Colorado School of Mines

Practicum 2016

Precipitation Type Forecasting Using Statistical Clssification Methods

Ed Prentice

November 13, 2017

Abstract

The purpose of this project is to classify precipitation types as either rain, snow, freezing rain, or ice pellets based on their vertical temperature profiles. The initial model was quadratic discriminant analysis (QDA) to classify the precipitation types. There were some problems with this method because the temperatures at each vertical location are highly correlated. To avoid this problem, regularized discriminant analysis (RDA) was proposed. With this approach, 93.61% of the observations were correctly classified. For this project, we wanted to explore other possible methods that would improve the overall classifications probabilities. More specifically, we first used spatial methods to account for the high correlation. We removed the trend of a spatially varying mean, and then reparameterized the covariance matrices (and eventually regularized the covariance matrices). This new method correctly classified 94.1% of the observations. However, this method is about 3 times more computationally expensive than RDA.

Also, another problem is that the data is not multivariate normal. Therefore, we explored the possibility of nonparametric density estimation. However, the dimensions in this problem are not independent. We used principal component analysis to turn the correlated data into uncorrelated data and also to reduce the dimensions in the problem. With these principal components, we used a kernel density estimate, using both the default bandwidth estimation in the `KS` package in `R` and Scott's plug-in bandwidth. We found that both of these bandwidths had a better percentage of correct classifications than RDA. Using Scott's bandwidth we had a classification of 94.8%. Overall this was a better method in classifying the four precipitation types.

Introduction

The main goal of this project is to correctly classify what kind of precipitation will fall based on the vertical temperature profiles. The main focus is on four precipitation types: rain, snow, freezing rain, and ice pellets. The ability to predict these different precipitation types is critical for a community's ability to prepare for damaging weather.

There are certain challenges that arise when forecasting the precipitation type. First, the vertical temperature profiles are discrete and then they are interpolated so they become continuous. As we will show later, the vertical temperatures are highly correlated, so there is very little uncertainty within the interpolation of the vertical temperatures. Also, surface level temperatures are only available where a weather station is located. This would mean that a high level of interpolation is needed to forecast a given precipitation type at locations where there are no measurements, raising the uncertainty of the forecast.

In order to determine if a probabilistic forecast is good, it needs to be reliable and as specific as possible (predicting a probability close to 0 or 1). An example of a reliable forecast is, if we predict a probability of 80% chance of rain, we would expect rain to fall 80% of all days with that prediction probability. There are different quantitative measures of the quality of the forecast. For this project, we will evaluate our performance using the brier skill score (BSS) (see Equation 2) and the percentage of correct classifications. This score measures the overall quality of our forecast compared to a climatological forecast. A climatological forecast only looks at the history of the climate in a certain time period. We need a brier score (BS) for both the climatological forcast as well as the testing observation in order to calculate the BSS. The BS for both the testing observation and climatological observation is defined as follows:

$$\text{BS} = \sum_{i=1}^{n} (p_i - o_i)^2 \tag{1}$$

Where p_i is the event probability forecast, and o_i is the observed precipitation type ($o_i = 1$ if the event occurred, and $o_i = 0$ if not). The BSS is defined as:

$$\text{BSS} = 1 - \frac{BS}{BS_{ref}} \tag{2}$$

Where the BS_{ref} is the BS of the climatological forecast. Therefore, we want our BS to be significantly lower than our BS_{ref}, meaning that a BSS closer to 1 is better.

Description of Data

The the date range for our data set is November 1996 through December 2013 with a focus only on the cold seasons (September through May). The data was collected from 551 different station locations (506 in the Continental US, 26 in Alaska, and 19 in Canada). Each of these stations report the observed precipitation type, as well as the vertical temperature profiles associated with the precipitation. The vertical profiles range from ground level to 3000 meters above ground level (AGL) in 100 meter increments. In all, there are 230,321 observations (155,977 observations of rain, 70,482 observations of snow, 3110 observations of freezing rain, and 752 observations of ice pellets). Therefore, the vertical temperature matrix is 230,321 by 31 (observations by height AGL).

We then divide the data into rolling training and testing periods. Where the training periods are 5 years, and the testing period is the following year. For example, September 1996 through May 2001 is the first training period and September 2001 through May 2002 is the first testing period, and so on. Therefore, there are 12 training and testing periods.

Exploratory Analysis

First, we plotted the mean vertical temperatures for each of the precipitation types for all the training periods (see Figure 1).

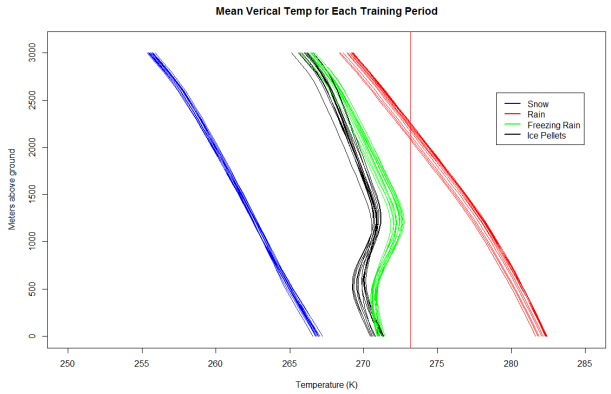

Figure 1: Mean vertical temperature profiles for all 12 training periods

We see that for snow and rain, the vertical temperatures follow a fairly linear pattern, where as freezing rain and ice pellets contain temperature inversions. This gives us an indication that the temperatures at each vertical location are strongly correlated. To verify this, we look at the correlation matrices for each precipitation type, where we can see that there is a strong correlation between all the vertical temperatures (see Figure 2). Also, we can justify reducing the dimensions of the vertical temperatures down from 31 to 16, because the interesting features of the temperature profiles appear between 0 and 1500 meters AGL.

