Application of PCA and KPCA to Ozone and CO2 Emissions Data

Callum Weinberg: PSTAT 262 FE Final Project

June 8th 2022

Abstract

One week of hourly, Los Angeles ozone concentration data and CO2 emissions data (by seven different emission production sectors) were processed for analysis. The CO2 emissions data was studied via exploratory data analysis (EDA), principal component analysis (PCA), and kernel principal component analysis (KPCR). PCA provided additional insight beyond EDA into the structure of the CO2 covariance and how CO2 emissions behaved throughout the day by sector. KPCA provided a method for analyzing the CO2 data via kernel methods. The principal components from PCA and KPCA were used in regressions, and indicated that ozone can be modeled well by kernel principal components when the periodic kernel is chosen. Kernel methods were studied in detail for the CO2 data.

Contents

Introduction

Accessing this Report

This report was completed for final exam credit for PSTAT 262 FE at University of California Santa Barbara in the spring quarter of 2022. The "CO2-Analysis-with-PCA-and-KPCA" project on my Github page "https://github.com/leoncw/" contains this report, a PDF with slides corresponding to the presentation of this report, and a R-Markdown file that contains the relative statistical programming for this report.[1] The Github project folder also contains the processed data used in the report and other files related to running the code and processing the data in the report.

Background

Greenhouse gas emissions from human activity are known to significantly contribute to global warming and climate change. Carbon dioxide emissions are the most prevalent greenhouse gas emissions in the United States.[8] Carbon dioxide is known to diffuse through the global atmosphere easily,[19][15, p. 2497] and therefore carbon dioxide levels in the atmosphere are generally similar throughout the

[1] Direct link: https://github.com/leoncw/CO2-Analysis-with-PCA-and-KPCA

globe.[17] However, local concentrations of carbon dioxide can be higher, especially close to emission sources.[17][15, p. 2497]

As of 2010, the impact of CO2 concentrations on local outcomes has not been studied extensively. In a 2010 paper, Mark Jacobson researches the effects of ”domes” of CO2 over cities. The results of the paper suggest that CO2 levels may increase ozone and particulate matter and may be associated with higher premature mortality.[15, p. 2497] Heightened ground-level ozone concentrations are associated with negative health outcomes.[7]. Therefore it is important to understand any relationship between local CO2 emissions and local ozone concentration.

Geospatial data and time series data are often high dimensional.[2] Therefore applying feature extraction methods is often advantageous for statistical learning tasks. Principal Component Analysis (PCA) is a fundamental feature extraction method. PCA is a method of performing a linear projection of a data set onto a subspace in which the variance of the data is maximized.[22, Part II Section 3.1] The resulting principal components have the useful properties that each is orthogonal to each other, and the maximum variance is captured by the first component and then each component has less of the variance. Therefore PCA is useful when variables (potential covariate in a model) are correlated with each other, as an orthogonal projection of the data can be extracted. Additionally, since principal components capture subsequently less variance of the data, PCA can be used as a dimensionality reduction technique, as a subset of the principal components may capture a large amount of the variation in the data.

PCA relies on linear basis projections of the original data.[22, Part II Section 4] An generalization of PCA is kernel principal component analysis (KPCA). The motivation for KPCA is that a linear basis may not be optimal for describing the data. KPCA allows for non-linear projections of the data by implementing a feature map. The data is mapped into a higher dimensional space that allows for non-linear representations of the projected data to be found. This may be useful for data with a clear covariance structure (e.g. time series data may be well represented by the periodic kernel - see the Methodology section).

Purpose

The purpose of this report is twofold. The first objective is to explore the relationship between ozone levels and carbon emissions at a localized level. The data in this report (see Data section) is limited to a time series for a single location (Los Angeles), for a number of covariates representing different sector-sources of carbon emissions. Although this objective was inspired by Jacobson’s 2010 paper,[15] this report does not attempt to replicate the analysis of the paper.

The second objective is to complete the first objective via PCA and kernel PCA methods. Specifically, PCA and KPCA are explored in detail through the CO2 data. Then Principal Component regression and kernel principal component regression models are fit to relate the CO2 data and the ozone data.

It need be stressed the results of this report should not in any way be construed as evidence for a causal relationship between localized carbon emissions and ozone levels. While such a relationship is a driving interest of this report, the scope of the report falls far short of the level of work that would be required to prove or disprove a causal relationship. Specifically, this work would likely need to be expanded to a geospatial-time model that controlled for confounding variables and the modeling would require an understanding of potential mechanisms that cause such a relationship between CO2 and Ozone.

Methodology

PCA

The key idea of PCA is to reduce the number of dimensions of a data set, and in doing so, create orthogonal principal components. PCA is accomplished by finding a linear subspace that maximizes

[2] high dimensional data refers to data sets in which the number of variables is large, especially if the number of variables is close to or greater than the number of observations.

the variance of the data, when the data is projected onto the subspace.[22, Part II Section 3.1] The following illustrates this concept:

$$X_{N \times p} W_{p \times p} = Z_{N \times d}$$

Where N is the number of observations in the data and p is the number of variables in the data. X is the original matrix of the data, W is the projection matrix, and Z is the transformed data. The columns of Z are referred to as the principal components. They are orthogonal to each other. The first principal component is effectively a new variable that explains the most variation in the data and is formed from a linear combination of the variables (as is seen in $X_{N \times p} W_{p \times p}$). The variance explained by the first principal component is then "subtracted out", and each additional principal component explains less of the data. In PCA, there can be up to p principal components. However for high dimensional data, it may only be necessary to keep a small subset of the principal components.

The columns of data matrix X should be centered before performing PCA, so that the projection captures the direction of the variance and not the mean.[10] Additionally, the columns of X should be scaled to a variance of one if the variances for each column (variable) are not roughly the same. Otherwise, the magnitude of the variance will be captured by earlier principal components.

PCA can be more generally formulated as special case of the eigenvalue problem:[22, Part IV Section 2]

$$C_X W_X = \lambda W_X$$

Where C_X is the covariance Matrix of X: $X'X$.

Methods For Performing PCA

One method to perform PCA is eigenvalue decomposition of C_X. The eigenvectors comprise W_X. The vectors should be sorted descending on the eigenvalues, as the vectors with largest corresponding eigenvalues have the maximum variance (the eigenvalues are the percent of the variance of each component).

Another method for performing PCA is Singular Value Decomposition (SVD). SVD of a matrix takes the following form:

$$X = U \Sigma V^T$$

Where X is the centered (and often scaled) data. Then $U\Sigma$ results in the principal components (and Σ are the singular values, which are the square root of the eigenvalues[22, Part II Section 3.7] . The projection matrix is given by V. SVD results in the columns of U being sorted descending on eigenvalues, providing a small benefit over eigenvalue decomposition.[5]

Both eigenvalue decomposition and SVD require the computation of all principal components. The NIPALS algorithm (Nonlinear Iterative Partial Least Squares) is a method for finding principal components that allows for only the first q components to be found. This can be useful if the number of variables p is larger, or if the Kernel Matrix K (see below) is large. One implementation of the NIPALS algorithm is as follows:

1. Center (and scale) $X_{N \times p}$
2. Initialize a vector $t_{N \times 1}$ to some random value (can use a column of X)
3. Calculate vector v (called a loading vector) $v = \frac{t^T X}{t^T t}$
4. Normalize v, such that $v = \frac{v}{\sqrt{v^T v}}$

5. Calculate t by regressing X onto p: $t_{new} = \frac{Xv}{v^T v}$
6. Check the sum of squared difference between t_{new} and t. If it is less than some threshold, say 10^{-9}, stop iteration of this loop.[5]
7. Set the pth vector of the principal component matrix equal to t and the pth vector of the projection matrix equal to v
8. Deflate the matrix X such that $X = t_{new} v^T$
9. iterate for i principal components, up to the number of columns of X

[5][22, Part II Section 3.8]

An advantage for the NIPALS algorithm over SVD and eigen decomposition is not all of the principal components have to be calculated: some subset $i < p$ could be calculated. As will be discussed in the analysis section, NIPALS is likely slower if all of the principal components are being calculated.

Note that the prcomp() function from the stats package in R uses SVD to perform PCA [24].

Kernel PCA

PCA involves creating principal components from a linear projection of the data. It may be that the principal components are better represented by a non-linear projection of the original data. The general idea of KPCA is to transform the data via a kernel (called the Gram matrix) into a higher dimensional space. In this higher dimensional space, a linear projection may be suitable and PCA can be performed. A non-linear combination of features is chosen (called a feature mapping), represented by $\phi()$. Then the Kernel matrix is defined as $K = \phi(X)\phi^T(X)$. $\phi(X)$ is often unknown (as it is in this report). However K can be calculated from variable X, which results in a matrix representing a set of points in the feature space. K is sufficient to perform PCA, and is similar to the Covariance Matrix C_x in linear PCA. Using K instead of $\phi(X)\phi^T(X)$ is called the "Kernel Trick."[22, Part II Section 4.1][21]

So KPCA can also be generally formulated as special case of the eigenvalue problem:[22, Part IV Section 2]

$$KW_X = \lambda W_X$$

.

Therefore, the KPCA principal components and projection vectors can be obtained using the same methodologies as PCA once the Kernel Matrix is determined. However if the feature mapping is unknown (as it is in this project), it is very challenging to find the point in the original space of X that the feature space maps to. This is called the "pre-image" problem,[22, Part II Section 4.5] and is outside the scope of this project (this report will only work with the principal components in the feature space).

Kernels

In order to apply the kernel trick, a kernel must be chosen. The kernel is applied to the feature space using some kernel function.[22, Part I Section 4] Many kernels are possible. PCA is a special case of KPCA when the linear kernel is chosen.

A very commonly implemented kernel is the radial basis function kernel (RBF), also called the isotropic stationary Kernel and the squared exponential Kernel. It takes the following form:[22, Part II Section 4.3]

$$k(x, x') = \sigma^2 exp\left(\frac{-(x - x')^2}{2l^2}\right)$$

σ^2 is often ignored when applying this kernel (and is ignored in the following two kernels). l^2 must be selected carefully, and its choice will affect the nature of the principal components.

Another kernel is the rational quadratic kernel (RQ), of the form: [22, Part II Section 4.3]

$$k(x,x') = \sigma^2 \left(1 + \frac{-(x-x')^2}{2\alpha l^2}\right)^{-\alpha}$$

.

This kernel has two hyperparameters (not including σ^2). α should be positive. It is often a good first choice for time series data.[3]. Both of these kernels are explored in the KPCA section of this report.

A third kernel considered is the periodic kernel (PER). This kernel is useful when modeling data that follows a cyclical pattern. It takes the following form:[6]

$$k(x,x') = \sigma^2 exp\left(\frac{-2sin^2(\pi|x-x'|/p)}{l^2}\right)$$

where l is similar in functionality to the RBF kernel. p is the hyperparameter that "determines the distance between repetitions of the function"[6].

