4. Use the result of Exercise 3 to determine a reasonable topology for computing the following functions:
   (a) The probability distribution function of a normal three-variable Gaussian
   (b) A three-dimensional step function
5. Predict and verify the minimal topology needed to learn $f(x, y) = (y - x)^2$ with a two-input net. Postulate a simple preprocessing method that would simplify the net and verify that it does simplify the net (i.e., that the preprocessing reduces the number of weights and biases to be optimized).

References

[1] K. M. Hornik, M. Stinchcombe, and H. White. "Multilayer feedforward networks are universal approximators." *Neural Networks*, 4(5):359–366, 1989.

[2] V. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.

[3] D. J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, Cambridge, U. K., 2003.

[4] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, New York, 2007.

[5] S. Haykin. *Neural Networks and Learning Machines*. Prentice Hall, Upper Saddle River, New Jersey, third edition, 2008.

[6] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Macmillan College Publishing Company, New York, 1994.

[7] R. P. Lippmann. "An introduction to computing with neural nets." *IEEE ASSP Magazine*, 4:4–22, 1987.

[8] T. M. Cover. "Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition." *IEEE Transactions on Electronic Computers*, 14:326–334, 1965.

[9] T. Hofmann, B. Schölkopf, and A. J. Smola. "Kernel methods in machine learning." *The Annals of Statistics*, 36(3):1171–1220, March 2008.

[10] A. J. Smola and B. Schölkopf. "A tutorial on support vector regression." *Statistics and Computing*, 2001.

[11] H. Hidalgo, S. S. León, and E. Gómez-Treviño. "Application of the kernel method to the inverse geosounding problem." *Neural Networks*, 16(3-4):349–353, 2003.

[12] M. A. Mohandes, T. O. Halawani, S. Rehman, and A. A. Hussain. "Support vector machines for wind speed prediction." *Renewable Energy*, 29(6):939–947, 2004.

[13] H. Zhan, P. Shi, and C. Chen. "Retrieval of oceanic chlorophyll concentration using support vector machines." *IEEE Trans. on Geosci. and Remote Sens.*, 41:2947–2951, December 2003.

[14] S. C. Chen, C. F. Cowan, and P. M. Grant. "Orthogonal least squares learning algorithm for radial basis function networks." *IEEE Trans. on Neural Networks*, 2(2):302–309, March 1991.

[15] J. Moody and C. J. Darken. "Fast learning in networks of locally-tuned processing units." *Neural Networks*, 6(4):525–535, 1989.

6

A Practical Guide to Neural Network Training

We now focus on the training process for a feedforward multilayer perceptron neural network, although the methodologies discussed are applicable to other network types. We provide practical guidance on many of the principal challenges in network training, especially those that we feel are commonly misused or insufficiently documented in the contemporary remote sensing literature. These include data set preparation, model selection, network initialization, parameter optimization, regularization, and performance assessment. Brief theoretical explanations are presented, however, the interested reader is referred to several excellent references [1–4] for more rigorous expositions.

6.1 Data Set Assembly and Organization

We begin with the most fundamental element of any statistical inference method, the assembly and organization of the data sets from which the statistical relationships are derived and evaluated. The most sophisticated and well-constructed inference methods cannot overcome flaws in the data sets on which they operate. The most useful data sets are accurate, comprehensive, and extensive, and the absence of any of these three key attributes severely undermines the overall effectiveness and applicability of the resulting neural network. Inaccurate and incomplete data sets can lead to poor network generalization, where generalization refers to the ability of the network to effectively estimate targets corresponding to inputs that have not been included in the data used to train the network.

6.1.1 Data Set Integrity

The common aphorism "garbage in, garbage out" is most apposite in the context of statistical inference. Any noise or other inaccuracies inherent in the data directly affects the quality of the inference, and, in particular, the power and flexibility of neural networks can lead to a tendency to "fit the noise" in the data. Any a priori knowledge of the statistical properties of the noise, including any dependencies to the underlying information content in the data, can be used to regularize the network and thereby improve the immunity to noise. However, the most effective way to minimize the effect of noise is to minimize the noise itself, to the extent possible. Given an abundance of data, one criterion that should be carefully considered is the accuracy of the available data, and only the data points with highest accuracy should be selected.

6.1.2 The Importance of an Extensive and Comprehensive Data Set

Network generalization can be improved by ensuring that the training data are *extensive* (that is, the entire dynamic range of the multidimensional input and target space is exercised) and *comprehensive* (that is, all the relevant feature sets are included). Consider, for example, the retrieval of the atmospheric temperature profile from space-based measurements of upwelling radiance. An extensive training set would include all geographic regions of the Earth, all seasons, night and day cases, and so forth. A comprehensive training set would include many examples of interesting phenomenology at all sensor viewing configurations, for example, temperature inversions, various types and amounts of clouds, dust storms, and volcanic eruptions.

6.1.3 Data Set Partitioning

The data set can be used for various stages of the neural network training process. For example, the weights and biases can be chosen by minimizing a cost function evaluated over the data. The performance for each training iteration (commonly termed an "epoch") can be used to terminate the optimization process if successive error reductions are insignificant (or for other reasons to be discussed later). The neural network model can be selected by examining the performance of several model alternatives using the same data set. The overall performance (for example, sum-squared error) of the trained neural network can be evaluated.

It would be most convenient to use the same data set to perform all of these tasks. This approach, however, is fraught with peril, as there is a

potential for the network to tune to the data and generalize poorly to a different data set. This idea is best formalized by considering separately the bias and the variance of the estimation error. The bias refers to the degree to which the targets are accurately fitted and accounts only for the given data, but not for the level of generalization to other data. The variance refers to the deviation of the neural network learning performance from one data sample to another sample that could be described by the same target function model. This is the statistical variance that accounts for the degree to which the neural network fits the examples without regard to the specificities of the given data. An optimal estimator minimizes the bias and variance jointly, and it therefore follows that at least two separate data sets should be used to train the neural network: one to evaluate the bias (that is, the performance on only the training data) and one to evaluate the variance (that is, the performance on a separate data set). It is further recommended that a third data set be used to evaluate the ability of the optimized network to effectively generalize to unseen inputs. We therefore introduce the following terminology for these three data sets:

Training set:	Data used to optimize the network weights and biases.
Validation set:	Data used to (1) determine when to terminate the training process, (2) evaluate model complexity, and (3) evaluate multiple training runs.
Testing set:	Data used to assess generalization of the network.

The neural network performance can be improved if both the error bias and the variance are reduced. However, there is a natural trade-off between the bias and variance, the so called *bias/variance dilemma* [5]. A neural network that closely fits the training data has a low bias but not necessarily a low variance; this must be checked by assessing the performance of the network using the testing data. The best strategy for reducing both the error bias and variance is to maximize the amount of training data and to use the simplest network architecture required to achieve the desired performance. Note that the upper bound on the complexity of the network that can be reliably used is set by the amount of data available. A common rule of thumb is to ensure that the number of vectors in the training set exceeds the total number of weights in biases in the network by a factor of about 5–10, although this factor can depend heavily on the problem at hand, and the ratio can be higher for highly nonlinear or non-Gaussian relationships. Given an abundance of data, a 60-20-20 split is often used to construct the three sets. Practical aspects of data set organization and the bias/variance dilemma will be discussed in more detail in Section 6.5.

6.2 Model Selection

The issues of network model selection are inextricably linked to issues of training data completeness. All things being equal, complicated models require more training and validation data to ensure proper generalization and prevent overfitting. There is often motivation to use many more nodes or hidden layers than could be necessary to be absolutely sure that a given neural network is sufficiently capable of fitting the data at hand. There are drawbacks to this approach. The amount of training data required increases markedly with network complexity, and the resulting time and computer memory required to train the network also grows. The error surfaces of complex networks can be very complicated, further posing a risk that the optimization can be trapped in a local minimum. Therefore, careful choices must be made to trade network complexity with network stability and training time.

6.2.1 Number of Inputs

At first glance, it may seem pointless to pick a subset of inputs to present as inputs to a neural network. Why not present all available inputs and let the neural network determine which are relevant? This approach is reasonable if there are no restrictions on the training set size or the time required for training. This is usually not the case, and the efficiency of the training process can be substantially increased if excessively noisy, unimportant, or redundant inputs are removed. In Chapter 7, we discuss preprocessing techniques that can be used to represent the input data in a statistically compact form. The neural network then wastes less resources extracting and fitting irrelevant features. This can be especially helpful for hyperspectral infrared data, where thousands of variables are available for use, but there is a high degree of correlation among them.

6.2.2 Number of Hidden Layers and Nodes

It has been proven that neural networks with a single hidden layer are universal approximators capable of representing any real-valued continuous function to arbitrary precision over a finite domain if enough hidden nodes are used [6]. However, networks with multiple hidden layers can sometimes perform better than single-hidden-layer networks, with fewer total nodes (see Section 5.3.2, for example). A few simple trial-and-error experiments with various topologies can quickly identify reasonable choices for the number of nodes and layers to be used. Generally, a higher degree of nonlinearity requires more hidden nodes and layers. Simple linear

and quadratic regressions can provide valuable insight into the degree of nonlinearity (see Section 3.3.2, for example).

6.2.3 Adaptive Model Building Techniques

There are a number of innovative techniques that can be used during network training to adaptively determine the optimal network topology. One such example is "network pruning," where unimportant weights in the network are removed. The general approach is to use a Taylor series expansion of the network sum-squared error as a function of the weights [7, 8]:

$$\delta E = \underbrace{\nabla E(\mathbf{w})^T \mathbf{dw}}_{\approx 0} + \frac{1}{2}\mathbf{dw}^T \nabla^2 E(\mathbf{w})\mathbf{dw} + \underbrace{O(\|\mathbf{dw}\|^3) + \cdots}_{\approx 0} \tag{6.1}$$

where $\nabla E(\mathbf{w})$ and $\nabla^2 E(\mathbf{w})$ are the gradient vector and the Hessian matrix of the network error, respectively. At a local minimum, the first term vanishes, and we ignore the higher-order terms. The method proceeds by finding the single weight that can be set to zero with the minimum resulting increase in error given by (6.1). The solution requires the inverse Hessian, which can be calculated as the network is trained (see Section 6.4, for example).

An alternative strategy to network pruning is to begin training with a relatively small number of weights and add weights as needed as the training progresses. One advantage of this approach is that the total training time can be reduced relative to the pruning approach, especially if the initial choice of the number of nodes to prune is unnecessarily large. An example of model building by node accretion is given by Refenes [9].

6.3 Network Initialization

The initial state of the neural network can impact both the learning rate and the error performance of the trained network. The error surfaces for most practical problems are sufficiently complicated that suboptimal terminations in local minima are not only possible but probable. One approach is to randomly initialize the network multiple times and pick the best-performing network. A simple but often effective initialization method is to set each weight to a uniform random number in the range of $[-1, 1]$. An improvement was proposed by Nguyen and Widrow [10] for a network with a single hidden layer with n inputs and p hidden nodes. The weights $w_{ij}^{(0)}$ are first assigned a uniform random number in the range of $[-1, 1]$ and then normalized as

follows:

$$w_{ij} = \frac{0.7p^{1/n} \cdot w_{ij}^{(0)}}{\|w_j^{(0)}\|} \tag{6.2}$$

This normalization of initial weight values tends to spread the weighted inputs over a larger region of the sigmoidal activation regions and speeds up training, sometimes by a factor of 10 or more. More recent techniques have been proposed that are based on sensitivity analysis, which uses a linear training algorithm for the hidden and output layers to initialize the weights [11].

6.4 Network Training

We now consider a multilayer feedforward neural network consisting of an input layer, an arbitrary number of hidden layers (usually one or two), and an output layer. The hidden layers typically contain sigmoidal activation functions of the form $z_j = \tanh(a_j)$, where $a_j = \sum_{i=1}^{d} w_{ij}x_i + b_j$. The output layer is typically linear. The weights (w_{ij}) and biases (b_j) for the j^{th} neuron are chosen to minimize a cost function over a set of P training patterns. A common choice for the cost function is the sum-squared error, defined as

$$E(\mathbf{w}) = \frac{1}{2}\sum_p \sum_k \left(t_k^{(p)} - y_k^{(p)}\right)^2 \tag{6.3}$$

where $y_k^{(p)}$ and $t_k^{(p)}$ denote the network outputs and target responses, respectively, of each output node k given a pattern p, and $\mathbf{w}$ is a vector containing all the weights and biases of the network. We denote the sum-squared error for a single pattern p as $E^{(p)}$. The "training" or "learning" process involves iteratively finding the weights and biases that minimize the cost function. This is usually done in two stages. First, the derivatives of the cost function with respect to the weights and biases (that is, the error gradient) are calculated with a backpropagation technique. These derivatives are then used to update the weights and biases through some numerical optimization procedure, such as simple gradient descent. The term "backpropagation" is sometimes used to refer to the combination of these two steps.

6.4.1 Calculation of the Error Gradient Using Backpropagation

Minimization of the error (6.3) with respect to the network weights requires the calculation of the derivative of the sum-squared error with respect to the network weights, that is, the gradient vector, $\nabla E(\mathbf{w})$. Rumelhart et al. introduced the backpropagation technique [12] for calculating the

gradient vector for a general network having arbitrary feedforward topology, arbitrary differentiable activation functions, and an arbitrary differentiable error function. Here we briefly review the methodology.

The sum-squared error over a set of P training patterns can be computed simply by summing the error of each pattern:

$$E = \sum_p E^{(p)} \tag{6.4}$$

For a given input pattern, we can consider the evaluation of the derivative of $E^{(p)}$ with respect to some weight w_{ij}. The chain rule can be applied to give

$$\frac{\partial E^{(p)}}{\partial w_{ij}} = \frac{\partial E^{(p)}}{\partial a_j} \frac{\partial a_j}{\partial w_{ij}} \tag{6.5}$$

where we have made use of the fact that $E^{(p)}$ depends on the weight w_{ij} only through the summed input a_j to unit j. The second term in (6.5) can be expressed simply as:

$$\frac{\partial a_j}{\partial w_{ij}} = x_i \tag{6.6}$$

where we use x_i to denote the node inputs for a given layer. We introduce the following notation for the first term in (6.5):

$$\delta_j \equiv \frac{\partial E^{(p)}}{\partial a_j} \tag{6.7}$$

and rewrite (6.5) as

$$\frac{\partial E^{(p)}}{\partial w_{ij}} = \delta_j x_i \tag{6.8}$$

The simplicity of (6.8) is the key to the utility of backpropagation. The gradient can be constructed by multiplying the value of δ for the node at the output end of the weight by the value of x for the node at the input end of the weight. Once the δ_js are calculated for each hidden and output node in the network, the derivatives can be evaluated using (6.8).

For linear output nodes, δ_k can be expressed simply as:

$$\delta_k = t_k - y_k \tag{6.9}$$

For the hidden nodes, where $z_j = \tanh(a_j)$, the chain rule may be used to write

$$\delta_j = (1 - z_j^2) \sum_{k=1}^{c} w_{jk} \delta_k \tag{6.10}$$

where the sum runs over all output nodes and we have used the fact that the derivative of the $\tanh$ function can be expressed as $1 - \tanh^2$. The derivatives with respect to the first-layer and second-layer weights are then:

$$\frac{\partial E^{(p)}}{\partial w_{ij}} = \delta_j x_i \tag{6.11}$$

$$\frac{\partial E^{(p)}}{\partial w_{jk}} = \delta_k z_j \tag{6.12}$$

6.4.2 First-Order Optimization: Gradient Descent

Now that we have an efficient method for calculating the error gradients, we can numerically optimize the weights and biases using an iterative procedure. We now consider the fixed-step gradient descent method, which simply adjusts the weights in the direction opposite the error gradient (that is, the direction of the greatest rate of decrease of the error):

$$\mathbf{dw} = -\eta \nabla E(\mathbf{w}) \tag{6.13}$$

where the parameter η is the *learning rate*. There are a multitude of variations and improved versions of the standard gradient descent method, including gradient descent with momentum, conjugate gradients, and scaled conjugate gradients [1].

6.4.3 Second-Order Optimization: Levenberg-Marquardt

The local approximation of the cost function (6.3) by a quadratic form is given by (6.1). We can directly obtain the location of the minimum by setting the derivative of (6.1) to zero and solving for the weight update vector $\mathbf{dw}$. The solution yields the "Newton step":

$$\mathbf{dw} = -[\nabla^2 E(\mathbf{w})]^{-1} \nabla E(\mathbf{w}) \tag{6.14}$$

Direct application of (6.14) is difficult in practice, because computation of the Hessian matrix (and its inverse) is nontrivial and usually needs to be repeated at each iteration of network training. For sum-squared error cost functions, it can be shown that

$$\nabla E(\mathbf{w}) = \mathbf{J}^T \mathbf{e} \tag{6.15}$$

$$\nabla^2 E(\mathbf{w}) = \mathbf{J}^T \mathbf{J} + \mathbf{S} \tag{6.16}$$

where $\mathbf{J}$ is the Jacobian matrix that contains first derivatives of the network errors with respect to the weights and biases, $\mathbf{e}$ is a vector of network

errors, and $\mathbf{S} = \sum_{p=1}^{P} \mathbf{e}_p \nabla^2 \mathbf{e}_p$. The Jacobian matrix can be computed using the backpropagation technique discussed in the previous section, and this is significantly more computationally efficient than direct calculation of the Hessian matrix [13]. However, an inversion of a square matrix with dimensions equal to the total number of weights and biases in the network is required. For the Gauss-Newton method, it is assumed that $\mathbf{S}$ is zero (a reasonable assumption only near the solution), and the update equation (6.14) becomes

$$\mathbf{dw} = -[\mathbf{J}^T\mathbf{J}]^{-1}\mathbf{Je} \tag{6.17}$$

The Levenberg-Marquardt modification [14] to the Gauss-Newton method is

$$\mathbf{dw} = -[\mathbf{J}^T\mathbf{J} + \mu\mathbf{I}]^{-1}\mathbf{Je} \tag{6.18}$$

As μ varies between zero and ∞, $\mathbf{dw}$ varies continuously between the Gauss-Newton step and steepest descent. The Levenberg-Marquardt method is thus an example of a model trust region approach in which the model (in this case the linearized approximation of the error function) is trusted only within some region around the current search point [1]. The size of this region is governed by the value μ.

Levenberg-Marquardt optimization is often substantially faster than gradient-based methods. However, a large amount of computer memory is required for networks with many weights and biases. One approach is to divide the Jacobian matrix into submatrices and compute the Hessian in steps as follows:

$$\mathbf{H} = \mathbf{J}^T\mathbf{J} = \begin{bmatrix}\mathbf{J}_1^T\mathbf{J}_2\end{bmatrix}\begin{bmatrix}\mathbf{J}_1\\ \mathbf{J}_2\end{bmatrix} = \mathbf{J}_1^T\mathbf{J}_1 + \mathbf{J}_2^T\mathbf{J}_2 \tag{6.19}$$

This approach saves memory, but slows down the optimization procedure.

6.5 Underfitting and Overfitting

We have described all the tools necessary to train a neural network but have not yet offered any insight into how the performance could be evaluated. Perhaps the two most fundamental neural network performance attributes are the quality of the fit to the training data and the quality of the fit to unseen data (that is, generalization capability). We have discussed earlier that the complexity of the network (the number of layers and hidden nodes) affects both of these attributes. A network with too few nodes will be unable to accurately represent the underlying statistical relationship between the inputs and the targets (as several examples presented in Chapter 5 demonstrated), a phenomenon known as "underfitting." Too many nodes can

also be problematic, because there is a danger for the neural network to exploit patterns in the training data that are irrelevant to the true statistical dependencies relating the inputs and outputs. For example, if the training data are noisy, it is possible that the neural network will "fit the noise."

We now provide guidance on the detection and mitigation of under- and overfitting. One of the best ways to monitor these phenomena is to monitor the error on the training and validation data set as the network trains. The error trend for the training data is generally (but not strictly) monotonically decreasing. The error trend for the validation data will start to increase, however, when the network starts to overfit. We illustrate these effects with a simple example.

Consider a damped sinusoid of the form

$$f(x) = e^{-ax} \sin(2\pi x/T) \tag{6.20}$$

This function is shown in Figure 6.1 for $a = 0.2$ and $T = 1.5$ for x within $[0, 10]$. This function is sampled (without noise) in increments of 0.01 for a total of 1,000 samples. We exclude 150 points from 4.5 to 6.0 from the training set and set them aside for the validation set. A neural network with a single hidden layer of 30 nodes is then trained for 5,000 epochs using Levenberg-Marquardt optimization.

The neural network estimates (denoted in Figure 6.1 by a dashed line) very closely approximate the true function over the training set. However, we see that the validation error (denoted in the figure by a dash-dot line) at the end of 5,000 epochs is quite poor. In fact, the validation error after 5,000 epochs is significantly worse than that at 986 epochs (the minimum value). A plot of the training and validation error as a function of epoch is shown in Figure 6.2.

This example demonstrates the merit of an "early-stopping" technique that can be used to minimize network overfitting. By monitoring the trends in the validation error as the network trains, the onset of overfitting can be detected and the training can be terminated. If the validation error decreases in a monotonic fashion, it could be an indication of underfitting, and an experiment should be conducted with additional nodes added to the network.

There is another interesting feature of overfitting, where the weights of the network start to become large as the network attempts to fit irrelevant features in the data. Figure 6.3 shows histograms of the weight values in the hidden layer for the damped sinusoid example for the "minimum validation error" case at 986 training epochs and at the end of 5,000 training epochs. Note that some weights grow very large relative to the weights obtained at the minimum of the validation error curve as the network starts to overfit beyond 986 epochs. This behavior motivates the use of a regularization technique to combat overfitting, where the sum-squared value of all the weights is jointly

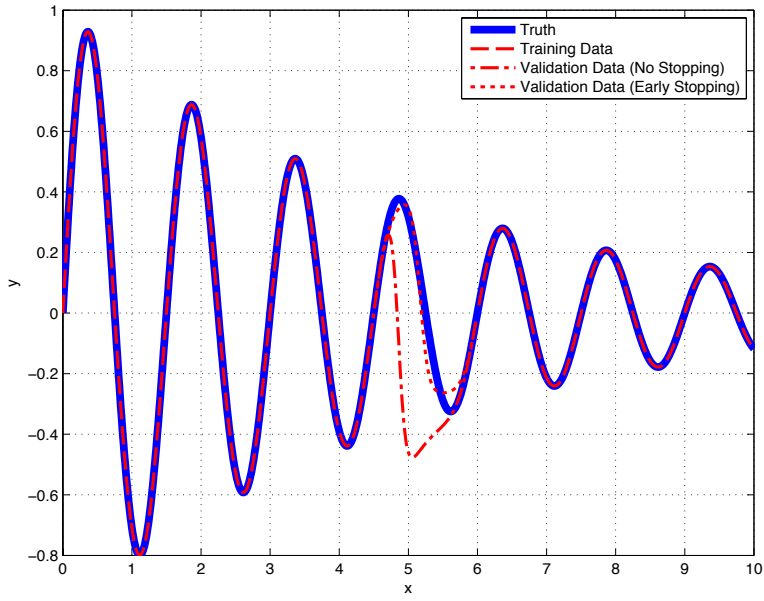


Figure 6.1 Approximation of a damped sinusoid using a neural network with a single hidden layer of 30 sigmoidal nodes. The region from 4.5 to 6 was excluded from the training set. The thick line indicates the true function to be approximated. The dashed line indicates the quality of fit of the training data. The dash-dot line indicates the fit of the validation data obtained by stopping network training after 5,000 epochs, and the dotted line indicates the fit of the validation obtained by stopping network training when the minimum validation was obtained (in epoch 986; see Figure 6.2).

minimized with the error cost function. This technique is discussed in the following section.

6.6 Regularization Techniques

We have seen that early stopping of network training through the use of a validation set effectively mitigates overfitting. It also possible to employ techniques as part of the numerical optimization that help reduce overfitting by desensitizing the network to interfering features, such as observation noise. This can be done by modifying the training data or by modifying the learning procedure, or both. We illustrate two common techniques in the context of sensor noise, but the concepts generalize to other forms of interference,

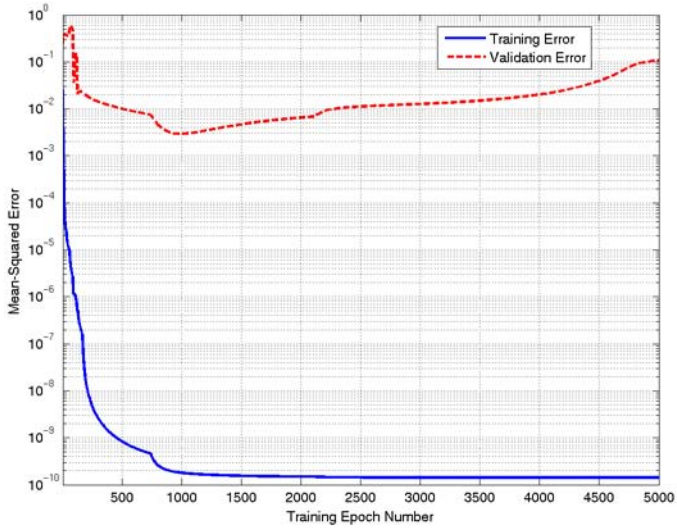


Figure 6.2 Mean-squared error as a function of training epoch for the training and validation sets. A single hidden layer of 30 sigmoidal nodes was used. The training error decreases monotonically, but the validation error reaches a minimum at epoch 986.

including geophysical crosstalk from clouds and surface phenomena, for example.

6.6.1 Treatment of Noisy Data

Noise, and possibly other interfering signals, pervade most measurements. The noise further complicates the already challenging task of geophysical parameter retrieval, because the pertinent relationships to be extracted can be subtle and therefore easily obscured by noise. Even more pernicious is the potential of neural networks to infer specious patterns and features from the noise as if they legitimately describe the statistical relationships between the inputs and outputs. Care must be taken to ensure that the neural network is not fitting to noise.

Neural networks are often used with simulated observations, where a noise model is used in conjunction with a random number generator to produce noise with the necessary statistics. In these cases, it is imperative that a new realization of the sensor noise is created for each training epoch. If this is not done, it is all but certain that even simple neural networks will begin

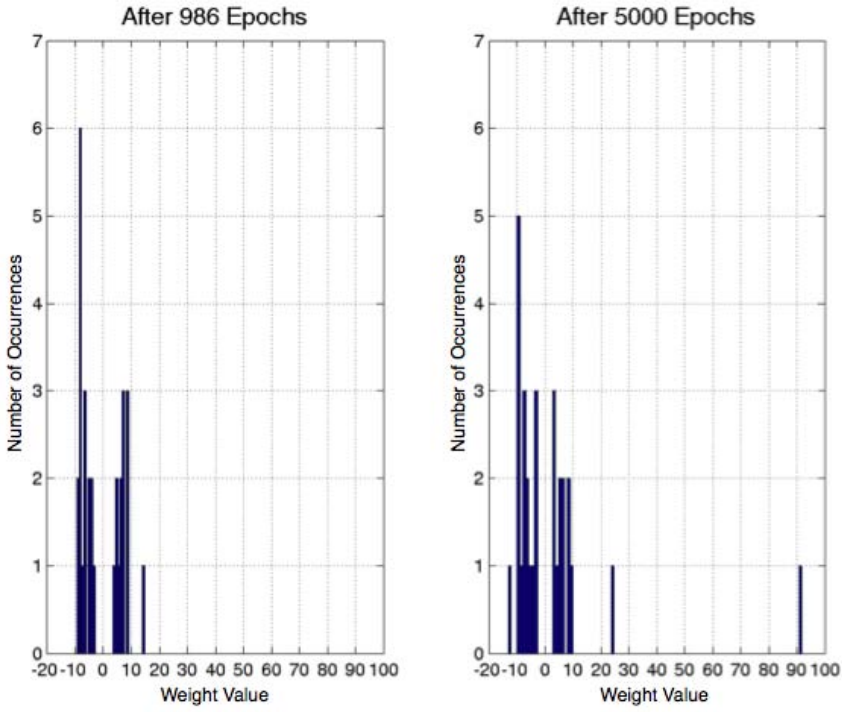


Figure 6.3 Histograms of the neural network weight values in the hidden layer for the damped sinusoid example.

to fit to this deterministic realization. We illustrate this point by returning to the simple damped sinusoid example and training a network with 20 hidden nodes. We reduced the number of nodes in the previous examples by one third to account for the noise. This reduction is a simple form of regularization. We generate a single realization of random noise with standard deviation of 0.01 and add this to the training data. The network is then trained for 1,000 epochs. The results are shown in Figure 6.4.

We see that the neural network begins to fit to the noise in the training data early in the learning process, with disastrous consequences at the end of training. Early stopping holds the validation error to approximately 0.01.

We repeat the experiment, this time using a new random realization of the sensor noise for each learning epoch. The results are shown in Figure 6.5. The validation error is improved throughout the learning process, with a minimum of approximately 0.001, one order of magnitude lower than the previous case with "deterministic" noise.

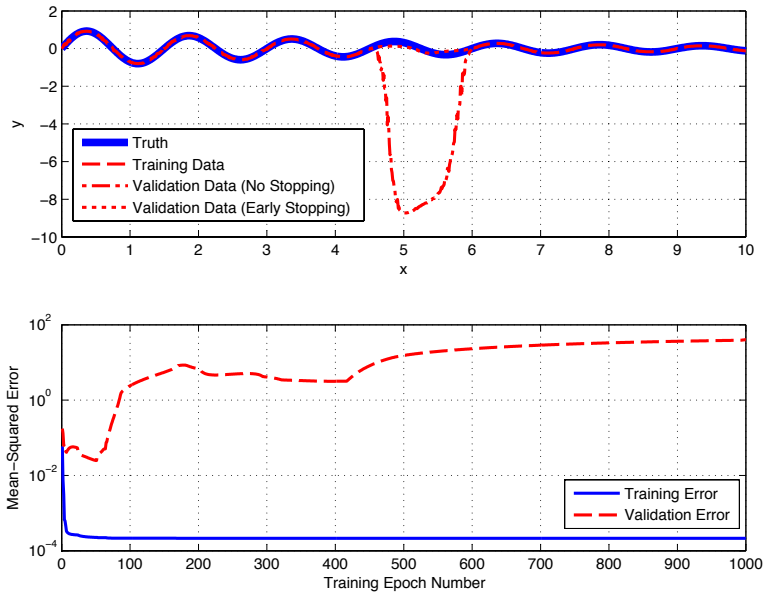


Figure 6.4 Approximation of a noisy damped sinusoid using a neural network with a single hidden layer of 20 sigmoidal nodes. The validation error is markedly worse than that shown in Figure 6.1.

