APPLICATION OF THE RELEVANCE VECTOR MACHINE TO CANAL FLOW PREDICTION IN THE SEVIER RIVER BASIN

by

John T. Flake

A thesis submitted in partial fulfillment
of the requirements for the degree

of

MASTER OF SCIENCE

in

Electrical Engineering

Approved:

Dr. Todd K. Moon Major Professor	Dr. Jacob H. Gunther Committee Member
Dr. YangQuan Chen Committee Member	Dr. Byron R. Burnham Dean of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2007

Abstract

Application of the Relevance Vector Machine to Canal Flow Prediction
in the Sevier River Basin

by

John T. Flake, Master of Science
Utah State University, 2007

Major Professor: Dr. Todd K. Moon
Department: Electrical and Computer Engineering

This work addresses management of the scarce water resource for irrigation in arid regions where significant delays between the time of order and the time of delivery present major difficulties. Motivated by improvements to water management that will be facilitated by an ability to predict water demand, this work employs a data-driven approach to developing canal flow prediction models using the Relevance Vector Machine (RVM), a probabilistic kernel-based learning machine. Beyond the RVM learning process, which establishes the set of relevant vectors from the training data, a search is performed across model attributes including input set, kernel scale parameter, and model update scheme for models providing superior prediction capability. Models are developed for two canals in the Sevier River Basin of southern Utah for prediction horizons of up to five days. Appendices provide the RVM derivation in detail.

(107 pages)

To my father, James T. Flake, who teaches me to zag when all I know is to zig.

Acknowledgments

At the conclusion of any large work one must reflect on the means that delivered the worker to its conclusion. Rarely does this reflection fail to reveal many whose contribution was invaluable and who are deserving of thanks. These include Dr. Todd K. Moon whose enthusiasm was a powerful ingredient that I myself could not always evince; my officemates Nisha, Roger, Daren, and Jake who cheerfully supported me through my many vocalizations of frustration; and many friends and family whose interest and support prompted them to query (sometimes to my chagrin) as to my progress. The latter group includes some of special note: I thank my father, James, who was persistent in encouraging me to simplify and not to complicate, my mother, Deanne, for her consistent mothering spirit, my friend, Stephanie, for often caring more about what I was saying than I did, Lee Burke for encouraging me not to become an ABD, and Richard Gordon for being serviceable when my soul was tired and tried. Most notable, I am grateful for help from above with the comforting promise given that "as thy days may demand so thy succor shall be."

John Flake

Contents

List of Tables

List of Figures

Figure Page

Chapter 1

Introduction and Background

Over the course of the last several decades there has been a large influx of people into many of the arid regions in the world. Historically, the water available in these areas has been managed at various levels of sophistication to meet a variety of water needs including, but not limited to, crop irrigation, drinking water, culinary water, and sanitation. For many of these arid regions, where water is already a scarce resource, the influx of people has provided a strain on the water management systems in meeting all of the water demands in the region as constrained by the limited resources available to the region, especially the limited water resource itself. This strain has prompted the implementation of a variety of management practices, and deployment of various items of infrastructure in an attempt to optimize the use of the limited resource to assist in the challenge of meeting the water demands in the regions. The influx is projected to continue in the decades to come, further strapping the capability of water management systems in meeting the increasing demands. To meet this challenge research is being conducted by a variety of institutions.

One of the biggest challenges in areas with limited water is getting the necessary amounts of water to the desired places at the appropriate times, with the ever-present objective of providing the water with minimal loss in transmission and minimal excess so as to maximize the water available for other water demands. Meeting this challenge is problematic when, as is often the case, the amounts of water needed, the locations and times of need, and the losses that will occur are not precisely known at the time when water management and diversion decisions are made. One important area of focus, then, is the development of models for predicting water demand. Such has become a focus for research on the Sevier River Basin.

The Sevier River Basin is a closed river basin in south central Utah covering approximately 12.5% of the state's area. Due to the arid climate of the region, irrigation is essential to crop growth and water is in high demand. Various efforts have been used to improve water management in the Sevier River Basin. A system of reservoirs and canals has been developed to meet the water needs in the basin with management of the water resource evolving over the years in answer to these changes [1]. More recently, in an attempt to improve water management practices in the basin, the canal system has been heavily instrumented for measurement and control purposes [2]. The instrumentation system includes measurement devices as well as communication hardware and software which collect hourly data for many points in the basin and log this data in a single Internet-accessible database. Measurements include water levels and flow rates as well as several weather indices collected at weather stations in the basin. This automated data collection has been ongoing since the year 2000. There are now roughly seven years of data for many measurement points within the basin [3]. This data, which is publicly accessible, has been used mainly for monitoring purposes, until recently, when some work has been done to use the data with statistical learning machines to predict reservoir releases [4]. This work has met with some success, prompting further interest in investigations of potential improvements to water management that may come as a result of an increased ability to predict water demands in the basin.

The work of this thesis is the investigation and development of canal flow prediction capability in the Sevier River Basin using the Relevance Vector Machine (RVM). The methods and tools used for prediction in the Sevier River Basin are expected to have application to other regions where water is in high demand.

We continue this chapter with a more detailed discussion of the situation in the Sevier River Basin, explaining how the available data lends itself to a learning machine approach. In Chapter 2 we give the basic theoretical background for the RVM, our learning machine of choice, followed by a discussion in Chapter 3 of some concepts in learning and how these relate to RVM theory and mechanics. We describe the application of the RVM to canal prediction in the Sevier River Basin and discuss our results in Chapter 4. We conclude in Chapter 5 by discussing how this work can be carried forward.

Many challenges confront the water users in the Sevier River Basin. Depending on their location in the basin, farmers must place water orders as many as five days in advance of the time they can expect to receive it. A large portion of available water is lost in transmission from reservoir to field. For example, in the Richfield Canal a 42% increase is made to water orders to account for anticipated water losses. The mechanism for water delivery is relatively inflexible; delivery times are rigid and order cancellation is generally not an option. These and other issues necessitate careful management of the limited water resource.

At present, canal operators make flow decisions based on farmer orders, transmission loss rates, canal limitations and experience. To provide for the water needs of the farmers, operators are dependent on the receipt of water orders. When setting the flow to meet current orders the canal operator has little knowledge of future orders, nor, therefore, of the corresponding future flow. Several improvements to water management might be possible if canal operators and water managers did have more knowledge of future orders with which to make canal flow decisions. For example, it is known that transmission loss can be reduced by minimizing the fluctuation of water level in a canal. However, to best minimize fluctuation the operator must know something about future orders so as to smooth out the transition from current flow to future flow. In order to enable this improvement (and others) it would seem that future orders need to be predicted. This is not trivial. Individual water orders come as a result of crop need as assessed by farmers based on individual fields with potentially different crops. Crop water need by itself has been relatively well modeled [5]. On the other hand, farmer assessment of the same is less deterministic. The orders, both as to amount and timing, also depend on a number of other complicated factors, some of which might include the remaining amount of water to which the farmer has rights, the preparation of the farmer to receive needed water (i.e. manpower), and the intuition of the farmer as to upcoming weather, including, on occasion, precipitation. Market is another issue that affects orders both as to which crops to plant and as to which of the planted crops the most attention should be given. These types of information, even if available, would be difficult to interpret into a mathematically representable form. Rather than attempting to predict individual orders, one might consider predicting the orders in sum, as the sum of orders basically determines canal flow. Intuitively, this might average out the unpredictable behavior of farmers and make for a prediction model that would be more closely tied to crops, weather and other physical quantities. However, predictions must rely on data that is consistently and readily available, particularly for a system that intends to provide automated predictions at regular time steps. The primary source for consistent data in the basin is the aforementioned database. However, the available data is limited to reservoir levels, canal flow rates, and weather data including temperature, humidity, solar radiation, wind speed, and precipitation. Since other data—such as crop type and acreage—is not readily available, a physical model is not the most likely approach. Instead, a data driven model could be considered.

The main consideration for a data driven model is how best to use the available data. We seek a functional relationship between a set of inputs and an output, where the output is the item we desire to predict and all inputs and the output are contained within the available data. While we have spoken of water orders as the quantity we would like to predict, orders are not one of the data items in the database, nor are they readily available otherwise. Instead we will choose the canal flow itself as the item of prediction. This choice fills the same role as water orders and is arguably a better choice. We justify this as follows: In setting canal flow, individual farmer orders are combined additively to form a total water order. Expected water loss is accounted for with a multiplicative factor. Some modifications are likely made by the canal operator based on his strategies for respecting canal limitations, maintenance needs, and other objectives. These result in a quantity that can be thought of as the intended canal flow. The actual canal flow differs from this intended flow only by limitations in the precision of the operator at meeting his intentions. Such control limitations are a matter of the tools at the disposal of the operator for setting canal flow; they can be modeled as noise. Finally, the measured canal flow—which is the data item available in the database—is the actual canal flow with noise introduced through measurement. The measured canal flow, then, is the inclusion of control and measurement noise on an intended flow that is meant to meet the water orders given by the farmers. For purposes of setting canal flow we can predict this intended flow directly, which is equivalent to predicting water orders and then determining the intended flow from the orders. The direct approach eliminates computations while suiting itself to the available data.

If our description above is accurate, intended flow, which we will hereafter call demand, is directly related to the water orders placed by farmers and is generated to match those orders by taking into account the losses associated with transmission while remaining within the bounds of operation for the canal. The type of inputs that would be used to predict farmer orders are generally the same inputs that will be effective in predicting demand (intended flow).

We choose the RVM as our tool for prediction. The Relevance Vector Machine (RVM) is a kernel-based learning machine that has the same functional form as the popular Support Vector Machine (SVM). Its form is a linear combination of data-centered basis functions that are generally nonlinear. The RVM has been shown to provide equivalent and often superior results to the SVM both in generalization ability and sparseness of the model.

Having chosen the RVM and given the data items available in the database, forming a model for prediction is a matter of experimenting with the choice of inputs to find the set of inputs that provide the best functional description of the output, that is, the set of inputs that produce a model with the lowest prediction error. This process is relatively intuitive but requires some experimentation. It also requires some understanding of the physical system. After giving the theoretical background of the RVM in Chapter 2 and discussing in Chapter 3 how to utilize the RVM in light of some basic concepts in learning, we will describe in Chapter 4 much of our experimentation for investigating the most appropriate inputs to the system.

Chapter 2

Predictive Function Estimation and the Relevance Vector Machine

The following chapter borrows significantly from one of the original expositions on the Relevance Vector Machine (RVM). In particular, the organization of the chapter, many of its general ideas, its notation, vocabulary, and in a few cases small pieces of phraseology come from [6]. This being said, no additional explicit citations to this source are made except as reference in the case of significant ideas that are not borne out by the presentation of this chapter.

Prediction is the deduction or estimation of a system condition based on some functional or intuitive relationship between that condition and other conditions in the system. Without both the knowledge of other conditions in the system and the existence of some relationship of these to the desired condition there are no grounds for prediction. Prediction, then, requires observations of conditions in the system that are functionally related to the condition to be predicted as well as knowledge of the functional relationship. Often the true challenge to prediction is in determining the functional relationship.

The task of machine learning is to determine or estimate this functional relationship between the desired condition and other conditions in the system from a set of paired examples or observations of the same. In other words, if we call the value of the desired condition a *target*, and denote it t_n, and call the vector value of the system conditions that yield the target an *input*, and label it $\mathbf{x}_n$, then the task of machine learning is to estimate the functional relationship that relates inputs $\mathbf{x}_n$ to their associated targets t_n using a finite set of examples of the same, $\{t_n, \mathbf{x}_n\}_{n=1}^{N}$, hereafter referred to as the training data.

While theoretically we seek to elucidate a function or model $y(\mathbf{x})$ from the set of all possible functions, to be practical the problem is often reduced to finding a function of the form $y(\mathbf{x}, \mathbf{w}) = \sum_{i=1}^{M} w_i \phi_i(\mathbf{x})$, which is the linearly-weighted sum of M fixed basis functions $\phi_i(\mathbf{x})$. This form is both flexible in that it can describe many functional relationships, and easy to work with as it tends to problems that can be solved using linear algebra techniques due to its vector representation $y(\mathbf{x}, \mathbf{w}) = \mathbf{w}^{\mathrm{T}} \boldsymbol{\phi}(\mathbf{x})$, where $\mathbf{w} = (w_1, w_2, \ldots, w_M)^{\mathrm{T}}$, and $\boldsymbol{\phi}(\mathbf{x}) = (\phi_1(\mathbf{x}), \phi_2(\mathbf{x}), \ldots, \phi_M(\mathbf{x}))^{\mathrm{T}}$. The set of basis functions from which the system model can be chosen is still very large with the choice of basis set, $\boldsymbol{\phi}(\mathbf{x})$, unspecified. With our knowledge about the system limited to the set of examples, it is natural to suppose that a model for the system elucidated from the data would in some way utilize that data to form the model. In fact, an increasingly popular approach to machine learning is to select models of the form

$$y(\mathbf{x}, \mathbf{w}) = \sum_{i=1}^{N} w_i K(\mathbf{x}, \mathbf{x}_i) + w_0, \tag{2.1}$$

where the basis functions are now specified as kernel functions that incorporate the training inputs $\mathbf{x}_n$, with one kernel for each input from the data. Again, the function can be represented in vector form as $y(\mathbf{x}, \mathbf{w}) = \mathbf{w}^{\mathrm{T}} \boldsymbol{\phi}(\mathbf{x})$, only now where $\mathbf{w} = (w_0, w_1, \ldots, w_N)^{\mathrm{T}}$ and $\boldsymbol{\phi}(\mathbf{x}) = [1, K(\mathbf{x}, \mathbf{x}_1), K(\mathbf{x}, \mathbf{x}_2), \ldots, K(\mathbf{x}, \mathbf{x}_N)]^{\mathrm{T}}$. With such a form, estimating the model requires only the choice of a kernel function type for the model and determination of values for the linear weights. A learning machine utilizing a model of this form is known as a kernel-based learning machine, or simply, a kernel machine.

One such learning machine that is particularly well known is the Support Vector Machine (SVM). The performance of the SVM in machine learning tasks (classification or regression) is often used as a standard of comparison for the development and deployment of alternative learning machines. Another learning machine that shares the same functional form as the SVM is the Relevance Vector Machine (RVM). In several respects this learning machine, which is the subject of this chapter, has been shown to be comparable to if not better than the state-of-the-art SVM. The purpose of this chapter is to give some theoretical background for the RVM to enable later discussion of its merits and mechanics as it applies to the canal prediction problem.

The RVM is actually a specialization of a learning framework known as Sparse Bayesian

Learning. Founded in the Bayesian context, the RVM relies on the concept of marginalization to deal with unknown variables [7]. For the RVM this powerful concept facilitates estimation of a distribution for the output of a parameterized function for which parameter values are unknown.

Returning to the problem at hand, our purpose is to find good values for the model weights, that will generalize to unseen data so that predictions can be performed. By common practice the targets are modeled as the function on the inputs with additive white Gaussian noise,

$$t_n = y(\mathbf{x}_n, \mathbf{w}) + \epsilon_n, \tag{2.2}$$

where for our purposes the function $y(\mathbf{x}_n)$ is written as $y(\mathbf{x}_n, \mathbf{w})$ to explicitly denote its functional dependence on both the inputs and the weights. It should be noted that according to the formulation in (2.1) the function also depends on the choice of kernel. Also note the addition of noise to the model in (2.2) which accommodates measurement error on the targets. The implications are that each target is determined from the corresponding input independently of all other inputs (except, of course, in the sense that that the training inputs are used to form the basis or set of kernels in the model function) and that the noise is independent between samples. With this formulation—given that we know $y(\mathbf{x}_n)$—each target is independently distributed as Gaussian with mean $y(\mathbf{x}_n)$, and variance σ^2,

$$p(t_n|y(\mathbf{x}_n), \sigma^2) \sim \mathcal{N}(t_n|y(\mathbf{x}_n), \sigma^2), \tag{2.3}$$

where σ^2 is the variance of the noise process. For reasons of clarity we could alternatively denote this distribution by $p(t_n|\mathbf{w}, \boldsymbol{\phi}(\mathbf{x}_n), \sigma^2)$, where as described before $\boldsymbol{\phi}(\mathbf{x}_n) = [1, K(\mathbf{x}_n, \mathbf{x}_1), K(\mathbf{x}_n, \mathbf{x}_2), \ldots, K(\mathbf{x}_n, \mathbf{x}_N)]^{\mathrm{T}}$, which provides a means to show dependence on the kernel type in the notation. When forming the joint distribution over all the targets the vectors of kernel functions can be stacked in a matrix as $\mathbf{\Phi} = [\boldsymbol{\phi}(\mathbf{x}_1), \boldsymbol{\phi}(\mathbf{x}_2), \ldots, \boldsymbol{\phi}(\mathbf{x}_N)]^{\mathrm{T}}$ which gives a compact representation for the vector of model functions,

$$\mathbf{y} = [y(\mathbf{x}_1, \mathbf{w}), y(\mathbf{x}_2, \mathbf{w}), \ldots, y(\mathbf{x}_N, \mathbf{w})]^{\mathrm{T}} = \mathbf{\Phi}\mathbf{w},$$

and allows the joint conditional distribution over the targets to be written as

$$p(\mathbf{t}|\mathbf{y},\sigma^2) = p(\mathbf{t}|\mathbf{w},\mathbf{\Phi},\sigma^2) = (2\pi\sigma^2)^{-N/2}\exp\left\{-\frac{1}{2\sigma^2}\|\mathbf{t}-\mathbf{\Phi}\mathbf{w}\|^2\right\}, \tag{2.4}$$

where $\mathbf{t} = (t_1, t_2, \ldots, t_N)^{\mathrm{T}}$. Note that knowing $\mathbf{\Phi}$ is equivalent to knowing the kernel type and the set of inputs, where the inputs (from the training data) are utilized in the kernel functions both as the inputs associated with the training targets and as the kernel centers. After training the model—that is, after finding good values for $\mathbf{w}$—the inputs from the training set will remain part of the model as the centers of the kernel functions, but previously unseen inputs will now occupy the other position in the kernels.

We do not seek to probabilistically model the kernel type so that its determination is not part of the training optimization. Rather we treat the kernel type as known and fixed so that the distribution function need not be conditioned upon it in the sense of its being a random variable. We also choose to omit any indication of conditioning on the inputs, although this latter choice does not effect the problem but rather is done merely for brevity and convenience. Therefore, in line with these stipulations we drop the matrix of kernel functions, $\mathbf{\Phi}$, from our notation leaving the distribution notated as $p(\mathbf{t}|\mathbf{w},\sigma^2)$, which retains its equivalence to (2.4).

A seemingly natural thing to do at this point would be to recognize (2.4) as the likelihood function for the set of targets and seek to determine the maximum likelihood solution, that is, determine the values for $\mathbf{w}$ and σ^2 which maximize the function. However, we should remember that we are seeking values for the weights that are equally valid for the set of training data as well as for data that is not yet available (including data that may be withheld from the training set for validation purposes). Maximum likelihood estimation tends to produce values for the weights that are overspecialized to the training data, giving a model that does not generalize well to unseen data. To see why this is so, consider the squared norm in the exponent of the likelihood function, $\|\mathbf{t}-\mathbf{\Phi}\mathbf{w}\|^2$. It could alternately be written as $\|\mathbf{t}-\mathbf{y}\|^2$ where as we remember $\mathbf{t} = (t_1, t_2, \ldots, t_N)^{\mathrm{T}}$ and $\mathbf{y} = [y(\mathbf{x}_1,\mathbf{w}), y(\mathbf{x}_2,\mathbf{w}), \ldots, y(\mathbf{x}_N,\mathbf{w})]^{\mathrm{T}}$. Then it can be seen with reference to the noise model (2.2) that the norm can be written as $\|\boldsymbol{\epsilon}\|^2$, where $\boldsymbol{\epsilon}$ is the vector with elements ϵ_n, which is the noise on the targets, so that the squared norm is the sum of the squared noise values or the squared errors between model and target. Now assume a fixed variance and see that the exponential is maximized for small squared errors. This is the least squares solution. It favors values for the weights that minimize the difference between target and model. In other words, it favors a solution where the model closely 'fits' the targets. But, since there is no limitation to how complicated the function (2.1) can be, the solution selects as complicated a function as is necessary to provide the best (least squares) fit to the targets, without any consideration that a less complicated function, allowing for more noise, may generalize better to additional data. Essentially, the solution presumes little noise and relegates much of what is actually noise on the targets to instead be part of the system it is trying to model; it models the noise peculiarities of the training data, which being independent of other noise, makes for a model that cannot generalize.

A common approach to preventing this overspecialization is to provide some complexity control for the model function. When optimizing weight values using a cost or loss function, this is often accomplished through the inclusion of a penalty term than prevents models with large weight values from yielding the maximum (or minimum) of the cost function. This is effective because small weight values generally give less complicated functions. However, in the Bayesian perspective complexity control of the model function is addressed in a much different manner. Instead of using a penalty term, parameters are constrained by a prior distribution, which predisposes the weight parameters to take on small values. This can be accomplished using a normal distribution over each of the weights, that is centered at zero. This yields the joint distribution $p(\mathbf{w}|\alpha) = \prod_{i=1}^{N} \mathcal{N}(w_i|0, \alpha^{-1})$. Notice that this distribution includes an inverse variance term α which is shared for all weight distributions. The size of this term can be used to modify the strength of the predisposition for weights to take on small (near-zero) values, as it controls the width of the distribution of weight values around the mean of zero.

The RVM uses such a prior distribution, but with an additional feature that provides for one of the distinctive characterisitics of the RVM. For the RVM, the prior distribution over the weights is

$$p(\mathbf{w}|\boldsymbol{\alpha}) = \prod_{n=1}^{N} \mathcal{N}(w_i|0, \alpha_i^{-1}), \tag{2.5}$$

where the difference is the inclusion of an individual inverse variance parameter for each of the weights. This allows for an independent choice as to the strength of the predisposition for a weight value to be near zero. Further, each of the inverse variance parameters α_i as well as the model noise parameter σ^2, collectively called the hyperparameters, are given prior distributions known as hyperpriors, to provide a fully probabilistic specification for the weight prior and the noise model. The complete two-level prior is known as a hierarchal prior. To prevent unwarranted constraint to the values of the hyperparameters a hyperprior is chosen that is uniform over a logarithmic scale. The more general Gamma distribution is used for the hyperpriors in the original RVM specification as $p(\boldsymbol{\alpha}) = \prod_{i=0}^{N} \mathrm{Gamma}(\alpha_i|a, b)$ and $p(\beta) = \mathrm{Gamma}(\beta|c, d)$ where $\beta \equiv \sigma^{-2}$ and $\mathrm{Gamma}(\alpha|a, b) = \Gamma(a)^{-1} b^a \alpha^{a-1} e^{-b\alpha}$, but it is employed with parameters $a = b = c = d = 0$ to give hyperpriors that are uniform as described.

This construction of the prior provides for additional complexity control by inducing model sparseness. (Sparsity refers to a model where many of the weights are set precisely to zero.) With a uniform distribution over a logarithmic scale, the inverse variance parameters can take on any nonnegative value with equal *a priori* probability for each order of magnitude of the parameter. This provides for the possibility of very large (even infinite) inverse variance values α and correspondingly small (even zero-valued) variance values α^{-1}, which—if supported by the data—will yield weight distributions with all of the posterior probability mass concentrated at the mean value of zero. Zero-valued weights effectively remove the corresponding basis functions from the model leaving only those basis functions (a sparse set) that are formed from the 'relevant' training vectors. Hence, the name Relevance Vector Machine.

Having specified the prior distribution over the weights we must now determine good values for the weights by incorporating the knowledge available from the data. This concept, known as Bayesian inference, is accomplished through the use of Bayes theorem and marginalization. Ideally, we seek to determine the joint distribution of all the unknown parameters given the data (the posterior), which using Bayes' theorem is given by

$$p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t}) = \frac{p(\mathbf{t}|\mathbf{w}, \boldsymbol{\alpha}, \sigma^2)\, p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2)}{p(\mathbf{t})}, \tag{2.6}$$

so that we can then marginalize the joint distribution of all unknowns (including the target we are predicting t_*) over the parameters as

$$p(t_*|\mathbf{t}) = \int p(t_*, \mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t})\, d\mathbf{w}\, d\boldsymbol{\alpha}\, d\sigma^2 = \int p(t_*|\mathbf{w}, \sigma^2)\, p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t})\, d\mathbf{w}\, d\boldsymbol{\alpha}\, d\sigma^2 \tag{2.7}$$

to get a distribution for the new target t_*. However, we cannot compute the posterior $p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t})$ because we cannot perform the normalizing integral

$$p(\mathbf{t}) = \int p(\mathbf{t}|\mathbf{w}, \boldsymbol{\alpha}, \sigma^2)\, p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2)\, d\mathbf{w}\, d\boldsymbol{\alpha}\, d\sigma^2$$

in the denominator. Instead, an approximation for the posterior must be obtained. This proceeds by decomposing the posterior into two parts as

$$p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t}) = p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2)\, p(\boldsymbol{\alpha}, \sigma^2|\mathbf{t}), \tag{2.8}$$

where $p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2)$ is that portion of the posterior that can be computed exactly, leaving for approximation only the posterior over the hyperparameters, $p(\boldsymbol{\alpha}, \sigma^2|\mathbf{t})$. This hyperparameter posterior is replaced with a delta function at its mode $\delta(\boldsymbol{\alpha}_{\mathrm{MP}}, \sigma^2_{\mathrm{MP}})$, where $\boldsymbol{\alpha}_{\mathrm{MP}}$ and σ^2_{MP} are the most probable values of the hyperparameters. For reasons discussed by Tipping [6] this provides a good approximation. These most probable values are determined by maximizing $p(\boldsymbol{\alpha}, \sigma^2|\mathbf{t}) \propto p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2)p(\boldsymbol{\alpha})p(\sigma^2)$, which for uniform hyperpriors is equivalent to maximizing $p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2)$. The two distributions, therefore, that are required to approximate the joint distribution over the parameters in (2.8) are the posterior over the weights, $p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2)$, and what is known as the marginal likelihood, $p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2)$, for the $p(\boldsymbol{\alpha}, \sigma^2|\mathbf{t})$ approximation. These can be obtained together using Bayes' Theorem and by completing the square [6] (see Appendix A for the full derivation). With the inverse variance terms collected as $\mathbf{A} = \mathrm{diag}(\alpha_0, \alpha_1, \ldots, \alpha_N)$ this gives

$$p(\mathbf{w}|\mathbf{t}\,\boldsymbol{\alpha}, \sigma^2) = (2\pi)^{-(N+1)/2}|\boldsymbol{\Sigma}|^{-1/2} \exp\left\{-\frac{1}{2}(\mathbf{w}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{w}-\boldsymbol{\mu})\right\} \tag{2.9}$$

for the posterior, with covariance $\boldsymbol{\Sigma} = (\sigma^{-2}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi} + \mathbf{A})^{-1}$ and mean $\boldsymbol{\mu} = \sigma^{-2}\boldsymbol{\Sigma}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}$, and

$$p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2) = (2\pi)^{-N/2}|\sigma^2\mathbf{I} + \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathrm{T}}|^{-1/2} \exp\left\{-\frac{1}{2}\mathbf{t}^{\mathrm{T}}(\sigma^2\mathbf{I} + \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathrm{T}})^{-1}\mathbf{t}\right\} \tag{2.10}$$

for the marginal likelihood. It should be noted that while both distributions are obtained in a short-cut fashion by completing the square in a product of known distributions, computation really relies on the concept of marginalization which is the key to Bayesian inference [7].

The maximization of (2.10) to obtain the most probable hyperparameter values cannot be computed in closed form. Instead, the marginal likelihood is maximized by an iterative re-estimation of the hyperparameters. This can be arranged using derivatives or through application of the expectation-maximization (EM) algorithm. Both methods lead to the same update equations for the hyperparameters, however, a small modification of the derivative results leads to update equations that provide much faster convergence. Specifically, the derivatives are

$$\frac{\partial \mathcal{L}}{\partial \log \alpha_i} = \frac{1}{2}\left[1 - \alpha_i\left(\mu_i^2 + \Sigma_{ii}\right)\right] \tag{2.11}$$

and

$$\frac{\partial \mathcal{L}}{\partial \log \beta} = \frac{1}{2}\left[N - \beta \mathrm{tr}\left(\boldsymbol{\Sigma}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right) - \beta\left\|\mathbf{t} - \boldsymbol{\Phi}\boldsymbol{\mu}\right\|^2\right] \tag{2.12}$$

where $\mathcal{L}$ is the objective function formed from (2.10), as further detailed in Appendix A. Equating these results to zero leads to the updates

$$\alpha_i^{\mathrm{new}} = \frac{1}{\mu_i^2 + \Sigma_{ii}} \tag{2.13}$$

and

$$(\sigma^2)^{\text{new}} = \frac{\|\mathbf{t} - \mathbf{\Phi}\boldsymbol{\mu}\|^2 + \text{tr}\left(\mathbf{\Sigma}\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)}{N} \tag{2.14}$$

which are equivalent to updates obtained using the EM algorithm. The faster converging alternative is obtained using a modification suggested by MacKay [8] in which the quantities $\gamma_i \equiv 1 - \alpha_i \Sigma_{ii}$ are defined, each of which can be interpreted as a measure of how well the corresponding weight parameter is determined by the data. Substituting these quantities into the derivatives—directly into the first derivative, and into the second by a rewriting of the quantity $\beta\text{tr}\left(\mathbf{\Sigma}\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)$ as $\sum_i \gamma_i$ —leads to the updates

$$\alpha_i^{\text{new}} = \frac{\gamma_i}{\mu_i^2} \tag{2.15}$$

and

$$(\sigma^2)^{\text{new}} = \frac{\|\mathbf{t} - \mathbf{\Phi}\boldsymbol{\mu}\|^2}{N - \sum_i \gamma_i}. \tag{2.16}$$

The values for the hyperparameters, then, are determined by iterating alternate updates of the hyperparameters $\boldsymbol{\alpha}$, σ^2 and the statistics $\mathbf{\Sigma}$, $\boldsymbol{\mu}$ until convergence. In this process sparsity is realized as many of the α_i tend to infinity.