Note that for all of these kernel definitions, scalar values that can be incorporated into a hyperparameter (i.e. 2 in RBF can be combined with l) are often ignored in computation. Similarly, l^2 is just specified as l in the remainder of this report and in the statistical code attached to this report.

Statistical Code and Software

All of the statistical programming in this report is completed in R. R version 4.2.0 was used at the time this report was written.[11] The R-Markdown file related to this report can be found on the author's Github page (see Introduction). The following packages were implemented in addition to the functions available in 'base' R: [23], [28], [27], [30], [29], [12], [25], [20], [3], [18], [16], [32], [33], [26].

Data

The primary data sets used in this report are 1) ozone data from the U.S. Environmental Protection Agency (EPA)[4] and 2) CO2 emissions data from the Vulcan High-Resolution Hourly Fossil Fuel data set repository provided by the Oak Ridge National Laboratory Distributed Active Archive Center (ORNL DAAC). [13] CO2 emissions are estimated for the following sectors: residential, commercial, industrial, electricity production, onroad, nonroad, commercial marine vessel, airport, rail, and cement.[13] Seven of these ten CO2 variables will be used as the covariates (X matrix of data) in this report.

Additionally, a county boundary shapefile from the California Open Data Portal[2] was used for the map in figure 1 below and is available in the "Raw" folder on the Github page.

Data Access

Ozone data is available from the EPA at the hourly timescale for sensors throughout the United States.[4] It is made publicly available as part of the EPA's air quality monitoring program. The unit of measurement is "parts per million." The specific file chosen for the report is "hourly_44201_2014", which is available on the EPA's cite as a Zip File. Only the zip file was loaded to the Github repository: to fully rerun the analysis, extract the file in the folder "Raw/EPA_Air_Data/". For analysis in this report, the following variables are relevant: State.Name, County.Name, Date.Local, Datum, Longitude, Latitude, Site.Num, Sample.Measurement, and Units.of.Measurement. Only "Sample.Measurement" and "Units.of.Measurement", i.e. the ozone concentration (in parts per million) and the date information (in Date.Local) is relevant to the modeling of the report. The other variables are used to limit

[3] Author lacks an explicit citation for this, but wrote it down in a conversation with the professor in class

the data to the desired scope.

CO2 emissions data is provided in metric tons for each kilometer by kilometer grid of the continental United states.[13] [4] The data is made available by date by emission sector on the daac.ornl.gov website. For example, hourly airport emission data for the continental U.S. for March 10th, 2014 can be found in the file "Vulcan.v3.US.hourly.1km.airport.mn.2014.d069.nc4" The data is stored in the .nc4 file format. Users are required to create an account to download the data, but the data is free and "openly share, without restriction."[13] The CO2 data was too large and did not compress well to load to the the Github page for this project (see introduction). Therefore the file "CO2_Long_v1.Rdata" (available in the "Intermediate" folder on the Github page) can be accessed as a starting point for rerunning the analyses in this report or extending them. Alternatively, a user can download a list of the files (see "Raw/Vulcan_CO2/March2014/List_of_Files.txt") and download them and place them in the "Raw/Vulcan_CO2/March2014/" folder to fully rerun the analysis.

Data Scope

The raw data files for both the ozone data and the CO2 data are large. Therefore a limited timescale was chosen to analyze the data. Specifically, the data is limited to 4PM on March 9th, 2014 until 3 PM on March 17th, 2014. [5] This results in slightly more than a week of data. The Vulcan data is available from 2010 until 2015. March 9th was chosen as a starting date since it is after daylight savings time of that year (2 AM of that data), in order to avoid any data-processing issues.[6]

Both data sets are limited to the Los Angeles area. For the ozone data, fourteen sensor sites measured ozone data during this time frame. Sites 9033 and 6012 were excluded from this analysis as they are in Lancaster, CA and Santa Clarita, CA. See Figure 1 below. Both sensors are North of mountain ranges and not near the center of the county. The data is available in the WG S84 reference coordinate system[31], with latitude and longitude of sensor-sites provided.

[4] Note that Alaska data is available as well, but is in separate files.

[5] The data is provided in UTC, and this date-range is the data after converting to Pacific Timezone.

[6] 2014 was chosen as it was towards the end of of the time frame that the CO2 data is available. The month of March was chosen somewhat arbitrarily, although the author enjoyed choosing a week that contained March 16th as on that day in 2014, the University of Virginia beat Duke University in the ACC basketball championship 72 to 63 [9].

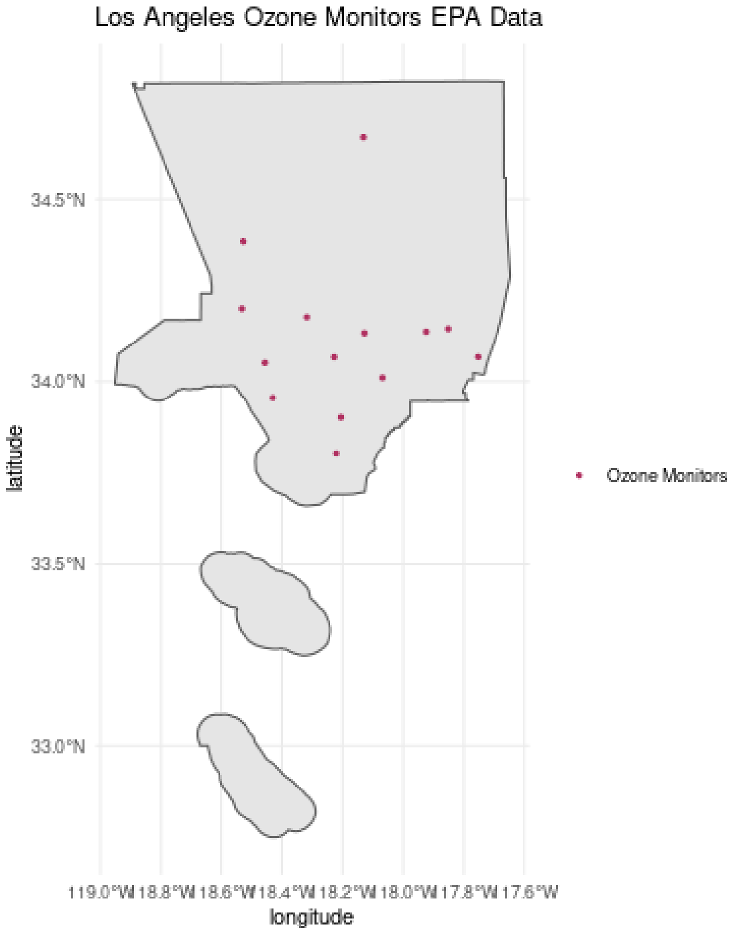
Los Angeles Ozone Monitors EPA Data
34.5°N
34.0°N
33.5°N
33.0°N
latitude
119.0°W 18.8°W 18.6°W 18.4°W 18.2°W 18.0°W 17.8°W 17.6°W
longitude
● Ozone Monitors

Figure 1: Ozone Monitors from EPA data, reporting ozone concentrations between PM on March 9th, 2014 until 3 PM on March 17th, 2014. L.A. County Site Numbers 9033 and 6012 are the two most northern points on the map.

The CO2 data is also limited to the Los Angeles area. The bounds are set to be the maximum and minimum latitude and longitudes of the ozone data, since the ozone data is more restricted. The Vulcan CO2 .nc4 data sets provide the data referenced in meters instead of longitude and latitude, and in the Lambert Conformal Conic reference system. See the R-markdown file (see Introduction) for a description of how the measurements were converted to align the geospatial references of both data sets.

Data Processing

The ozone and CO2 raw data sets were processed in the following steps to get data sets that are used for Explanatory Data Analysis (EDA), PCA, and KPCA. For the CO2 data, the .nc4 files are read into R using the ncdf4 package.[23] .nc4 file types are a bit challenging to work with in R: the author found the following resource very helpful for extracting information from .nc4 files for storage in data frames.[1] Specifically, the latitude and longitude equivalents (in meters, for Lambert Conformal Conic reference system) were extracted from the "y" and "x" objects for the nc4 R-object (as created by the ncd4 package). The CO2 data is extracted using the ncvar_get() function for the "carbon_emissions" variable. This is done for all of the .nc4 files for all ten of the CO2 emission types. The data is limited to: $y >= -401193.688560163, y <= -353976.762435499$, $x >= -1942544.51850291, x <= -1880037.22186271$. These bounds were found to be the equivalent for the maximum longitude and latitude bounds for the ozone sensors. The date-data was provided in "hours since 2010-01-01." This was converted to a date-time variable.

The ozone data was limited to Los Angeles county, California from the original data file (and WGS 84 reference system observations).

For both ozone and CO2 data (for each emissions source), the average across sensors for a given hour was calculated for use in the EDA and Analysis sections. For ozone concentration, this is likely the appropriate metric given the data is being aggregated over a region. For the CO2 data this decision is more circumspect. Arguably, total CO2 emissions by sector for a given hour may be more interesting for studying the relationship between ozone and localized CO2 emissions. However, the complicated nature and large size of the .nc4 data made studying the sensor sites a challenging problem. An average over the region was taken primarily because the author could not determine how often sensors may be missing data for the time frame. Finally the data sets were merged on the date and hour.

For the CO2 sector variables, cement and commercial marine vessel were not found to have any emission data for the time frame. Rail emissions were reported as constant for the time frame. As the rail emissions had no variability, they would not be useful in PCA, KPCA, or a regression problem, and are excluded from the analysis.

See the R Markdown file (see Introduction) for a complete description of the data processing.

Exploratory Data Analysis

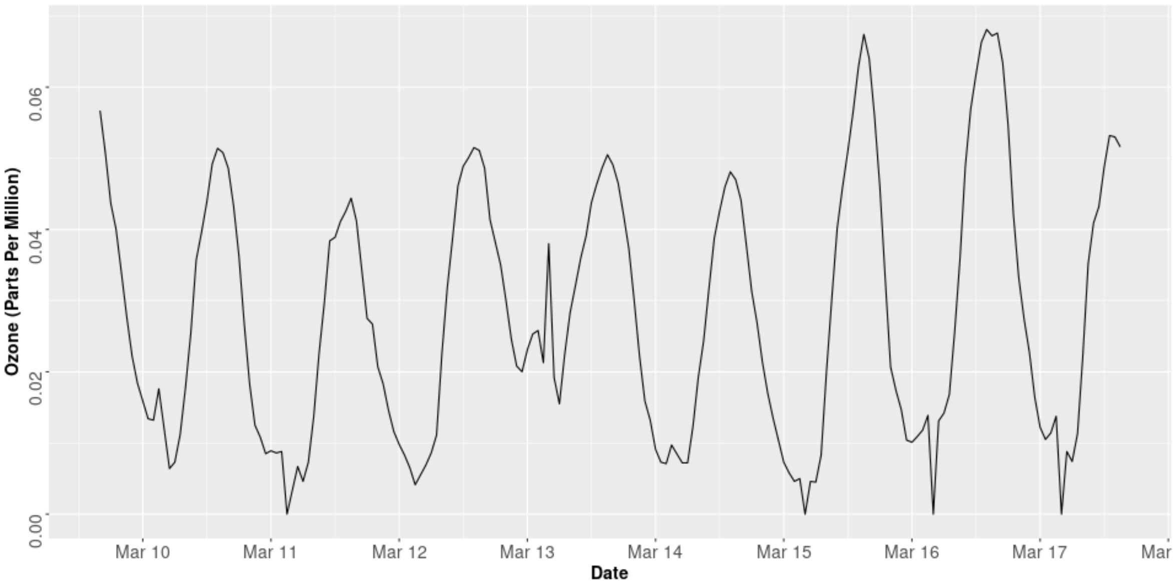

Figure 2: Average Ozone Concentration (Parts Per Million) Across Los Angeles County Sensors. Hourly from 4 PM May 9th, 2014 until 3 PM May 16th, 2014 (PST). Ozone sensor data excludes L.A. County Site Numbers 9033 and 6012 for the time period.