6.6.2 Weight Decay

The a priori statistical description of the noise in the training data may not be available, thereby precluding the use of the noise regularization technique in the previous section. This is often the case when sensor observations are co-located with some form of ground truth. Another approach that can be used to desensitize the network to noise is by assigning a penalty for large weight values. This penalty can be included in the cost function as follows:

$$E(\mathbf{w}) = \frac{1}{2}\sum_{p}\sum_{k}\left(t_k^{(p)} - y_k^{(p)}\right)^2 + \mathbf{w}^T\mathbf{A}\mathbf{w} \tag{6.21}$$

where the matrix $\mathbf{A}$ can be used to assign more importance to particular weights. The cost function can be minimized using simple modifications to the numerical optimization routines used to update the weights. This approach is termed *weight decay* and is analogous to the ridge regression method presented in Section 3.3.2. Figure 6.6 shows the learning curves for the neural network trained with weight decay. The validation error at the end of training,

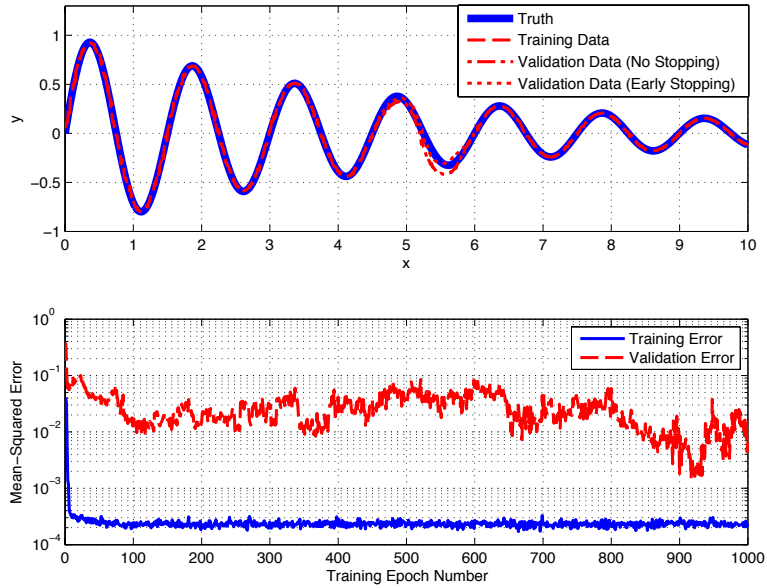


Figure 6.5 Approximation of a noisy damped sinusoid using a neural network with a single hidden layer of 20 sigmoidal nodes. A new random realization of the sensor noise is generated for each learning epoch.

while not as low as the minimum value, is substantially lower than that obtained without weight decay (see Figure 6.4).

6.7 Performance Evaluation

The previous examples have demonstrated isolated experiments analyzing a single training attribute, and we have therefore not used both a validation and testing set. In practice, many such experiments are typically conducted to ensure that a network performs adequately and is stable with respect to input noise. The validation data set is used to optimize the topology, pick the most suitable regularization technique, and evaluate multiple initializations to pick the best optimization. Once these factors are settled, the testing set should be used to assess the final performance of the network. The validation set alone is not adequate for this purpose, as it is possible that the characteristics of the network training that have been chosen through extensive evaluation with the validation set are "tuned" to the validation set and may not generalize well to other data sets. It is for this reason that a separate testing set is needed for an

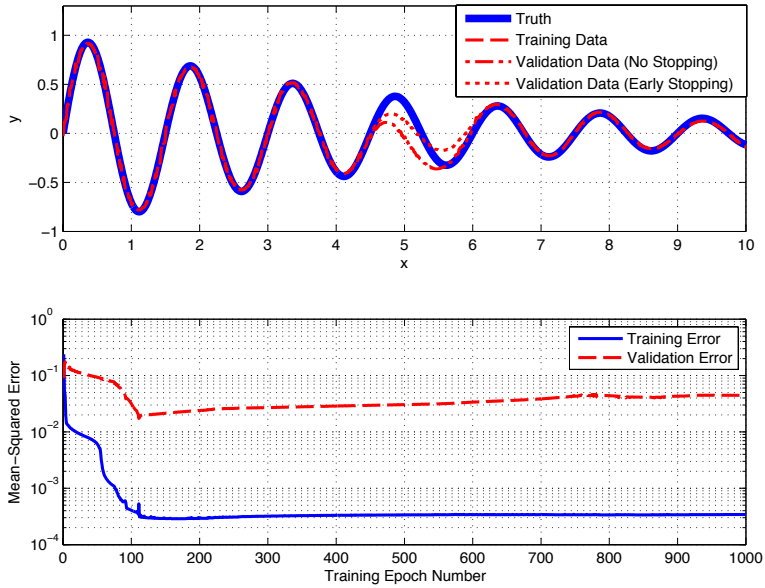


Figure 6.6 Approximation of a noisy damped sinusoid using a neural network with a single hidden layer of 20 sigmoidal nodes and weight decay.

objective evaluation of performance. In Chapters 9 and 10 we provide detailed examples of training optimization for real-world problems in atmospheric remote sensing and present a detailed performance and error analysis for these cases.

6.8 Summary

Neural network training involves a number of sequential steps, many of which should be iterated and optimized through a process of trial and error. The available data set should be partitioned into three sets for training, validation, and testing of the network. Selection of network topology is an often underemphasized step in the neural network training process. The optimal network topology is related to the number of degrees of freedom in the inputs and outputs and the complexity of their statistical relationship. The number of nodes may be limited by the amount of training data available, and thorough analysis should be performed to ensure that the network is sufficiently stable and does not overfit to the training data. Early stopping can be used to improve the ability of the network to generalize, as can

regularization techniques such as weight decay. Neural networks can be desensitized to noisy data by generating random realizations of the noise at each training epoch. Sophisticated second-order optimization techniques such as Levenberg-Marquardt can be used to expedite the training process, at the expense of increased memory usage.

References

[1] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, Oxford, U. K., 1995.

[2] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Macmillan College Publishing Company, New York, 1994.

[3] S. Haykin. *Neural Networks and Learning Machines*. Prentice Hall, Upper Saddle River, New Jersey, third edition, 2008.

[4] D. J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, Cambridge, U. K., 2003.

[5] S. Geman, E. Bienenstock, and R. Doursat. "Neural networks and the bias/variance dilemma." *Neural Computation*, 4(1):1–58, 1992.

[6] K. M. Hornik, M. Stinchcombe, and H. White. "Multilayer feedforward networks are universal approximators." *Neural Networks*, 4(5):359–366, 1989.

[7] Y. Le Cun, J. S. Denker, and S. A. Solla. "Optimal brain damage." In *Advances in Neural Information Processing Systems 2*, pages 598–605. Morgan Kaufmann, San Francisco, California, 1990.

[8] B. Hassibi and D. G. Stork. "Second order derivatives for network pruning: Optimal brain surgeon." In *Advances in Neural Information Processing Systems 5*, pages 164–171. Morgan Kaufmann, San Francisco, California, 1993.

[9] A. N. Refenes. "Constructive learning and its application to currency exchange rate forecasting." *Neural Network Applications in Investment and Finance Services*, 1991.

[10] D. Nguyen and B. Widrow. "Improving the learning speed of two-layer neural networks by choosing initial values of the adaptive weights." *IJCNN*, 3:21–26, 1990.

[11] E. Castillo, B. Guijarro-Berdiñas, O. Fontenla-Romero, and A. Alonso-Betanzos. "A very fast learning method for neural networks based on sensitivity analysis." *Journal of Machine Learning Research*, 7:1159–1182, 2006.

[12] D. E. Rumelhart, G. Hinton, and R. Williams. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Vol. 1: Foundations.* D. E. Rumelhart and J. L. McClelland (Eds.). MIT Press, Cambridge, Massachusetts, 1986.

[13] M. T. Hagan and M. B. Menhaj. "Training feedforward networks with the Marquardt algorithm." *IEEE Trans. Neural Networks*, 5:989–993, November 1994.

[14] P. E. Gill, W. Murray, and M. H. Wright. "The Levenberg-Marquardt method." In *Practical Optimization*. Academic Press, London, 1981.

7

Pre- and Post-Processing of Atmospheric Data

A neural network is capable of approximating a wide range of complicated, multidimensional functional relationships, but this power and universal applicability is not without caveats. We have seen that as the complexity of the neural network is increased by the addition of nodes or layers there is a concomitant increase in the amount of computation required for training (or an increase in the amount of time required for a given computational budget), the amount of training data required, and the potential for instabilities. A motivation therefore exists to keep the network as simple as possible (but no simpler, to quote Albert Einstein).

Pre- and post-processing of the neural network inputs and outputs, respectively, using linear or simple nonlinear operators can substantially reduce the required complexity of the network architecture by simplifying the mathematical relationships that a neural network has to learn. This reduction can be realized in a number of ways. For example, it may be possible to represent the inputs or outputs in a more statistically compact form. It may also be possible to remove or reduce the presence of interfering signals, such as sensor noise, clouds, or surface variability, that are irrelevant to the desired geophysical processes, but nonetheless may be "learned" by the network. We saw in Chapter 4 that principal components transforms can be used to both reduce dimensionality and reduce sensor noise. Furthermore, simple nonlinear transforms may be used to exploit a priori knowledge of the functional dependencies among the inputs or outputs. In this chapter, we explore pre- and post-processing issues in detail and present examples of common techniques.

7.1 Mathematical Overview

The following notation was introduced in Chapter 4 to describe the decomposition and synthesis of a random vector of N components:

$$\widehat{R}_r = \mathbf{g}_r(\mathbf{f}_r(R_N)) \tag{7.1}$$

where the decomposition operator $\mathbf{f}_r(R_N)$ is generally a nonlinear vector-valued function that returns a vector output with r elements. If $r < N$, we refer to the operators $\mathbf{f}$ and $\mathbf{g}$ as compression and decompression operators, respectively. We have used the vector variable R to denote sensor radiance, which is commonly a neural network input in the remote sensing context. The $\mathbf{f}$ and $\mathbf{g}$ operators can also be applied to the neural network outputs, and we therefore make this explicit as follows:

$$\widehat{R}_r = \mathbf{g}_r^{\text{pre}}(\mathbf{f}_r^{\text{pre}}(R_N)) \tag{7.2}$$

$$\widehat{S}_p = \mathbf{g}_p^{\text{post}}(\mathbf{f}_p^{\text{post}}(S_M)) \tag{7.3}$$

and denote the operations on the neural network inputs as "preprocessing" and the operations on the neural network outputs as "post-processing." Given a neural network, $\mathbf{n}(\cdot)$, we can derive a general expression for the estimate of the atmospheric state S (with M components) given a noisy radiance observation $\widetilde{R}$ as follows:

$$\widehat{S}(\widetilde{R}) = \mathbf{g}_p^{\text{post}}(\mathbf{n}(\mathbf{f}_r^{\text{pre}}(\widetilde{R}))) \tag{7.4}$$

We consider these operators separately in this book. However, there is no reason that they cannot be jointly optimized. The joint optimization of pre- and post-processing operations is a rich area of current research, and the interested reader is referred to [1] for an excellent introduction.

There are three desirable properties of pre- and post-processing operations on neural network inputs and outputs. First, if $r < N$ or $p < M$, we have achieved "data compression" by representing the data with fewer degrees of freedom. Second, if the reconstructed radiance distortion, $E[(\widehat{R}_r - R)^T(\widehat{R}_r - R)]$, is less than the radiance distortion due to noise, $E[(\widetilde{R} - R)^T(\widetilde{R} - R)]$, we have achieved "noise filtration" by reducing an interfering noise signal in the data. The third property cannot be described using a metric derived from the input or output data, but instead is evaluated with respect to the network performance. We refer to "data warping" as an operation (usually nonlinear) on the inputs or outputs that may or may not compress the data or filter noise, but does in some way improve network performance either by enabling faster training or better estimation performance. We now present practical examples involving data compression, noise filtration, and data warping.

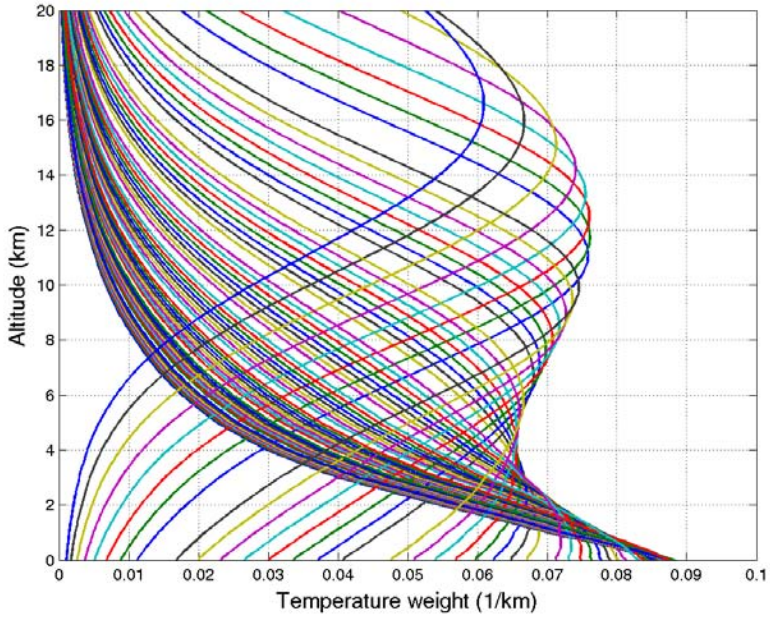


Figure 7.1 Temperature weighting functions of a hypothetical microwave sounding system with 64 channels near the 118.75-GHz oxygen line are shown.

7.2 Data Compression

Consider a simulated spaceborne microwave sounding system operating near the opaque 118.75-GHz oxygen line (see Figure 2.2). The system consists of 64 channels, each of approximately 500-MHz bandwidth equally spaced over a 5-GHz intermediate frequency bandwidth. Channel 1 is the most transparent channel (farthest from the center of the oxygen line) and channel 64 is the most opaque channel (closest to the center of the oxygen line). The temperature weighting functions (see Section 2.5.4.1) for this 64-channel microwave sounder are shown in Figure 7.1. Vertical coverage of the atmosphere is most dense in the lower atmosphere where variability is the largest and becomes sparser with increasing altitude.

The channel sensitivities (ΔT_{rms}) are each approximately 0.2K. The measurements are simulated using a radiative transfer algorithm [2] and an ocean surface model [3]. The NOAA88b global ensemble [4] of over 7,000 profiles was used to produce the training, validation, and testing sets, with an

80–10–10 split of the data.

First, a linear regression temperature profile retrieval operator was derived. The linear regression RMS error on the testing set is shown in Figure 7.2. A neural network was then trained for each of two cases, one with no input compression, and one with a 64:15 compression ratio obtained using the projected principal components transform to represent the original 64 channels with only 15 degrees of freedom. Both FFMLP networks included a single hidden layer of 30 nodes that was initialized using the Nguyen-Widrow procedure and trained with the Levenberg-Marquardt learning algorithm. Random noise was added to the training set at each iteration, and early stopping (after approximately 50 epochs) was used to prevent overfitting by terminating the learning algorithm if the validation error failed to decrease for any of five successive epochs. The temperature profile was estimated at 50 levels from the surface to approximately 12 km. The neural networks were each trained three times, and the validation data set was used to select the network with the best performance.

Neural network retrieval performance for both cases is shown in Figure 7.2. The 64–30–50 network (3,500 weights) trained at a rate of 546 seconds per epoch on a desktop Intel Xeon PC operating at a clock speed of 3 GHz, while the 15–30-50 network (2,030 weights) trained at a rate of 225 seconds per epoch. In addition to faster training times afforded by the smaller network, it is interesting that the estimation error of the smaller network is also superior. This indicates that the large network did not train optimally, either because local minima were encountered, or because irrelevant features were identified in the input data and the network wasted resources fitting these features. The retrieval performance of both neural networks is superior to that of linear regression throughout the atmosphere.

This example illustrates the substantial utility of compression of the neural network inputs (radiances), both from a perspective of computational burden and error performance. Further improvement could be realized by also compressing the network outputs (the atmospheric state). Other investigators have implemented compression of network inputs and outputs to great advantage [5].

7.3 Filtering of Interfering Signals

We have demonstrated in Chapter 6 the undesirable potential of neural networks to fit any noise in the input data. It is therefore beneficial to remove noise from the inputs prior to network training. This is best accomplished if a priori statistical knowledge of the noise and its relationship to the inputs and targets is available. We now present two examples: (1) a simple application of

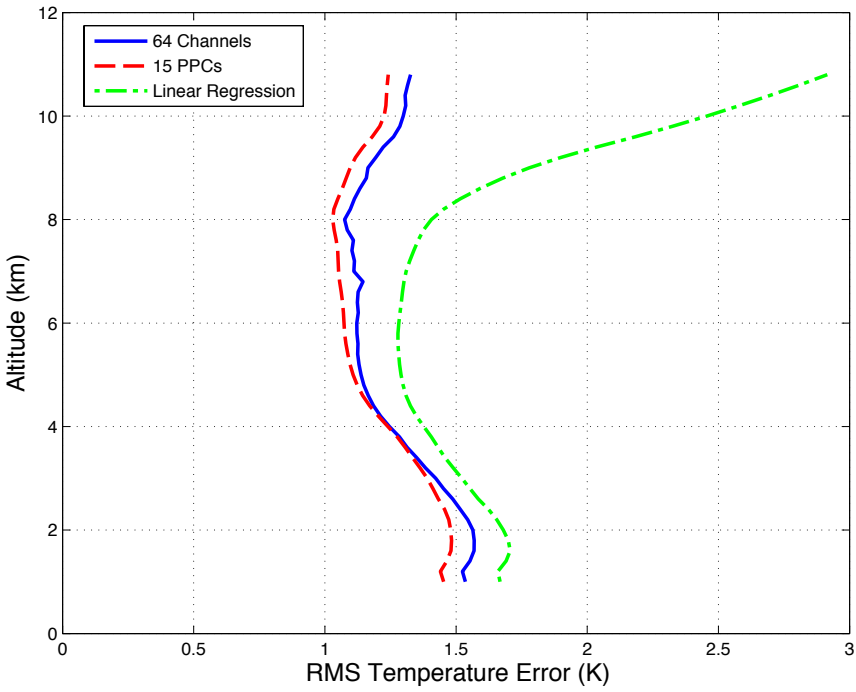


Figure 7.2 RMS temperature profile retrieval error for neural network and linear regression estimators. Two neural networks were trained, one with the unprocessed 64 channels as inputs and one using the projected principal components transform with 15 components retained.

an optimal linear filter, and (2) a more complicated methodology to remove "cloud noise" from observed hyperspectral infrared radiance observations.

7.3.1 The Wiener Filter

Returning to the simple additive noise model,

$$\widetilde{R} = R + \Psi \tag{7.5}$$

we seek the linear filter $\mathbf{H}$ such that the filtered radiance $\widehat{R} = \mathbf{H}\widetilde{R}$ minimizes the distortion given by the sum-squared-error (SSE) cost criterion:

$$c(\cdot) = E[(\widehat{R} - R)^T(\widehat{R} - R)] \tag{7.6}$$

The optimal linear filter that minimizes (7.6) is the Wiener filter,

$$\mathbf{H} = \mathbf{C}_{RR}(\mathbf{C}_{RR} + \mathbf{C}_{\Psi\Psi})^{-1} \tag{7.7}$$

where we have assumed that the additive noise is uncorrelated with the radiances.[1] It is interesting to note that the linear operator of rank r that minimizes (7.6) is not given by the r principal components of $\widetilde{R}$ (or of R), but by the reduced rank reconstruction of the Wiener filter (see Section 4.2.4).

7.3.2 Stochastic Cloud Clearing

In many atmospheric remote sensing inversion problems, the corruptive influence of random sensor noise, while problematic, is often not the dominant source of retrieval error. The influence of clouds and surface effects can be much more pernicious due to their nonlinear and non-Gaussian relationships to the observed radiances and the relatively large signal level of their contribution. Here we present a new methodology developed by Cho and Staelin [6] to estimate and remove radiance perturbations in hyperspectral infrared sounding data due to clouds. The inputs to the algorithm are the cloud-impacted microwave and infrared observations together with several variables that characterize the sensor viewing geometry and surface type (land or water). The algorithm output is the so-called infrared "cloud-cleared radiance," that is, the radiance that would have been measured by the infrared sensor had there been no cloud present.

This cloud clearing method exploits knowledge of the random (or "stochastic") nature of clouds and their statistical dependence to the infrared observations and is henceforth termed *stochastic cloud clearing* (SCC). The microwave measurements are largely unaffected by the clouds, but offer much poorer vertical resolution than do the infrared measurements. The SCC method can be regarded as both a data fusion operator (combining the cloud-immune but vertically coarse microwave measurements with the cloud-impacted but vertically fine infrared measurements) and a noise filter (the cloud perturbations are estimated and removed).

7.3.2.1 SCC Algorithm Description

The SCC algorithm attempts to estimate Atmospheric Infrared Sounder (AIRS [7]) radiances in the 3.7–15.4 micron spectral band that would be observed from space in the absence of clouds. This algorithm examines 3×3 sets of 15-km AIRS fields of view (FOVs), selects the clearest fields, and then estimates a single cloud-cleared infrared spectrum for the 3×3 set using a series of simple linear and nonlinear operations on both the infrared and companion Advanced Microwave Sounding Unit (AMSU [8]) microwave

1. If the noise is correlated with the radiances, then the covariance matrix to be inverted in (7.7) cannot be expressed simply as the sum of the radiance and noise covariances.

channels. These instruments were launched on the NASA Aqua satellite in May 2002. The SCC algorithm was both trained and tested within 70° of the equator using global numerical weather analyses generated by the European Center for Medium-range Weather Forecasts (ECMWF).

SCC is a new data-trained stochastic method for correcting effects of clouds on hyperspectral infrared radiances. This contrasts with cloud-clearing approaches employing physical models for clouds and radiative transfer. Stochastic methods are computationally efficient and can readily access information hidden in hundreds, or even thousands, of infrared channels. For example, this statistical information reflects to some unknown degree the radiance properties of three-dimensional cloud assemblies with complex shapes and hydrometeor distributions that are difficult to model physically. Stochastic clearing (SC) methods can increasingly access this information as a result of technological advances that increase computer power and the size of training data sets.

At nadir AIRS observes nine 15-km FOVs within a single AMSU 45-km FOV, which is called a "golf ball." The stochastic cloud-clearing algorithm produces one set of cleared AIRS radiances for each golf ball on the basis of inputs that include: (1) the AIRS radiances for N channels of interest, where N is generally more than 300 for each of nine FOVs, (2) the brightness temperatures for five AMSU channels sensitive to tropospheric temperatures, (3) the secant of the instrument scan angle, θ, which is zero at nadir, and (4) the a priori fraction of land in the golf ball.

The SCC algorithm is diagrammed in Figure 7.3 and consists of five main steps: (1) the FOVs to be used for each golf ball are selected and their radiances are averaged for each of N channels, (2) an initial linear estimate of cloudiness is made (operator A), (3) the cloudiness estimate is multiplied by the secant of scan angle, θ, and then, along with the inputs to operator A, this product is fed to a second linear operator (operator B), which estimates two brightness temperatures sounding low altitudes that are used to classify each golf ball as either "less cloudy" or "more cloudy," (4) a final estimate of four principal components of the radiance correction spectrum is made using operator C or D for the less or more cloudy golf balls, respectively, and (5) this correction spectrum is added to the average spectrum of the warmest FOVs for that golf ball to yield the final N cloud-cleared radiances.

7.3.2.2 SCC Results

The SCC algorithm was trained on 314 (of 2,378) AIRS channels over daytime ocean and within $\pm 40^\circ$ latitude and was then applied to a typical daytime AIRS granule obtained on July 14, 2003, more than a month earlier

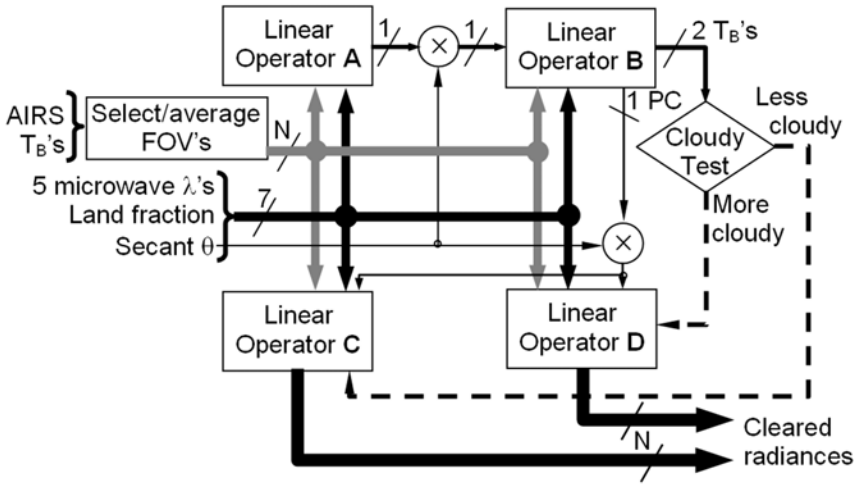


Figure 7.3 The stochastic cloud-clearing algorithm. © AGU 2006 [6].

than any of the 1,500 golf balls contained in the training set. The granule is centered southwest of Hawaii near 175°W, 5°N. Figure 7.4 (top) shows the original AIRS 15-km FOV brightness temperatures at 2,187.8 cm^{-1}; at this wave number the weighting function peaks ∼230m above the nominal surface and has some sensitivity to CO. Since CO is less significant near the equator and generally smoothly distributed locally, its contributions to Figure 7.4 are presumably negligible. Each vertical scan line contains 90 FOVs. Angle flattening has been performed toward the limb by averaging the SCC results for all scans and restoring that average decrease with angle to both the top and middle images. Within the major clouds it can be seen that only a few golf balls have even one FOV with cloud perturbations less than 5 K. Figure 7.4 (middle) shows the angle-flattened SCC cloud-cleared radiances, most of which fit within a 2-K dynamic range and, more locally, within a ∼0.6-K range. Each vertical scan line contains 30 golf balls that have been bilinearly interpolated. It is evident that most clouds have been cleared with reasonable accuracy even without any fully clear FOVs, and that only the more intense clouds remain evident. The original image is everywhere colder than the cleared image, the offset being approximately 1 K for the clearest golf balls. The cleared image has a temperature difference left-to-right of 1.36 K, whereas the corresponding difference for the NOAA/NCEP-provided sea surface temperatures in Figure 7.4 (bottom) is ∼1.6 K. Both the SCC-cleared and NCEP sea surface data exhibit the same sharp thermal front

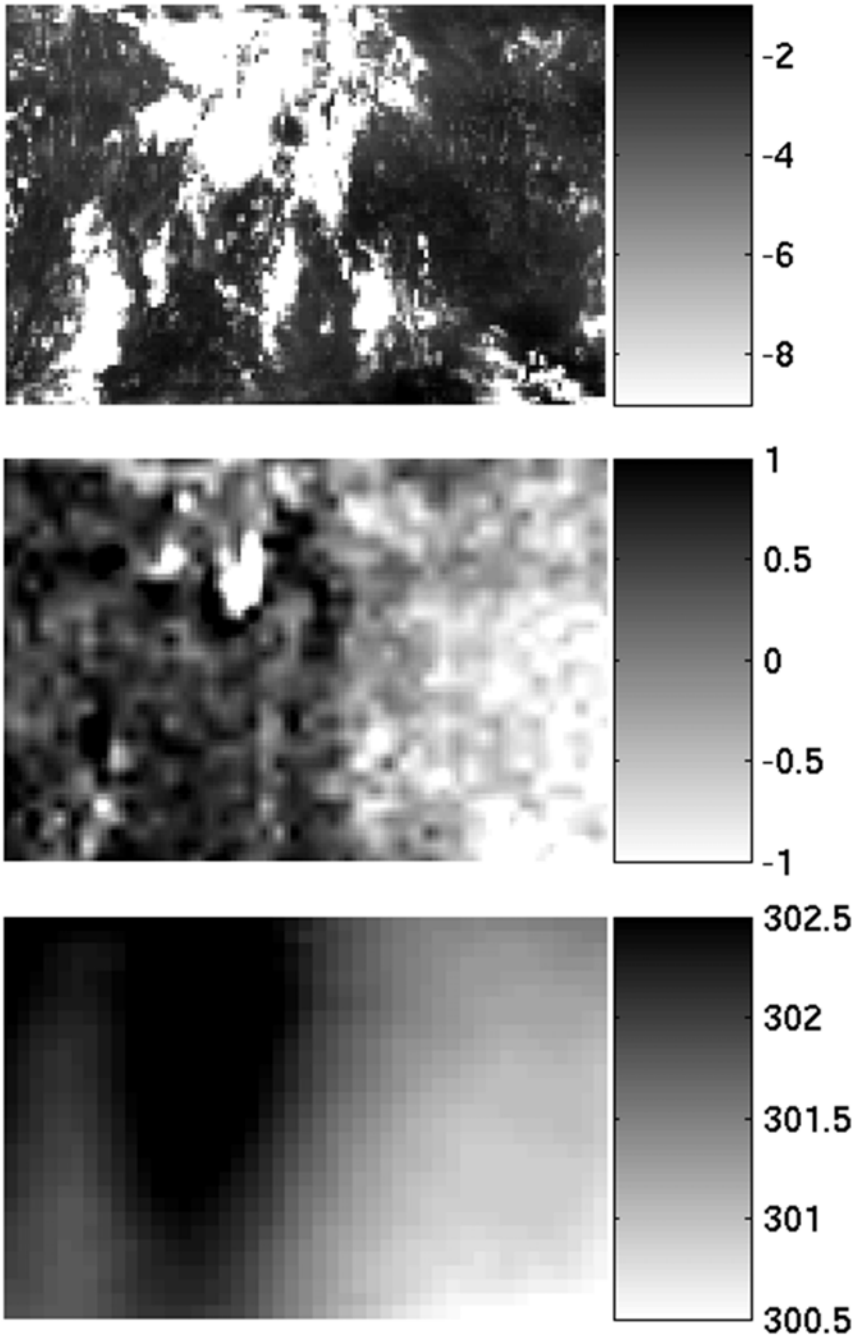


Figure 7.4 (Top) AIRS 2,187.8 cm^{-1} angle-corrected relative brightness temperatures (K) near Hawaii on July 14, 2003, (middle) the corresponding angle-corrected SC cloud-cleared temperatures, and (bottom) NOAA/NCEP estimated sea surface temperatures. © AGU 2006 [6].

centered in both images.

SCC preprocessing of cloud-impacted neural network input radiance data has recently been demonstrated to substantially improve both temperature and water vapor profile retrieval accuracy [9]. The performance was most dramatically improved in areas of heavy cloudiness, where assimilation of atmospheric data into numerical weather prediction models is perhaps most critical.

7.4 Data Warping

In addition to compressing the input and target data and reducing the interfering noise, it might also be desirable to transform the inputs and targets into a new representation that facilitates network training. Recall from Chapter 5 that kernel methods such as support vector machines cast the input data into a high-dimensional feature space that is more likely to be linearly separable. This "data warping" is essentially a preprocessing operation, which can be helpful in a broad range of neural network applications. The data warping need not be a nonlinear mapping. A simple multidimensional rotation, using a principal components transform, for example, can sometimes expedite training and improve estimation performance.

We illustrate the concept of data warping through an example involving topological processing, that is, a transformation of the data into a representation that accounts for the geometric properties and spatial relationships of the data. Perhaps the most important variation of this kind in atmospheric remote sensing is circular, or periodic, variations. Circular variations can be treated as a representative type from which all other variations requiring consideration of topological issues can be derived.