After determining values for $\boldsymbol{\alpha}_{\text{MP}}$ and σ^2_{MP} we can proceed with prediction as in (2.7) by replacing the posterior over all unknowns by its approximation $p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}_{\text{MP}}, \sigma^2_{\text{MP}})$, the posterior over the weights conditioned on the maximizing values of the hyperparameters. This gives the predictive distribution

$$p(t_*|\mathbf{t}, \boldsymbol{\alpha}_{\text{MP}}, \sigma^2_{\text{MP}}) = \int p(t_*, \mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}_{\text{MP}}, \sigma^2_{\text{MP}})\, d\mathbf{w} = \int p(t_*|\mathbf{w}, \sigma^2_{\text{MP}})\, p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}_{\text{MP}}, \sigma^2_{\text{MP}})\, d\mathbf{w} \tag{2.17}$$

which as a convolution of Guassians is Gaussian:

$$p(t_*|\mathbf{t}, \boldsymbol{\alpha}_{\text{MP}}, \sigma^2_{\text{MP}}) = \mathcal{N}(t_*|y_*, \sigma^2_*)$$

with mean $y_* = \boldsymbol{\mu}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_*)$ and variance $\sigma^2_* = \sigma^2_{\text{MP}} + \boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\mathbf{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*)$. As such, we choose as our predicted value for t_* the mean of the predictive distribution, which is nothing more than the sum of the basis functions weighted by the mean of the posterior weight distribution. Therefore the mean (or mode) of the weight distribution is the set of "good" values for the weights which, in linear combination with the set of basis functions, forms the estimate of the functional relationship which we seek between inputs and targets and from which we can predict the value of targets using as-yet-unseen system inputs from the same system. This posterior weight distribution, is determined by incorporating the training data, through the use of Bayesian inference, into the well constructed sparseness-favoring prior.

As a method of validating the use of the RVM for prediction we provide another model as a baseline against which to reference RVM prediction results. Instead of forming a model $y(\mathbf{x}, \mathbf{w}) = \sum_{i=1}^{M} w_i \phi_i(\mathbf{x})$ which is the linear combination of basis functions $\phi_i(\mathbf{x})$ (which are functions of the input vector $\mathbf{x}$) we form a much less sophisticated model

$$y(\mathbf{x}, \mathbf{a}) = \sum_{i=1}^{P} a_i x_i = \mathbf{a}^{\mathrm{T}}\mathbf{x},$$

which is a linear combination of the elements x_i of the input vector, where the number of elements forming the input vector is P so that $\mathbf{x} = (x_1, x_2, \ldots, x_P)^{\mathrm{T}}$. We can model each of the training targets as the function on the corresponding training input with an added noise component e_n to give

$$t_n = y(\mathbf{x_n}, \mathbf{a}) + e_n = \sum_{i=1}^{P} a_i x_{n,i} + e_n.$$

Having already defined $\mathbf{x}_n = (x_{n,1}, x_{n,2}, \ldots, x_{n,P})^{\mathrm{T}}$ to be the nth training vector input let us choose another notation $\mathbf{z}_i = (x_{1,i}, x_{2,i}, \ldots, x_{N,i})^{\mathrm{T}}$ to represent the vector containing the ith element of each of the N input vectors so we can stack the set of target equations to get

$$\mathbf{t} = \sum_{i=1}^{P} a_i \mathbf{z}_i + \mathbf{e},$$

where $\mathbf{e} = (e_1, e_2, ..., e_n)^{\mathrm{T}}$ is the error vector. To determine values for the weights a_i we constrain the error to be orthogonal to the data, which fixes the weights at values which minimize the squared error. This proceeds by first solving for the error vector which gives

$$\mathbf{e} = \mathbf{t} - \sum_{i=1}^{P} a_i \mathbf{z}_i.$$

Then we make the orthogonality constraint by setting the inner product of the error vector with each of the data vectors, $\mathbf{z_i}$, to zero:

$$\langle \mathbf{t} - \mathbf{e}, \mathbf{z_j} \rangle = \left\langle \mathbf{t} - \sum_{i=1}^{P} a_i \mathbf{z}_i, \mathbf{z}_j \right\rangle = 0 \quad \text{for } j = 1, 2, \ldots, P.$$

Some manipulation leads to

$$\sum_{i=1}^{P} a_i \langle \mathbf{z}_i, \mathbf{z}_j \rangle = \langle \mathbf{t}, \mathbf{z}_j \rangle \quad \text{for } j = 1, 2, \ldots, P$$

which can be written as the vector product

$$\mathbf{r}_j^{\mathrm{T}} \mathbf{a} = p_j \;\; \text{for } j = 1, 2, \ldots, P$$

where $\mathbf{r}_j = [\langle \mathbf{z}_1, \mathbf{z}_j \rangle, \langle \mathbf{z}_2, \mathbf{z}_j \rangle, \ldots, \langle \mathbf{z}_P, \mathbf{z}_j \rangle]^{\mathrm{T}}$ and $p_j = \langle \mathbf{t}, \mathbf{z}_j \rangle$. Then stacking the equations gives

$$\mathbf{Ra} = \mathbf{p}$$

where $\mathbf{R} = [\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_P]^{\mathrm{T}}$ is the invertible Grammian matrix and $\mathbf{p} = (p_1, p_2, \ldots, p_P)^{\mathrm{T}}$ is the correlation vector. Thus the weight values are given by

$$\mathbf{a} = \mathbf{R}^{-1}\mathbf{p},$$

which completes the reference model.

Chapter 3

Learning Concepts and RVM Mechanics

Through sufficient understanding of the RVM, general learning concepts can be connected to the theory and mathematics defining the RVM. A few of these ideas and their connections with learning machines in general and the RVM in particular will be discussed before the specific design choices for the developed models are introduced.

The relevance vector machine produces a function which is comprised of a set of kernel (basis) functions and a set of weights. The function represents a model for the system that was presented to the learning process through a set of training data samples. The kernels and weights yielded by the learning process are fixed and as such the model function defined by the weighted sum of kernels is fixed. As a fixed model it represents a system that is stationary. Strictly speaking, the RVM can only be used to model stationary systems. However, nonstationary systems can be treated as stationary systems across a limited time span. That is, even a system that is changing over time (which normally would not be represented by a model that lacks a facility for adaptation) can be modeled by a fixed model for a small period of time in which the change to the system is sufficiently small. For this reason, the RVM can be utilized to model a nonstationary system over a small timespan. For a nonstationary system over a larger timespan the system model must be updated at appropriate intervals to continue to provide an accurate estimate of the system.

As previously discussed, the RVM forms a model for the system using a finite set of input-output pair samples or vectors from the system. Further, from this set of training vectors the RVM selects a sparse subset of input vectors which are deemed ‘relevant’ by the probabilistic learning scheme for building a function that estimates the output of the system from the inputs. These relevant vectors are used to form the basis functions which in linear combination comprise the model function. In a sense, the degree of relevance of each of these remaining vectors is determined by the size of the weighting given to the corresponding kernel (relative to the other weights) in the linear combination. Considering a nonstationary system, it is not hard to see that a model formed from the relevant input-output samples of the past would not readily produce accurate outputs for a system that is no longer well represented by past samples. To operate with the RVM in a nonstationary system it becomes necessary to be selective as to the set of training data that are presented to the learning machine so as to control the timespan of data from which the model is formed. It is advisable to consider observable changes in the statistics of the physical system both as to inputs and outputs when selecting an appropriate timespan of training data. While the RVM can handle multi-modal outputs, if the current mode can be determined independently from the training process the user is arguably better off obtaining outputs from a model trained only with data representing the current mode. Otherwise relevant vectors are selected from both modes, and the model is likely to be less sparse.

In the original RVM development, the training samples have no indicated time relationship, nor any other discriminating feature that can be used for vector preference, aside from the values of the vectors themselves. In fact, before incorporating the data, the prior probabilities for all weights are identically distributed. Therefore, *a priori*, each vector is an equal candidate with all others for relevant status. Granted, it is the content of the vectors that—through the comparative assessment implemented by the learning process—determines which vectors will be relevant, however, for a given vector, before incorporating all other vectors (the data), no knowledge as to relevance (the size of the weight) is available. So, speaking from the pre-learning vantage point, each vector has equal probability of being relevant (having a nonzero weight). It follows that when applying the RVM to time series, there is no preference for more recent data except as that data may establish itself through the learning process as the more relevant data in defining a model for the system. Again, this prompts care in selecting the timespan of data that best represents the system to be modeled, recognizing that regardless of the time relationship of a vector to the current time it will be treated the same as all other training vectors included in the training set.

In the RVM development shown in Chapter 3 the form of the kernel functions was left unspecified except that the notation $K(\mathbf{x}, \mathbf{x}_n)$ indicates that each kernel is a function both of the current input vector $\mathbf{x}$ and one of the training input vectors $\mathbf{x}_n$. Aside from this, the form of the kernel functions is arbitrary. However, in general, the purpose of a kernel function is to provide some measure of similarity between the current input and the training input of the kernel so as to moderate the contribution (to the model) of the kernel when the current input is dissimilar to the training input of the kernel. One of the most popular kernel functions and the one used in our implementation is the Gaussian kernel which has form $K(\mathbf{x}, \mathbf{x}_n) = \exp\left\{-\eta\|\mathbf{x} - \mathbf{x}_n\|^2\right\}$. The squared norm of the difference establishes a measure of how different the two vectors are. A vector that is very different than the training vector will give a large squared norm; while a vector that is very similar will give a small squared norm. With the negative of the squared norm inserted into the exponent, the kernel function becomes a measure of similarity, that is, a vector that is similar to the training vector, which gives a small squared norm in the exponent, will give a value for the kernel that is nearly one, while a vector that is dissimilar with large squared norm will give a value for the kernel that is nearly zero. In this way, a very similar vector causes the additive contribution of a large portion of the kernel weight in the output of the model function and a very dissimilar vector causes the contribution of only a small portion of the kernel weight. The magnitude of each of the additive training-vector constituents of the model function is specified based on the similarity of the input vector to the respective training vector.

In general, the quality of the model produced by any learning process for the functional relationship between the inputs of a system and its outputs is limited by the set of inputs available to the learning process. The mathematical function produced by the learning process is a data-based estimate of some true physical function that perfectly describes a quantity, the output, in light of another set of quantities, the inputs. In the practical world, many complicated systems are described by simpler functions that approximate the true underlying physical relationship between outputs and inputs. Under conditions where an approximation has an acceptable level of accuracy, the approximation may be used to simplify a computation, minimize data gathering requirements, or provide some other advantage that warrants the approximation. In other cases an approximation may be used when the true physical relationship is unknown or cannot be determined or when some of the inputs to a known functional relationship cannot be measured so that the inputs themselves must be approximated or the function must be modified to eliminate the unavailable quantities as necessary inputs. In all such cases the approximate function will only come so far toward modeling the output. The difference between the output of the approximation and the true output of the physical system is the error of the approximation. A data-based estimate of a functional relationship is an approximation that is often used when a true physical relationship is too complicated to determine and/or when the relative appropriateness of the available data as inputs is in question. While such an approximation is much further removed from the physical sense of the system—that is, the additive or multiplicative relationships between inputs, relative proportionalities, or other such mathematical concepts that we are used to thinking of as connecting inputs in an intuitive fashion based on their physical relationships—the data-based model is an estimate of the physical system, and as such must have available to it those physical quantities upon which the output truly depends, in order to produce a good estimate of the system. When we exclude from the learning process an input that contains information about the output we limit the accuracy of the resulting model, that is, we increase the error of the approximation. The quality of the model produced by the learning process, then, is limited by the set of inputs available to the learning process. The question to ask when attempting to learn a data-based functional model is whether all of the important inputs to the system are available in the data set and how important the missing inputs are, in a physically intuitive sense, to describing the output.

To summarize we have discussed the following:

- Strictly speaking, the RVM can only be used to model a stationary system.
- If modeling a nonstationary system the system model determined from a training data set is only valid for timespans that are well represented by the data samples in the

data set.

- To model a nonstationary system over large timespans the model must be updated at appropriate intervals.

- Before the learning process (*a priori*), all training vectors have equal probability of being a member of the set of relevant basis functions. Time relationship has no bearing on vector relevance.

- The purpose of a kernel function is to provide a measure of similarity between the current input vector and the kernel's training vector so as to moderate the size of the kernel's inclusion in the model for dissimilar vectors.

- The learned model is an estimate of an actual functional relationship between inputs and outputs. A good model will be built from the set of all inputs upon which the output actually depends. Any depletion to the input set lowers the quality (increases the error) of the model.

Chapter 4

Application to Canal Flows in the Sevier River Basin

In this chapter we begin by discussing the various inputs that were used in the application of the RVM to prediction of canal flows in the Sevier River Basin. These inputs were derived from the database of measurements taken from various points in the Sevier River canal system. We discuss each of the inputs and the merits for their inclusion in a model for canal flow prediction. We do this by first introducing one of the canals in the Sevier River canal system.

The Richfield Canal is one of the largest canals in the Sevier River Basin. It is a diversion from the Sevier River starting a little south and west of Elsinore, Utah. The canal flows generally northeast until turning eastward to parallel the southern edge of Elsinore, then it makes a large arching turn to the north toward the town of Richfield, passes to the east of the town and continues to the northeast to where most of the acreage irrigated by the canal is located. Historical flow data for this canal is represented in fig. 4.1. The data, which is available in the SRWUA database, indicates significant periods of flow during the summer months interrupted by short periods of zero or near-zero flow. This flow pattern is indicative of the water requirements of the major crops in the irrigated area of the canal, which yield several cuttings during a single season. Generally speaking, the periods of near-zero flow correspond to the times of harvest when no irrigation is necessary. The periods of significant flow, which we will refer to as flow humps, correspond to the periods of crop growth between cuttings. There is much that can be said about differences to the flow pattern for the several years shown in the graph, which observations serve to open our discussion of flow prediction and reveal some of the difficulties involved. For example, the beginning date of major canal flow for the several years shown varies within a period of about two weeks; the earliest and latest starts to flow are from the year 2000 on April 13 and the year 2003 on April 27,

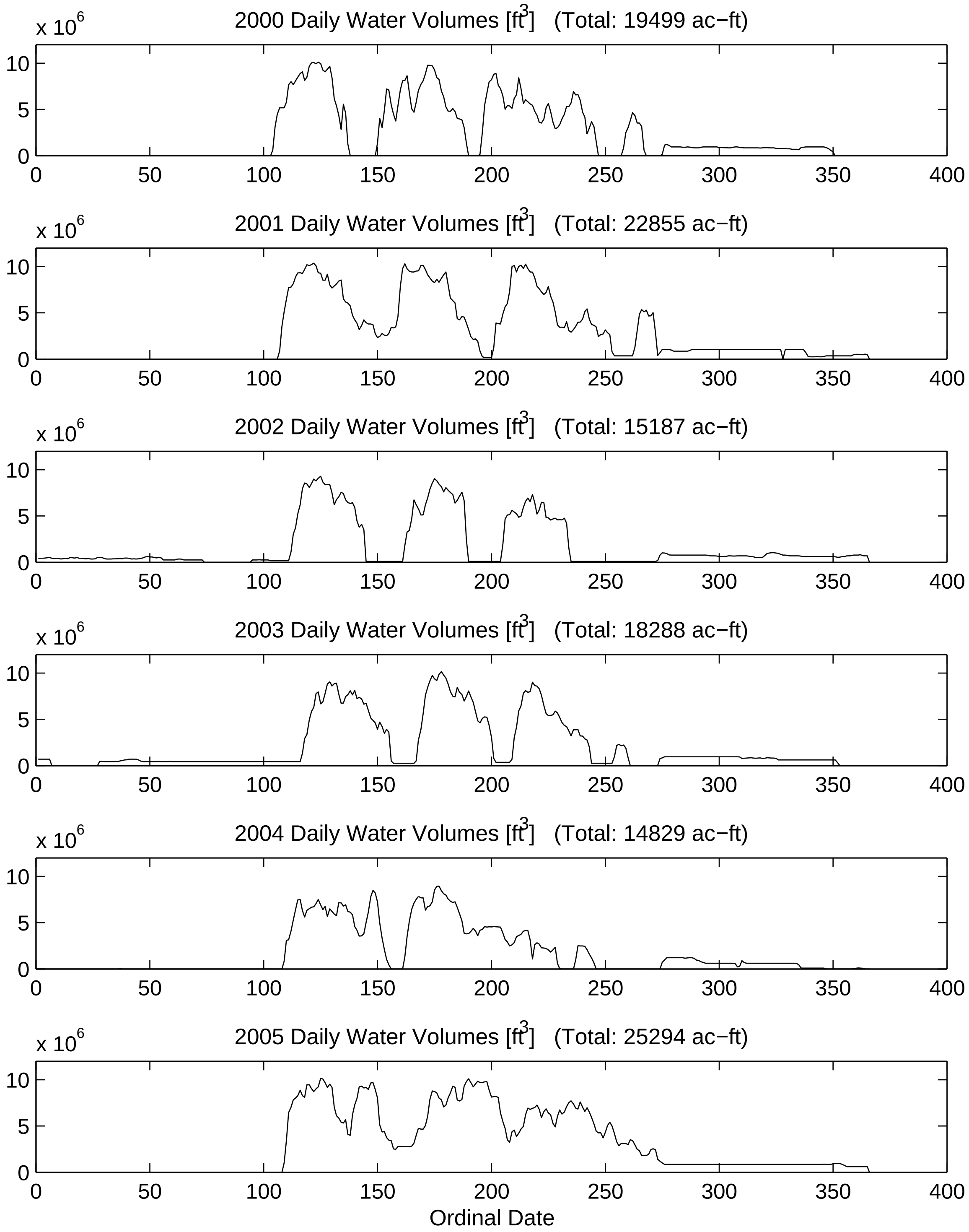

Figure 4.1: Historical flow data for the Richfield Canal from the years 2000 to 2005 represented by the daily volume of water flow passing measuring devices at the head of the canal from January 1 to December 31.

respectively. The duration of major flow, which generally occurs between mid April and early September, also varies, with the shortest watering season lasting only 122 days in the year 2002 and the longest, lasting 164 days, in the year 2005. For some years (especially 2000, 2002, and 2003) the flow humps are well defined with flat-lined gaps of near-zero flow between humps, while for other years (particularly 2005 and the first part of 2001) flow between humps does not bottom-out, but rather appears as a depression in the flow curve. The figure also indicates the total yearly volume of water (determined from hourly flow rates) that passed the measurement point (at the head of the canal) for each year. With only this set of flow data we can begin to hypothesize reasons for the patterns that we see in the data. (Note that the examples given are ideas that seem to be supported by the data, but are not necessarily expected to be an accurate specification of the real conditions. They are discussed only to prime the mind of the reader.) For example, we notice that the years with the largest total water (2001 with 22,855 ac-ft of water and 2005 with 25,294 ac-ft) coincide with the years for which flow humps are less well-defined. If these are years for which water availability in the basin was high (which appears to be the case based on high total flows for other canals in the basin during these years), then this pattern might represent a relaxation to strict water use practices—less insistence on frugality in times of excess. It might even represent a mechanism for unloading excess upstream reservoir water. With another look at the figure we can see that the shape of the first flow hump is very similar for the years 2001, 2002, and 2003 but that very few other flow humps can be identified whose shape can be compared. Some of the later flow humps (in particular, the third humps for years 2000, 2001, and 2003 and the second hump for year 2004) appear to be elongated, that is they start with high flow and then, rather than stopping abruptly as with other humps, they taper off more slowly, or shift to a lower secondary flow, and are then followed by a much smaller flow hump lasting for a much shorter duration. These patterns may reflect the agency of individual farmers as to how many crops to grow in a season and when to grow and cut them. For example, in 2001, the secondary flow level on the third hump along with similar holdouts (but with much shorter duration) on the first and second humps (one of which obscures the distinction between the first two humps) might simply be a set of farmers with a later start on the season watering and harvesting at a delayed time as afforded by the apparently long season (possibly due to weather conditions). The flow in 2004 might be explained by a set of farmers insistent on harvesting three crops even in a low water year in which most farmers settled for two. Many of the above conjectures while intrinsically related to weather patterns (water availability based on precipitation and snowpack, season duration based on annual climate, farmer agency based on water availability) are seasonally macroscopic concepts that may not be addressed well by the hourly data, primarily weather data, that are available is the database. However, they are introduced for two reasons. First, they serve to arouse the mind of the reader to the idea of learning from data so as to make a link with what a learning machine does, as discussed presently, and second, they begin to reveal the complications inherent in the flow data, which make application of a learning machine rather difficult.

In thinking about using observed patterns for prediction, we turn the picture around, that is, we observe (or measure) elements like those mentioned such as weather patterns and total available water and then, based on the patterns we have recognized, make an estimate or prediction of the flow. This is the essence of the task that is accomplished in machine learning. The machine uses a set of observations of flow and quantities related to flow—collectively the training data—to determine a general relationship (the functional model) between the related quantities and the flow, which can then be used to determine the flow (a prediction) that corresponds to a particular measurement of the related quantities. The application of a learning machine to a problem is not a trivial or straightforward task. Though a machine can be very powerful at determining a functional mapping from input to output it does so effectively only if well directed. It is best to treat the machine as ignorant while doing much to posture the data externally. Obviously, models determined via a learning machine can only incorporate the data that are presented to the machine. Furthermore, the capacity of the machine for establishing the model is limited by its learning mechanism, so that when an input is provided to the machine, the intuitive concepts that connect the input to the desired output and which motivate its inclusion as an input may not be interpretable by the machine via its learning mechanism. For this reason establishing a model necessitates selection of model inputs based on the potential for exploitation by the machine.

4.1 Model Inputs

Much of the experimentation for determining the merit of potential inputs for our flow models was performed using years 2002 and 2003 flow data measured on the Richfield canal. These years of flow data were chosen for their simple and comparable flow patterns with very distinct flow humps.

4.1.1 Past Flow

The flow for Richfield Canal in 2002 is shown in fig. 4.2 for hourly measurements taken at the head of the canal. Flow measurements at the head of the canal best reflect the control being exerted by the canal operator. As such, these measurements include some changes in flow that appear to be instantaneous. These jumps in the flow are the result of changes to the flow made by the operator between the discrete hourly measurements. Aside from these generally small hourly jumps, and the much larger jumps at the beginning and ending of each crop hump, the flow within a hump is basically smooth. This indicates that a good input for prediction might be a simple past flow or an average of recent past flows. In general, for a smoothly changing flow a good first estimate of flow at a particular time is the previous flow. If the flow is broadly smooth but with erratic noisy changes occurring at very small times steps then a good first estimate of flow at a particular time is an average of previous flows. Except for times following large hourly changes, an average of the past flow is already in the vicinity of the current flow. From the viewpoint of prediction error, at times when the flow is undergoing little change the past average predicts the current flow with a small error. At times when the flow is undergoing a large change the past average predicts the current flow with a larger error. This is easy to see by visualizing the average flow as a smoothed version of the flow that is then delayed. For example, fig. 4.3 shows

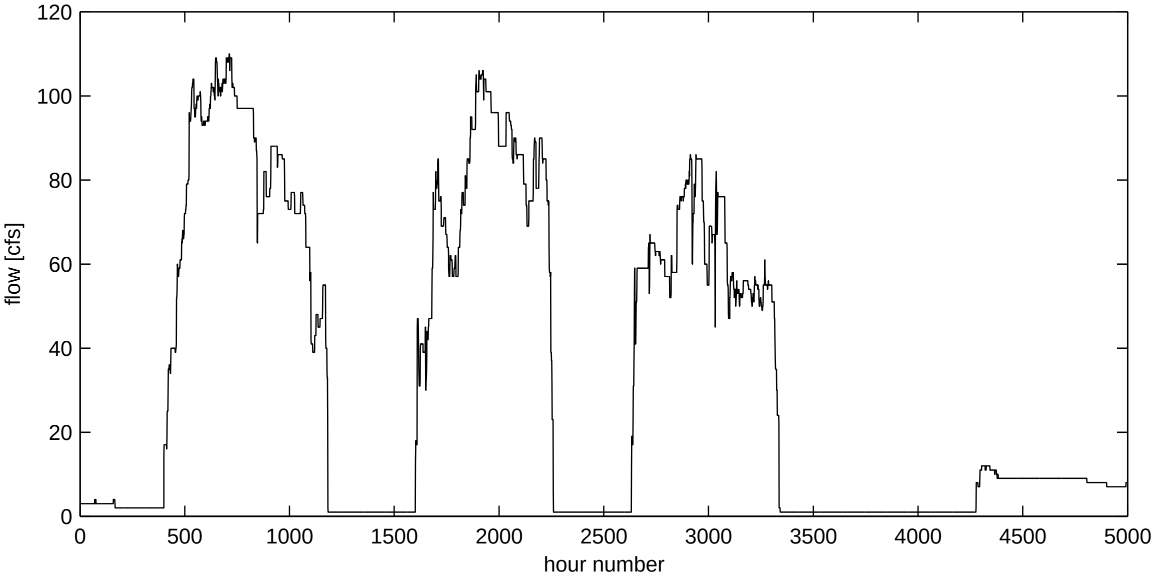

Figure 4.2: Richfield Canal flow at its head from April to October 2002.

an average flow formed from 24 hourly flow measurements. This flow is plotted against the actual flow so that the 24 hour average is plotted at the same time as the actual flow from the 25th hour. Plotting in this manner shows that the past average looks smoothed and delayed by about 12 hours. We can think of this as a very basic one-hour-ahead predictor where the current flow is being predicted by the average of the past 24 hours. The error in the prediction is also shown in the figure. For smoothly changing actual flows the average has a small error (is a good predictor). For increasingly steep changes in the canal flow, the difference between the average and the current flow is much more pronounced. An increase in the number of measurements in the average or an increase in the number of hours between the current flow and the measurements used in the average, serves to extend the delay and increase the size of the error. By itself the past average is insufficient at fully describing the current flow, but it provides a rough indication and becomes a good input for inclusion.

4.1.2 Date

In the process of choosing inputs some thought went into the inclusion of dates as

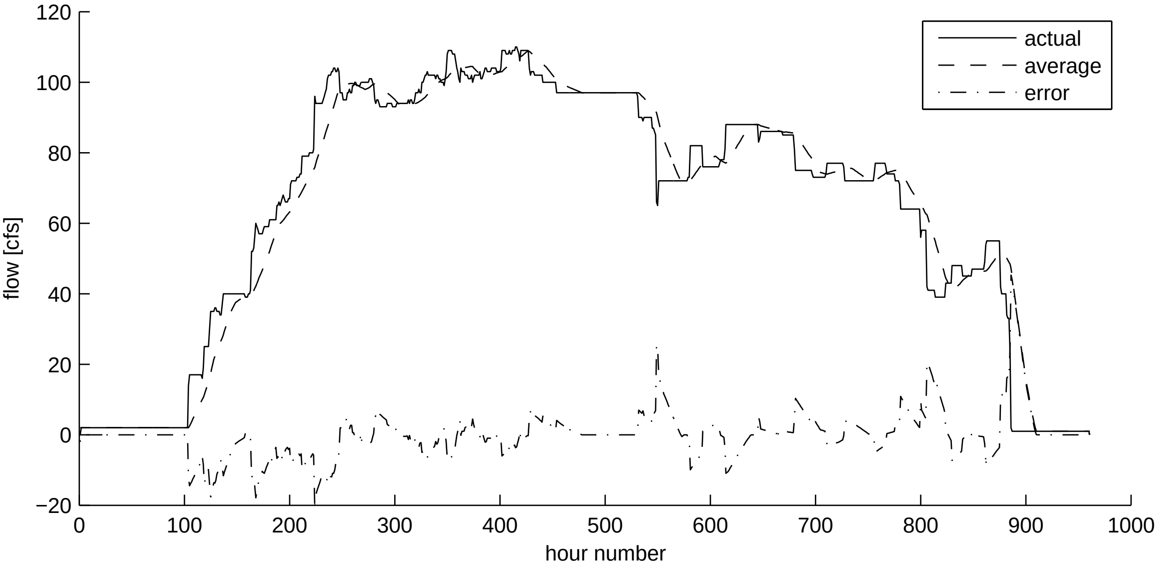

Figure 4.3: Past average flow as a basic predictor for the first crop hump of Richfield Canal 2002.

inputs. While not an explicit data item, the calendar date and time are associated with each measurement and recorded with the measurements in the database. Dates are considered valuable inputs for several reasons. The date is a concept that provides some indication of the activities of canal use. Basically, there are some dates (spans of time) when irrigation is occurring and others, such as between crop cuttings, when no irrigation occurs. Inputs that are close in date are often close in output value. Date is also indirectly an indication of water use. As a season progresses the numerical date increases. This increase roughly parallels the sum of water use through the season; a large (late) date implies a large volume of water use up to that point in the season while an earlier date indicates a smaller volume of water.