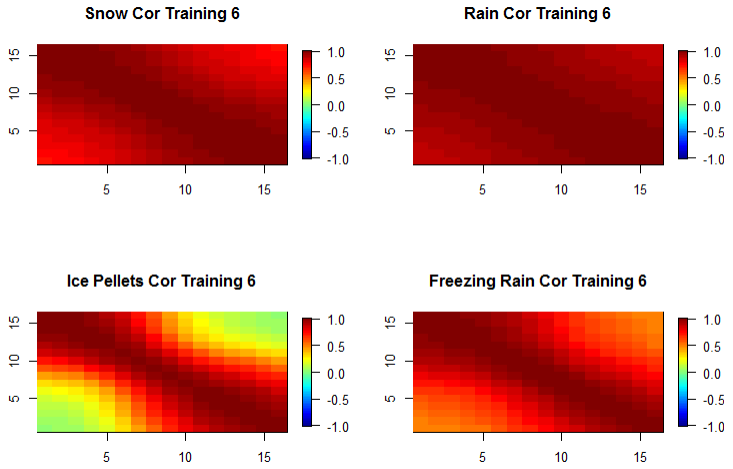

Figure 2: Correlation matrices for all vertical profiles for September 2001- May 2006

In Figure 2 the correlation matrices represented on a color scale of -1 to 1 for the different precipitation types in the sixth training period (September 2001 - May 2006). We see that for snow and rain, almost all of the correlations are close to 1, where as ice pellets and freezing rain have some correlations closer to zero (and there are some that are negative for certain training periods) because of the temperature inversion.

Next, we want to verify the Gaussian assumption by looking at the histograms of the temperatures at each of the 16 vertical profiles for each precipitation type (see Figure 3).

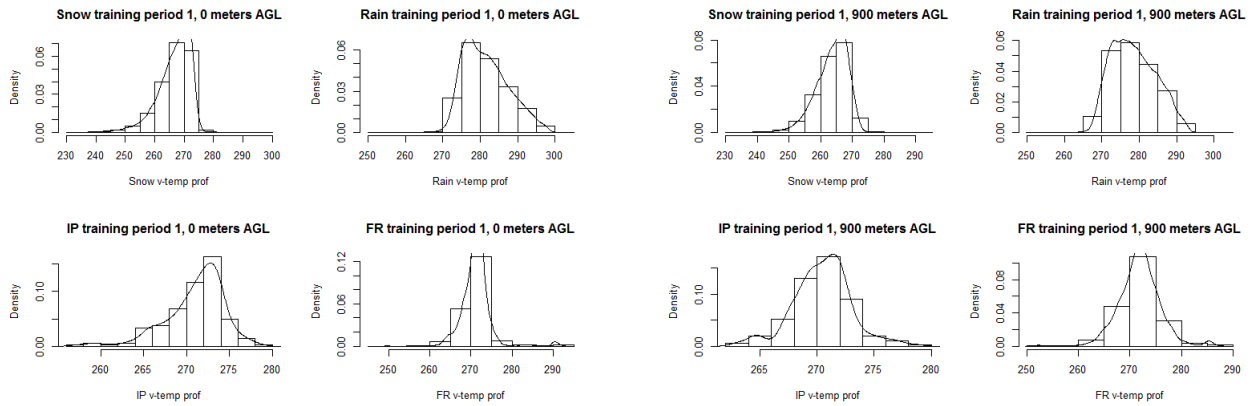

Figure 3: Left: The histogram for September 1996 - May 2001 at ground level **Right**: The histogram for September 1996 - May 2001 at 900 meters AGL

We see that it is hard to justify the assumption of multivariate normality because the histograms are skewed, which suggests that a transformation or a nonparametric density estimation is necessary.

Previous Methods

Quadratic Discriminant Analysis (QDA)

Classification analysis is where multivariate techniques are used to classify new objects into previously defined groups. In this data set, we have four classes for the different precipitation types. For a testing observation, we assign a probability for each precipitation type. Whichever precipitation type has the highest probability, we classify that observation as that precipitation type.

In order to come up with a classification probability, we use Bayes' theorem to allocate a new observation to the population with the largest posterior probability. The Bayes' rule for this data set is:

$$P(\text{ptype} = k|x_i) = \frac{\pi_{skm}\dot{\phi}_j(x_i)}{\sum_{i=4}^{4}\pi_{skm}\dot{\phi}_k(x_i)} \tag{3}$$

where k is the precipitation type, π_{skm} is the prior probability of an observation at location s in month m belonging to one of the 4 groups, $\phi_k(\cdot)$ is the corresponding probability density function (PDF) (we assume the multivariate normal PDF with mean $\hat{\mu}_k$ and variance $\hat{\Sigma}_k$, evaluated at observation x_i).

To find the density of each precipitation type, the first step is to calculate the mean and covariance for the 16 levels of the vertical temperature for each precipitation type. The mean temperatures are shown in Figure 1. Figure 4 is a visual representation of the covariances in the sixth training period (September 2001 - May 2006).

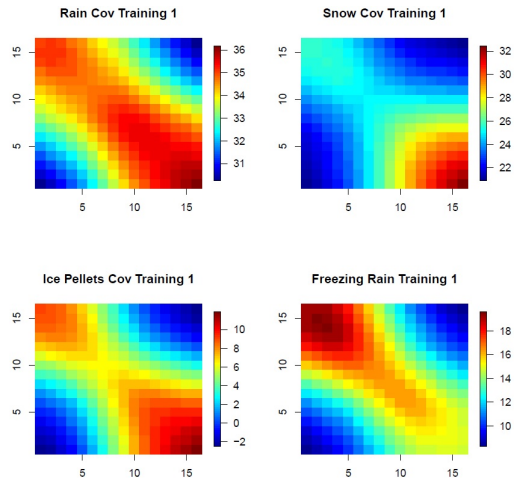

Figure 4: Covariance matrices for September 2001 - May 2006

We then compute the prior probabilities (π_{skm}), by looking at the observed frequency of a precipitation type at each station in each month. Below in Figure 5 is a visual representation of the prior probabilities in the month of December for each precipitation type.

