# Determining Optimal Wind Turbine Locations

Egan McClave, emcclave

11 October 2018

## Contents



## List of Figures



## List of Tables

# 1 Introduction

Wind is a renewable energy resource and if humanity can effectively harness the use and collection of energy from wind then we can significantly reduce the carbon footprint humanity places on the environment. Despite the necessity of wind energy many people object to having wind turbines near populated areas. This makes it difficult to locate areas where wind turbines can be placed strategically. Thus, it is of interest to environmentalists, scientists, and statisticians to determine the location that has the largest poential for generating energy (for a given wind speed $v$ traveling through a wind turbine, the estimated energy is proportional to $v^3$). The main reserach goals this paper will attempt to answer are the following:

- which location has the largest $\mathbb{E}[v^3]$ value?
- what is the corresponding standard error value for this estimate?

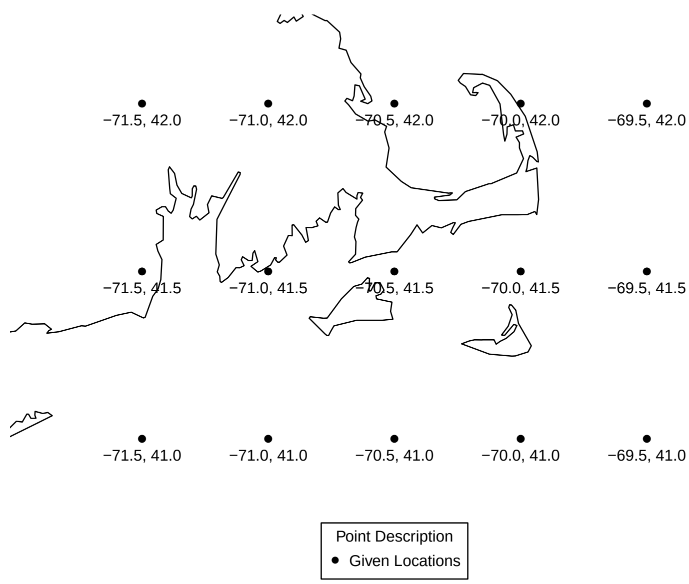


Figure 1: Spatial Grid of Locations for Data Collected

The variables within our data are `Year` which ranges from 2014 to 2018, `Month` which displays the month the data was collected on, `Day` which displays the day the data was collected on, `Time` which displays the specific time, every 6 hours, the data was collected in, and 15 spatial coordinates: (‘`-69.5, 41.0`’, ‘`-70.0, 41.0`’, ‘`-70.5, 41.0`’, ‘`-71.0, 41.0`’, ‘`-71.5, 41.0`’, ‘`-69.5, 41.5`’, ‘`-70.0, 41.5`’, ‘`-70.5, 41.5`’, ‘`-71.0, 41.5`’, ‘`-71.5, 41.5`’, ‘`-69.5, 42.0`’, ‘`-70.0, 42.0`’, ‘`-70.5, 42.0`’, ‘`-71.0, 42.0`’, ‘`-71.5, 42.0`’) for a total of 6932 observations. Figure 1 depicts the orientation of the grid of these spatial coordinates where data were collected.

# 2 Methods

## 2.1 Exploratory Data Analysis

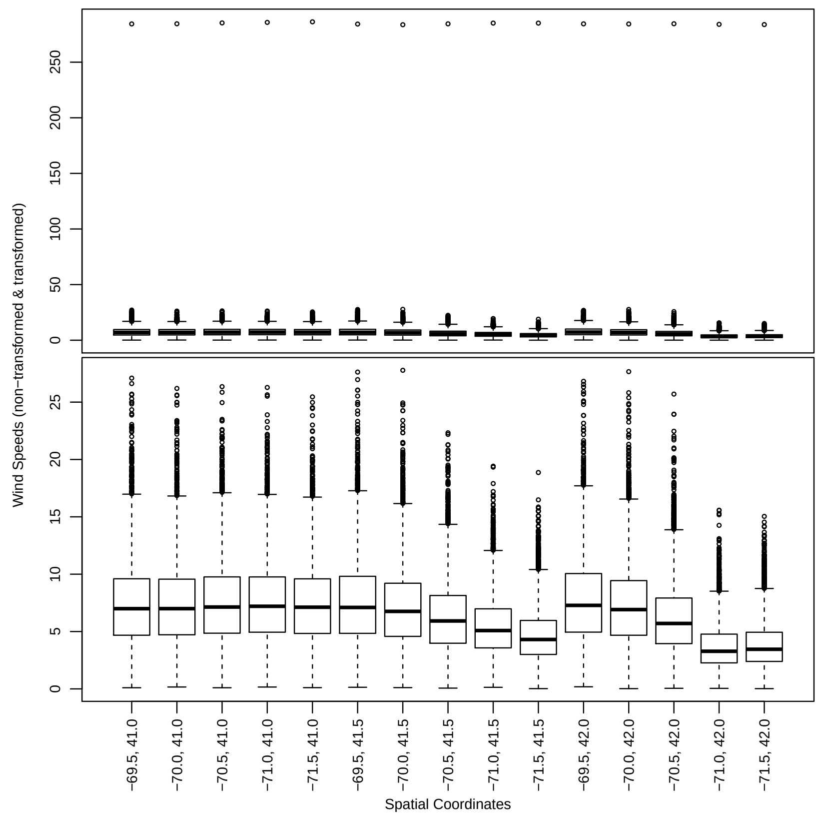


Figure 2: Univariate Analysis for Spatial Coordinates

Initial inspection of the data, as seen in the top plot of Figure 2, depicts a large outlier skewing the distributions for each spatial point in the side-by-side boxplot. This outlier corresponds to data collected at noon on May 28$^{th}$, 2018. For this observation the recorded wind speeds for the 15 different points are all ≈280 m/s. This is a cause for concern because the maximum wind speed currently recorded is 113 m/s. Therefore, it is likely that the observation is an outlier due to some data collection malfunction. The bottom plot in Figure 2 is a side-by-side boxplot for each of the different spatial points after removing the discussed point. The distributions are now more clear and so the rest of the data analysis will be done with this observation removed (there are now 6931 observations).

