Using Historical Data for Retrospective Prediction of Rainfall In the Midwest

REU Site: Interdisciplinary Program in High Performance Computing

Ephraim Alfa[1], Huiyi Chen[2], Kristen Hansen[3], Mathew Prindle[4]
Graduate assistants: Sai Popuri and Nadeesri Wijekoon[1]
Faculty mentor: Kofi Adragni[1] and Client: Amita Mehta[5]

[1]Department of Mathematics and Statistics,UMBC,
[2]Department of Mathematics, University of California at Merced
[3]Department of Psychology and Department of Statistics, University of Minnesota, Morris
[4]Department of Mathematics, California State University, Sacramento
[5]JCET and Geography and Environmental Science, UMBC

Technical Report HPCF–2016–11, hpcf.umbc.edu > Publications

Abstract

The Missouri River Basin (MRB) is an important food-producing region in the United States and Canada. Climate variability and water availability affect crops production in this region. Past climate data have been recorded at various locations in the basin over a period of ten years. We use the data for a retrospective prediction of rainfall. As the dimension of the data is relatively large, a sufficient dimension reduction approach is used to reduce the dimensionality of the data while preserving the regression information pertinent to rainfall. We use the nascent dimension reduction methodology called Minimum Average Deviance Estimation or MADE to reduce the dimensionality of the climate data. Since MADE is still a tool in development, we explored two of its intrinsic prediction methods and compared them to the Nadaraya-Watson prediction approach by a cross-validation. A parallel implementation of MADE and its prediction methods on a high performance computer were carried out. A performance study was performed along with the application of the best prediction method to the MRB climate data.

1 Introduction

Fresh water is naturally available in lakes, rivers, and through rainfall. It sustains life on earth. Rainfalls fill aquifers and keep rivers and lakes filled. Generally, in agriculture, water availability is a necessary condition for an optimal yield of crops. The Missouri River Basin is incredibly important for food production in the United States. It is one of the largest river basins in the United States, covering more than 500,000 square miles also covering parts 10 U.S. states and 2 Canadian Provinces. This region produces about 46% of U.S. wheat and 22% of U.S. grain corn. Approximately 90% of the basins cropland is not irrigated and is entirely dependent on precipitation. According to the drought monitor for June 2016 from the National Drought Mitigation Center, there is an increase in intensity and areal stage of drought conditions in the northern Missouri river basin compared to May. June precipitation

in the Missouri river basin was about 50% of the average. This increased drought prevalence in the area provides a need for better precipitation prediction in order to manage cropland and agricultural yield.

Climate is the average weather conditions – such as atmospheric pressure, temperature, humidity, wind speed, and precipitation – in a certain area over a period of many years. n particular, we have the data for MRB collected at different locations over a period of ten years. Our interest is in the rainfall. An ideal endeavor would be to model the rainfall across space and time. But restrictions due to time and availability of tools limit our work to a single location. Our goal was to develop a predictive model for the rainfall in at a given location using historical time series data.

As the dimensionality of the data is large, we opted at using a methodology for dimension reduction to help reduce the dimensionality of the covariates, and then use the reduced data to predict the rainfall. Dimension reduction methods abound in the literature. They are generally for some specific response variable. In our case, the response, rainfall measurements, were non-negative. To deal with this specific type of response, we consider a nascent dimension reduction methodology called Minimum Average Deviance Estimation, or MADE, in order to reduce the dimension of the data. Once the reduction is obtained, it is used to predict the response.

The rest of this paper is organized in the following way: Section 2 gives a detailed description of the statistical methodology of Minimum Average Deviance Estimation, provides the estimation method of its parameters, and also describes three prediction methods to be compared via a simulation. Section 3 describes a series of simulation studies performed in order to determine the effectiveness of MADE. Section 4 presents the results of the application of MADE to the climate model data set provided by the client, and section 5 includes discussion of the results and conclusions of our project.

2 Statistical Methodologies

Sufficient Dimension Reduction is achieved when the dimensionality of a data set is reduced while still retaining most regression information. There are numerous ways to reduce data dimensionality present in statistical literature. When considering the classical regression problem of $Y|X$, a sufficient dimension reduction is a reduction $R(X)$ of the data-matrix X, that still holds all the regression information about Y that was contained in X.