Circular data are found in many atmospheric remote sensing contexts, including diurnal, seasonal, or geographic variations that may be present in the observations of temperature profile, water vapor profile, and precipitation. Satellite observations of upwelling brightness temperature may also exhibit diurnal variations [10]. Circular variables such as time of day, day of year, and geolocation can present difficulties in estimation problems because common representations of these variables could suffer from discontinuities or topological distortions. For example, a variable t representing the time of day in hours with a range between 0 and 24 is discontinuous at midnight, and this discontinuity could complicate the analysis of variations occurring near midnight. One solution is to create a new functional relationship with an additional dimension (degree of freedom) so that the discontinuity can be removed.

One approach to problems involving circular data is to create a nonstandard neural network component that can store angular information. Kirby and Miranda [11] and Hundley et al. [12] developed circular and spherical nodes, respectively. A circular node is actually a pair of nodes whose outputs are constrained to lie on a unit circle in $\mathbf{R}^2$, and a spherical node is a set of three nodes whose outputs are constrained to lie on a unit sphere in $\mathbf{R}^3$. In those studies, bottleneck (autoassociative) neural networks with circular or spherical nodes were used to construct maps between a circle or sphere and topologically equivalent surfaces.

The data-warping approach involves changing the representation of data to facilitate learning by a conventional neural network. The extent to which a neural network is able to learn a mathematical relationship depends on many factors, one of which is the choice of inputs. This can influence the complexity necessary to train a neural network to a specified accuracy, and an increased complexity can increase the training time and network instability. For example, Pao [13] and Klassen et al. [14] proposed functional-link networks in which inputs were transformed to enhance the representations of inputs, and these transformations resulted in shorter training times.

We now present three examples involving fictitious geophysical data to illustrate data-warping techniques. In each example, conventional representations of circular data will be compared with other representations with varying degrees of topological appropriateness. Each network was trained using the Levenberg-Marquardt method and initialized with random weights using the Nguyen-Widrow method [15]. Each network was trained for up to 1,000 epochs, and the training was stopped if the RMS error of the neural network over the validation set increased over six consecutive epochs. For all examples, network topologies with up to two hidden layers and with up to 50 weights and biases were compared.

7.4.1 Function of Time of Day

Let t be the time of day in hours with $t = 0$ corresponding to midnight ($0 \leq t < 24$). Any neural network that relies on t as an input must deal with discontinuities at $t = 0$ and as $t \to 24$. Because of this discontinuity, a training algorithm is likely to treat the input ranges near 0 and 24 as if they were unrelated. One way to force continuity across the midnight discontinuity is to replace t with a set of inputs that trace a circle as t varies from 0 to 24 such as the following set (see Figure 7.5):

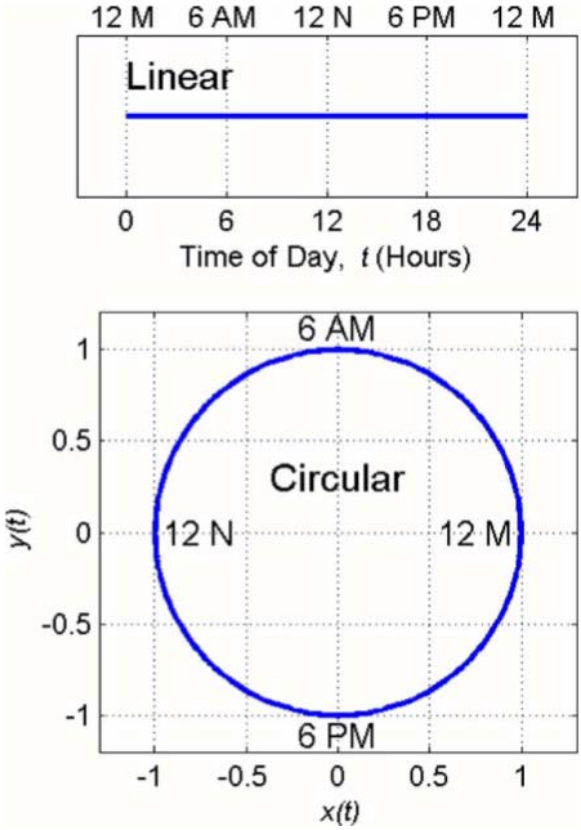


Figure 7.5 (Top) Linear and (bottom) circular representations of time of day. © 2007 IEEE [16].

$$x(t) = \cos\frac{2\pi t}{24} \tag{7.8}$$

$$y(t) = \sin\frac{2\pi t}{24} \tag{7.9}$$

With the circular representation, training examples immediately following midnight are likely to be considered highly correlated with those just before midnight. Also, this substitution is intuitively satisfying because one would expect that the training examples separated by 12 hours would be least likely to share similarities. With a linear representation, measurements at 3 A.M. and 9 P.M. would be considered 18 hours apart instead of six. The

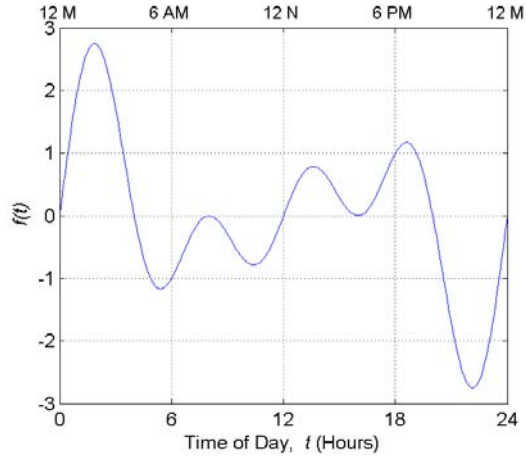


Figure 7.6 The target function for the example involving time of day. © 2007 IEEE [16].

circular representation is used to define the mean of a circular random variable (e.g., time of day, 2-D direction) in [17].

To examine the circular representation of time of day, neural networks were trained to approximate the following function (see Figure 7.6):

$$f(t) = \sin\frac{2\pi t}{12} + \sin\frac{2\pi t}{8} + \sin\frac{2\pi t}{6} \tag{7.10}$$

A total of 720 examples were randomly generated according to a uniform distribution over all t and were divided equally among the training, validation, and testing sets. Any 1-hour interval would contribute, on average, ten samples to the training set.

Figure 7.7 shows the RMS error versus the number of weights and biases for a variety of neural networks. The RMS errors for circular-domain networks tend to be significantly lower than those for the linear-domain networks with comparable numbers of weights and biases. Notable exceptions are seen in the set of data points with RMS errors near 0.5 and with over 15 weights and biases. These data points correspond to three-layer networks with only one node in the first hidden layer. In such networks, because two numbers are essentially compressed into one number, there is information loss through the first hidden layer, and the addition of weights in the second hidden layer has no effect on performance. The performance of such networks is comparable to the performance of a two-layer network with one hidden node, which in this case is 0.86. The periodic structure in the error curves results

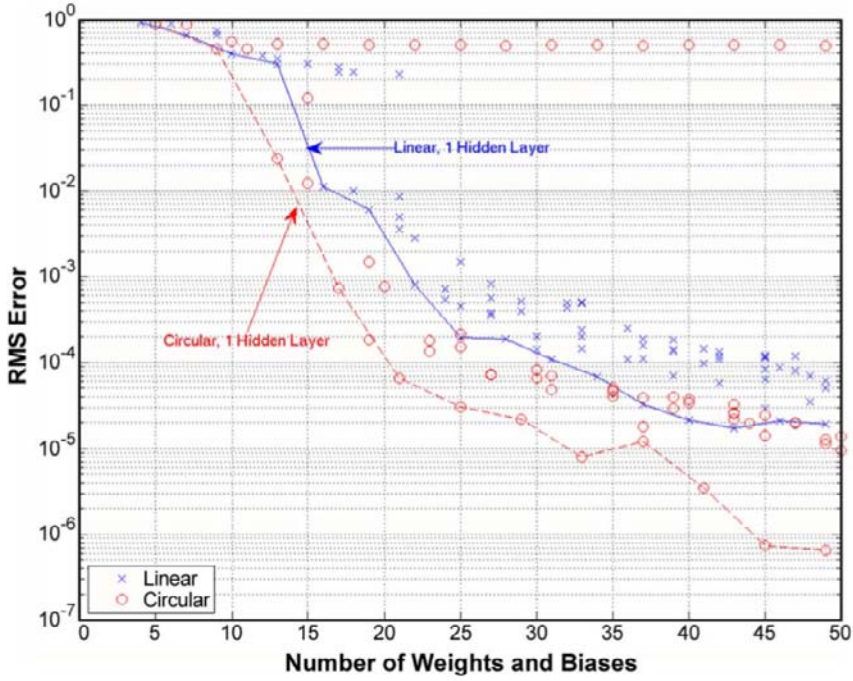


Figure 7.7 RMS error versus the number of weights and biases to be optimized for neural networks learning the fictitious geophysical function of time of day. © 2007 IEEE [16].

from the incremental addition of weights and biases to the two- and three-layer networks. Addition of a new node to a two-layer network results in the addition of four weights and biases (one weight for each of two inputs, one bias, and one weight for the subsequent output layer). Addition of a new node to the second hidden layer of a three-layer network results in the addition of $N + 2$ weights and biases, where N is the number of nodes in the first hidden layer of the network.

It is interesting to compare the linear-domain and circular-domain networks by examining the RMS errors achievable with a maximum number of weights and biases. In networks with fewer than 15 weights and biases, the linear-domain networks achieve an RMS error of 0.303 while circular-domain networks achieve an RMS error of 0.0123, an improvement of a factor of almost 25. The linear-domain and circular-domain networks can also be compared by determining the minimum number of weights and biases necessary to achieve a specified level of performance. For example, to achieve

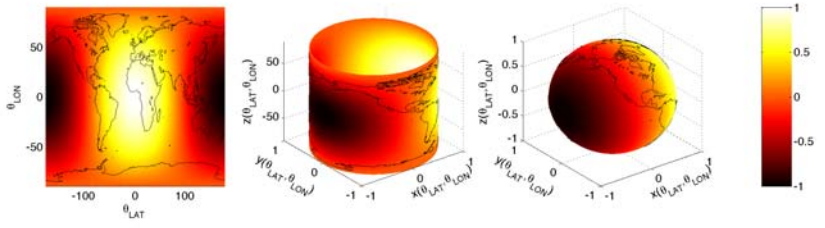


Figure 7.8 The fictitious geophysical function of geolocation shown using rectangular (left), cylindrical (middle), and spherical (right) representations. © 2007 IEEE [16].

an RMS error of 0.03 or less, a linear-domain network requires 16 weights and biases while a circular-domain network requires only 13.

7.4.2 Function of Geolocation

We now evaluate the following three representations of geolocation (that is, the position of a point on the surface of the Earth): rectangular, cylindrical, and spherical (see Figure 7.8). The rectangular representation is described as follows:

$$x(\theta_{LAT}, \theta_{LON}) = \theta_{LON} \quad (7.11)$$

$$y(\theta_{LAT}, \theta_{LON}) = \theta_{LAT} \quad (7.12)$$

where θ_{LAT} and θ_{LON} are the latitude and longitude, respectively, in degrees. The rectangular representation is discontinuous at $180°$ E/W (the International Date Line), which results in points around $180°$ E/W being treated as unrelated when they are in fact very close to each other. Another problem is the topological distortions around $90°$ N and $90°$ S, the Geographic North and South Poles, respectively. The poles are treated not as points but as curves, therefore, points such as $90°$ N, $90°$ E and $90°$ N, $90°$ W are likely to be misinterpreted as unrelated points even though they are in fact the same point.

The cylindrical representation is described as follows:

$$x(\theta_{LAT}, \theta_{LON}) = \cos\theta_{LON} \quad (7.13)$$

$$y(\theta_{LAT}, \theta_{LON}) = \sin\theta_{LON} \quad (7.14)$$

$$z(\theta_{LAT}, \theta_{LON}) = \theta_{LAT} \quad (7.15)$$

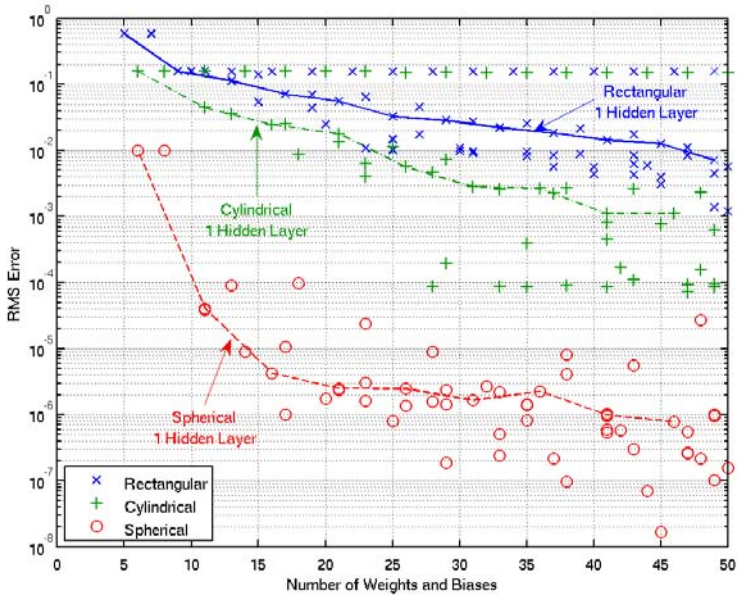


Figure 7.9 RMS error versus the number of weights and biases to be optimized for neural nets learning the fictitious geophysical function of geolocation. © 2007 IEEE [16].

This representation eliminates the discontinuity along the International Date Line in the rectangular representation. However, the topological distortions at the poles remain.

The spherical representation is described as follows:

$$x(\theta_{LAT}, \theta_{LON}) = \cos\theta_{LAT} \cos\theta_{LON} \tag{7.16}$$

$$y(\theta_{LAT}, \theta_{LON}) = \cos\theta_{LAT} \sin\theta_{LON} \tag{7.17}$$

$$z(\theta_{LAT}, \theta_{LON}) = \sin\theta_{LAT} \tag{7.18}$$

The spherical representation eliminates both the discontinuity along the International Date Line in the rectangular representation and the topological distortions at the poles in the rectangular and cylindrical representations. As with the circular representation of time of day, the spherical representation of geolocation preserves the intuitive notion of physical separation. A similar transformation is used to define mean 3-D direction in [18].

Neural networks were trained to approximate the function shown in Figure 7.8 using rectangular, cylindrical, and spherical representations of the geolocation (input) data. This function is a simple dot product of the position vector in 3-D rectangular coordinates of a given point with that of 0° N/S, 0° E/W. Approximately 3,000 examples were randomly generated according to a uniform distribution over the entire globe and were divided equally among the training, validation, and testing sets. For example, a circle with a radius of 500 km would have on average 1.5 samples from the training set.

Figure 7.9 shows the RMS error versus the number of weights and biases for the neural network with the lowest RMS error for each topology. The cylindrical representation, while an imperfect representation of a spherical set, still results in a significant improvement over the rectangular representation since it captures the circular variation over all longitudes. However, the spherical representation results in a much greater improvement. This is clearly seen in Figure 7.9 as the RMS error values for the spherical representation occupy a region lower than and distinct from that occupied by the RMS error values for the rectangular and cylindrical representations. For networks with up to 10 weights and biases, spherical-domain nets achieved RMS errors more than a factor of ten lower than those achieved with rectangular-domain and cylindrical-domain nets. For nets with up to 17 weights and biases, spherical-domain nets achieved RMS errors lower by more than four orders of magnitude.

It is clear that a spherical representation for this geolocation problem will lower the number of weights and biases necessary to achieve a specified performance. For example, to achieve an RMS error of less than 0.015, a network that uses the rectangular representation needs 23 weights and biases, a network that uses the cylindrical representation needs 18, and a network that uses the spherical representation needs only 6.

7.4.3 Function of Time of Year

In this example, a neural network was trained to learn a fictitious temperature anomaly function over the course of one year, as temperature typically depends on both the time of day and season.

Time can be represented as a single number over an entire year, or as two numbers: one representing the time of day and one representing the day of year. Each of these numbers can also be represented circularly. We consider the following six representations:

- **Linear**: Time of year as a single number
- **Circular**: Time of year represented circularly

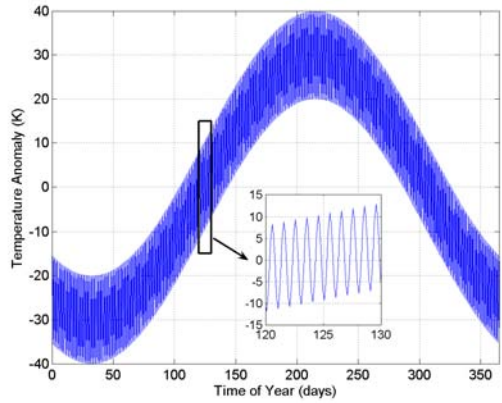


Figure 7.10 The fictitious temperature anomaly function as a function of time of year. © 2007 IEEE [16].

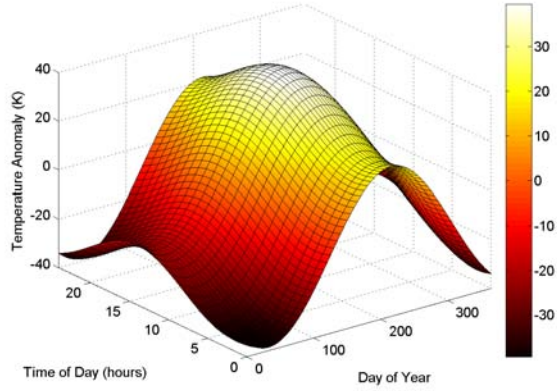


Figure 7.11 Fictitious temperature anomaly function of time of day and day of year. © 2007 IEEE [16].

- **Rectangular**: Time of day and day of year each represented as single numbers
- **Cylindrical with circular day of year**: Day of year represented circularly and time of day represented as a single number
- **Cylindrical with circular time of day**: Time of day represented circularly and day of year represented as a single number

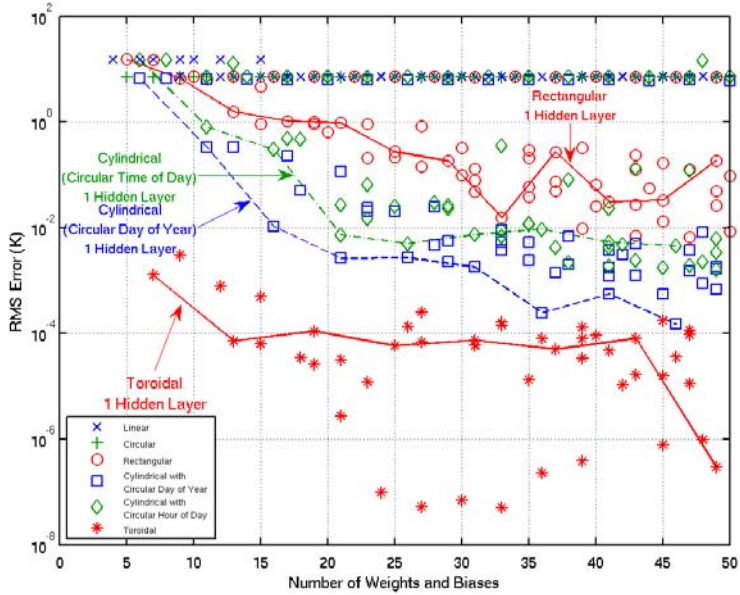


Figure 7.12 RMS error versus the number of weights and biases to be optimized for the fictitious temperature anomaly function. © 2007 IEEE [16].

- **Toroidal**: Time of day and day of year each represented circularly

The first two representations will be called *1-manifold representations*, and the remaining will be called *2-manifold representations*.

Figures 7.10 and 7.11 show the fictitious temperature anomaly function to be learned as a function of time of year and as a function of day of year and time of day, respectively. Approximately 1,250 times of year were randomly selected and divided equally among the training, validation, and testing sets. Any week would include on average 8 training samples.

Figure 7.12 shows the RMS error versus the number of weights and biases for the neural network with the lowest RMS error for each topology. The toroidal representation results in lower RMS errors than any of the other representations. For networks with fewer than 10 weights and biases, the best RMS error achieved with the toroidal representation was almost four orders of magnitude lower than the best achieved with any other representation. Moreover, to achieve the RMS error of the simplest network topology using

the toroidal representation (with 7 weights and biases), a network using any other representation would need at least 36 weights and biases.

Several other notable observations can be made from Figure 7.12. First, 2-manifold representations result in lower RMS errors than 1-manifold representations. Moreover, networks using 1-manifold representations were not able to achieve RMS errors lower than about 10. The poor performance of nets using 1-manifold representations is not surprising since the derivative of the temperature anomaly function with respect to time of year changes sign 730 times (Figure 7.10). The discussion in Section 5.3.1.6 suggests that a two-layer neural network using a linear representation would need at least 365 hidden nodes to accurately represent all of the sign changes in the derivative, resulting in a neural net with at least 1,096 weights and biases to be optimized. On the other hand, when the temperature anomaly function is expressed as a function of time of day and day of year separately, the functional shape is much simpler (see Figure 7.11). This shape permitted good approximations with 2-manifold representations using a sub-Nyquist sampled training set. Second, the cylindrical representation with circular day of year tends to result in lower RMS errors than the cylindrical representation with circular time of day. This is to be expected because the seasonal variation is larger than the diurnal variation.

7.5 Summary

In this chapter, we have presented several types of pre- and post-processing methods that can both simplify neural network training and improve estimation performance. Three broad categories of data processing were discussed: compression, noise filtration, and warping. Data compression allows the data to be represented in more statistically compact form. Noise filtration removes interfering signals that may be wastefully fitted by the neural network. Data warping transforms the data into a new multidimensional space that reveals features that are more easily learned. While we have focused on the preprocessing of neural network inputs in this chapter, the concepts are equally applicable to the post-processing of neural network outputs.

References

[1] H. S. Malvar and D. H. Staelin. "Optimal pre- and postfilters for multichannel signal processing." *IEEE Trans. Acoust. Speech Signal Process.*, 36(2):287–289, February 1988.

[2] P. W. Rosenkranz. "Radiative transfer solution using initial values in a scattering and absorbing atmosphere with surface reflection." *IEEE Trans. Geosci. Remote Sens.*, 40(8):1889–1892, August 2002.

[3] S. English and T. Hewison. "A fast generic millimeter-wave emissivity model." *SPIE Proceedings*, 3503:288–300, 1998.

[4] E. Borbas, S. Seemann, H. L. Huang, J. Li, and W. P. Menzel. "Global profile training database for satellite regression retrievals with estimates of skin temperature and emissivity." *Proc. Int. ATOVS Study Conf.*, 14, 2005.

[5] F. Del Frate and G. Schiavon. "A combined natural orthogonal functions/neural network technique for the radiometric estimation of atmospheric profiles." *Radio Sci.*, 33(2):405–410, March 1998.

[6] C. Cho and D. H. Staelin. "AIRS observations versus numerical weather predictions of cloud-cleared radiances." *J. of Geophys. Res.*, 111, 2006.

[7] H. H. Aumann, et al. "AIRS/AMSU/HSB on the Aqua mission: Design, science objectives, data products, and processing systems." *IEEE Trans. Geosci. Remote Sens.*, 41(2):253–264, February 2003.

[8] B. H. Lambrigtsen. "Calibration of the AIRS microwave instruments." *IEEE Trans. Geosci. Remote Sens.*, 41(2):369–378, February 2003.

[9] W. J. Blackwell, M. Pieper, and L. G. Jairam. "Neural network estimation of atmospheric profiles using AIRS/IASI/AMSU data in the presence of clouds." *SPIE Asia-Pacific Remote Sensing Symposium*, November 2008.

[10] T. Mo. "Diurnal variation of the AMSU-A brightness temperatures over the Amazon rainforest." *IEEE Trans. Geosci. Remote Sens.*, 45(4):958–969, April 2007.

[11] M. J. Kirby and R. Miranda. "Circular nodes in neural networks." *Neural Computation*, 8(2):390–402, February 1996.

[12] D. R. Hundley, M. J. Kirby, and R. Miranda. "Spherical nodes in neural networks." *Intelligent Engineering Systems through Artificial Neural Networks: Proceedings of Artificial Neural Networks in Engineering* (ed. C.H. Dagli), 5:27–32, 1995.

[13] Y. Pao. *Adaptive Pattern Recognition and Neural Networks*. Addison-Wesley, Reading, Massachusetts, 1989.

[14] M. Klassen, Y. Pao, and V. Chen. "Characteristics of the functional link net: A higher order delta rule net." *Proc. IEEE Int. Conf. Neural Networks*, pages 507–513, 1988.

[15] D. Nguyen and B. Widrow. "Improving the learning speed of two-layer neural networks by choosing initial values of the adaptive weights." *IJCNN*, 3:21–26, 1990.

[16] F. W. Chen. "Neural network characterization of geophysical processes with circular dependencies." *IEEE Trans. Geosci. Remote Sens.*, 45(10):3037–3043, October 2007.

[17] N. I. Fisher. *Statistical Analysis of Circular Data*. Cambridge University Press, New York, 1993.

[18] N. I. Fisher, T. Lewis, and B. J. J. Embleton. *Statistical Analysis of Spherical Data*. Cambridge University Press, New York, 1987.

8

Neural Network Jacobian Analysis

We have previously discussed how separate training, validation, and testing data sets are used to ensure that an optimized neural network not only estimates the target values with sufficient accuracy, but also generalizes well to inputs that are not contained in the training set. In this chapter, we further evaluate performance by examining the neural network Jacobian, that is, the sensitivity of the output values to changes in either the input values or the network weights. Jacobian analysis can be used to assess neural network performance in a variety of ways [1, 2]. For example, the effect of sensor noise (or other interfering signals) on retrieval accuracy can be easily evaluated. Jacobians provide information on the relevance of network inputs and can therefore be used (usually in concert with other techniques) to select only the most significant inputs. Finally, Jacobians facilitate system optimization, where various parameters of the observing system can be optimized jointly.

A complete performance characterization of a given atmospheric retrieval algorithm requires an analysis of the retrieval errors, as discussed in Chapter 3. The retrieval error depends on several components, including sensor noise, the vertical resolution of the sensor, and many others. The assessment of these components in isolation is difficult for complex retrieval schemes, and finite-differencing methods are often used to approximate the effects of these contributions over a limited set of atmospheric cases. A significant advantage of atmospheric retrievals based on neural networks is that Jacobians can be calculated analytically, and the calculations can be carried out using relatively simple methods, such as the backpropagation algorithm presented in Chapter 6. We now present basic methods for calculating neural network Jacobians and simple examples illustrating system characterization and optimization.

8.1 Calculation of the Neural Network Jacobian

The retrieval averaging kernel was introduced in Section 3.6.2:

$$\frac{\partial \widehat{S}}{\partial S} = \frac{\partial R}{\partial S}\frac{\partial \widehat{S}}{\partial R} \tag{8.1}$$

We now focus on the second term in (8.1), the retrieval Jacobian, which consists of the partial derivatives of the outputs (estimates of the atmospheric state vector, S) with respect to the inputs (the radiance vector, R). We reconcile neural network and atmospheric retrieval terminologies by equating the neural network inputs to the observed radiances, $X = R$, and equating the neural network outputs to the estimates of the atmospheric states, $Y = \widehat{S}$. Returning to the equation given in Section 5.2 relating the inputs and outputs of a simple feedforward multilayer perceptron with one hidden layer of m nodes and a single output, y_k:

$$y_k = g\left(\sum_{j=1}^{m} v_{jk} f\left(\sum_{i=1}^{n} w_{ij} x_i + b_j\right) + c_k\right) \tag{8.2}$$

we can express the derivative dy_k/dx_i using the chain rule, as follows:

$$\frac{dy_k}{dx_i} = g'(a_k)\sum_{j=1}^{m} v_{jk} f'(a_j) w_{ij} \tag{8.3}$$

where a_k and a_j are the weighted sum of the inputs to the output and hidden layers, respectively. We assume that all inputs other than x_i are fixed. It was mentioned in Section 5.1.5.1 that the derivative of the hyperbolic tangent function is related to the function itself as

$$f' = 1 - f^2 \tag{8.4}$$

If a linear output layer is used, $g'(a_k) = 1$, and (8.3) becomes

$$\frac{dy_k}{dx_i} = \sum_{j=1}^{m} v_{jk}(1 - f^2(a_j)) w_{ij} \tag{8.5}$$

and we see that the Jacobian is easily calculated from the network outputs and weights. This result for a network with a single hidden layer is readily generalized to networks with multiple hidden layers. Note that the mapping function implemented by the neural network could be highly nonlinear, and

therefore care must be taken to ensure that the Jacobian is evaluated near an appropriate operating point. The network Jacobian is generated as a simple "byproduct" of the forward propagation of the inputs through the network. Because of this, neural networks are well suited to complicated function approximation problems, as the computation required for error analysis is greatly reduced in comparison to other methods requiring numerical finite-difference techniques.

8.2 Neural Network Error Analysis Using the Jacobian

We now present a typical case study to illustrate several facets of retrieval system analysis using the Jacobian. We return to the neural network example presented in Section 7.2, which is based on a simulated spaceborne microwave sounding system with 64 channels operating near the opaque 118.75-GHz oxygen line used to retrieve the temperature profile at 50 levels. The NOAA88b global ensemble [3] of over 7,000 profiles was used to produce the training, validation, and testing sets, with an 80–10–10 split of the data. An FFMLP network with a single hidden layer of 30 nodes was initialized using the Nguyen-Widrow procedure and trained with the Levenberg-Marquardt learning algorithm. Random noise ($\sigma = 0.2$ K) was added to the training set at each iteration, and early stopping was used to prevent overfitting. Both the simulated radiances and the temperature profile elements were normalized to unit standard deviation and zero mean to simplify the interpretation of the resulting Jacobians.

8.2.1 The Network Weight Jacobian

The Jacobians (both with respect to the network weights and inputs) can now be calculated using the trained neural network as discussed in Section 8.1. First, we will examine the "network weight Jacobian," the derivative of the outputs with respect to the weights. It was shown in Chapter 6 that this derivative is the primary component of network learning algorithms based on gradient descent. Large values of the Jacobian indicate that a given weight has high influence on the output parameter (that is, the output is highly sensitive to the value of that weight) and small values indicate a small influence. The network weight Jacobian for the microwave sounder example is shown in Figure 8.1. Only the 1,920 (64 inputs $\times$ 30 nodes) hidden layer weights are shown in the figure. The weights have been arranged in increasing order of the derivative at 10 km. There are several interesting features evident in this figure. First, a band of approximately 100 weights (near the vertical center in the figure) have very small influence on any output. This suggests that the neural

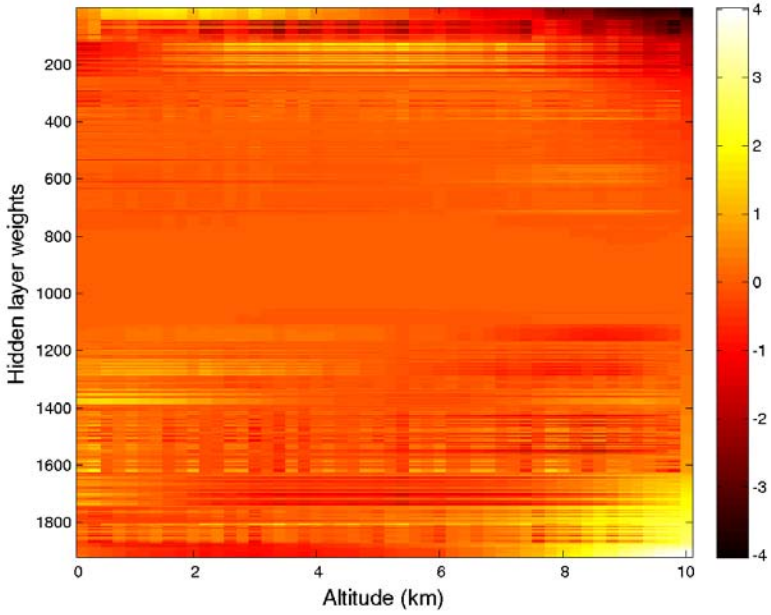


Figure 8.1 The derivative of the neural network output with respect to the network weights in the hidden layer. The network outputs consist of the temperature profile estimates from 0 to 10 km in 200-m steps.

network could be simplified by removing the connections corresponding to the least influential weights. Second, outputs corresponding to altitudes above 8 km or so are highly influenced (both in the positive and negative directions) by some of the weights. This could indicate that the relationships between the radiometric observations and the temperature at these altitudes are substantially nonlinear. Recall that in Section 3.3.2.2 a similar example demonstrated this in fact to be the case, because the addition of high-order terms to the polynomial regression operator significantly reduced the retrieval error in altitudes exceeding 8 km.