The variable we wish to analyze is the estimated energy generated by wind, so it is not in our interest to continue to analyze the data while not transformed. Figure 3 displays the cubed wind speeds over time across all 15 locations. The data is fit with a smoothing spline (colored in Red) which tells us that there is some sort of seasonal trend that we can extract from the data. Table 1 displays the summary statistics for the transformed dataset. As with Figure 2, the information describes the distributions to be nearly identical except for the bottom right locations in the grid which just slightly deviate with their summary statistics.

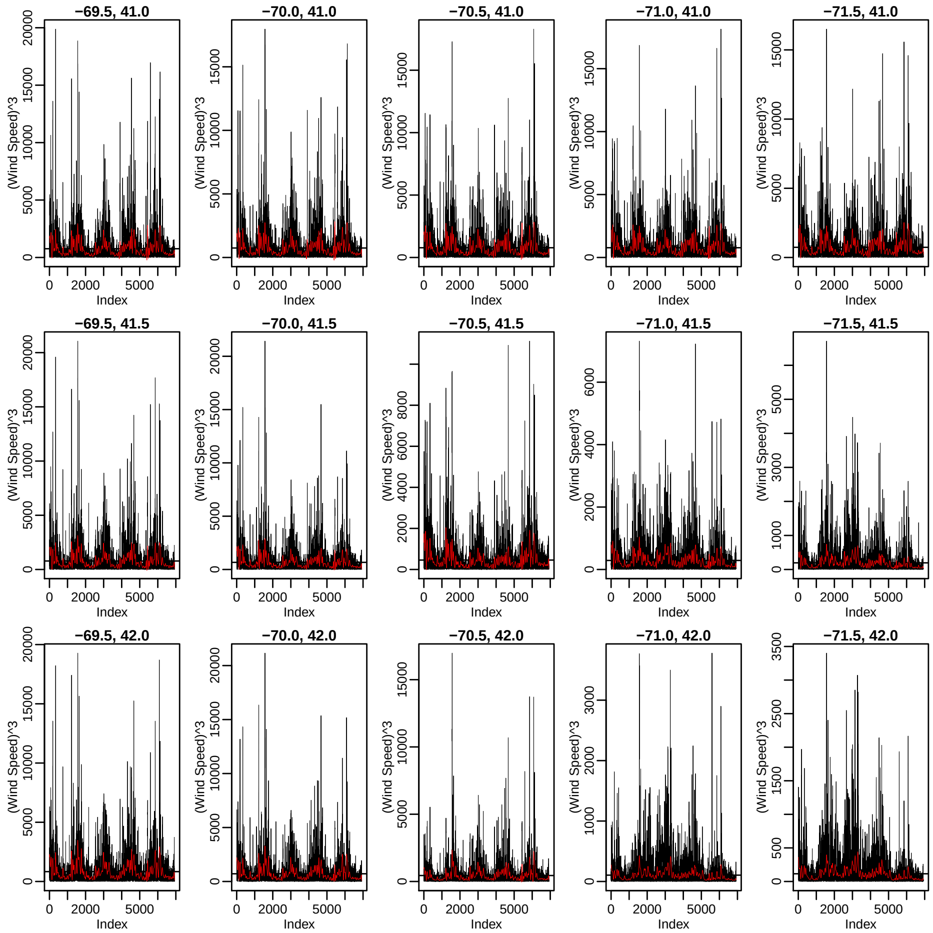


Figure 3: (Wind Speeds)$^3$ Over Time

| | Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. | Std. Dev |
|---|---|---|---|---|---|---|---|
| -69.5, 41.0 | 0.00 | 102.39 | 342.01 | 760.67 | 884.52 | 19885.43 | 1299.89 |
| -70.0, 41.0 | 0.00 | 104.91 | 342.52 | 745.40 | 875.33 | 17970.64 | 1232.07 |
| -70.5, 41.0 | 0.00 | 114.44 | 363.51 | 778.80 | 931.05 | 18295.92 | 1221.13 |
| -71.0, 41.0 | 0.00 | 120.76 | 373.60 | 775.23 | 929.88 | 18146.04 | 1194.67 |
| -71.5, 41.0 | 0.00 | 112.95 | 360.92 | 734.05 | 883.21 | 16506.25 | 1119.05 |
| -69.5, 41.5 | 0.00 | 113.48 | 357.90 | 783.05 | 945.39 | 21067.15 | 1297.69 |
| -70.0, 41.5 | 0.00 | 95.89 | 308.68 | 666.53 | 781.38 | 21427.98 | 1117.70 |
| -70.5, 41.5 | 0.00 | 63.08 | 207.79 | 458.83 | 539.93 | 11125.49 | 774.03 |
| -71.0, 41.5 | 0.00 | 45.69 | 131.52 | 288.95 | 339.97 | 7317.28 | 460.86 |
| -71.5, 41.5 | 0.00 | 27.08 | 79.91 | 194.70 | 212.56 | 6712.47 | 340.05 |
| -69.5, 42.0 | 0.01 | 121.27 | 385.84 | 832.63 | 1016.74 | 19283.38 | 1330.14 |
| -70.0, 42.0 | 0.00 | 102.36 | 330.40 | 714.59 | 841.42 | 21159.17 | 1184.09 |
| -70.5, 42.0 | 0.00 | 61.48 | 185.59 | 442.95 | 497.46 | 16977.64 | 812.26 |
| -71.0, 42.0 | 0.00 | 11.67 | 35.32 | 105.17 | 108.76 | 3779.19 | 211.73 |
| -71.5, 42.0 | 0.00 | 13.72 | 41.22 | 114.82 | 120.47 | 3402.04 | 218.57 |