2.1 Minimum Average Deviance Estimation Model

The developing tool of Minimum Average Deviance Estimation was used in order to reduce dimensionality of data. Because this is a developing method there is much to be learned about its functionality with different types of data. This method is to obtain a reduction $R(X)$ so that $Y|X$ has the same distribution as $Y|R(X)$ where $Y|X$ follows an exponential family distribution. The exponential family distributions have the form

$$f(Y \mid \vartheta(X)) = f_0(Y, \phi) \exp\left\{[Y\vartheta(X) - b(\vartheta(X))]/a(\phi)\right\}. \tag{2.1}$$

For a particular dataset, a specific form of exponential family distribution would be assumed. The canonical parameter $\vartheta(X)$ is related to the mean function $E(Y|X)$ through a link function g meaning $g(E(Y|X)) = \vartheta(X)$. The specific distribution of $Y|X$ determines which link function is used. Let $(Y_i, X_i), i = 1, \ldots, n$ represent independent samples from the distribution of (Y, X) so that $Y_i|X_i$ has the distribution [**?**]. Assuming that $\vartheta(X)$ is continuous and smooth so that at any point X the first order linear expansion is as follows,

$$\vartheta(X_i) \approx \vartheta(X) + [\nabla\vartheta(X)]^T(X_i - X) \tag{2.2}$$

for any X_i in the neighborhood of X. Let $\alpha = \vartheta(X)$ and $\Gamma = \nabla\vartheta(X)$. As X varies in its sample space, Γ describes a u-dimensional subspace $\mathcal{S}$ in $\mathbb{R}^p$ with $u \leq \min(n, p)$. Let U be an orthonormal basis of $\mathcal{S}$ so that $\Gamma = U\delta$ for some $\delta = \delta(X) \in \mathbb{R}^{u\times 1}$. It follows that $\vartheta(X_i) = \vartheta(U^T X_i)$. Consequently, $U^T X$ is a sufficient reduction of X as the distribution of $Y \mid X$ is approximately the same as that of $Y \mid U^T X$. These reduction subspaces are not unique and could not be the minimum possible dimension of the kernel matrix B. Regression based on the local log-likelihood evaluated at a given $X \in \mathbb{R}^p$ can be written as

$$\begin{aligned} L_X(\alpha, \gamma, B) &= \sum_{i=1}^{n} w_{0i}(X) \log f(Y_i|\alpha + \gamma^T B^T(X_i - X)) \\ &= \sum_{i=1}^{n} w_{0i}(X) \left\{ \frac{Y_i(\alpha + \gamma^T B^T(X_i - X)) - b(\alpha + \gamma^T B^T(X_i - X))}{a_i(\phi)} + \log f_0(Y_i, \phi) \right\}. \end{aligned} \tag{2.3}$$

The weights $w_{01}(X), \ldots, w_{0n}(X)$ represent the contribution of each observation toward the local likelihood $L_X(\alpha, \gamma, B)$. A local deviance can be expressed as

$$D(Y_j, \vartheta(B^T X_j)) = 2\left[\max_{\vartheta} \log f(Y_j \mid \vartheta) - L_{X_j}(\alpha_j, \gamma_j, B)\right]. \tag{2.4}$$

The term $\max_\vartheta \log f(Y_j \mid \vartheta)$ is the maximum likelihood achievable for an individual observation. Consider minimizing the average deviance $n^{-1}\sum_{j=1}^{n} D(Y_j, \vartheta(B^T X_j))$ with respect to $(\alpha_j, \gamma_j) \in \mathbb{R}^{d+1}$ for $j = 1, \ldots, n$ and $B \in \mathbb{R}^{p\times d}$ such that $B^T B = I$. This is equivalent to maximizing