8.2.2 The Network Input Jacobian

We now consider the "network input Jacobian," the derivative of the outputs with respect to the inputs. This Jacobian reveals the inputs that most significantly influence the outputs. The influence of input perturbations on

the outputs can be calculated as follows [4]:

$$\Delta y_k \simeq \sum_i \frac{dy_k}{dx_i} \Delta x_i \tag{8.6}$$

This simple formalism allows many attributes of a remote sensing system to be evaluated by propagating the perturbations through to the outputs using the Jacobian. Furthermore, second-order analysis is facilitated by expressing the output covariance matrix as a function of the input covariance (or the covariance of the input perturbation) and the Jacobian as follows:

$$\mathbf{C}_{\Delta Y} \approx \frac{\partial Y}{\partial X} \mathbf{C}_{\Delta X} \left(\frac{\partial Y}{\partial X} \right)^T \tag{8.7}$$

Second-order analysis of this type is a very powerful tool both for diagnostic and optimization purposes, and we shall introduce examples of each later in this chapter.

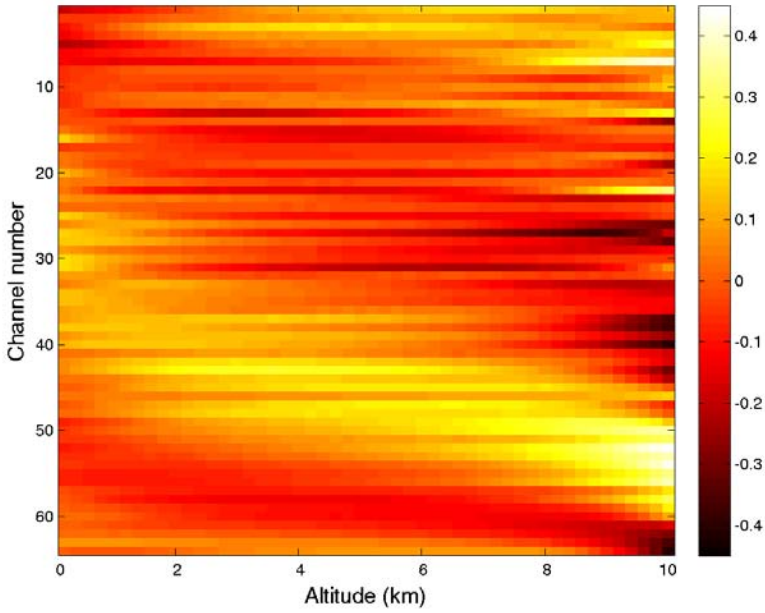
The neural network input Jacobian for the 64-channel microwave example is shown in Figure 8.2(a). It is evident from the figure that higher channel numbers generally exhibit more influence on outputs corresponding to higher altitudes, as is expected, because the channels are ordered in increasing opacity. Furthermore, outputs corresponding to higher altitudes generally are more influenced by the inputs than are the low-altitude outputs.

For comparison, the analogous Jacobian image for a linear regression operator is shown in Figure 8.2(b). There are general features common to both images. However, the smooth gradients in the linear regression Jacobian image are immediately apparent, in contrast to the high-frequency structure in the corresponding neural network Jacobian image. This structure could be evidence of nonlinear or non-Gaussian relationships between the inputs and the outputs that the neural network is exploiting.

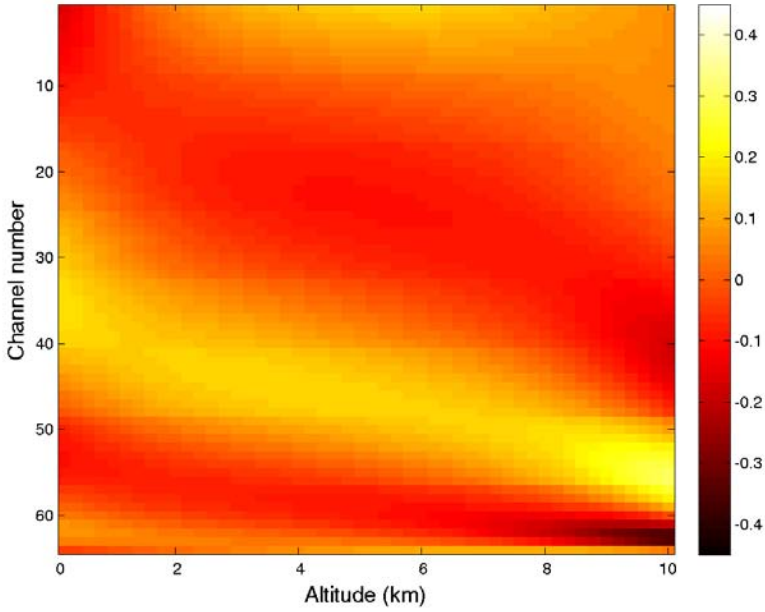
8.2.3 Use of the Jacobian to Assess Noise Contribution

To explore these Jacobians further, we now use them to compute the contribution of random measurement noise to the retrieval error using (8.7) with $\mathbf{C}_{\Delta X}$ replaced by the noise covariance matrix $\mathbf{C}_{\Psi\Psi}$. The bottom panel of Figure 8.3 shows the relative noise contribution for the neural network and linear regression estimators. It is interesting that the noise contribution from the linear regression retrieval is smaller than that for the neural network for all but the lowest 1 km of the atmosphere.

The top panel of Figure 8.3 shows the normalized RMS retrieval errors for each retrieval technique, both with and without measurement noise. The



(a) Neural network



(b) Linear regression

Figure 8.2 The derivative of the retrieval output with respect to the inputs for: (a) neural network and (b) linear regression. The outputs consist of the temperature profile estimates from 0 to 10 km in 200-m steps.

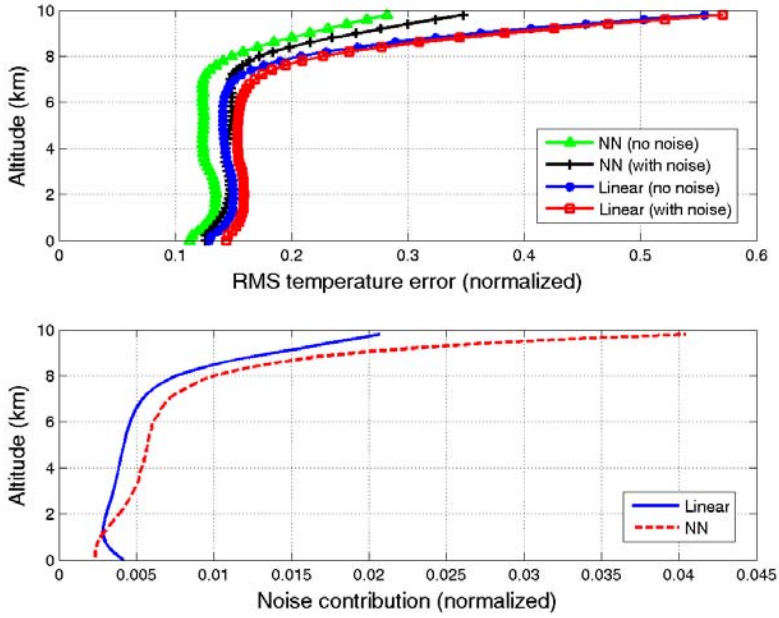


Figure 8.3 The contribution of the retrieval error due to sensor noise for the neural network and linear regression retrievals.

neural network retrieval RMS error in the absence of noise is substantially better than the corresponding linear regression retrieval without noise. In fact, the neural network retrieval in the presence of noise exceeds the performance of the linear regression without noise at almost every level in the atmosphere (there are exceptions near 5–6 km). This indicates that the neural network is in fact exploiting higher-order statistical relationships, even at the expense of increasing the noise contribution relative to the linear regression retrieval. The net effect is that the neural network retrieval performance substantially exceeds that of the linear regression retrieval throughout the atmosphere.

8.3 Retrieval System Optimization Using the Jacobian

We conclude the chapter with an example demonstrating the use of the network input Jacobian to optimize a system parameter. The Advanced Technology Microwave Sounder (ATMS) is a cross-track total-power spectrometer system that will fly as part of the National Polar-Orbiting Environmental

Satellite System (NPOESS) beginning in 2011 [5]. When operational, ATMS will be the first microwave cross-track sounder to offer spatial Nyquist sampling in the temperature sounding channels. For example, two channels (23.8 and 31.4 GHz) will provide information on integrated water vapor and cloud liquid water over ocean at an antenna beamwidth (full-width at half maximum, or FWHM) of 5.2 degrees, corresponding to a nadir footprint diameter of approximately 80 km. However, these channels are spaced in increments of 1.1 degrees, both down-track (parallel to satellite motion) and cross-track (perpendicular to satellite motion). Each spatial dimension is oversampled by almost a factor of five ($5.2/1.1$), and the spatial resolution can therefore be effectively "sharpened" using a spatial high-pass filter.

This situation results in an interesting optimization problem, with two competing performance metrics. As the beam is sharpened, the sensor noise is amplified, and this tends to increase the retrieval error. As the beam is broadened, the atmosphere is smoothed, and this also tends to increase the retrieval error. The question is then, "How much spatial sharpening (spatial filtering) should be applied to these channels to optimize the retrieval performance?" The neural network input Jacobian allows this question to be directly and quantitatively addressed with simple perturbation analysis.

8.3.1 Noise Smoothing Versus Atmospheric Smoothing

We begin by examining the trade-off between noise amplification (resulting from sharpening the beam) and atmospheric smoothing (resulting from broadening the beam). Figure 8.4 shows the RMS contribution of each of these components to the retrieval error. These errors were derived using an ensemble of atmospheric scenes. The ATMS footprints were convolved with the atmospheric scenes, and the smoothing error and sensor noise amplification factors were computed. The spatial processing substantially improves upon the performance obtained using only the native footprint (indicated in the figure by the star and the diamond for the 23.8- and 31.4-GHz channels, respectively).

To optimize retrieval performance, we pick a point on each of these curves that lies "close" to the origin, although the relative contribution of each error component on the retrieval performance is not obvious. A brute-force optimization approach is to perform a retrieval for a variety of beam sharpening parameters (that is, spatial cutoff frequencies) and pick the parameters that result in the best retrieval performance. This approach requires the execution of many retrievals, and this is time-consuming and inefficient. With neural network Jacobians, a simple optimization problem can

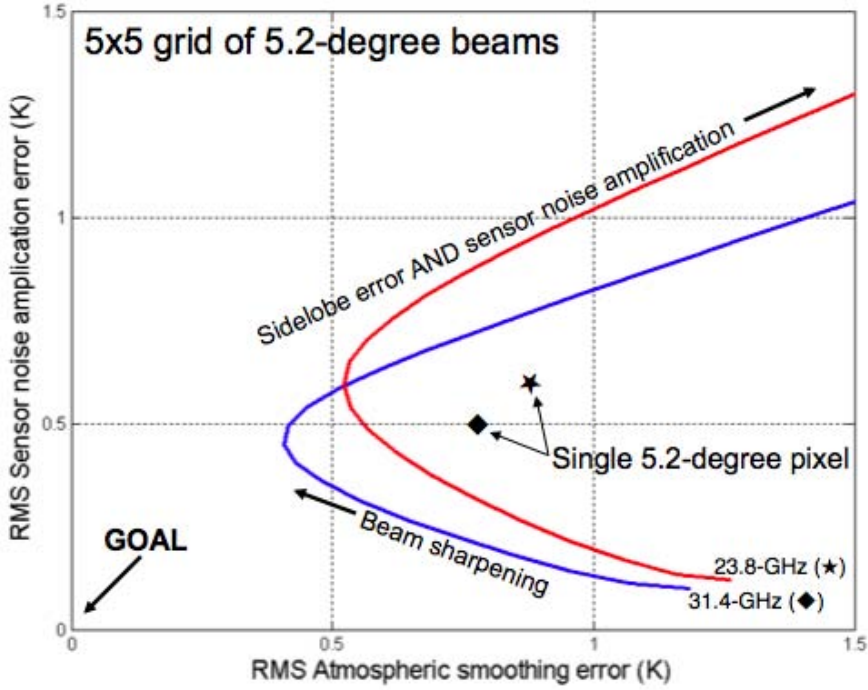


Figure 8.4 Trade-off between beam sharpening (noise amplification) and beam broadening (atmospheric smoothing) for the ATMS 23.8- and 31.4-GHz channels.

be constructed, and the parameters can then be optimized with only a single retrieval to a very good approximation.

8.3.2 Optimization Approach

For a given channel, we define the finite impulse response spatial linear filter, H, to be convolved with the radiance measurements, R, as follows:

$$R_{\text{filtered}} = \sum_{i=1}^{N} H(i)R(i) \tag{8.8}$$

where the fields of view, $R(i)$, are typically arranged in a 5×5 square grid (that is, $N = 25$). We can then calculate the sensor noise and atmospheric

noise contributions for a given choice of H:

$$\sigma^2_{\text{filtered\_noise}} = \sigma^2_{\text{sensor\_noise}} \sum_{i=1}^{N} H(i)^2 \tag{8.9}$$

$$\sigma^2_{\text{atm\_noise}} = \frac{\sum_{j=1}^{M} \left(R_{\text{filtered}}(j) - R_{\text{truth}}(j)\right)^2}{M} \tag{8.10}$$

The atmospheric noise is determined by calculating the mean-squared difference between the filtered radiances and the "true" radiances derived from a high-resolution two-dimensional measurement field consisting of M footprints. We then pick H to minimize the variance of the retrieval output perturbation:

$$\sigma^2_y(H) = \left(\frac{dy}{dx}\right)^2 \left(\sigma^2_{\text{filtered\_noise}} + \sigma^2_{\text{atm\_noise}}\right) \tag{8.11}$$

We have considered only a single channel and a retrieval of a single parameter, but this analysis is readily extended to the vector cases.

8.3.3 Optimization Results

We now demonstrate this optimization approach using ATMS retrievals of integrated water vapor (IWV) and integrated cloud liquid water (ICLW) over ocean. Two neural networks were trained to estimate these scalar parameters (one parameter for each network). A simple gradient descent procedure was then used to optimize the spatial filters (25 coefficients) for each case by minimizing (8.11) with respect to H. The results are shown in Figure 8.5, and are indicated by the solid circles. Also plotted in the figure are curves showing the retrieval error for a range of antenna beamwidths. This analysis suggests that beam sharpening is not needed for these two ATMS channels, as the optimized retrieval error is obtained with a synthesized beamwidth that is very close to the native beamwidth of 5.2 degrees. Note that the optimal filter does improve performance with respect to no filtering (indicated on the plot by asterisks), because the filtering does reduce sensor noise.

8.4 Summary

Neural network Jacobian analysis is a powerful tool that can be used to assess a variety of performance metrics. The network Jacobians are easily calculated using analytical expressions related to network outputs, and therefore very little additional computation is required. The Jacobians can be used to

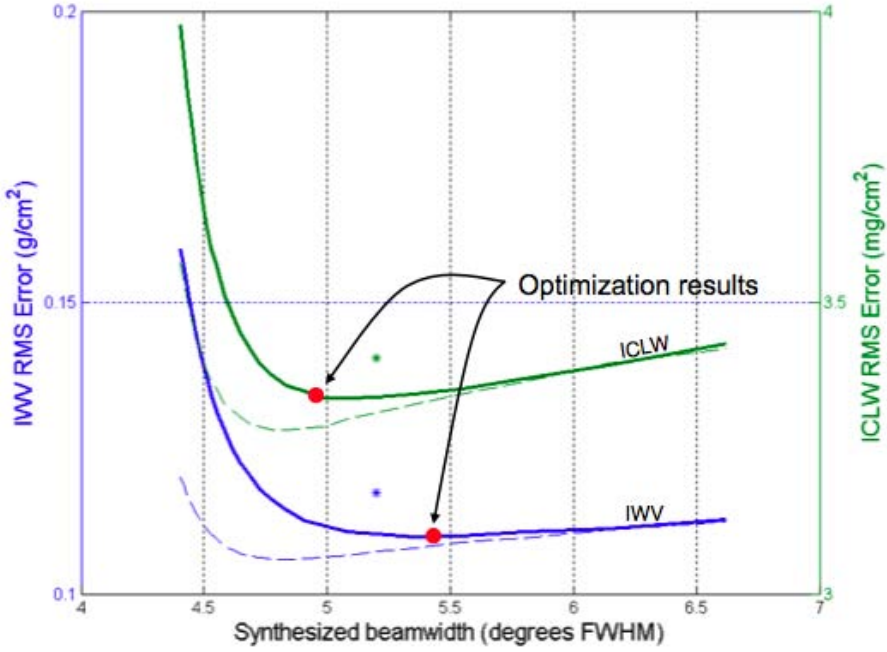


Figure 8.5 Results of the spatial filter optimization for ATMS 23.8- and 31.4-GHz channels is shown. The retrieved scalar parameters are integrated water vapor (IWV) and integrated cloud liquid water (ICLW), both over ocean. The retrieval error obtained with the optimized filter for each case is indicated with a solid circle. The asterisks indicate the retrieval performance with no spatial filtering (that is, a two-dimensional impulse response) and the dashed lines indicate performance with sensor noise set to zero.

determine the sensitivity of the network outputs to changes in both the network weights and the inputs. This information can provide insight into network topology optimization by identifying connections that do not significantly influence the outputs. Jacobians are also useful for perturbation analysis, where input perturbations due to sensor noise or other interfering signals can be easily propagated through the neural network, and the resulting impact on the outputs can be determined.

References

[1] H. E. Motteler, L. L. Strow, L. McMillin, and J. A. Gualtieri. "Comparison of neural networks and regression based methods for temperature retrievals." *Appl. Opt.*, 34(24):5390–5397, August 1995.

[2] F. Aires, C. Prigent, and W. B. Rossow. "Neural network uncertainty assessment using Bayesian statistics with application to remote sensing: 2. Output errors." *J. Geophys. Res.*, 109, May 2004.

[3] E. Borbas, S. Seemann, H. L. Huang, J. Li, and W. P. Menzel. "Global profile training database for satellite regression retrievals with estimates of skin temperature and emissivity." *Proc. Int. ATOVS Study Conf.*, 14, 2005.

[4] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, Oxford, U. K., 1995.

[5] C. Muth, P. S. Lee, J. C. Shiue, and W. A. Webb. "Advanced technology microwave sounder on NPOESS and NPP." *IEEE International Geoscience and Remote Sensing Symposium, 2004.*, 4:2454–2458, September 2004.

9

Neural Network Retrieval of Precipitation from Passive Microwave Observations

In the preceding chapters, we described the physical basis of atmospheric remote sensing using passive microwave and infrared measurements from satellites and the ability of neural networks to learn the mathematical relationships between these measurements and the state of the atmosphere. We now proceed to describe studies in which neural networks have been used to develop atmospheric estimation algorithms. The algorithms presented in this book use many of the concepts presented in the preceding chapters.

In this chapter we present an algorithm for estimating precipitation using passive microwave measurements from the Advanced Microwave Sounding Unit (AMSU) aboard the National Oceanic and Atmospheric Administration NOAA-15 satellite [1]. The highly nonlinear and non-Gaussian nature of the relationships between the satellite observations and the intensity and type of precipitation is well suited to neural network estimation. Furthermore, we will see that physical and statistical properties of the observations and the atmospheric phenomenology can be exploited in pre- and post-processing operations to substantially simplify and improve the final algorithm.

9.1 Structure of the Algorithm

Chapter 3 showed that estimation by a physics-based direct inversion of the data would be difficult because of the highly complex and nonlinear dependence of radiometric observations on atmospheric parameters and

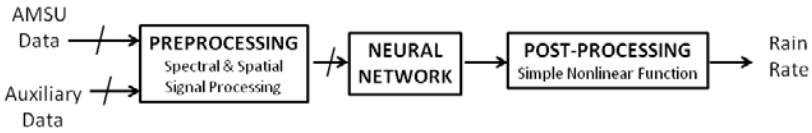


Figure 9.1 Basic structure of the algorithm.

properties of any existing hydrometeors (water particles of various densities). Therefore, a statistics-based method has been employed.

The basic structure of the algorithm is shown in Figure 9.1. In Chapter 7, it was noted that preprocessing is usually necessary to transform data into a useful form. The preprocessing for this algorithm includes a variety of spectral and spatial signal processing that transform the data into a form that characterize the most important degrees of freedom related to precipitation rate such as atmospheric temperature profile, water vapor profile, cloud-top altitude, particle size distribution, and vertical updraft velocity. The neural net is trained to learn the nonlinear dependencies of precipitation rate on these variables. The dependence of precipitation rate on these variables should be monotonic, so the neural net does not need to be complicated. A feedforward neural net with one or more hidden layers of tangent sigmoid nodes (with transfer function $f(x) = \tanh x$) and one linear output node is appropriate (Figure 5.4) [2]. Also, post-processing is necessary because precipitation varies over logarithmic scales. Training a neural network to estimate rain rate directly may result in infrequent heavy rain dominating the training at the expense of more common lighter rain rates.

The most recent version of the precipitation retrieval algorithm is described in Figures 9.2 and 9.3. Different stages of the development of the algorithm are presented in [1, 3–6].

9.1.1 Physical Basis of Preprocessing

AMSU provides a wealth of information on atmospheric state. AMSU was first launched in May 1998 aboard the NOAA-15 satellite, and since then several identical or similar instruments have been launched aboard NOAA-16, NOAA-17, Aqua (National Aeronautics and Space Administration), NOAA-18, NOAA-19, and METOP-A (European Space Agency). A similar instrument, ATMS (Advanced Technology Microwave Sounder), will begin replacing AMSU sometime after 2011 and is expected to yield still better

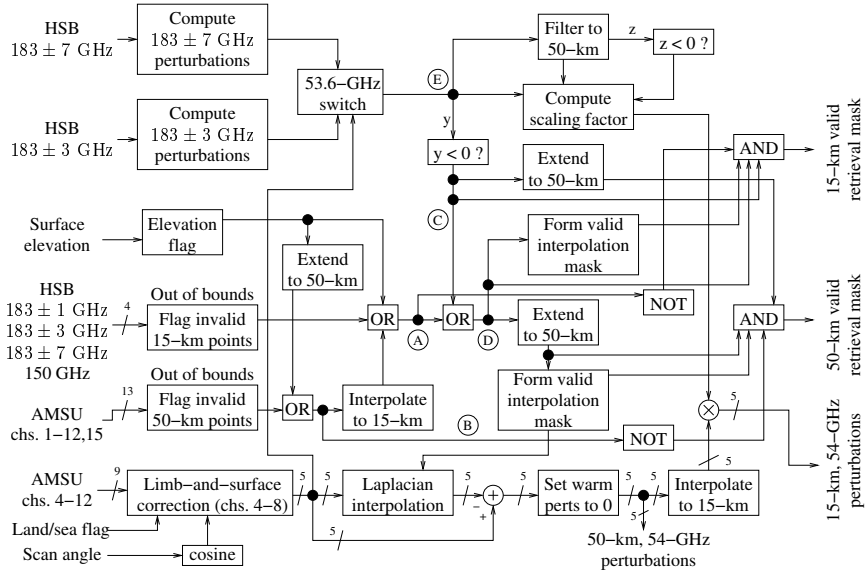


Figure 9.2 Block diagram of the algorithm, part 1. © 2003 IEEE [1].

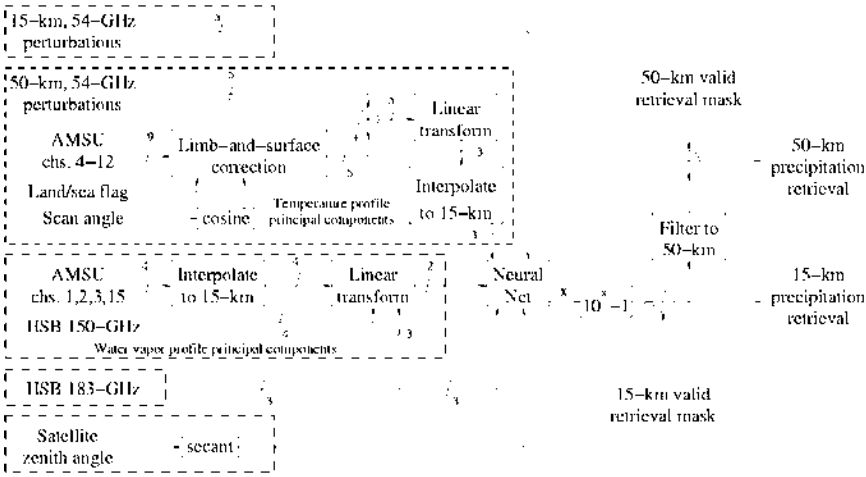


Figure 9.3 Block diagram of the algorithm, part 2. © 2003 IEEE [1].

precipitation retrievals. AMSU includes transparent channels (also called *window channels*) near 23.8, 31.4, 89.0, and 150 GHz that traditionally have

been included aboard instruments used to estimate precipitation. Window channels are useful because of their sensitivity to water vapor absorption over ocean and to hydrometeor scattering [7]. However, AMSU also includes several channels near a series of oxygen resonances lines near 54 GHz and several channels near the 183.31-GHz water vapor resonance line. As the frequency at which a radiometer takes measurements approaches a resonance line, the measurements become less sensitive to surface features (Figure 2.2). Channels at such frequencies are therefore called *opaque channels*. The altitude of peak sensitivity of a channel can be adjusted by varying the channel frequency. With several channels around 54 GHz and 183.31 GHz, one can glean information about temperature and water vapor profile, both of which play important roles in the development of precipitation particles and precipitation rate. A list of AMSU channels with frequencies and bandwidths can be found in [1].

In addition, the sensitivity of each channel to varying altitudes allows one to glean information about the three-dimensional structure of a precipitation cloud and vertical updraft velocity, and the sensitivity of each frequency band to different ranges of particle sizes allows one to glean information about particle size distribution. These features of opaque channels have been demonstrated by Leslie and Staelin with colocated and simultaneous observations from the opaque 54-, 118-, 183-, and 425-GHz bands on the NPOESS Aircraft Sounder Testbed-Microwave (NAST-M) aircraft-based instrument [8].

Although the measurements collected by AMSU contain very valuable information concerning precipitation, the physics that relate atmospheric state to the measurements is highly nonlinear and very complicated. Extracting the information is likely to require nontrivial preprocessing methods. The following section describes the structure of the algorithm and the preprocessing chosen for this algorithm.

In addition to extracting information about atmospheric state, it is also necessary to fuse data of multiple resolutions to make the best use of AMSU data. AMSU provides data from the 23.8-, 31.4-, and 89-GHz channels and the 54-GHz oxygen channels at 50-km resolution and data from the 150-GHz channel and the 183.31-GHz water vapor channels at 15-km resolution. Data at 15-km resolution shows more fine morphological features than data at 50-km resolution, and the nonlinear relationship between atmospheric state and satellite measurements can cause precipitation to go undetected at 50-km resolution. Multiresolution data fusion can allow information from 50-km data to be available for use at 15-km resolution.

9.1.2 Physical Basis of Post-Processing

The statistical nature of precipitation must be adequately addressed during the training of the neural net. In addition to minimizing the RMS error between the estimated rain rate and the ground truth, one should also make sure that the estimator provides estimates that closely reflect the type of rain seen by the radiometer. Precipitation rates can range over logarithmic scales. Drizzle can have precipitation rates as low as 0.25 mm/h while tropical downpours can have rates as high as 200 mm/h [9]. Precipitation rates can have a lognormal distribution with a standard deviation that is at least twice the mean [10]. In order to prevent very high precipitation rates that are rare from dominating the training at the expense of lower rates that are more common, the neural network is trained to estimate a quantity close to the base-10 logarithm of the precipitation rate.

9.2 Signal Processing Components

This section describes in detail the signal processing components that process AMSU data into forms that characterize information about the degrees of freedom most relevant to precipitation rate. An effort has been made to make the inputs to the neural net as insensitive to surface variations as possible so that the algorithm can be applied over land and sea.

9.2.1 Limb-and-Surface Corrections

AMSU measurements show significant scan-angle-dependent variations. AMSU observes at angles up to 49° away from nadir. For angles further away from nadir, electromagnetic energy originating from a given altitude and atmospheric state travels longer paths before reaching the radiometer and therefore is subject to more absorption and scattering. This results in scan-angle dependent brightness temperature images as shown in Figure 9.4(a). A limb-and-surface correction method for AMSU-A channels 4–8 brightness temperatures is needed in order to make precipitation-induced perturbations more apparent and for extracting information about atmospheric conditions. AMSU-A channels 4 and 5 will also be corrected for surface variations since they are sensitive to the surface. For these two channels, brightness temperatures for pixels over ocean will be corrected to what might be observed for the same atmospheric conditions over land. AMSU-A channels 9–14 brightness temperatures are not corrected because they are not significantly perturbed by clouds and therefore are not used for anything besides limb-correction.