Table 1: Summary Statistics for Cubed Wind Speed

## 2.2 Checking Assumptions

Figure 4 displays the autocovariance function lagged out to the equivalent of 2 years of data (365 days/year $\times$ 4 observations/day $\times$ 2 years = 2120 observations). This plot supports the ideas from the previous subsection. There does exist a seasonal trend with respect to time in our data. It is also interesting to note that for '-71.0, 42.0' and '-71.5, 42.0' the acf plots have much thicker seasonal trends which seems to imply that the wind speeds collected from these locations are quite variable over time.

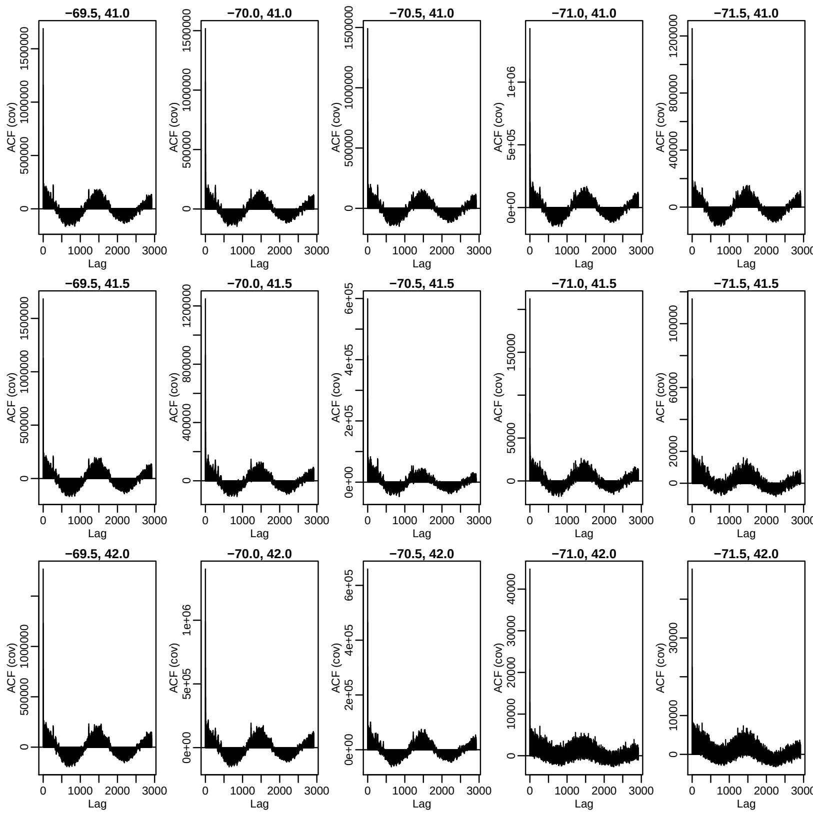


Figure 4: Autocovariance Function for Each Spatial Location of Seasonally Unadjusted Data

The data are detrended by determining the average wind speed for every 6 hour observation for every day of every month of every year which will create a 1460 by 15 dimension trend matrix to represent a full year (365 $\times$ 4 = 1460). This trend matrix is then subtracted from each year of our data to detrend with respect to the `Time` variable.

Figure 5 displays the autocovariance function again lagged out to 2120 observations of the seasonally adjusted dataset. These plots depict a detrended data set as there is no longer an indication of a pattern of the covariance with respect time. Table 2 displays the summary statistics for the seasonally adjusted transformed wind data. The mean column in this table is nearly 0 which makes sense because we have detrended the data.

