MATH 5110

APPLIED LINEAR ALGEBRA AND MATRIX ANALYSIS

PROJECT REPORT

PRUDENTIAL LIFE INSURANCE

RISK PREDICTION

By

AILIN DOLSON-FAZIO

ABHILASHA JAIN

MS APPLIED MATHEMATICS, FALL 2022

NORTHEASTERN UNIVERSITY, BOSTON

CONTENT

1. Introduction

Principal Components Analysis (PCA) is one of the most significant methods used in data preparation, in a wide range of applications. The PCA algorithm is linear. It involves taking the original data and combining it linearly in a creative way, which can assist to highlight less evident patterns in the data. The kernel PCA approach was created to address the issue of non-linearity in the data. The kernel version allows for dealing with more complicated data patterns that would not be apparent under linear transformations alone.

Principal components analysis (PCA) on the original data is the first step in principal components regression. Next, dimension reduction is done by choosing the number of principal components (m) using cross-validation or test set error. Finally, regression is done using the first m-dimension reduced principal components.

Data for this research was obtained from a Kaggle search on Prudential Life Insurance. Prudential Life Insurance categorizes their clients on risk factors ranked one to eight – one being least risk while eight is most risk. The project's objective is to assist Prudential Life Insurance in improving the efficiency of processing time as well as reducing labor intensive for new and current clients by utilizing Principal components analysis (PCA) and Kernel Principal Components Analysis. The difficulty for them is that the application process time is outmoded due to the vast number of factors to be considered.

2. Introducing Techniques

2.1 Principal Component Analysis

The underlying concepts of PCA are shared by more sophisticated approaches, making it one of the most straightforward and well-liked dimensionality reduction techniques. PCA is an unsupervised, linear, and global technique. The objective function for PCA is:

$$\min_{\boldsymbol{A}} \|\boldsymbol{X}-\boldsymbol{A}\|_{\boldsymbol{F}} \ \textit{subject to}\ \boldsymbol{rank}(\boldsymbol{A}) = \boldsymbol{k}$$

Where, $\|\boldsymbol{M}\|_{\boldsymbol{F}} = \sum_{\boldsymbol{i}}\sum_{\boldsymbol{j}} \boldsymbol{M}_{\boldsymbol{ij}}^{\boldsymbol{2}}$ is the "Frobenius norm" and **A** is a "low rank" (rank k) approximation of **X**.

The solution to this problem is given by the SVD:

$$\boldsymbol{A} = \boldsymbol{U}_{\boldsymbol{k}} \sum\nolimits_{\boldsymbol{k}} \boldsymbol{V}_{\boldsymbol{k}}^{\boldsymbol{T}}$$

where $\boldsymbol{U_k}$ contains the leading k left singular vectors of $\mathbf{Y}$ and $\boldsymbol{V_k^T}$ contains the leading k right singular vectors of $\mathbf{Y}$.

The non-zero eigenvalues of a matrix also influence its rank. Another way to think about PCA for dimensionality reduction is that it only reconstructs the data using the eigenvectors associated with the k largest eigenvalues of the covariance matrix. The variance of the data along the principal component axes is also provided by the eigenvalues of the covariance data. As a result, PCA provides a rank-k approximation of the data that preserves the most variance. (Wang, 2021)

One of the most popular methods for dimensionality reduction is PCA since it is:

Unsupervised - The statistics were not calculated using the labels.

Projectable: Projecting a fresh data point into the smaller dimensional space is also simple.

Invertible - It is simple to go from the low-dimensional space to the high-dimensional space.

Closed-form solution: The SVD offers a successful resolution to the issue.

Its flaw is that it requires a Gaussian noise model for the data and is linear in the original space. This means the PCA model may be enhanced in many beneficial ways when datasets include extra structure, either in terms of the signal or noise model. As a result, PCA comes in many different forms, such as:

Sparse PCA - constrain PCs or coefficients to be sparse (small number of non-zeros)

Linear discriminant analysis - supervised classification-based PCA that maximizes the inter-class variance

Robust PCA - good for cases where there is sparse data and/or large errors/outliers