Several different date inputs are to be considered. The first is a seasonal calendar date that starts at a set day just before the beginning of canal flow for the season. This seasonal date marks the real-valued number of days from this starting point, increasing in one hour increments. Such a date will be most effective when the time spans of canal use are similar from year to year. In the extreme case, with identical flow patterns from year to year, knowing the seasonal date would be as good as knowing the flow itself, as a simple look-up table could show the flow associated with the date. Naturally, this is not the case, but the example serves to show that date can provide at least a general indication of flow. However, some changes from year to year make for reasonably large differences in the starting time of flow in the canal. A date that indicates the beginning of flow for one year may be several days offset from the date for another year. This fact begs the question of using a date that, rather than starting at the same calendar date, starts at the beginning of canal flow for each season. We call this seasonal flow date. This type of date lines up the seasonal date by the beginning of canal use, however, as with the seasonal calendar date the seasonal flow date is subject to year-to-year changes. For example, if the growing season is shorter one year than another, then the two flow patterns may line-up well by date at the beginning of the season, but poorly at the end of the season. In both cases year-to-year differences can invalidate cross-year generalization ability.

A third date option is a hump date, or a date that restarts at the beginning of each crop hump. Since the flow for the gap between humps takes on a constant value, once within the gap no prediction of canal flow is required until the time of the next hump. If the starting time of the next flow hump can be ascertained independent of prediction then there is also no need to determine the duration of the gap. Predictions can be carried out just for the time spans of the flow humps. With a hump date, all flow humps are lined-up by the beginning of flow in the hump. Still, questions of validity across flow humps exist due to differences in the duration of flow humps and fundamental differences that may exist between humps at different times in a season.

4.1.3 Total Flow

To have a more direct indicator of water used in a season another input is considered. This input, the total flow, is simply a running tally of the flow values at each of the measurement times up to the time of prediction. Technically, the flow measurements in units of cubic feet per second (cfs) should not be added together, but since all units are the same and all time steps are equal, the sum across time is proportional to the volume of water. The RVM is invariant to scaling of the inputs and a change of units is nothing more than scaling by a unit conversion factor, therefore, for the RVM, the sum of flows is equivalent to a volume quantity. Such an input indicates how much water has been used collectively in the season. This is important because farmer ordering behavior in a system with limited water can be a function of the amount of water already used. Unfortunately, this total flow does not indicate the portion of available water used, just the magnitude. Since water availability differs greatly between seasons, a total flow value that represents most of the available water from one year may only be a small portion of the available water for another. These differences in water availability may invalidate generalization to other years.

4.1.4 Reference Crop Evapotranspiration

An input that is expected to be the most informative for predicting the canal flow is reference crop evapotranspiration. Reference crop evapotranspiration, hereafter referred to simply as evapotranspiration, is a physical quantity that measures the evaporative power of the air for a known reference crop. It is computed using temperature, humidity, solar radiation, and wind speed measurements and indicates the depth of water that is evaporated and transpirated from the reference crop under the weather conditions used in the computation [5]. It equates to the water need of the reference crop for proper development under those conditions. Evapotranspiration should be an effective input because it is conceptually linked to the canal flow. Evaporation and transpiration leave a crop in need of water. Farmers place water orders as they ascertain the needs of their crops. Though they do not likely compute the need in terms of evapotranspiration it is the same set of conditions that generate the need that is ascertained by the farmers in other ways. Farmer orders, therefore, are based implicitly on the evapotranspiration that is occurring. Each of the weather indices necessary for computation of evapotranspiration are available in the database at hourly time steps as collected from weather stations in the basin.

4.1.5 Input Format

In recognizing a set of potential inputs and choosing a subset of those inputs, there is still some flexibility for the format with which the inputs will be used. Raw measurements taken directly from the database might be considered the best choice, but perusal of the database reveals many missing or invalid entries that make direct utilization problematic. Rather than diminishing the dataset by eliminating missing or invalid data, raw measurements as inputs can be replaced with short time averages or other combinatory calculations. This allows the inclusion of time steps involving missing and invalid data by providing substitute values for the same. Given that the database consists of hourly measurements for all data items it may still be tempting to utilize all of the data in hourly time steps to take full advantage of the data resolution. We here argue that the data resolution is most important for the target quantities of the prediction, in our case the canal flow values. If good prediction can be achieved for hourly time steps, it does not matter if the inputs for the prediction model are less resolved. In fact, averages or other combinatory calculations can actually provide a way to include more information into the inputs. For example, superior results were achieved by Khalil et al. [4] for the first principal component of a set of weather indices versus inclusion of all indices separately. The calculation for reference crop evapotranspiration incorporates a number of weather indices into a single quantity. This quantity has more direct bearing on the prediction problem than any of the weather indices individually and it is arguably better than including all weather indices separately as it is a functional combination of the weather data, that speaks directly to water demand. We need not require the learning machine to learn a function that is already known. Further, while the calculation of reference crop evapotranspiration can be computed at an hourly time scale, accurate computation involves several complications. A less precise hourly method tends to some over- and under-predictions during the course of a day, which in sum over the hours of the day provide values comparable to the daily calculation [5]. In using daily reference crop evapotranspiration as an input these difficulties are prevented and instances of missing weather data can often be overcome through the average, maximum and minimum statistics that are required for the computation. For these reasons the daily time scale seems the most appropriate. It combines over time and across data type allowing recovery from missing data and providing a physically based input that is intuitively linked to the prediction problem. In the same vein, rainfall as an average or total is a more useful input than rainfall during a specific hour because it speaks to how much water is being received rather than the possible short burst of precipitation that may be represented by an hourly rainfall quantity. In brief, we choose to use inputs in daily quantities but with values that can be determined for any 24-hour period (i.e. even periods bridging across two adjacent days) so that the relative time offset of the input to the target can be maintained for every hourly time instant of the target flow.

4.1.6 Notation

To ease the discussion of the various experiments we here introduce some notation for the inputs used. Each quantity in the input vector is given a letter to denote the input type with a subscript that indicates the relative time in days that the input quantity precedes the target quantity. Each of the inputs that is a combinatory calculation—for example, an evapotranspiration value or a past average flow—is denoted with a capital letter, while quantities that come more or less directly from a single measurement in the database—for example, the seasonal or hump date or the target flow itself—are denoted with a lower case letter. For uppercase combinatory quantities the relative time subscript indicates the relative time between the most recent measurement contributing to the calculation and the target quantity. For quantities that are a combination of measurements from a 24-hour time period the time subscript is sufficient to specify the set of hours (relative to the target flow) from which measurements were taken to calculate the flow. For example, a daily evapotranspiration quantity preceding the target flow by two days is denoted E_2 where by the subscript we know that the quantity was calculated using weather data from 48 through 71 hours preceeding the time of the target flow. Past flows averages are denoted with F, rainfall with R, total flow with T, and dates with d. Since the time of the date quantity is linked with the time of the target flow the subscript for the date instead indicates the type of date with an s for seasonal date and an h for hump date. As an example, a training vector including three evapotranspiration values and three past flow averages starting one day back from the target flow and utilizing a hump date would be given by

$$\begin{bmatrix} F_1 & F_2 & F_3 & E_1 & E_2 & E_3 & d_h & | & t \end{bmatrix}^{\mathrm{T}}, \tag{4.1}$$

where the last element in the vector denotes the target flow t. For experiments in which the most recent measurement used to form an input is not from 24, 48, or some other multiple of 24 hours previous to the target flow, the input subscript will instead indicate the span of hours used to form the input. For example, some experiments were performed with a one hour offset so that inputs were formed using data from hour spans such as 25-48, 49-72, and 73-96 rather than the hour spans 24-47, 48-71, and 72-95 which we have represented with the subscripts 1, 2, and 3, respectively. Using the previous example but with this additional hour of offset the input vector is

$$\begin{bmatrix} F_{25:48} & F_{49:72} & F_{73:96} & E_{25:48} & E_{49:72} & E_{73:96} & d_h & | & t \end{bmatrix}^{\mathrm{T}}.$$

In all experiments data files containing the desired set of inputs and the corresponding target were formed for the time series of interest. In our implementation, every row of a data file is a vector containing each of the inputs to be used for predicting the flow at a specific time instant along with the actual flow from that instant as the last element—similar to the example given in (4.1) only now we think of adding an index that specifies the target flow number from a series of flows so that each row (for the example input set) is of the form

$$\begin{bmatrix} F_{i,1} & F_{i,2} & F_{i,3} & E_{i,1} & E_{i,2} & E_{i,3} & d_{i,h} & t_i \end{bmatrix}.$$

The file contains one row for each flow value so that the whole training set can be represented in a matrix as

$$\begin{bmatrix} F_{1,1} & F_{1,2} & F_{1,3} & E_{1,1} & E_{1,2} & E_{1,3} & d_{1,h} & t_1 \\ F_{2,1} & F_{2,2} & F_{2,3} & E_{2,1} & E_{2,2} & E_{2,3} & d_{2,h} & t_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ F_{N,1} & F_{N,2} & F_{N,3} & E_{N,1} & E_{N,2} & E_{N,3} & d_{N,h} & t_N \end{bmatrix}.$$

The inputs of each row are ordered consistently between rows so that each column (except the last) contains the time series for a particular input while the last column contains the time series for the actual flow. For purposes of reducing computation time a random selection of row vectors from the dataset can be provided to the learning process which itself further reduces the set in the process of determining which vectors are relevant.

Thinking in terms of the training vectors for the RVM, each of which consists of an input-output pair, we can represent the matrix as

$$\begin{bmatrix} \mathbf{x}_1^{\mathrm{T}} & t_1 \\ \mathbf{x}_2^{\mathrm{T}} & t_2 \\ \vdots & \vdots \\ \mathbf{x}_N^{\mathrm{T}} & t_N \end{bmatrix},$$

where each $\mathbf{x}_i^{\mathrm{T}}$ is one of the input rows. Then, stacking the input vectors into a matrix and the targets into a vector we have

$$\begin{bmatrix} \mathbf{X} & \mathbf{t} \end{bmatrix}$$

which we recognize as the input set $\mathbf{X}$ and the target set $\mathbf{t}$ from the RVM development (in Appendix A).

4.2 Experimentation

Having discussed the set of inputs that are considered in our experiments we now discuss some of the experiments themselves. These involve the assessment of prediction capability using several prediction schemes.

4.2.1 Predicting with Distinct Seasons

We start with the basic approach of using two distinct seasons to test prediction capability. Specifically, we use Richfield Canal 2002 data to form a model and Richfield Canal 2003 data to test the model. Under this framework, some basic experiments are employed to ascertain the value of particular inputs in predicting canal flow.

As a point of reference for the results of some of the following initial experiments we take the idea presented previously of using past average flow as a direct predictor of current flow as in fig. 4.3 and determine the error of the prediction. For this and future experiments we use mean absolute error (MAE) and root mean-squared error (RMSE) as the measures of prediction quality. The error values computed across all three humps of Richfield Canal 2003 are an MAE of 4.65 cfs and an RMSE of 6.53 cfs. As we begin our experiments we look to improve upon this baseline result. Similarly, for a day-ahead predictor we have an MAE of 10.17 cfs and an RMSE of 13.17 cfs.

Our first inputs of inspection are daily averages of past canal flow as model inputs (rather than direct predictors). Aside from the basic use of past flow just mentioned the physical justification for such an input is that when a farmer is to place a water order he must consider how much water his crop has received recently. The past flow is an indication of the water that has been provided for crop irrigation.

As a first experiment we form an hour-ahead predictor that utilizes a single past average taken from the 24-hour period starting one hour before the time of prediction. The data file for this experiment has rows of form $[F_{1:24}|t]$. The set of actual flow values from Richfield Canal 2002—the targets—coupled with the corresponding past average flow values—the inputs—form the set of training data. We train the model using the set of input-target pairs, or vectors, that correspond to target values occurring during periods of major canal flow in 2002, that is, we exclude vectors for targets preceeding, between, and following the three irrigation humps (see fig. 4.1). After forming the model using the RVM we test the model with input-target pairs taken from the Richfield Canal 2003 data. The model outputs for the set of inputs are compared with the targets using the error measures to assess the quality of the prediction. Occasionally the hyperparameter estimation procedure may reduce the set of relevant vectors so that the only remaining kernel function with a nonzero weight is the constant kernel $\phi(\mathbf{x}) = 1$. This causes all predictions to become constant having a value equal to the value of the nonzero weight. This is the case for the hour-ahead predictor with a single past flow input. As a result prediction quality is poor. However, inclusion of a second past flow $[F_{1:24}F_{25:48}|t]$ prevents such a trivial model so that testing (on Richfield Canal 2003 data) gives an improvement in prediction quality (over the single input model) achieving an MAE of 3.97 cfs and an RMSE of 5.52 cfs, which is also an improvement over the direct hour-ahead predictor. It is possible that prediction quality can be further improved by including more flow data in the input set. From the perspective of the farmer this is like placing a water order with regard to irrigation received in the past several days. Figure 4.4 shows the RMSE and MAE values as a function of the number of past average inputs included from adjacent 24-hour periods $[F_{1:24}F_{25:48}\cdots|t]$. For the hour-ahead predictor, the prediction quality only shows improvement for up to two past flows $[F_{1:24}F_{25:48}|t]$ after which additional inputs serve to degrade the performance. The result for a day-ahead predictor is similar in that the model including a single flow input $[F_1|t]$ trivializes to a constant prediction. However, improvement continues beyond the two $[F_1F_2|t]$ flow model—which achieves an MAE of 9.04 cfs and an RMSE of 11.29 cfs—to the three flow model $[F_1F_2F_3|t]$ which achieves an MAE of 8.91 cfs and an RMSE of 11.01 cfs. Either of these results is better than the direct day-ahead past average predictor. The output flow for the experiment with three past flows is plotted along with the target flow in fig. 4.5. For more than three past flow inputs the performance begins to degrade as shown in fig. 4.6.

Similar types of experiments can be performed for reference crop evapotranspiration,

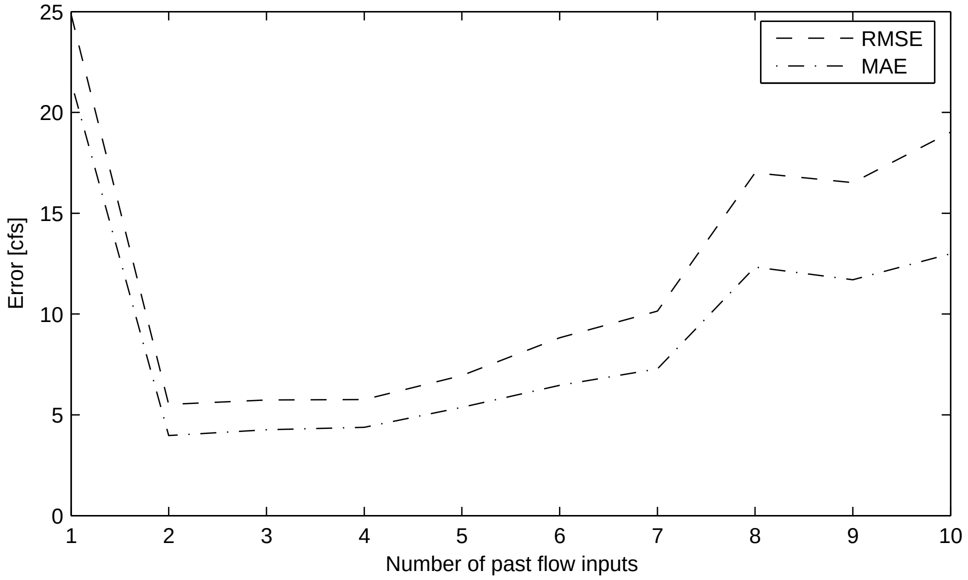

Figure 4.4: Effect of including additional past average flow values as model inputs for an hour-ahead predictor.

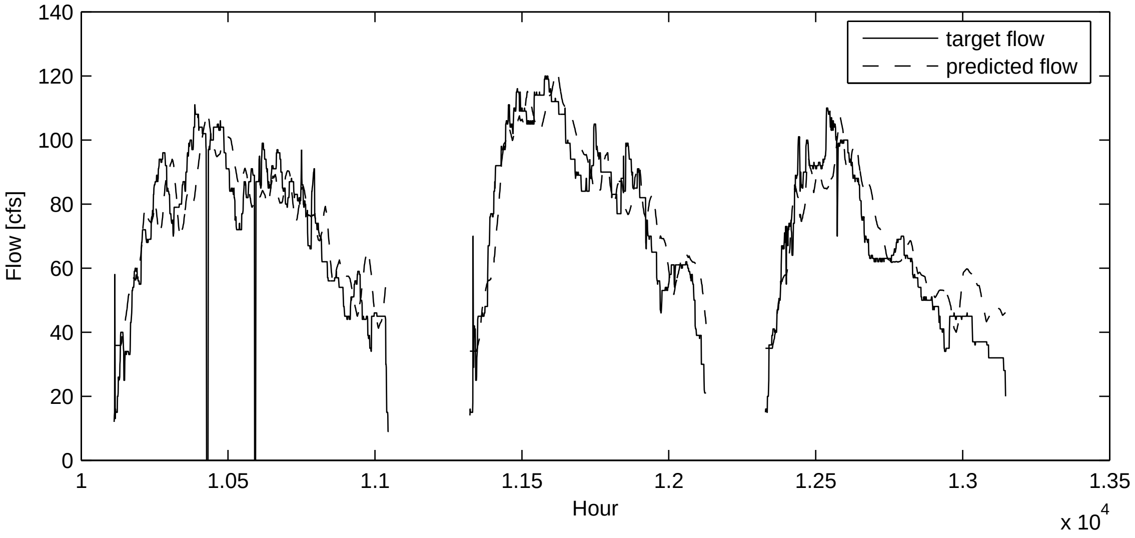

Figure 4.5: Predicted flow plotted against target flow for a day-ahead predictor with three past flows as inputs.

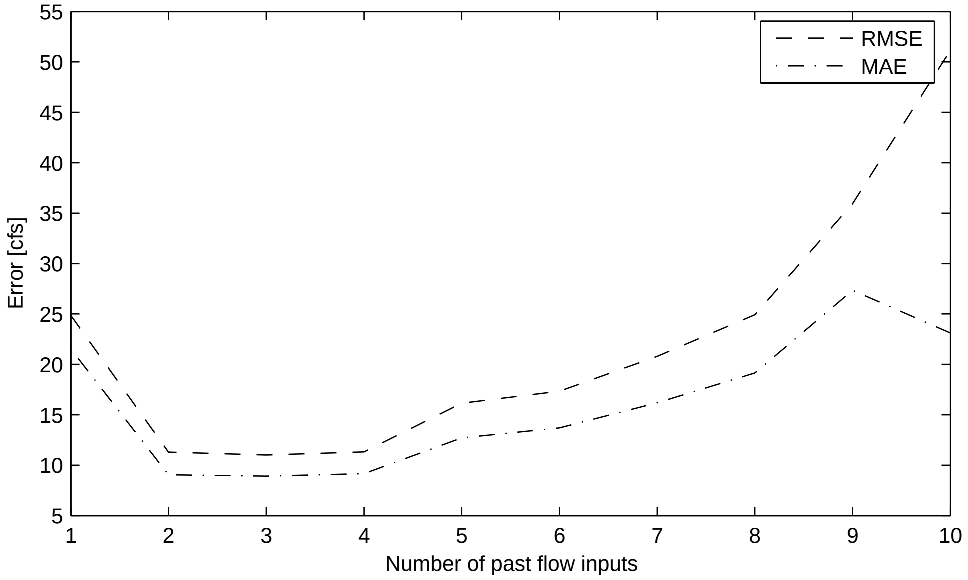

Figure 4.6: Effect of including additional past average flow values as model inputs for a day-ahead predictor.

hereafter referred to simply as evapotranspiration. This input is the amount of water that would be evaporated and transpirated from a reference crop under the weather conditions used in its computation. As a measure of crop water loss it is a direct indicator of crop water need. We should expect a strong connection between evapotranspiration and water demand that can be leveraged to improve prediction capability. Initial experiments are disheartening. For both hour-ahead and day-ahead predictors the shape of the predicted flow has very little resemblance to the shape of the target flow. The best prediction quality—an MAE of 20.83 cfs and an RMSE of 24.34 cfs—is achieved for two evapotranspirations as inputs for an hour-ahead predictor $[E_{1:24}E_{25:48}|t]$.

To obtain the anticipated improvement due to evapotranspiration we consider using past flow and evapotranspiration inputs together. We do this by adding evapotranspiration inputs to a model having a single past flow input. Adding evapotranspiration does prevent the single flow model from trivializing in both hour-ahead and day-ahead cases. With reference to fig. 4.7 the best hour-ahead model—one flow and one evapotranspiration $[F_{1:24}E_{1:24}|t]$—achieves an MAE of 4.78 cfs and an RMSE of 6.38 cfs, with degradation to prediction quality at the inclusion of more than one evapotranspiration. This best result,

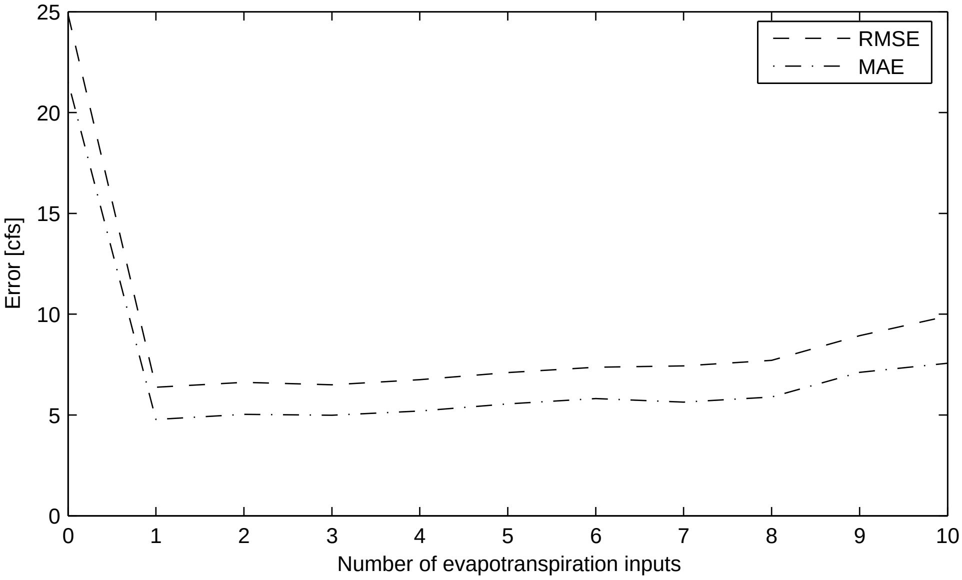

Figure 4.7: Effect of including additional evapotranspiration values as model inputs for an hour-ahead predictor with a single average past flow.

however, is not an improvement over the model with two flow inputs $[F_{1:24}F_{25:48}|t]$. For day-ahead prediction the model with best performance—one past flow and two evapotranspirations $[F_1E_1E_2|t]$ with an MAE of 9.67 cfs and an RMSE of 12.33 cfs—is also not an improvement over the earlier flow model (having three flow inputs $[F_1F_2F_3|t]$). So now we see if any improvement is obtained by adding evapotranspiration inputs to the models having the optimum number of past flows. For hour-ahead prediction we add evapotranspiration inputs to the model having two past flows. Attempting to add evapotranspirations gives a degradation for each successive inclusion starting even with the inclusion of a single evapotranspiration $[F_{1:24}F_{25:48}E_{1:24}|t]$. For day-ahead prediction we add to the model having three past flows. Again, no improvements are obtained by the inclusion of evapotranspiration inputs.

Day-ahead models were also formed with training data sets that include data for the gaps between flow humps. For such models with exclusively flow inputs, the best model—three past flows $[F_1F_2F_3|t]$ achieving an MAE of 8.56 cfs and an RMSE of 10.75 cfs—has better performance than the best model excluding data from flow gaps. Adding evapotranspiration inputs does not further improve the performance, the greatest candidates being the model with three past flows and two evapotranspiration inputs $[F_1F_2F_3E_1E_2|t]$ achieving an MAE of 8.53 cfs and an RMSE of 11.01 cfs and the model with two past flows and two evapotranspirations $[F_1F_2E_1E_2|t]$ achieving an MAE of 8.52 cfs and an RMSE of 11.09 cfs both of which attain almost the same performance as the flow input model.

From the few experiments described the nature of our task is demonstrated. It is a matter of making experiments and checking the relative performance of the results in an attempt to find the best scheme for establishing canal flow prediction capability including selection of the most favorable set of inputs for the task. (Results for the preceeding experiments are summarized in Tables 4.1, 4.2, and 4.3.) The experiments above are arguably a natural place to start. We have chosen one of the larger canals in the Sevier River Basin, which appears to be one with a more regular flow pattern. As a larger canal it is anticipated that the flow patterns are the result of a larger number of water users therefore taking on more of an average characteristic across time than a canal whose flow depends only on a few users. We have also chosen two seasons from the canal flow which seem to be the most similar in shape in the anticipation that the input-to-target functions that are developed using one year of data will be representative of the input-to-target behavior in the other.

The level of effectiveness we have seen for past flow as an input is not surprising. Due to the relatively smoothly changing flow pattern of the Richfield Canal the change in flow over the time period from an hour up to a day is small so that an average of past flow is a basic indicator of future flow. When canal flow is increasing past flow is a small underprediction of upcoming flow and when canal flow is decreasing past flow is a small overprediction of upcoming flow. From the standpoint of using the past flow as a direct predictor of target flow the error of the prediction is small, reflecting the generally small changes to flow in the hour-to-day time frame. From the standpoint of using past flow as an input for RVM training vectors, predictions are a matter of the RVM finding training vectors with past flow values similar to the testing vector past flow and providing an output that is the composite of the kernels for these training vectors. Due to the similar shape of the two seasons of flow this is also effective.

Table 4.1: Hour-ahead prediction error.

#Flow	#Evap	MAE	RMSE
1	0	21.52	24.86
2	0	3.97	5.52
3	0	4.26	5.74
4	0	4.38	5.76
0	2	20.83	24.34
0	3	20.98	24.52
0	4	22.34	26.74
0	5	22.41	26.84
1	0	21.52	24.86
1	1	4.78	6.38
1	2	5.03	6.62
1	3	4.99	6.50
1	4	5.19	6.75
2	0	3.97	5.52
2	1	4.05	5.56
2	2	4.15	5.69
2	3	4.26	5.83
2	4	4.76	6.39

Table 4.2: Day-ahead prediction error.

#Flow	#Evap	MAE	RMSE
1	0	21.52	24.86
2	0	9.04	11.29
3	0	8.91	11.01
4	0	9.16	11.31
0	1	21.52	24.86
0	2	21.10	24.63
0	3	21.21	24.83
0	4	21.81	26.06
1	0	21.52	24.86
1	1	9.84	12.35
1	2	9.67	12.33
1	3	9.92	12.57
2	0	9.04	11.29
2	1	9.23	11.67
2	2	9.02	11.53
2	3	9.85	12.48
3	0	8.91	11.01
3	1	9.24	11.61
3	2	9.38	12.01
3	3	9.55	12.02
4	0	9.16	11.31
4	1	10.89	13.43
4	2	9.60	11.93
4	3	9.94	12.26

Table 4.3: Error for day-ahead prediction including data between humps.

#Flow	#Evap	MAE	RMSE
2	0	8.90	11.36
3	0	8.56	10.75
4	0	10.66	13.42
1	1	9.89	12.46
1	2	9.77	12.34
1	3	9.94	12.70
2	0	8.90	11.36
2	1	8.72	11.13
2	2	8.52	11.09
2	3	9.31	11.73
3	0	8.56	10.75
3	1	8.86	11.24
3	2	8.53	11.01
3	3	10.55	13.20

On the other hand the evapotranspiration input which has direct bearing on crop water need has not shown itself to be as effective an input as we anticipated. We briefly provide some conjecture as to possible reasons for the supposed impairment of the input. Possibilities include the following. Farmers may not readily follow the watering principles connected to evapotranspiration; perhaps many farmers follow a simple schedule for watering their crops. Water flow in the canal may not be a true reflection of demand, perhaps due to canal operators not setting flow to match water orders or modifying flow based on other significant water requirements that are unrelated to irrigation demands. Another possibility, which has more of a bearing on how we choose our prediction scheme, is simply that fundamental differences may exist from one year to another. As examples, there may be changes from year to year in the set of farmers who are served by a canal, where different farmers have different irrigating strategies, or farmers may choose to plant different crops on different years so that basic water requirements are altered, or there may be significant year-to-year differences in weather patterns which were not examined when selecting the years of inspection. These conjectures lead to consideration of alternatives for our scheme of predicting canal flow.