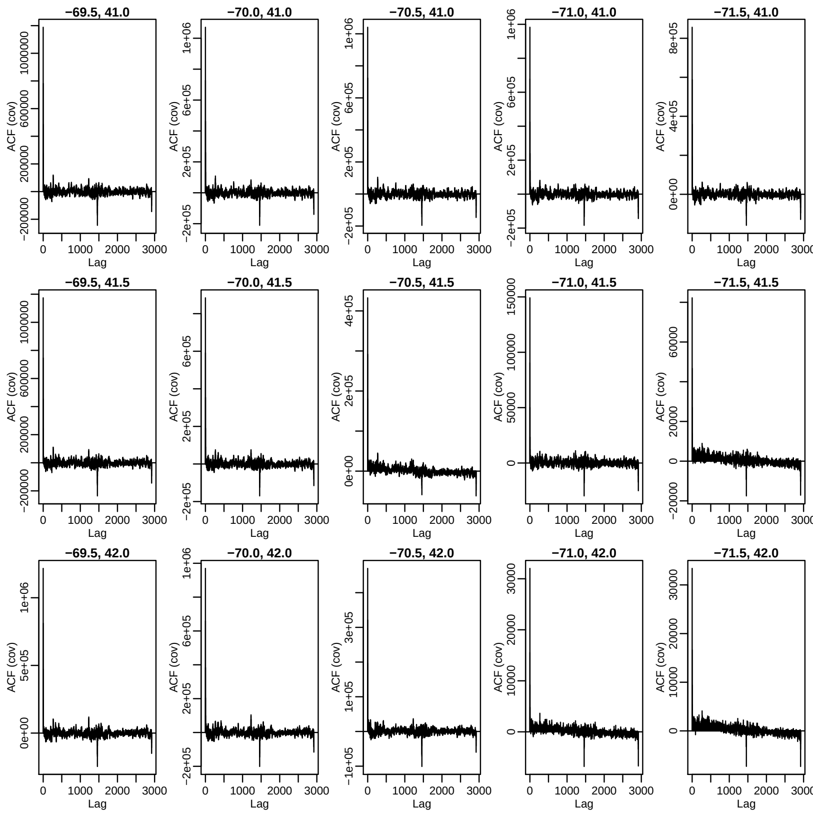


Figure 5: Autocovariance Function for Each Spatial Location of Seasonally Adjusted Data

| | Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. | Std. Dev |
|---|---|---|---|---|---|---|---|
| -69.5, 41.0 | -5143.36 | -383.05 | -85.70 | 0.12 | 214.05 | 15249.95 | 1090.15 |
| -70.0, 41.0 | -4990.89 | -379.86 | -92.87 | 0.16 | 217.22 | 13033.30 | 1035.02 |
| -70.5, 41.0 | -5661.11 | -403.77 | -100.72 | 0.19 | 224.06 | 12451.63 | 1020.32 |
| -71.0, 41.0 | -5504.24 | -401.09 | -102.11 | 0.18 | 224.11 | 12408.10 | 991.37 |
| -71.5, 41.0 | -4822.97 | -383.07 | -97.89 | 0.14 | 212.35 | 11678.51 | 925.61 |
| -69.5, 41.5 | -4929.19 | -389.29 | -95.42 | 0.10 | 225.23 | 15334.39 | 1083.67 |
| -70.0, 41.5 | -5334.66 | -337.67 | -88.68 | 0.12 | 192.76 | 15831.72 | 940.46 |
| -70.5, 41.5 | -2731.90 | -245.42 | -70.32 | 0.09 | 135.05 | 8141.51 | 657.84 |
| -71.0, 41.5 | -1949.10 | -153.30 | -39.50 | 0.09 | 80.70 | 5345.63 | 386.24 |
| -71.5, 41.5 | -1601.20 | -104.36 | -27.24 | 0.04 | 52.01 | 5056.90 | 286.80 |
| -69.5, 42.0 | -5950.98 | -422.72 | -111.91 | 0.11 | 249.16 | 14066.76 | 1103.26 |
| -70.0, 42.0 | -5063.28 | -351.41 | -91.63 | 0.11 | 205.25 | 15721.05 | 984.86 |
| -70.5, 42.0 | -3907.56 | -220.74 | -59.63 | 0.09 | 121.06 | 12855.84 | 685.58 |
| -71.0, 42.0 | -1067.67 | -55.61 | -13.18 | 0.04 | 23.67 | 2895.03 | 179.00 |
| -71.5, 42.0 | -1083.91 | -58.79 | -14.36 | 0.03 | 26.88 | 2566.36 | 182.76 |

Table 2: Summary Statistics for Seasonally Adjusted Cubed Wind Speed

Tables 3 and 4 display the covariance matrix between the data collected for the 6 leftmost and 6 right most points in the grid. We can see that overall among the two $6 \times 6$ matrices the covariances appear very roughly similar overall. We can also utilize these matrices to point out that the covariance between locations and their nearby neighbors are similar between the two matrices until you consider the locations where data were collected in-land where they slightly differ.

| | -69.5, 41.0 | -70.0, 41.0 | -69.5, 41.5 | -70.0, 41.5 | -69.5, 42.0 | -70.0, 42.0 |
|---|---|---|---|---|---|---|
| -69.5, 41.0 | 1188420.00 | 1093033.70 | 1121352.44 | 934594.81 | 1051368.80 | 914150.72 |
| -70.0, 41.0 | 1093033.70 | 1071268.24 | 1034728.77 | 910674.53 | 990065.55 | 874507.53 |
| -69.5, 41.5 | 1121352.44 | 1034728.77 | 1174348.50 | 963723.58 | 1138371.76 | 978817.55 |
| -70.0, 41.5 | 934594.81 | 910674.53 | 963723.58 | 884458.96 | 956416.95 | 872266.28 |
| -69.5, 42.0 | 1051368.80 | 990065.55 | 1138371.76 | 956416.95 | 1217175.69 | 1042732.45 |
| -70.0, 42.0 | 914150.72 | 874507.53 | 978817.55 | 872266.28 | 1042732.45 | 969940.32 |

Table 3: Covariance Matrix for 'East Points' for Seasonally Adjusted Cubed Wind

| | -71.0, 41.0 | -71.5, 41.0 | -71.0, 41.5 | -71.5, 41.5 | -71.0, 42.0 | -71.5, 42.0 |
|---|---|---|---|---|---|---|
| -71.0, 41.0 | 982813.87 | 881714.30 | 334197.71 | 208190.08 | 109950.50 | 107491.55 |
| -71.5, 41.0 | 881714.30 | 856745.33 | 311040.24 | 204889.64 | 104843.53 | 103592.49 |
| -71.0, 41.5 | 334197.71 | 311040.24 | 149179.52 | 101345.78 | 58130.90 | 57032.04 |
| -71.5, 41.5 | 208190.08 | 204889.64 | 101345.78 | 82256.50 | 45213.55 | 47101.72 |
| -71.0, 42.0 | 109950.50 | 104843.53 | 58130.90 | 45213.55 | 32040.32 | 30339.67 |
| -71.5, 42.0 | 107491.55 | 103592.49 | 57032.04 | 47101.72 | 30339.67 | 33401.01 |

Table 4: Covariance Matrix for 'West Points' for Seasonally Adjusted Cubed Wind

The information from Figure 5 and from Tables 3 and 4 point to the assumptions of having a stationary and isotropic dataset after detrending the data.

