Channel Islands
CALIFORNIA STATE UNIVERSITY

Application of Principal Component Analysis

Mathematics

Names. Isaak Reyes, Norberto Huerta, Blake Meschke, Corey Tabor   Advisor: Dr. Ivona Grzegorczyk

ABSTRACT/PURPOSE

When working with a regression problem an issue that often arises is having an abundance of data that hinders our model. A dataset may include a large set of variables, but some of the variables do not contain much of the necessary information. One of the solutions to this problem is implementing an algorithm called Principal Component Analysis, this algorithm reduces the dimensions of our data set. Principal Component Analysis helps us in visualizing the main component in our data set and improves our modeling results. We are using data from a Kaggle competition that includes a training and testing data set, with 24 variables included in each data set. Our goal is to take an extensive look at the process of Principal Component Analysis and how it performs. We will do this by comparing the modelling results when using Principal Component Analysis and when not using Principal Component Analysis on the Kaggle dataset.

RESULTS

Multilinear Linear Regression

- **We wanted to create a Multiple Linear Regression Model to later compare the results to our Principal Component Analysis Model.**
- **The image on the right shows our linear model using our 24 variables. This model was obtained using the testing dataset.**
- **We also calculated the R^2,Root Mean Square Error, and mean absolute error using our testing dataset.**
- **The higher the R2 value means, that our model accounts for a higher percentage of our response variable variance.**
- **The lower the RMSE and MAE means that the model was performing better.**

```
Residuals:
    Min      1Q  Median      3Q     Max
-8.1514 -1.8066 -0.3543  1.0614  9.4182

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)
(Intercept)           -2.184022   1.348481  -1.620  0.10709
minutes_past          -0.005712   0.012995  -0.440  0.66080
radardist_km           0.234396   0.077924   3.008  0.00301 **
Ref                    0.095947   0.084211   1.139  0.25609
Ref_5x5_10th           0.032102   0.065657   0.489  0.62549
Ref_5x5_50th          -0.334997   0.141548  -2.367  0.01903 *
Ref_5x5_90th           0.212841   0.104995   2.027  0.04415 *
RefComposite          -0.150030   0.111229  -1.349  0.17911
RefComposite_5x5_10th -0.306883   0.094275  -3.255  0.00136 **
RefComposite_5x5_50th  0.428391   0.214755   1.995  0.04760 *
RefComposite_5x5_90th  0.106871   0.130885   0.817  0.41530
RhoHV                 -0.846304   1.485706  -0.570  0.56965
RhoHV_5x5_10th        -1.069291   1.339369  -0.798  0.42573
RhoHV_5x5_50th         1.865122   1.689013   1.104  0.27098
RhoHV_5x5_90th        -0.528266   1.034144  -0.511  0.61011
Zdr                   -0.123846   0.310989  -0.398  0.69094
Zdr_5x5_10th           0.159655   0.502030   0.318  0.75085
Zdr_5x5_50th           0.276974   0.743188   0.373  0.70983
Zdr_5x5_90th          -0.515309   0.281191  -1.833  0.06854 .
Kdp                    0.022515   0.068759   0.327  0.74372
Kdp_5x5_10th           0.125414   0.112032   1.119  0.26446
Kdp_5x5_50th           0.110888   0.122942   0.902  0.36831
Kdp_5x5_90th          -0.122505   0.105149  -1.165  0.24556
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.936 on 177 degrees of freedom
Multiple R-squared:  0.3791,    Adjusted R-squared:  0.302
F-statistic: 4.913 on 22 and 177 DF,  p-value: 5.281e-10
```

R2 <dbl>	RMSE <dbl>	MAE <dbl>
0.2574564	2.356512	1.923406

CONCLUSIONS/IMPLICATIONS

- Why did our PCA model result in a lower R2 and a higher RMSE and a higher MAE? Does this necessarily mean our PCA model was worse?