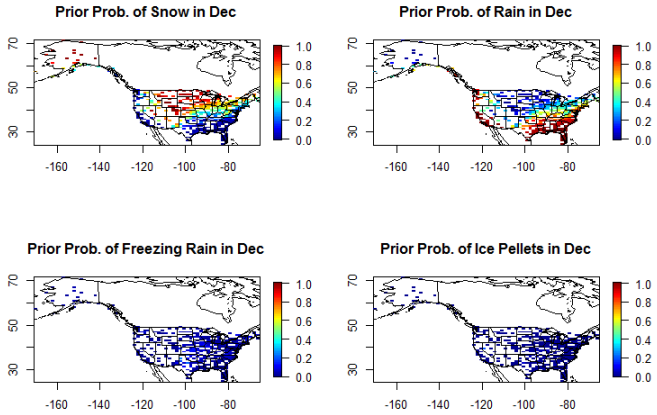

Figure 5: Prior probabilities across the US in December

QDA correctly classified 86.4% of the observations and the $BSS = -0.0172$ The negative BSS indicates that this method is actually worse at classifying precipitation types than just using the climatological probabilities. Therefore, further methods need to be explored.

Regularized Discriminant Analysis (RDA)

RDA is virtually the same as QDA, except that it takes into account the strong correlation in the vertical temperature profiles. RDA proposes an alternative to the covariance matrices used in the previous method. These alternative covariance matrices are characterized by two parameters (a and b), which are found by minimizing an estimate of future misclassification risk. So for this data set, our new covariance matrices will be defined as:

$$\tilde{\Sigma}_k = a\hat{\Sigma}_k + b\mathbf{I} \tag{4}$$

where a and b are constants, $\hat{\Sigma}_k$ are our old covariance matrices, and $\mathbf{I}$ is the identity matrix. The parameters are found by minimizing the negative of the BSS over the training set. Therefore, we will have 12 different values of a and b. . Below we have the regularized covariance matrices for each precipitation type in the second training period (September 1997 - May 2002).

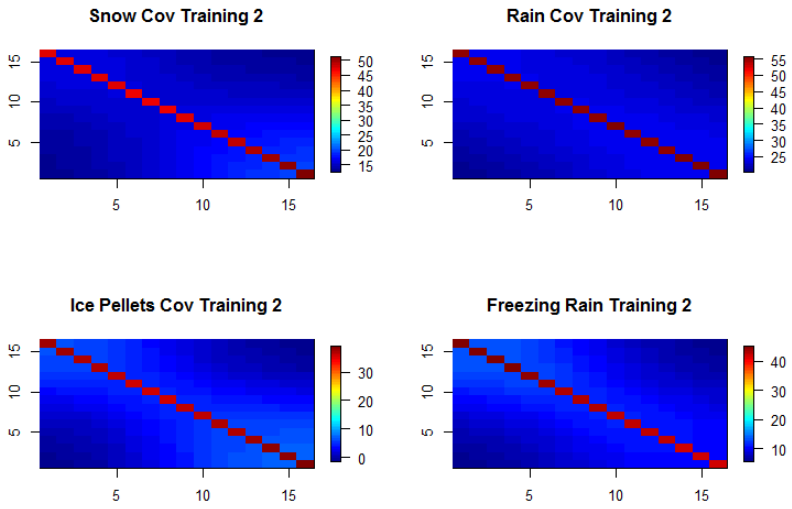

Figure 6: Regularized covarience matrices for September 1997 - May 2002

RDA correctly classified 93.7% of the observations (see table 1), and the $BSS = 0.5993$, which is notably better than QDA which had an accuracy of 86.4%. Below is a confusion matrix for RDA.

Table 1: Classification Probabilities

	Snow	**Rain**	**FRZA**	**IP**
Snow	0.2855	0.0286	0.0046	0.0015
Rain	0.0160	0.6500	0.0033	0.0012
FRZA	0.0003	0.0002	0.0002	0.0004
IP	0.0017	0.0022	0.0040	0.0005

The rows represent the actual observation, and the columns represent the predicted observation. .

Spatial Methods

Our main area of interest is the fact that the temperatures are highly correlated at each vertical location (see Figure 2). As we described in the previous methods section, RDA is a solution for this. However, because RDA is computationally expensive, we want to explore the possibility of reparameterizing the covariance matrices using spatial methods.

Our 31 vertical temperatures for each observation will be our attribute values.

$$Z(S_1), Z(S_2), ..., Z(S_{31}) = \text{Temperature AGL} \tag{5}$$

where our domain is only one-dimensional. We also need to meet the assumption of second order stationary (constant mean, and the covaraince only depends on the difference between s_i and s_j).

A first step is to look at the semivariograms for each precipitation type

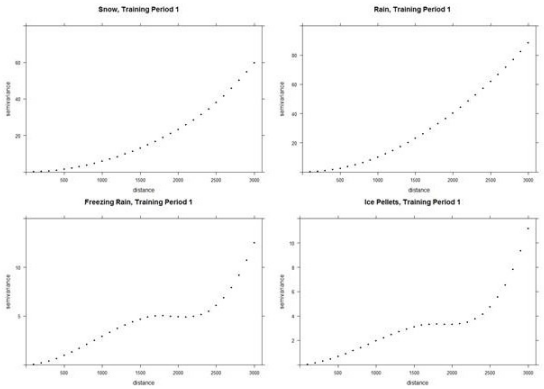

Figure 7: Semivariograms for September 1996-May 2001 across all vertical temperatures

We see that the semivariograms never level off, indicating that there is a spatially varying mean, which violates our assumption of second order stationary. Therefore, we need to remove the trend by modeling our mean structure. To do this, we parameterize the mean in terms of height above ground level. When we explore Figure 1, both snow and rain have a slight bend in their profiles, and ice pellets and freezing rain have an inversion that is more distinct. Using linear regression analysis, we saw that for rain and snow, a y and y^2 component were both significant at the 1% level with an $r^2 = .99$. For ice pellets and freezing rain, a y and y^3 component were both significant at the 1% level with an $r^2 = .96$ for ice pellets and an $r^2 = .95$ for freezing rain (see Equation 6).

$$\begin{aligned} \text{Rain: } & \text{temp} = \beta_0 + \beta_1 y + \beta_2 y^2 \\ \text{Snow: } & \text{temp} = \beta_0 + \beta_1 y + \beta_2 y^2 \\ \text{Ice Pellets: } & \text{temp} = \beta_0 + \beta_1 y + \beta_2 y^3 \\ \text{Freezing Rain: } & \text{temp} = \beta_0 + \beta_1 y + \beta_2 y^3 \end{aligned} \tag{6}$$

Now that the trend has been removed, we can recalculate the semivariograms based on the residuals from our mean structure.