Figure 2 above shows the hourly time series for the average ozone concentration data for the L.A. County area (excluding the two sites, as explained in the above section). There is some noise to the data, but the ozone data is fairly cyclical. It peaks around .05 to .07 parts per million around 1 or 2 PM local time, and reaches the minimum of less than .01 parts per million around 1 or 2 AM most nights. May 13th is an exception, with ozone concentration reaching minimum of just below .02 parts per million.

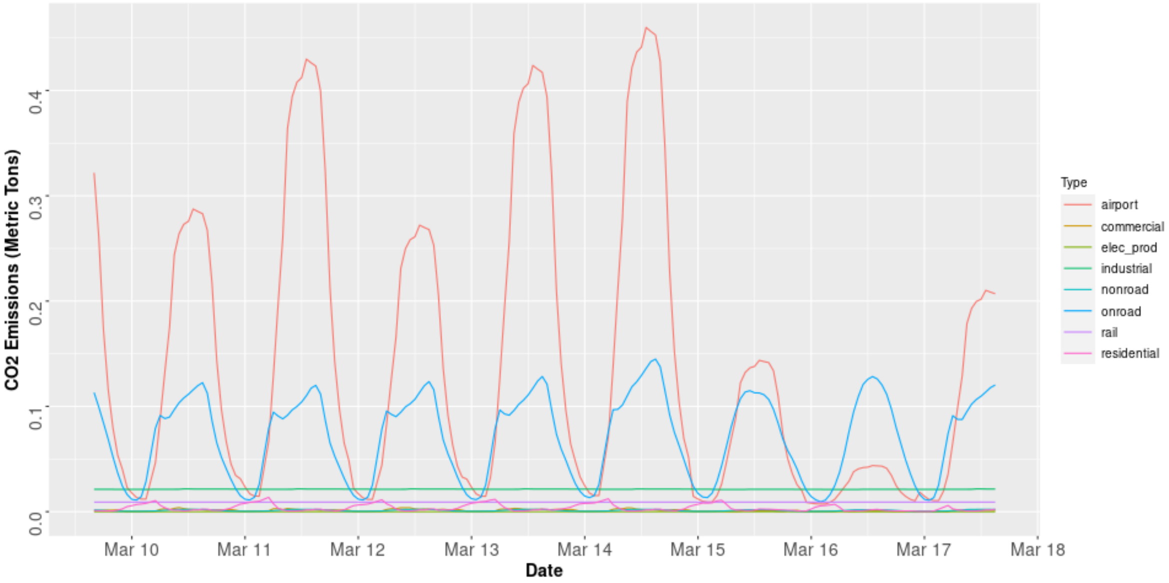

Figure 3: Average Metric Tons of CO2 Across Los Angeles County Sensors. Hourly from 4 PM May 9th, 2014 until 3 PM May 16th, 2014 (PST). Sensors Limited to bounds of Ozone Sensors, see Data Processing Section.

Figure 3 shows the hourly time series for average CO2 Emissions (Metric Tons). Airport emissions are by far the largest source of CO2 in the L.A. area for these dates. Peak airport emissions vary a lot of the course of the day and throughout the week, with lower peak emissions on May 15 and 16th (Saturday and Sunday). Onroad emissions are the next highest source of emissions, with peaks in the middle of the day and lows around midnight. Other emission sources are significantly lower, and cannot be discerned in the same graph. Note that rail emissions are estimated by Vulcan to be constant (but non-missing) over time.

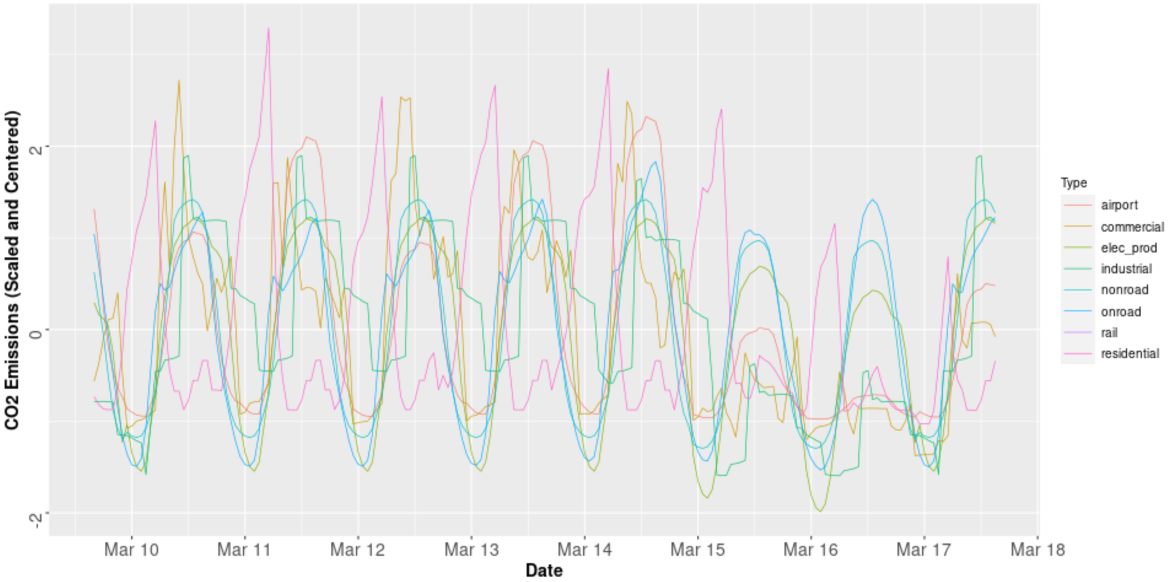

Figure 4: Average CO2 Scaled and Centered Across Los Angeles County Sensors. Hourly from 4 PM May 9th, 2014 until 3 PM May 16th, 2014 (PST). Rail emissions are not shown in this graph. Sensors Limited to bounds of Ozone Sensors, see Data Processing Section.

Figure 4 shows the sources of emissions scaled to a variance of 1 and centered to a mean of 0. This allows for visualizing the time series patterns of the sources with lower average emissions. Most emission source peak around midday and reach their minimum around midnight. The exception is residential emissions, which peak late evening to morning. Residential emissions also decrease over the weekend, which is interesting and perhaps unexpected. If individuals are commuting less, it may have been hypothesized that individuals were spending more time (and creating more emissions) at their residence.

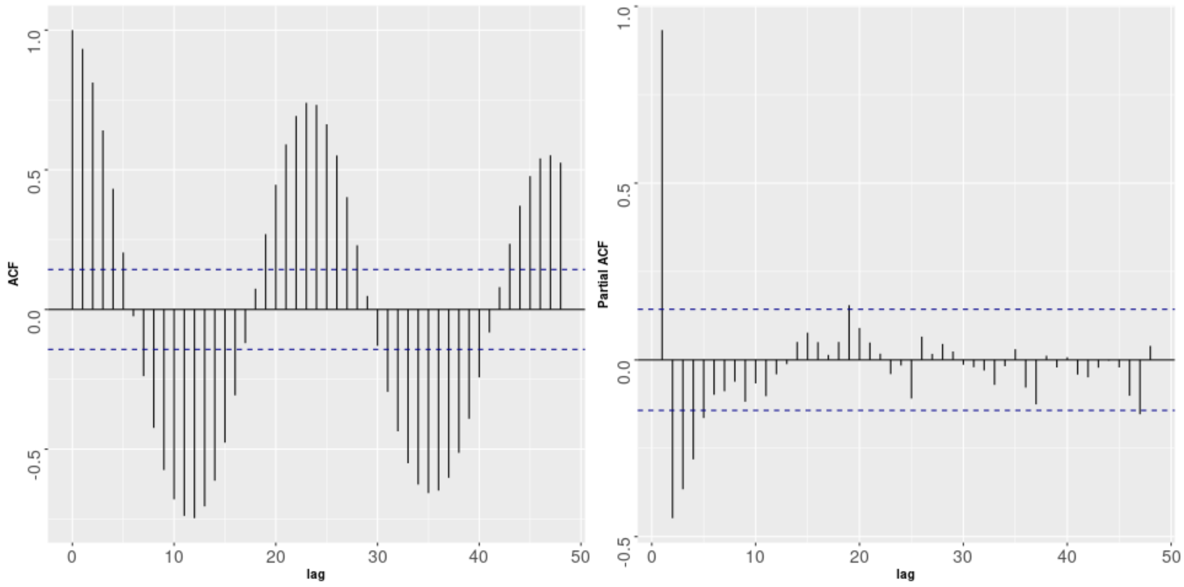

Figure 5: Autocorrelation function and partial autocorrelation function for ozone data over the time period, up to a lag of 48 hours. The blue lines indicate 95% confidence intervals for the ACF values.

The autocorrelation function (ACF) and partial autocorrelation function (PACF) for the ozone data is shown in Figure 5. In time series analysis, the ACF measures how correlated a observation at time t is to an observation at $t + lag$, for some $lag > 0$. The PACF is the correlation between the two lags while controlling for all other lags. These two graphs indicate that there is significant autocorrelation over time between the ozone concentrations. The ACF makes clear the cyclical nature of the ozone data, with observations near in hours positively correlated, and observations 12 hours apart negatively correlated. The PACF is used to inspect for some sort of "seasonal" trend, i.e. that observations a day apart are significantly correlated, after controlling for the other lags. The PACF indicates that the correlation is mainly happening at a lag of one hour, two hours or three hours. There does not appear to be daily season trend day-to-day (i.e. ozone concentration is not that correlated with ozone concentration 254 hours later). Note that this data set only spans one week. It is possible there would be weekly (and possibly annual) seasonal trends if more data were included in the report.