We were expecting the PCA model to perform slightly worse than the multilinear regression model because, models using PCA sacrifice some accuracy for a simpler data set. The simpler data set is easier to work with and ensures the data that is less significant to not be included, when creating the model.

- Why would you choose to implement PCA?

In today's world, there are huge datasets. Principal Component Analysis is able to simplify and compress those datasets into something simpler which still captures most of the information from the original dataset. This allows for faster computations within and upon those datasets.

METHODOLOGY

- Problem Approach
  - Our main goal was to reduce the amount of dimensions of our dataset using Principal Component Analysis. This allows for us to retain most of the datasets variance and information while making a large dataset smaller and therefore easier to work with. We then see how the Principal Component Analysis model holds up to a Multiple Linear Regression Model.
- Data
  - First we used a dataset that included 24 variables in order to predict "the actual gauge observation in millimeters" for rain(Kaggle).
  - This dataset had upwards of 1,000,000 entries. A lot of these rows had a vast amount of missing data so we ended up using only 252 rows to do our models upon.
- Multiple Linear Regression
  - First, we split our revised dataset into a training and test dataset.
  - Then we ran a multiple linear regression which did not incorporate any Principal Component Analysis and used all variables in our dataset.
  - We used our model to get Predicted values for the expected rainfall and compared it to the actual values of the rainfall.
  - We observed the R^2 and RMSE for this model in order to see how it compared to the Principal Component Analysis Model.
- Principal Component Analysis
  - Like in Multiple Linear Regression, we split our revised dataset into a training and test dataset using the same seed we set in Multiple Linear Regression.
  - We then ran Principal Component Analysis on our 24 variables in order to see how many principal components we want in our new model.
  - When we ran a summary of after running Principal Component Analysis, it gave us a table that included Validation:RMSEP and TRAINING: % variance explained.
  - From Validation:RMSEP, it calculates the RMSE as you add more components. The Lowest RMSE happens at 15 components.
  - From Training Percent Variance, it tells us the variance in our response variable(expected) which is explained by the amount of principal components. At all 22 components you reach 100 percent, but at 15 you already reached 98.68 percent. Therefore based off of Validation:RMSEP and TRAINING: %,we deduced it was best to run 15 components onto our new model.
  - We then ran the model based off of 15 principal components and recorded R^2 , RMSE and MAE.