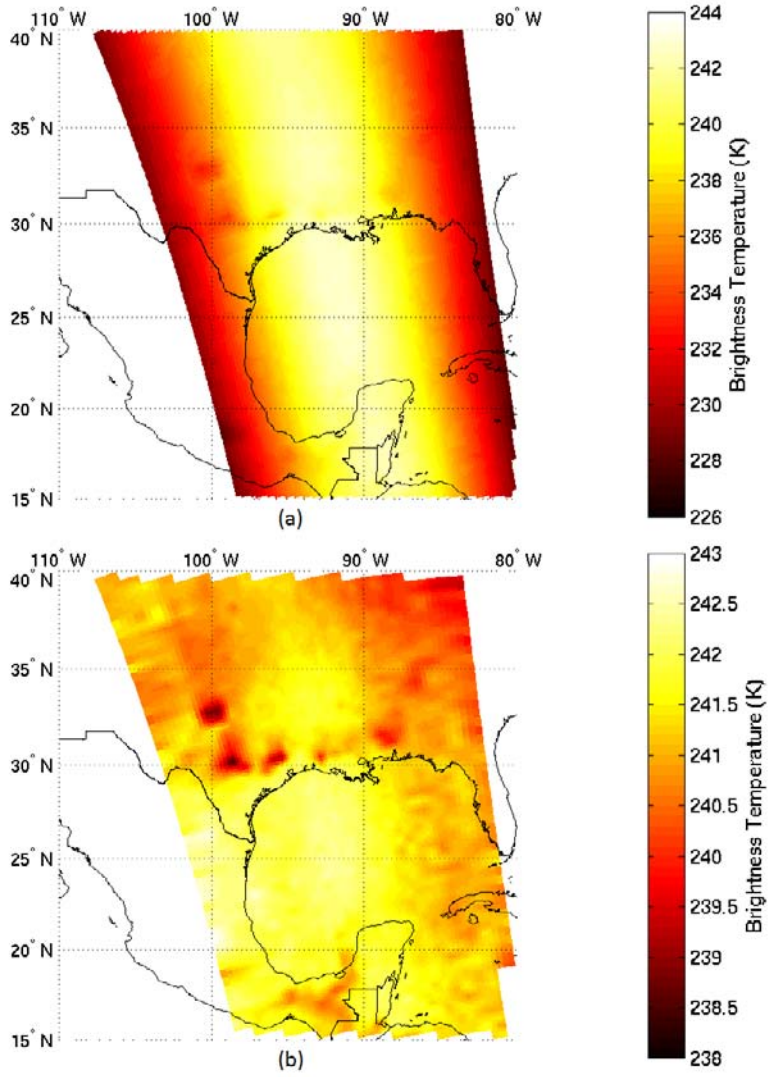


Figure 9.4 NOAA-15 AMSU-A 54.4-GHz brightness temperatures for a northbound track on September 13, 2000. (a) Uncorrected, and (b) limb-and-surface corrected. © 2004 MIT [12].

Limb-and-surface correction was done by training a neural net of the type shown in Figure 5.4 to estimate nadir-viewing brightness temperatures.

Table 9.1
Data Used in Limb-and-Surface Correction of AMSU-A Channels

AMSU-A Channel	Inputs used for limb-and-surface correction
4	AMSU-A channels 4-12, land/sea flag, $\cos\phi$
5	AMSU-A channels 5-12, land/sea flag, $\cos\phi$
6	AMSU-A channels 6-12, $\cos\phi$
7	AMSU-A channels 6-12, $\cos\phi$
8	AMSU-A channels 6-12, $\cos\phi$

For each pixel, the neural net uses brightness temperatures from several channels at that pixel to estimate the brightness temperature seen at the pixel closest to nadir at a nearly identical latitude and at nearly the same time. It is assumed that the temperature field does not vary significantly over one scan. Limb-and-surface correction was done for AMSU-A channels 4 to 8. The data used to correct each of those channels are listed in Table 9.1 (ϕ is the scan angle). No attempt has been made to correct for the scan-angle-dependent asymmetry in the brightness temperatures. These neural nets were trained using data between 55° N and 55° S from seven orbits spaced over one year. Channels 4 and 5 are surface sensitive, so they were trained to estimate brightness temperatures that would be seen over land.

Figure 9.4 shows a sample of (a) uncorrected and (b) limb-and-surface-corrected 54.4-GHz brightness temperatures. The shapes of precipitation systems over Texas and the Mexico-Guatemala border are more apparent after the limb correction. In Figure 9.4(a) the difference between brightness temperatures at nadir and the swath edge is as high as 18K. In Figure 9.4(b), the angle-dependent variation is less than 3K.

9.2.2 Precipitation Detection

The 15-km resolution precipitation-rate retrieval algorithm, summarized in Figures 9.2 and 9.3, begins with identification of potentially precipitating pixels. Neural networks are trained over only potentially precipitating pixels. This saves time during training since the neural network does not have to learn variations of data for nonprecipitating pixels and the training algorithm can work with a smaller training set. Additionally, the neural network topology can be simplified. Channels used for this purpose should be sensitive to precipitation but should not exhibit large angle-dependent variations and surface variations. Unprocessed brightness temperatures from AMSU-A

channels 4–14 are not used directly because they exhibit excessive angle-dependent variations or have weighting functions that peak far above the range of altitudes in which most precipitation exists. AMSU-A channels 1, 2, 3, and 15, and the AMSU-B 89.0-GHz and 150-GHz channels were not used because of their sensitivity to surface variations. The AMSU-B channels ordered by opacity are as follows: 89-GHz (most transparent), 150-GHz, 183 ± 7-GHz, 183 ± 3-GHz, and 183 ± 1-GHz (most opaque). The 183 ± 7-GHz channel is the least opaque channel that is sensitive to precipitation but does not show excessive surface variations. Figure 9.5 shows brightness temperatures at 150 GHz and 183 ± 7 GHz over the southern United States and Mexico. This figure shows the varying degrees of sensitivity to the surface of the 150-GHz channel. In this channel, the contrast between land and sea is greatest around the Gulf of California. On the other hand, the 183 ± 7-GHz channel does not show any surface variations while still being sensitive to most of the precipitation seen in the 150-GHz channel.

The 183 ± 7-GHz channel is reasonably good for detecting precipitation also because the angle-dependent variation of precipitation-free brightness temperatures is small when compared to the variation due to precipitation. All 15-km pixels with brightness temperatures at 183 ± 7 GHz that are below a threshold T_7 are flagged as potentially precipitating, where

$$T_7 = 0.667(T_{53.6} - 248) + 252 + 6\cos\theta \tag{9.1}$$

and where θ is the satellite zenith angle and $T_{53.6}$ is the spatially filtered limb-corrected 53.6-GHz brightness temperature obtained by selecting the warmest brightness temperature within a 7×7 array of AMSU-B pixels. It has been seen that the threshold T_7 can vary with atmospheric temperature. This threshold was determined empirically. However, the 183 ± 7-GHz channel can become sensitive to surface variations in very cold, dry atmospheric conditions. Figure 9.6 shows 183 ± 7-GHz brightness temperatures in such conditions. The edges of the Great Lakes and Hudson Bay are evident in this image. When $T_{53.6}$ is less than 248 K, the 183 ± 3-GHz brightness temperature is compared to a threshold T_3.

$$T_3 = 242.5 + 5\cos\theta \tag{9.2}$$

The thresholds T_7 and T_3 are slightly colder than a saturated atmosphere would be, implying the presence of a microwave-absorbing or scattering cloud.

It is possible for even the 183 ± 3-GHz and the 183 ± 1-GHz channels to be sensitive to surface variations. Figure 9.7 shows 183 ± 1-GHz brightness temperatures images over the Laptev Sea (Russia) taken $\sim$23.6 hours apart.

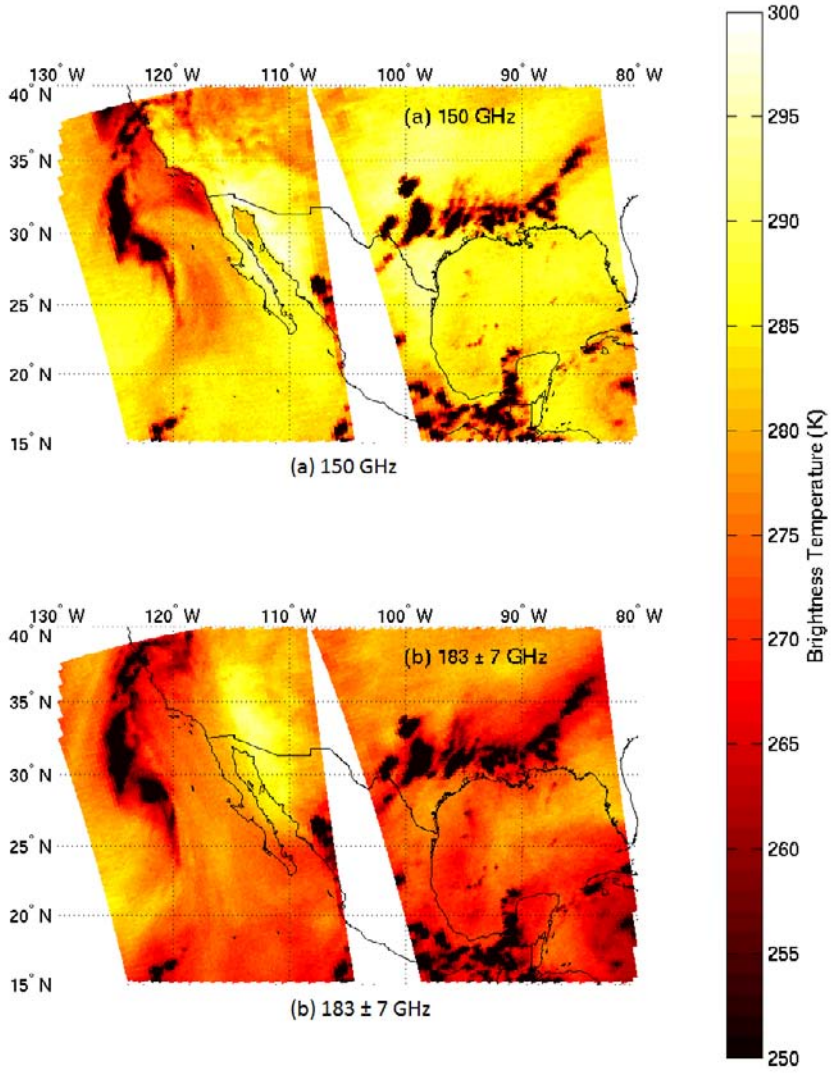


Figure 9.5 NOAA-15 AMSU-B brightness temperatures for northbound tracks on September 13, 2000, at (a) 150 GHz, and (b) 183 ± 7 GHz. © 2004 MIT [12].

Inside the black rectangles is a feature that does not move over time. If $T_{53.6}$ is less than 242 K, then the pixel is assumed not to be precipitating.

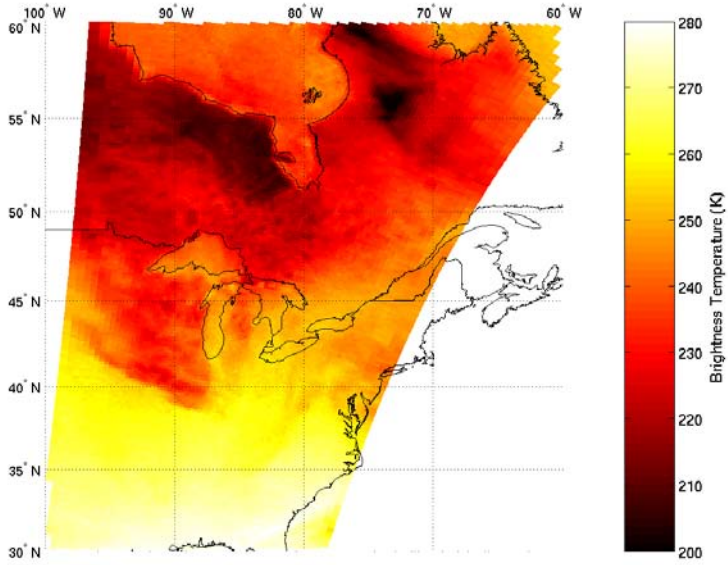


Figure 9.6 NOAA-15 AMSU-B 183 ± 7-GHz brightness temperatures, January 20, 2000, 1339 to 1349 UTC. © 2004 MIT [12].

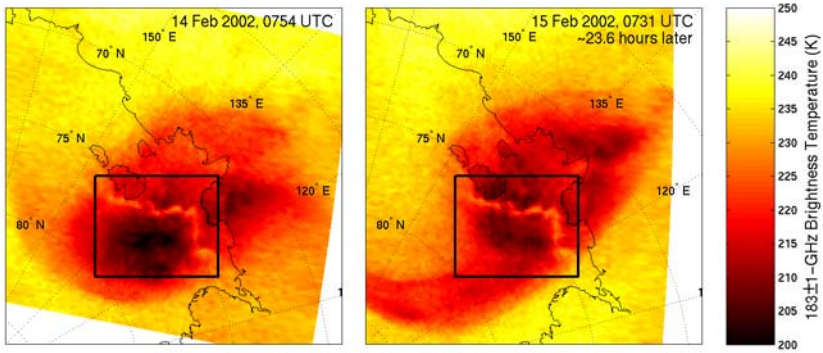


Figure 9.7 NOAA-15 AMSU-B 183 ± 1-GHz brightness temperatures over a dry hole on February 14, 2002, 0754 UTC (left), and February 15, 2002, 0731 UTC (right). © 2004 MIT [12].

Pixels with invalid or missing data from AMSU-A channels 1–12, and 15, and AMSU-B channels 2 to 5 are treated as potentially precipitating even though no retrieval will be done over them. Pixels where the altitude is higher than a latitude-dependent threshold $A(\Theta_{lat})$ are also treated as potentially precipitating (Θ_{lat} is in units of degrees and is defined to be positive for the northern hemisphere).

$$A(\Theta_{lat}) = \begin{cases} 2{,}000\text{m} & \text{for } \|\Theta_{lat}\| < 60 \\ 1{,}500\text{m} & \text{for } 60 \leq \Theta_{lat} < 70 \\ 500\text{m} & \text{otherwise} \end{cases} \tag{9.3}$$

These decisions minimize the likelihood that corrupt data will corrupt the computation of cloud-cleared brightness temperatures over pixels without corrupt data. Cloud clearing is described in Section 9.2.3.

9.2.3 Cloud Clearing by Regional Laplacian Interpolation

When estimating precipitation rate in a precipitating region, it is helpful to obtain information about the atmospheric conditions in the surrounding cloud-free area that give rise to precipitation. For example, warmer air can hold more water vapor and can more easily produce heavy rain. The estimate at a pixel is likely to depend not only on the information at that pixel but also on that in some surrounding pixels. For this reason, spatial processing of the AMSU measurements is useful. Section 9.2.4 describes the use of temperature-profile principal components in the algorithm.

Regional Laplacian interpolation is used to estimate measurements in precipitating regions as if they were free of precipitation. After the precipitation mask has been computed, it is used to filter the brightness temperature data so that the precipitation-induced signatures do not appear. It has been observed that in the absence of precipitating clouds and surface features, local brightness temperature images approximately satisfy Laplace's equation:

$$\nabla^2 \Phi = 0 \tag{9.4}$$

where Φ is a scalar field (in this case brightness temperatures). This approximation appears to be reasonably valid for our purposes over distances of several hundred kilometers. Precipitation-induced signatures from brightness temperature data are filtered out by forcing data within potentially precipitating regions to satisfy Laplace's equation given boundary conditions determined by the precipitation mask. An example of this is shown in Figure 9.8. This technique was used in [11].

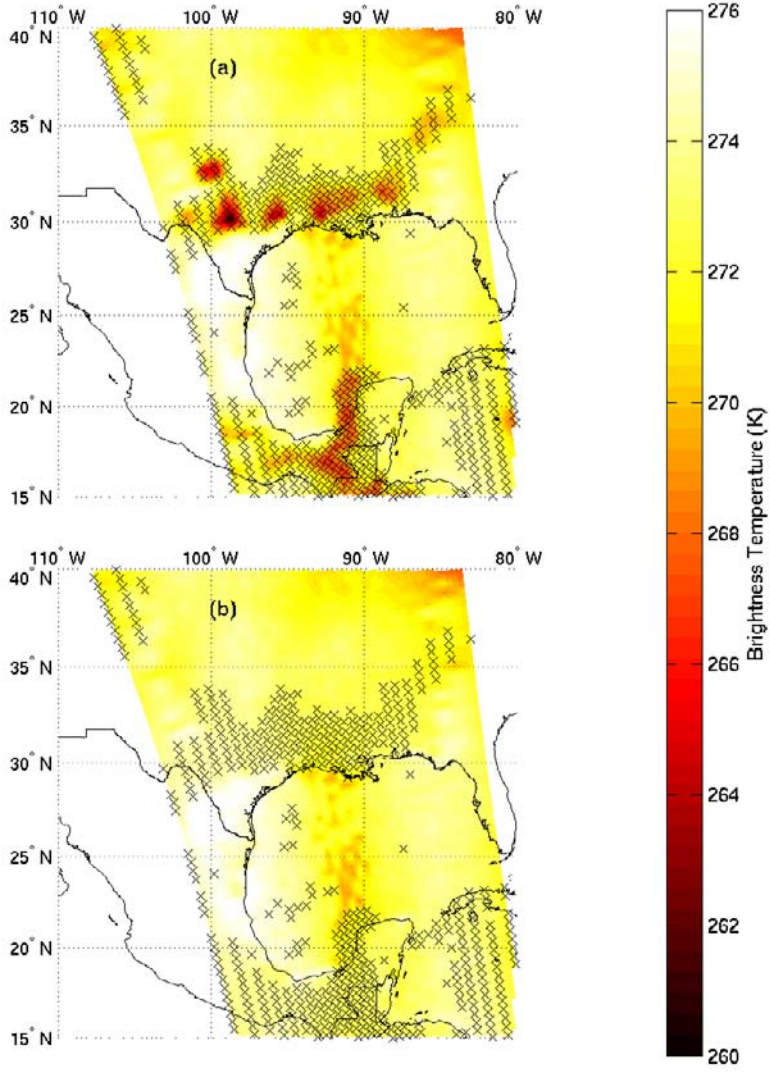


Figure 9.8 NOAA-15 AMSU-A 52.8-GHz brightness temperatures for a northbound track on September 13, 2000: (a) before cloud-clearing, and (b) after cloud-clearing using Laplacian interpolation. AMSU-A pixels within potentially precipitating regions are marked with x's. © 2004 MIT [12].

9.2.3.1 The Morphology of Precipitation

For each region, before Laplacian interpolation takes place, special processing may be required depending on the morphology and location of the region.

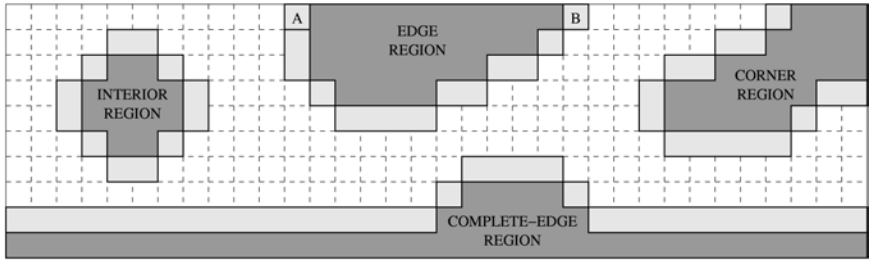


Figure 9.9 Types of regions encountered by the cloud-clearing method. Boundary pixels are shaded in light gray. © 2004 MIT [12].

Pixels flagged as potentially precipitating are divided into disjoint, non-adjacent 4-connected regions.[1] The types of regions that this cloud clearing procedure could encounter can be classified into these four types:

1. *Interior regions*, regions that do not include points at the edge nor the corners of the image being processed;
2. *Edge regions*, regions that include points at the edge but not the corner of the image being processed;
3. *Vertex regions* (or *corner regions*), regions that include points at any of the corners of the image but not an entire edge;
4. *Complete-edge regions*, regions that include all of the pixels along one or more edges.

These four types are illustrated in Figure 9.9. The boundary pixels of each region are found by dilating by one pixel vertically and horizontally. Boundary pixels are not part of the flagged regions. Interior regions do not require any special processing before Laplacian interpolation because their boundary points lie within the image.

Edge regions require extra steps in order to form a closed boundary over which Laplacian interpolation can be performed. The brightness temperatures at pixels along the edge are linearly interpolated based on the boundary points that are at the edge. Then a new set of boundary points is formed that includes the first set of boundary pixels and the portions of the edges in the edge region, and Laplacian interpolation is performed over the edge region without the pixels on the edge using this set of boundary pixels. For example, for the edge

1. A *4-connected region* is a set of pixels with an ordering of all pixels (possibly with some repetition) such that each pair of consecutive pixels are adjacent vertically or horizontally. For an *8-connected region*, consecutive pixels can be vertically, horizontally, or diagonally adjacent.

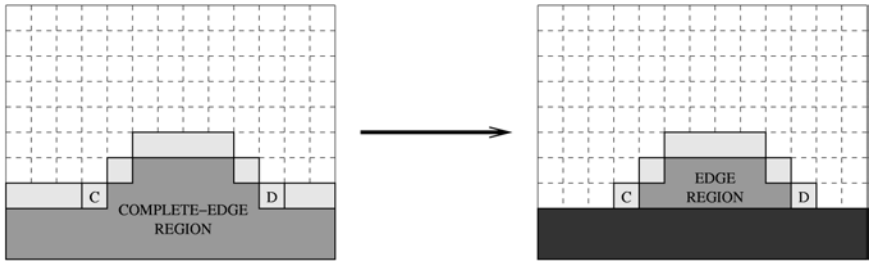


Figure 9.10 Salvaging pixels in a complete-edge region. Eliminating the bottom two rows of the image on the left leaves a portion that can be treated as an edge region. © 2004 MIT [12].

region in Figure 9.9 the cloud-cleared brightness temperatures for the pixels between pixels A and B are assumed to be the linearly interpolated brightness temperature based on those at A and B. Then, Laplacian interpolation is performed using the boundary pixels of the edge region along with the pixels between A and B.

In this work, Laplacian interpolation was not applied to corner regions. The convexity of the corner regions at corner points prevents interpolation of cloud-cleared brightness temperatures along the portions of the edges that include corner points. However, cloud-cleared brightness temperatures may be computed for some of the pixels in a corner region. A method for processing such regions is described in [12] and may be implemented in the future. The amount of precipitation that falls in corner regions is very small for data sets that are sufficiently large. AMSU-A/B data sets typically cover one orbit ($\sim$ 770 AMSU-A scans). Aqua AMSU/HSB granules cover 45 AMSU-A scans, but consecutive granules can be concatenated to reduce the effect of ignoring precipitation in corner regions.

Complete-edge regions can occur when entire scans of data are invalid or missing. Laplacian interpolation can sometimes be used over portions of complete-edge regions. Contiguous sets of rows or contiguous sets of columns that include an edge are excluded from the interpolation process. In Figure 9.10, the complete edge region becomes an edge region, and linear interpolation can be done on the pixels between pixels C and D before Laplacian interpolation. Some complete-edge regions will become corner regions and therefore will not yield any useful information because the current version of the algorithm ignores corner regions.

9.2.3.2 Channel-Specific Interpolation Masks

For AMSU-A channel 4 (52.8 GHz) the mask used for Laplacian interpolation is obtained using the precipitation detection method. However, for AMSU-A channels 5 to 8, the mask used is slightly different. The points selected for Laplacian interpolation do not include those for which the magnitude of the perturbation in channel 4 is less than 1 K. At such pixels, the precipitation signature is assumed to be weak enough that it would not significantly perturb channels 5 to 8.

9.2.4 Temperature-Profile and Water-Vapor-Profile Principal Components

One important determinant of precipitation is the temperature profile. Warmer atmospheres can hold more water vapor and result in higher vertical updraft velocities. Therefore, inputs to the neural net in Figure 9.1 should include some that have information about temperature profile. For each of AMSU-A channels 4 to 8, the brightness temperatures were corrected for limb and surface effects and then processed to eliminate precipitation signatures with the methods described in Sections 9.2.1 to 9.2.3. The corrected brightness temperatures from all five of these channels could have been inputs to the neural net in Figure 9.1, but it was determined that a more compact representation of these channels was sufficient. The PC transform was applied to these five channels, and the first three principal components were found to be sufficient for characterizing the temperature profile. Adding the fourth and fifth principal components did not significantly improve the training of the neural net.

Another important determinant of precipitation is the water vapor profile. Higher concentrations of water vapor can result in higher precipitation rates. The water-vapor principal components are computed using AMSU-A channels 1, 2, 3, and 15, and the AMSU-B 150-, 183 ± 7-, 183 ± 3-, and 183 ± 1-GHz channels. Some of these channels are sensitive to surface variations. Therefore it is necessary to project the vector of these observations onto a subspace that is not significantly sensitive to surface variations. Constrained PCA [13], was used to compute the water-vapor principal components. A set of pixels without precipitation and with different types of surfaces was selected to compute surface-sensitive eigenvectors using PCA. The surface-sensitive eigenvectors were determined by visual inspection of the pre-constraint principal components for correlation with surface features (e.g., land/sea boundaries). Then, a set of data that also included precipitation was selected. The observations over this set were projected onto a linear subspace that was orthogonal to the subspace spanned by the surface-sensitive

eigenvectors. Then, PC analysis was performed on the resulting data set in order to determine the water-vapor principal components. It was found that two water-vapor principal components were adequate for characterizing the eight channels.

9.2.5 Image Sharpening

Because of the nonlinear relationship between precipitation and atmospheric conditions, data collected at 50-km resolution can miss or weakly detect smaller rain cells that contribute a significant amount of rainfall. Therefore, 15-km retrievals are useful to detect and characterize smaller and more intense rain cells.

The 54-GHz perturbations are calculated at 50-km resolution, and the information contained in such images might not be suitable for the production of 15-km retrievals. We therefore use image sharpening on the native 54-GHz perturbation data at 50-km resolution, together with 15-km observations available at frequencies near 183 GHz, to produce a 54-GHz perturbation at an "effective" 15-km resolution.

Within regions flagged as potentially precipitating, strong precipitation is generally characterized by cold cloud-induced perturbations of the AMSU-A tropospheric temperature sounding channels in the range 52.5–55.6 GHz. Examples of 183 ± 7-GHz data and the corresponding 50-km cold perturbations at 52.8 GHz are illustrated in Figure 9.11(a, c). Physical considerations and aircraft data show that convective cells near 54 GHz typically appear slightly off-center and less extended relative to the 183-GHz image [14, 15]. The small interpolation errors in converting 54-GHz perturbations to 15-km resolution contribute to the total errors and discrepancies discussed in Section 9.4. These 50-km resolution 52.8-GHz perturbations $\Delta T_{50;52.8}$ are then used to infer the perturbations $\Delta T_{15;52.8}$ (Figure 9.11(d)) that might have been observed at 52.8 GHz with 15-km resolution had those perturbations been distributed spatially in the same way as the cold perturbations observed at either 183 ± 7 or 183 ± 3 GHz, the choice between these two channels being the same as that for precipitation detection which is described in Section 9.2.2. This requires the bilinearly interpolated 50-km AMSU data to be resampled at the HSB beam positions. The 50-km perturbations in the 54-GHz band $\Delta T_{50;54}$ are sharpened to 15-km resolution $\Delta T_{15;54}$ by computing the value of $\Delta T_{15;54}$ that will make the ratio of the 15-km perturbation to the 50-km perturbation for the 54-GHz band equal to corresponding ratio for the 183-GHz band.

$$\frac{\Delta T_{15;54}}{\Delta T_{50;54}} = \frac{\Delta T_{15;183}}{\Delta T_{50;183}} \tag{9.5}$$

The perturbation near 183 GHz is defined to be the difference between the observed radiance and the appropriate threshold given by (9.1) or (9.2). The perturbation $\Delta T_{50;54}$ near 54 GHz is defined to be the difference between the observed radiance and the Laplacian-interpolated radiance based on those pixels surrounding the flagged region [6]. Any warm perturbations in the images of $\Delta T_{15;183}$ and $\Delta T_{50;54}$ are set to zero. Then, these inferred 15-km perturbations can be computed for five AMSU-A channels using the following formula

$$\Delta T_{15;54} = \frac{\Delta T_{15;183}}{\Delta T_{50;183}} \Delta T_{50;54} \tag{9.6}$$

To prevent this computation from becoming unstable, the hyperbolic tangent function is applied to the ratio of the 183-GHz perturbations.

$$\Delta T_{15;54} = 20 \tanh \left(\frac{\Delta T_{15;183}}{20 \Delta T_{50;183}} \right) \Delta T_{50;54} \tag{9.7}$$

Given the filter used to filter $\Delta T_{15;183}$ to 50-km resolution, the ratio $\Delta T_{15;183}/\Delta T_{50;183}$ should not exceed 7.

Figure 9.11 shows an example of image sharpening. Figure 9.11(d) shows $\Delta T_{15;52.8}$ 50-km 52.8-GHz perturbations sharpened to 15-km resolution for a frontal system. Visually, this image appears to be appropriate for use in estimating 15-km precipitation rates.

9.3 Development of the Algorithm

The current AMSU/HSB precipitation retrieval algorithm is based on NOAA-15 AMSU-A/B comparisons with the National Weather Service's Next Generation Weather Radar (NEXRAD) over the eastern United States during 38 orbits that exhibited significant precipitation and were distributed throughout the year. These orbits are listed in Table 9.2. The primary precipitation-rate retrieval products of AMSU/HSB are 15- and 50-km-resolution contiguous retrievals over the viewing positions of HSB and AMSU, respectively, within 43° of nadir. The two outermost 50-km and six outermost 15-km viewing positions on each side of the swath are omitted due to the potentially excessive limb effects at such angles. After the algorithm architectures for these two retrieval methods are presented below, the derivation of the numerical coefficients characterizing the neural network is described.

The neural net in Figure 9.3 produces 15-km precipitation-rate estimates using the following inputs:

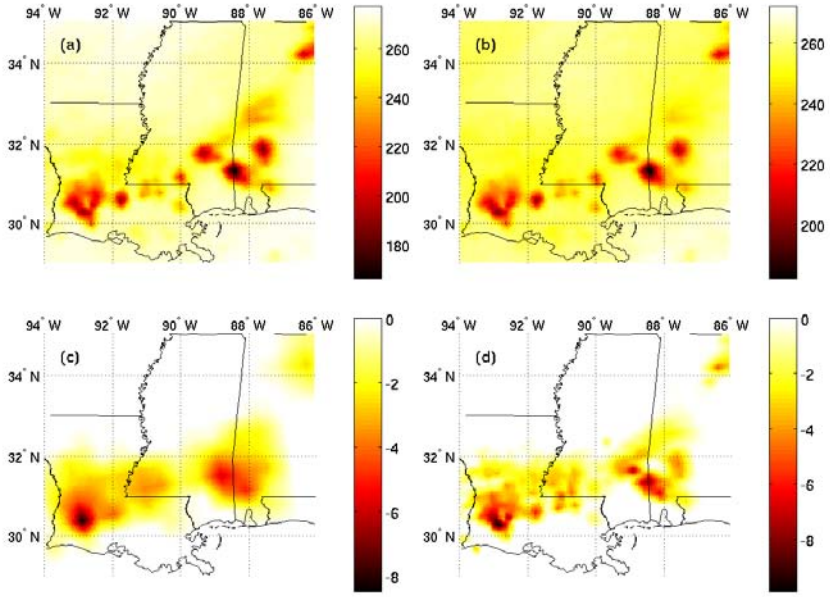


Figure 9.11 Frontal system on September 13, 2000, 0130 UTC. (a) Brightness temperatures (K) near 183 ± 7-GHz. (b) Brightness temperatures (K) near 183 ± 3-GHz. (c) Brightness temperature perturbations (K) near 52.8 GHz. (d) Inferred 15-km-resolution brightness temperature perturbations (K) near 52.8 GHz. © 2003 IEEE [1].

- Inferred 15-km perturbations at 52.8, 53.6, 54.4, 54.9, and 55.5 GHz (corresponding to AMSU-A channels 4, 5, 6, 7, and 8, respectively);
- 183 ± 1-, 183 ± 3-, and 183 ± 7-GHz 15-km HSB data;
- Three principal components characterizing the original five corrected 50-km AMSU-A temperature radiances;
- Surface-insensitive principal components that characterize the window channels at 23.8, 31.4, 50, and 89 GHz, plus the four HSB channels;
- $\sec \theta$.