4.2.2 Predicting with a Regularly Updated Model

We will attempt to perform flow predictions using only data from the current year of prediction. This eliminates differences in farmers served and crops planted as well as year-to-year variability that may be introduced as a result of significant changes in weather or overall water availability. It will, of course, leave some of the other challenges described that exist within flow years.

Before introducing our method for prediction utilizing only data for a single year we mention the following item. The hour-ahead and day-ahead experiments do provide some improvement over the direct use of past flow average as a predictor of target flow even though the small time differences allow the past flow itself to be a good predictor of target flow. It is significant then that even at as small a prediction time as one hour the RVM models can achieve a better prediction than the past flow. While hour-ahead predictions have served as an effective starting point for experimentation they are not practical given the time overhead required to collect data, calculate a prediction and act on the result. For this reason and the suspicion that there is not likely to be a useful application for canal flow predictions on the order of an hour ahead, we will abandon further hour-ahead prediction experiments while focusing on day-ahead and longer predictions. Now as our proposed method we form a scheme for updating a model as prediction progresses through a time series. After a certain number of predictions (at successive time-steps) an updated set of training data—formed from all of the data that was used to form the current predictive model plus any data that has become 'past' data due to the progression of time—is presented to the learning process to form a new updated model. This scheme allows for input data at all legitimate time-steps to be included in the model, limited only by the frequency of update. A model updated at every time-step ensures inclusion of all applicable data in the learning process. This entertains the idea discussed by Khalil et al. [4] that new system concepts can be incorporated through an update to the model, utilizing newly available data. Rather than following their method to assess when such an update is necessary, our scheme provides regular updates at an arbitrary interval.

In our attempts to determine the best set of inputs for our predictive model we performed another set of experiments that also utilize only a single season of data. These experiments involved dividing a set of data vectors into two subsets, one a randomly selected training set and the other a testing set consisting of all of the vectors not in the training set. As such there was no temporal difference between training and testing sets, that is, the training subset did not precede the testing subset in time. For this reason the experiments were not attempts at prediction, but rather smoothing between known data points. The purpose of these experiments was to validate inputs and input combinations without being limited by year-to-year differences. The prediction capability of these experiments was not expected to be representative of the results that would be obtained for a truly predictive scheme, however, there was some expectation that relative performance for various input combinations as determined in these experiments would hold true for a predictive scheme. Later when the prediction scheme for a single year of data was developed (as just described), the results of these types of experiments were used to determine which input combinations to use for the more time-intensive scheme (for which testing a large number of input combinations would be less practical). As a whole the split-validation results were pleasing. Inputs which before had not provided much contribution to prediction capability, such as evapotranspiration, now seemed to establish themselves through the split-validation experiments as very effective and useful inputs. Even more, increasing the number of inputs served to provide greater prediction capability. For example, in one experiment along with a single past flow input we provided daily evapotranspiration inputs for 13 adjacent 24-hour periods starting roughly one day preceeding the time of prediction. This achieved an RMSE of 3.18 cfs. These favorable results were initially interpreted as an indication that we had determined a method to fully leverage the potential contribution of the inputs, which led to using the results as guidance for the input sets to use for our regularly updated model. More recent considerations have caused us to reassess and discard this conclusion. For this reason, as well as a fault in our implementation of the regularly updated model for many of these early experiments, we do not discuss the experiments motivated by the split-validation results nor do we more fully discuss the split-validation results themselves.

Instead, our experiments with the regularly updated model rely upon the relative results obtained in the initial experiments with training and testing data from distinct seasons (see sec. 4.2.1). We form regularly updated day-ahead prediction models that consist of two or three past flows as inputs. Each model uses data that is from a single season as discussed except that in order to test the machine over the whole 2003 flow pattern for comparison with earlier experiments the initial model must have some training data that precedes the beginning of flow for the 2003 season. For this purpose 10 days of data were arbitrarily chosen from the 2002 season to include in the model. After each prediction is made the model is updated (an update interval of one hour) by adding to the set of training data any input-output vectors which can be legitimately used without violating the day-ahead status. For example, when predicting in the third flow hump the training set includes all data from the first and second humps as well as any data from the third hump which precedes the time of prediction by at least one day. The model with two flow inputs $[F_1F_2|t]$ achieves an MAE of 9.38 cfs and an RMSE of 11.70 cfs and the model with three flow inputs $[F_1F_2F_3|t]$ achieves an MAE of 8.91 cfs and an RMSE of 11.09 cfs. The latter result is comparable to the previous day-ahead models which had training and testing data from distinct years, however, an improvement was anticipated for the updated model as it includes the most recent and arguably the most pertinent data. If the updated model cannot give better results then there is no justification for its computationally intensive updates. With a model using training and testing data from a single year, evapotranspiration inputs might now be expected to afford some improvement as any weather differences between years can be avoided. Performing the same experiment but adding a single evapotranspiration input gives an MAE of 13.63 cfs and an RMSE of 43.01 cfs for the two flow model $[F_1F_2E_1|t]$ (or 9.74 and 12.73 if a set of very poor predictions at the beginning of the season are removed) and an MAE of 10.22 cfs and an RMSE of 13.10 cfs for the three flow model $[F_1F_2F_3E_1|t]$. The additional evapotranspiration input provides no added utility for prediction in these regularly updated models.

The regularly updated model can also be applied across years in order to provide the most recent data without restricting the input data to the same year. This however begins to encroach upon the computational limitations of the computers used for these experiments. For example, a regularly updated model was formed to predict Richfield 2003 canal flow. The model started with the full season of data from 2002 and added data from 2003 as it became legitimately available through the progression of time. At each update step (each hourly time step in this case) the training set was expanded to include another hour of measurements. Even with data taken only from 2002 and 2003, training sets grew to include more than 4500 input vectors, with the learning process requiring inversions for matrices of the corresponding order for each update step. This experiment proceeded smoothly in its updates for predictions in the first and second flow humps. However, shortly into the third hump, with the set of input vectors becoming very large, memory resources

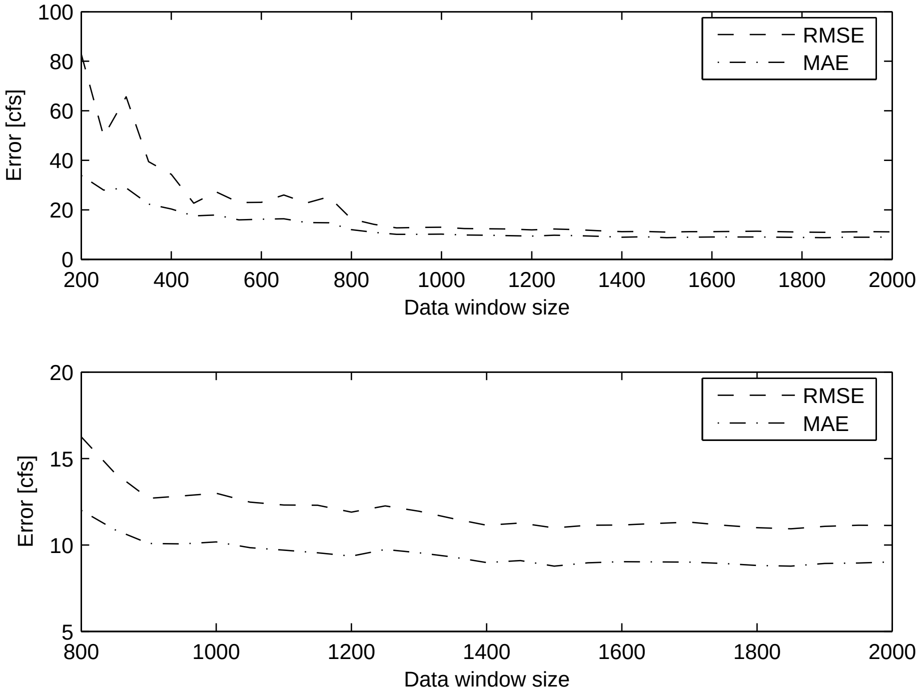

Figure 4.8: Prediction error as a function of the data window length for a three-flow regularly updated model.

were exceeded so that only a few predictions could be obtained. Predictions in the first two humps yield an MAE of 9.19 cfs and an RMSE of 11.29 cfs which is an improvement over the predictions of the first two humps from the single-year update model which give an MAE of 9.93 cfs and an RMSE of 12.07 cfs (which may imply that having additional data even if from the previous year can improve prediction) though it is not a good as the first two humps from the distinct-season model which give an MAE of 8.89 cfs and an RMSE of 11.10 cfs.

These memory difficulties lead to the consideration of regularly updated models with limited data windows, so that the most recent data is still made available to the model, while older data beyond the extent of the window is excluded. Experiments of this nature were performed with various window lengths. Figure 4.8 shows the error for a three-flow model $[F_1F_2F_3|t]$ as a function of increasing window length. Error values are quite large up until a window length of about 800 hours which achieves an MAE of 12.00 cfs and an RMSE of 16.26 cfs. For larger window lengths the error continues to drop, albeit more gradually.

Above a window length of 1400 hours (up to 2000 hours) the error appears to level out, though with some fluctuation. In this range the error values are roughly comparable to that achieved for prediction using training and testing data from distinct years (an MAE of 8.91 cfs and an RMSE of 11.01 for the three flow model). Only window lengths of 1500, 1800, and 1850 hours actually give an improvement with the best performance being for the model with an 1850 hour window—an MAE of 8.78 cfs and an RMSE of 10.93 cfs. However, in a generalizing situation in which an appropriate window is to be chosen *a priori* these results might lead to the reasonable choice of a window length above 1400 hours for good performance but they are not likely to dictate the specific window lengths that will achieve the best performance. Our experiments seem to indicate, then, that for the increased computation of the regularly updated models we can only expect to achieve performance comparable to what can be obtained much more easily with training and testing data from distinct years.

We compare the windowed update results with a multi-regressive (MR) update scheme which we introduced as a baseline for comparison at the end of Chapter 2. The MR model is updated at hourly increments for a variety of window lengths as was done for the regularly updated RVM model so that any performance comparisons between the two can be for models formed with exactly the same set of data. Figure 4.9 shows the earlier results for the three-flow model (originally reported in fig. 4.8) but now with the results for the regularly updated three-flow MR model overlaid. In both cases performance increases with window length until for the MR model a minimum error is reached at a window length of 1200 hours and for the RVM model a basically constant error is achieved starting at 1400 hours. For small window lengths, up to about 800 hours, the MR model actually has much better performance than the RVM model, however, at window lengths of 1400 hours and above the much-improved RVM model is consistently better than the MR model, though by a relatively small margin. As it turns out the best input set for an MR model is not the one with three flow inputs; this linear predictor actually achieves slightly better performance for a model with only two flow inputs. However, the difference between the three-flow and

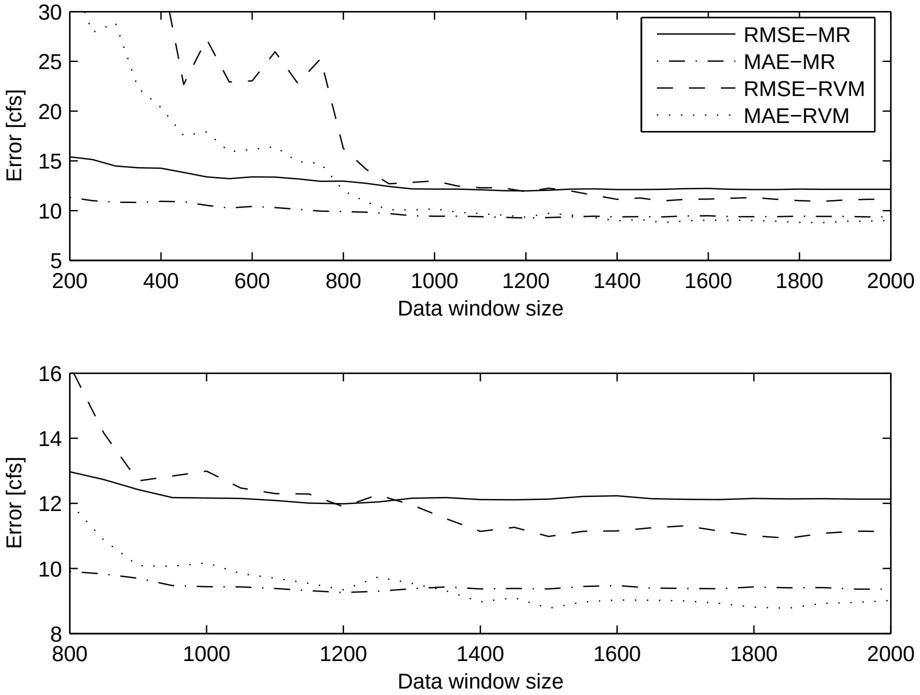

Figure 4.9: Prediction error as a function of the data window length comparing three-flow regularly updated RVM and MR models.

two-flow MR models is small enough that the above observations apply equally well for the comparison of a two-flow MR model with a three-flow RVM model as shown in fig. 4.10.

4.2.3 Delaying and Advancing Prediction Results

As a preface to our experiments and a baseline for comparison the past flow was discussed as a direct predictor of canal flow. For such a scenario predictions are a delayed (and smoothed) version of the actual flow. Experimental RVM results have been assessed in terms of improvement upon the quality of this basic predictor. For hour-ahead and day-ahead predictions small gains have been obtained for certain input combinations. Unfortunately, even for cases of improvement the prediction results appear to incorporate a delay with respect to the actual flow. For an example of this see fig. 4.11 which is a close up of the first crop hump from the 24-hour predictor of fig. 4.5. This observation motivates the set of experiments which follow. The experiments described are a return to the prediction scheme having distinct seasons for training and testing data. In all experiments up to this point

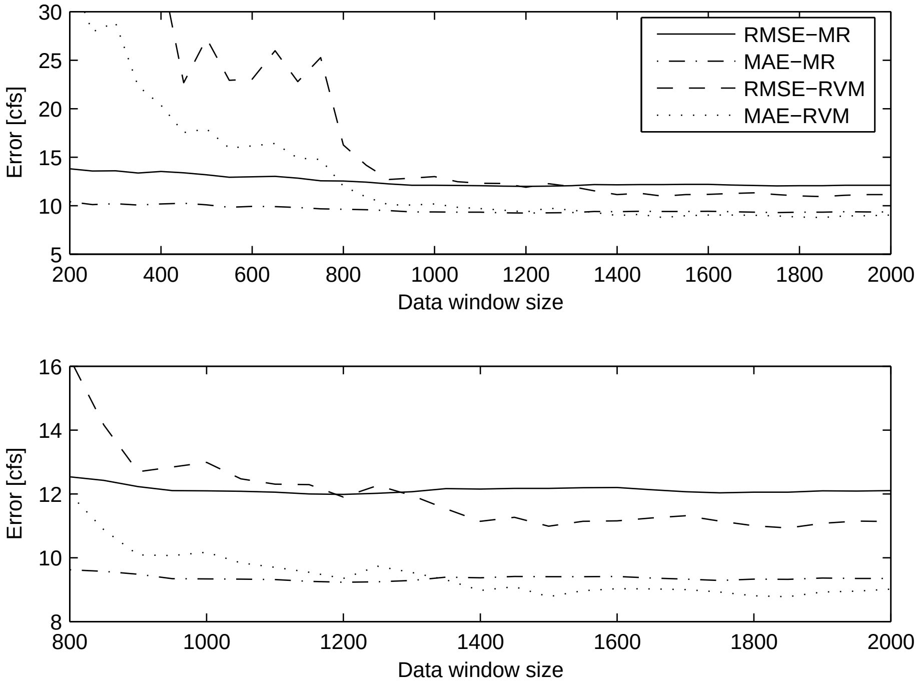

Figure 4.10: Prediction error as a function of the data window length comparing three-flow RVM and two-flow MR regularly updated models.

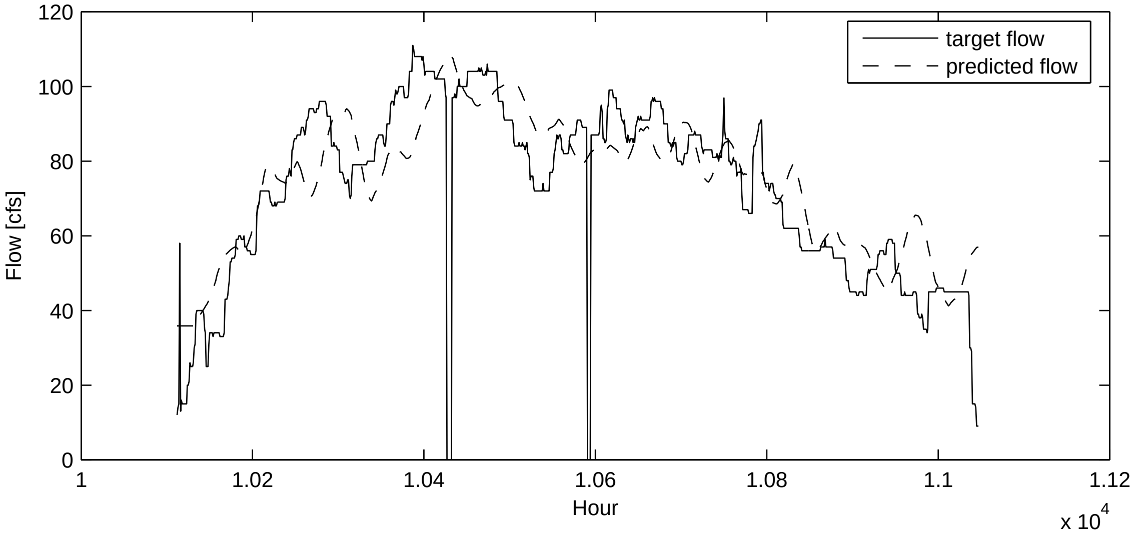

Figure 4.11: An example of predicted flow appearing to incorporate a delay with respect to target flow (a common occurrence).

time relationships between input and output data have been consistently maintained for testing data as compared with training data. For example, in a day-ahead predictor where the model is determined from input-output vectors with a 24-hour gap between the most recent input quantity and the output, testing input vectors are used to predict outputs that are also 24 hours after the most recent input quantity. However, prediction results exhibiting delays prompt modification to this tactic. Specifically, a time advance of the prediction curve can be used to shorten the delay and improve the prediction quality. Unfortunately, this is equivalent to reducing the time span of the prediction by the magnitude of the time advance. For example, advancing the prediction curve of a day-ahead predictor by four hours changes the day-ahead predictor to a 20-hour-ahead predictor. Thus the anticipated performance gain cannot be obtained except by a reduction in the prediction time (which, generally speaking we already know provides better prediction). Still, the concept introduces the idea of a balance between extending training prediction time and reducing prediction delay. The first attempt was to train a model having a 48-hour gap between input and output then advance the prediction result by 24 hours to yield a day-ahead predictor. It was anticipated that the 48-hour prediction would itself be poorer than a 24-hour prediction, but that the 24-hour advance of the result would serve to recover prediction quality by removing some of the expected delay and might thereby yield a net gain in prediction quality over the 24-hour prediction. This possibility was not realized for a 24-hour advance of the 48-hour prediction, however, each of the combinations of prediction time from 1 hour to 48 hours with the corresponding prediction advance (or delay) that yields a day-ahead predictor was attempted. Figure 4.12 shows that some gain over the basic 24-hour predictor is achieved especially for a base prediction time of from four to eleven hours coupled with the corresponding prediction delay of from 20 down to 13 hours. The three-flow model with a prediction time of nine hours and a delay of 15 hours achieves an MAE of 8.50 cfs and an RMSE of 10.59 cfs. A similar result—an MAE of 8.50 cfs and an RMSE of 10.64 cfs—is achieved for a model which includes only two flow inputs at the same base prediction and delay. These results are comparable in value to those obtained previously by including the

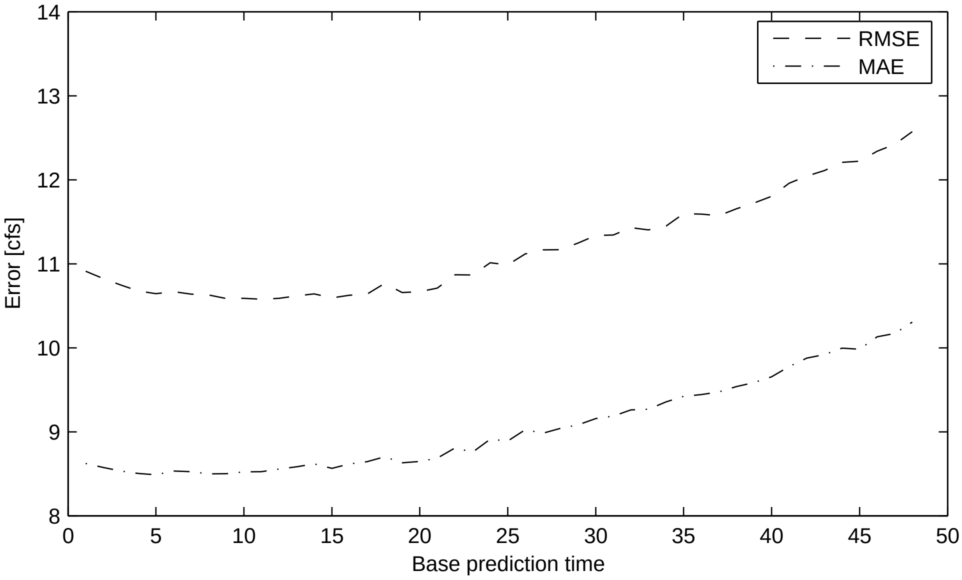

Figure 4.12: Prediction error for three-flow day-ahead predictors formed by offsetting base prediction with the appropriate delay or advance.

data between flow gaps. Applying the method just discussed to the between-hump model with three flow inputs yields only one prediction/delay combination that improves upon the basic 24-hour prediction. This is only a marginal improvement obtained at a prediction time of 25 hours coupled with an advance of one hour which has an MAE of 8.55 cfs and an RMSE of 10.70 cfs. The 24-hour prediction, previously taken as an improvement over the model without data from flow gaps, achieved such only as part of a short minimum on the error curve (see the base prediction times of 24 and 25 in fig. 4.13), while as it turns out the model excluding flow gap data actually achieves a slightly better performance, and for a larger range of prediction/delay combinations (see base prediction times from 4 to 11 in fig. 4.12).

4.2.4 Adjusting the Input Scale Parameter

As previously mentioned, $K(\mathbf{x}, \mathbf{x}_n) = \exp\left\{-\eta \|\mathbf{x} - \mathbf{x}_n\|^2\right\}$ is the form of the kernel function in use, with scale parameter η. All experiments discussed up to this point have used a scale parameter of $\eta = 1$. This parameter serves to scale the squared norm thereby tempering the effect of the vector difference in setting the value of the kernel. With a large enough

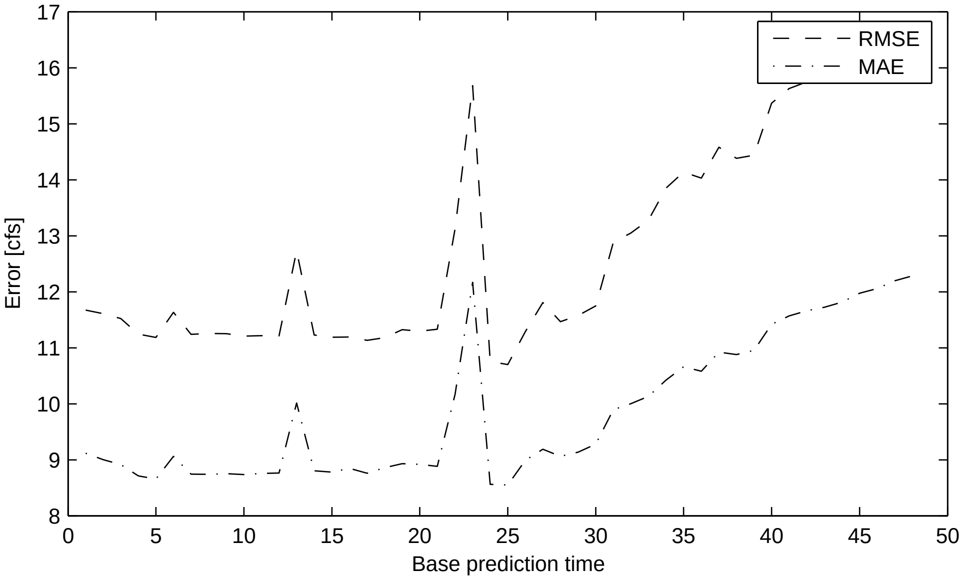

Figure 4.13: Prediction error for three-flow day-ahead predictors formed by offsetting base prediction with the appropriate delay or advance. The results shown are for models including between-hump data.

scale parameter even minor differences between training and testing vectors can be made to diminish the kernel size significantly or on the other hand with a small scale parameter even relatively large differences between training and testing vectors can be tempered to prevent a large reduction in kernel size. Some have made efforts to set this parameter in an optimal way [9]. It is also possible to set this parameter separately for each element of the vector so that the kernel can be considered to have form $K(\mathbf{x}, \mathbf{x}_n) = \exp\left\{-\sum_{d=1}^{D} \eta_d (x_d - x_{nd})^2\right\}$ where η_d is the scale parameter for the dth element [6]. For our purposes a few experiments in which the value of the scale parameter is varied will suffice to determine the effect on prediction quality and allow for the selection of good values for the scale parameter. Experiments were performed by varying the scale parameter from 0.1 to 7.0 in steps of 0.1 for two-flow and three-flow models with and without evapotranspirations. Figure 4.14 shows the results for two-flow models with increasing numbers of evapotranspiration inputs while fig. 4.15 shows the results for three-flow models. In either case increasing the scale values to be larger than $\eta = 1$ gives an improvement. We choose the value $\eta = 3.5$ which consistently yields low error across the combination of flow and evapotranspiration input sets tested

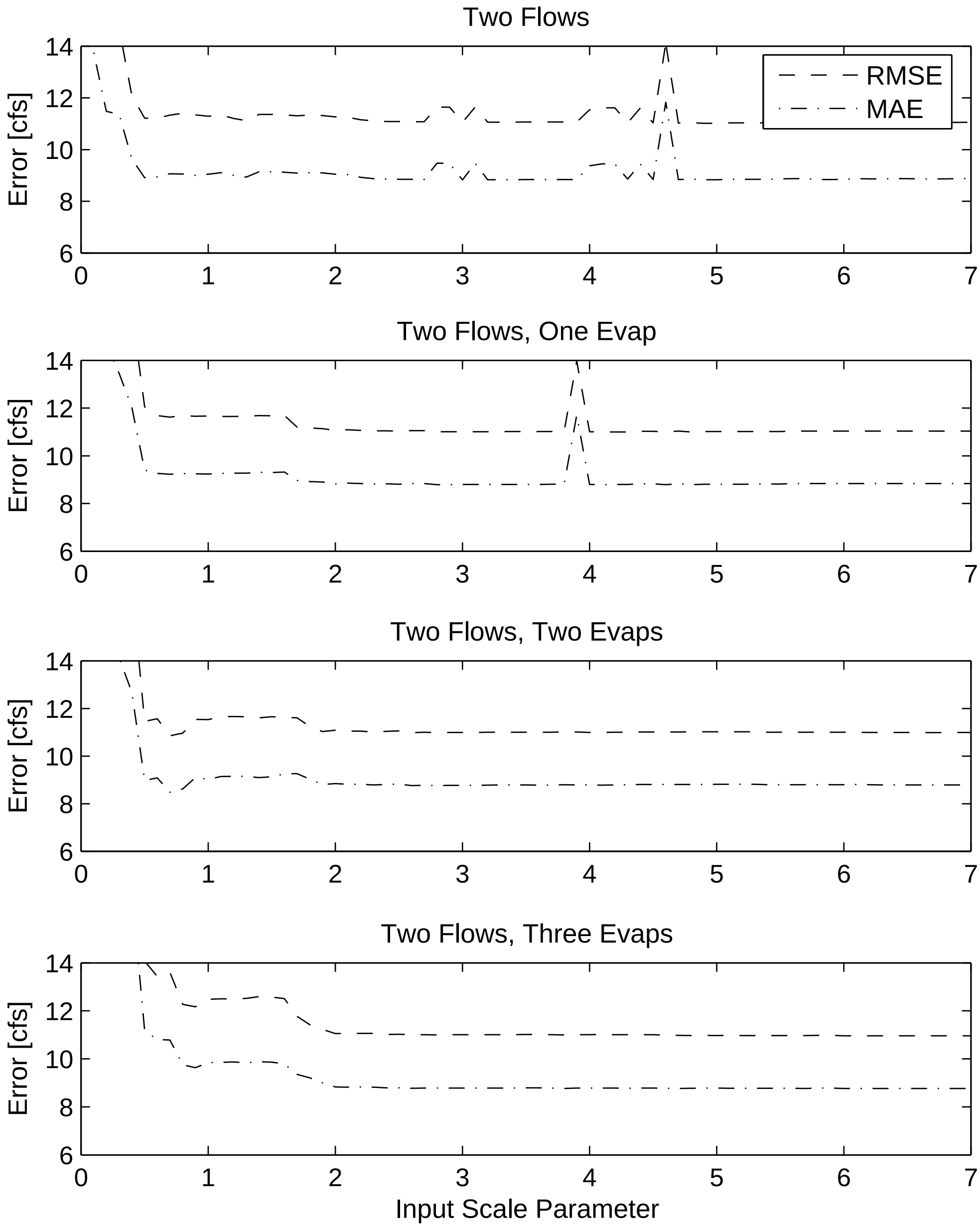

Figure 4.14: Prediction error as a function of input scale parameter for models with two flow inputs and zero, one, two, or three evapotranspiration inputs.