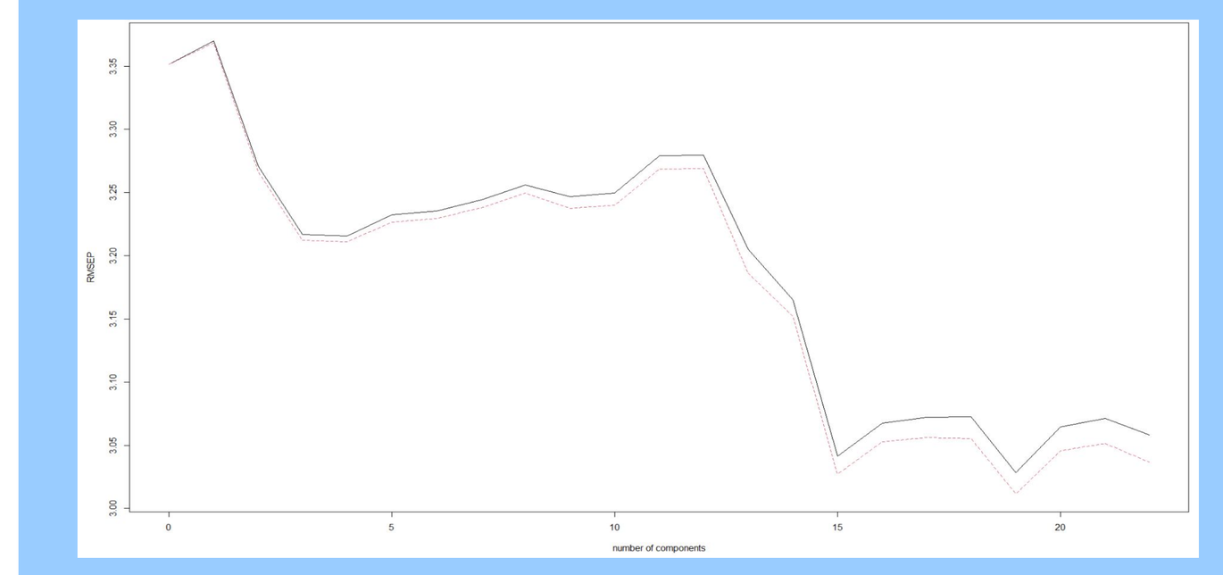
RESULTS

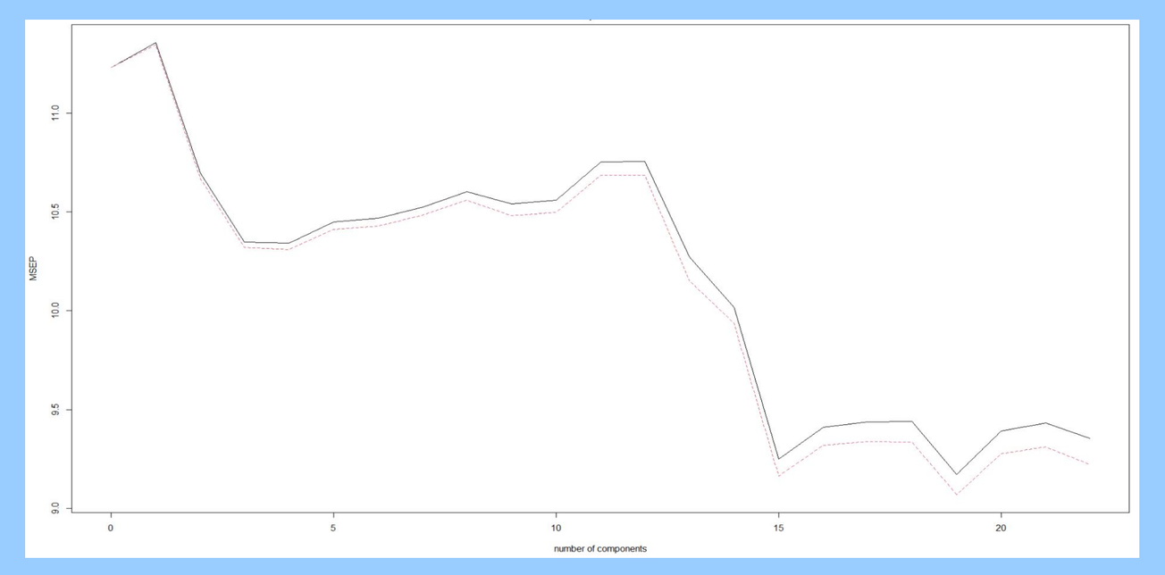
PCA Model

- To create our model using PCA, we first needed to calculate our principle components.
- The image on the right shows the principle components calculated by PCA and this was information we used to decide, which components to include in our model.

```
Data:   X dimension: 251 22
        Y dimension: 251 1
Fit method: svdpc
Number of components considered: 22

VALIDATION: RMSEP
Cross-validated using 10 random segments.
       (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
CV           3.351    3.370    3.271    3.217    3.216    3.233    3.235    3.244
adjCV        3.351    3.368    3.267    3.212    3.211    3.227    3.229    3.238
       8 comps  9 comps  10 comps  11 comps  12 comps  13 comps  14 comps  15 comps
CV       3.256    3.247      3.25     3.279     3.280     3.205     3.165     3.041
adjCV    3.250    3.238      3.24     3.269     3.269     3.186     3.152     3.027
       16 comps  17 comps  18 comps  19 comps  20 comps  21 comps  22 comps
CV        3.067     3.072     3.072     3.028     3.064     3.071     3.058
adjCV     3.053     3.056     3.055     3.012     3.046     3.051     3.037

TRAINING: % variance explained
          1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps
X         36.8525   54.432    65.54    72.09    76.81    81.22    84.66    87.92
Expected   0.1154    8.679    11.66    12.47    12.50    12.51    12.58    12.62
          9 comps  10 comps  11 comps  12 comps  13 comps  14 comps  15 comps
X           90.49     92.67     94.61     96.06     97.10     98.01     98.68
Expected    14.52     15.34     15.39     16.16     25.23     25.88     30.46
          16 comps  17 comps  18 comps  19 comps  20 comps  21 comps  22 comps
X            99.14     99.45     99.66     99.82     99.91     99.97    100.00
Expected     30.48     31.40     31.61     33.74     34.19     34.58     36.18
Data:   X dimension: 200 22
        Y dimension: 200 1
Fit method: svdpc
Number of components considered: 22

VALIDATION: RMSEP
Cross-validated using 10 random segments.
       (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
CV           3.523    3.523    3.362    3.304    3.305    3.322    3.340    3.373
adjCV        3.523    3.522    3.360    3.302    3.301    3.317    3.335    3.367
       8 comps  9 comps  10 comps  11 comps  12 comps  13 comps  14 comps  15 comps
CV       3.387    3.405     3.408     3.432     3.446     3.351     3.320     3.271