$$\begin{aligned} Q(\alpha, \gamma, B) &= \sum_{j=1}^{n} L_{X_j}(\alpha_j, \gamma_j, B) \\ &= \sum_{j=1}^{n}\sum_{i=1}^{n} w_{0i}(X_j) \left\{ \frac{Y_i(\alpha_j + \gamma_j^T B^T(X_i - X_j)) - b(\alpha_j + \gamma_j^T B^T(X_i - X_j))}{a_i(\phi)} + \log f_0(Y_i, \phi) \right\}, \end{aligned} \tag{2.5}$$

which is the full local log-likelihood evaluated at each of the sample points, where $\alpha = (\alpha_1, \ldots, \alpha_n)$ and $\gamma = (\gamma_1, \ldots, \gamma_n)$. While each sample point X_j has its own regression coefficients α_j and γ_j, they all share a common dimension reduction kernel matrix B. The dimension d of the reduction kernel matrix B is to be estimated. Three estimation methods were devised to estimate d: a sequential permutation test, a bootstrap method, and cross-validation.

2.2 Algorithm for Estimation

The parameters of interest are $\alpha_j, \gamma_j, j = 1, \ldots, n$, and $B \in \mathbb{R}^{p\times d}$. We start by assuming that the dimension d is known. For any orthogonal matrix O, $\gamma^T B^T = \gamma^T O O^T B^T$, which implies that γ and B are not uniquely determined but obtained up to an orthogonal transformation. Furthermore, refined weights based on the Gaussian kernel $w_i(B^T X)$ with $H = hI$ depend on B only through $BB^T = BOO^T B^T$. In this setting, the MADE problem is invariant to orthogonal transformation of B in the sense that

$$Q(\alpha, \gamma, B) = Q(\alpha, O^T\gamma_1, \ldots, O^T\gamma_n, BO). \tag{2.6}$$

The parameter space of B is the set of d-dimensional subspaces in $\mathbb{R}^p$ known as the Grassmann manifold of dimension $d(p-d)$. However, the estimation method we adopt does not estimate all the parameters jointly, but works iteratively. For fixed values of α_j and γ_j, $j = 1, \ldots, n$, the parameter space of B is the set of d-dimensional orthonormal matrices in $\mathbb{R}^p$, also known as Stiefel manifold of dimension $pd - d(d+1)/2$. The dimension of Steifel and Grassmann manifolds is discussed in [1]. In the following, we present an iterative method to maximize (2.5) for a given dimension d, and later discuss selection of d.

To estimate the parameters $(\alpha_j, \gamma_j) \in \mathbb{R}^{d+1}, j = 1, \ldots, n$, we start by fixing B in (2.5). We see that maximizing Q over (α_j, γ_j) is equivalent to maximizing each $L_{X_j}(\alpha_j, \gamma_j; B)$ separately. There is no closed-form solution of the estimator, except in certain special cases such as Gaussian outcomes. Instead, we proceed with a multivariate Newton-Raphson iterative approach. For a particular $L_X(\alpha, \gamma; B)$, let $\xi = (\alpha, \gamma^T)^T$, $Z_i = (1, (X_i - X)^T B)^T$, and $w_i = w_i(B^T X)$ so that

$$L_X(\alpha, \gamma; B) = \sum_{i=1}^{n} w_i [\frac{Y_i \cdot Z_i^T\xi - b(Z_i^T\xi)}{a_i(\phi)} + \log f_0(Y_i, \phi)]. \tag{2.7}$$

Let $Z = (Z_1, \ldots, Z_n)^T$, $W = diag(w_1, \ldots, w_n)$ and $H(\xi) : \mathbb{R}^{d+1} \to \mathbb{R}^n$ with entries $[Y_i - b'(Z_i^T\xi)]/a_i(\phi)$ for $i = 1, \ldots, n$. The first derivative at X is then

$$\frac{\partial}{\partial\xi} L_X(\alpha, \gamma; B) = \sum_{i=1}^{n} w_i \frac{Y_i - b'(Z_i^T\xi)}{a_i(\phi)} Z_i = Z^T W H(\xi). \tag{2.8}$$