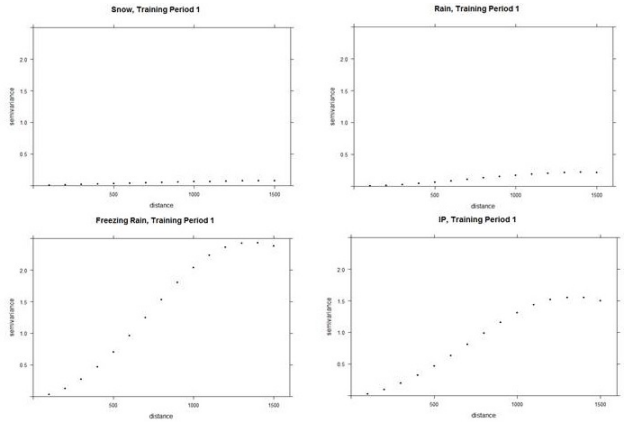

Figure 8: Semivariograms with the trend removed in training period 1

We see that now all the semivariograms level off. Also, these are on the same scale, so freezing rain and ice pellets have significantly more variability than rain and snow, which is expected due to the limited number of observation that each precipitation type has.

Models and fitting

The next step is to fit a model to these empirical semivariograms. Looking at each semivariogram, we have three reasonable choices for models (Gaussian, Exponential, and Spherical)

$$\begin{aligned}
\text{Gaussian: } \gamma(h,\underline{\theta}) &= \theta_1(1-\exp(-\frac{h^2}{\theta_2^2})) \\
\text{Exponential: } \gamma(h,\underline{\theta}) &= \theta_1(1-\exp(-\frac{h}{\theta_2})) \\
\text{Spherical: } \gamma(h,\underline{\theta}) &= \begin{cases} \theta_1(\frac{3h}{2\theta_2} - .5(\frac{h}{\theta_2})^3), & \text{if } 0 < h \leq \theta_2 \\ \theta_2, & \text{if } h > \theta_2 \end{cases}
\end{aligned} \tag{7}$$

where θ_1 is the sill, θ_2 is the range, and h is our distance matrix (16x16). When we fit these models to our semivariograms, we can come up with estimates for our sill and range using weighted least squares. The Gaussian model fits well for rain, ice pellets, and freezing rain, while the exponential model fit works best for snow (see Figure 9).

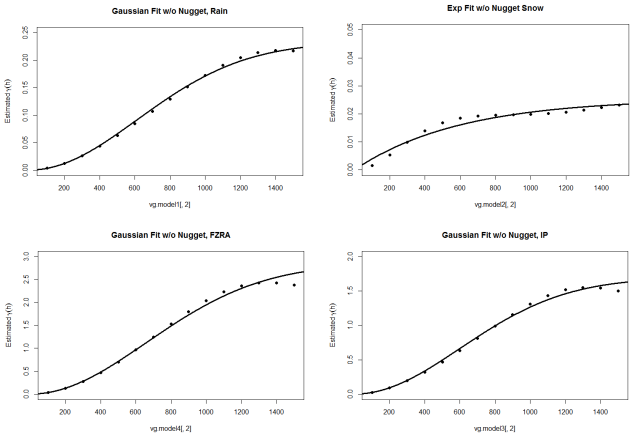

Figure 9: Semivariograms with model fits in training period 1

When we plug in the distance matrix into these different models, we extract the reparameterized covariance matrices. Figure 10 shows an image plot of these reparameterize covariance matrices.

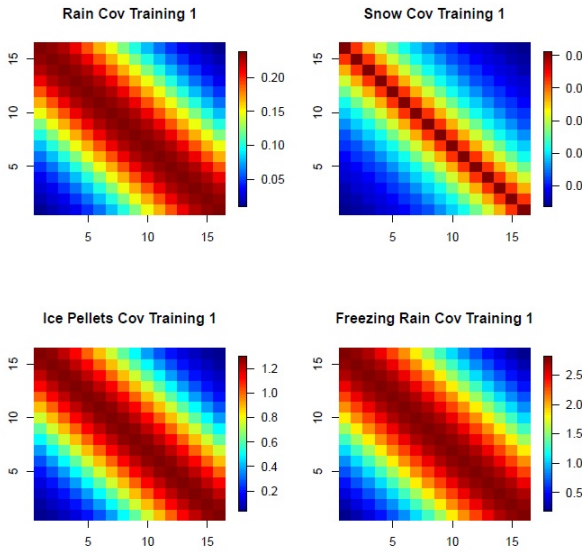

Figure 10: Reparameterized covariance matrices for training period 1 (September 1996-May 2001)

However, these covariance matrices caused problems because the determinants were virtually zero. This is a problem because the determinant of these covariance matrices is in the denominator of the multivariate normal PDF (Equation 8).

$$f_x(x_1, ..., x_k) = \frac{1}{\sqrt{2\pi^k|\Sigma|}} \exp(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)) \tag{8}$$

Moving forward, we decided to use semivariogram models that gave us covariance matrices with determinants as far a away from zero as possible. Figure 11 below shows which models achieved this goal.