adjCV    3.378    3.396     3.394     3.419     3.432     3.335     3.301     3.250
       16 comps  17 comps  18 comps  19 comps  20 comps  21 comps  22 comps
CV        3.268     3.313     3.323     3.297     3.291     3.303     3.282
adjCV     3.249     3.288     3.300     3.271     3.264     3.275     3.253

TRAINING: % variance explained
          1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps
X         36.7972    54.51    65.41    72.19    76.90    81.43    84.81    87.89
Expected   0.3035    10.52    13.79    15.12    15.12    15.19    15.19    15.75
          9 comps  10 comps  11 comps  12 comps  13 comps  14 comps  15 comps
X           90.46     92.70     94.65     96.05     97.12     98.03     98.66
Expected    16.09     17.34     17.39     19.16     26.58     29.02     32.13
          16 comps  17 comps  18 comps  19 comps  20 comps  21 comps  22 comps
X            99.11     99.42     99.66     99.81     99.90     99.97    100.00
Expected     32.14     33.28     33.83     35.14     36.08     36.48     37.91
```

To decide which components to include in our model, we looked at the RMSEP and Training % Variance explained in our model when using the training data. We wanted to find the number of components which we can include in our model that would not give us diminishing returns,





Below are the metrics in which we calculate how well our model performed, when using 15 components in our model.

R2 <dbl>	RMSE <dbl>	MAE <dbl>
0.1966326	2.453501	1.992071

Our PCA model resulted in a lower R2 value, higher RMSE, and higher MAE value.

BIBLIOGRAPHY

Ph.D., Benjamin Obi Tayo. "Machine Learning: Dimensionality Reduction via Principal Component Analysis." Medium, Towards AI, 11 June 2019, https://pub.towardsai.net/machine-learning-dimensionality-reduction-via-principal-component-analysis-1bdc77462831.

Jaadi, Zakaria. "A Step-by-Step Explanation of Principal Component Analysis (PCA)." *Built In*, 1 Apr. 2021, https://builtin.com/data-science/step-step-explanation-principal-component-analysis.

Zach. "Principal Components Analysis in R: Step-by-Step Example." *Statology*, 1 Dec. 2020, https://www.statology.org/principal-components-analysis-in-r/.

"How Much Did It Rain? II." *Kaggle*, 7 Dec. 2015, https://www.kaggle.com/competitions/how-much-did-it-rain-ii/data.

ACKNOWLEDGEMENTS

Thank you Dr. Ivona Grzegorczyk for the guidance and encouragement that you have given us throughout this project.
Thank you Kaggle for providing the dataset.