2.2 Kernel Principal Component Analysis

Kernel Principal Component Analysis is an extension of Principal Component Analysis. In other words, the PCA techniques are evaluated in Kernel space allowing us to solve problems with non-linearity. (Pelliccia, 2020)Kernel Principal Component Analysis can be used to solve non-linear problems because it uses the eigenvalues of the kernel matrix in place of the covariance. The eigenvalues are stated as such:

$$\boldsymbol{K_{ij} = \kappa(\overrightarrow{x_i}, \overrightarrow{x_j})}$$

Where $\boldsymbol{\kappa}$ is the kernel function, and the radial basis function is as such:

$$\kappa_{rbf}(x_i, x_j) = \exp\left(-\gamma \left|\left|x_i - x_j\right|\right|^2\right)$$

2.3 Principal Component Regression

Principal Component Regression (PCR) is a regression technique used in data science. What makes the PCR technique unique form the derived input columns $\boldsymbol{z_m = Xv_m}$ and then regresses y on z_1, z_2, ….z_m for some $\boldsymbol{m \leq p}$. Principal components regression discards the p–m smallest eigenvalue components.

PCR is used when the explanatory variables are considered highly correlated or colinear. Its main purpose is to eliminate any sense of multicollinearity by strictly using principal components not associated with small eigenvalues. Dimension reduction is accomplished by manually setting the projection onto the main component directions with tiny eigenvalues set to zero (i.e., only keeping the large ones). In some ways, PCR and ridge regression are extremely similar. Ridge regression may be conceptualized as decreasing the projection on each principal component direction after projecting the y vector onto the directions of the principal components. The variation of the primary component determines how much shrinkage occurs. Ridge regression causes everything to be smaller, but it never causes everything to become zero. In comparison, PCR either eliminates or hardly reduces a component.

There are two main steps to the principal component regression. The first step is to apply principal component analysis to generate the principal components from the predictor values. The number of principal components should equal the number of original values. The second step is to fit the linear regression model. It should be noted that the first principal component is determined by cross-validation and is kept representing variance. The first principal component will be smaller than the number of original features. The smaller the number of principal components the smaller the variance should be.

There are many pros and cons to using the PCR technique. PCR has two key features that make it beneficial. The first one is that it reduces overfitting. The second is that it allows for significant performance improvements in data sets with high cardinality by eliminating multicollinearity. The biggest con to using PCR is there is no guarantee that the principal component with the biggest variance will be the one that gives the

most accurate prediction. This is in part because the predictor values lose their explainability when the original values are turned into linear combinations. (Leung, n.d.)

3. Data Description

The Prudential Life Insurance dataset contains a training data and a testing data. It has 128 features in total and the response variable is an ordinal measure of risk that has eight levels – one being of least risk and eight being of most risk. (Anna Montoya, 2015)

Training data: 59381

Testing data: 19765 (no labels in response variable)

Variable	Description
Id	A unique identifier associated with an application.
Product_Info_1-7	A set of normalized variables relating to the product applied for
Ins_Age	Normalized age of applicant
Ht	Normalized height of applicant
Wt	Normalized weight of applicant
BMI	Normalized BMI of applicant
Employment_Info_1-6	A set of normalized variables relating to the employment history of the applicant.
InsuredInfo_1-6	A set of normalized variables providing information about the applicant.
Insurance_History_1-9	A set of normalized variables relating to the insurance history of the applicant.
Family_Hist_1-5	A set of normalized variables relating to the family history of the applicant.
Medical_History_1-41	A set of normalized variables relating to the medical history of the applicant.
Medical_Keyword_1-48	A set of dummy variables relating to the presence of/absence of a medical keyword being associated with the application.
Response	This is the target variable, an ordinal variable relating to the final decision associated with an application

Figure 1. Data Description

4. Data Pre-processing

4.1 Check Null Values

Initiated the project by checking if there is any missing value in the dataset. Python was used to extract the columns which contains the missing value as well as the percentage of missing value in both training and testing dataset. The plots are as follows:

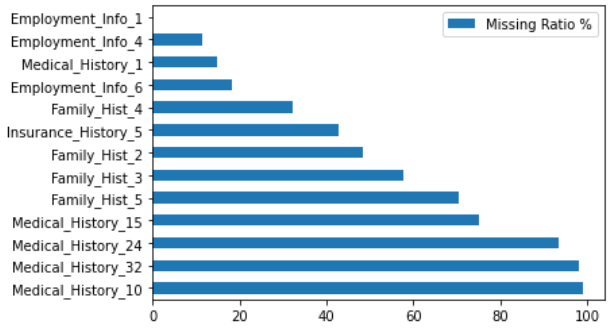

Figure 2. Testing Set

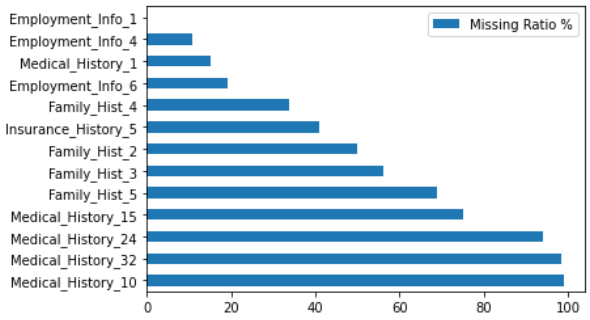

Figure 2. Test Set

4.2 Imputing Null Values

As it is visible from the graph, most of the values in Medical_History_10, Medical_History_24 and Medical_History_32 is missing and a few more columns have more than 50% of values are missing. There are several ways to compute missing values such as case analysis, imputation, and missing indicator. In this case, a null threshold of 40% was set and all features above the 40% mark was removed. The results of null plots are as follows:

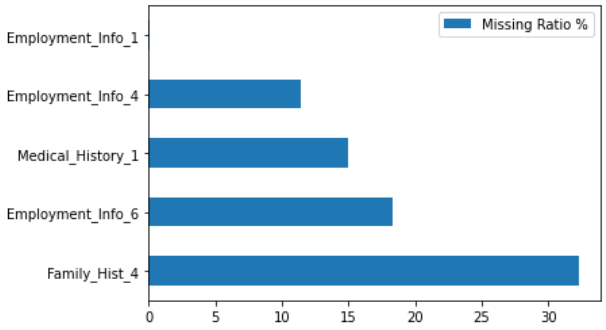

Figure 4. Testing Set

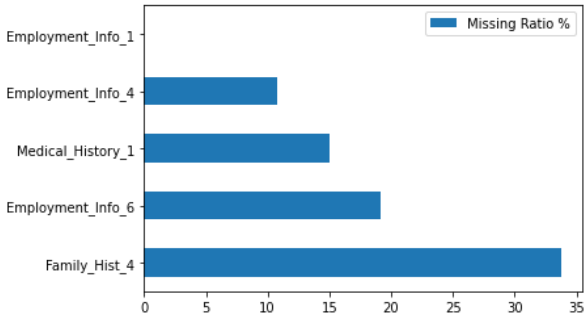

Figure 3. Test Set

For rest of the columns, resulting values were used to compute their corresponding skewedness and replaced the null values with mean values; if the data was not skewed median values were placed. The determined mean value is $\overline{x} = 0.0776$ and the determined

median value is $M = 0.0600$. Because the data was skewed to the left, determined mean value were used. First column was dropped as its value was so close to zero that it became negligible. Because the data followed a normal distribution, both the numerical median and the categorical median was used to impute the data. Below is a box and whisker plot and a histogram of the Employment_Info_1.

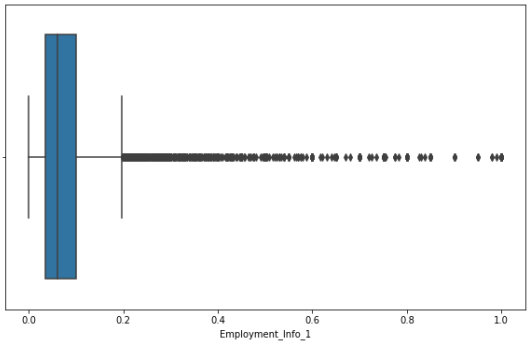

Figure 4. Box and Whisker Plot of Employment_Info_1

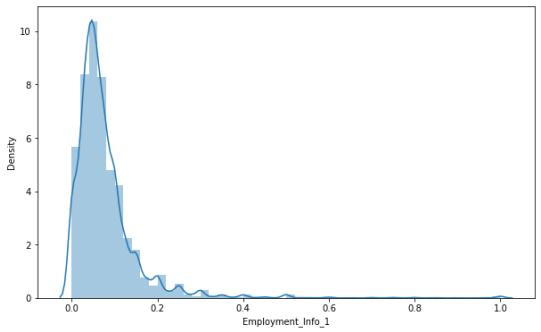

Figure 5. Histogram of Employment_Info_1

4.3 Removing Outliers

Removed any outliers. Response was also excluded as it was considered a target variable. From the yielded results, correlation was checked before dropping the results with the least correlation.