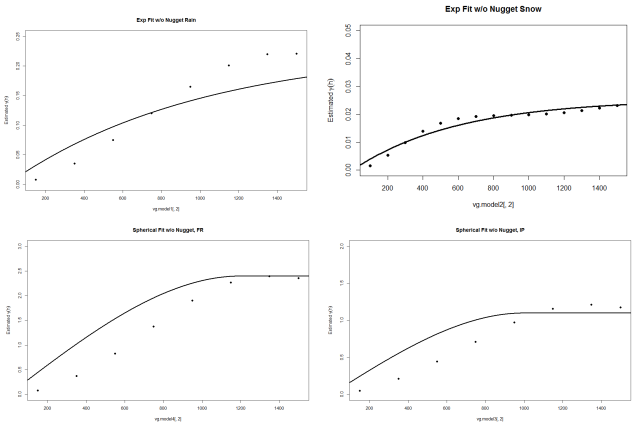

Figure 11: Semivariograms with model fits in training period 1

Rain and snow were refitted with an exponential model, and then freezing rain and ice pellets were refitted with a spherical model. Figure 12 shows the new covariance matrices.

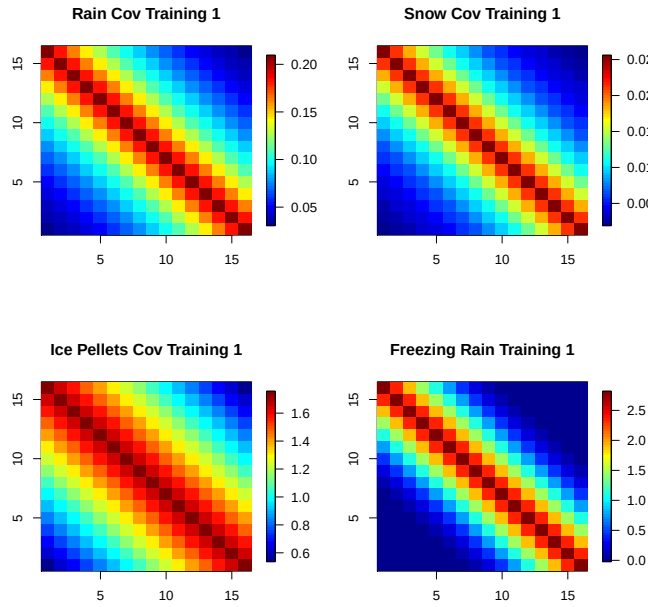

Figure 12: Reparameterized covariance matrices for training period 1

Again, while the determinants of these matrices are better than before, they are still very close to zero. A solution for this is to regularize these covariance matrices (see Equation 4). It takes about 20 hours in order to get the two parameters (a and b) for all the training periods. Below in table 2, we report what the a and b values are.

Table 2: a and b values for each training period

Training Period	a	b
1	11.3091	200.2834
2	127.862	100.9980
3	87.6131	121.7144
4	37.8365	121.5970
5	33.7211	117.4882
6	86.2086	112.4854
7	28.6657	143.1918
8	28.3790	157.5094
9	18.0906	169.1167
10	21.4062	169.8121
11	18.8647	164.9042
12	90.1429	128.6661

We speculate that is takes more computational time than when we first implemented RDA (approximately 3 times longer) because there is such a large range for the a's and b's. Below in Figure 13, we show the new regularized covariance matrices

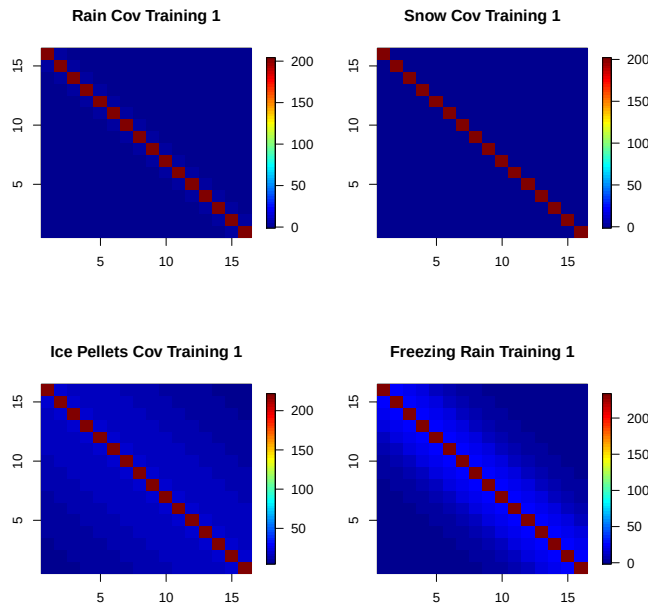

Figure 13: Regularized covariance matrices for training period 1

Results

The best way to visualize the performance of this method is to look at the confusion matrix (see Table 3).

Table 3: Confusion Matrix

	Snow	**Rain**	**FRZA**	**IP**
Snow	0.2887	0.0290	0.0057	0.0019
Rain	0.0132	0.6519	0.0063	0.0016
FRZA	0.0008	0.0000	0.0001	0.0000
IP	0.0008	0.0001	0.0000	0.0000

When we add up the diagonals, we see that our percentage of correct classifications is 94.076%. We notice that this method can predict rain and snow very well, but struggles to correctly classify freezing rain and ice pellets. To asses the reliability of this method, we look at the BSS (Equation 2).

$$BS = 0.09074$$

$$\Rightarrow BSS = 0.5848$$

Nonparametric Density Estimation

As we saw in Figure 3, the Guassian assumption does not hold; therefore, we explored the possibility of a nonparametric density estimate, rather than using the multivariate normal pdf in Bayes' Theorem. More specifically, we would use a multivariate kernel density estimation. The density estimate is composed of a choice of a plug-in bandwidth and a kernel function. The choice of the kernel function is not crucial to the accuracy of the density estimate, therefore the standard choice is the multivariate normal PDF. Therefore, our kernel density estimate (KDE) is:

$$f_k(x_1, x_2, ..., x_{16}) = \frac{1}{n_k b_{1,k} \cdots b_{16,k}} \sum_{i=1}^{n_k} K(u_1) \cdots K(u_{16}) \tag{9}$$