The function $H(\xi)$ has an $n \times (d+1)$ Jacobian

$$J_H(\xi) = \left(\frac{\partial}{\partial\xi_j} \frac{Y_i - b'(Z_i^T\xi)}{a_i(\phi)}\right) = -\frac{1}{a_i(\phi)} \begin{pmatrix} b''(z_1^T\xi)Z_{1,1} & \cdots & b''(z_1^T\xi)Z_{1,d+1} \\ \vdots & \ddots & \vdots \\ b''(z_n^T\xi)Z_{n,1} & \cdots & b''(z_n^T\xi)Z_{n,d+1} \end{pmatrix}.$$

To formulate Newton-Raphson iterations, suppose $\xi^{(g)}$ is a given iterate and $\xi^{(g)} + \Delta\xi$ will be the next iterate. To solve for $\Delta\xi$ approximately, set the first order Taylor expansion of (2.8),

$$Z^T W H(\xi^{(g)} + \Delta\xi) \approx Z^T W H(\xi^{(g)}) + Z^T W J_H(\xi^{(g)})\Delta\xi,$$

to zero and solve to obtain $\Delta\xi = -\{Z^T W J_H(\xi^{(g)})\}^{-1} Z^T W H(\xi^{(g)})$. This suggests the update of ξ as $\xi^{(g+1)} = \xi^{(g)} - \{Z^T W J_H(\xi^{(g)})\}^{-1} Z^T W H(\xi^{(g)})$. These steps are iterated until the gth iteration where $\|\xi^{(g)} - \xi^{(g-1)}\| < \varepsilon$ for some small prescribed $\varepsilon > 0$.

To estimate B, we suppose that (α_j, γ_j), $j = 1, \ldots, n$, are fixed and known. Omitting the terms of the objective function (2.5) that are free of B, estimation of B is carried out by maximizing

$$Q(B) = \sum_{j=1}^{n}\sum_{i=1}^{n} w_i(B^T X_j)\frac{1}{a_i(\phi)}\left\{Y_i(\alpha_j + \gamma_j^T B^T(X_i - X_j)) - b(\alpha_j + \gamma_j^T B^T(X_i - X_j))\right\}, \tag{2.9}$$

over the set of d dimensional semi-orthogonal matrices in $\mathbb{R}^p$. It is noteworthy that in the setting of MAVE, [4], and [5] used a quadratic programming method, where a vectorized form of B is estimated in an iterative fashion, and orthonormalized afterwards. In the present work, B is estimated in its natural parameter space, a Stiefel manifold, which naturally honors the orthonormality constraint.

$$\frac{\partial Q(B)}{\partial B_{rs}} = \sum_{j=1}^{n}\sum_{i=1}^{n} w_{ij}\Big\{\frac{Y_i - \mu(\alpha_j + \gamma_j^T B^T(X_i - X_j))}{a_i(\phi)}\Big\}\gamma_{js}(X_{ir} - X_{jr}), \tag{2.10}$$

for $r \in \{1, \ldots p\}$ and $s \in \{1, \ldots, d\}$. Here, $\mu(\vartheta) = b'(\vartheta)$ represents the mean function for the exponential family and B_{rs} represents the (r, s)th element of B. The optimization on the Stiefel manifold converges when $TrH^T H - (1/2)A^T A < \epsilon$ for a user-specified $\epsilon > 0$, where $H = \nabla Q(\hat{B})$ and $A = \hat{B}^T \nabla Q(\hat{B})$ with $\nabla Q(B) = \partial Q/\partial B - B(\partial Q/\partial B)^T B$.

2.3 Prediction Methods

Suppose we wish to estimate $E(Y|X)$ for a new observation $X = X^*$. Let $\hat{B}$ denote the estimate of B based on n independent observations. We provide three different prediction methods that do not rely on the exact specification of the regression function to predict the response corresponding to a new observation X^*. Let $\{w_{i*}\}_{i=1}^n$ denote the set of kernel weights obtained as

$$w_{i*} = K_H(\hat{B}^T(X_i - X_*))/\textstyle\sum_{m=1}^n K_H(\hat{B}^T(X_m - X_*))$$

The first prediction method yields the predicted response as

$$\hat{E}(Y \mid X^*) = \sum_{i=1}^{n} w_{i*} Y_i = \frac{\sum_{i=1}^n K_H(\hat{B}^T(X_i - X_*))Y_i}{\sum_{j=1}^n K_H(\hat{B}^T(X_j - X_*))}. \tag{2.11}$$

This prediction method is a Nadaraya-Watson estimator which is typical for nonparametric methods, and was used in [4] in the context of cross-validation. The Nadaraya-Watson estimator can be used with any sufficient dimension reduction method that could provide an estimate for B. We will refer to this prediction method as the NW method.