Figures 6 and 7 below show similar ACFs and PACFs for Airport CO2 and Residential CO2 emissions. However, the 24 lag (seen in the PACF) is significant in both cases. This suggests there is a predictable day-to-day relationship for CO2 emissions.

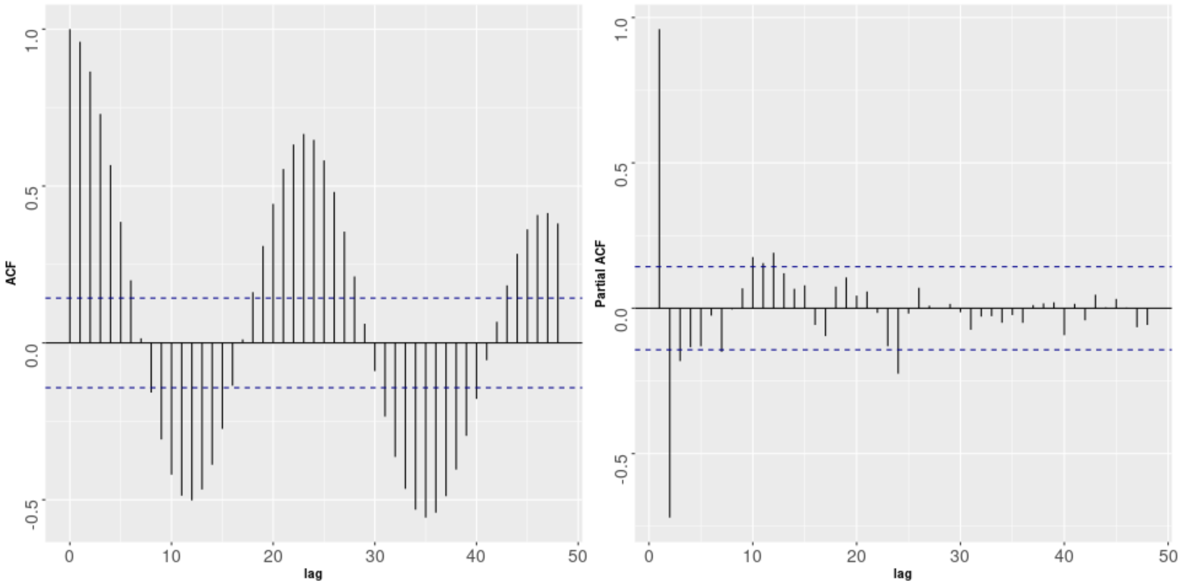

Figure 6: Autocorrelation function and partial autocorrelation function for Airport CO2 emissions data over the time period, up to a lag of 48 hours.

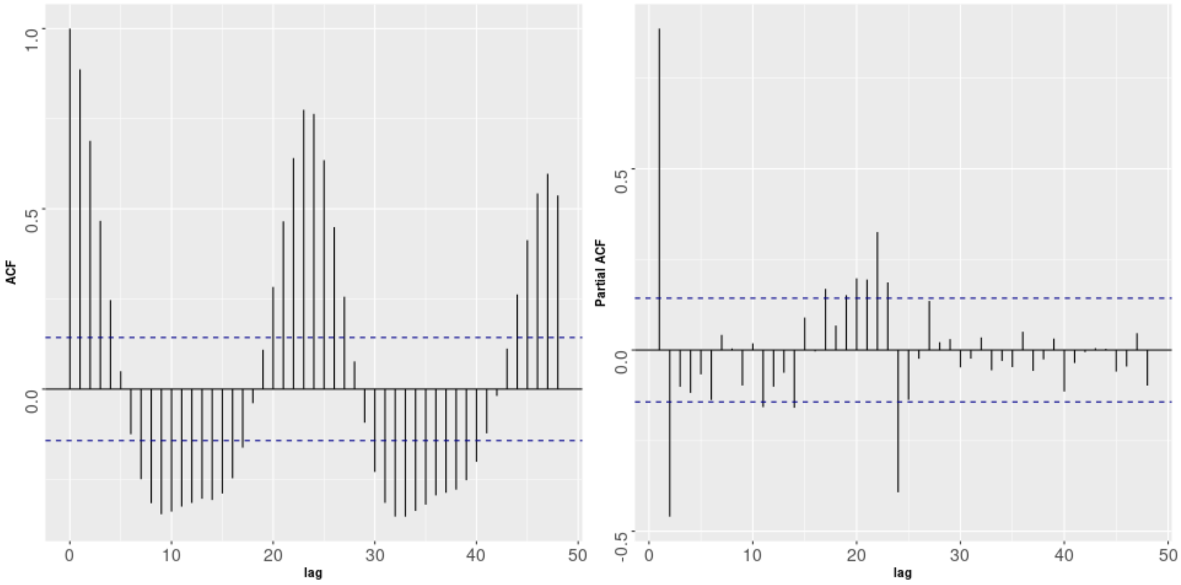

Figure 7: Autocorrelation function and partial autocorrelation function for Residential CO2 emissions data over the time period, up to a lag of 48 hours.

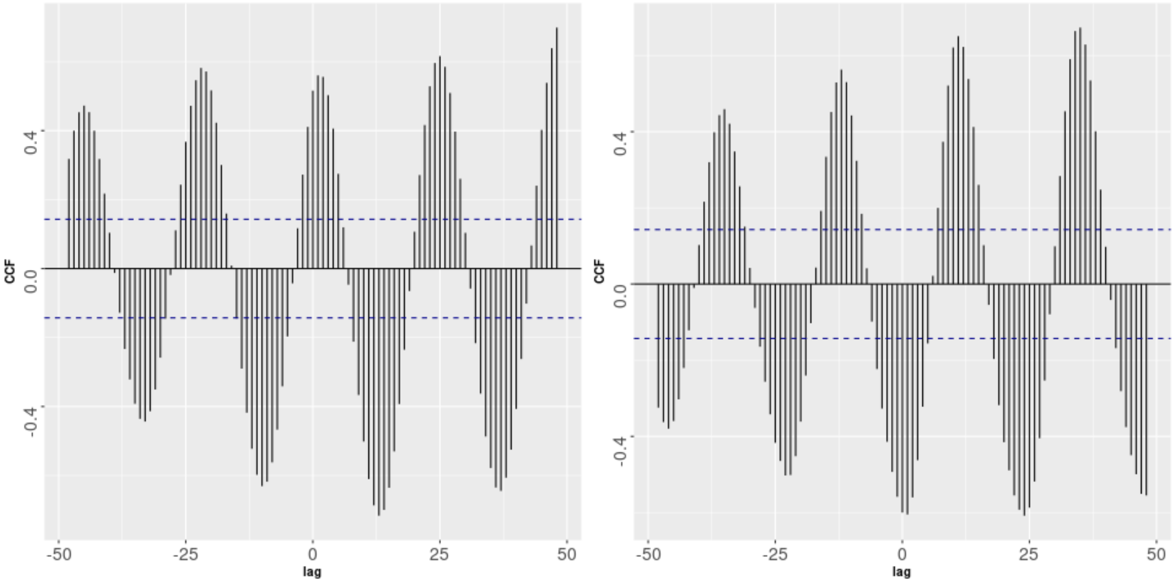

Figure 8: Cross-Correlation function functions Ozone vs. Airport CO2 (left) and Ozone vs. Residential CO2 (Right) over the time period, up to a lag of 48 hours.

Finally, figure 8 shows the cross-correlation functions between Ozone and Airport CO2 and Ozone and Residential CO2. Both show significant patterns of correlation between ozone and each variable.

All eight variables considered for the analysis portion of this report exhibit wave-like time series over the period. There is clear correlation between the variables, likely being driven by hour-of-the-day phenomena (e.g. there are not a lot of flights at the airport at midnight, and residential activities take place in the mornings and evenings while individuals are home). This correlation structure suggests a kernel method may be useful in capturing the relationships between these variables.

Analysis and Results

For the analysis section, X refers to the centered and scaled CO2 data, which is limited to the seven sectors: airport, commercial, electricity production, industrial, non-road, on-road, and residential. It contains 188 observations. Y refers to the Ozone data, which is one variable and 188 observations.

PCA Computation

As mentioned in the methodology section, a number of algorithms can be used to run PCA. Figure 9 illustrates the computational time taken to perform PCA using five different approaches.

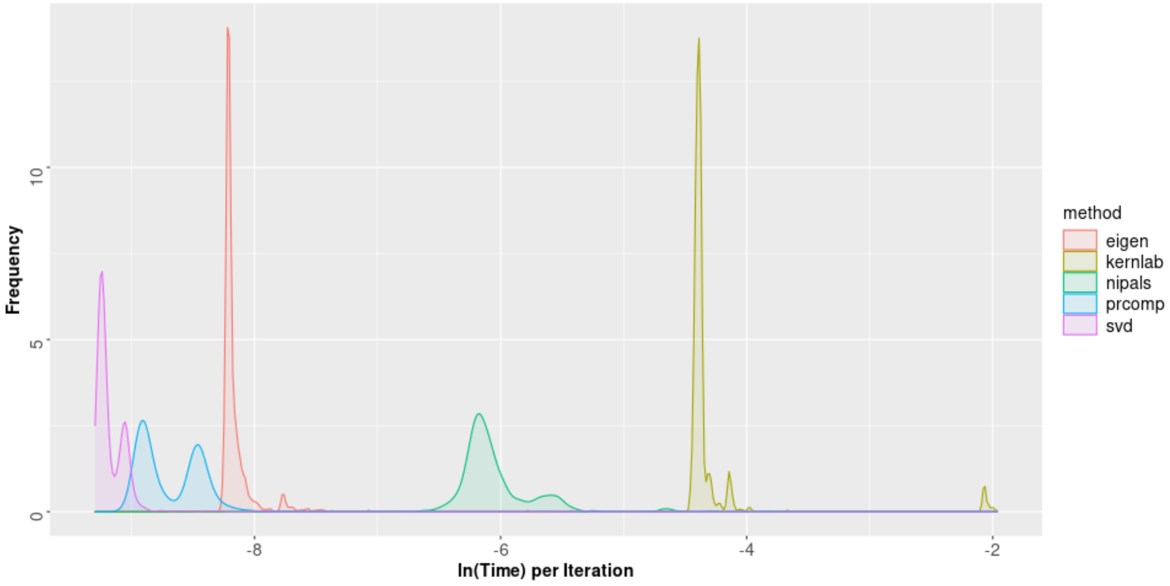

Figure 9: Log of time reported for visual clarity. 1000 iterations of performing PCA for five different methods. The data $X_{188\times7}$ is the matrix of centered and scaled CO2 data.

The natural log of time is shown to reduce the impact of outliers on the graph. The log of times farther to the left correspond to shorter times to perform PCA. 1000 iterations were performed as there is some randomness to how long any given machine will perform this task, even while using the same data X_c.

Performing PCA via R's svd() function (part of the 'base' R functions) performs the best, while the kernlab package[16] (using the "vanilladot" kernel, kernlab's term for the linear kernel). NIPALS performs better than kernlab. Both have higher variance than the other three methods.