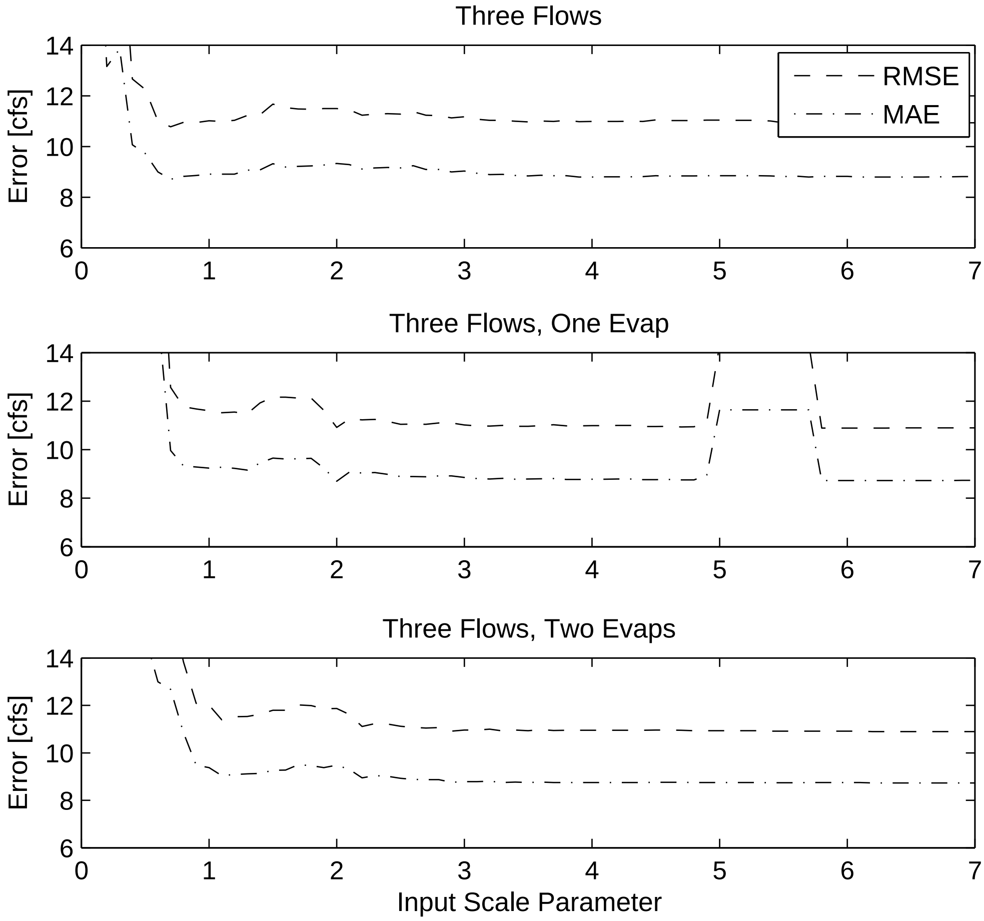

Figure 4.15: Prediction error as a function of input scale parameter for models with two flow inputs and zero, one, or two evapotranspiration inputs.

(those displayed in figs. 4.14 and 4.15). Using this value gives an MAE of 8.84 cfs and an RMSE of 11.06 for the two-flow model and an MAE of 8.84 cfs and an RMSE of 10.97 for the three-flow model, with slightly better results for added evapotranspiration inputs down to an MAE of 8.78 cfs and an RMSE of 11.00 cfs for the model with two flows and two evapotranspirations as well as an MAE of 8.75 cfs and an RMSE of 10.93 cfs for the model with three flows and two evapotranspirations.

4.2.5 Extending the Prediction Time

Next the performance of the RVM was gauged for extension of prediction times beyond one day. In the Sevier River Basin some farmer water orders are required up to five or more days in advance of the time of delivery. For this reason, prediction capability at prediction offsets of this magnitude is desired. Extension of the time offset is straightforward. Data sets are adjusted so that the output column is further offset in time from the most recent input columns. Training data sets must also be regulated to ensure that the output for the most recent training vector precedes the time of prediction by the desired prediction offset. Experiments proceed utilizing the scale parameter $\eta = 3.5$. For prediction times of 24, 48, 72, 96, and 120 hours (one, two, three, four, and five days) models were tested with each combination of numbers of flow and evapotranspiration inputs for between one and 10 flows and between zero and 10 evapotranspirations. Results are given in Table 4.4 (across two pages) for all combinations of up to 10 flows and seven evapotranspirations. The first section on each page of the table shows results for models containing flow inputs exclusively. These are followed by sections where the number of flows (1-4 on the first page and 5-10 on the second) is held constant while the number of evapotranspirations is varied. For the first section on each page (with exclusively flow inputs) the best result for each prediction time is shown in bold. For the remaining sections, groups of models with low error (relative to other models for the same prediction time) are shown in bold. With the table arranged in this fashion it is easy to see that as the prediction time is extended the best prediction models include increasing numbers of flow inputs, or said another way, the number of flow inputs required to obtain optimal performance increases as the prediction time is extended.

In all cases but one—four-day-ahead prediction—the model with overall best performance (for a given prediction time) occurs within the group of superior models containing the best only-flow model. Also, in all these cases (excepting the one) the difference in error between the overall best model and the flow model is at most about 0.2 cfs. This means that the performance for models with only flow inputs is generally a good indicator of the best performance that can be expected, with the inclusion of evapotranspiration inputs providing only marginal improvements, if any. Inclusion of the evapotranspiration inputs appears merely to provide a means of stirring up the pot with the learning process reaching a function that obtains nearly the same result from a modified input set. The exception is for four-day-ahead prediction where the overall best performance—an MAE of 12.78 cfs and an RMSE of 16.58 cfs—is achieved for the model with 10 flow inputs and one evapotranspiration while the best performance for an only-flow model—an MAE of 13.43 cfs and an RMSE of 16.55 cfs—is for the model having only eight flow inputs. Not only do the two results occur for different numbers of flow inputs but performance discrepancy (the difference between the two errors) is large. The 10-flow one-evapotranspiration result is an example of a solitary minimum as compared to the pockets of adjacent similarly-performing models. If such a minimum were to occur in another flow year or canal to which one hoped to generalize it would be difficult to find from the viewpoint of an *a priori* model selection process, whereas finding one of the models within a pocket of similarly-performing models would be much more feasible.

To more fully appreciate the quality of these extended predictions the error results for the best only-flow models are plotted in fig. 4.16 along with the error obtained by treating the average past flow as a direct predictor of current flow (as previously explored for hour-ahead and day-ahead predictors). The performance discrepancy becomes quite large for extended prediction times.

We should here note that in the process of extending the prediction times using the scale parameter $\eta = 3.5$ (as dictated by previous results) we discovered that the models giving best performance for day-ahead predictors were no longer those with two or three flow inputs

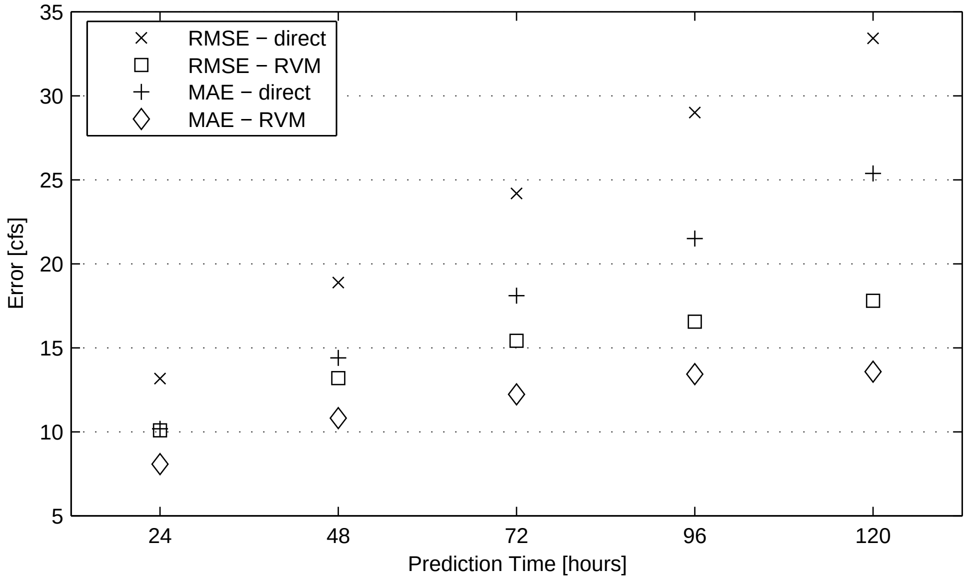

Figure 4.16: Comparison of prediction error at extended prediction times for RVM model predictions and direct past flow predictions for Richfield Canal 2003.

(which gave the best results for the scale parameter set at $\eta = 1$) so that even though the scale parameter adjustment did bring improvement to these models, superior performance is achieved for models with larger numbers of flow inputs with the best performance being achieved for the model with five flow inputs: an MAE of 8.07 cfs and an RMSE of 10.08 cfs.

4.2.6 Other Considerations

In early experiments a date input was included almost as a matter of course for each of the models of experimentation. Initial inclusion of the date input was prompted by a model—similar to those discussed in sec. 4.2.1—which trivialized to a single nonzero weight giving a near-constant prediction. Such an occurrence usually only happens in the case of a model with a single input. For this model (and similarly for those discussed in sec. 4.2.1), including the date input provided more data for the machine to work with, thereby preventing such trivialization. Thereafter the date input was indiscriminately included in most models. However, date inputs proved to be problematic in a number of ways. For one, generating data files with date inputs requires more attention. Since date inputs are chosen to count from the beginning of flow in a season or a flow hump some manipulation of the raw measurement date is required, customized to the season or hump for which the data file is being generated. Seasonal and hump dates introduce the trouble of choosing how to assign dates to data that might be included from before the start of flow in the season or between flow humps with the issue of date values that are potentially negative. Another issue of some concern has to do with scaling of date inputs. For all experiments the datasets are scaled so that numerical values are in the range from zero to one. This scaling is done independently and linearly for each input and the output in such a way that the smallest value in the dataset for a particular input is scaled to have a value of zero while the largest value is scaled to have a value of one. Vectors containing the minimum and maximum values for each input are saved to allow for rescaling of prediction results back into the scale of the physical measurements. For the inputs that have received the most attention such as flow and evapotranspiration there is a fixed range within which each quantity usually varies. For example in the Richfield Canal flow values range from zero up to about 115 cfs and daily reference crop evapotranspiration values fall in the range from zero to 15 mm. If a dataset does not include instances of an input that are close in value to the more general minimums and maximums then the data for that input will be scaled differently than if such values existed in the set. The relative scaling between inputs can be thereby modified, causing significant changes to the model. A sizeable data set will usually have values that fill most of the range for flow or evapotranspiration. However, for inputs like seasonal date and total flow, which are increasing through a season, the dataset must include the full season (or at least data from the beginning and the end) in order to get the extreme values for the input. Such a situation may not always be desired. This issue may be prevented by inserting artifical vectors containing the overall minimum and maximum values for the purpose of retaining consistent scaling or by always scaling the entire dataset before extracting a desired subset. Aside from the above difficulties one issue of greatest concern prompted exclusion of date and date-like inputs (such as total flow) from the models herein reported. This issue involves the effect date inputs have on the output values of the kernel functions. As previously discussed kernel functions provide a means of measuring the similarity of a vector with each of the relevant training vectors retained in the model so as to moderate the portion of each weight value that will contribute to the model output. Roughly speaking, the RVM finds values for the weights in such a way that each training target value can be closely reached by the sum of kernel-moderated weights when its corresponding training input vector is treated as the input to the model function. After the learning process is complete (the weight values are set) model output values for as-yet-unseen input vectors are formed by the give and take of the kernel function comparisons between the input vector and each of the training vectors; each of the training vectors to which the input is similar provides a contribution to the output value. The motivation for a date input is to turn the attention of the model to data occurring at the time of the season that is comparable to that of the input. This may have application when flow patterns are largely affected by the time in the season (seasonal date) or the time within a hump (hump date). On the other hand including such an input may also serve to cause a model to disregard data from other times of the season—because of a kernel-reducing disparity in the date element of the input vector—that might otherwise have application to the prediction. In this way a date input might serve to impoverish an otherwise rich data set by compartmentalizing the data. We did not perform sufficient experiments with and without date inputs to establish the superiority of one model over the other but did observe some results in which excluding the date input improved model performance. We have chosen to avoid the date and date-like inputs of seasonal date, hump date, and total flow.

4.3 Prediction for Other Canals

Our focus on input combinations yielding good prediction results and on developing a model update structure that leverages available data has necessitated a selection of data to work with and to comparatively validate our experiments. We chose the 2002 and 2003 flow data for the Richfield Canal. Our desire is for methods that will provide more general prediction capability that can be applied to any canal in the basin and perhaps even more

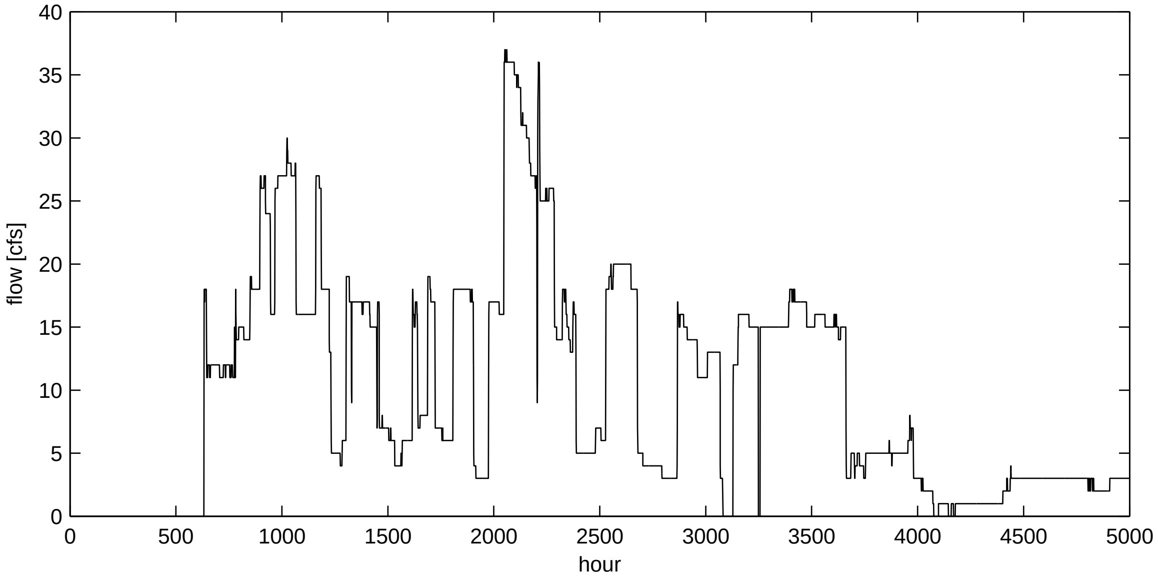

Figure 4.17: Brooklyn Canal flow at its head from April to October 2002.

generally. Some canals in the basin have features which frustrate such a desire. For example, the Brooklyn Canal, being one of the smaller canals in the basin, delivers only an average of about 4900 ac-ft of water annually compared to the average of about 19300 ac-ft for the Richfield Canal. Thinking of canal flow as the superposition of farmer orders one might expect a smaller canal, which presumably serves fewer fields and fewer farmers, to have a flow pattern with more discontinuities as evidence of the specific irrigation demands of its handful of users, while a larger canal should have a smoother flow, less indicative of individual orders. This expectation is realized for the Brooklyn Canal. A typical flow pattern for the Brooklyn Canal is shown in fig. 4.17. The flow consists almost exclusively of periods of constant flow separated by sharp discontinuities (immediate flow changes). When considering prediction of such a flow, we cannot hardly consider a set of smoothly changing weather inputs as a candidate for good functional descriptors. Further, the validity of past average inputs is eliminated by the sharp discontinuities; at the very least the idea of a recent past average as a good indicator of current flow no longer has good application. In order to provide a good prediction for such a flow we must have some way to anticipate when a discontinuity will occur and what the magnitude of the flow will be after the change.

Though these changes are likely driven by crop water needs (a function of recent irrigation and weather patterns) they are not a continuous function of them. It is not anticipated that a good prediction model can be built for flows from the Brooklyn Canal.

On the other hand the methods used on the Richfield Canal can be applied successfully to the South Bend Canal. To demonstrate this we used data from 2002 for training and data from 2003 for testing. We started with input scale experiments (similar to those in sec. 4.2.4) from which an input scale parameter value of 3.2 was chosen. Independent choice of the scale parameter for the South Bend Canal experiments was made due to a smaller range of flow values for the South Bend Canal as compared with the Richfield Canal. With this scale parameter value the extended-prediction-time experiments of sec. 4.2.5 were repeated on the South Bend Canal. Again, experiments were performed for prediction times of 24, 48, 72, 96, and 120 hours (one, two, three, four, and five days) in which models were tested with each combination of numbers of flow and evapotranspiration inputs for between one and 10 flows and between zero and 10 evapotranspirations. Results are given in Table 4.5 for all combinations of up to 10 flows and seven evapotranspirations. The table is given as before with a first section (on each page) showing the results for the flow-only models and each of the remaining sections showing results for experiments where the number of evapotranspiration inputs is varied for a fixed number of flow inputs (1-4 flows on the first page and 5-10 on the second). In this case a trend requiring larger numbers of flow inputs for good performance at extended prediction times is not apparent. Bold values on this table are for the best flow-only model as well as the overall best model for each prediction time. It can be noted that most of these values (with the exception of the best 24-hour-ahead flow-only predictor) can be reached for models with between five and seven flows and either zero or one evapotranspirations which is a relatively small search set.

In fig. 4.18 the error results for the best only-flow models are again plotted with the direct past-flow predictor results. We again observe the increasingly favorable performance discrepancy at extended prediction times for the RVM models over the direct flow predictor. Direct flow predictors have an almost linear increase in error as prediction time is extended

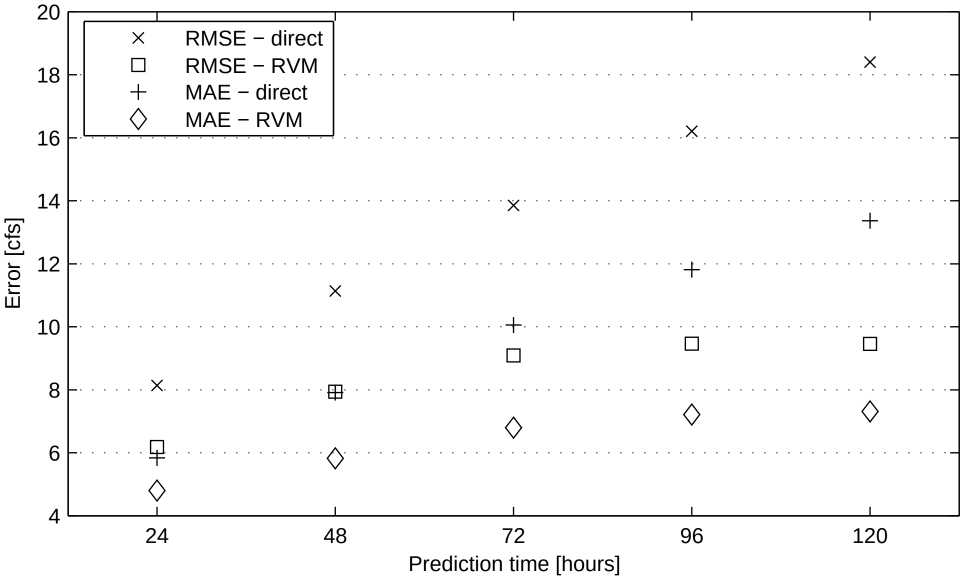

Figure 4.18: Comparison of error at extended prediction times for RVM model predictions and direct past flow predictions for South Bend Canal flow in 2003.

while the RVM models give a more logarithmic increase. This can be said for experiments on both the Richfield (see fig. 4.16) and South Bend Canals. As a whole, error magnitudes for experiments on the South Bend Canal are comparatively smaller than those for the Richfield Canal. This is a reflection of the greater volume of water passing through the Richfield Canal. As previously mentioned the Richfield Canal admits an average annual volume of water of about 19300 ac-ft while the South Bend Canal admits an average of only about 13000 ac-ft annually. With this in mind we determine an average flow rate for each of the canals and make an error plot in fig. 4.19 with each MAE and RMSE value given as a percentage of the respective average flow. This allows a comparison of prediction performance for the two canals. In the figure we see that normalized predictions for the Richfield Canal are slightly better than those for the South Bend canal. However, the difference is small enough that the MAE values for the South Bend Canal are are still better than the RMSE values for the Richfield Canal. The largest difference for normalized MAE across all prediction times is about 1.8% while the largest difference for normalized RMSE is about 3.3%. This shows that our methods are almost equally effective at providing prediction capability for the two canals.

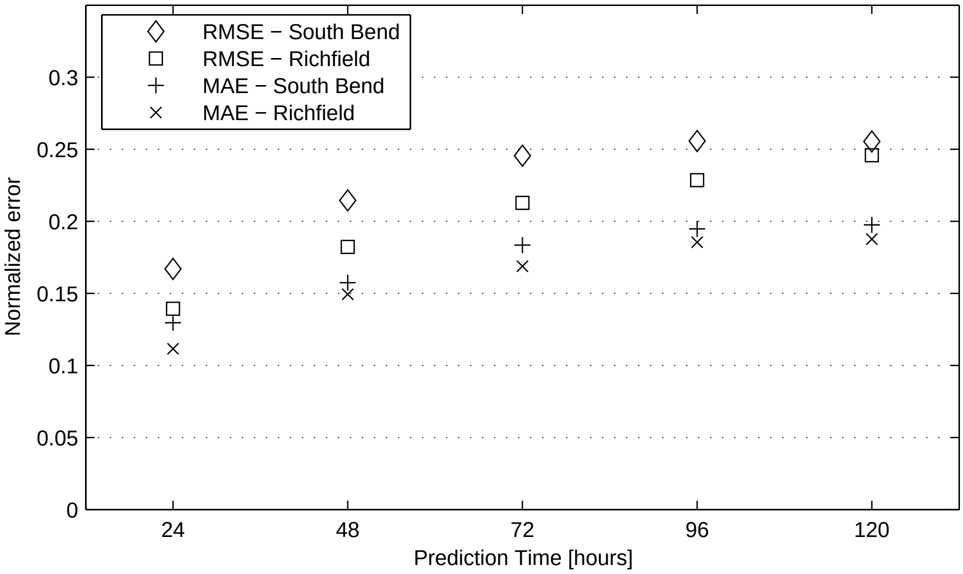

Figure 4.19: Comparison of normalized prediction errors for Richfield and South Bend Canals in 2003.

Table 4.4: Error for one-, two-, three-, four-, and five-day-ahead predictions.

		24 hours		**48 hours**		**72 hours**		**96 hours**		**120 hours**	
#F	**#E**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**
1	0	21.52	24.86	21.52	24.86	21.52	24.86	21.52	24.86	21.52	24.86
2	0	9.46	11.63	11.97	14.55	14.24	16.89	15.72	18.83	17.21	20.29
3	0	8.84	10.97	11.88	14.29	13.76	16.41	15.33	18.27	16.98	20.03
4	0	8.52	10.48	10.86	13.29	13.12	15.62	15.05	17.97	16.57	19.62
5	0	**8.07**	**10.08**	**10.81**	**13.20**	13.18	15.67	15.24	18.06	15.76	18.76
6	0	8.25	10.16	10.93	13.28	13.66	16.09	14.77	17.51	15.25	18.44
7	0	8.66	10.58	11.22	13.52	13.52	16.01	14.23	17.01	14.97	18.42
8	0	8.73	10.71	10.80	13.42	**12.22**	**15.41**	**13.43**	**16.55**	14.54	18.51
9	0	8.78	10.88	11.31	14.38	13.93	17.76	15.63	19.60	14.73	19.16
10	0	8.89	11.11	12.68	16.04	19.34	25.93	14.60	18.62	**13.58**	**17.80**
1	0	21.52	24.86	21.52	24.86	21.52	24.86	21.52	24.86	21.52	24.86
1	1	9.60	11.90	13.03	15.70	15.68	18.62	17.56	20.74	19.06	22.34
1	2	9.55	11.85	13.03	15.73	15.67	18.61	17.57	20.76	19.06	22.35
1	3	9.55	11.85	13.03	15.71	15.67	18.62	17.55	20.78	19.57	23.29
1	4	9.53	11.82	13.01	15.71	15.66	18.81	18.76	22.55	20.31	24.44
1	5	9.54	11.83	12.99	15.81	17.18	20.97	19.23	23.93	20.71	25.14
1	6	9.44	11.86	15.28	19.11	18.40	23.02	19.53	24.35	20.68	25.09
1	7	11.01	13.94	16.12	20.09	18.78	23.41	20.24	25.27	21.00	25.55
2	0	8.84	11.06	11.97	14.55	14.24	16.89	15.72	18.83	17.21	20.29
2	1	8.80	11.01	11.93	14.51	14.33	17.02	15.83	18.84	17.30	20.37
2	2	8.78	11.00	11.95	14.53	14.23	16.87	15.69	18.75	17.26	20.39
2	3	8.79	11.01	11.90	14.48	14.17	16.89	15.53	18.70	17.62	21.00
2	4	8.79	11.04	11.84	14.46	14.08	16.91	15.60	19.02	19.19	23.28
2	5	8.83	11.06	11.86	14.46	14.29	17.31	18.92	23.34	19.84	24.21
2	6	8.73	11.03	12.18	14.95	18.85	23.45	19.08	24.02	20.69	25.25
2	7	9.05	11.53	16.06	20.15	19.04	23.78	19.68	24.98	19.99	24.36
3	0	8.84	10.97	11.88	14.29	13.76	16.41	15.33	18.27	16.98	20.03
3	1	8.79	10.96	11.65	14.11	13.79	16.41	15.46	18.39	17.08	20.18
3	2	8.75	10.93	11.64	14.12	13.73	16.32	15.37	18.30	16.97	20.11
3	3	8.74	10.96	11.56	14.01	13.54	16.17	15.14	18.17	17.59	21.25
3	4	8.76	10.95	11.56	14.09	13.46	16.19	17.60	21.30	19.63	24.13
3	5	8.75	10.96	11.52	14.04	13.38	16.18	18.37	22.75	20.09	24.63
3	6	8.71	10.96	12.00	14.68	13.60	16.67	18.07	22.10	20.72	25.48
3	7	10.41	13.07	13.30	16.57	18.66	23.23	18.72	22.97	20.41	25.06
4	0	8.52	10.48	10.86	13.29	13.12	15.62	15.05	17.97	16.57	19.62
4	1	8.44	10.41	11.00	13.40	13.40	15.86	15.14	18.07	16.63	19.77
4	2	8.40	10.37	11.02	13.43	13.30	15.75	15.15	18.09	16.60	19.78
4	3	8.44	10.40	10.93	13.32	13.27	15.87	14.89	18.00	16.65	20.22
4	4	8.53	10.53	10.90	13.39	12.99	15.70	17.55	21.55	19.18	23.78
4	5	8.48	10.49	10.84	13.29	13.51	16.33	17.53	21.80	19.13	23.81
4	6	8.60	10.65	10.83	13.33	15.81	19.58	18.75	24.19	19.90	25.16
4	7	9.66	12.11	12.72	15.82	16.75	20.79	18.96	23.93	19.75	24.70

Table 4.4 cont'd.