The second prediction method is related to a local likelihood regression. We continue to make the assumption that the n independent samples are used to obtain $\hat{B}$, an estimator of the reduction matrix B. Now, for the new observation X^*, $\hat{\alpha}$ and $\hat{\gamma}$ may be are obtained as

$$(\hat{\alpha}, \hat{\gamma}) = \underset{\alpha,\gamma}{\operatorname{argmax}} \sum_{i=1}^{n} w_i(\hat{B}^T X^*) \Big\{ \frac{Y_i(\alpha + \gamma^T \hat{B}^T(X_i - X^*)) - b(\alpha + \gamma^T \hat{B}^T(X_i - X^*))}{a_i(\phi)} \Big\}, \tag{2.12}$$

In the original Taylor expansion, $\alpha = \vartheta(B^T X^*)$. As is often done in local likelihood literature [2, 3] we can predict Y^* using only the intercept as $g^{-1}(\hat{\alpha})$. We will denote this as local likelihood prediction I, or $\hat{E}(Y|X^*)_{LL_I}$.

We also consider local likelihood prediction II, computed as

$$\hat{E}(Y|X^*)_{LL_{II}} = g^{-1}\left(\sum_{i=1}^{n} (\hat{\alpha} + \hat{\gamma}^T \hat{B}^T(X_i - X^*)) w_{i*} \right), \tag{2.13}$$

which incorporates the estimate for the slope as well. Both LL1 and LL2 are associated to MADE.

We used a simulation study to compare these three prediction methods. We used the mean squared prediction error to calculate the prediction errors for each of the three prediction methods. The mean squared prediction error is calculated by

$$MSPE = n_e^{-1} \sum_{i=1}^{n_e} (Y_i - \hat{Y}_i)^2$$

. Relatively, a method with a smaller mean squared prediction error is preferred.

3 Simulation Study

The native R code provided by our mentor and co-author of MADE was not written for a parallel routine on hign performance computer, but in serial for a single processor. We rewrote the code and took advantage of some R packages designed to facilitate R codes in parallel on computers with distributed processors.

We wanted to evaluate how the parallel version of the R code for MADE would run with high dimensional data on the clusters. In order to run a simulation and a performance study, we generated the data under a typical linear regression setting of the form $Y = X\beta + \epsilon$. The data-matrix X was generated using the standard normal distribution; β was a column-vector generated from an uniform distribution with minimum of -2 and maximum of 2. The outcome Y was obtained by adding the noise ϵ simulated from the normal distribution of $\sigma = .5$. A constraint was added to the Y data-vector so that it can't be less than 0. This was due to the fact that the intended data that we will be using in application is rainfall data which cannot have a value smaller than 0 for the Y vector.

3.1 Performance Study

The initial code of MADE was written by the authors for serial processes and is composed with three main parts: estimation of γ and α with a Newton-Raphson iteration, the estimation of B using a Stiefel manifold optimization code, and the prediction. Each part requires complex operations. For our project, we analyzed to identify its main intensive portions. Consequently, we devised a parallelization procedure of these main portions. The parallelization was implemented using snowslurm and snow packages in R. We created a slurm cluster and then ran using the parallelization of functions that were converted to be in an apply setting.

We ran a performance study after parallelization. The goal is to determine whether the procedure would speed up the runtime of the time when several processors are used. For this study, the sample size of the data was set at $n = 500$ with $p = 40$ covariates. The data was replicated 100 times.