# 3 Results

## 3.1 Fitted Correlation Calculation

We can estimate a covariance function using the correlation function between points because of our assumption that the seasonally adjusted data have both stationarity and isotrophy (just as we did in Lecture 7/8). Figure 6 plots the distances between each of the locations along with their correlations. Since the data are collected on a grid we see discrete values along the x-axis on our plot. Ideally we would have a continuous spread of points for all distances to estimate an exponential correlation function. Since this is not possible due to our data collection constraints we can average all the points of distances between 40 and 60 kms and use this value to determine a rough estimate for how much the correlation changes in 50 kms. Finally, we can use non-linear least squares to calculate a more concrete estimate of the exponential decay model. This final estimated model is featured in Figure 6 by the black curve.

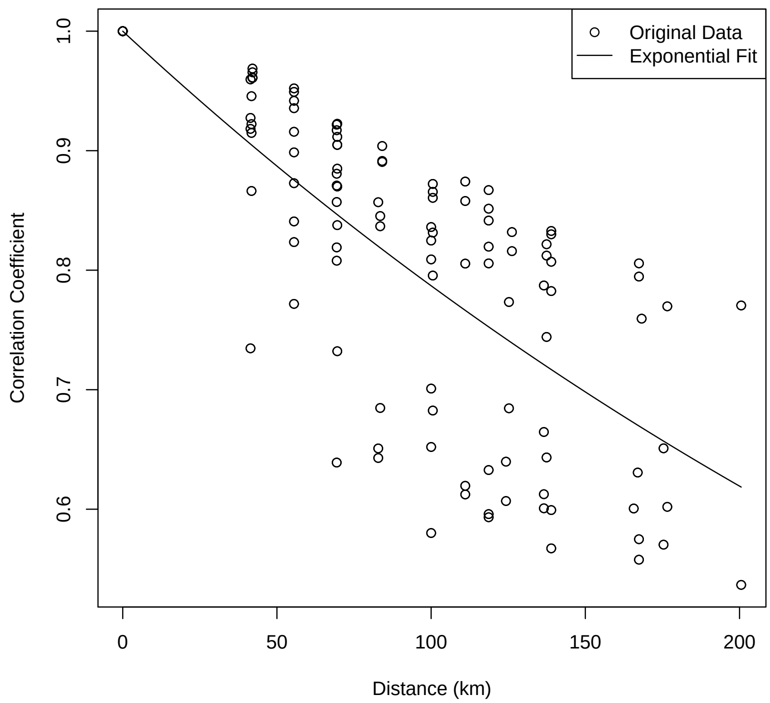


Figure 6: Correlation over Distances Fitted with Exponential Decay

### 3.2 Kriging Given Locations

We can now utilize the concept of Kriging which uses the exponentially fitted correlations to calculate the 14 different slope coefficients for each of the 15 locations. There is no need to calculate an intercept because we have seasonally adjusted the data so the means across all locations must be centered at approximately 0 (which they are as seen in Table 2). We can use these beta coefficients to calculate predictions for every 6 hour observation for a particular spatial location. This process is described by Equation 1. For a given spatial location $i$, predict all 6931 observations of our data utilizing sum of the mean trend for location $i$ and the multiplication of the data, not including location $i$, and $\beta$ the estimated coefficients from the remaining locations. This process is calculated for all 15 initial locations and the mean of the predictions for each location is reported in Table 5 along with their standard errors.

$$\underset{6931\times 1}{\widehat{v_i^3}} = \underset{1\times 1}{\texttt{mean}(\text{Trend}_{\texttt{i}})} + \underset{6931\times 14}{\text{Data}_{-i}} \times \underset{14\times 1}{\beta} \tag{1}$$

| | Estimates | Std. Errs |
|---|---|---|
| -69.5, 41.0 | 765.48 | 269.14 |
| -70.0, 41.0 | 751.44 | 269.14 |
| -70.5, 41.0 | 786.62 | 269.14 |
| -71.0, 41.0 | 783.75 | 269.14 |
| -71.5, 41.0 | 743.13 | 269.14 |
| -69.5, 41.5 | 788.04 | 269.14 |
| -70.0, 41.5 | 672.95 | 269.14 |
| -70.5, 41.5 | 464.35 | 269.14 |
| -71.0, 41.5 | 290.79 | 269.14 |
| -71.5, 41.5 | 195.33 | 269.14 |
| -69.5, 42.0 | 838.40 | 269.14 |
| -70.0, 42.0 | 720.49 | 269.14 |
| -70.5, 42.0 | 445.91 | 269.14 |
| -71.0, 42.0 | 105.00 | 269.14 |
| -71.5, 42.0 | 114.85 | 269.14 |

Table 5: Expected Wind Power Across all Initial Locations (with Standard Error)

## 3.3 Kriging Hypothesized Locations

The inital largest estimate for power is at '`-69.5, 42.0`' so it is important to approximate the power of nearby points to determine if we have calculated a local maximum instead of the global maximum within our grid. The new search space is limited to within the grid for interpolation purposes because extrapolation of space seems dubious as we do not know if the underlying assumptions we have made so far exist outside of our initial domain. Figure 7 describes the new search space in relation to the initial grid locations.