It is interesting that SVD performs slightly better than the prcomp function, since prcomp reportedly uses svd[24]. It is possible that prcomp involves some extra computation that is not necessary to just extract the principal components. SVD will be used for PCA and KPCA as it performed most efficiently for X.

PCA Analysis

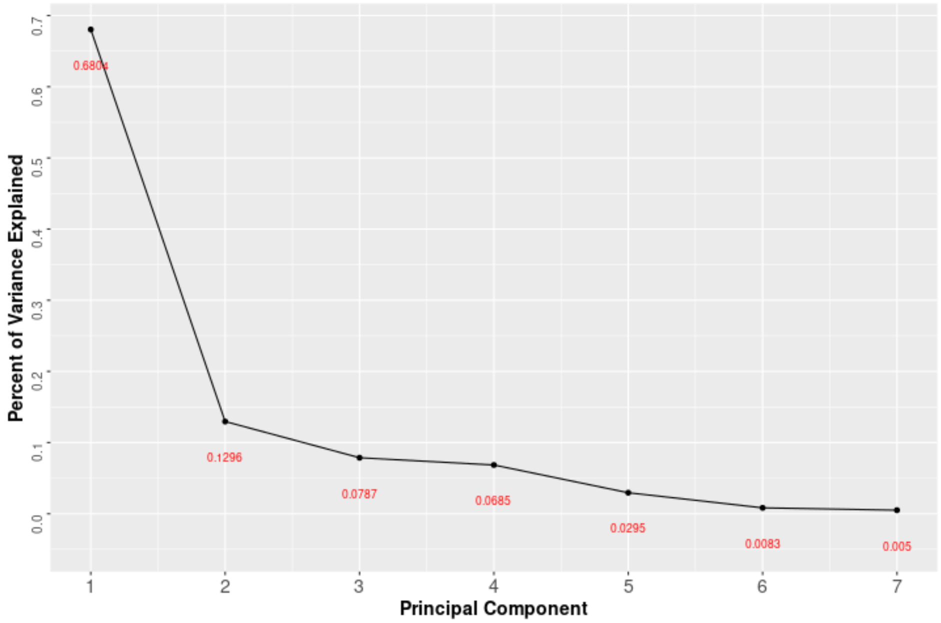

Figure 10: The percent of variation explained by each principal component for the CO2 data.

Figure 10 shows the scree plot for the PCA of X_c. The percent variance explained for each principal component can be found in a number of ways. For SVD, the eigenvalues are equal to the squared singular values (diagonal of matrix Σ), and the normalized eigenvalues (in descending order, as is constructed in SVD) represent the percentage of variation explained by each principal component. Here, 68% of the variation in the data can be explained by the principal component. The second principal component explains the next 13% of the variation. So with just 2 of 7 principal components, 81% of the total variance can be explained.

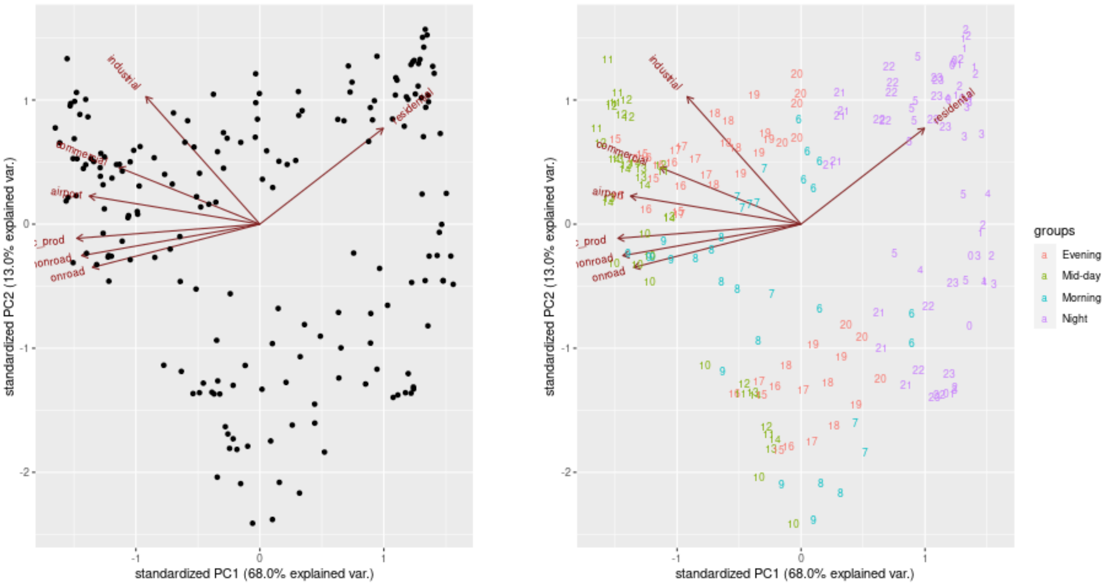

Figure 11: First and Second Principal Components Compared.

Above are the bi-plots of the first and second principal components.[7] The loading vectors (columns of the projection matrix) are shown by the arrows. The points represent the values of the first and second principal components for each observation. The bi-plot on the right groups the data based on time of day. Figure 12 shows normal ellipses drawn around the groupings. The following grouping was assigned by inspection:

- Hours 21, 22, 23, 0, 1, 2, 3, 4, 5 grouped as ”Night”
- Hours 15, 16, 17, 18, 19, 20 grouped as ”Evening”
- Hours 6, 7, 8, 9 grouped as ”Morning”
- Hours 10, 11, 12, 13, 14 grouped as ”Mid-day”

All of the seven CO2 variables are contributing significantly to the first component. Residential and industrial contribute the most to the second component. There is a clear distinction between the group defined as ”night” and the rest of the day. This also corresponds to the residential loading vector, indicating the variation in residential CO2 emissions is distinct from the other emissions sources. This is expected given the breakdown shown in Figure 4 of the time series by group. Careful inspection of industrial emissions in figure 4 suggests a somewhat irregular pattern, possibly explaining its high contribution to components one and two. Although less distinct than night-emissions, it does seem that mid-day emission levels correspond to the non-residential emission sources.

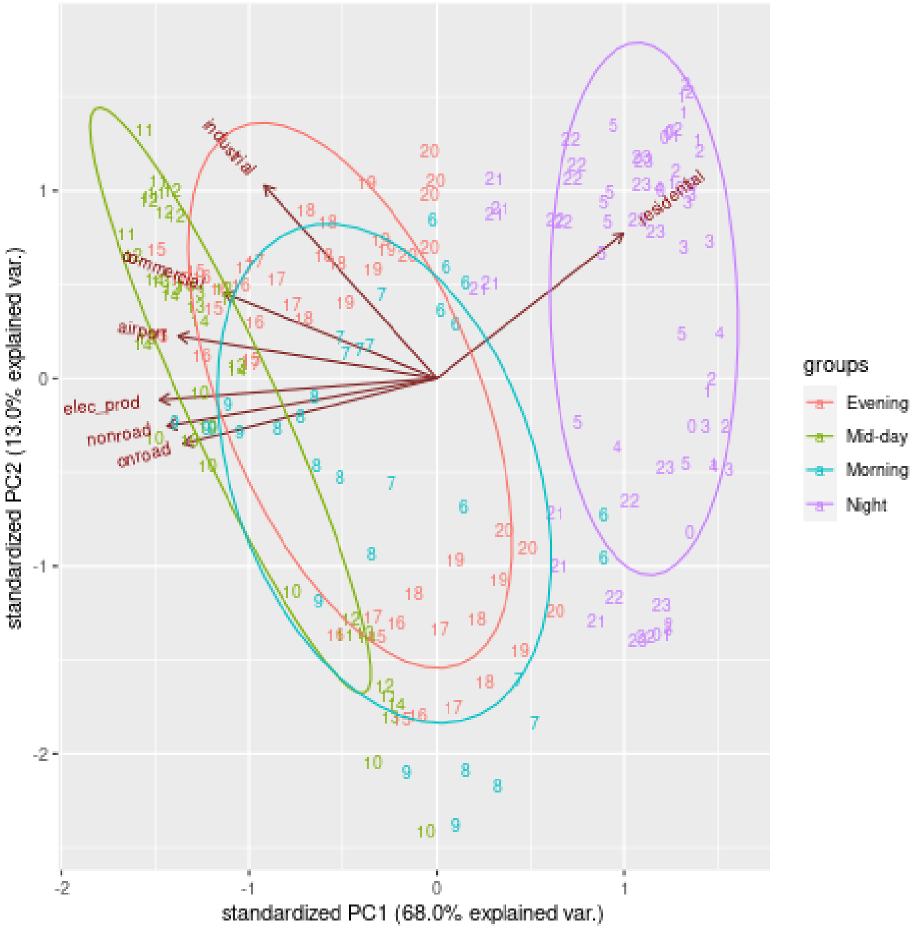

Figure 12: First and Second Principal Components Compared, with normal ellipse for groups.

[7] The following resource proved useful in implementing the ggbiplot[26] package. [14]

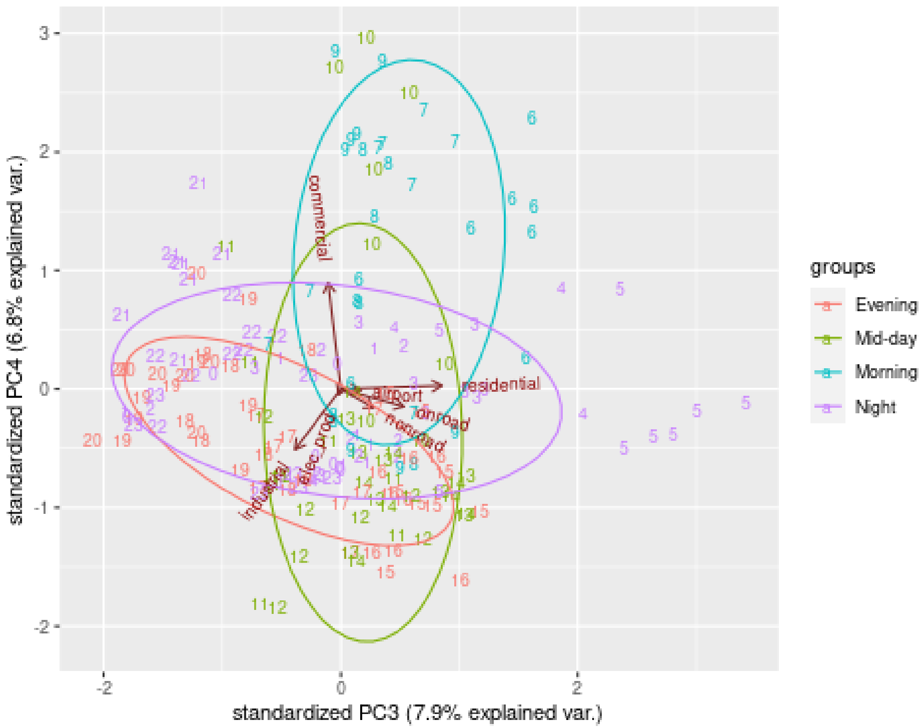

Figure 13: Third and fourth Principal Components Compared, with normal ellipse for groups.