# Processors	1	2	4	8	16	32	48	64
Time (*hours*)	16.27	8.27	8.91	8.23	8.27	8.92	8.22	8.71
Speedup	1	1.97	1.82	1.98	1.97	1.82	1.98	1.87

Table 3.1: Performance Study Times

The results of the performance study shown in 3.1 show there is a significant decrease in time to run the code on this simulation from 1 to 2 processors. Increasing farther than 2 processes however does not show any improvement. This suggests that running the code in parallel does produce an improvement in running time but only in a split from 1 to 2 processes, at least for the present R code setting. The plot 3.1 gives the same information with a closer look at the improvement between 1 and 2 processes to show the significant improvement in run time.

3.2 Prediction Performance

We aimed at comparing the three prediction methods associated to MADE. These methods can't be compared analytically so recourse was to a simulation. Using the same setting for data simulation as in the previous section, we study the performance of these prediction methods described in section 2.3 when the sample size was increased.

Several settings were considered with three different sample sizes $n = 50, 250$, and 500, and four different number of predictors $p = 5, 10, 20$, and 40. We ran 100 replications of each level and evaluated the performance. The results are summarized in Table 3.2. There is no clear trends in the change of prediction error when n or p is changing. Perhaps more investigation should be carried to determine whether these results should be confirmed.

The results on Figure 3.2 comparing the three prediction methods suggest that LL_I performs better than the other two in the setting of these simulation. It might be necessary to consider other possible settings. For example, when the predictors are from skewed or

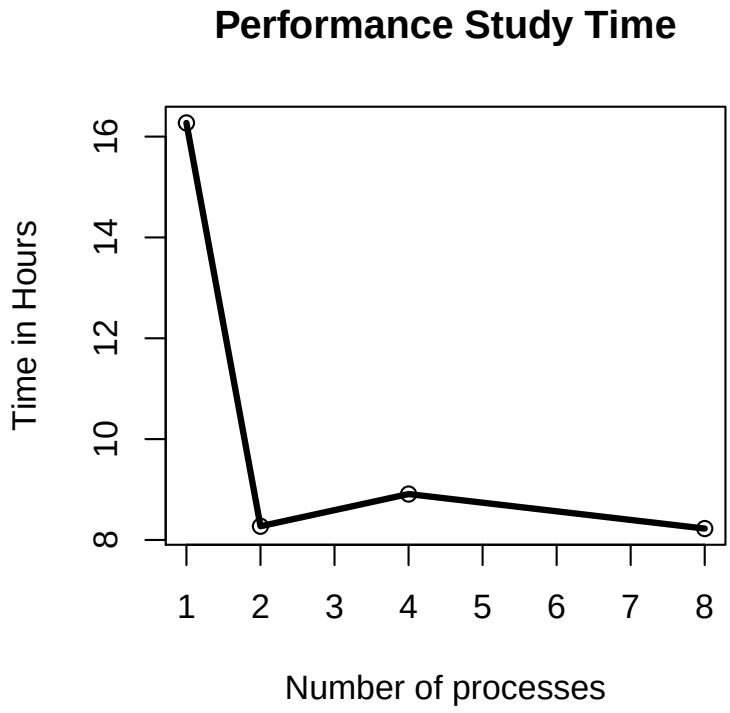

Figure 3.1: Plot of times from 1 to 8 processes

discrete distributions. There is no firm conclusion at this point, but we believe ulterior study needs to be carried out for possibly conclusive results.

	$n = 50$	$n = 250$	$n = 500$
$p = 5$	0.050	0.036	0.040
$p = 10$	0.078	0.036	0.011
$p = 20$	0.097	0.139	0.059
$p = 40$	0.597	0.065	0.0205

Table 3.2: Median Prediction Error for LL_I

4 Application to Missouri River Basin Data

To explore the possible usefulness of MADE with time series data we applied the methodology to the Missouri River Basin (MRB) climate data. The MRB data was collected over the period from 1949–2005. Access to this data was provided by JCET. There were observed values and climate model produced values from the MIROC5 climate model. The data set