		24 hours		**48 hours**		**72 hours**		**96 hours**		**120 hours**	
#F	#E	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**
1	0	21.52	24.86	21.52	24.86	21.52	24.86	21.52	24.86	21.52	24.86
2	0	9.46	11.63	11.97	14.55	14.24	16.89	15.72	18.83	17.21	20.29
3	0	8.84	10.97	11.88	14.29	13.76	16.41	15.33	18.27	16.98	20.03
4	0	8.52	10.48	10.86	13.29	13.12	15.62	15.05	17.97	16.57	19.62
5	0	**8.07**	**10.08**	**10.81**	**13.20**	13.18	15.67	15.24	18.06	15.76	18.76
6	0	8.25	10.16	10.93	13.28	13.66	16.09	14.77	17.51	15.25	18.44
7	0	8.66	10.58	11.22	13.52	13.52	16.01	14.23	17.01	14.97	18.42
8	0	8.73	10.71	10.80	13.42	**12.22**	**15.41**	**13.43**	**16.55**	14.54	18.51
9	0	8.78	10.88	11.31	14.38	13.93	17.76	15.63	19.60	14.73	19.16
10	0	8.89	11.11	12.68	16.04	19.34	25.93	14.60	18.62	**13.58**	**17.80**
5	0	**8.07**	**10.08**	**10.81**	**13.20**	13.18	15.67	15.24	18.06	15.76	18.76
5	1	**8.08**	**10.14**	**10.71**	**13.10**	13.46	15.94	15.28	18.21	15.88	19.01
5	2	**8.15**	**10.26**	**10.73**	**13.22**	13.27	15.78	15.45	18.53	15.89	19.04
5	3	**8.24**	**10.30**	**10.62**	**13.04**	13.18	15.81	16.64	19.96	18.10	22.03
5	4	**8.50**	**10.56**	**10.81**	**13.39**	14.06	17.07	16.13	20.90	17.17	22.01
5	5	**8.42**	**10.56**	11.12	13.80	15.17	18.58	16.37	21.32	17.34	22.56
5	6	**8.43**	**10.52**	11.61	14.45	16.14	20.90	17.88	22.98	18.00	23.48
5	7	8.91	11.17	13.37	16.94	17.06	21.55	17.88	22.91	17.66	22.95
6	0	8.25	10.16	10.93	13.28	13.66	16.09	14.77	17.51	15.25	18.44
6	1	8.37	10.35	10.97	13.40	13.77	16.37	14.95	17.75	15.41	18.63
6	2	8.11	10.15	11.05	13.61	14.05	16.80	15.44	18.81	15.75	19.23
6	3	8.20	10.31	10.85	13.33	13.79	16.83	15.89	19.45	15.53	19.74
6	4	8.62	10.74	11.21	14.12	13.16	17.05	15.07	19.76	15.46	20.29
6	5	8.72	10.96	11.72	14.75	15.63	20.65	15.88	20.86	15.90	20.96
6	6	8.79	11.15	16.21	20.62	15.51	20.45	16.77	22.27	17.72	22.91
6	7	11.61	14.65	15.04	19.20	15.68	20.99	16.55	21.86	17.62	22.59
7	0	8.66	10.58	11.22	13.52	13.52	16.01	14.23	17.01	14.97	18.42
7	1	8.39	10.42	11.32	13.80	13.51	16.11	14.62	17.64	15.01	18.61
7	2	8.27	10.44	11.15	13.85	13.26	16.36	14.85	18.17	14.95	18.71
7	3	8.51	10.77	11.54	14.33	13.47	16.68	14.84	18.47	15.98	20.28
7	4	8.83	11.11	11.56	14.72	13.38	16.78	14.19	18.05	17.56	23.34
7	5	8.93	11.39	11.73	14.94	14.62	18.14	18.09	22.56	16.79	21.69
7	6	9.09	11.62	12.56	16.84	14.14	18.61	17.11	21.70	17.82	22.57
7	7	10.61	13.56	14.60	19.01	16.67	22.12	16.96	21.57	16.73	21.58
8	0	8.73	10.71	**10.80**	**13.42**	**12.22**	**15.41**	**13.43**	**16.55**	14.54	18.51
8	1	8.43	10.57	**10.68**	**13.15**	**12.05**	**14.95**	**13.85**	**16.84**	15.01	19.50
8	2	8.69	10.72	**10.72**	**13.31**	**12.00**	**14.84**	14.15	17.85	14.10	17.95
8	3	8.81	10.93	**10.68**	**13.26**	**12.28**	**15.21**	15.54	19.22	16.57	21.34
8	4	9.03	11.22	**10.66**	**13.39**	**12.37**	**15.52**	16.61	20.90	18.98	26.29
8	5	9.08	11.37	11.21	14.24	13.06	16.63	15.16	19.24	20.59	29.80
8	6	9.47	11.76	13.55	18.23	15.76	19.71	16.96	21.62	20.76	28.77
8	7	10.61	13.62	13.49	17.45	15.84	20.28	18.17	23.12	20.21	27.52
9	0	8.78	10.88	11.31	14.38	13.93	17.76	15.63	19.60	14.73	19.16
9	1	8.79	10.79	13.97	17.48	16.35	20.07	15.97	20.18	13.97	18.14
9	2	8.62	10.55	13.49	17.17	14.02	17.56	17.98	22.88	16.35	22.11
9	3	8.83	10.86	13.53	16.83	12.60	15.75	17.38	21.85	16.83	21.10
9	4	9.01	11.09	11.32	13.89	15.45	19.75	17.10	21.72	18.03	23.84
9	5	8.71	10.79	12.35	15.58	14.26	17.70	17.04	22.40	19.81	27.39
9	6	9.18	11.55	14.24	17.36	15.84	20.85	18.93	24.74	20.21	27.56
9	7	9.87	12.57	14.92	18.94	16.79	22.07	18.54	24.28	19.02	26.77
10	0	8.89	11.11	12.68	16.04	19.34	25.93	14.60	18.62	**13.58**	**17.80**
10	1	8.98	11.18	14.75	18.32	18.90	23.41	**12.78**	**16.58**	**13.60**	**17.95**
10	2	9.10	11.48	18.14	22.13	19.65	24.93	21.56	28.67	**13.41**	**18.12**
10	3	9.23	11.47	14.80	18.71	19.17	24.34	71.12	125.65	14.68	18.97
10	4	10.04	12.54	14.94	18.18	18.23	23.15	19.72	25.53	17.39	22.89
10	5	10.12	12.47	14.90	17.91	18.00	23.89	18.87	24.52	20.76	26.47
10	6	9.37	11.59	14.34	17.86	18.38	23.97	21.59	29.44	20.73	26.79
10	7	9.82	12.44	15.95	20.40	19.70	25.54	20.07	27.30	23.74	33.99

Table 4.5: Error in extended predictions of South Bend Canal flow for 2003.

		24 hours		**48 hours**		**72 hours**		**96 hours**		**120 hours**	
#F	#E	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**
1	0	11.84	13.62	-	-	7.54	9.85	8.43	10.85	11.36	13.08
2	0	4.89	6.35	6.82	9.03	7.32	9.40	8.68	10.76	8.63	11.15
3	0	**4.80**	**6.18**	6.01	7.78	6.98	9.10	7.58	9.99	8.42	11.21
4	0	4.95	6.31	6.10	7.90	6.94	9.03	7.80	10.16	8.38	11.03
5	0	4.86	6.26	6.10	7.94	7.07	9.43	7.55	9.88	8.03	10.44
6	0	4.91	6.31	6.18	8.19	**6.80**	**9.09**	**7.21**	**9.47**	**7.31**	**9.46**
7	0	5.06	6.71	**5.82**	**7.94**	6.99	9.19	7.55	9.97	8.34	10.80
8	0	5.01	6.63	5.89	8.02	7.29	9.79	7.51	10.19	8.58	11.30
9	0	5.05	6.73	6.26	8.52	7.07	9.47	7.83	10.56	10.09	12.68
10	0	5.16	6.95	6.80	9.50	9.22	11.78	9.22	11.78	10.60	13.50
1	0	11.84	13.62	-	-	7.54	9.85	8.43	10.85	11.36	13.08
1	1	5.27	6.78	6.77	8.65	8.03	10.03	9.02	11.05	9.35	11.68
1	2	5.26	6.77	6.71	8.66	8.05	10.00	8.75	10.79	9.15	11.55
1	3	5.29	7.00	6.76	8.69	7.81	9.81	8.60	10.74	8.93	11.39
1	4	5.42	7.06	6.72	8.65	7.65	9.82	8.35	10.64	8.51	11.23
1	5	5.33	6.99	6.53	8.67	7.44	9.69	8.29	10.57	8.53	11.30
1	6	5.28	7.08	6.64	9.08	7.50	10.02	8.10	10.86	8.44	11.37
1	7	5.33	7.23	6.84	9.37	7.61	10.36	7.85	10.81	8.54	11.55
1	8	5.47	7.42	6.78	9.39	7.86	10.74	8.62	11.83	9.01	12.07
2	0	4.89	6.35	6.82	9.03	7.32	9.40	8.68	10.76	8.63	11.15
2	1	4.97	6.40	6.29	8.00	7.51	9.31	9.14	11.24	8.95	11.11
2	2	4.98	6.41	6.42	8.25	7.73	9.46	8.54	10.37	8.80	11.02
2	3	5.13	6.70	6.52	8.42	7.76	9.54	8.01	10.42	8.33	10.77
2	4	5.16	6.71	6.44	8.29	7.10	9.22	7.63	9.86	8.43	10.72
2	5	5.11	6.61	6.09	8.04	6.94	9.23	7.60	9.94	7.98	10.52
2	6	5.02	6.53	6.13	8.38	6.99	9.48	7.59	9.82	8.11	10.87
2	7	5.11	6.71	6.86	9.16	6.89	9.49	7.58	9.89	7.99	10.58
3	0	**4.80**	**6.18**	6.01	7.78	6.98	9.10	7.58	9.99	8.42	11.21
3	1	4.87	6.18	6.07	7.72	7.07	8.92	8.10	9.97	9.81	12.05
3	2	4.91	6.26	6.24	8.00	7.44	9.20	8.28	10.09	8.96	11.27
3	3	5.00	6.54	6.44	8.24	7.48	9.25	8.28	10.16	8.76	11.16
3	4	5.02	6.54	5.87	7.71	6.57	8.96	7.33	9.91	8.50	11.01
3	5	4.77	6.32	5.92	8.00	6.83	9.50	7.53	10.70	8.40	10.99
3	6	4.82	6.42	6.11	8.40	6.86	9.42	7.72	10.55	7.78	10.45
3	7	4.99	6.65	6.57	8.92	7.03	9.70	7.98	10.99	8.43	11.13
4	0	4.95	6.31	6.10	7.90	6.94	9.03	7.80	10.16	8.38	11.03
4	1	4.76	6.08	5.90	7.61	6.98	8.85	8.04	10.05	8.63	10.98
4	2	4.84	6.18	6.00	7.75	7.22	9.04	8.17	10.11	8.80	11.20
4	3	5.17	6.70	6.34	8.27	7.14	9.26	7.43	9.81	8.50	10.93
4	4	5.58	7.10	6.48	8.46	7.06	9.65	7.52	10.29	7.43	9.85
4	5	5.24	6.60	6.46	8.53	7.46	10.20	7.62	10.11	7.47	10.11
4	6	5.19	6.64	6.49	8.75	7.51	10.40	7.48	9.91	7.58	10.12
4	7	5.23	6.74	6.70	8.98	7.68	10.71	7.78	10.51	8.13	10.80

Table 4.5 cont'd.

		24 hours		**48 hours**		**72 hours**		**96 hours**		**120 hours**	
#F	#E	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**
1	0	11.84	13.62	-	-	7.54	9.85	8.43	10.85	11.36	13.08
2	0	4.89	6.35	6.82	9.03	7.32	9.40	8.68	10.76	8.63	11.15
3	0	**4.80**	**6.18**	6.01	7.78	6.98	9.10	7.58	9.99	8.42	11.21
4	0	4.95	6.31	6.10	7.90	6.94	9.03	7.80	10.16	8.38	11.03
5	0	4.86	6.26	6.10	7.94	7.07	9.43	7.55	9.88	8.03	10.44
6	0	4.91	6.31	6.18	8.19	**6.80**	**9.09**	**7.21**	**9.47**	**7.31**	**9.46**
7	0	5.06	6.71	**5.82**	**7.94**	6.99	9.19	7.55	9.97	8.34	10.80
8	0	5.01	6.63	5.89	8.02	7.29	9.79	7.51	10.19	8.58	11.30
9	0	5.05	6.73	6.26	8.52	7.07	9.47	7.83	10.56	10.09	12.68
10	0	5.16	6.95	6.80	9.50	9.22	11.78	9.22	11.78	10.60	13.50
5	0	4.86	6.26	6.10	7.94	7.07	9.43	7.55	9.88	8.03	10.44
5	1	**4.73**	**6.09**	**5.85**	**7.66**	7.07	9.18	7.90	9.90	8.62	11.04
5	2	4.83	6.23	6.09	7.94	7.24	9.37	8.21	10.45	8.68	11.18
5	3	5.10	6.72	6.21	8.08	7.25	9.38	7.36	9.50	8.56	11.01
5	4	5.57	7.23	6.52	8.44	7.27	9.61	7.33	9.57	8.15	10.68
5	5	5.42	6.98	6.98	9.59	7.65	10.81	7.58	9.76	7.98	10.65
5	6	5.38	6.94	6.96	9.79	7.94	11.05	7.75	9.98	8.58	11.76
5	7	5.34	7.04	6.83	9.62	7.80	11.15	7.27	9.52	8.97	12.07
6	0	4.91	6.31	6.18	8.19	**6.80**	**9.09**	**7.21**	**9.47**	**7.31**	**9.46**
6	1	4.75	6.16	5.88	7.90	6.98	9.11	7.62	9.61	7.99	10.43
6	2	4.82	6.28	6.15	8.20	7.48	9.69	7.91	10.18	7.90	10.45
6	3	5.33	7.00	6.95	9.29	7.80	10.43	8.06	10.95	8.75	11.81
6	4	5.45	7.12	7.11	9.45	8.10	10.83	7.86	10.61	9.58	12.87
6	5	5.69	7.31	7.17	9.67	7.71	10.29	8.12	10.96	9.11	12.39
6	6	5.48	7.08	7.21	9.67	7.66	10.42	8.21	11.29	9.73	13.25
6	7	5.60	7.31	7.24	9.80	8.08	10.70	8.03	10.91	9.59	12.73
7	0	5.06	6.71	**5.82**	**7.94**	6.99	9.19	7.55	9.97	8.34	10.80
7	1	4.83	6.43	5.78	7.84	**6.49**	**8.64**	**6.82**	**8.88**	7.75	10.28
7	2	4.89	6.48	6.34	8.46	6.95	9.23	7.40	9.82	8.10	10.73
7	3	5.38	6.96	7.16	9.91	9.04	12.82	9.02	12.34	8.37	11.11
7	4	5.87	7.61	7.65	10.75	8.83	12.46	9.25	12.98	9.73	12.99
7	5	6.04	7.90	7.94	11.16	8.77	12.04	10.66	14.76	10.43	14.19
7	6	6.25	8.24	7.90	11.14	9.49	13.31	9.82	13.51	11.08	15.21
7	7	6.26	8.20	7.76	10.60	9.31	13.75	10.45	14.61	10.83	14.76
8	0	5.01	6.63	5.89	8.02	7.29	9.79	7.51	10.19	8.58	11.30
8	1	4.90	6.47	5.86	8.10	6.66	9.06	8.28	12.15	8.69	11.47
8	2	5.02	6.68	5.76	7.99	7.46	10.50	9.02	13.77	8.60	11.37
8	3	5.49	7.20	8.28	11.83	8.97	13.16	9.23	12.93	8.84	11.67
8	4	6.27	8.36	8.75	12.35	10.47	15.15	9.87	13.02	9.93	12.88
8	5	6.33	8.54	8.65	11.93	10.09	14.09	10.93	15.03	11.33	14.60
8	6	6.29	8.82	8.97	12.43	10.66	14.93	11.05	14.98	11.36	15.43
8	7	6.37	8.83	9.20	13.14	11.09	15.34	11.71	16.16	11.56	16.06
9	0	5.05	6.73	6.26	8.52	7.07	9.47	7.83	10.56	10.09	12.68
9	1	5.15	7.02	6.05	8.31	8.18	11.80	8.76	12.05	9.21	11.83
9	2	5.39	7.43	6.24	8.66	8.30	12.00	9.05	12.57	9.27	11.86
9	3	5.53	7.54	8.09	11.41	9.08	13.97	8.58	11.16	8.81	11.41
9	4	6.87	9.60	10.34	14.29	11.15	16.29	10.39	13.54	11.72	15.58
9	5	6.92	9.51	9.75	12.91	10.99	15.25	11.67	15.44	12.73	16.45
9	6	6.67	8.85	9.46	13.22	11.46	15.91	11.52	14.87	12.84	16.76
9	7	7.00	9.33	10.55	14.59	11.36	16.06	11.93	16.11	11.70	15.77
10	0	5.16	6.95	6.80	9.50	9.22	11.78	9.22	11.78	10.60	13.50
10	1	4.97	6.68	6.37	8.86	9.77	13.26	9.77	13.26	9.59	12.05
10	2	5.09	7.03	6.71	9.55	10.27	14.24	10.27	14.24	9.66	12.20
10	3	5.13	7.04	7.04	9.65	9.90	13.20	9.90	13.20	9.69	12.23
10	4	5.88	7.82	10.96	15.45	12.23	16.36	12.23	16.36	12.23	16.96
10	5	6.02	8.04	11.62	16.06	13.74	17.95	13.74	17.95	12.91	17.09
10	6	7.20	9.30	11.46	15.43	13.20	17.07	13.20	17.07	13.54	17.85
10	7	7.41	9.80	11.69	16.42	13.59	18.37	13.59	18.37	12.46	17.14

Chapter 5

Summary and Future Work

5.1 Summary

We have established a method for predicting canal flow for canals in the Sevier River Basin. These efforts have included the following:

- The formulation of potential learning machine inputs from raw measurements of weather and flow found in the SRWUA database.
- A thorough search across a number of model attributes including input set and scale parameter to find models with superior generalizing performance.
- Generation of a framework for testing the tradeoff between prediction time and the delay/advance of the prediction for minimizing the apparent delay in prediction results.
- Determination of prediction capability for up to five-day-ahead prediction with reassessment of model inputs for various prediction times.
- Development of a framework for performing regular updates to a prediction model as data becomes available through the progression of time along with determination of prediction capability for such regularly updated models.
- Comparison of the prediction capability of RVM models to basic models including comparison of the regularly updated RVM model to a regularly updated multi-regressive model.
- Successful application of prediction methods to a second canal in the basin along with a comparison of results for the two canals.

In summary, we have made a search across a multi-dimensional space of model parameters, model attributes and prediction schemes to find canal flow prediction models with minimum error. Such a search cannot be exhaustive though we claim a measure of thoroughness leading to the local minima which we have found. While strictly adhering to data time requirements to ensure declared prediction horizons we have benefitted from the ability to validate various prediction models as afforded by the large set of data—thereby designating specific models as superior based on the comparative validation results. However, application to real-time data does not allow relative evaluation of models except in hindsight. As such, comparative validation aids in model selection only inasmuch as previous evaluations as to models with superior prediction capability have application to the circumstances of interest. General application of a superior model is limited across time and space, that is, a model determined to be superior amongst a set of potential models may not retain that status as applied to a later season or to another canal. Specifically, limitations to general application include appropriateness of model weights for application across canals or over a long time period for a particular canal, the optimal composition of the model input set especially for application across canals but also possibly across large time periods, and value of the scale parameter in application across either time period or canal. The limitation as to model weights can be overcome through the use of a regularly updated RVM model under the understanding that the RVM sets the weights appropriately for the data set presented to the learning process so that by providing the most recent data the RVM is allowed to generate a model (set weight values) which represents the current system. Unfortunately, optimization of input set composition and scale parameter value are not currently afforded by the RVM learning process, which situation motivates much of the experimentation in this thesis. While we have discovered input set compositions that give superior prediction models for each of the two canals of experimentation we have not demonstrated that these inputs sets will provide for models that are superior at other time periods on the same canal (though obtaining good results from testing and training on adjacent years already implies some measure of generalization ability across time, at least for our seasons of choice) or for other canals. This motivates some direction for future work which we discuss presently.

5.2 Future Work

Direct use of our results to guide model selection for real-time applications requires a tolerance for potentially sub-optimal results due to ignorance as to some attributes of the best model (input set, scale factor, etc.) before performing predictions. Overcoming the limitations to general application to real-time data created by the need for selection of the input set and the scale parameter are not easily overcome. The following are suggestions for future work which may lead to a solution for this problem.

One approach would be to perform a thorough search across a set of potential model attributes—similar to the work we have done for the Richfield Canal—for each season and for every canal in the basin (for which data is available in the SRWUA database) and classify each season and canal by the model or models which provide the best prediction capability. The results could then be used to look for patterns across season and canal (time and space) that may lead to a more general choice as to superior models or more guidance in model selection for the situation of interest. This classification could also be corroborated with macroscopic information like the amount of seasonal water availability or the cash crops of the season that might allow for the deduction of related superior model patterns. For example, suppose it was found that low water years generally require models with a large number of flow inputs then this could lead to selection of a model with many flow inputs for a year that was anticipated to have less water.

Another possibility would be to provide an "on the go" model update for attributes such as input set composition. This could be done by generating predictions for a set of potential models simultaneously. Performance for each model can then be established, as actual measurements become available, by using a window of error values computed from the most recent measurements and the predictions of the model. The attributes of the model yielding the best performance under the most recent window of validation are then selected as the attributes of the updated model. This process provides a continuous—though somewhat delayed (based on the length of the validation window)—model update.

A third possibility has to do with the learning process of the RVM. We previously mentioned the possibility of setting the input scale parameter for the Gaussian kernel in an optimal way as well as the possibility of setting an individual scale parameter for each input in the model. Presumably, setting individual scale parameters allows for the possibility of eliminating dependence of the model function on a particular input by setting the corresponding scale parameter to zero. Therefore with an appropriate method for setting individual input scale parameters the optimization over input set composition could be accommodated by including all potential inputs as part of the input set and then by allowing the method to determine whether an input should be excluded from the model. The challenge for this proposition is in establishing the method for setting the individual scale parameters. If the selection of individual input scale parameters—and therefore the composition of the input set—can be included in the RVM learning process then much of the necessary experimentation for determining superior prediction models can be eliminated. This prospect has been investigated to some degree by Tipping [6].

References

[1] B. Berger, R. Hansen, and R. Jensen, “Sevier river basin system description,” Sevier River Water Users Association, Tech. Rep., Jun. 2003 [Online]. Available: `http://www.sevierriver.org/sys_desc/sysdes.pdf`.

[2] B. Berger, R. Hansen, and I. Cowley, “Developing a virtual watershed: The Sevier river basin,” in *Decision Support Systems for Water Resources Management (Specialty Conference).* American Water Resources Association, Jun. 2001.

[3] “Sevierriver.org,” Sevier River Water Users Association [Online]. Available: `http://www.sevierriver.org`.

[4] A. Khalil, M. McKee, M. Kemblowski, and T. Asefa, “Sparse Bayesian learning machine for real-time management of reservoir releases,” *Water Resources Research*, vol. 41, 2005.

[5] R. G. Allen, L. S. Pereira, D. Raes, and M. Smith, “Crop evapotranspiration - guidelines for computing crop water requirements,” Food and Agriculture Organization of the United Nations, Rome, Italy, Irrigation and Drainage Paper 56, 1998.

[6] M. E. Tipping, “Sparse Bayesian learning and the relevance vector machine,” *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[7] M. E. Tipping, “Bayesian inference: An introduction to principles and practice in machine learning,” in *Advanced Lectures on Machine Learning*, pp. 41–62. Berlin: Springer, 2004.

[8] D. J. C. MacKay, “Bayesian interpolation,” *Neural Computation*, vol. 4, pp. 415–447, 1992.

[9] J. Quionero-Candela and L. K. Hansen, “Time series prediction based on the relevance vector machine with adaptive kernels,” in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 985–988, 2002.

[10] T. K. Moon and W. C. Stirling, *Mathematical Methods and Algorithms for Signal Processing*, 1st ed. Upper Saddle River, New Jersey: Prentice Hall, 2000.

Appendices

Appendix A

Computations of the Bayesian Inference

Rather than including the kernel type as one of the parameters in the following derivations, it is assumed that the kernel type is known and fixed. The inputs, however, are included (at first) to allow an understanding of their role in the Bayesian inference. For reference, the specification of the noise process that relates the model to the targets is

$$t_n = y(\mathbf{x}_n, \mathbf{w}) + \epsilon_n.$$

A.1 Prediction through Marginalization

Start with the joint distribution of all the parameters and the training targets, conditioned on the training inputs: $p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2, \mathbf{t}|\mathbf{X})$, where $\mathbf{w}, \boldsymbol{\alpha}, \sigma^2$ are the parameters, namely $\mathbf{w}$ is the vector of weights for the basis functions, $\boldsymbol{\alpha}$is the vector of inverse variances for the weights and σ^2 is the variance of the noise process between model and target and where $\mathbf{t}, \mathbf{X}$ are the training data, namely $\mathbf{t}$ is the vector of targets t_n and $\mathbf{X}$ is a matrix formed from the corresponding input vectors $\mathbf{x}_n$. Decompose this distribution in two ways as

$$p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2, \mathbf{t}|\mathbf{X}) = p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t}, \mathbf{X})p(\mathbf{t}|\mathbf{X}) = p(\mathbf{t}|\mathbf{w}, \boldsymbol{\alpha}, \sigma^2, \mathbf{X})p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{X}),$$

and then solve for the joint distribution of all the unknown parameters given the data (the posterior over the parameters)

$$p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t}, \mathbf{X}) = \frac{p(\mathbf{t}|\mathbf{w}, \boldsymbol{\alpha}, \sigma^2, \mathbf{X})p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{X})}{p(\mathbf{t}|\mathbf{X})}, \tag{A.1}$$

which by omitting the inputs $\mathbf{X}$ from the notation (as in the body of the paper) is the same as (2.6).

For prediction, write the joint distribution of all unknowns given the data, denoted by $p(t_*, \mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t}, \mathbf{X}, \mathbf{x}_*)$, where now the set of unknowns includes the target we are predicting t_*, and the data includes the new input $\mathbf{x}_*$. Marginalizing this distribution over the unknown parameters yields the distribution of the new target given the data (the posterior over the new target):

$$p(t_*|\mathbf{t}, \mathbf{X}, \mathbf{x}_*) = \int p(t_*, \mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t}, \mathbf{X}, \mathbf{x}_*)\, d\mathbf{w}\, d\boldsymbol{\alpha}\, d\sigma^2. \tag{A.2}$$

The distribution in the integral is determined by decomposing as

$$p(t_*, \mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t}, \mathbf{X}, \mathbf{x}_*) = p(t_*|\mathbf{w}, \boldsymbol{\alpha}, \sigma^2, \mathbf{t}, \mathbf{X}, \mathbf{x}_*) p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t}, \mathbf{X}, \mathbf{x}_*),$$

where $p(t_*|\mathbf{w}, \boldsymbol{\alpha}, \sigma^2, \mathbf{t}, \mathbf{X}, \mathbf{x}_*) = p(t_*|\mathbf{w}, \sigma^2, \mathbf{x}_*)$ is the distribution of a single target given the model which is distributed as $\mathcal{N}(t_*|y(\mathbf{x}_*), \sigma^2)$ as in (2.3) and where $p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t}, \mathbf{X}, \mathbf{x}_*)$ is equivalent to the posterior distribution of the parameters $p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t}, \mathbf{X})$ in (A.1), because knowing the new input $\mathbf{x}_*$ without knowing the new target t_* tells us nothing more about the parameters. Substituting these distributions back into the marginalizing integral of (A.2) gives

$$p(t_*|\mathbf{t}, \mathbf{X}, \mathbf{x}_*) = \int p(t_*|\mathbf{w}, \sigma^2, \mathbf{x}_*) p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t}, \mathbf{X})\, d\mathbf{w}\, d\boldsymbol{\alpha}\, d\sigma^2 \tag{A.3}$$

which, by again omitting the inputs ($\mathbf{X}$ and $\mathbf{x}_*$) in the notation, is equivalent to (2.7). This is the distribution from which we desire to select the predicted value. In particular, we would choose the mode of the distribution as the value of the new target. Unfortunately, the posterior over the unknown parameters found in (A.3) and given by (A.1) cannot be determined because the normalizing integral $p(\mathbf{t}|\mathbf{X})$ cannot be computed. Prediction, then, requires an approximation for the posterior over the parameters.

A.2 Parameter Posterior Approximation

To approximate the parameter posterior, decompose as

$$p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t}) = p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2)\, p(\boldsymbol{\alpha}, \sigma^2|\mathbf{t}), \tag{A.4}$$

where for convenience—having seen that our purposes only require distributions conditional on the inputs rather than distributions over the inputs—the inputs are dropped permanently from the notation. The first term can be computed exactly, as we will see later, therefore, approximation is restricted to the second term of the decomposition. This distribution is replaced by a delta function at its mode $\delta(\boldsymbol{\alpha}_{\mathrm{MP}}, \sigma^2_{\mathrm{MP}})$ (with the notation MP indicating the 'most probable' values), which, when combined with the first term, gives the parameter posterior approximation:

$$p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t}) \approx p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2)\,\delta(\boldsymbol{\alpha}_{\mathrm{MP}}, \sigma^2_{\mathrm{MP}}) = p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}_{\mathrm{MP}}, \sigma^2_{\mathrm{MP}})\,\delta(\boldsymbol{\alpha}_{\mathrm{MP}}, \sigma^2_{\mathrm{MP}}).$$

Substituting this approximation into the marginalizing integral of (A.3) (and dropping the inputs from the notation) as follows:

$$\begin{aligned} p(t_*|\mathbf{t}) &= \textstyle\int p(t_*|\mathbf{w}, \sigma^2)\,p(\mathbf{w}, \boldsymbol{\alpha}, \sigma^2|\mathbf{t})\,d\mathbf{w}\,d\boldsymbol{\alpha}\,d\sigma^2 \\ &= \textstyle\int p(t_*|\mathbf{w}, \sigma^2)\,p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2)\,p(\boldsymbol{\alpha}, \sigma^2|\mathbf{t})\,d\mathbf{w}\,d\boldsymbol{\alpha}\,d\sigma^2 \\ &\approx \textstyle\int p(t_*|\mathbf{w}, \sigma^2)\,p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2)\,\delta(\boldsymbol{\alpha}_{\mathrm{MP}}, \sigma^2_{\mathrm{MP}})\,d\mathbf{w}\,d\boldsymbol{\alpha}\,d\sigma^2 \\ &= \textstyle\int p(t_*|\mathbf{w}, \sigma^2_{\mathrm{MP}})\,p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}_{\mathrm{MP}}, \sigma^2_{\mathrm{MP}})\,\delta(\boldsymbol{\alpha}_{\mathrm{MP}}, \sigma^2_{\mathrm{MP}})\,d\mathbf{w}\,d\boldsymbol{\alpha}\,d\sigma^2 \\ &= \textstyle\int p(t_*|\mathbf{w}, \sigma^2_{\mathrm{MP}})\,p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}_{\mathrm{MP}}, \sigma^2_{\mathrm{MP}})\,d\mathbf{w}, \end{aligned}$$

gives

$$p(t_*|\mathbf{t}) \approx p(t_*|\mathbf{t}, \boldsymbol{\alpha}_{\mathrm{MP}}, \sigma^2_{\mathrm{MP}}) = \int p(t_*|\mathbf{w}, \sigma^2_{\mathrm{MP}})\,p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}_{\mathrm{MP}}, \sigma^2_{\mathrm{MP}})\,d\mathbf{w}. \tag{A.5}$$

Prediction, then, requires knowledge of the particular values for $\boldsymbol{\alpha}_{\mathrm{MP}}$ and σ^2_{MP} and the distributions $p(t_*|\mathbf{w}, \sigma^2)$ and $p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2)$ with the conditioning parameters assigned to those values.