Where K is the multivariate normal PDF, which takes in parameter u_i

$$u_i = \frac{x_1 - x_{1,i}}{b_{1,k}}, ..., \frac{x_{16} - x_{16,i}}{b_{16,k}} \tag{10}$$

and each $x_{1,i}, ..., x_{16,i}$ is the vector of observation (i) in the given training period with a given precipitation type at each level (1-16). The $x_1, ..., x_{16}$ is the testing observation vertical temperature at each level. Also, $b_{i,k}$ is the bandwidth for the ith level for precipitation type k. For the bandwidth, we used Scott's plug-in bandwidth

$$\hat{b}_x = \hat{\sigma}_x N^{-1/d+4} \tag{11}$$

where d is the number of dimensions. Also $\hat{b}_x$ is a 16x16 matrix for each training period.

However, this KDE assumes that each $K(u_i)$ is independent, which we showed earlier is not the case (see figure 2). We will explore the possibility of a nonparametric density estimate after we make the dimensions independent through principal component analysis.

Principal Component Analysis (PCA)

Principal component analysis is the process of making orthogonal transformation of the data to make a set of observations of correlated variables into a set of linearly uncorrelated variables called principal components (PC's). In other words, it finds the underlying structure in the data. PCA is primarily used to reduce the dimensions in the data set, because it breaks down the data down into its basic parts, stripping away any unnecessary parts.

While dimension reduction will hopefully help with our classifications (especially when it comes to computation time), the most useful quality is that the components will now be uncorrelated which makes our dimensions independent, thus we can do a nonparametric density estimation on our vertical profiles.

We find the principal components using the covariance method. The first step is to find the eigenvectors and eigenvalues of the covariance matrix. The eigenvalues and the eigenvectors are ordered and paired (meaning the ith eigenvalue corresponds to the ith eigenvector).

$$\text{Eigenvalues: } \lambda_1 \geq \lambda_2 \geq ... \geq \lambda_p$$

$$\text{Eigenvectors: } \bar{e}_1, \bar{e}_2, ..., \bar{e}_p$$

The eigenvalues are ordered in decreasing order and the eigenvectors are ordered by the variation that they explain. Now the principal components (orthogonal transformation) are:

$$Y_1 = e_1^T X, \quad Y_2 = e_2^T X \quad ,..., \quad Y_p = e_p^T X$$

Therefore, the first principal component will have most of the variance explained. There are a couple of guidelines for choosing the number of principal components. One is to have enough PC's such that 90% of the variance is explained. A good way to visualize how much of the variance is explained is to use a scree plot, which plots the variance against the number of PC's (see Figure 14).

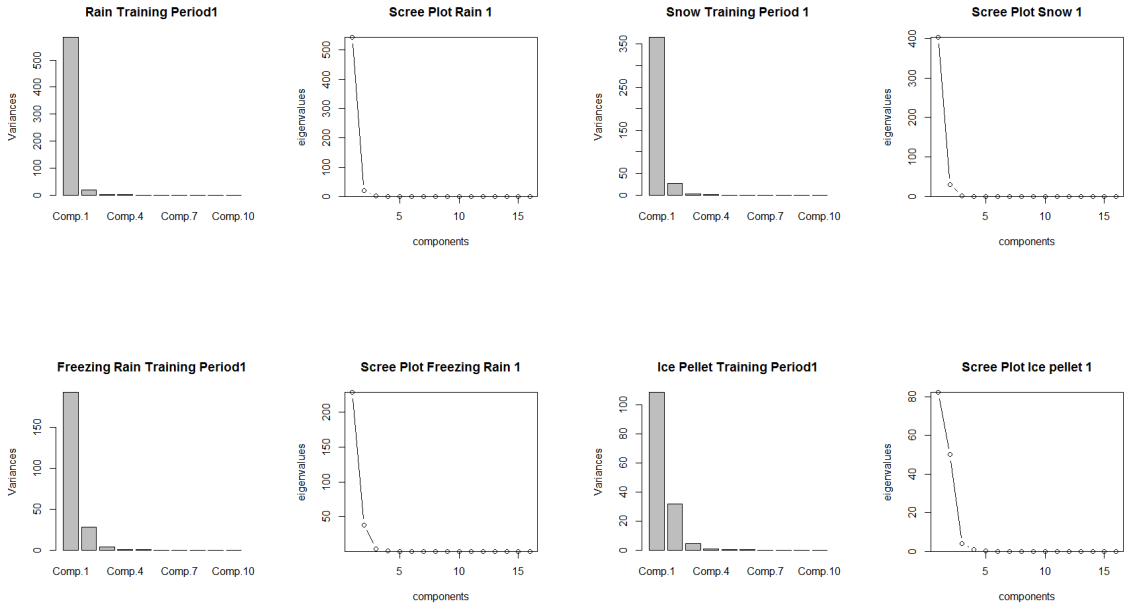

Figure 14: Scree plots for each precipitation type in training period 1

While rain and snow has over 99% of the variance explained in the first two components, ice pellets and freezing rain each take 3 components to have 99% of the variance explained. There is minimal cost in including

more PC's; therefore, we will use 3 PC's for each precipitation type. To show that the dimensions are now uncorrelated, we have the image plot of the correlations of the components below in Figure 15. We see that all of the off diagonals are zero.

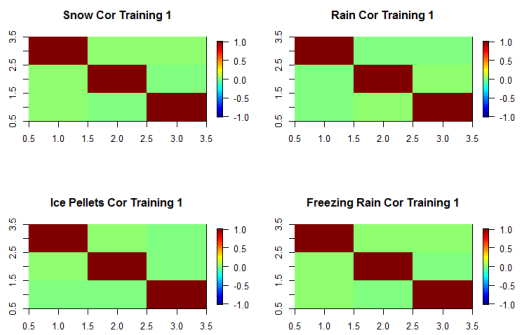

Figure 15: Correlations for each precipitation type's PCs for training period 1

The next step is to check the Gaussian assumption for the components. We look at the histograms for each PC for each precipitation type in Figure 16.