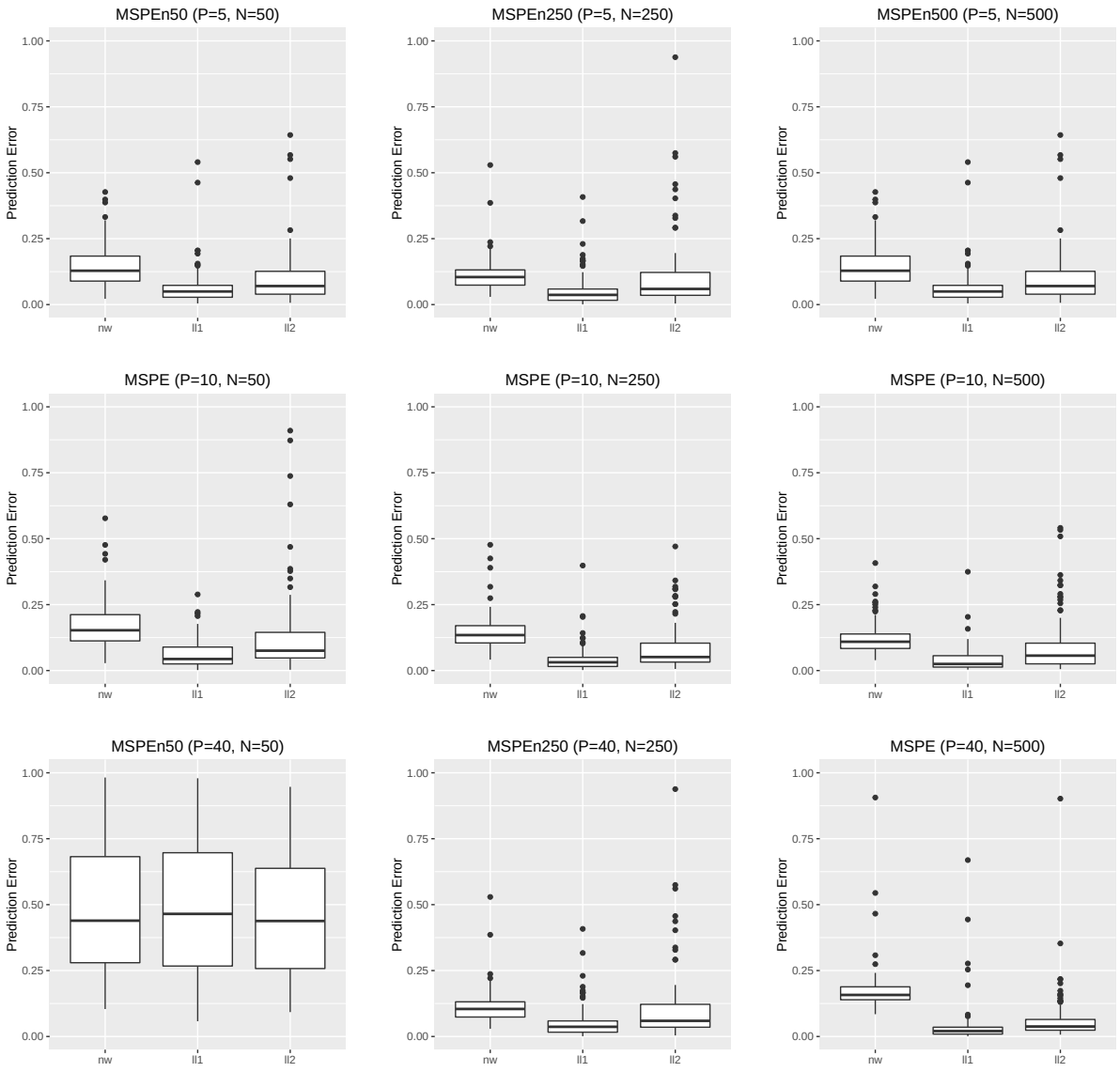

Figure 3.2: Prediction error using NW, LL1, and LL2.

was collected at over 18,000 locations. This dataset consists of variables such as latitude, longitude, date recorded, precipitation measurements, and various other climate variables. Precipitation was the variable of interest. The dataset can be split into either daily or monthly sets, we operated exclusively with monthly data.