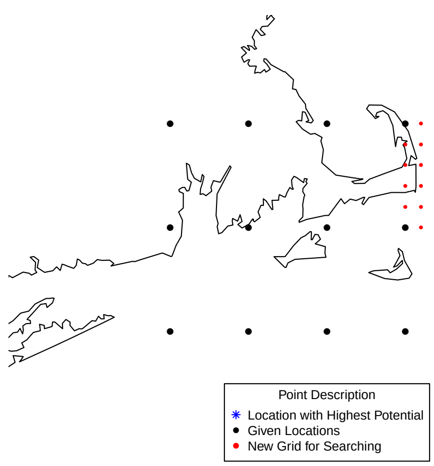


Figure 7: Spatial Grid of Original Locations and New Hypothesized Search Zone

The procedure for this kriging operation is nearly identical to the previous section. First, we define the points used for the exponential decay model to the correlation between distances of the four corners where we have known data and the hypothesized point in space. The correlations are then used for calculating the coefficients just as before. This slightly altered process is described in Equation 2. For a given unknown spatial coordinate $i$, predict all the 6931 observations by taking the sum of the global mean across all the four known points and the multiplication of wind power at these locations and the estimated coefficients. This process is calculated for all 32 new points and the mean of these predictions for each location is reported in Table 6 along with their standard errors.

$$\underset{6931\times 1}{\widehat{v_i^3}} = \underset{1\times 1}{\texttt{mean}(\text{Trend})} + \underset{6931\times 4}{\text{Data}} \times \underset{4\times 1}{\beta} \tag{2}$$

| | Estimates | Std. Errs |
|---|---|---|
| -69.9, 41.5 | 771.85 | 16.47 |
| -70.0, 41.5 | 771.85 | 16.47 |
| -69.5, 41.6 | 771.81 | 16.47 |
| -69.6, 41.6 | 771.83 | 16.47 |
| -69.7, 41.6 | 771.84 | 16.47 |
| -69.8, 41.6 | 771.85 | 16.47 |
| -69.9, 41.6 | 771.85 | 16.47 |
| -70.0, 41.6 | 771.84 | 16.47 |
| -69.5, 41.7 | 771.81 | 16.47 |
| -69.6, 41.7 | 771.82 | 16.47 |
| -69.7, 41.7 | 771.83 | 16.47 |
| -69.8, 41.7 | 771.84 | 16.47 |
| -69.9, 41.7 | 771.84 | 16.47 |
| -70.0, 41.7 | 771.84 | 16.47 |
| -69.5, 41.8 | 771.81 | 16.47 |
| -69.6, 41.8 | 771.82 | 16.47 |
| -69.7, 41.8 | 771.83 | 16.47 |
| -69.8, 41.8 | 771.83 | 16.47 |
| -69.9, 41.8 | 771.83 | 16.47 |
| -70.0, 41.8 | 771.83 | 16.47 |
| -69.5, 41.9 | 771.81 | 16.47 |
| -69.6, 41.9 | 771.82 | 16.47 |
| -69.7, 41.9 | 771.82 | 16.47 |
| -69.8, 41.9 | 771.82 | 16.47 |
| -69.9, 41.9 | 771.83 | 16.47 |
| -70.0, 41.9 | 771.83 | 16.47 |
| -69.5, 42.0 | 771.81 | 16.47 |
| -69.6, 42.0 | 771.81 | 16.47 |
| -69.7, 42.0 | 771.81 | 16.47 |
| -69.8, 42.0 | 771.82 | 16.47 |
| -69.9, 42.0 | 771.82 | 16.47 |
| -70.0, 42.0 | 771.82 | 16.47 |

Table 6: Expected Wind Power Across New Search Space (with Standard Error)

# 4 Discussion

Based on the results of our analysis it appears the location with the highest potential wind power is '`-69.5, 42.0`' with a value of $\approx 838(m/s)^3$ and a standard error of 269.14. Although the results seem adequate, they should be taken with a grain of salt for a variety of reasons. Firstly, it is unknown if the spatial point that was identified truly is the global max - even within the grid itself. Because of the methodology for producing estimates of wind power, the grid had to be manually divided into sub-grids; thus it is impossible to determine if the ideal location might be slightly off from any of the smaller search grids.

Another point of concern on our results is the variability in the covariance matrices between the 'East' and 'West' points. This was worrisome because of the obvious difference between the in-land and out-land values which in turn makes us consider the validity of the our isotrophic assumption.

Future exploration of this data would include exploring this data with different methods of detrending on different granularities of time. In this report, the trend was captured over the different 6 hour intervals which might have different lasting effects if the trend was captured over daily, or monthly, or even yearly. Another point of future inquiry would be to remove trends with some automatic means like subtracting the smoothing spline result (as done in Lecture 6) instead of through aggregation.

# 5 Code Appendix

## Preliminary Work

```
# Load packages
pkgs <- c('xtable', 'plyr', 'sp', 'mapdata')
invisible(lapply(pkgs, library, character.only=T))

# Load data
ccw <- read.csv('./ccw.csv')

# Data reformatting
lon <- c(rep('41.0', 5), rep('41.5', 5), rep('42.0', 5))
lat <- rep(paste0(c('-69.', '-70.', '-70.', '-71.', '-71.'), c(5,0)), 3)
i_locs <- paste(lat, lon, sep=', ')
colnames(ccw) <- c('Time', i_locs)

# Reformat dataframe
ccw$Year <- as.numeric(format(as.Date(ccw$Time), '%Y'))
ccw$Month <- as.numeric(format(as.Date(ccw$Time), '%m'))
ccw$Day <- as.numeric(format(as.Date(ccw$Time), '%d'))
ccw$Time <- format(as.POSIXct(ccw$Time), '%H:%M:%S')
ccw <- ccw[,c(17:19, 1:16)]

# Cache chunk options
opts_chunk$set(echo=F, cache=T, autodep=T, cache.comments=F)
```