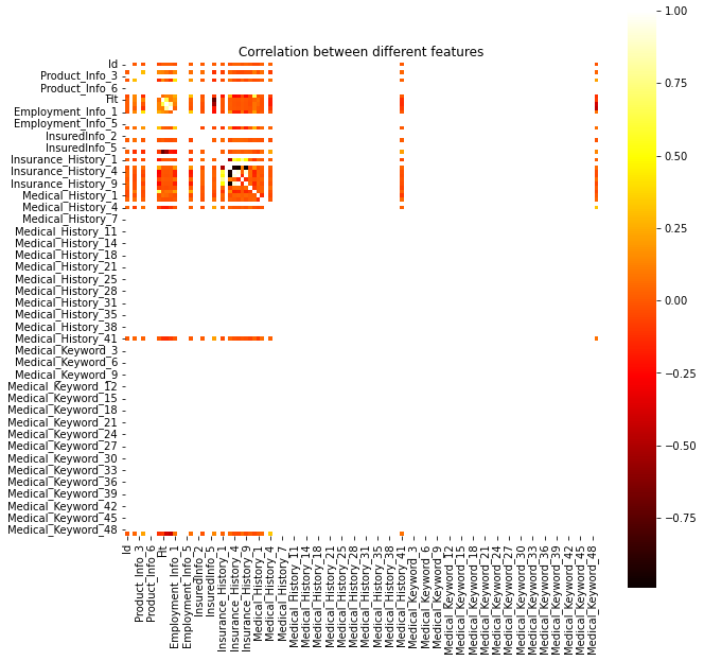

Figure 6. Correlation Heat Map

However, since the dataset was too large, correlation was not deduced. Because of this, feature score using Mutual Independence (χ^2) was calculated. The resulting image is as follows:

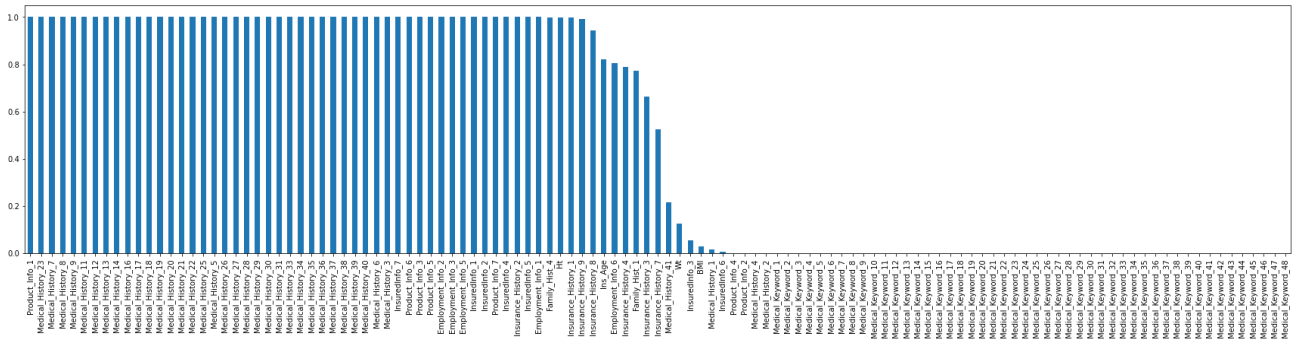

Figure 7. Feature Score using Mutual Independence (χ^2)

By calculating the feature score using Mutual Independence, some columns with extremely low importance were yielded, thus allowing to drop them before normalizing the remaining data. We were then able to use Grid Search to cross validate the parameters within multiple different models. Using Grid Search Cross Validation, we chose the best model of logistic regression and gradient booster. Anomaly from chosen model is depicted below:

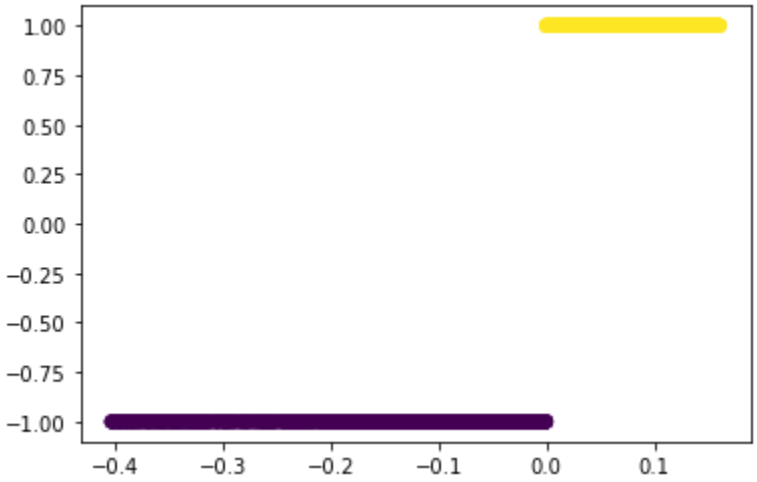

Figure 8. Anomalies vs Score calculated using Isolation Forest

5. Machine Learning Models Used

For the given dataset, 50% Principal Components from PCA and kernel PCA on 50-50 Train – Test split on Training Dataset is used to train the following models. The type of Kernel that was used is Radial Basis Function Kernel.

5.1 Logistic regression

5.2 Linear regression

5.3 Support vector classification

5.4 SGD classifier

5.5 XGB classifier

5.6 Random Forest Classifier

5.7 K-Neighbors Classifier

5.8 Artificial Neural Networks.

6. Results

ML Models	*Accuracy Using PCA*	*Accuracy Using Kernel PCA*
Logistic Regression	99.80	94.43
Linear Regression	94.83	94.33
Support Vector Classifier	94.25	94.35
SGD Classifier	58.05	94.43
XGB Classifier	99.95	94.43
Random Forest Classifier	99.93	94.00
K-Neighbors Classifier	94.25	94.43
Artificial Neural Networks	94.25	94.43

7. Conclusion

Principal Component Regression, Principal Component Analysis, and Kernel Principal Component Analysis are powerful tools for logistic linearization. Principal component analysis (PCA) is a well-known dimensionality reduction technique; however, it also serves as the foundation for Principal Component Regression (PCR). It is a regression technique that serves the same goal as standard linear regression — model the relationship between a target variable and the predictor variables. When the data set demonstrates multicollinearity, it is employed. Thus, variances may be too far from the true value even while least squares estimates are skewed. The

regression model's bias is increased, and the standard error is decreased through PCA. A data set representing the calculation of risk factors for Prudential Life insurance allowed for the implementation of Principal Component Regression and the comparison of Principal Component Analysis versus Kernel Principal Component Analysis. In this model, Radial Basis Function (RBF) Kernel was used. The comparison yielded the following results: higher accuracy scores for Principal Component Analysis when calculating logistic regression, linear regression, XGB classifier, and Random Forrest Classifier. While the Kernel Principal Component Analysis yielded higher accuracy scores for support vector classification, SGD classifier, K-Neighbor Classifier, and artificial neural network. It was also found that Principal Component Analysis and Kernel Principal Component Analysis yielded mostly comparable results for all tests except for SGD classifier in which the Kernel Principal Component Analysis performed significantly better than the Principal Component Analysis. From the observed accuracies, it can be said that the dataset is better fitted for ensemble and boosting models.

8. References

Anna Montoya, B. B. (2015). *Prudential Life Insurance Assessment*. Retrieved from Kaggle: https://kaggle.com/competitions/prudential-life-insurance-assessment

COMP652. (2016, March 14). Retrieved from McGill: https://www.cs.mcgill.ca/~dprecup/courses/ML/Lectures/ml-lecture13.pdf

Leung, K. (n.d.). *Principal Component Regression*. Retrieved from Towards Data Science: https://towardsdatascience.com/principal-component-regression-clearly-explained-and-implemented-608471530a2f

Pelliccia, D. (2020, 6 10). *PCA and Kernel PCA Explained*. Retrieved from NirPy Research: https://nirpyresearch.com/pca-kernel-pca-explained/

Wang, H. (2021). *Principal Component Analysis.* Boston: Machine Learning and Statistical Learning Theory.