The most probable values for the hyperparameters, given by $\boldsymbol{\alpha}_{\mathrm{MP}}$ and σ^2_{MP}, are the values at the mode of the hyperparameter posterior $p(\boldsymbol{\alpha}, \sigma^2|\mathbf{t})$. To determine the values at the mode, start with the joint distribution over the hyperparameters and the targets, decompose in two ways as $p(\boldsymbol{\alpha}, \sigma^2, \mathbf{t}) = p(\boldsymbol{\alpha}, \sigma^2|\mathbf{t})p(\mathbf{t}) = p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2)p(\boldsymbol{\alpha}, \sigma^2)$ and solve for the hyperparameter posterior

$$p(\boldsymbol{\alpha}, \sigma^2|\mathbf{t}) = \frac{p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2)p(\boldsymbol{\alpha}, \sigma^2)}{p(\mathbf{t})}.$$

Again, we find the intractable normalizing integral $p(\mathbf{t})$, revealing why this term must be approximated. However, we can eliminate the intractable denominator and still retain proportionality, which is enough to determine the position of the mode for our approximation. So, with a little further expansion, we have

$$p(\boldsymbol{\alpha}, \sigma^2|\mathbf{t}) \propto p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2)p(\boldsymbol{\alpha})p(\sigma^2),$$

where, due to proportionality, the mode of the right hand side occurs for the same hyperparameter values as the mode of the left hand side. Remembering that the hyperpriors are uniformly distributed, we simply determine the most probable hyperparameter values $\boldsymbol{\alpha}_{\mathrm{MP}}$ and σ^2_{MP} from the values that maximize (or are at the mode of) $p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2)$.

A.3 Determining the Marginal Likelihood and the Weight Posterior

This distribution, known as the marginal likelihood, is determined together with the posterior over the weights $p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2)$, by starting with the distribution over the weights and targets conditioned on the hyperparameters and decomposing in two ways as

$$p(\mathbf{t}, \mathbf{w}|\boldsymbol{\alpha}, \sigma^2) = p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2)p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2) = p(\mathbf{t}|\mathbf{w}, \boldsymbol{\alpha}, \sigma^2)p(\mathbf{w}|\boldsymbol{\alpha}, \sigma^2).$$

The second decomposition can be simplified by removing all the conditioning variables that do not influence the distributions, giving

$$p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2)p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2) = p(\mathbf{t}|\mathbf{w}, \sigma^2)p(\mathbf{w}|\boldsymbol{\alpha}),$$

in which we can recognize the two desired distributions on the left and two known Gaussian distributions, namely the likelihood of the data $p(\mathbf{t}|\mathbf{w}, \sigma^2)$ given in (2.4) and the prior over the weights $p(\mathbf{w}|\boldsymbol{\alpha})$ given in (2.5), on the right. The two desired distributions are determined from the known distributions by first forming a Gaussian distribution (the weight posterior) using all the terms in $\mathbf{w}$, and secondly forming another Gaussian (the marginal likelihood) with all the remaining terms. This is done by following the suggestion given by Tipping [6] to collect terms in $\mathbf{w}$ in the combined exponent of the known distributions and complete the square to obtain the weight posterior. The newly introduced terms in $\mathbf{t}$ give the exponent of the marginal likelihood.

To derive this, write the likelihood of the data as

$$\begin{aligned} p(\mathbf{t}|\mathbf{w},\sigma^2) &= (2\pi\sigma^2)^{-N/2}\exp\left\{-\tfrac{1}{2\sigma^2}\left\|\mathbf{t}-\boldsymbol{\Phi}\mathbf{w}\right\|^2\right\} \\ &= \tfrac{1}{(2\pi\sigma^2)^{N/2}}\exp\left\{-\tfrac{1}{2\sigma^2}(\mathbf{t}-\boldsymbol{\Phi}\mathbf{w})^{\mathrm{T}}(\mathbf{t}-\boldsymbol{\Phi}\mathbf{w})\right\}, \end{aligned}$$

and the weight prior as

$$\begin{aligned} p(\mathbf{w}|\boldsymbol{\alpha}) &= \textstyle\prod_{i=0}^{N}\mathcal{N}(w_i|0,\alpha_i^{-1}) \\ &= \textstyle\prod_{i=0}^{N}\left(\frac{\alpha_i}{2\pi}\right)^{1/2}\exp\left\{-\frac{\alpha_i}{2}w_i^2\right\} \\ &= \tfrac{1}{(2\pi)^{(N+1)/2}}\left(\textstyle\prod_{i=0}^{N}\alpha_i\right)^{1/2}\exp\left\{-\tfrac{1}{2}\textstyle\sum_{i=0}^{N}\alpha_i w_i^2\right\} \\ &= \tfrac{1}{(2\pi)^{(N+1)/2}}\tfrac{1}{|\mathbf{A}^{-1}|^{1/2}}\exp\left\{-\tfrac{1}{2}\mathbf{w}^{\mathrm{T}}\mathbf{A}\mathbf{w}\right\}, \end{aligned}$$

where $\mathbf{A} = \mathrm{diag}(\alpha_0, \alpha_1, \ldots, \alpha_N)$ so that the product with the exponents combined is

$$\begin{aligned} p(\mathbf{t}|\mathbf{w},\sigma^2)p(\mathbf{w}|\boldsymbol{\alpha}) = &\frac{1}{(2\pi\sigma^2)^{N/2}}\frac{1}{(2\pi)^{(N+1)/2}}\frac{1}{|\mathbf{A}^{-1}|^{1/2}} \\ &\times\exp\left\{-\frac{1}{2}\left[\frac{1}{\sigma^2}(\mathbf{t}-\boldsymbol{\Phi}\mathbf{w})^{\mathrm{T}}(\mathbf{t}-\boldsymbol{\Phi}\mathbf{w})+\mathbf{w}^{\mathrm{T}}\mathbf{A}\mathbf{w}\right]\right\}. \end{aligned}$$

Take the quantity in the exponent, expand, and combine terms in $\mathbf{w}$ as

$$\begin{aligned} \text{Exponent} &= -\tfrac{1}{2}\left[\tfrac{1}{\sigma^2}(\mathbf{t}-\boldsymbol{\Phi}\mathbf{w})^{\mathrm{T}}(\mathbf{t}-\boldsymbol{\Phi}\mathbf{w})+\mathbf{w}^{\mathrm{T}}\mathbf{A}\mathbf{w}\right] \\ &= -\tfrac{1}{2}\left[\tfrac{1}{\sigma^2}\mathbf{t}^{\mathrm{T}}\mathbf{t}-\tfrac{2}{\sigma^2}\mathbf{w}^{\mathrm{T}}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}+\tfrac{1}{\sigma^2}\mathbf{w}^{\mathrm{T}}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\mathbf{w}+\mathbf{w}^{\mathrm{T}}\mathbf{A}\mathbf{w}\right] \\ &= -\tfrac{1}{2}\left[\mathbf{w}^{\mathrm{T}}\left(\mathbf{A}+\tfrac{1}{\sigma^2}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right)\mathbf{w}-\tfrac{2}{\sigma^2}\mathbf{w}^{\mathrm{T}}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}+\tfrac{1}{\sigma^2}\mathbf{t}^{\mathrm{T}}\mathbf{t}\right]. \end{aligned}$$

Then complete the square in $\mathbf{w}$ and combine the newly introduced terms in $\mathbf{t}$ which gives,

$$\text{Exponent} = -\tfrac{1}{2}\left[(\mathbf{w}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{w}-\boldsymbol{\mu}) + \mathbf{t}^{\mathrm{T}}\left(\tfrac{1}{\sigma^2}\mathbf{I} - \tfrac{1}{\sigma^4}\boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Phi}^{\mathrm{T}}\right)\mathbf{t}\right],$$

where $\boldsymbol{\Sigma} = \left(\mathbf{A} + \sigma^{-2}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right)^{-1}$ and $\boldsymbol{\mu} = \sigma^{-2}\boldsymbol{\Sigma}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}$. Using a generalized version of Woodbury's identity [10] rewrite the terms in $\mathbf{t}$ to give

$$\text{Exponent} = -\tfrac{1}{2}\left[(\mathbf{w}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{w}-\boldsymbol{\mu}) + \mathbf{t}^{\mathrm{T}}\left(\sigma^2\mathbf{I} + \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\right)^{-1}\mathbf{t}\right],$$

which contains the exponents of the two desired Gaussian distributions. To complete the derivation take the scaling terms of the distribution product and divide out the appropriate scaling terms for the Gaussian in $\mathbf{w}$, which leaves the appropriate scaling terms of the Gaussian in $\mathbf{t}$:

$$\begin{aligned}\text{Scales} &= \frac{1}{(2\pi\sigma^2)^{N/2}}\frac{1}{(2\pi)^{(N+1)/2}}\frac{1}{|\mathbf{A}^{-1}|^{1/2}}\\ &= \frac{1}{(2\pi\sigma^2)^{N/2}}\frac{1}{(2\pi)^{(N+1)/2}}\frac{1}{|\mathbf{A}^{-1}|^{1/2}}\frac{(2\pi)^{(N+1)/2}|\boldsymbol{\Sigma}|^{1/2}}{(2\pi)^{(N+1)/2}|\boldsymbol{\Sigma}|^{1/2}}\\ &= \left(\frac{1}{(2\pi\sigma^2)^{N/2}}\frac{|\boldsymbol{\Sigma}|^{1/2}}{|\mathbf{A}^{-1}|^{1/2}}\right)\left(\frac{1}{(2\pi)^{(N+1)/2}|\boldsymbol{\Sigma}|^{1/2}}\right).\end{aligned}$$

To verify that the scaling terms are correct for the Gaussian in $\mathbf{t}$ manipulate the first scale term as

$$\begin{aligned}\text{Scale} &= \frac{1}{(2\pi\sigma^2)^{N/2}}\frac{|\boldsymbol{\Sigma}|^{1/2}}{|\mathbf{A}^{-1}|^{1/2}}\\ &= \frac{1}{(2\pi)^{N/2}}\frac{1}{(\sigma^2)^{N/2}}\frac{1}{|\mathbf{A}^{-1}\boldsymbol{\Sigma}^{-1}|^{1/2}}\\ &= \frac{1}{(2\pi)^{N/2}}\frac{1}{|\sigma^2\mathbf{I}_N|^{1/2}}\frac{1}{\left|\mathbf{I}_M+\sigma^{-2}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right|^{1/2}}\\ &= \frac{1}{(2\pi)^{N/2}}\frac{1}{|\sigma^2\mathbf{I}_N|^{1/2}}\frac{1}{\left|\mathbf{I}_N+\sigma^{-2}\boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\right|^{1/2}}\\ &= \frac{1}{(2\pi)^{N/2}}\frac{1}{\left|\sigma^2\mathbf{I}_N+\boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\right|^{1/2}}.\end{aligned}$$

Then write the full result splitting exponent and scale terms as

$$\begin{aligned}p(\mathbf{t}|\mathbf{w},\sigma^2)p(\mathbf{w}|\boldsymbol{\alpha}) = &\frac{1}{(2\pi)^{N/2}}\frac{1}{\left|\sigma^2\mathbf{I}+\boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\right|^{1/2}}\exp\left\{-\tfrac{1}{2}\mathbf{t}^{\mathrm{T}}\left(\sigma^2\mathbf{I}+\boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\right)^{-1}\mathbf{t}\right\}\\ &\times\frac{1}{(2\pi)^{(N+1)/2}|\boldsymbol{\Sigma}|^{1/2}}\exp\left\{-\tfrac{1}{2}(\mathbf{w}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{w}-\boldsymbol{\mu})\right\}\end{aligned}$$

to give the two desired distributions: the marginal likelihood

$$p(\mathbf{t}|\mathbf{w},\sigma^2) = \frac{1}{(2\pi)^{N/2}}\frac{1}{\left|\sigma^2\mathbf{I}+\mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^{\mathrm{T}}\right|^{1/2}}\exp\left\{-\frac{1}{2}\mathbf{t}^{\mathrm{T}}\left(\sigma^2\mathbf{I}+\mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^{\mathrm{T}}\right)^{-1}\mathbf{t}\right\}, \qquad \text{(A.6)}$$

and the weight posterior

$$p(\mathbf{w}|\mathbf{t},\boldsymbol{\alpha},\sigma^2) = \frac{1}{(2\pi)^{(N+1)/2}\left|\mathbf{\Sigma}\right|^{1/2}}\exp\left\{-\frac{1}{2}\left(\mathbf{w}-\boldsymbol{\mu}\right)^{\mathrm{T}}\mathbf{\Sigma}^{-1}\left(\mathbf{w}-\boldsymbol{\mu}\right)\right\}, \qquad \text{(A.7)}$$

where $\mathbf{\Sigma} = \left(\mathbf{A}+\sigma^{-2}\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}$ and $\boldsymbol{\mu} = \sigma^{-2}\mathbf{\Sigma}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$.

A.4 The Predictive Distribution

We recognize that given the weights and the variance of the noise the new target has the same distribution as any other target, that is

$$p(t_*|\mathbf{w},\sigma^2) \sim \mathcal{N}(t_*|y(\mathbf{x}_*),\sigma^2),$$

as in (2.3) or

$$p(t_*|\mathbf{w},\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left\{-\frac{1}{2\sigma^2}\left(t_*-\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_*)\right)^2\right\}. \qquad \text{(A.8)}$$

Knowing this distribution, having determined the posterior over the weights in (A.7) and having derived a method for estimating the particular values for the hyperparameters $\boldsymbol{\alpha}_{\mathrm{MP}}$ and σ^2_{MP} using the marginal likelihood in (A.6) (see Appendix B for the derivation) we are now prepared to make predictions by computing the integral in (A.5)

$$p(t_*|\mathbf{t}) \approx p(t_*|\mathbf{t},\boldsymbol{\alpha}_{\mathrm{MP}},\sigma^2_{\mathrm{MP}})) = \int p(t_*|\mathbf{w},\sigma^2_{\mathrm{MP}})\,p(\mathbf{w}|\mathbf{t},\boldsymbol{\alpha}_{\mathrm{MP}},\sigma^2_{\mathrm{MP}})\,d\mathbf{w}.$$

We start by taking the product of the posterior and the distribution over the new target where both distributions are conditioned on the estimated hyperparameter values. Our object is to determine two probability density functions from the product, one a distribution in t_* which does not depend on $\mathbf{w}$ and can, therefore, be pulled out of the integral, and the other a distribution in $\mathbf{w}$ that goes to one in the integral over all $\mathbf{w}$, leaving just the distribution in t_*, which is the distribution of interest, that is, the predictive distribution. We start by ignoring the scale terms and looking exclusively at the product of the exponential factors. With reference to (A.7) and (A.8) we have

$$\exp\left\{-\frac{1}{2}\left[(\mathbf{w}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{w}-\boldsymbol{\mu})+\frac{1}{\sigma^2}\left(t_*-\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_*)\right)^2\right]\right\}.$$

Now taking just the exponent (excluding the $-\frac{1}{2}$ scale) labeled as e, and expanding we have

$$e=\mathbf{w}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\mathbf{w}-2\mathbf{w}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}+\boldsymbol{\mu}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}+\frac{t_*^2}{\sigma^2}-\frac{2\mathbf{w}^{\mathrm{T}}t_*\boldsymbol{\phi}(\mathbf{x}_*)}{\sigma^2}+\frac{\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_*)\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\mathbf{w}}{\sigma^2}.$$

Collecting terms in $\mathbf{w}$ gives

$$\begin{aligned} e &= \mathbf{w}^{\mathrm{T}}\left(\underbrace{\boldsymbol{\Sigma}^{-1}+\frac{\boldsymbol{\phi}(\mathbf{x}_*)\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}}{\sigma^2}}_{\mathbf{R}^{-1}}\right)\mathbf{w}-2\mathbf{w}^{\mathrm{T}}\left(\underbrace{\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}+\frac{t_*\boldsymbol{\phi}(\mathbf{x}_*)}{\sigma^2}}_{\mathbf{b}}\right)+\underbrace{\boldsymbol{\mu}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}+\frac{t_*^2}{\sigma^2}}_{c} \\ &= \mathbf{w}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{w}-2\mathbf{w}^{\mathrm{T}}\mathbf{b}+c, \end{aligned}$$

where the coefficients of $\mathbf{w}$ are labeled to facilitate the next step, which is completing the square in $\mathbf{w}$. The perfect square for a sum or difference of vector variables $\mathbf{x}$ and $\mathbf{y}$ with a non-identity weighting matrix $\mathbf{A}$ has form $\mathbf{x}^{\mathrm{T}}\mathbf{A}\mathbf{x}\pm 2\mathbf{x}^{\mathrm{T}}\mathbf{A}\mathbf{y}+\mathbf{y}^{\mathrm{T}}\mathbf{A}\mathbf{y}$ and can be factored as $(\mathbf{x}\pm\mathbf{y})^{\mathrm{T}}\mathbf{A}(\mathbf{x}\pm\mathbf{y})$. In our case we must complete a square involving the vector $\mathbf{w}$ with weighting matrix $\mathbf{R}^{-1}$. Choosing $\mathbf{z}$ to represent the second vector and choosing to complete the square of a difference based on the current form of our exponent, the square will have form

$$\mathbf{w}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{w}-2\mathbf{w}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{z}+\mathbf{z}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{z}. \tag{A.9}$$

Obtaining this form requires that we introduce the matrix $\mathbf{R}^{-1}$ in the second term of the exponent. We do this by inserting the identity matrix $\mathbf{R}^{-1}\mathbf{R}$ in the second term as

$$e=\mathbf{w}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{w}-2\mathbf{w}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{R}\mathbf{b}+c.$$

Then we see by comparison of our second term with the second term of the form given in (A.9) that $\mathbf{z} = \mathbf{R}\mathbf{b}$. Making this substitution we have

$$e = \mathbf{w}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{w} - 2\mathbf{w}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{z} + c.$$

To complete the square we must introduce the term $\mathbf{z}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{z}$. We do this by adding and subtracting the term to get

$$e = \mathbf{w}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{w} - 2\mathbf{w}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{z} + \mathbf{z}\mathbf{R}^{-1}\mathbf{z} + \underbrace{c - \mathbf{z}\mathbf{R}^{-1}\mathbf{z}}_{d},$$

where for clarity we label all terms that are not part of the square as d. Factoring the square that we have formed gives

$$e = (\mathbf{w} - \mathbf{z})^{\mathrm{T}}\mathbf{R}^{-1}(\mathbf{w} - \mathbf{z}) + d,$$

where $\mathbf{R}$ is the covariance of a Gaussian distribution in $\mathbf{w}$, the exponent of which we have just formed, and $\mathbf{z}$ is the mean of the distribution, given by $\mathbf{z} = \mathbf{R}\mathbf{b}$, as introduced through completion of the square. The term d, formed from all of the constants (with respect to $\mathbf{w}$) that remain, is given by $d = c - \mathbf{z}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{z}$. We continue by manipulating this term with the purpose of making it look like the exponent of a Gaussian distribution in $\mathbf{t}$. This manipulation is detailed below, starting with

$$\begin{aligned} d &= c - \mathbf{z}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{z} \\ &= c - \mathbf{b}^{\mathrm{T}}\mathbf{R}\mathbf{R}^{-1}\mathbf{R}\mathbf{b} \\ &= c - \mathbf{b}^{\mathrm{T}}\mathbf{R}\mathbf{b} \\ &= c - \mathbf{b}^{\mathrm{T}}\left(\boldsymbol{\Sigma}^{-1} + \frac{\boldsymbol{\phi}(\mathbf{x}_*)\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}}{\sigma^2}\right)^{-1}\mathbf{b}. \end{aligned}$$

Using Woodbury's identity we can rewrite the inverse term to give

$$d = c - \mathbf{b}^{\mathrm{T}}\left(\boldsymbol{\Sigma} - \boldsymbol{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*)\left(\sigma^2 + \boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*)\right)^{-1}\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\right)\mathbf{b}.$$

Then making the substitution $\sigma_*^2 = \sigma^2 + \boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*)$ we have

$$\begin{aligned} d &= c - \mathbf{b}^{\mathrm{T}}\left(\boldsymbol{\Sigma} - \boldsymbol{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*)\frac{1}{\sigma_*^2}\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\right)\mathbf{b} \\ &= c - \mathbf{b}^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{b} + \mathbf{b}^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*)\frac{1}{\sigma_*^2}\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{b} \\ &= c - \mathbf{b}^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{b} + \frac{1}{\sigma_*^2}\left(\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{b}\right)^2 . \end{aligned}$$

The quantity $\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{b}$ is expanded and manipulated as

$$\begin{aligned} \boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{b} &= \boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\left(\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \frac{t_*\boldsymbol{\phi}(\mathbf{x}_*)}{\sigma^2}\right) \\ &= \boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\mu} + \frac{t_*}{\sigma^2}\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*) \\ &= \boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\mu} - t_* + t_* + \frac{t_*}{\sigma^2}\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*) \\ &= -\left(t_* - \boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\mu}\right) + \frac{t_*}{\sigma^2}\left(\sigma^2 + \boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*)\right) \\ &= -\left(t_* - y_*\right) + \frac{t_*}{\sigma^2}\sigma_*^2 \\ &= -\left(\left(t_* - y_*\right) - \frac{t_*}{\sigma^2}\sigma_*^2\right), \end{aligned}$$

where we have made the substitutions $y_* = \boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\mu}$ and again $\sigma_*^2 = \sigma^2 + \boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*)$. If we substitute this quantity for $\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{b}$ we have

$$\begin{aligned} d &= c - \mathbf{b}^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{b} + \frac{1}{\sigma_*^2}\left(\left(t_* - y_*\right) - \frac{t_*}{\sigma^2}\sigma_*^2\right)^2 \\ &= c - \mathbf{b}^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{b} + \frac{1}{\sigma_*^2}\left(\left(t_* - y_*\right)^2 - 2\left(t_* - y_*\right)\frac{t_*}{\sigma^2}\sigma_*^2 + \frac{t_*^2}{\sigma^4}\sigma_*^4\right) \\ &= c - \mathbf{y}^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{y} + \frac{1}{\sigma_*^2}\left(t_* - y_*\right)^2 - 2\left(t_* - y_*\right)\frac{t_*}{\sigma^2} + \frac{t_*^2}{\sigma^4}\sigma_*^2 \\ &= \frac{1}{\sigma_*^2}\left(t_* - y_*\right)^2 + \underbrace{c - \mathbf{b}^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{b} - 2\frac{t_*}{\sigma^2}\left(t_* - y_*\right) + \frac{t_*^2}{\sigma^4}\sigma_*^2}_{d_0} \\ &= \frac{1}{\sigma_*^2}\left(t_* - y_*\right)^2 , \end{aligned} \tag{A.10}$$

where the last step follows from the fact that the final four terms, labeled as d_0, can be shown to sum to zero after expanding $c - \mathbf{b}^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{b}$ and manipulating as

$$\begin{aligned}
c - \mathbf{b}^{\mathrm{T}}\boldsymbol{\Sigma}\mathbf{b} &= \boldsymbol{\mu}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \frac{t_*^2}{\sigma^2} - \left(\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \frac{t_*\boldsymbol{\phi}(\mathbf{x}_*)}{\sigma^2}\right)^{\mathrm{T}} \boldsymbol{\Sigma} \left(\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \frac{t_*\boldsymbol{\phi}(\mathbf{x}_*)}{\sigma^2}\right) \\
&= \boldsymbol{\mu}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \frac{t_*^2}{\sigma^2} - \left(\boldsymbol{\mu}^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + 2\frac{t_*}{\sigma^2}\boldsymbol{\mu}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_*) + \frac{t_*^2}{\sigma^4}\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*)\right) \\
&= \frac{t_*^2}{\sigma^2} - 2\frac{t_*}{\sigma^2}\boldsymbol{\mu}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_*) - \frac{t_*^2}{\sigma^4}\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*) \\
&= \frac{t_*^2}{\sigma^2} + \frac{t_*^2}{\sigma^2} - \frac{t_*^2}{\sigma^2} - 2\frac{t_*}{\sigma^2}\boldsymbol{\mu}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_*) - \frac{t_*^2}{\sigma^4}\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*) \\
&= 2\frac{t_*^2}{\sigma^2} - 2\frac{t_*}{\sigma^2}\boldsymbol{\mu}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_*) - \frac{t_*^2}{\sigma^2} - \frac{t_*^2}{\sigma^4}\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*) \\
&= 2\frac{t_*}{\sigma^2}\left(t_* - \boldsymbol{\mu}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_*)\right) - \frac{t_*^2}{\sigma^4}\left(\sigma^2 - \boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*)\right) \\
&= 2\frac{t_*}{\sigma^2}\left(t_* - y_*\right) - \frac{t_*^2}{\sigma^4}\sigma_*^2.
\end{aligned}$$

The result for d in (A.10) is the exponent (excluding the $-\frac{1}{2}$ scale) of the predictive distribution $p(t_*|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2)$. We can also show the scale term of the desired distribution by dividing the scale term of the distribution we formed in $\mathbf{w}$ from the product of the scale terms of the original two distributions (A.7) and (A.8). To show this we write the product of the two scale terms and the inverse of the scale term for the distribution formed in $\mathbf{w}$, which product we label s, and manipulate as

$$\begin{aligned}
s &= \left(\frac{1}{(2\pi)^{\frac{N+1}{2}}|\mathbf{\Sigma}|^{\frac{1}{2}}}\right)\left(\frac{1}{(2\pi)^{\frac{1}{2}}(\sigma^2)^{\frac{1}{2}}}\right)\left(\frac{(2\pi)^{\frac{N+1}{2}}|\mathbf{R}|^{\frac{1}{2}}}{1}\right)\\
&= \frac{1}{(2\pi)^{\frac{1}{2}}(\sigma^2)^{\frac{1}{2}}|\mathbf{\Sigma}|^{\frac{1}{2}}|\mathbf{R}^{-1}|^{\frac{1}{2}}}\\
&= \frac{1}{(2\pi)^{\frac{1}{2}}\left[\sigma^2|\mathbf{\Sigma}|\cdot|\mathbf{R}^{-1}|\right]^{\frac{1}{2}}}\\
&= \frac{1}{(2\pi)^{\frac{1}{2}}\left[\sigma^2\det(\mathbf{\Sigma})\det\bigl(\mathbf{\Sigma}^{-1}+\frac{1}{\sigma^2}\boldsymbol{\phi}(\mathbf{x}_*)\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\bigr)\right]^{\frac{1}{2}}}\\
&= \frac{1}{(2\pi)^{\frac{1}{2}}\left[\sigma^2\det\bigl(\mathbf{\Sigma}\left(\mathbf{\Sigma}^{-1}+\frac{1}{\sigma^2}\boldsymbol{\phi}(\mathbf{x}_*)\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\right)\bigr)\right]^{\frac{1}{2}}}\\
&= \frac{1}{(2\pi)^{\frac{1}{2}}\left[\sigma^2\det\bigl(\mathbf{I}+\frac{1}{\sigma^2}\mathbf{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*)\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\bigr)\right]^{\frac{1}{2}}}\\
&= \frac{1}{(2\pi)^{\frac{1}{2}}\left[\sigma^2\left(1+\mathrm{tr}\bigl(\frac{1}{\sigma^2}\mathbf{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*)\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\bigr)\right)\right]^{\frac{1}{2}}}\\
&= \frac{1}{(2\pi)^{\frac{1}{2}}\left[\sigma^2+\mathrm{tr}\left(\mathbf{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*)\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\right)\right]^{\frac{1}{2}}}\\
&= \frac{1}{(2\pi)^{\frac{1}{2}}\left[\sigma^2+\mathrm{tr}\left(\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\mathbf{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*)\right)\right]^{\frac{1}{2}}}\\
&= \frac{1}{(2\pi)^{\frac{1}{2}}\left[\sigma^2+\boldsymbol{\phi}(\mathbf{x}_*)^{\mathrm{T}}\mathbf{\Sigma}\boldsymbol{\phi}(\mathbf{x}_*)\right]^{\frac{1}{2}}}\\
&= \frac{1}{(2\pi)^{\frac{1}{2}}\left(\sigma_*^2\right)^{\frac{1}{2}}}\\
&= \frac{1}{\sqrt{2\pi\sigma_*^2}},
\end{aligned}$$

where we use the identity $\det(\mathbf{I}+\mathbf{A}) = 1+\mathrm{tr}(\mathbf{A})$ for a matrix $\mathbf{A}$ with rank equal to one. The result is the scale term for the desired distribution of the new target t_*. Combining the scale term and exponent gives the distribution of the new target given the data and the hyperparameters

$$p(t_*|\mathbf{t}) \approx p(t_*|\mathbf{t},\boldsymbol{\alpha},\sigma^2) = \frac{1}{\sqrt{2\pi\sigma_*^2}}\exp\left\{-\frac{1}{2\sigma_*^2}\left(t_*-y_*\right)^2\right\}. \qquad \text{(A.11)}$$