We took three climate variables from the climate model data, which consisted of max temperature, sea level pressure, and relative humidity. As well as subsetting those three variables we created two lag variables for precipitation. The years 1950 to 1960 were used to run this analysis. This created an X matrix of 132×5 with desired dimension reduction ($B^T X$) of only one dimension, $d = 1$. The Y vector was created using the precipitation values in the same location. We ran a leave-one-out cross validation on these data at a

single latitude longitude point in the Missouri River Basin. We used the MADE specific estimator (LL1) to estimate the effectiveness of the MADE reduction on these data. The cross validation took out a single month's observations at a certain time and used the rest of the data to predict that month's precipitation value. This produced a mean squared prediction error value of 0.114. This shows that MADE was able to successfully reduce the dimension of X from 5 to 1, and use the reduction for subsequent prediction.

Further analyses using MADE on a larger subset of the original covariates from the data and larger area of the region would be suggested in order to test how well MADE performs.

5 Conclusions

The results of the simulation study suggest that MADE estimators are better than the Nadaraya-Watson estimator because the prediction errors of MADE estimators were better than the Nadaraya-Watson estimator. This was clearly evident when the number of observations increased because the Nadaraya-Watson estimator had a consistent mean squared prediction error whereas the MADE specific estimators improved. We tested how quickly the MADE code could be implemented with large sets of data, so we conducted a performance study. This performance study suggested that MADE did in fact require parallelization by comparing the time it took for different numbers of processes. In serial (1 processor) the code ran for over 16 hours. Then, when increasing the number of processes to 2, the performance of MADE improved significantly as it took 8 hours to perform. However, when continuing to increase the number of processes on MADE, there was no clear relationship between the number of processes and the amount of time it took to perform. Then, we used climate data from 1950 - 1960 for one location in the MRB and applied MADE to that data. The results showed that MADE successfully reduced the dimensions of the data and still kept the regression information. Since MADE is a relatively new technique, there needs to be further exploration of the use of MADE with time series data in order to determine what kinds of data could benefit from the sufficient dimension reduction of MADE. The code of MADE could also improve so that it can be implemented more quickly on the large data sets for which it was created.

Acknowledgments

These results were obtained as part of the REU Site: Interdisciplinary Program in High Performance Computing (hpcreu.umbc.edu) in the Department of Mathematics and Statistics at the University of Maryland, Baltimore County (UMBC) in Summer 2016. This program is funded by the National Science Foundation (NSF), the National Security Agency (NSA), and the Department of Defense (DoD), with additional support from UMBC, the Department of Mathematics and Statistics, the Center for Interdisciplinary Research and Consulting (CIRC), and the UMBC High Performance Computing Facility (HPCF). HPCF is supported by the U.S. National Science Foundation through the MRI program (grant nos. CNS–0821258 and CNS–1228778) and the SCREMS program (grant no. DMS–0821311), with additional

substantial support from UMBC. Co-author Ephraim Alfa was supported, in part, by the UMBC National Security Agency (NSA) Scholars Program through a contract with the NSA. Graduate assistants Sai Popuri and Nadeesri Wijekoon were supported by UMBC.

References

[1] A Edelman, T Arias, and S Smith. The geometry of algorithms with orthogonality constraints. *SIAM Journal of Matrix Analysis and Applications*, 20(2):303–353, 1998.

[2] Jianqing Fan and Irene Gijbels. Variable bandwidth and local linear regression smoothers. *The Annals of Statistics*, 20:2008–2036, 1992.

[3] Clive Loader. *Local Regression and Likelihood*. Statistics and Computing. Springer New York, 1999.

[4] Yingcun Xia, Howell Tong, W. K. Li, and Li-Xing Zhu. An adaptive estimation of dimension reduction space. *Journal of the Royal Statistical Society Series B*, 64(3):363–410, 2002.

[5] Xiangrong Yin and Bing Li. Sufficient dimension reduction based on an ensemble of minimum average variance estimators. *Ann. Statist.*, 39(6):3392–3416, 12 2011.