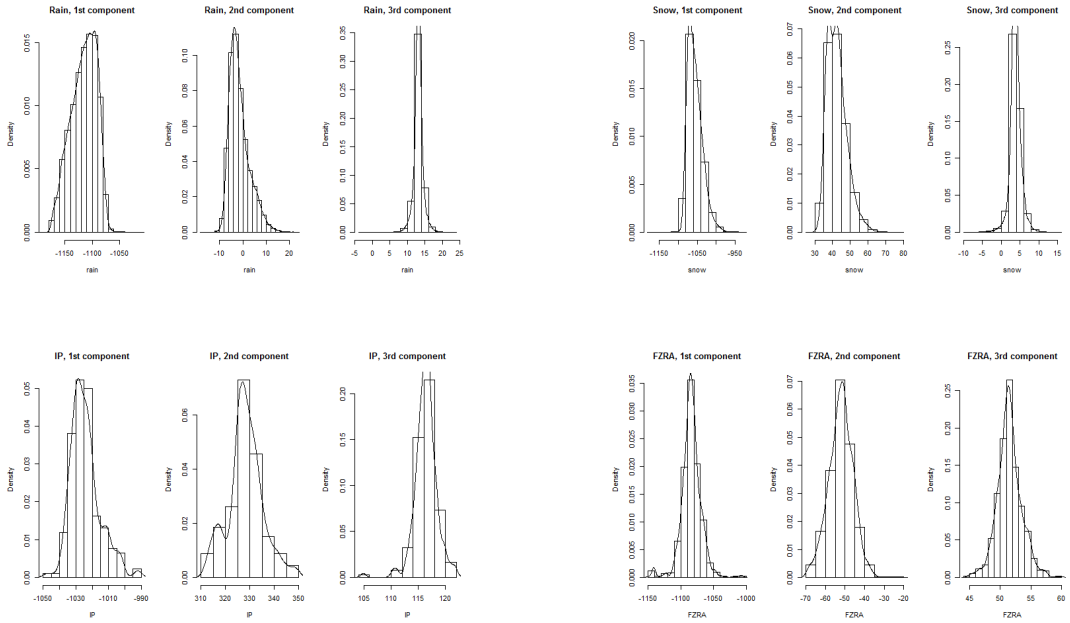

Figure 16: Histograms for each precipitation type's PCs for training period 1

The histograms seen in Figure 16 appear to be slightly skewed and do not follow a normal density curve. This is consistent among all 12 training periods. There are a few histograms scattered throughout the data that do have a normal density, however, the majority of the histograms are not normally distributed. Therefore, we will do a nonparametric density estimation of the PC's.

Nonparametric Density Estimation of Principal Components

We have the same set up for the KDE that we had earlier in this section (Equation 9), except now we only have 3 dimensions that are independent. Therefore, our new KDE is:

$$f_k(x_1, x_2, x_3) = \frac{1}{n_k b_{1,k} b_{2,k} b_{3,k}} \sum_{i=1}^{n_k} K(u_1) K(u_2) K(u_3) \tag{12}$$

Where K is the multivariate normal PDF, which takes in parameter u_i

$$u_1 = \frac{x_1 - x_{1,i}}{b_{1,k}}, \quad u_2 = \frac{x_2 - x_{2,i}}{b_{2,k}}, \quad u_3 = \frac{x_3 - x_{3,i}}{b_{3,k}} \tag{13}$$

and each $x_{1,i}, x_{2,i}, x_{3,i}$ is the vector of all observations in a training period of a certain precipitation type for that PC. Again, $b_{i,k}$ is the bandwidth for the ith component for precipitation type k. The `KS` package in `R` has a function `KDE` that will estimate the bandwidth automatically, using a Wand-Jones bandwidth estimation. We initially used this bandwidth estimation. Computing the KDE took around 20 minutes, which is significantly longer than using the multivariate normal density. However, since we used PCA to make the dimensions independent, we will not need to use RDA, which was the most computationally expensive method.

Results

Below in table 4 we computed the confusion matrix for the kernel density estimation using the internal Wand-Jones bandwidth estimate.

Table 4: Confusion Matrix (default bandwidth)

	Snow	**Rain**	**FRZA**	**IP**
Snow	0.2859	0.0228	0.0046	0.0015
Rain	0.0146	0.6529	0.0018	0.0009
FRZA	0.0021	0.0049	0.0054	0.0005
IP	0.0008	0.0004	0.0003	0.0006

We see a sizable increase in the ability of the method to predict freezing rain and ice pellets correctly over RDA. The percentage of correct classifications is 94.5%. Using equation 1 and 2 we calculated the BS and BSS.

$$BS = 0.0809$$

$$\Rightarrow BSS = 0.6298$$

We tried both the default bandwidth selection in the `KDE` function, as well as Scott's plug-in bandwidth (Equation 12). We found that using Scott's bandwidth we were able to get a slightly better BSS than when we used the default bandwidth, which uses the Wand and Jones bandwidth estimate (see Table 4).