The relative insensitivity of these inputs to surface emissivity is important to the success of this technique over land, ice, and snow.

This network was trained to minimize the RMS value of the difference between the logarithms of the (AMSU + 1-mm/h) and (NEXRAD + 1-mm/h) retrievals; use of logarithms reduced the emphasis on the heaviest rain rates,

Table 9.2
List of Rainy Orbits Used for Training, Validation, and Testing

16 Oct 1999, 0030 UTC	30 Apr 2000, 1430 UTC
31 Oct 1999, 0130 UTC	14 May 2000, 0030 UTC
2 Nov 1999, 0045 UTC	19 May 2000, 0015 UTC
4 Dec 1999, 1445 UTC	19 May 2000, 0145 UTC
12 Dec 1999, 0100 UTC	20 May 2000, 0130 UTC
28 Jan 2000, 0200 UTC	25 May 2000, 0115 UTC
31 Jan 2000, 0045 UTC	10 Jun 2000, 0200 UTC
14 Feb 2000, 0045 UTC	16 Jun 2000, 0130 UTC
27 Feb 2000, 0045 UTC	30 Jun 2000, 0115 UTC
11 Mar 2000, 0100 UTC	4 Jul 2000, 0115 UTC
17 Mar 2000, 0015 UTC	15 Jul 2000, 0030 UTC
17 Mar 2000, 0200 UTC	1 Aug 2000, 0045 UTC
19 Mar 2000, 0115 UTC	8 Aug 2000, 0145 UTC
2 Apr 2000, 0100 UTC	18 Aug 2000, 0115 UTC
4 Apr 2000, 0015 UTC	23 Aug 2000, 1315 UTC
8 Apr 2000, 0030 UTC	23 Sep 2000, 1315 UTC
12 Apr 2000, 0045 UTC	5 Oct 2000, 0130 UTC
12 Apr 2000, 0215 UTC	6 Oct 2000, 0100 UTC
20 Apr 2000, 0100 UTC	14 Oct 2000, 0130 UTC

which were roughly three orders of magnitude greater than the lightest rates. Adding 1 mm/h reduced the emphasis on the lightest rain rates that are more noise-dominated. These intuitive choices clearly impact the retrieval error distribution, and therefore further study should enable additional algorithm improvements. Retrievals with training optimized for low rain rates did not markedly improve that regime, however.

NEXRAD precipitation retrievals with 2-km resolution were smoothed to approximate Gaussian spatial averages that were centered on and approximated the view-angle-distorted 15- or 50-km antenna beam patterns. The accuracy of NEXRAD precipitation observations is known to vary with distance from the nearest NEXRAD radar site, so only points beyond 30 km but within 110 km of each NEXRAD radar site were included in the data used to train and test the neural nets.

Eighty different networks were trained using the Levenberg-Marquardt algorithm, each with different numbers of nodes and water-vapor principal components. (Before experiments using water-vapor principal components

were done, it had been determined that the leading three temperature-profile principal components were sufficient for characterizing the atmospheric temperature profile.) A network with nearly the best performance over the testing data set was chosen; it used two surface-blind water-vapor principal components, and only slightly better performance was achieved with five water-vapor principal components with increased surface sensitivity. The final network had one hidden layer with five nodes that used the hyperbolic tangent activation function. These neural networks are similar to those described in Figure 5.4. The resulting 15-km-resolution precipitation retrievals are then smoothed to yield 50-km retrievals.

The 15-km retrieval neural network was trained using precipitation data from the 38 orbits listed in Table 9.2. During this period the radio interference to AMSU-B was negligible relative to other sources of retrieval error. Each 15-km pixel flagged as potentially precipitating using 183 ± 7- or 183 ± 3-GHz radiances (see Figures 9.2) was used either for training, validation, or testing of the neural network. For these 38 orbits over the United States, 15,160 15-km pixels were flagged and considered suitable for training, validation, and testing. Half were used for training, and one quarter were used for each of validation and testing, where the validation pixels were used to determine when the training of the neural network should cease. Based on the final AMSU and NEXRAD 15-km retrievals, approximately 14% and 38%, respectively, of the flagged 15-km pixels appear to have been precipitating less than 0.1 mm/h for the test set.

9.4 Retrieval Performance Evaluation

This section presents three forms of evaluation for this initial precipitation-rate retrieval algorithm:

- Representative qualitative comparisons of AMSU and NEXRAD precipitation-rate images;
- Quantitative comparisons of AMSU and NEXRAD retrievals stratified by NEXRAD rain rate;
- Representative precipitation images at more extreme latitudes beyond the NEXRAD training zone.

9.4.1 Image Comparisons of NEXRAD and AMSU/HSB

Each NEXRAD comparison at 15-km resolution occurred within 8 minutes of a satellite overpass; such coincidence is needed to characterize single-pixel retrievals because convective precipitation evolves rapidly on this spatial

scale. Although comparison with instruments such as TRMM and SSM/I would be useful, their orbits unfortunately overlap those of AMSU within 8 minutes so infrequently (if ever) that comparisons over precipitation will be too rare to be useful until several years of data have been analyzed. This challenge of simultaneity and the sporadic character of rain have restricted most prior instrument comparisons (passive microwave satellites, radar, rain gauges) to dimensions over 100 km and to periods of an hour to a month [17–19]. The uniformity and extent of the NEXRAD network offer a unique degree of simultaneity on 15- and 50-km scales and even the ability to match the Gaussian shape of the AMSU antenna beams.

Although these AMSU/HSB-NEXRAD comparisons are encouraging because they involve single pixels and independent physics and facilities, further extensive analyses are required for real validation. For example, comparisons of precipitation averages and differences over the same time/space units used to validate other precipitation measurement systems (e.g., SSM/I [19], ATOVS, TRMM, rain gauges) will be needed to characterize variances and systematic biases based on precipitation rate, type, location, or season. These biases will include any present in the NEXRAD data used to train the AMSU/HSB algorithm; once characterized, they can be diminished. Any excess variance experienced for rain cells too small to be resolved by AMSU/HSB can also eventually be better characterized, although it is believed to be modest for cells with microwave signatures larger than 10 km. Smaller isolated cells contribute little to total rainfall.

Figure 9.12(a, b) presents 15-km-resolution precipitation retrieval images for September 13, 2000, obtained from NEXRAD and AMSU, respectively. On this occasion, both sensors yield rain rates over 50 mm/h at similar locations and lower rain rates down to 0.5 mm/h over comparable areas. The revealed morphology is thus very similar even though AMSU observes 6 minutes before NEXRAD, and they sense altitudes that may be separated by several kilometers; rain falling at a nominal rate of 10 m/s takes 10 minutes to fall 6 km.

9.4.2 Numerical Comparisons of NEXRAD and AMSU/HSB Retrievals

Figure 9.13 shows the scatter between the 15-km AMSU and NEXRAD rain rate retrievals for the test pixels not used for training or validation. Figure 9.14 shows the scatter between the 50-km AMSU and NEXRAD rain rate retrievals over all points flagged as precipitating. The relative sensitivity of AMSU and NEXRAD to light and heavy rain can be seen from Figure 9.14. In general, these images suggest that AMSU responds less to the highest radar

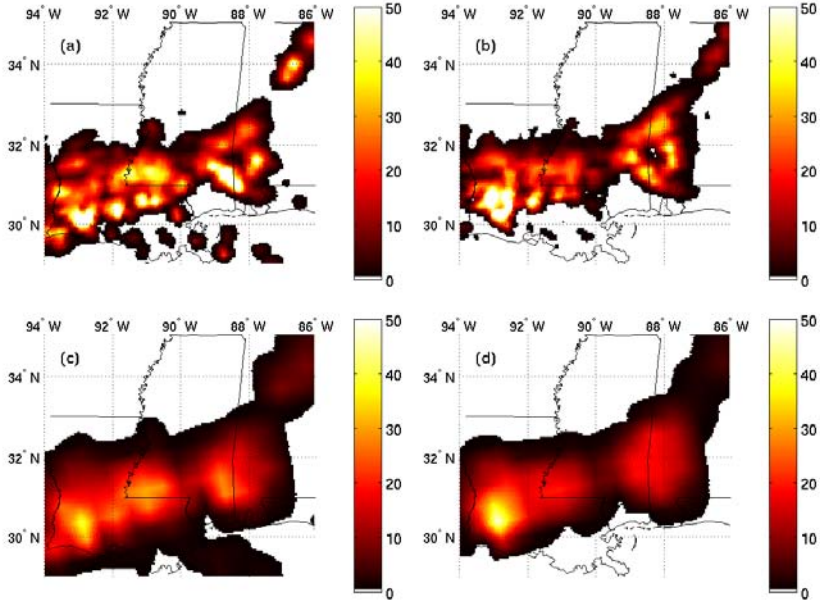


Figure 9.12 Precipitation rates (mm/h) above 0.5 mm/h observed on September 13, 2000, 0130 UTC. (a) 15-km-resolution NEXRAD retrievals, (b) 15-km-resolution AMSU retrievals, (c) 50-km-resolution NEXRAD retrievals, and (d) 50-km-resolution AMSU retrievals. © 2003 IEEE [1].

rain rates, perhaps because AMSU is less sensitive to the bright-band or hail anomalies that affect radar. They also suggest the risk of false rain detections increases for AMSU retrievals below 0.5 mm/h at 50-km resolution, although further study will be required. Greater accuracy at these low rates will require more space-time averaging and careful calibration. The risk of overestimating rain rate also appears to be limited. Only 3.3% of the total AMSU-derived rainfall was in areas where AMSU saw more than 1 mm/h and NEXRAD saw less than 1 mm/h. Only 7.6% of the total NEXRAD-derived rainfall was in areas where NEXRAD saw more than 1 mm/h and AMSU saw less than 1 mm/h. These percentages can be compared to the total percentages of AMSU and NEXRAD rain that fell at rates above 1 mm/h, which are 94 and 97, respectively. It is also interesting to see to what degree each sensor retrieves rain when the other does not, and how much rain each sensor misses. For example, of the 73 NEXRAD 15-km rain rate retrievals in Figure 9.13 above 54 mm/h, none was found by AMSU to be below 3 mm/h, and of the

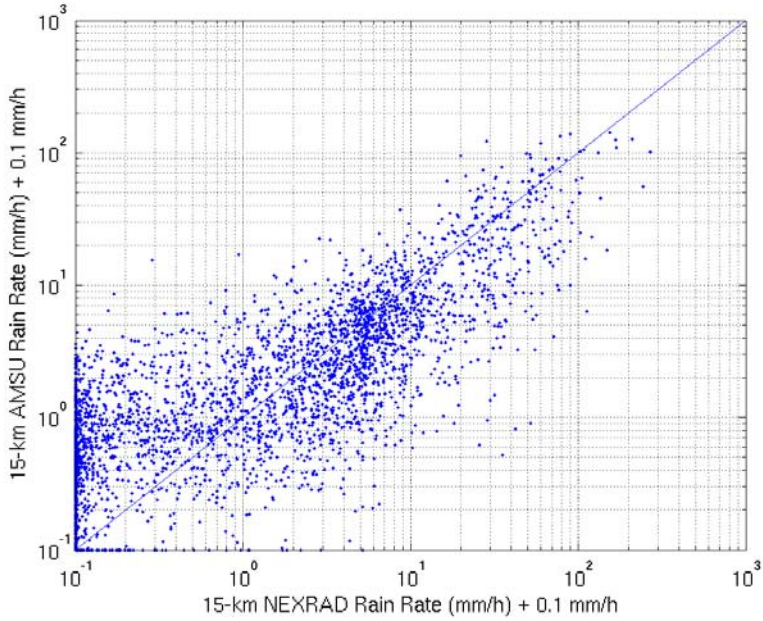


Figure 9.13 Comparison of AMSU and NEXRAD estimates of rain rate at 15-km resolution. © 2003 IEEE [1].

61 AMSU 15-km retrievals above 45 mm/h, none was found by NEXRAD to be below 16 mm/h. Also, of the 69 NEXRAD 50-km rain rate retrievals in Figure 9.14 above 30 mm/h, none was found by AMSU to be below 5 mm/h, and of the 102 AMSU 50-km retrievals above 16 mm/h, none was found by NEXRAD to be below 10 mm/h.

Perhaps the most significant AMSU precipitation performance metric is the RMS difference between the NEXRAD and AMSU rain rate retrievals; these are calculated for AMSU and HSB pixels that are flagged as potentially precipitating and do not have corrupted data. The pixels are grouped by retrieved NEXRAD rain rates in octaves. Only pixels that were from the central 26 AMSU-A scan angles and central 78 AMSU-B scan angles and were from regions between 30 and 110 km of any NEXRAD radar site were included in these evaluations; only the outermost two AMSU-A angles on each side were omitted. These comparisons used all 50-km pixels and only those 15-km pixels not used for training or validation. The results are listed in Table 9.3. The smoothing of the 15-km NEXRAD and AMSU results to nominal 50-km resolution was consistent with an AMSU-A Gaussian

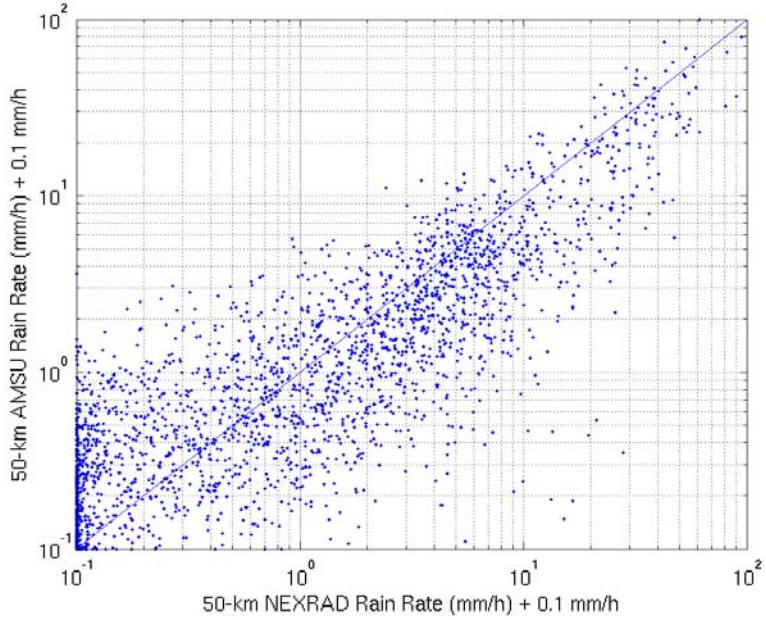


Figure 9.14 Comparison of AMSU and NEXRAD estimates of rain rate at 50-km resolution. © 2003 IEEE [1].

Table 9.3
RMS AMSU/NEXRAD Discrepancies (mm/h) at 15-km and 50-km Resolution for Pixels in the Ranges 30–110 km and 110–230 km of NEXRAD Radar Sites

NEXRAD Range	15-km Resolution		50-km Resolution	
	30–110 km	110–230 km	30–110 km	110–230 km
< 0.5 mm/h	1.0	1.4	0.5	0.5
0.5–1 mm/h	2.0	2.6	0.9	1.1
1–2 mm/h	2.3	2.7	1.1	1.5
2–4 mm/h	2.7	3.9	1.8	2.3
4–8 mm/h	3.5	7.4	3.2	5.2
8–16 mm/h	6.9	8.4	6.6	6.5
16–32 mm/h	19.0	17.2	12.9	14.6
> 32 mm/h	42.9	39.2	22.1	21.7

beamwidth of $3.33°$.

The RMS agreement between these two very different precipitation-rate sensors appears surprisingly good, particularly since a single AMSU neural network is used over all angles, seasons, and latitudes. The 3-GHz radar retrievals respond most strongly to the largest hydrometeors, especially those in and below the bright band near the freezing level, while AMSU interacts with the general population of hydrometeors in the top few kilometers of the precipitation cell, which may lie several kilometers above the freezing level. Much of the agreement between AMSU and NEXRAD rain rate retrievals must therefore result from the statistical consistency of the relations between rain rate and its various electromagnetic signatures. It is difficult to say how much of the observed discrepancy is due to each sensor or to say how well each correlates with precipitation reaching the ground.

This study furthermore provided an opportunity for evaluation of radar data. The RMS discrepancies between AMSU and NEXRAD retrievals were separately calculated over all points at ranges from 110 to 230 km from any radar. For NEXRAD precipitation rates below 16 mm/h, these rms discrepancies were approximately 40% greater than those computed for test points at the 30- to 110-km range. At rain rates greater than 16 mm/h, the accuracies beyond 110 km were more comparable (Table 9.3). The NEXRAD radar beams that are directed at angles farthest from zenith may miss some low-altitude precipitation for regions further than 110 km away from any NEXRAD site due to factors such as curvature of the Earth's surface and surface-based obstructions (mountains, for example). However, these beams are still able to sense heavy precipitation at high altitudes over the same regions. Most points in the eastern United States are more than 110 km from any NEXRAD radar site, and radar is even more sparse in the western half.

9.4.3 Global Retrievals of Rain and Snow

One of the principal Aqua validation activities involves testing and tuning of the precipitation retrievals for climates not well represented in the NEXRAD training data set. Figure 9.15 illustrates precipitation-rate retrievals at points around the globe where radar confirmation data is scarce. Figure 9.15(a) shows precipitation retrievals in the tropics over a mix of land and sea, while Figure 9.15(b) shows a more intense tropical event. Figure 9.15(c) illustrates strong precipitation near 72° to 74° N, again over both land and sea. Finally, Figure 9.15(d) illustrates the March 5, 2001, New England snowstorm that deposited roughly a foot of snow within a few hours, an accumulation somewhat greater than is indicated by the retrieved rain rates of 1.2 mm/h. This applicability of the algorithm to snowfall rate should be expected because the observed radio emission originates exclusively at high

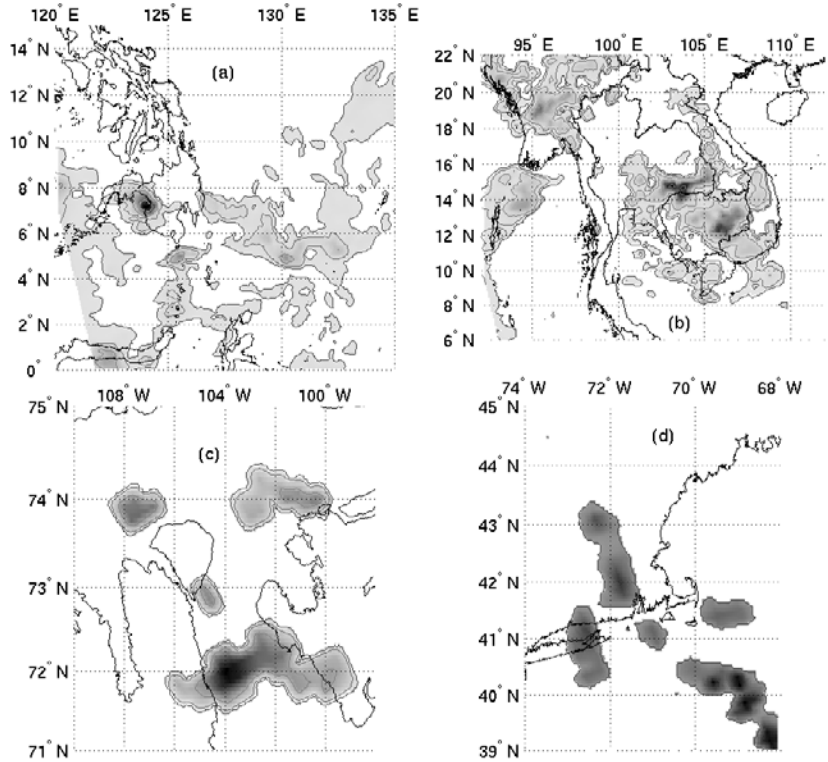


Figure 9.15 AMSU precipitation-rate retrievals (mm/h) with 15-km resolution. (a) Philippines, April 16, 2000, (b) Indochina, July 5, 2000, (c) Canada, August 2, 2000, and (d) New England snowstorm, March 5, 2001. Precipitation contours are drawn for 0.5, 2, 8, 32, and 128 mm/h. © 2003 IEEE [1].

altitudes. Whether the hydrometeors are rain or snow upon impact depends only on air temperatures near the surface, far below those altitudes being probed. For essentially all of the pixels shown in Figure 9.15, the adjacent clear air exhibited temperature and humidity profiles (inferred from AMSU) within the range of the training set. Nonetheless regional biases are expected and will require evaluation. For example, polar stratiform precipitation is expected to exhibit relatively weaker radiometric signatures in winter when the temperature lapse rates are lower, and snow-covered mountains in cold polar air can produce false detections.

9.5 Summary

In this chapter, we have described the use of a neural network for estimating precipitation from satellite-based passive microwave measurements. A prominent feature of this algorithm was the use of nontrivial signal processing for preprocessing the measurements. The signal processing methods chosen were consistent with the types of information that would be important to the development of precipitating clouds. Moreover, the variation of precipitation rates over logarithmic scales also suggested a post-processing step. The neural network was trained to estimate a quantity close to the base-10 logarithm of the precipitation rate so that training would not be dominated by heavy precipitation rates at the expense of lighter precipitation rates which are more common.

Future precipitation-rate estimation algorithms are likely to benefit from improvements in the preprocessing and post-processing methods and from the availability of greater quantities of higher-quality data. For example, data warping as presented in Chapter 7 is likely to be useful for capturing diurnal and seasonal variations.

References

[1] F. W. Chen and D. H. Staelin. "AIRS/AMSU/HSB precipitation estimates." *IEEE Trans. Geosci. Remote Sens.*, 41(2):410–417, February 2003.

[2] R. P. Lippmann. "An introduction to computing with neural nets." *IEEE ASSP Magazine*, 4:4–22, 1987.

[3] F. W. Chen and D. H. Staelin. "Millimeter-wave observations of precipitation using AMSU on the NOAA-15 satellite." *IEEE International Geoscience and Remote Sensing Symposium Proceedings*, pages 1044–1045, July 2001.

[4] F. W. Chen and D. H. Staelin. "Global millimeter-wave observations of precipitation using AMSU on the NOAA-15 satellite." *IEEE International Geoscience and Remote Sensing Symposium Proceedings*, pages 460–462, June 2002.

[5] D. H. Staelin, F. W. Chen, and A. Fuentes. "Precipitation measurements using 183-GHz AMSU satellite observations." *IEEE International Geoscience and Remote Sensing Symposium Proceedings*, pages 2069–2071, June 1999.

[6] D. H. Staelin and F. W. Chen. "Precipitation observations near 54 and 183 GHz using the NOAA-15 satellite." *IEEE Trans. Geosci. and Remote Sens.*, pages 2322–2332, September 2000.

[7] T. Wilheit, C. D. Kummerow, and R. Ferraro. "Rainfall algorithms for AMSR-E." *IEEE Trans. Geosci. Remote Sens.*, 41(2):204–213, February 2003.

[8] R. V. Leslie and D. H. Staelin. "NPOESS aircraft sounder testbed-microwave: Observations of clouds and precipitation at 54, 118, 183, and 425 GHz." *IEEE Trans. Geosci. Remote Sensing*, 42(10):2240–2247, October 2004.

[9] F. T. Ulaby, R. K. Moore, and A. K. Fung. *Microwave Remote Sensing: Active and Passive*, Volume 1. Addison Wesley Publishing Co., Reading, Massachusetts, 1981.

[10] B. Kedem, H. Pavlopoulos, X. Guan, and D. A. Short. "A probability distribution model for rain rate." *J. Appl. Meteorol.*, 33(12):1486–1493, December 1994.

[11] A. Fuentes-Loyola. *Precipitation Measurements Using 54- and 183-GHz AMSU Satellite Observations.* Master's thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, June 1999.

[12] F. W. Chen. *Global Estimation of Precipitation Using Opaque Microwave Bands*. Ph.D. thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, June 2004.

[13] I. T. Joliffe. *Principal Component Analysis*. Springer-Verlag, New York, 2002.

[14] W. J. Blackwell, J. W. Barrett, F. W. Chen, R. V. Leslie, P. W. Rosenkranz, M. J. Schwartz, and D. H. Staelin. "NPOESS aircraft sounder testbed-microwave (NAST-M): Instrument description and initial flight results." *IEEE Trans. Geosci. Remote Sens.*, 39(11):2444–2453, November 2001.

[15] A. J. Gasiewski, D. M. Jackson, J. R. Wang, P. E. Racette, and D. S. Zacharias. "Airborne imaging of tropospheric emission at millimeter and submillimeter wavelengths." *IEEE*

International Geoscience and Remote Sensing Symposium Proceedings, pages 663–665, August 1994.

[16] R. F. Adler, C. Kidd, G. Petty, M. Morissey, and H. M. Goodman. "Intercomparison of global precipitation products: The third precipitation intercomparison project (pip-3)." *Bull. Amer. Meteorol. Soc.*, 82(7):1377–1396, July 2001.

[17] E. A. Smith, J. E. Lamm, R. Adler, J. Alishouse, K. Aonashi, E. Barrett, P. Bauer, W. Berg, A. Chang, R. Ferraro, J. Ferriday, S. Goodman, N. Grody, C. Kidd, D. Kniveton, C. Kummerow, G. Liu, F. Marzano, A. Mugnai, W. Olson, G. Petty, A. Shibata, R. Spencer, F. Wentz, T. Wilheit, and E. Zipser. "Results of WETNET PIP-2 project." *J. Atmos. Sci.*, 55(9):1483–1536, May 1998.

[18] P. A. Arkin and P. Xie. "The global precipitation climatology project: First algorithm intercomparison project." *Bull. Amer. Meteorol. Soc.*, 75(3):401–420, March 1994.

[19] M. D. Conner and G. W. Petty. "Validation and intercomparison of SSM/I rain-rate retrieval methods over the continental United States." *J. of Appl. Meteorol.*, 37:679–700, 1998.

10

Neural Network Retrieval of Atmospheric Profiles from Microwave and Hyperspectral Infrared Observations

Modern atmospheric sounders measure radiance with unprecedented resolution (spatial, spectral, and temporal) and accuracy. For example, the Atmospheric Infrared Sounder (AIRS, operational in 2002) provides a spatial resolution of $\sim$15 km, a spectral resolution of $\nu/\Delta\nu \approx 1,200$ (with 2,378 channels from 650 cm^{-1} to 2,675 cm^{-1}), and a radiometric accuracy on the order of ± 0.2 K. Approximately 90% of the Earth's atmosphere is measured (in the horizontal dimension) every 12 hours. This wealth of data presents two major challenges from the point of view of retrieval algorithm development. The first concerns the robustness of the retrieval operator and involves maximal use of the geophysical content of the radiance data with minimal interference from instrument and atmospheric noise. The second concerns computational efficiency, where it is desirable to implement a robust algorithm within a given computational budget.

In this chapter, we present a nonlinear retrieval algorithm that offers the numerical stability and efficiency of statistical methods without sacrificing the accuracy of physical, model-based methods. The algorithm is implemented in two stages: a linear transform is first used to reduce the dimensionality and optimally extract geophysical information, and a multilayer feedforward neural network (NN) is subsequently used to estimate the desired geophysical profiles. The preprocessing in the first stage allows relatively small neural networks to be used and consequently yields better retrieval accuracies, reduced

sensitivity to input noise, improved resistance to instabilities (overfitting, for example), and reduced computational burden.

This chapter is organized as follows. First, the projected principal components (PPC) radiance compression operator derived in Chapter 4 is combined with a feedforward multilayer perceptron neural network to form the PPC/NN retrieval algorithm. Performance analyses comparing the PPC/NN algorithm to traditional retrieval methods are then presented, including both simulated clear-air and observed partially cloudy data from the AIRS and AMSU (Advanced Microwave Sounding Unit) sensors. While only AIRS/AMSU observations are considered here, the PPC/NN algorithm could easily be applied to other microwave and hyperspectral infrared sensor suites such as the Cross-track Infrared Sounder (CrIS) and the Advanced Technology Microwave Sounder (ATMS) scheduled for launch on the NPP/NPOESS satellite series and the Infrared Atmospheric Sounding Interferometer (IASI) and AMSU currently operational on the EUMETSAT Polar System (EPS). Retrieval sensitivity analyses are presented, including the effects of sensor scan angle, orbit type (ascending or descending), cloud fraction, and training set comprehensiveness. Finally, an overview of future work is given, including a discussion on how the PPC/NN algorithm could be used in operational systems.

10.1 The PPC/NN Algorithm

The use of multilayer feedforward neural networks, such as the multilayer perceptron (MLP), to retrieve temperature profiles from hyperspectral radiance measurements has been addressed by several investigators (see [1–4], for example). Neural network retrieval of moisture profiles [5–7] and trace gas amounts [8] from hyperspectral data is relatively new, but follows the same methodology used to retrieve temperature.

A first attempt to combine the properties of both neural network estimators and principal component transforms for the inversion of microwave radiometric data to retrieve atmospheric temperature and moisture profiles is reported in [9], and a more recent study with hyperspectral data is presented in [5]. A conceptually similar approach is taken in this work by combining the PPC compression preprocessing technique with the neural network estimation. As shown in Chapter 4, PPC compression offers substantial performance advantages over traditional PCA and is the cornerstone of the work presented in this chapter.

10.1.1 Network Topology

All multilayer perceptrons used in the PPC/NN algorithm are composed of one or two hidden layers of nonlinear (hyperbolic tangent) nodes and an output layer of linear nodes. For the temperature retrieval, 25 PPC coefficients are input to six neural networks, each with a single hidden layer of 15 nodes. Separate neural networks are used for different vertical regions of the atmosphere; a total of six networks are used to estimate the temperature profile at 65 points from the surface to 50 mbar. For the water vapor retrieval, 35 PPC coefficients are input to nine neural networks, each with a single hidden layer of 25 nodes. The water vapor profile (mass mixing ratio) is estimated at 58 points from the surface to 75 mbar. These network parameters were determined largely through empirical analyses but could have been dynamically optimized as the neural network trained, as discussed in Chapter 6. Separate training, validation, and testing data sets were used; these data sets will be discussed in Sections 10.2.1 and 10.3.2.

10.1.2 Network Training

The weights and biases were initialized using the Nguyen-Widrow initialization method [10], and the neural network was trained using the Levenberg-Marquardt backpropagation algorithm discussed in Section 6.4.3. For each epoch, the μ parameter was initialized to 0.001. If a successful step was taken (i.e., $E(\mathbf{w} + \mathbf{dw}) < E(\mathbf{w})$), then μ was decreased by a factor of 10. If the current step was unsuccessful, the value of μ was increased by a factor of 10 until a successful step could be found (or until μ reached 10^{10}). The network training was stopped when the error on the validation data set did not decrease for 10 consecutive epochs. The sensor noise that is added to the simulated radiances was changed on each training epoch to desensitize the network to radiance measurement errors. Multiple training runs were carried out to increase the likelihood that a global minimum was obtained.

10.2 Retrieval Performance Comparisons with Simulated Clear-Air AIRS Radiances

In this section, the temperature and moisture profile retrieval performance is compared for three methods using simulated clear-air AIRS radiances. The three retrieval methods are linear regression, the PPC/NN algorithm, and an iterated minimum-variance (IMV) technique.