Figure 13 shows the third and fourth principal components. The groupings are less distinct here. Residential CO2 emission variance appears completely deflated by principal component four, whereas commercial and industrial are still contributing (commercial just to principal component four).

In summary, these graphs suggest only one or two principal components are necessary to explain significant amounts of variation in the data. The first principal component is well defined by the time of day, whereas the second, third, and fourth are less so. Less than seven principal components may be sufficient to connect the CO2 emissions data to the ozone data, as is explored in the next section.

Principal Component Regression

There are many reasons why principal component regression would be useful for modeling ozone and CO2 data in this problem. One motivation is that the CO2 variables are highly collinear. This results in an unstable model. For example, the variance inflation factor (VIF) for three of the seven CO2 variables when a model is fit is above 10 (a VIF of 5 is conventionally considered large and indicates multicollineary problems for a model). The principal components will all have a VIF of one since each vector is orthogonal to each other by design. See the R Markdown file for an illustration of this phenomena.

Instead, principal component regression is performed (PCR). For modeling ozone as a function of CO2 the main question is how many principal components to include. A model fit with all principal components results in high significance for the first, second, and fourth principal components and an adjusted R^2 of .1753. It seems unnecessary to include all seven principal components, as the previous section illustrated that most of the variation was captured by the first few principal components.

Instead, if four components are selected, the significance of one, two and four, and the value of their coefficients barely changes. The adjusted R^2 increases to .181. So the ozone data can be equally well modeled by the first four principal components of the CO2 data.

The final model is shown below:

PC	Coefficient	Standard Error	P-Value
PC1	-0.0055698	0.0010155	$1.35x10^-7$
PC2	-0.0057490	0.0023270	0.01440
PC3	0.0001153	0.0029865	0.96924
PC4	-0.0097947	0.0032011	0.00255

Table 1: Principal Component Regression between Ozone and the Principal Components of CO2 Data. 184 df, Adjusted $R^2 = .181$

KPCA

In this section a number of kernel choices are explored in detail via computational methods. First, some checks are performed to ascertain the RBF kernel is correctly calculated by the author by comparing it to the results of the kernlab package[16]. Then the choice of hyperparameters and the subsequent effect on percentage of variation explained for each principal component are studied. Finally, a kernel principal component regression (KPCR) model is fit using one of the kernel choices.

RBF Kernel, $l = 2$

Packages such as kernlab[16] can perform KPCA for a number of kernels. However there are many kernels not defined by such a package: in this project, neither the RQ nor PER kernels are defined by the kernlab package. Therefore it is necessary to define these kernels in the statistical code associated with this report.

To get some assurance this definition is done properly, the author checks that the RBF kernel they have defined matches the RBF kernel as produced by the kernlab package (specifically using the kernelMatrix() function and the "rbfdot" choice of kernel). Note that the kernlab package describes the parameter "sigma inverse kernel width for the Radial Basis kernel function"[16, p. 32]. Sigma in the RBF package appears to be $\frac{1}{l}$ as defined in this report in the methods section. Therefore $l = 2$ and $\sigma = \frac{1}{2}$ are chosen to determine if the same kernel (Gram matrix) is produced by the author's code and by kernlab.

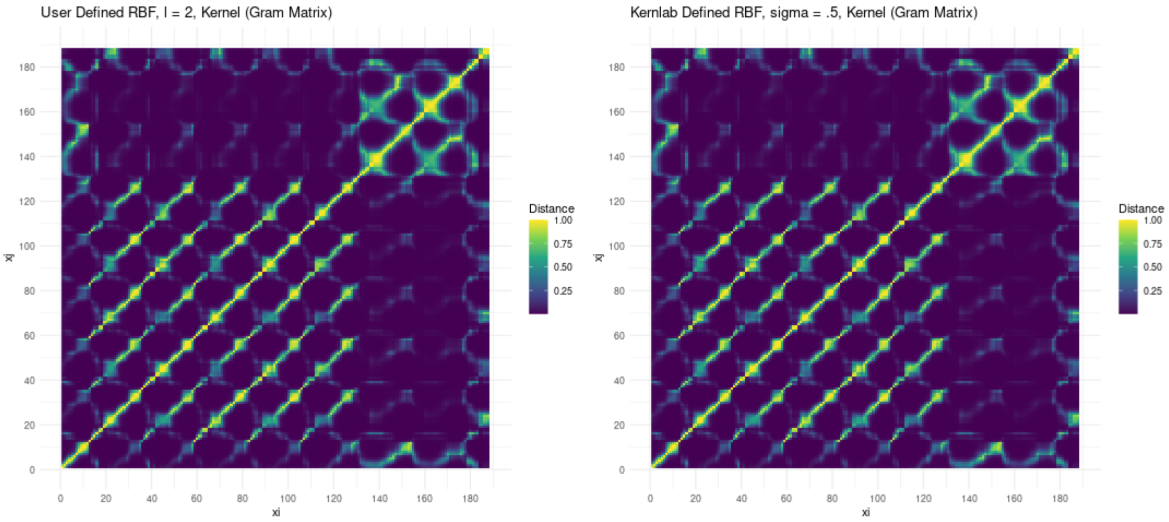

Figure 14: Comparison of the kernel matrix for the user defined function for RBF ($l = 2$) and kernlab defined ”rbfdot” kernel matrix ($\sigma = .5$).

Note that the above matrices appear identical. A quick check (see the statistical code associated with this report) reveals these matrices are the same down to computational differences at 10^{-15} decimal places.

Inspection of the kernel matrix also makes clear the periodic nature of the underlying data, with spikes in closeness (distance = 1 indicates two observations have the same distance from each other, i.e. the same CO2 values since $x - x' = 0$ and $e^0 = 1$) approximately 24 observations apart. This suggests the periodic kernel may work well in representing the data, for a choice of $p = 24$.

Kernel Hyperparameter Selection

Radial Basis Function

The choice of hyperparameters for these kernels will significantly affect the structure of the kernel matrix, and in turn will affect KPCA. A way to study this is to vary the hyperparameters, and see how the eigenvalues of the kernel matrix are affected. As with C_X in PCA, in KPCA the eigenvalues (when sorted descending) correspond the percent of variation explained by each principal component. A scree plot can be used to illustrate this.

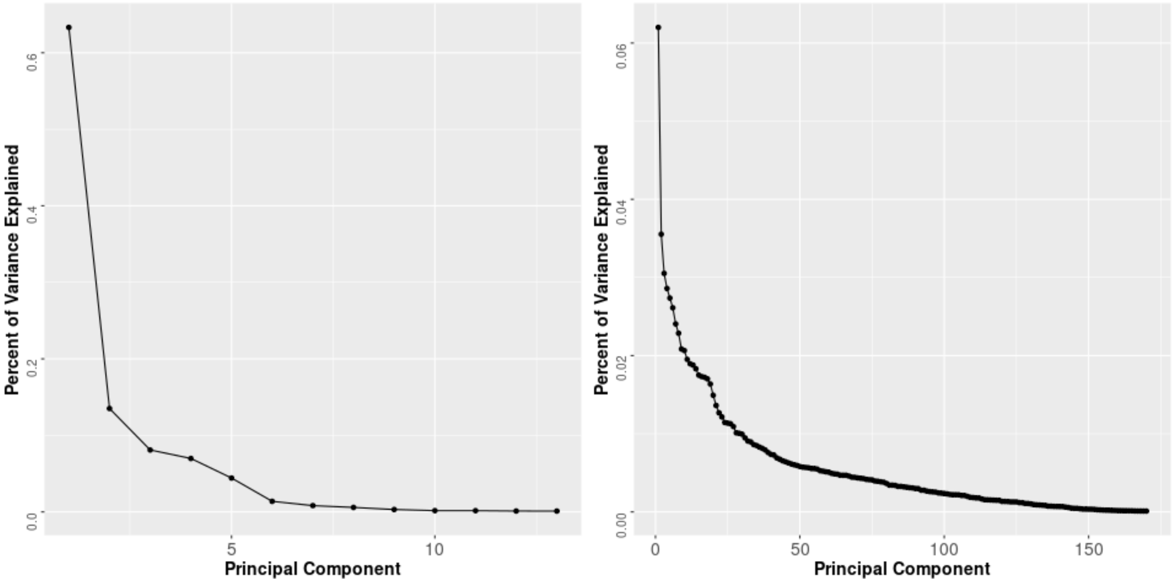

Figure 15: $l = .01$ (left) and $l = 2$ (right).

Figure 15 compares the percent of variation explained when l = .01 versus when l = 2 for the RBF kernel. When $l = .01$, it only takes a few principal components to explain most of the variation in the data. Whereas with l = 2, a significant portion of the total explained variation in the data is being captured by principal components above 15, and maybe even 30.

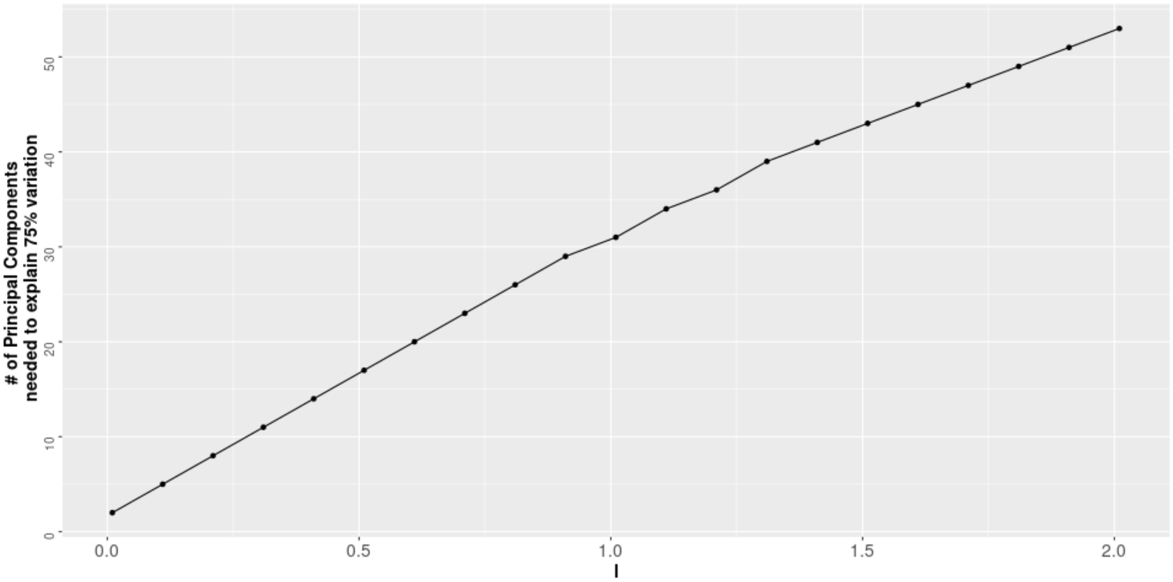

Figure 16: Number of principal components required to explain 75% of the variation in the data, for some l.