For each of the training periods and precipitation type, the bandwidths will be a 3x3 matrix with zeros on the off diagonals, because the PC's are uncorrelated. Using Scott's bandwidth, we saw an increase in the overall performance of the method. Below in Table 5 is the confusion matrix:

Table 5: Confusion Matrix (Scott's bandwidth)

	Snow	**Rain**	**FRZA**	**IP**
Snow	0.2872	0.0226	0.0046	0.0015
Rain	0.0137	0.6546	0.0020	0.0008
FRZA	0.0018	0.0034	0.0052	0.0005
IP	0.0007	0.0003	0.0003	0.0006

We see that our percentage of correct classifications is 94.8%, we also recalculated our BS and BSS using the Equations 1 and 2. We get a slightly better percentage of correct classifications and a higher BSS when we use Scott's bandwidth compared to the default bandwidth estimation in `KDE`.

$$BS = 0.0765$$

$$\Rightarrow BSS = 0.6500$$

Conclusion and Discussion

The goal of this project was to reliably forecast which type of precipitation will fall. RDA showed significant improvement over QDA. Since RDA is fairly computationally expensive, we first explored the possibility of accounting for the correlations through spatial methods. We removed the trend of a spatially varying mean, and then reparameterized the covariance matrices based on different semivariogram models. However, this method ended up not working without regularizing the covariance matrices. The fact that our method did not work before regularization still needs further analysis. Below we have all of the different covariance matrices from beginning to end.

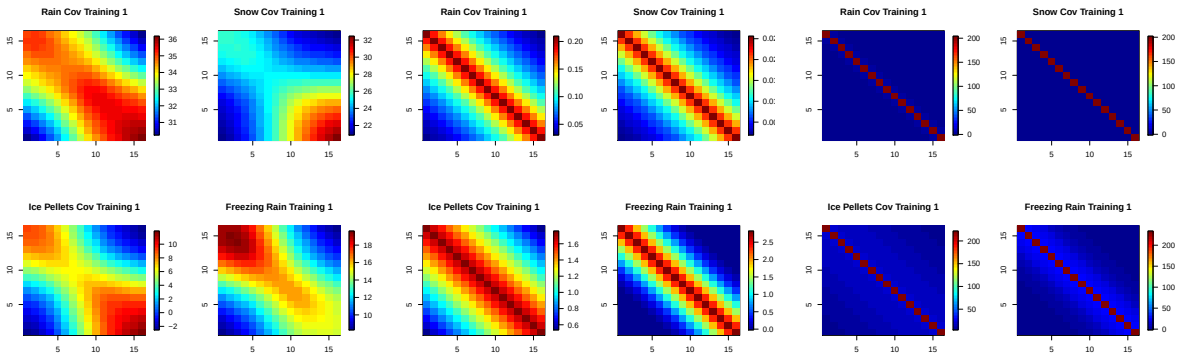

Figure 17: From left to right: First covariance matrices, Reparameterized covariance matrices, Regularized covariance matrices

The percentage of correct classifications for our method is marginally better than RDA, but our BSS is lower. This is expected because this method struggles to correctly classify ice pellets and freezing rain. RDA is certainly better option over this method. It is hard to justify a 0.5% increase in correct classifications for a method that takes 3 times as long computationally. Also, RDA is better at getting specific forecasts (obtaining probabilities closer to 1) based on the higher BSS.

However, the KDE of the PC's proved to be the best method. It had the highest percentage of correct classifications, a higher BSS, and it was significantly less computationally expensive than RDA. Below, we have all of the confusion matrices for each method:

Table 6: Confusion Matrices: Top left: QDA, Top right: RDA, Bottom left: Spatial, Bottom right: KDE with Scott's bandwidth

QDA	**Snow**	**Rain**	**FRZA**	**IP**
Snow	0.2311	0.0361	0.0040	0.0009
Rain	0.0609	0.6270	0.0025	0.0011
FRZA	0.0091	0.0154	0.0049	0.0005
IP	0.0023	0.0025	0.0007	0.0010

RDA	**Snow**	**Rain**	**FRZA**	**IP**
Snow	0.2855	0.0286	0.0046	0.0015
Rain	0.0160	0.6500	0.0033	0.0012
FRZA	0.0003	0.0002	0.0002	0.0004
IP	0.0017	0.0022	0.0040	0.0005

Spatial	**Snow**	**Rain**	**FRZA**	**IP**
Snow	0.2887	0.0290	0.0057	0.0019
Rain	0.0132	0.6519	0.0063	0.0016
FRZA	0.0008	0.0000	0.0001	0.0000
IP	0.0008	0.0001	0.0000	0.0000

KDE	**Snow**	**Rain**	**FRZA**	**IP**
Snow	0.2872	0.0226	0.0046	0.0015
Rain	0.0137	0.6546	0.0020	0.0008
FRZA	0.0018	0.0034	0.0052	0.0005
IP	0.0007	0.0003	0.0003	0.0006

Table 7: Comparing all the methods

Method	**Accuracy**	**BSS**	**Comp time**
QDA	86.4%	-0.0172	$\sim$ 5 min
RDA	93.6%	0.5990	$\sim$ 6 hrs
Spatial Method	94.1%	0.5848	$\sim$ 20 hrs
KDE (default bandwidth)	94.5%	0.6298	$\sim$ 20 min
KDE (Scott's bandwidth)	94.8%	0.6500	$\sim$ 17 min

Above in Table 7, we have the overall comparison between all 5 methods. While all three of our methods provided better deterministic metrics, the spatial method had a lower BSS and is too computationally expensive to justify using it. The combination of PCA and nonparametric density estimation proved to be the most valuable method. The KDE using Scott's plug-in bandwidth provided the best accuracy and is significantly less computationally expensive than RDA. Overall it would be advisable to use the KDE with Scott's bandwidth for the best forecast models.

References

[1] Friedman, Jerome H. "Regularized Discriminant Analysis." Stanford University (1988): Web. 13 Apr. 2016.

[2] Herring, Amanda. "Spatial Statistics Course Notes." Web. Spring 2016

[3] Scheuerer, Michael. NOAA Precipitation Data, 2015. Web. 25 Jan. 2016.

[4] Waller, Lance A., and Carol A. Gotway. Applied Spatial Statistics for Public Health Data. Hoboken, NJ: John Wiley & Sons, 2004.

[5] Shlens, Jon. "A TUTORIAL ON PRINCIPAL COMPONENT ANALYSIS." 25 Mar. 2003. Web. 25 Apr. 2016.

[6] Duong, Tarn. "Package 'ks'" R-project.org. 4 Apr. 2016. Web. 25 Apr. 2016.