10.2.1 Simulation of AIRS Radiances

The Atmospheric Infrared Sounder (AIRS) is a high-resolution grating spectrometer with 2,378 channels from 650 cm^{-1} to 2,675 cm^{-1} [11]. Nominal noise (expressed in units of "noise-equivalent delta temperature," or NEΔT) values used in this study are shown in Table 10.1. These values were converted to units of spectral radiance ($mW/ster \cdot m^2 \cdot cm^{-1}$), and noise values for the full spectrum were obtained using linear interpolation. While the noise values of actual AIRS data exhibit significantly more spectral structure than those used here, subsequent PPC/NN simulation studies have demonstrated that the simple noise model is adequate for use in general retrieval performance assessments like those we present later in the chapter.

10.2.1.1 Atmospheric Transmittance and Radiative Transfer Models

All transmittance and radiative transfer computations performed in this study (including both the statistical and physical retrieval algorithms) were based on the algorithm of Strow et al. [12]. Surface reflection of solar radiation is included in the model.

10.2.1.2 Radiosonde Data Set

The ground truth for this study was the NOAA88b radiosonde data set, which contains 7,547 radiosonde/rocketsonde profiles, globally distributed seasonally and geographically. Atmospheric temperature, moisture, and ozone are given at 100 discrete levels ranging in pressure from 0.0160 mbar to 1,100 mbar. It should be noted that water vapor values above the highest altitude radiosonde report, generally between 500 and 300 mbar, are subject to artifacts resulting from the method used to combine low- and high-altitude data. Therefore, water vapor measurements above approximately 300 mbar are of questionable quality. Skin surface temperature is also recorded. Three mutually exclusive sets were randomly selected from these 7,547 profiles with approximately an 80–10–10 split.

10.2.1.3 Surface Model

The surface pressure was fixed at 1,013.25 mbar for all profiles (variable surface pressure is explored in Section 10.3.4), and a Lambertian surface was assumed. Frequency dependence of the surface emissivity was modeled as a piecewise-linear function with six random (Gaussian) hinge points. The means and standard deviations of the hinge points are shown in Table 10.2. The correlation matrix of the hinge points was assumed to be a Toeplitz

Table 10.1
Assumed NEΔT Noise Characteristics of AIRS at 250K

ν (cm^{-1})	NEΔT (K)	ν (cm^{-1})	NEΔT (K)
648.93	0.370	980.39	0.100
684.93	0.320	2,325.58	0.100
740.19	0.300	2,439.02	0.140
740.74	0.180	2,564.10	0.200
847.46	0.140	2,702.70	0.250
900.90	0.110		

Table 10.2
Means and Standard Deviations of the Surface Emissivity Hinge Points

ν (cm^{-1})	Mean	Standard Deviation
769	0.9501	0.0102
909	0.9336	0.0170
1,111	0.9172	0.0238
2,105	0.9007	0.0303
2,500	0.8844	0.0367
2,857	0.8678	0.0432

matrix, with a first column of 1.0000, 0.9193, 0.8348, 0.7501, 0.6697, and 0.5820.

10.2.2 An Iterated Minimum-Variance Technique for the Retrieval of Atmospheric Profiles

We now develop an iterated, physical retrieval method of the type described in Section 3.4. Let S be a vector that represents some atmospheric state variable, such as the temperature profile, and R be a vector of measurements. The Bayesian retrieval approach seeks to find the most probable value of the atmospheric state S given the measurements R. In the case of Gaussian error statistics, the most probable solution is that which minimizes the following cost function [13]:

$$\begin{aligned} \gamma(\cdot) &= (S - S_0)^T \mathbf{C}_{SS}^{-1}(S - S_0) \\ &+ (R - \mathbf{f}(S))^T \mathbf{C}_{\Psi\Psi}^{-1}(R - \mathbf{f}(S)), \end{aligned} \tag{10.1}$$

where S_0 is a "first guess" with expected error covariance $\mathbf{C}_{SS}$, $\mathbf{f}(S)$ is the "forward model," and $\mathbf{C}_{\Psi\Psi}$ is the expected covariance of the measurement error.

As discussed in Section 3.4, the forward model can be expanded as a Taylor series about a guessed value S_0, and Newtonian iteration can be used to iteratively converge to a solution. The solution after the nth iteration can be expressed in closed form as:

$$\begin{aligned} S_{n+1} &= S_0 + \mathbf{C}_{SS}\mathbf{K}_n^T(\mathbf{K}_n\mathbf{C}_{SS}\mathbf{K}_n^T + \mathbf{C}_{\Psi\Psi})^{-1} \\ &\times [R - R_n - \mathbf{K}_n(S_0 - S_n)], \end{aligned} \tag{10.2}$$

where $\mathbf{K} = \partial\mathbf{f}/\partial S$ is the Fréchet derivative, which is computed at each iteration.

In this work, the retrievals are carried out in four successive steps after a first guess is computed using linear regression, as suggested by Susskind et al. [14]. The retrieval steps are: (1) surface temperature and emissivity (53 channels used), (2) temperature profile (147 channels used), (3) moisture profile (66 channels used), and (4) improved temperature profile (154 channels used). Step four is identical to step two, with the exception that seven additional channels in the water vapor band that produce sharp temperature weighting functions are included. The channel selection is identical to that of Susskind et al. [14]. Each step is carried to completion before the execution of the next step. The measurement error covariance matrix $\mathbf{C}_{\Psi\Psi}$ includes uncertainties due to surface temperature and emissivity, temperature profile, water vapor profile, ozone profile, and instrument noise. These uncertainties are updated at the completion of each retrieval step.

It should be stated that while the IMV retrieval methodology presented here is similar to that used operationally to process AIRS data [14], there are notable differences, including the regularization method used to stabilize the solution and the use of output kernel functions. Direct comparison of the PPC/NN method with the AIRS Level 2 operational algorithm (version 3) is given in Section 10.3.

10.2.3 Retrieval Performance Comparisons

10.2.3.1 Retrieval Accuracy

Temperature profile retrieval performance results on the testing data set for linear regression, the PPC/NN method, and the IMV method are shown in Figure 10.1; water vapor retrieval results are shown in Figure 10.2. Both figures show errors in 1-km vertical layers. The water vapor results are expressed as a percentage, where, for each profile, the mass mixing ratio error in the layer is weighted by the true mass mixing ratio in the layer.

The PPC/NN temperature profile retrieval performance is superior to that of the iterated minimum-variance algorithm over almost all of the atmosphere and is substantially better near the surface. The PPC/NN water vapor profile retrieval performance is also superior to that of the iterated minimum-variance algorithm over almost all of the atmosphere, and is substantially better in the upper troposphere, although this is probably due to the neural network fitting to statistical artifacts inherent in the NOAA88b data. The IMV water vapor performance near the surface is better than that of the PPC/NN by approximately 1.5%.

A separate experiment was performed where the PPC/NN estimates were used as the first guess for the IMV algorithm. It was found that the IMV method offered virtually no improvement over the PPC/NN estimates.

10.2.3.2 Computational Efficiency

There are performance benefits of the PPC/NN method other than retrieval accuracy. The high degree of radiance compression afforded by the PPC method allows relatively small neural networks (with a few thousand free parameters) to be used, thus allowing the use of fast (but memory-intensive) training algorithms, such as Levenberg-Marquardt, on commodity personal computers. The algorithm is extremely fast once trained (retrieval of the temperature and moisture profiles for a single radiance observation takes on the order of a millisecond), and training of the complete algorithm over a global ensemble of 10,000 profiles takes less than an hour on a desktop workstation (Intel Xeon operating at 3 GHz). The required computation time for most iterated model-based retrieval techniques can exceed that of the PPC/NN approach by two orders of magnitude. Furthermore, changes in instrument behavior (for example, bad or excessively noisy channels and instrument spectral response function shape and/or center changes) can be accommodated by recalculating the PPC coefficients (with relatively light computational burden). It is usually not necessary to retrain the neural networks if only a small number (less than about 1%) of the original channels are affected, as their marginal impact on the resulting PPC coefficients is small because of the large amount of spectral redundancy.

10.2.4 Discussion

The retrieval accuracy of the PPC/NN methods exceeds that of the IMV method over most of the troposphere, and there are several possible reasons for this improved performance. First, the PPC/NN method used all available AIRS channels (2,378) whereas the IMV method used only 273. The spectral redundancy afforded by the use of all AIRS channels facilitates the filtering of

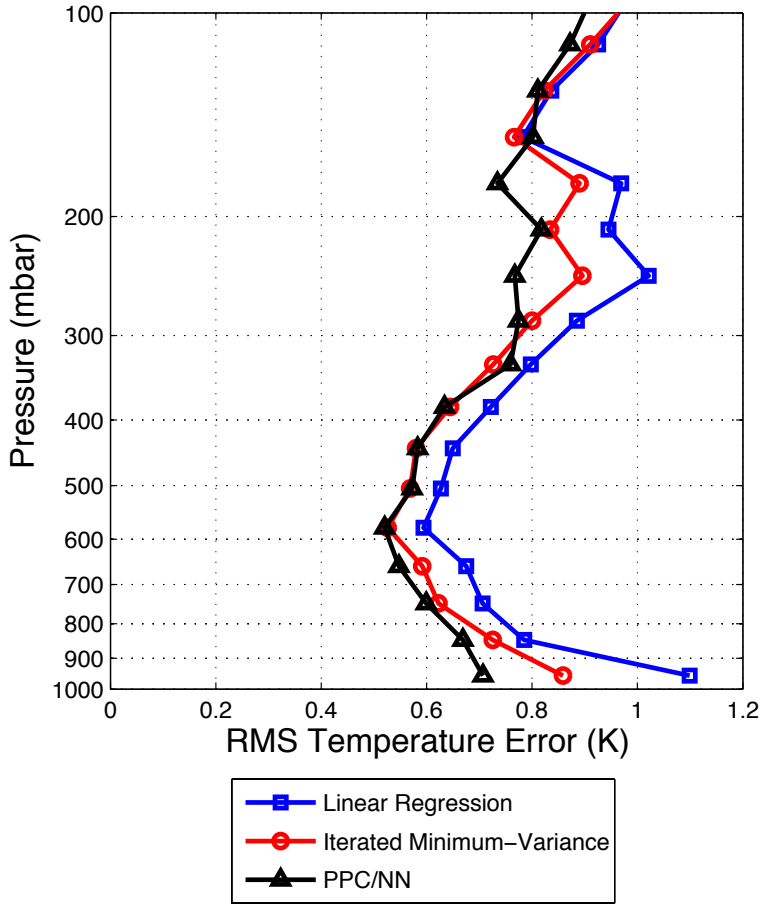


Figure 10.1 Temperature retrieval performance of the PPC/NN, IMV, and linear regression methods. RMS errors of 1-km layer means are shown. © 2005 IEEE [1].

noise from the radiances via the PPC transform. Second, correlation between temperature and moisture is taken into account during NN training, but not included in the IMV a priori statistics because temperature and moisture are retrieved in separate steps. Third, although each step in the IMV retrieval was iterated to convergence, it is possible that the IMV may not have found the overall minimum of the cost function. Finally, minimization of (10.1) yields the minimum-error-variance solution (even for non-Gaussian statistics) provided the measurement errors are uncorrelated with the atmospheric

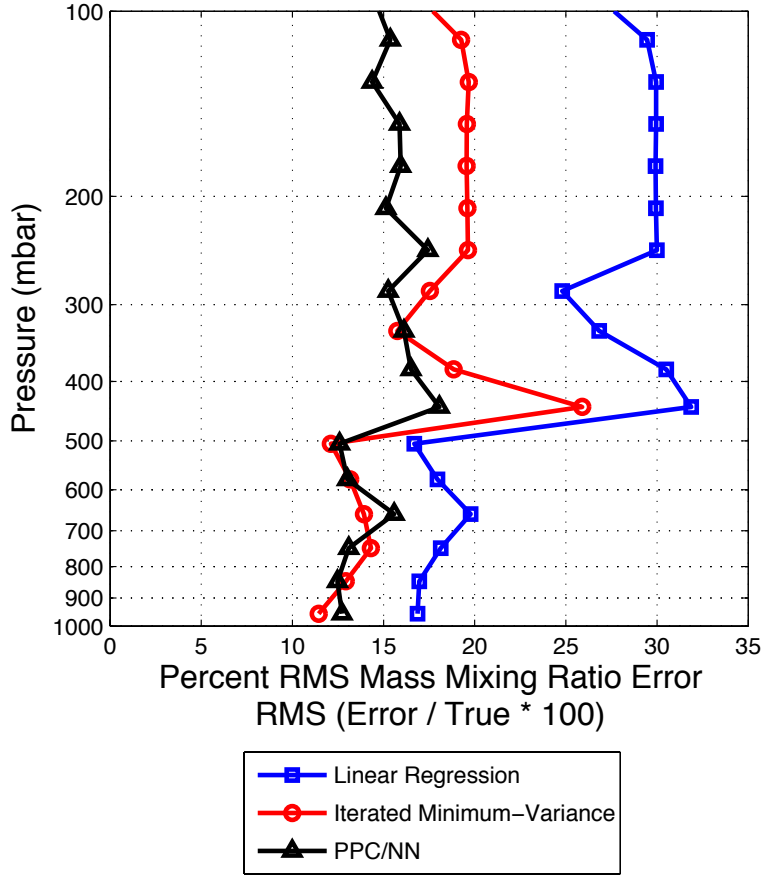


Figure 10.2 Water vapor (mass mixing ratio) retrieval performance of the PPC/NN, IMV, and linear regression methods. RMS errors of 1-km layer means are shown. Errors (as a function of pressure, P) are expressed as a percentage, that is, $100 \times \frac{q_{true}(P) - q_{est}(P)}{q_{true}(P)}$. © 2005 IEEE [1].

state. However, in this work the measurements were expressed as brightness temperatures and not as spectral radiances; therefore, the independent-error assumption may be violated. The neural network is able to fully exploit any non-Gaussian statistical relationships that exist between the radiances and the geophysical parameters.

10.3 Validation of the PPC/NN Algorithm with AIRS/AMSU Observations of Partially Cloudy Scenes over Land and Ocean

In this section, the performance of the PPC/NN algorithm is evaluated using cloud-cleared AIRS data (where the cloud-clearing is performed using both AIRS and AMSU data) and colocated European Center for Medium-range Weather Forecasts (ECMWF) atmospheric fields. The PPC/NN retrieval performance is compared with that obtained using the NASA AIRS "Level 2" algorithm.[1] Both ocean and land cases are considered, including elevated surface terrain, and retrievals at all sensor scan angles (out to $\pm 48^{\circ}$) are derived. Finally, sensitivity analyses of PPC/NN retrieval performance are presented with respect to scan angle, orbit type (ascending or descending), cloud amount, and training set comprehensiveness.

10.3.1 Cloud Clearing of AIRS Radiances

The cloud-clearing approach discussed in Susskind et al. [14, 15] was applied to the AIRS data as part of the Version-3 Level 2 processing. The algorithm seeks to estimate a clear-column radiance (the radiance that would have been measured if the scene were cloud-free) from a number of adjacent cloud-impacted fields of view.

10.3.2 AIRS/AMSU/ECMWF Data Set

The performance of the PPC/NN algorithm was evaluated using 352,903 AIRS/AMSU observations and colocated ECMWF atmospheric fields collected on 7 days throughout 2002 and 2003: September 6, 2002, January 25, 2003, June 8, 2003, August 21, 2003, September 3, 2003, October 12, 2003, and December 5, 2003. The 352,903 observations were randomly divided into training, validation, and testing sets with approximately an 80–10–10 split. Approximately two-thirds of the profiles were observed over an ocean surface. The a priori RMS variation of the temperature and water vapor (mass mixing ratio) profiles are shown in Figure 10.3. The observations were matched with AIRS Level 2 retrievals obtained from the Earth Observing System (EOS) Data Gateway (EDG). As advised in the AIRS Version 3.0 L2 Data Release Documentation, only retrievals that met certain quality standards (specifically,

1. The "Version-3" AIRS Level 2 product was used in this work for simple illustrative purposes only. Readers interested in the use of AIRS Level 2 products for detailed, high-fidelity comparisons are encouraged to obtain the most recent version, which offers substantial improvements in both accuracy and yield in relation to the Version-3 products.

RetQAFlag = 0 for ocean and RetQAFlag = 256 for land) were included in the analyses. There were 17,856 AIRS Level 2 retrievals (all within $\pm 40^{\circ}$ latitude) that met this criterion.

In order to facilitate comparison with results published in the AIRS v3.0 Validation Report [16], layer error statistics are calculated slightly differently in this section than they were in the previous section. First, layer averages are calculated in layers of approximately (but not exactly) 1-km width; the exact layer widths can be found in Appendix III in the AIRS v3.0 Validation Report. Second, weighted water vapor errors in each layer are calculated by dividing the RMS mass mixing ratio error by the RMS variation of the true mass mixing ratio (as opposed to dividing the mass mixing ratio error of each profile by the true mass mixing ratio for that profile and computing the RMS of the resulting ensemble).

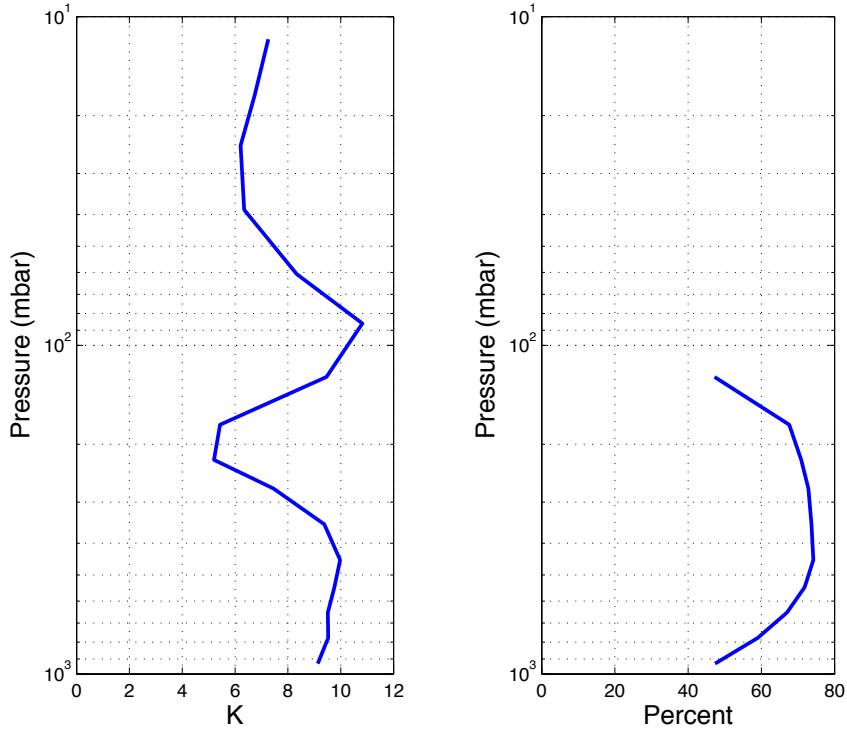
10.3.3 AIRS/AMSU Channel Selection

Thirty-seven percent (888 of the 2,378) of the AIRS channels were discarded for the analysis, as the radiance values for these channels frequently were flagged as invalid by the AIRS calibration software. A simulated AIRS brightness temperature spectrum is shown in Figure 10.4, which shows the original 2,378 AIRS channels and the 1,490 channels that were selected for use with the PPC/NN algorithm. All 15 AMSU channels were used. The algorithm automatically discounts channels that are excessively corrupted by sensor noise (for example, AMSU channel 7) or other interfering signals (for example, the effects of nonlocal thermodynamic equilibrium) because the corruptive signals are largely uncorrelated with the geophysical parameters that are to be estimated.

10.3.4 PPC/NN Retrieval Enhancements for Variable Sensor Scan Angle and Surface Pressure

The PPC/NN retrieval results presented in Section 10.2 corresponded to simulated measurements at a fixed scan angle (nadir) and a fixed surface pressure (1,013.25 mbar). When dealing with real AIRS/AMSU data, a variety of scan angles and surface pressures must be accommodated. Therefore, two additional inputs were added to the neural networks discussed in Section 10.2: (1) the secant of the scan angle, and (2) the forecast surface pressure (in mbar) divided by 1,013.25. The resulting temperature and water vapor profile estimates were reported on a variable pressure grid anchored by the forecast surface pressure.

Because of the number of inputs to the neural networks increased, the number of hidden nodes in networks used for temperature retrievals was



(a) a priori temperature error

(b) a priori mass mixing ratio (MMR) error

Figure 10.3 Temperature (a) and water vapor (b) profile statistics for the testing data set used in the analysis. See the text for details on how the statistics are computed at each layer. © 2005 IEEE [1].

increased from 15 to 20. For water vapor retrievals, the number of hidden nodes in the first hidden layer was maintained at 25, but a second hidden layer of 15 hidden nodes was added.

10.3.5 Retrieval Performance

We now compare the retrieval performance of the PPC/NN, linear regression, and AIRS Level 2 methods. For both the ocean and land cases, the PPC/NN and linear regression retrievals were derived using the same training set, and the same testing set was used to evaluate the performance of all methods.

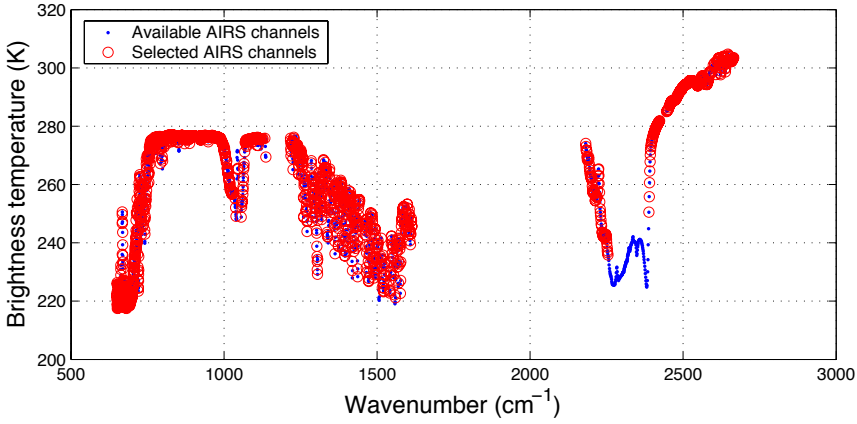


Figure 10.4 A typical AIRS spectrum (simulated) is shown; 1,490 out of 2,378 AIRS channels were selected. © 2005 IEEE [1].

10.3.5.1 Ocean Cases

The temperature profile retrieval performance over ocean for the linear regression retrieval, the PPC/NN retrieval, and the AIRS Level 2 retrieval is shown in Figure 10.5, and the water vapor retrieval performance is shown in Figure 10.6. The error statistics were calculated using the 13,156 (out of 40,000) AIRS Level 2 retrievals that converged successfully. A bias of approximately 1K near 100 mbar was found between the AIRS Level 2 temperature retrievals and the ECMWF data (ECMWF was colder). This bias was removed prior to computation of the AIRS Level 2 retrieval error statistics, which are shown in Figure 10.5.

10.3.5.2 Land Cases

The temperature profile retrieval performance over land for the linear regression retrieval, the PPC/NN retrieval, and the AIRS Level 2 retrieval is shown in Figure 10.7, and the water vapor retrieval performance is shown in Figure 10.8. The error statistics were calculated using the 4,700 (out of 10,000) AIRS Level 2 retrievals that converged successfully.

10.3.5.3 Discussion

There are several features in Figures 10.5 to 10.8 that are worthy of note. First, for all retrieval methods, the performance over land is worse than that

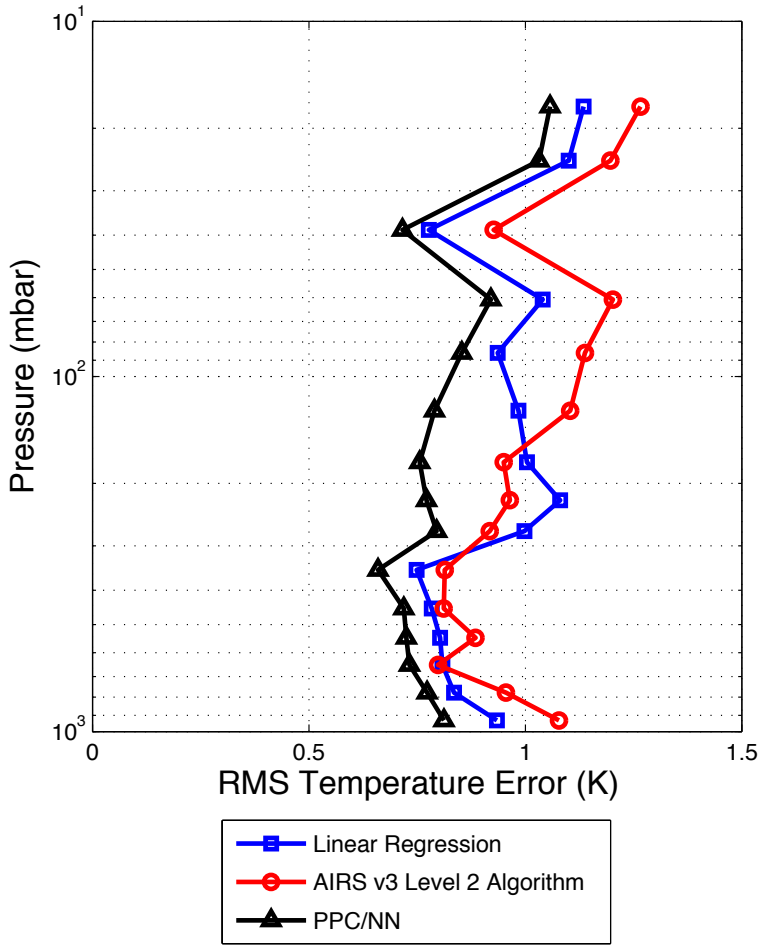


Figure 10.5 Temperature retrieval performance of the PPC/NN, linear regression, and AIRS Level 2 methods over ocean. Statistics were calculated over 13,156 fields of regard. © 2005 IEEE [1].

over ocean, as expected. The cloud-clearing problem is significantly more difficult over land, as variations in surface emissivity can be mistaken for cloud perturbations, thus resulting in improper radiance corrections. Second, the magnitude of the temperature profile error degradation for land versus ocean is larger for the PPC/NN algorithm than for the AIRS Level 2 algorithm. In fact, the temperature profile retrieval performance of the AIRS Level 2 algorithm is superior to that of the PPC/NN algorithm throughout most of the

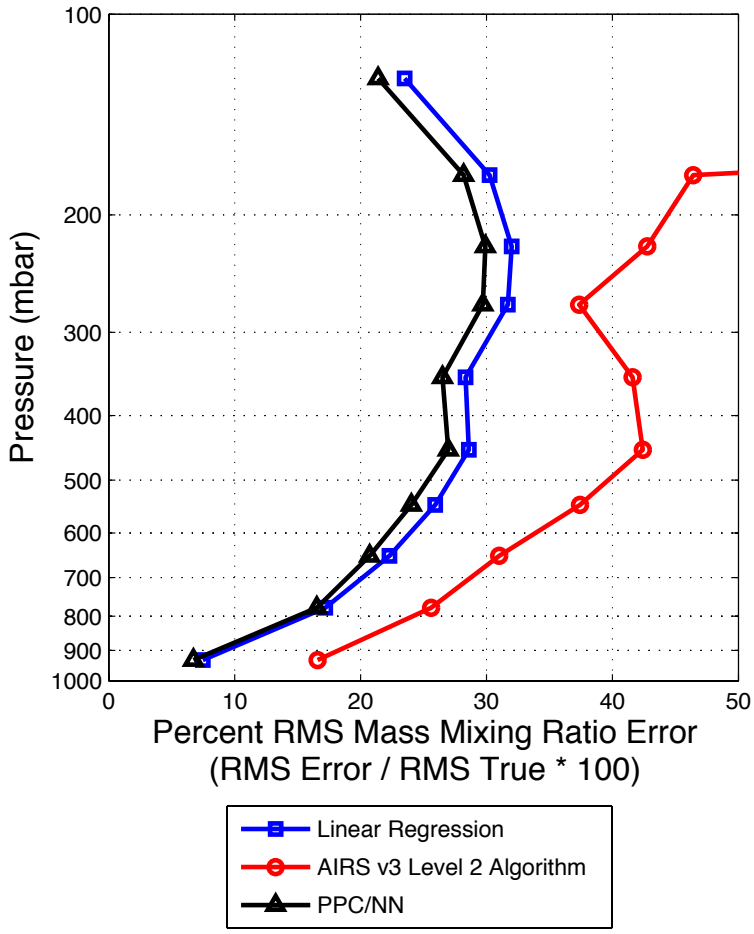


Figure 10.6 Water vapor (mass mixing ratio) retrieval performance of the PPC/NN, linear regression, and AIRS Level 2 methods over ocean. Statistics were calculated over 13,156 fields of regard. © 2005 IEEE [1].

lower troposphere over land. Further analyses of this discrepancy suggest that the performance of the PPC/NN method over elevated terrain is suboptimal, and could be improved. Recent work [17] combining stochastic cloud clearing with neural network estimation (referred to as the SCC/NN algorithm) has shown excellent promise, and global comparisons with the latest AIRS Level 2 products reveal that SCC/NN performance exceeds that of AIRS Level 2 relative to ECMWF throughout the troposphere.

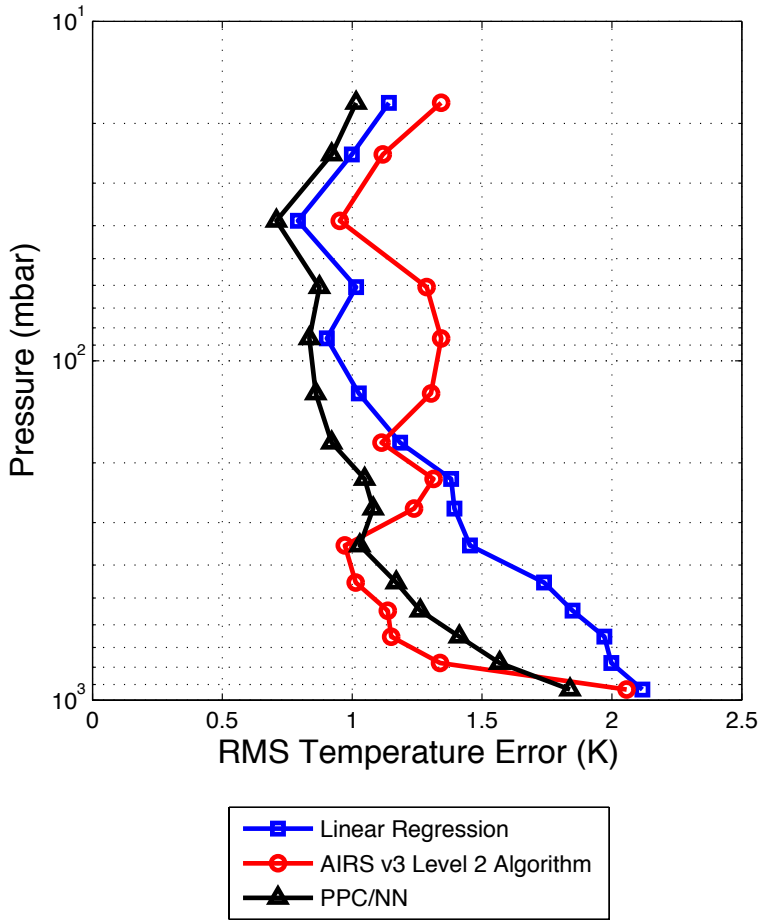


Figure 10.7 Temperature retrieval performance of the PPC/NN, linear regression, and AIRS Level 2 methods over land. Statistics were calculated over 4,700 fields of regard. © 2005 IEEE [1].