Figure 16 illustrates how many principal components are required to explained 75% of total variation for each choice of l, with l varied from .01 to 2.1 by increments of .1. As can be seen, there is a clear linear relationship in the single parameter RBF kernel. Increasing the size of l linearly increases the number of principal components needed, linearly. [8]. The optimal choice of l will depend on the use of

[8] Note that l is the simplified parameter for l^2, so depending on the way the parameter is specified the relationship may look different

KPCA. If the goal is dimensionality reduction, a small value of l is likely optimal.

Rational Quadratic

The RBF kernel only has one parameter to vary. The RQ kernel has two, which makes hyperparameter selection more challenging.

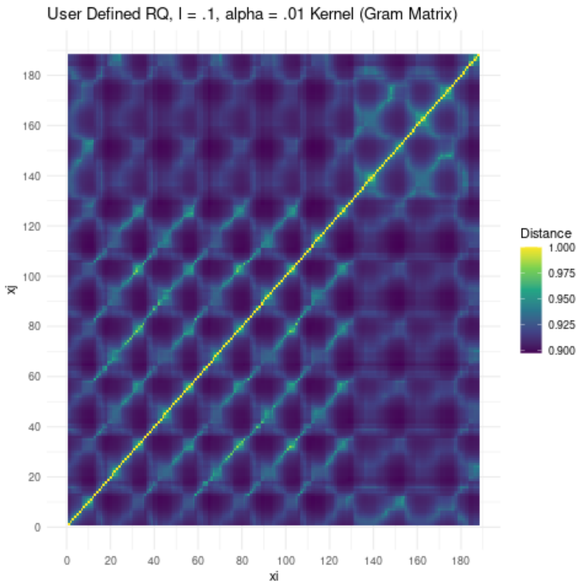

Figure 17: l and α are the hyperparameters for RQ

Figure 17 shows the kernel matrix for X when $l = .1$ and $\alpha = .01$. For such a selection, we see that we get similar structure of the kernel matrix to the RBF kernel (see figure 14, l = 2). However the similarity between observations 24 hours apart is much lower. In both Figure 17 and Figure 14, there is an interesting "blur" to lags in the range of six to seven days. One possible explanation is the kernel is capturing similarity for the week days, but as shown in Figures 3 and 4, the weekend emissions are not as consistent with the weekdays as the weekdays are with each other.

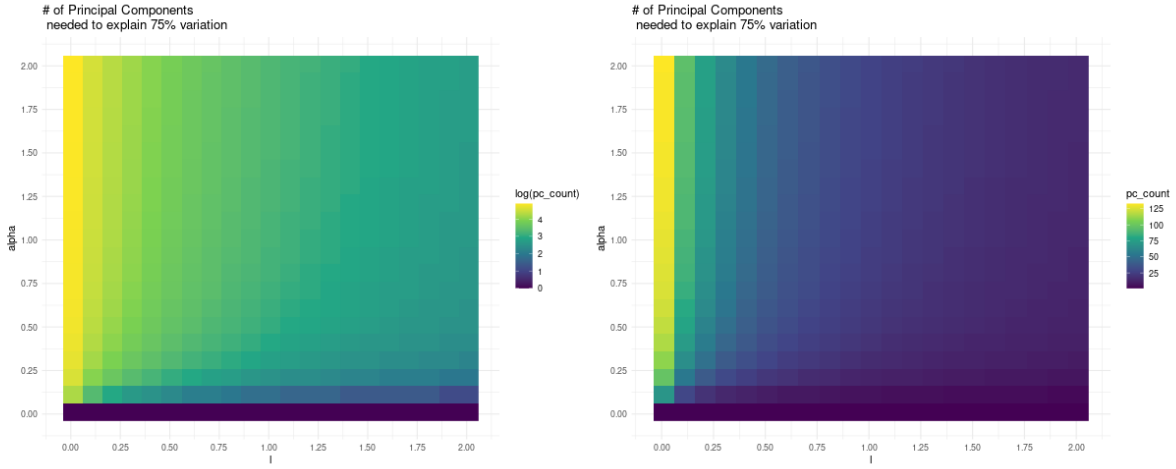

Figure 18: Number of principal components required to explain 75% of the variation in the data, for some l and α. The left-hand figure is the natural log of "number of principal components", to better express the difference visually.

Figure 18 serves the same purpose as Figure 16, but in two dimensions. Warmer colors (more yellow) indicate more principal components are required, whereas deeper purples indicate fewer principal components are required. The left-hand figure in Figure 18 is the natural log of the count of principal components (in order to help with visual separation of the values).

Values of α close to zero always results in a small number of principal components required to meet the 75% of variation explained threshold (call this value k for brevity). Large values of l also result in small values of k. But for most values of α in the range, a small value of l results in a large k. The author limited l and α from .01 to 2.01. Other ranges of the values were considered but did not provide more distinct separation of the hyperparameters and k.

Again, an optimal threshold will depend on the task. But in order to get some sort of kernel structure that appears to capture the periodic nature of the data, a very small l is not a good choice for most α. Note that unlike in the RBF case, as l gets bigger k gets smaller, for fixed α.

Periodic Kernel

The periodic kernel is useful for representing a functional space with cyclical structure. Figure 19 shows the kernel matrix for the periodic kernel, for two choices of p.

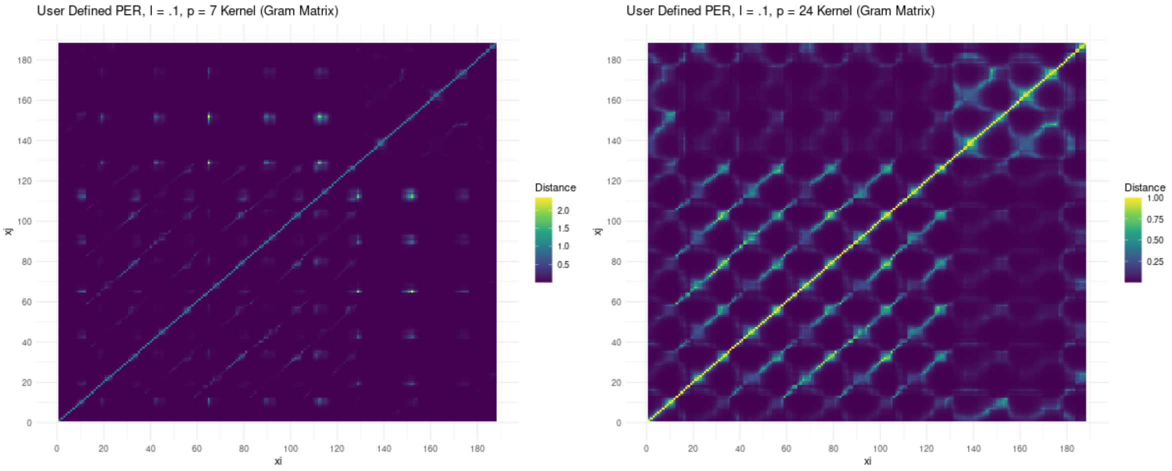

Figure 19: Periodic Kernel. l = .01. p = 7 (left), p = 24 (right)

The choice of p makes an obvious difference for this kernel. For $p = 24$, a kernel matrix somewhat similar to figure 17 is recaptured. For $p = 7$, the kernel matrix captures much less of the similarity in the data. This makes sense. The obvious period in this data is 24 hours. There is no evidence of any sort of period on a seven hour cycle. So a well structured Gram matrix is achievable with the periodic kernel, but may require that we know something about the periodicity of the data.

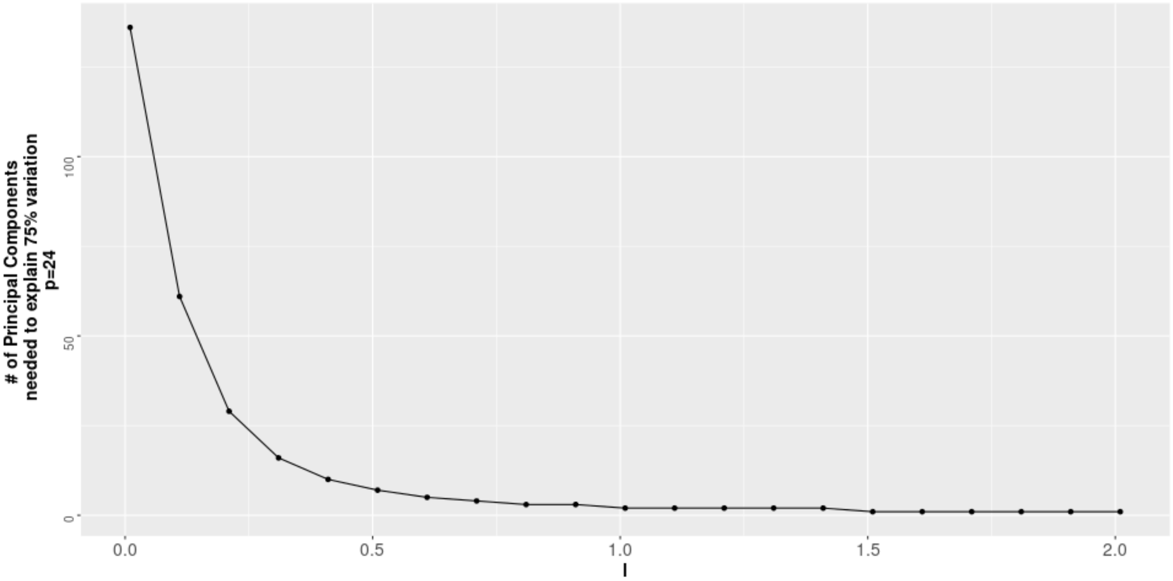

Figure 20: Number of principal components required to explain 75% of the variation in the data for the periodic kernel, for varying l and $p = 24$.

Figure 20 indicates that for the periodic kernel (for fixed $p = 24$), the opposite is true of the RBF kernel. As l is increased, the number of principal components required to explain 75% of the data decreases. When l is .01, over half of the principal components are required to do so. When l is 1.11, only 2 principal components are required to do so.