## Introduction

```
# Utilize code from Lecture 8 to calculate distances
coords <- data.frame(Longitude=unlist(strsplit(i_locs, ', '))[c(F, T)],
  Latitude=unlist(strsplit(i_locs, ', '))[c(T, F)],
  locs=i_locs)
coords$x = as.numeric(as.character(coords[["Latitude"]]))
coords$y = as.numeric(as.character(coords[["Longitude"]]))
coordinates(coords) = ~x + y

# Plot grid of different locations
par(mar=c(0, 0, 0, 0))
map("worldHires", xlim = c(-72, -69), ylim = c(40.75, 42.25))
plot(coords, add=T, pch=16, cex=0.85)
text(coordinates(coords), pos=1, label=coords$locs, cex=0.75)
legend('bottom', 'Given Locations', title='Point Description',
       pch=16, col=1, xpd=T, inset=c(0, -0.1), cex=0.75)
```

## Methods

### Exploratory Data Analysis

```r
# Checking and removing outliers
par(mfrow = c(2,1), oma = c(4.75,3,0,0) + 0.1, mar = c(0,0,0,0) + 0.1,
    mgp=c(1.35, 0.75, 0), cex.lab=0.75, cex.axis=0.75)

# Plot boxplot of distribution for each location
boxplot(ccw[,i_locs], xaxt='n', outcex=0.45)
i_max <- which.max(ccw[,5])

# Plot boxplot of distribution with outlier removed
ccw <- ccw[-i_max,]
boxplot(ccw[,i_locs], xaxt='n', outcex=0.45)

axis(1, at=1:15, labels=i_locs, las=2)
title(xlab='Spatial Coordinates', outer = TRUE, line = 3.75)
title(ylab='Wind Speeds (non-transformed & transformed)', outer = TRUE, line = 2.25)
```

```r
# Cube results
ccw <- cbind(ccw[,1:4], ccw[,i_locs]^3)

# Plotting observed wind speeds over index
par(mfrow=c(3,5), mar=c(2.5, 2.25, 1, 0.5) + 0.1, mgp=c(1.35, 0.5, 0),
    cex.lab=0.85, cex.axis=0.85, cex.main=1)
for (i_loc in i_locs){
  plot(1:nrow(ccw), ccw[,i_loc], xlab='Index', ylab='(Wind Speed)^3', type='l', lwd=0.075)
  title(i_loc, line=0.2)
  abline(h=mean(ccw[,i_loc]), col='black')
  lines(smooth.spline(x=1:nrow(ccw), y=ccw[,i_loc]), col='red', lwd=0.25)
}

# Calculating summary statistics
stats <- t(apply(ccw[,i_locs], 2, function(col) { return(c(summary(col), sd(col))) }))
colnames(stats)[7] <- 'Std. Dev'
print(xtable(stats, label='tab:stats',
             caption='Summary Statistics for Cubed Wind Speed'),
      table.placement='H')
```

### Checking Assumptions

```r
# Plotting autocovariance plots for wind subsections
par(mfrow=c(3,5), mar=c(2.5, 2.25, 1, 0.5) + 0.1, mgp=c(1.35, 0.5, 0),
    cex.lab=0.85, cex.axis=0.85, cex.main=1)
for (i in 5:19){
  acf(ccw[,i], lag.max=365*8, type='covariance', main='')
  title(names(ccw)[i], line=0.2)
}
```

```r
# Calculate means of all hours across all observations
ccw_time <- ddply(ccw, .variables=c('Month', 'Day', 'Time'),
                  .fun=function(df) { colMeans(df[,i_locs]) })

ccw_time_means <- colMeans(ccw_time[,i_locs])

ccw_adj <- ddply(ccw, .variables=c('Year'), .fun=function(df) {
  cbind(df[,2:4], df[,i_locs] - ccw_time[1:nrow(df), i_locs]) })
```

```r
# Plotting autocovariance plots for wind speeds
par(mfrow=c(3,5), mar=c(2.5, 2.25, 1, 0.5) + 0.1, mgp=c(1.35, 0.5, 0),
    cex.lab=0.85, cex.axis=0.85, cex.main=1)
for (i in 5:19){
  acf(ccw_adj[,i], lag.max=365*8, type='covariance', main='')
  title(names(ccw_adj)[i], line=0.2)
}

# Confirming stationarity
stats <- t(apply(ccw_adj[,i_locs], 2, function(col) { return(c(summary(col), sd(col))) }))
colnames(stats)[7] <- 'Std. Dev'
print(xtable(stats, label='tab:stats_2',
             caption='Summary Statistics for Seasonally Adjusted Cubed Wind Speed'),
      table.placement='H')
```

```r
# Confirming spatial stationarity and isotrophy
i_east <- i_locs[c(T, T, F, F, F)]
i_west <- i_locs[c(F, F, F, T, T)]
print(xtable(cov(ccw_adj[,i_east]), label='tab:covs_1',
             caption='Covariance Matrix for \'East Points\' for Seasonally Adjusted Cubed Wind'),
      table.placement='H')
print(xtable(cov(ccw_adj[,i_west]), label='tab:covs_2',
             caption='Covariance Matrix for \'West Points\' for Seasonally Adjusted Cubed Wind'),
      table.placement='H')
```