10.3.6 Retrieval Performance Sensitivity Analyses

In this section, the PPC/NN retrieval performance over ocean is stratified by cloud amount, sensor scan angle, orbit type, and training set comprehensiveness, and it is shown that the PPC/NN retrieval is relatively insensitive to each of these parameters. The sensitivity of the PPC/NN retrieval to cloud fraction is compared with that exhibited by the AIRS Level 2 retrieval.

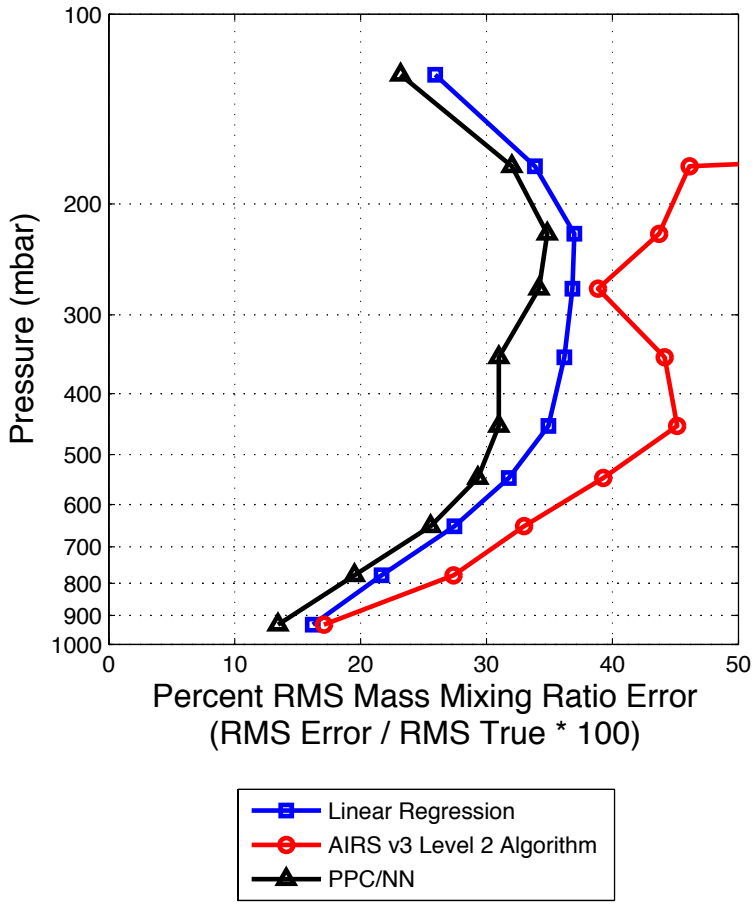


Figure 10.8 Water vapor (mass mixing ratio) retrieval performance of the PPC/NN, linear regression, and AIRS Level 2 methods over land. Statistics were calculated over 4,700 fields of regard. © 2005 IEEE [1].

10.3.6.1 Sensitivity to Cloud Amount

A plot of the temperature retrieval error in the layer closest to the surface as a function of the cloud fraction retrieved by the AIRS Level 2 algorithm (only cloud fractions less than 80% are reliably retrieved by the algorithm) is shown in Figure 10.9. Similar curves for the water vapor retrieval performance are shown in Figure 10.10. Both methods produce temperature and moisture retrievals with RMS errors near 1K and 15%, respectively, even in cases with large cloud fractions. The figures show that the PPC/NN temperature and

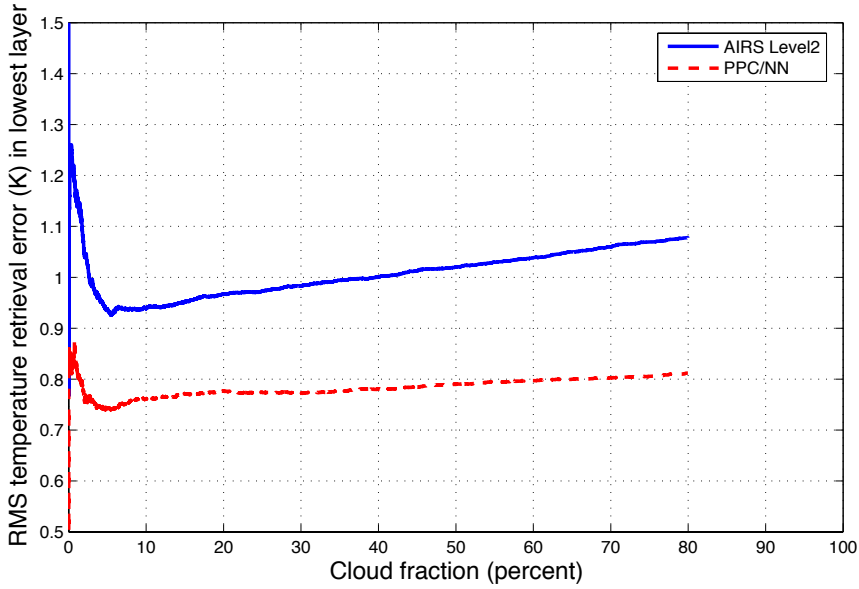


Figure 10.9 Cumulative RMS temperature error in the layer closest to the surface. Pixels were ranked in order of increasing cloudiness according to the retrieved cloud fraction from the AIRS Level 2 algorithm. No retrievals were attempted by the AIRS Level 2 algorithm if the retrieved cloud fraction exceeded 80%. © 2005 IEEE [1].

moisture retrievals are less sensitive than the AIRS Level 2 retrievals to cloud amount.

10.3.6.2 Sensitivity to Sensor Scan Angle

Figure 10.11 shows the PPC/NN temperature and moisture retrieval errors stratified by sensor scan angle. At the extreme scan angles, the temperature retrieval errors increase by approximately 0.2K and the moisture retrieval errors increase by approximately 4%.

10.3.6.3 Sensitivity to Satellite Orbit Type

Figure 10.12 shows the PPC/NN temperature and moisture retrieval errors stratified by satellite orbit type (ascending versus descending orbits). There is almost no dependence evident, which indicates the retrieval is robust to a variety of solar illumination conditions.

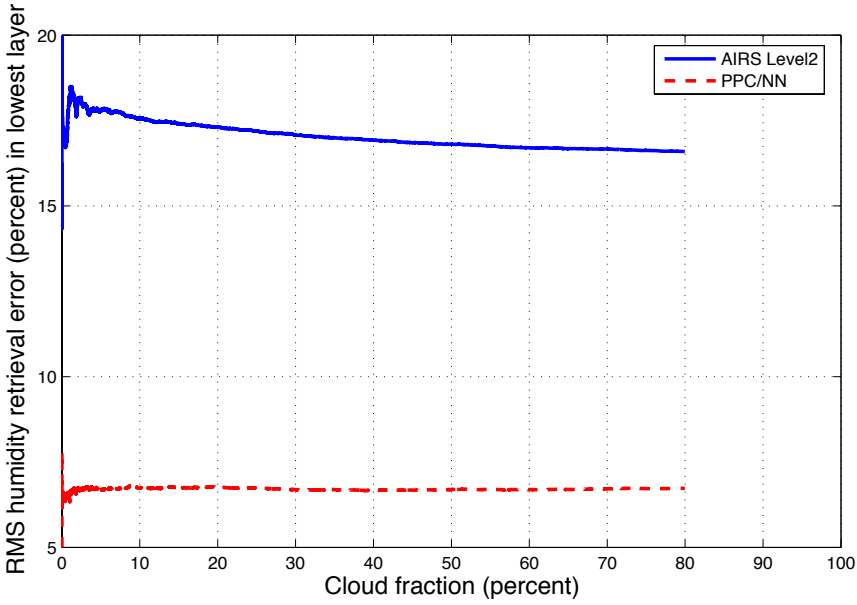
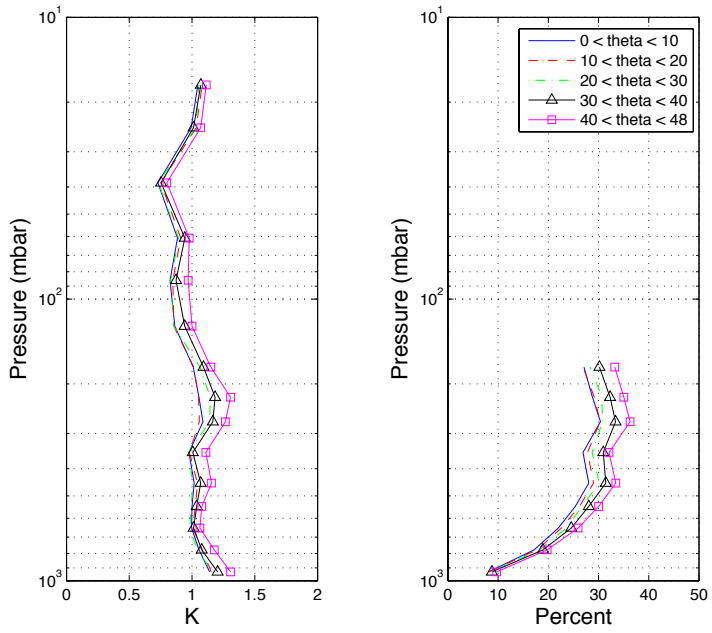


Figure 10.10 Cumulative RMS water vapor error in the layer closest to the surface. Pixels were ranked in order of increasing cloudiness, according to the retrieved cloud fraction from the AIRS Level 2 algorithm. No retrievals were attempted by the AIRS Level 2 algorithm if the retrieved cloud fraction exceeded 80%. © 2005 IEEE [1].

10.3.6.4 Sensitivity to Training Set Extensiveness

As a final test of the stability of the PPC/NN algorithm, the ability of the neural network to generalize to time periods not represented in the training set is examined. As discussed in Chapter 6, the extensiveness of a training data set refers to the extent that the entire dynamic range of the multidimensional input and target space is exercised. For this experiment, the first 2 days (September 6, 2002, and January 25, 2003) of the 7-day training set were removed. A new testing set was assembled, consisting only of the 2 days not represented in the training set. This new testing set was applied to the original PPC/NN algorithm, which had been trained using all seven days, and the PPC/NN algorithm which was trained with the training set containing only five days. Temperature and moisture retrieval results for this experiment are shown in Figure 10.13. The temperature and moisture retrieval errors increase by approximately 0.1 K and 2%, respectively, as a result of the limited training set. This suggests that the PPC/NN is relatively insensitive to seasonal



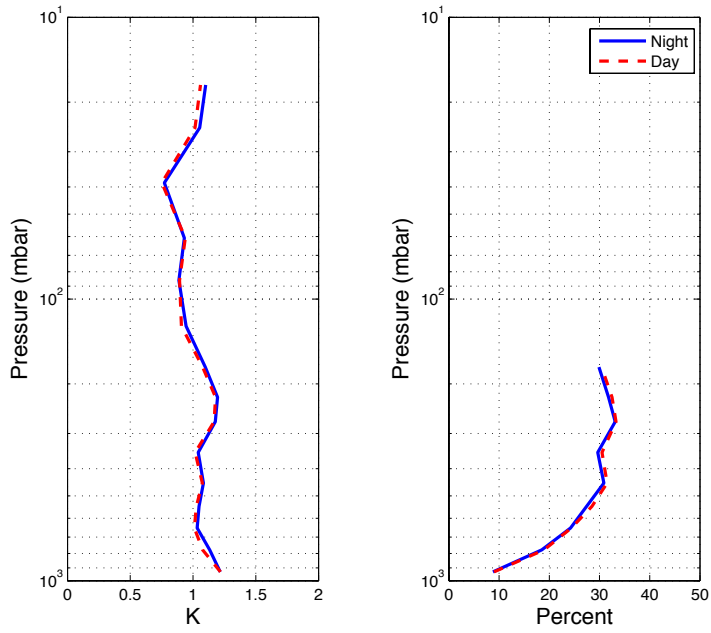
(a) Temperature error (b) Water vapor error

Figure 10.11 PPC/NN retrieval performance stratified by sensor scan angle for temperature profile retrieval (a) and water vapor profile retrieval error (b). Statistics were calculated over 40,000 fields of regard. © 2005 IEEE [1].

and year-to-year changes, and any residual error due to this effect could be reduced by stratifying the training set by season and, if necessary, updating the PPC/NN coefficients periodically.

10.3.7 Discussion and Future Work

While the PPC/NN performance results presented in the previous section are very encouraging, several caveats must be mentioned. The ECMWF fields used for "ground truth" contain errors, and the neural network will tune to these errors as part of its training process. Therefore, the PPC/NN RMS errors shown in the previous section may be underestimated, and the AIRS Level 2 RMS errors may be overestimated, as the ECMWF data are an imperfect representation of the true state of the atmosphere. Therefore, the "true" spread between the performance of the PPC/NN and AIRS Level 2 algorithms is almost certainly smaller than that shown here. Work is currently under way to test the performance of both the PPC/NN and AIRS Level 2



(a) Temperature error (b) Water vapor error

Figure 10.12 PPC/NN retrieval performance stratified by satellite orbit type: ascending (daytime) and descending (nighttime) for temperature profile retrieval (a) and water vapor profile retrieval error (b). Statistics were calculated over 40,000 fields of regard. © 2005 IEEE [1].

algorithms with additional ground truth data, including radiosonde data, and ground- and aircraft-based measurements [18]. It should be noted that the PPC/NN algorithm as implemented in this work is currently not a stand-alone system, as both AIRS cloud-cleared radiances and quality flags produced by the AIRS Level 2 algorithm are required. Recent work [19] has included the adaptation of the PPC/NN algorithm for use directly on cloudy AIRS/AMSU radiances, and quality assessments of the retrieved products are now produced by the neural network algorithm. Finally, assimilation of PPC/NN-derived atmospheric parameters into numerical weather prediction models is planned, and the resulting impact on forecast accuracy will be an excellent indicator of retrieval quality.

In light of the previous comments, it is necessary to consider the steps that would be required to implement the PPC/NN retrieval technique in an operational system. Most importantly, perhaps, is the training methodology that will be needed. For reasons previously discussed, it will probably not

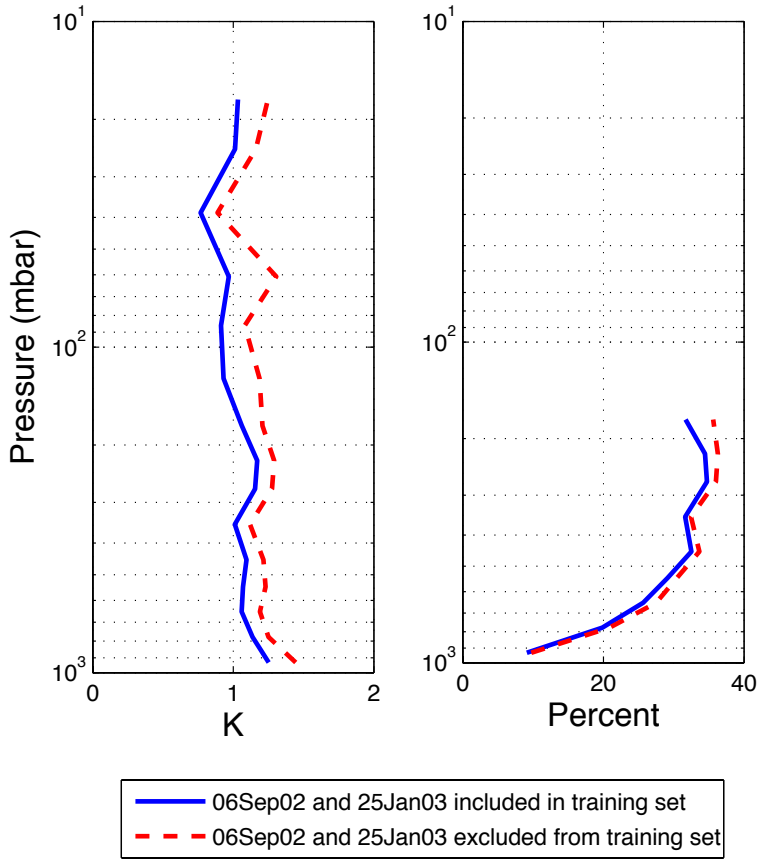


Figure 10.13 PPC/NN retrieval performance for two training sets. The same validation set was used in both cases: 11,877 profiles observed on 6 Sep 2002 and 8 Jun 2003. The first training set includes profiles from all seven days. For the second training set, all profiles observed on 6 Sep 2002 and 8 Jun 2003 were removed. © 2005 IEEE [1].

be feasible to derive the PPC/NN coefficients using a forecast model. Two products from the EOS-Aqua system should be an invaluable source of training data for future missions. The first is the database of radiances (both raw and cloud-cleared) and atmospheric parameter retrievals that is being generated. Second, and more importantly, is the improved validation of surface, cloud, and transmittance models as a direct result of product validation efforts. These models, together with the database of atmospheric retrievals being generated, will provide a "ground-truth laboratory" from

which training data sets for future sensors can be derived.

10.4 Summary and Conclusions

A novel statistical retrieval technique was introduced that combines a linear radiance compression operator with a neural network estimator. The projected principal components (PPC) transform was shown to be well suited for this application because information correlated to the geophysical quantity of interest is optimally represented with only a few dozen components. This substantial amount of radiance compression (approximately a factor of 100) allows relatively small neural networks to be used, thereby improving both the stability and computational efficiency of the algorithm. Test cases with both simulated clear-air and observed partially cloudy AIRS/AMSU data demonstrate that the PPC/NN temperature and moisture retrievals yield accuracies commensurate with those of physical methods at a substantially reduced computational burden. Retrieval accuracies (defined as agreement with ECMWF fields) near 1K for temperature and 25% for water vapor mass mixing ratio in layers of approximately 1-km thickness were obtained using the PPC/NN retrieval method with AIRS/AMSU data in partially cloudy areas. PPC/NN retrievals with partially cloudy AIRS/AMSU data over land were also performed. The PPC/NN retrieval technique was shown to be relatively insensitive to cloud amount, sensor scan angle, orbit type, and training set comprehensiveness. These results further suggest that the AIRS Level 2 algorithm used to produce the cloud-cleared radiances and quality flags used by the PPC/NN retrieval is performing well.

The high level of performance achieved by the PPC/NN algorithm suggests it would be a suitable candidate for the retrieval of geophysical parameters other than temperature and moisture from high resolution spectral data. Potential applications include the retrieval of ozone profiles and trace gas amounts. Future work will involve further evaluation of the algorithm with simulated and observed partially cloudy data, including global radiosonde data and ground- and aircraft-based observations.

References

[1] W. J. Blackwell. "A neural-network technique for the retrieval of atmospheric temperature and moisture profiles from high spectral resolution sounding data." *IEEE Trans. Geosci. Remote Sens.*, 43(11):2535–2546, November 2005.

[2] J. Escobar-Munoz, A. Chedin, F. Cheruy, and N. Scott. "Reseaux de neurons multi-couches pour la restitution de variables thermodynamiques atmosphériques à l'aide de sondeurs verticauz satellitairs." *Comptes-Rendus de L'Academie Des Sciences; Série II*, 317(7):911–918, 1993.

[3] H. E. Motteler, L. L. Strow, L. McMillin, and J. A. Gualtieri. "Comparison of neural networks and regression based methods for temperature retrievals." *Appl. Opt.*, 34(24):5390–5397, August 1995.

[4] F. Aires, A. Chédin, N. A. Scott, and W. B. Rossow. "A regularized neural net approach for retrieval of atmospheric and surface temperatures with the IASI instrument." *J. Appl. Meteorol.*, 41:144–159, February 2002.

[5] F. Aires, W. B. Rossow, N. A. Scott, and A. Chédin. "Remote sensing from the infrared atmospheric sounding interferometer instrument: 2. simultaneous retrieval of temperature, water vapor, and ozone atmospheric profiles." *J. Geophys. Res.*, 107, November 2002.

[6] W. J. Blackwell. "Validation of neural network atmospheric temperature and moisture retrievals using AIRS/AMSU radiances." *Proceedings of the SPIE Defense and Security Symposium*, Orlando, 5806, October 2005.

[7] W. J. Blackwell, F. W. Chen, L. Jairam, and M. Pieper. "Neural network estimation of atmospheric profiles using AIRS/IASI/AMSU data in the presence of clouds." *IEEE International Geoscience and Remote Sensing Symposium Proceedings*, July 2008.

[8] J. Hadji-Lazaro, C. Clerbaux, and S. Thiria. "An inversion algorithm using neural networks to retrieve atmospheric CO total columns from high resolution nadir radiances." *J. of Geophys. Res.*, 104:23841–23854, 1999.

[9] F. Del Frate and G. Schiavon. "A combined natural orthogonal functions/neural network technique for the radiometric estimation of atmospheric profiles." *Radio Sci.*, 33(2):405–410, March 1998.

[10] D. Nguyen and B. Widrow. "Improving the learning speed of two-layer neural networks by choosing initial values of the adaptive weights." *IJCNN*, 3:21–26, 1990.

[11] H. H. Aumann, et al. "AIRS/AMSU/HSB on the Aqua mission: Design, science objectives, data products, and processing systems." *IEEE Trans. Geosci. Remote Sens.*, 41(2):253–264, February 2003.

[12] L. Strow, S. Hannon, and S. Desouza-Machado. "An overview of the AIRS radiative transfer model." *IEEE Trans. Geosci. Remote Sens.*, 41(2), February 2003.

[13] J. R. Eyre. "Inversion of cloudy satellite sounding radiances by nonlinear optimal estimation. I: Theory and simulation for TOVS." *Q. J. R. Meteorol. Soc.*, 115:1001–1026, July 1989.

[14] J. Susskind, C. D. Barnet, and J. M. Blaisdell. "Retrieval of atmospheric and surface parameters from AIRS/AMSU/HSB data in the presence of clouds." *IEEE Trans. Geosci. Remote Sens.*, 41(2):390–409, February 2003.

[15] J. Susskind, et al. "Accuracy of geophysical parameters derived from AIRS/AMSU as a function of fractional cloud cover." *J. Geophys. Res.*, 111, 2006.

[16] H. H. Aumann, et al. "Validation of AIRS/AMSU/HSB core products for data release version 3.0." *NASA JPL Tech. Rep. D-26538*, August 2003.

[17] W. J. Blackwell, et al. "Retrieval of atmospheric temperature and moisture profiles from cloudy infrared and microwave sounding data using stochastic methods." *IEEE Trans. Geosci. Remote Sens.*, 2009, submitted for publication.

[18] W. J. Blackwell and F. W. Chen. "Recent progress in neural network estimation of atmospheric profiles using microwave and hyperspectral infrared sounding data in the presence of clouds." *Proceedings of the SPIE Defense and Security Symposium*, Orlando, 6966, October 2008.

[19] W. J. Blackwell, F. W. Chen, and L. Jairam. "Combined microwave and hyperspectral infrared retrievals of atmospheric profiles in the presence of clouds using nonlinear stochastic methods." *IEEE International Geoscience and Remote Sensing Symposium Proceedings*, August 2007.

11

Discussion of Future Work

We now cast an eye to the future and discuss several neural network techniques and applications that we feel will receive considerable attention in the coming years. We begin by presenting Bayesian approaches for neural network training and error characterization that offer theoretical advantages over traditional techniques, primarily through more direct use of the relevant conditional probability distribution functions. We then discuss techniques based on fuzzy set theory, where member elements may belong to several non-distinct feature sets, and the "degree of membership" can be assigned from a continuum of values. Finally, we examine the use of neural networks in cases that are temporally nonstationary. For example, climate studies of atmospheric temperature derived from infrared sounding observations must consider the rising level of carbon dioxide in the atmosphere because the absorption features most often used in the thermal infrared for remote sounding of temperature occur near carbon dioxide lines.

11.1 Bayesian Approaches for Neural Network Training and Error Characterization

The framework provided by Bayesian probability constructs offers several advantages for data modeling and parameter estimation. For example, Bayesian model comparisons can be used to optimize weight decay rates and to infer which of a set of neural network input variables are the most relevant [1]. Furthermore, parameter uncertainty can be incorporated into the predictions to improve the accuracy and the characterization of errors [2].

The Bayesian approach to machine learning essentially involves the following four steps [3]:

1. Formulation of probabilistic knowledge by defining a model with unknown parameters and a prior probability distribution for these parameters;
2. Assembly of data set;
3. Computation of the posterior probability distribution for the parameters given the observed data;
4. Derivation of prediction by averaging over the posterior distribution.

Bayes rule allows the posterior probability to be expressed as follows:

$$P(\text{parameters}|\text{data}) \propto P(\text{parameters})P(\text{data}|\text{parameters}) \tag{11.1}$$

The predictions are then made by integrating with respect to the posterior [3]:

$$P(\text{new data}|\text{data}) = \int_{\text{parameters}} P(\text{new data}|\text{parameters})P(\text{parameters}|\text{data}) \tag{11.2}$$

The practical challenges of Bayesian neural network approaches usually stem from the lack of a prior distribution. There are generally two approaches that are used to compute the needed posterior distribution. Gaussian approximations applied near the mode of the distribution can work well when the amount of data is large relative to the complexity of the model. Markov chain Monte Carlo methods can be constructed that eventually converge to the posterior distribution in a wide variety of problems.

A number of studies have been conducted involving the application of Bayesian neural networks to remote sensing problems [4–7]. Of particular interest is the recent work of Aires [6] that demonstrated improved error characterization of neural network estimates of surface temperature and emissivity and water vapor. The current availability of large atmospheric databases provided by the AIRS/AMSU and IASI/AMSU sensor suites and the future availability of similar data provided by the CrIS/ATMS systems will facilitate further examination of Bayesian approaches. Furthermore, the ever-increasing computational capacity should enable more comprehensive and sophisticated probability sampling approaches to be used.

11.2 Soft Computing: Neuro-Fuzzy Systems

Unlike the traditional, hard computing, soft computing accommodates the imprecision of the real world by exploiting the tolerance for imprecision, uncertainty, and partial truth to achieve tractability, robustness, and simplicity. Neuro-fuzzy systems aim to combine the humanlike reasoning of fuzzy

systems [8] with neural networks by segmenting the input feature space into a number of overlapping sets. The degree to which a particular feature belongs to a particular set is mathematically expressed by a "membership function."

A basic fuzzy inference system comprises five functional components [9]:

1. a rule base containing a number of fuzzy if-then rules;
2. a database that defines the membership functions of the fuzzy sets used in the fuzzy rules;
3. a decision-making unit that performs the inference operations on the rules;
4. a fuzzification interface that transforms the crisp inputs into degrees of match with linguistic values;
5. a defuzzification interface that transforms the fuzzy results of the inference into a crisp output.

We see that the fuzzification and defuzzification functions could be cast as pre- and post-processing operations, respectively, and a neural network could be trained to perform the inference. Such a hybrid approach is sometimes called a neuro-fuzzy system (NFS). Jang introduced the Adaptive Neuro-Fuzzy Inference System [9] (ANFIS) that can applied to a wide range of regression and classification problems.

NFS methodology could be used in a number of atmospheric remote sensing scenarios. For example, it might be useful to categorize precipitation type into a number of fuzzy sets prior to estimation by the neural network. This preclassification could help improve performance in a number of problematic cases, including stratiform and orographic rain.

11.3 Nonstationarity Considerations: Neural Network Applications for Climate Studies

We conclude the book with an examination of a problem we have thus far largely neglected, statistical nonstationarity of the relationship between observed radiances and the geophysical parameters to be estimated. The probability distribution function of a stationary random process does not change when the process is shifted in time or space. We have alluded to atmospheric processes that may depend on geographical location or season. Also problematic is the fact that the atmospheric composition is changing over time. For example, carbon dioxide levels in the Earth's atmosphere have increased by approximately 2 parts per million per year over the last 10 years or so. Therefore, a statistical estimator trained to estimate atmospheric temperature from atmospheric thermal emission measured near a carbon

dioxide absorption line in the thermal infrared will exhibit an estimation bias as the carbon dioxide levels rise over the years. There is thus a potential to confuse a temperature trend with a carbon dioxide trend.

There are a number of ways to mitigate nonstationarity when constructing a retrieval method. One approach is to construct multiple estimators sufficiently separated in time or space so that a trend can be identified and extrapolated in the future using a linear or nonlinear extrapolation operator. An alternate approach is to include the nonstationarity in the training data and allow the neural network to learn the trend. A third method could be to estimate the effect of the nonstationarity on the output and use a separate estimator to "correct" the trend.

These issues are extremely important in a climate context, because we usually desire to extract subtle trends over the course of many years. It is therefore paramount that any nonstationarities in the data are well understood so they are not confused with the climate trends.

References

[1] H. K. H. Lee. *Bayesian Nonparametrics Via Neural Networks*. American Statistical Association and the Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania, 2004.

[2] D. J. C. MacKay. "A practical Bayesian framework for backpropagation networks." *Neural Computation*, 4:448–472, 1992.

[3] R. M. Neal. *Bayesian Learning for Neural Networks*. Springer-Verlag, New York, 1996.

[4] H. H. Thodberg. "A review of Bayesian neural networks with an application to near infrared spectroscopy." *IEEE Trans. Neural Networks*, 7:56–72, 1996.

[5] F. Aires, C. Prigent, and W. B. Rossow. "Neural network uncertainty assessment using Bayesian statistics with application to remote sensing: 1. Network weights." *J. Geophys. Res.*, 109, May 2004.

[6] F. Aires, C. Prigent, and W. B. Rossow. "Neural network uncertainty assessment using Bayesian statistics with application to remote sensing: 2. Output errors." *J. Geophys. Res.*, 109, May 2004.

[7] F. Aires, C. Prigent, and W. B. Rossow. "Neural network uncertainty assessment using Bayesian statistics with application to remote sensing: 3. Network Jacobians." *J. Geophys. Res.*, 109, May 2004.

[8] L. A. Zadeh. "Fuzzy sets." *Information and Control*, 8(3):338–353, 1965.

[9] J. S. R. Jang. "ANFIS: Adaptive-network-based fuzzy inference systems." *IEEE Trans. Syst. Man Cybern.*, 23(3):665–685, May 1993.

About the Authors

William J. Blackwell is a senior staff member in the Sensor Technology and System Applications group of MIT Lincoln Laboratory in Lexington, Massachusetts. He received an Sc.D. degree in electrical engineering and computer science from MIT, where he was a National Science Foundation graduate fellow. Dr. Blackwell serves on the NASA AIRS/NPP science team and the NPOESS Sounding Operational Algorithm Team and is the NPOESS Sensor Scientist for the Advanced Technology Microwave Sounder. He has authored over 40 papers on sensor and algorithm technology for atmospheric remote sensing and is a senior member of the IEEE. He received the NOAA David Johnson Award for outstanding innovative use of Earth-observing satellite data in 2009.

Frederick W. Chen is a senior engineer at Signal Systems Corporation in Severna Park, Maryland. He received a Ph.D. degree in electrical engineering and computer science from MIT and has published extensively in the fields of signal processing, estimation, and neural networks. Dr. Chen has previously held positions at Argonne National Laboratory and MIT Lincoln Laboratory.

Index