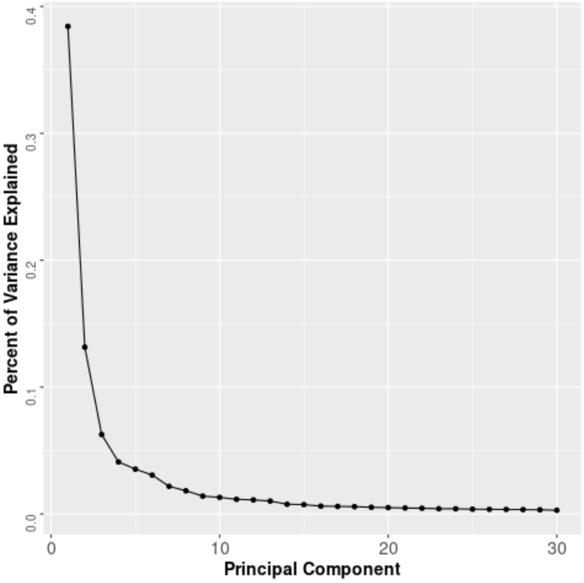

Figure 21: Scree plot for periodic kernel, $l = .4$ and $p = 24$.

In figure 21, we see that for choice of $l = .4$ (and continuing with $p = 24$, then the first principal component will explain about 37.5% of the variation, the second will explain about 12.5% of the variation, and principal components three through six add non-negligible amounts of explanation.

KPCR

In order to perform KPCR, two steps should be performed. A kernel should be selected, and then the hyperparameters need to be selected. The periodic kernel is selected as it seems appropriate for the data based on the exploratory data analysis. Additionally, it is clear a similar kernel matrix can be obtained by both the periodic kernel and the RBF kernel. Optimal hyperparameter selection is beyond the scope of this project.

KPCR is performed using the periodic kernel and a choice of $l = .4$ and $p = 24$

PC	Coefficient	Standard Error	P-Value
PC1	-0.005420	0.0001303	~ 0
PC2	-0.0067609	0.0003808	~ 0
PC3	-0.0031337	0.0007995	0.000126
PC4	0.0111344	0.0012223	~ 0
PC5	-0.0011377	0.0014171	0.423120
PC6	0.0119072	0.0016338	9.26×10^{-9}

Table 2: Kernel Principal Component Regression between Ozone and the Principal Components of CO2 Data using the Periodic Kernel, $l = .4$ and $p = 24$. 182 df, $R^2 = .9235$, Adjusted $R^2 = .921$

The coefficients and standard errors of the above model are not interpretable in this form, as they are representative of the functional space (as mentioned earlier in the report, this project does not deal with the "pre-image problem"). However it is worth noting we are able to capture a very large percentage of the overall variation in ozone concentration with this approach ($R^2 = .9235$). There is clearly over-fitting for this approach with such a large R^2. Still, the results of this selected kernel suggest that much more predictive models may be achievable with KPCR versus PCR if an appropriate kernel and feasible hyperparameters are selected.

Conclusion

Summary of Results

The results of this report indicate that there is clearly a relationship between ozone concentration and CO2 emissions in Los Angeles in the given week, and that relationship can be discovered in a number of ways. In the exploratory data analysis section, the time series plots, ACFs and PACFs, and cross-correlation functions all show a relationship between ozone concentration and CO2 emissions. PCA indicated there that the different CO2 emission sources can be transformed into a set of principal components, that still explain time of day and emission source well. KPCA resulted in the CO2 emissions for the week being modeled by a kernel that represents a higher dimensional projection of the data. Hyperparameter choice clearly effects the kernel structure, and suitable (though likely not optimal) hyperparameters were chosen for KPCR. PCR and KPCR resulted in linear models of a few principal components that can explain the variation in ozone concentration well. In particular, KPCR for the principal components from a periodic kernel can explain the variation in ozone well with just a a few principal components.

Limitations and Future Work

The data for this project was limited to just over one week of hourly data. The CO2 emissions data is very computationally intensive to process and difficult to store on a local machine. Processing more CO2 data would allow for modeling multi-week periods (which likely will have a more complicated covariance structure) and also will allow for validation and test sets to be constructed and implemented for PCR and KPCR.

The decision to take the average of CO2 emissions (by sector) for sensors in Los Angeles was done for practical purposes. The intensive data processing made it difficult to determine which sensors were reporting values for each hour. Taking the average was a more conservative approach that the total, as values may have been missing for certain sensors for certain hours. However, total CO2 emissions is likely more important in the relationship with ozone (especially if some causal relationship exists). Future work can be done to check if there is missing sensor data for the Vulcan CO2 emissions data.

This project did not consider optimal choice of hyperparameters for PCR and KPCR. One way to perform this would be to separate the data into training, validation, and test sets. The models would be fit on the training data for various combinations of hyperparameters, and the validation set would be used to get mean-squared error (or some other measure of fit). Then the final results would involve predicting the test data, and getting the error between the prediction and the actual test values.

Acknowledgements

Thank you to Professor Gareth Peters who taught the class on this material, held multiple meetings advising me on the project, and suggested the Vulcan CO2 Emissions data set for use in this report. Thank you to Laurel Abowd who helped me with the geospatial processing of the Ozone and CO2 data. In particular, she advised me how to process the .nc4 formatted data files and how work with the Lambert Conformal Conic projection in R.

References

[1] Pat Bartlein. *R for Earth-System Science: netCDF in R*. Available at: https://pjbartlein.github.io/REarthSysSci/netCDF.html.

[2] CA.gov. *CA County Boundaries*, 9 2019. Available at: https://data.ca.gov/dataset/ca-geographic-boundaries/resource/b0007416-a325-4777-9295-368ea6b710e6.

[3] Gabor Csardi. *rematch: Match Regular Expressions with a Nicer 'API'*, 2016. R package version 1.0.1.

[4] EPA Air Data. *Air Data: Pre-Generated Data Files*, 11 2021. Available at: https://aqs.epa.gov/aqsweb/airdata/download_files.html#Raw. File in Question referred to as "hourly_44201_2014.zip".

[5] Kevin Dunn. *6.5.14. Algorithms to calculate (build) PCA models*, 5 2022. Available at: https://learnche.org/pid/latent-variable-modelling/principal-component-analysis/algorithms-to-calculate-build-pca-models.

[6] David Duvenaud. *The Kernel Cookbook: Advice on Covariance functions.* Available at: https://www.cs.toronto.edu/ duvenaud/cookbook/. Also note that the author states this is part of their thesis: however the periodic kernel definition does not seem to appear in this form in their thesis.

[7] EPA. Health effects of ozone pollution. Technical report, U.S. Environmental Protection Agency, 2021. Available at: https://www.epa.gov/ground-level-ozone-pollution/health-effects-ozone-pollution.

[8] EPA. Climate change indicators: Greenhouse gases. Technical report, U.S. Environmental Protection Agency, 2022. Available at https://www.epa.gov/climate-indicators/greenhouse-gases.

[9] ESN. *ACC MENŚ TOURNAMENT FINAL 2014*, 3 014. Available at: https://www.espn.com/mens-college-basketball/game/_/gameId/400546817.

[10] Sydney Firmin. *CA County Boundaries*, 7 2019. Available at: https://towardsdatascience.com/tidying-up-with-pca-an-introduction-to-principal-components-analysis-f876599af383.

[11] The R Foundation. *The R Project for Statistical Computing*, 2022. Version 4.2.0: Vigorous Calisthenics.

[12] Garrett Grolemund and Hadley Wickham. Dates and times made easy with lubridate. *Journal of Statistical Software*, 40(3):1–25, 2011.

[13] K.R. Gurney, J. Liang, R. Patarasuk, Y. Song, J. Huang, and G. Roest. *Vulcan: High-Resolution Hourly Fossil Fuel CO2 Emissions in USA, 2010-2015, Version 3*, 10 2020. Available at: https://daac.ornl.gov/cgi-bin/dsviewer.pl?ds_id=1810.

[14] Luke Hayden. *Principal Component Analysis in R Tutorial*, 8 2018. Available at: https://www.datacamp.com/tutorial/pca-analysis-r.

[15] Mark Jacobson. Enhancement of local air pollution by urban co 2 domes. *Environmental Science & Technology*, 44(7):2497–2502, 3 2010. Available at https://web.stanford.edu/group/efmh/jacobson/Articles/V/es903018m.pdf.

[16] Alexandros Karatzoglou, Alex Smola, Kurt Hornik, and Achim Zeileis. kernlab – an S4 package for kernel methods in R. *Journal of Statistical Software*, 11(9):1–20, 2004.

[17] Pola Lem. Reader question: Does co2 disperse evenly around the earth? Technical report, NASA, 12 2016. Available at https://earthobservatory.nasa.gov/blogs/earthmatters/2016/12/05/reader-question-does-co2-disperse-evenly-around-the-earth/.

[18] Kristian Hovde Liland, Bjørn-Helge Mevik, and Ron Wehrens. *pls: Partial Least Squares and Principal Component Regression*, 2021. R package version 2.8-0.

[19] Charlotte Munson. You asked: How does carbon dioxide get so high up into the atmosphere? Technical report, Columbia Climate School, 9 2020. Available at https://news.climate.columbia.edu/2020/09/23/carbon-dioxide-distribution-atmosphere/.

[20] Edzer Pebesma. Simple Features for R: Standardized Support for Spatial Vector Data. *The R Journal*, 10(1):439–446, 2018.

[21] Daniel Pelliccia. *PCA and kernel PCA explained*, 6 2020. Available at: https://nirpyresearch.com/pca-kernel-pca-explained/.

[22] Gareth Peters. *PSTAT262FE Kernel Machines and Feature Extraction Methods*, 4 2022. Specific version provided as part of PSTAT 262 FE course. A 2017 version of these notes can be accessed here https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3050592.

[23] David Pierce. *ncdf4: Interface to Unidata netCDF (Version 4 or Earlier) Format Data Files*, 2021. R package version 1.19.

[24] R Core Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2022.

[25] Martijn Tennekes. tmap: Thematic maps in R. *Journal of Statistical Software*, 84(6):1–39, 2018.

[26] Vincent Q. Vu. *ggbiplot: A ggplot2 based biplot*, 2011. R package version 0.55.

[27] Hadley Wickham. *ggplot2: Elegant Graphics for Data Analysis.* Springer-Verlag New York, 2016.

[28] Hadley Wickham. *stringr: Simple, Consistent Wrappers for Common String Operations*, 2019. R package version 1.4.0.

[29] Hadley Wickham. *tidyr: Tidy Messy Data*, 2021. R package version 1.1.3.

[30] Hadley Wickham, Romain François, Lionel Henry, and Kirill Müller. *dplyr: A Grammar of Data Manipulation*, 2021. R package version 1.0.6.

[31] wiki.gis.com. *WGS84*. Description available at: https://wiki.gis.com/wiki/index.php/WGS84.

[32] Claus O. Wilke. *cowplot: Streamlined Plot Theme and Plot Annotations for 'ggplot2'*, 2020. R package version 1.1.1.

[33] Yihui Xie. knitr: A comprehensive tool for reproducible research in R. In Victoria Stodden, Friedrich Leisch, and Roger D. Peng, editors, *Implementing Reproducible Computational Research*. Chapman and Hall/CRC, 2014. ISBN 978-1466561595.