## Results

### Fitted Correlation Calculation

```r
# Calculate actual distances (in kilometers) between map points
dists <- spDists(coordinates(coords), longlat = TRUE)
dimnames(dists) <- list(i_locs, i_locs)

# Calculate correlation matrix of all coordinates
cor_mat <- cor(ccw_adj[,i_locs])
plot(as.vector(dists), as.vector(cor_mat),
     xlab='Distance (km)', ylab='Correlation Coefficient')
cor_50 <- mean(cor_mat[dists < 60 & dists > 40])
```

```
# Using non-linear least squares to determine optimal exponential decay model
L.rough <- -log(cor_50) / 50
L.final <- nls(as.vector(cor_mat) ~ exp(-lambda * as.vector(dists)),
               start = list(lambda = L.rough))
curve(exp(-x * coef(L.final)), add = TRUE)
legend("topright", legend=c('Original Data', 'Exponential Fit'),
       pch=c(1, NA), lty=c('blank', 'solid'), col=c('black', 'black'))

# Fitting an exponential curve to correlations
fitted_cors <- exp(-coef(L.final) * dists)
```

#### Kriging Given Locations

```
preds_est <- function(accum, loc) {
  i_loc = which(loc == i_locs)
  corYZ = fitted_cors[-i_loc, i_loc]
  corZ = fitted_cors[-i_loc, -i_loc]

  beta = solve(corZ) %*% corYZ
  pred = as.matrix(ccw_adj[,i_locs[-i_loc]]) %*% beta
  trend = ccw_time_means[i_loc]
  accum = c(accum, mean(pred + trend))
}

preds_err <- function(accum, loc) {
  i_loc = which(loc == i_locs)
  corYZ = fitted_cors[-i_loc, i_loc]
  corZ = fitted_cors[-i_loc, -i_loc]

  beta = solve(corZ) %*% corYZ
  trend = var(ccw_time_means)
  accum = c(accum, sqrt(trend - t(corYZ) %*% beta))
}

# Calculating estimates and standard errors
estimates <- Reduce(preds_est, i_locs, c())
std_errs <- Reduce(preds_err, i_locs, c())
results <- data.frame(estimates, std_errs)
dimnames(results) <- list(i_locs, c('Estimates', 'Std. Errs'))
print(xtable(results, label='tab:kriging_1',
             caption='Expected Wind Power Across all Initial Locations (with Standard Error)'),
      table.placement='H')
```

#### Kriging Hypothesized Locations

```
# Utilize code from Lecture 8 to calculate distances
coords_2 <- data.frame(Longitude=c(rep('41.5', 6), rep('41.6', 6), rep('41.7', 6),
              rep('41.8', 6),rep('41.9', 6), rep('42.0', 6)),
```

```r
  Latitude=rep(c(paste0('-69.', 5:9), '-70.0'), 6))
coords_2$locs <- paste(coords_2$Latitude, coords_2$Longitude, sep=', ')
coords_2$x = as.numeric(as.character(coords_2[["Latitude"]]))
coords_2$y = as.numeric(as.character(coords_2[["Longitude"]]))
coordinates(coords_2) = ~x + y

# Plot grid of different locations and new search grid
par(mar=c(0, 0, 0, 0))
map("worldHires", xlim = c(-72.5, -68.5), ylim = c(40.5, 42.5))
plot(coords_2, add=T, pch=16, cex=0.5, col='red')
plot(coords, add=T, pch=16, cex=0.85)
points(-69.5, 42.0, pch=8, cex=1.25, col='blue')
legend('bottom', title='Point Description',
       c('Location with Highest Potential', 'Given Locations',
         'New Grid for Searching'), pch=c(8, 16, 16), col=c(4, 1, 2),
       xpd=T, inset=c(0, -0.05), cex=0.75)
```

```r
# Calculate actual distances (in kilometers) between map points
dists_2 <- spDists(coordinates(coords_2), longlat = TRUE)
dimnames(dists_2) <- list(coords_2$locs, coords_2$locs)
fitted_cors <- exp(-coef(L.final) * dists_2)
best_locs <- c('-69.5, 42.0', '-70.0, 42.0', '-69.5, 41.5', '-70.0, 41.5')
i_best_locs <- which(best_locs %in% coords_2$locs)
best_means <- ccw_time_means[i_best_locs]

pred_new_est <- function(accum, loc) {
  i_loc = which(loc == colnames(fitted_cors))
  corYZ = fitted_cors[i_best_locs, i_loc]
  corZ = fitted_cors[i_best_locs, i_best_locs]
  beta = solve(corZ) %*% corYZ
  pred = as.matrix(ccw_adj[,i_locs[i_best_locs]]) %*% as.matrix(beta)
  trend = mean(best_means)
  accum <- c(accum, mean(pred + trend))
}

pred_new_err <- function(accum, loc) {
  i_loc = which(loc == colnames(fitted_cors))
  corYZ = fitted_cors[i_best_locs, i_loc]
  corZ = fitted_cors[i_best_locs, i_best_locs]
  beta = solve(corZ) %*% corYZ
  trend = var(best_means)
  accum <- c(accum, sqrt(trend - t(corYZ) %*% beta))
}

estimates <- Reduce(pred_new_est, coords_2$locs[-i_best_locs], c())
std_errs <- Reduce(pred_new_err, coords_2$locs[-i_best_locs], c())
results <- data.frame(estimates, std_errs)
dimnames(results) <- list(coords_2$locs[-i_best_locs],
                          c('Estimates', 'Std. Errs'))
print(xtable(results, label='tab:kriging_2',
```

```
            caption='Expected Wind Power Across New Search Space (with Standard Error)'),
      table.placement='H')
```