Appendix B

Computations of the Hyperparameter Estimation

For the iterative re-estimation of the hyperparameters that maximize the hyperparameter posterior $p(\boldsymbol{\alpha}, \sigma^2|\mathbf{t})$, appropriate update equations can be determined using derivatives. In Chapter 2 we said that the hyperparameter posterior was proportional to the product of the marginal likelihood and the hyperpriors, so that with uniform hyperpriors, the maximization of the posterior was equivalent to the maximization of the likelihood. Here, we retain the possibility of non-uniform hyperpriors in the derivation of update equations and therefore maximize the product $p(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2)p(\boldsymbol{\alpha})p(\sigma^2)$. Then we show the result for uniform hyperpriors by setting $a=b=c=d=0$. For convenience we choose to maximize the logarithm of the product, with hyperiors that are over the logarithm of the hyperparameters. Pursuant to this we write the log objective function $\mathcal{L} = \log p(\mathbf{t}|\log\boldsymbol{\alpha}, \log\beta)\, p(\log\boldsymbol{\alpha})\, p(\log\beta)$. Expansion of this function gives

$$\begin{aligned}\mathcal{L} &= \log p(\mathbf{t}|\log\boldsymbol{\alpha}, \log\beta)\, p(\log\boldsymbol{\alpha})\, p(\log\beta) \\ &= \log p(\mathbf{t}|\log\boldsymbol{\alpha}, \log\beta) + \log\left(\prod_{i=0}^{N} p(\log\alpha_i)\right) + \log p(\log\beta) \\ &= \log p(\mathbf{t}|\log\boldsymbol{\alpha}, \log\beta) + \sum_{i=0}^{N} \log p(\log\alpha_i) + \log p(\log\beta). \qquad \text{(B.1)}\end{aligned}$$

As the function is further expanded, any terms appearing which are not functions of either of the hyperparameters will be preemptively dropped from the objective function, as such terms will go to zero in the derivatives. The first term of (B.1), which is specified by (2.10), can be expanded as

$$\log p(\mathbf{t}|\log\boldsymbol{\alpha}, \log\beta) = -\frac{N}{2}\log(2\pi) - \frac{1}{2}\left[\log|\beta^{-1}\mathbf{I} + \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathrm{T}}| + \mathbf{t}^{\mathrm{T}}(\beta^{-1}\mathbf{I} + \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{\mathrm{T}})^{-1}\mathbf{t}\right], \qquad \text{(B.2)}$$

where the first term is not a function of the hyperparameters. To simplify the derivatives we seek to rewrite the two remaining terms. Using the determinant identity

$$|\mathbf{A}|\left|\beta^{-1}\mathbf{I}+\mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^{\mathrm{T}}\right| = \left|\beta^{-1}\mathbf{I}\right|\left|\mathbf{A}+\beta\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right|,$$

taking the logarithm of both sides, expanding, and solving for $\log\left|\beta^{-1}\mathbf{I}+\mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^{\mathrm{T}}\right|$ we can write

$$\begin{aligned}
\log|\beta^{-1}\mathbf{I}+\mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^{\mathrm{T}}| &= \log|\beta^{-1}\mathbf{I}| + \log|\mathbf{A}+\beta\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}| - \log|\mathbf{A}| \\
&= \log\beta^{-N} + \log|\mathbf{\Sigma}^{-1}| - \log|\mathbf{A}| \\
&= -N\log\beta - \log|\mathbf{\Sigma}| - \log|\mathbf{A}|. && \text{(B.3)}
\end{aligned}$$

Using Woodbury's identity $(\beta^{-1}\mathbf{I}+\mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi})^{-1} = \beta\mathbf{I} - \beta\mathbf{\Phi}(\mathbf{A}+\beta\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi})^{-1}\mathbf{\Phi}^{\mathrm{T}}\beta$, and the substitutions $\mathbf{\Sigma} = \left(\mathbf{A}+\beta\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}$ and $\boldsymbol{\mu} = \beta\mathbf{\Sigma}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}$, we can write

$$\begin{aligned}
\mathbf{t}^{\mathrm{T}}(\beta^{-1}\mathbf{I}+\mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^{\mathrm{T}})^{-1}\mathbf{t} &= \beta\mathbf{t}^{\mathrm{T}}\mathbf{t} - \beta\mathbf{t}^{\mathrm{T}}\mathbf{\Phi}(\mathbf{A}+\beta\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi})^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}\beta \\
&= \beta\mathbf{t}^{\mathrm{T}}\mathbf{t} - \beta\mathbf{t}^{\mathrm{T}}\mathbf{\Phi}\mathbf{\Sigma}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}\beta \\
&= \beta\mathbf{t}^{\mathrm{T}}\left(\mathbf{t} - \mathbf{\Phi}\beta\mathbf{\Sigma}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}\right) \\
&= \beta\mathbf{t}^{\mathrm{T}}\left(\mathbf{t} - \mathbf{\Phi}\boldsymbol{\mu}\right) && \text{(B.4)} \\
&= \beta\left(\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}\right)^{\mathrm{T}}\left(\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}\right) + \beta\left(\mathbf{\Phi}\boldsymbol{\mu}\right)^{\mathrm{T}}\left(\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}\right) \\
&= \beta||\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}||^2 + \beta\boldsymbol{\mu}^{\mathrm{T}}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t} - \beta\boldsymbol{\mu}^{\mathrm{T}}\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\boldsymbol{\mu} \\
&= \beta||\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}||^2 + \boldsymbol{\mu}^{\mathrm{T}}\mathbf{\Sigma}^{-1}\beta\mathbf{\Sigma}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t} - \boldsymbol{\mu}^{\mathrm{T}}\beta\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\boldsymbol{\mu} \\
&= \beta||\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}||^2 + \boldsymbol{\mu}^{\mathrm{T}}\mathbf{\Sigma}^{-1}\boldsymbol{\mu} - \boldsymbol{\mu}^{\mathrm{T}}\beta\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\boldsymbol{\mu} \\
&= \beta||\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}||^2 + \boldsymbol{\mu}^{\mathrm{T}}\left(\mathbf{\Sigma}^{-1} - \beta\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)\boldsymbol{\mu} \\
&= \beta||\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}||^2 + \boldsymbol{\mu}^{\mathrm{T}}\mathbf{A}\boldsymbol{\mu}. && \text{(B.5)}
\end{aligned}$$

Using the fact that $p(\log\alpha) = \alpha p(\alpha)$, each of the terms $\log p(\log\alpha)$ in (B.1) can be expanded as

$$\begin{aligned}
\log p(\log\alpha) &= \log \alpha p(\alpha) \\
&= \log\alpha + \log p(\alpha) \\
&= \log\alpha + \log \mathrm{Gamma}(\alpha|a,b) \\
&= \log\alpha + \log\left(\Gamma(a)^{-1} b^a \alpha^{a-1} e^{-b\alpha}\right) \\
&= \log\alpha - \log\Gamma(a) + a\log b + (a-1)\log\alpha - b\alpha \\
&= -\log\Gamma(a) + a\log b + a\log\alpha - b\alpha,
\end{aligned}$$

where here α represents any of the scalar hyperparameters α_i or β. Removing any of the additive terms not a function of the hyperparameter and replacing (α, a, b) with (α_i, a, b) or (β, c, d), respectively, gives $(a\log\alpha_i - b\alpha_i)$ and $(c\log\beta - d\beta)$.

Combining all the terms back into the objective function gives

$$\begin{aligned}
\mathcal{L} = \frac{1}{2}\Bigg[&\underbrace{N\log\beta}_{f(\beta)} + \underbrace{\log|\mathbf{\Sigma}|}_{f(\alpha_i,\beta)} + \underbrace{\log|\mathbf{A}|}_{f(\alpha_i)} - \underbrace{\beta\|\mathbf{t} - \mathbf{\Phi}\boldsymbol{\mu}\|^2}_{f(\beta)} - \underbrace{\boldsymbol{\mu}^{\mathrm{T}}\mathbf{A}\boldsymbol{\mu}}_{f(\alpha_i)}\Bigg] \\
&+ \textstyle\sum_{i=0}^{N} \underbrace{(a\log\alpha_i - b\alpha_i)}_{f(\alpha_i)} + \underbrace{c\log\beta - d\beta}_{f(\beta)},
\end{aligned}$$

where the under-braces for each term note whether the term is a function of α_i or β or both so that the derivatives can be given by

$$\frac{\partial\mathcal{L}}{\partial\log\alpha_i} = \frac{\partial\mathcal{L}_\alpha}{\partial\log\alpha_i} = \frac{\partial}{\partial\log\alpha_i}\left(\frac{1}{2}\left[\log|\mathbf{\Sigma}| + \log|\mathbf{A}| - \boldsymbol{\mu}^{\mathrm{T}}\mathbf{A}\boldsymbol{\mu}\right] + \textstyle\sum_{i=0}^{N}(a\log\alpha_i - b\alpha_i)\right),$$

and

$$\frac{\partial\mathcal{L}}{\partial\log\beta} = \frac{\partial\mathcal{L}_\beta}{\partial\log\beta} = \frac{\partial}{\partial\log\beta}\left(\frac{1}{2}\left[N\log\beta + \log|\mathbf{\Sigma}| - \beta\|\mathbf{t} - \mathbf{\Phi}\boldsymbol{\mu}\|^2\right] + c\log\beta - d\beta\right),$$

where

$$\mathcal{L}_\alpha = \frac{1}{2}\left[-\log|\mathbf{\Sigma^{-1}}| + \log|\mathbf{A}| - \boldsymbol{\mu}^{\mathrm{T}}\mathbf{A}\boldsymbol{\mu}\right] + \textstyle\sum_{i=0}^{N}\left(a\log\alpha_i - b\alpha_i\right), \tag{B.6}$$

and

$$\mathcal{L}_\beta = \frac{1}{2}\left[N\log\beta - \log|\mathbf{\Sigma^{-1}}| - \beta\|\mathbf{t} - \mathbf{\Phi}\boldsymbol{\mu}\|^2\right] + c\log\beta - d\beta \tag{B.7}$$

contain only those terms of $\mathcal{L}$ which are functions of α_i and β, respectively.

With $\mathbf{A} = \mathrm{diag}(\alpha_0, \alpha_1, \ldots, \alpha_N)$, two of the terms in (B.6) can be rewritten as explicit functions of α_i, that is, $\log|\mathbf{A}| = \log\prod_{j=0}^{M}\alpha_j = \sum_{j=0}^{M}\log\alpha_j$ and $\boldsymbol{\mu}^{\mathrm{T}}\mathbf{A}\boldsymbol{\mu} = \sum_{j=0}^{M}\mu_j^2\alpha_j$, leaving simple derivatives for all but the $\log|\mathbf{\Sigma^{-1}}|$ term. The derivative of this term is computed as follows. Make the substitutions

$$\mathbf{X} = \mathbf{A} + \beta\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi} = \mathbf{\Sigma}^{-1},$$

$$z_i = \log\alpha_i,$$

and

$$f(\mathbf{X}) = \log|\mathbf{X}|,$$

so that the derivative of the term can be shown as

$$\frac{\partial\log|\mathbf{\Sigma^{-1}}|}{\partial\log\alpha_i} = \frac{\partial f(\mathbf{X}(z_i))}{\partial z_i},$$

where the functional dependence of $\mathbf{X}$ on $z_i = \log\alpha_i$ is due only to the diagonal elements of $\mathbf{A}$ which are $\alpha_i = \mathrm{e}^{z_i}$. In general, a scalar function of a matrix $\mathbf{X}$ with respect to a scalar term z_i that appears within elements of the matrix, could be written as

$$f(\mathbf{X}(z_i)) = f(\{X_{mn}(z_i)\}),$$

where the function is now shown with explicit dependence on each element X_{mn} of the matrix $\mathbf{X}$, each of which is in turn a function of the scalar term z_i. By using the chain rule, the derivative of such a function is given by

$$\frac{\partial f(\mathbf{X}(z_i))}{\partial z_i} = \frac{\partial f(\{X_{mn}(z_i)\})}{\partial z_i} = \sum_n \sum_m \frac{\partial f}{\partial X_{mn}} \frac{\partial X_{mn}}{\partial z_i}.$$

In our case, where the scalar term z_i appears only on the diagonal of the matrix, in fact, only at the i, ith position of the matrix, the partial derivative $\frac{\partial X_{mn}}{\partial z_i}$ is nonzero only for $m = n = i$, and the desired derivative reduces to

$$\frac{\partial f(\mathbf{X}(z_i))}{\partial z_i} = \frac{\partial f}{\partial X_{ii}} \frac{\partial X_{ii}}{\partial z_i}. \tag{B.8}$$

The derivative with respect to a particular element of a matrix is that element of the derivative with respect to the matrix, that is, $\frac{\partial f}{\partial X_{ii}} = \left[\frac{\partial f}{\partial \mathbf{X}}\right]_{ii}$. So we use the derivative with respect to the matrix, which from Moon [10], is

$$\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} = \frac{\partial \log |\mathbf{X}|}{\partial \mathbf{X}} = 2\mathbf{X}^{-1} - \text{diag}\left(\mathbf{X}^{-1}\right). \tag{B.9}$$

For elements along the diagonal, which are the items of interest, this reduces to elements of the inverse matrix, that is,

$$\begin{aligned}\frac{\partial f}{\partial X_{ii}} = \left[\frac{\partial f}{\partial \mathbf{X}}\right]_{ii} &= \left[2\mathbf{X}^{-1} - \text{diag}\left(\mathbf{X}^{-1}\right)\right]_{ii} \\ &= 2[\mathbf{X}^{-1}]_{ii} - \left[\text{diag}\left(\mathbf{X}^{-1}\right)\right]_{ii} \\ &= 2\left[\mathbf{X}^{-1}\right]_{ii} - \left[\mathbf{X}^{-1}\right]_{ii} \\ &= \left[\mathbf{X}^{-1}\right]_{ii} \\ &= \Sigma_{ii}.\end{aligned}$$

We also find the derivative of a diagonal element $X_{ii} = [\mathbf{A} + \beta\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}]_{ii} = \mathrm{e}^{z_i} + [\beta\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}]_{ii}$ with respect to the scalar term z_i, which is

$$\begin{aligned}\frac{\partial X_{ii}}{\partial z_i} &= \frac{\partial}{\partial z_i}(\mathrm{e}^{z_i} + [\beta\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}]_{ii}) \\ &= \mathrm{e}^{z_i} \\ &= \alpha_i.\end{aligned}$$

If we substitute these results into (B.8) we have

$$\frac{\partial \log|\mathbf{\Sigma^{-1}}|}{\partial \log \alpha_i} = \frac{\partial f(\mathbf{X}(z_i))}{\partial z_i} = \Sigma_{ii}\alpha_i. \tag{B.10}$$

Recognizing the fact that

$$\frac{\partial}{\partial \log \alpha}\alpha = \frac{\partial}{\partial \log \alpha}\mathrm{e}^{\log \alpha} = \mathrm{e}^{\log \alpha} = \alpha,$$

then the derivatives of the other terms in (B.6) are

$$\frac{\partial}{\partial \log \alpha_i}\log|\mathbf{A}| = \frac{\partial}{\partial \log \alpha_i}\sum_{j=0}^{M}\log \alpha_j = 1, \tag{B.11}$$

$$\frac{\partial}{\partial \log \alpha_i}\boldsymbol{\mu}^{\mathrm{T}}\mathbf{A}\boldsymbol{\mu} = \frac{\partial}{\partial \log \alpha_i}\sum_{j=0}^{M}\mu_j^2\alpha_j = \mu_i^2\frac{\partial}{\partial \log \alpha_i}\alpha_i = \mu_i^2\alpha_i, \tag{B.12}$$

$$\frac{\partial}{\partial \log \alpha_i}\sum_{j=0}^{M} a \log \alpha_j = a\frac{\partial}{\partial \log \alpha_i}\log \alpha_i = a, \tag{B.13}$$

$$\frac{\partial}{\partial \log \alpha_i}\sum_{j=0}^{M} b\alpha_j = b\frac{\partial}{\partial \log \alpha_i}\alpha_i = b\alpha_i. \tag{B.14}$$

Combining the results of (B.10), (B.11), (B.12), (B.13), and (B.14) gives the final derivative result

$$\frac{\partial \mathcal{L}}{\partial \log \alpha_i} = \frac{\partial \mathcal{L}_\alpha}{\partial \log \alpha_i} = \frac{1}{2}\left[-\Sigma_{ii}\alpha_i + 1 - \mu_i^2\alpha_i\right] + a - b\alpha_i. \tag{B.15}$$

As with (B.6), the derivatives of (B.7) are simple except for the $\log|\boldsymbol{\Sigma}^{\mathbf{-1}}|$ term. The derivative of this term is computed as follows. Make the substitutions

$$\mathbf{X} = \mathbf{A} + \beta\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi} = \boldsymbol{\Sigma}^{-1},$$

$$z = \log\beta,$$

and

$$f(\mathbf{X}) = \log|\mathbf{X}|,$$

so that the derivative of the term can be shown as

$$\frac{\partial \log|\boldsymbol{\Sigma}^{\mathbf{-1}}|}{\partial \log\beta} = \frac{\partial f(\mathbf{X}(z))}{\partial z},$$

where, this time, the functional dependence of $\mathbf{X}$ on $z = \log\beta$ is due to all elements of the matrix $\mathbf{X}$ because of the appearance of $\beta = \mathrm{e}^z$ multiplying the matrix $\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}$. As a result, the derivative has the form

$$\frac{\partial f(\mathbf{X}(z))}{\partial z} = \frac{\partial f(\{X_{ij}(z)\})}{\partial z} = \sum_i\sum_j \frac{\partial f}{\partial X_{ij}}\frac{\partial X_{ij}}{\partial z}, \tag{B.16}$$

where now all terms in the summation are nonzero. We already have the derivative $\frac{\partial f}{\partial X_{ij}}$ as an element of (B.9) from before:

$$\frac{\partial f}{\partial X_{ij}} = \left[2\mathbf{X}^{-1} - \mathrm{diag}\left(\mathbf{X}^{-1}\right)\right]_{ij} = \left[2\boldsymbol{\Sigma} - \mathrm{diag}\left(\boldsymbol{\Sigma}\right)\right]_{ij}.$$

We find the derivative of an arbitrary element $X_{ij} = [\mathbf{A} + \beta\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}]_{ij} = \alpha_i\delta_{j-i} + \mathrm{e}^z\left[\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right]_{ij}$ with respect to the scalar term z, where δ_{j-i} is used to indicate the presence of the α_i term only for elements along the diagonal (where $j = i$). The derivative is

$$\begin{aligned}\frac{\partial X_{ij}}{\partial z} &= \frac{\partial}{\partial z}\left(\alpha_i \delta_{j-i} + \mathrm{e}^z \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ij}\right)\\ &= \mathrm{e}^z \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ij}\\ &= \beta \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ij}.\end{aligned}$$

If we substitute these two results into (B.16) we have

$$\begin{aligned}\frac{\partial \log |\mathbf{\Sigma^{-1}}|}{\partial \log \beta} = \frac{\partial f(\mathbf{X}(z))}{\partial z} &= \beta \sum_i \sum_j \left[2\mathbf{\Sigma} - \mathrm{diag}\left(\mathbf{\Sigma}\right)\right]_{ij} \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ij}\\ &= \beta \sum_i \sum_j \Sigma_{ij} \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ij} + \beta \sum_i \sum_j \left[\mathbf{\Sigma} - \mathrm{diag}\left(\mathbf{\Sigma}\right)\right]_{ij} \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ij}.\end{aligned} \tag{B.17}$$

In order to obtain the original RVM results [6] we must assume that the second term is zero-valued. The validity of this assumption will be discussed shortly. For now, using the assumption, the derivative result is the first term, which can be rewritten as

$$\begin{aligned}\frac{\partial \log |\mathbf{\Sigma^{-1}}|}{\partial \log \beta} &= \beta \sum_i \sum_j \Sigma_{ij} \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ij}\\ &= \beta \sum_i \sum_j \Sigma_{ij} \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ji}\\ &= \beta \sum_i \mathbf{\Sigma}_{i,:} \left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)_{:,i}\\ &= \beta \sum_i \left[\mathbf{\Sigma}\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ii}\\ &= \beta \mathrm{tr}\left(\mathbf{\Sigma}\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right).\end{aligned} \tag{B.18}$$

The derivatives of the other terms in (B.7) are

$$\frac{\partial}{\partial \log \beta} N \log \beta = N, \tag{B.19}$$

$$\frac{\partial}{\partial \log \beta} \beta \|\mathbf{t} - \mathbf{\Phi}\boldsymbol{\mu}\|^2 = \beta \|\mathbf{t} - \mathbf{\Phi}\boldsymbol{\mu}\|^2, \tag{B.20}$$

$$\frac{\partial}{\partial \log \beta} c \log \beta = c, \tag{B.21}$$

$$\frac{\partial}{\partial \log \alpha_i} d\beta = d\beta. \tag{B.22}$$

Combining the results of (B.18), (B.19), (B.20), (B.21), and (B.22) gives the final derivative result

$$\frac{\partial \mathcal{L}}{\partial \log \beta} = \frac{\partial \mathcal{L}_\beta}{\partial \log \beta} = \frac{1}{2}\left[N - \beta \mathrm{tr}\left(\mathbf{\Sigma}\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right) - \beta \left\|\mathbf{t} - \mathbf{\Phi}\boldsymbol{\mu}\right\|^2\right] + c - d\beta. \tag{B.23}$$

Equating the derivative results in (B.15) and (B.23) to zero and solving for the hyperparameters α_i and $\sigma^2 = \beta^{-1}$, gives the update equations

$$\alpha_i^{\mathrm{new}} = \frac{1 + 2a}{\mu_i^2 + \Sigma_{ii} + 2b}, \tag{B.24}$$

and

$$(\sigma^2)^{\mathrm{new}} = \frac{\left\|\mathbf{t} - \mathbf{\Phi}\boldsymbol{\mu}\right\|^2 + \mathrm{tr}\left(\mathbf{\Sigma}\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right) + 2d}{N + 2c}. \tag{B.25}$$

Considering the case with uniform hyperpriors where $a = b = c = d = 0$ the update equations become

$$\alpha_i^{\mathrm{new}} = \frac{1}{\mu_i^2 + \Sigma_{ii}}, \tag{B.26}$$

and

$$(\sigma^2)^{\mathrm{new}} = \frac{\left\|\mathbf{t} - \mathbf{\Phi}\boldsymbol{\mu}\right\|^2 + \mathrm{tr}\left(\mathbf{\Sigma}\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)}{N}. \tag{B.27}$$

These update equations can be shown to be equivalent to updates found through use of the EM algorithm [6].

Making the substitution $\gamma_i \equiv 1 - \alpha_i \Sigma_{ii}$ in the derivative results—as suggested by Mackay [8]—gives

$$\frac{\partial \mathcal{L}}{\partial \log \alpha_i} = \frac{1}{2}\left[\gamma_i - \mu_i^2 \alpha_i\right] + a - b\alpha_i, \tag{B.28}$$

for (B.15) and by rewriting (B.18) in terms of this substitution as

$$\begin{aligned}
\beta \mathrm{tr}\left(\mathbf{\Sigma}\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right) &= \mathrm{tr}\left(\mathbf{\Sigma}\beta\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right) \\
&= \mathrm{tr}\left(\mathbf{\Sigma}\left(\mathbf{\Sigma}^{-1} - \mathbf{A}\right)\right) \\
&= \mathrm{tr}\left(\mathbf{I} - \mathbf{\Sigma}\mathbf{A}\right) \\
&= \mathrm{tr}\left(\mathbf{I}\right) - \mathrm{tr}\left(\mathbf{\Sigma}\mathbf{A}\right) \\
&= \sum_i 1 - \sum_i \alpha_i \Sigma_{ii} \\
&= \sum_i \left(1 - \alpha_i \Sigma_{ii}\right) \\
&= \sum_i \gamma_i,
\end{aligned}$$

gives

$$\frac{\partial \mathcal{L}}{\partial \log \beta} = \frac{1}{2}\left[N - \sum_i \gamma_i - \beta \left\|\mathbf{t} - \mathbf{\Phi}\boldsymbol{\mu}\right\|^2\right] + c - d\beta \tag{B.29}$$

for (B.23). With this modification, (B.28) and (B.29) lead to the update equations

$$\alpha_i^{\mathrm{new}} = \frac{\gamma_i + a}{\mu_i^2 + b}, \tag{B.30}$$

and

$$(\sigma^2)^{\mathrm{new}} = \frac{\left\|\mathbf{t} - \mathbf{\Phi}\boldsymbol{\mu}\right\|^2 + d}{N - \sum_i \gamma_i + c}. \tag{B.31}$$

Again, considering the case with uniform hyperpriors, the update equations become

$$\alpha_i^{\mathrm{new}} = \frac{\gamma_i}{\mu_i^2}, \tag{B.32}$$

and

$$(\sigma^2)^{\mathrm{new}} = \frac{\left\|\mathbf{t} - \mathbf{\Phi}\boldsymbol{\mu}\right\|^2}{N - \sum_i \gamma_i}. \tag{B.33}$$

To address the assumption of a value of zero for the second term in (B.17), we can write this term as

$$\begin{aligned}
\beta \sum_i \sum_j \left[\mathbf{\Sigma} - \operatorname{diag}(\mathbf{\Sigma})\right]_{ij} \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ij} &= \beta \sum_i \sum_j \Sigma_{ij} \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ij} - \beta \sum_i \sum_j \left[\operatorname{diag}(\mathbf{\Sigma})\right]_{ij} \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ij} \\
&= \beta \operatorname{tr}\left(\mathbf{\Sigma}\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right) - \beta \sum_i \Sigma_{ii} \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ii} \\
&= \sum_i \gamma_i - \beta \sum_i \Sigma_{ii} \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ii} \qquad \text{(B.34)}
\end{aligned}$$

to see that the term is zero-valued if and only if $\beta \sum_i \Sigma_{ii} \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ii} = \sum_i \gamma_i$. Using $\mathbf{\Sigma}^{-1} = \mathbf{A} + \beta\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}$, we can write $\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi} = \beta^{-1}\left(\mathbf{\Sigma}^{-1} - \mathbf{A}\right)$, and thus

$$\left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ii} = \beta^{-1}\left(\left[\mathbf{\Sigma}^{-1}\right]_{ii} - \alpha_i\right).$$

Substituting this within the second term of (B.34) gives

$$\begin{aligned}
\beta \sum_i \Sigma_{ii} \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ii} &= \sum_i \Sigma_{ii} \left(\left[\mathbf{\Sigma}^{-1}\right]_{ii} - \alpha_i\right) \\
&= \sum_i \left[\Sigma_{ii} \left[\mathbf{\Sigma}^{-1}\right]_{ii} - \alpha_i \Sigma_{ii}\right] \qquad \text{(B.35)} \\
&= \sum_i \Sigma_{ii} \left[\mathbf{\Sigma}^{-1}\right]_{ii} - \sum_i \alpha_i \Sigma_{ii}, \qquad \text{(B.36)}
\end{aligned}$$

where (B.35) shows that

$$\begin{aligned}
\beta \sum_i \Sigma_{ii} \left[\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right]_{ii} &= \sum_i \left[1 - \alpha_i \Sigma_{ii}\right] \\
&= \sum_i \gamma_i
\end{aligned}$$

for

$$\frac{1}{\Sigma_{ii}} = \left[\mathbf{\Sigma}^{-1}\right]_{ii},$$

or less stringently (B.36) shows that

$$\begin{aligned}\beta \sum_i \Sigma_{ii} \left[\mathbf{\Phi}^{\mathrm{T}} \mathbf{\Phi}\right]_{ii} &= (N+1) - \sum_i \alpha_i \Sigma_{ii} \\ &= \sum_i \left[1 - \alpha_i \Sigma_{ii}\right] \\ &= \sum_i \gamma_i\end{aligned}$$

for

$$\sum_i \Sigma_{ii} \left[\mathbf{\Sigma}^{-1}\right]_{ii} = N + 1,$$

neither of which is true in general. The assumption, therefore, is not exact. The degree of validity for the assumption as an approximation is unknown.

blank page