Rancel Hernandez

D212 - Data Mining II

01-11-2025

TASK 2: DIMENSIONALITY REDUCTION METHODS

A1: PROPOSAL OF QUESTION

Can the PCA effectively reduce the dimensionality of the dataset while maintaining a comparable predictive performance in a logistic regression model that predicts the likelihood of readmission?

A2: DEFINED GOAL

One goal of the data analysis is to reduce the dimensionality of the training data while maintaining the ability to reliably predict the likelihood of readmission for a patient. Currently, there are millions of patients in hospitals that generate vast amounts of data, so reducing the dimensionality of the dataset can significantly decrease computational demands, improve model runtime, and optimize performance. Principal Component Analysis (PCA) can serve as a feature selection method by identifying the components that capture the most relevant feature variance [1]. By evaluating the variance explained by each component and testing model performance on the reduced subsets, the PCA can determine the optimal components to reduce the dimensionality without compromising the reliability of the predictions.

B1: EXPLANATION OF PCA

PCA analyzes the dataset by identifying the direction in the feature space with the most variance and assigning it as the first principal component (PC1). Principal components represent the directions of maximum variance in the data, so the features must be scaled to have a mean of zero and a standard deviation of one to ensure they are equally assessed. If the features were not scaled, their influence on the principal components could be inflated [2]. The features are assigned loadings based on how much their variance aligns with the current direction of maximum variance, and correlated features with strong variance in this direction will have higher loadings to reflect their greater influence on that principal component.

After calculating the loadings for each feature in the first component, PCA identifies the next direction with the highest variance. This direction must be orthogonal to PC1 and any previous component, meaning it is uncorrelated and pointing in a completely different direction that is perpendicular [3]. This is accomplished by choosing the direction that results in a dot product of zero and has the next greatest variance.

After determining the next principal component, the loadings are calculated for the features. This process continues until the total number of components is reached or until the number of components equals the number of features. However, since the components are chosen by maximizing variance, each subsequent component explains less variance [4]. The result is a matrix of loadings indicating how each feature influences the components, and this matrix can be multiplied with the original dataset to produce a DataFrame where each column represents a linear combination of the original features weighted by their component loadings [5]. This helps reduce the dimensionality of the original dataset because each component

captures an independent direction of variance where only the most relevant components are used to train the model.

B2: PCA ASSUMPTION

One assumption of the PCA is that the features used must be on the same scale, or the results could be misleading. PCA selects principal components by maximizing variance, so features with different scales could lead to a bias toward those with larger values [2]. Additionally, these features would have inflated loading values in the principal components, overshadowing smaller features that have variance pointing in the same direction. In other words, using unscaled features of different scales in PCA is redundant and can lead to bias because it effectively results in selecting larger features based solely on their wider spread or variance, which may not be meaningful. Therefore, it is best practice to standardize the features to have a mean of zero and a standard deviation of one each to ensure each feature contributes equally to the PCA [6].

C1: CONTINUOUS DATA SET VARIABLES

The following table contains the continuous variables and descriptions from the given dictionary used in the PCA.

Name	Description
Income	The annual income of the primary insurance holder of the patient.
Initial_days	The duration of the patient's initial hospitalization in days.
TotalCharge	The total charge divided by the number of days the patient was hospitalized, excluding specialized treatments.
Age	The patient's age in whole years.
Additional_charges	The average amount of miscellaneous charges for a patient.
Population	The population within a mile radius of the patient.
VitD_levels	The patient's vitamin D level measured in ng/mL.

C2: STANDARDIZATION OF DATA SET VARIABLES

The standardized variables are stored in a DataFrame called scaled_data and will be submitted in a separate CSV file.

D1: PRINCIPAL COMPONENTS

The matrix of all the principal components is shown in the following figures: *Figure 1* displays a table of the matrix, and *Figure 2* presents a heatmap highlighting the high loading values.

	Income	Initial_days	TotalCharge	Age	Additional_charges	Population	VitD_levels
PC1	-0.022638	0.700390	0.701504	0.092011	0.088339	0.023287	0.004773
PC2	-0.031521	-0.094914	-0.085333	0.700290	0.700747	-0.033206	0.012735
PC3	0.703051	0.016579	0.012366	0.020978	0.017479	-0.197016	-0.682445
PC4	-0.077643	-0.019794	-0.016737	0.012332	0.030017	0.932988	-0.348970
PC5	0.705808	0.001722	0.002318	0.018558	0.016180	0.298216	0.642096
PC6	-0.003940	0.028453	-0.032695	0.706772	-0.705995	0.011682	-0.003689
PC7	0.002028	-0.706376	0.706473	0.025796	-0.035397	-0.002020	-0.001801

Figure 1: Displays the matrix of all principal components.

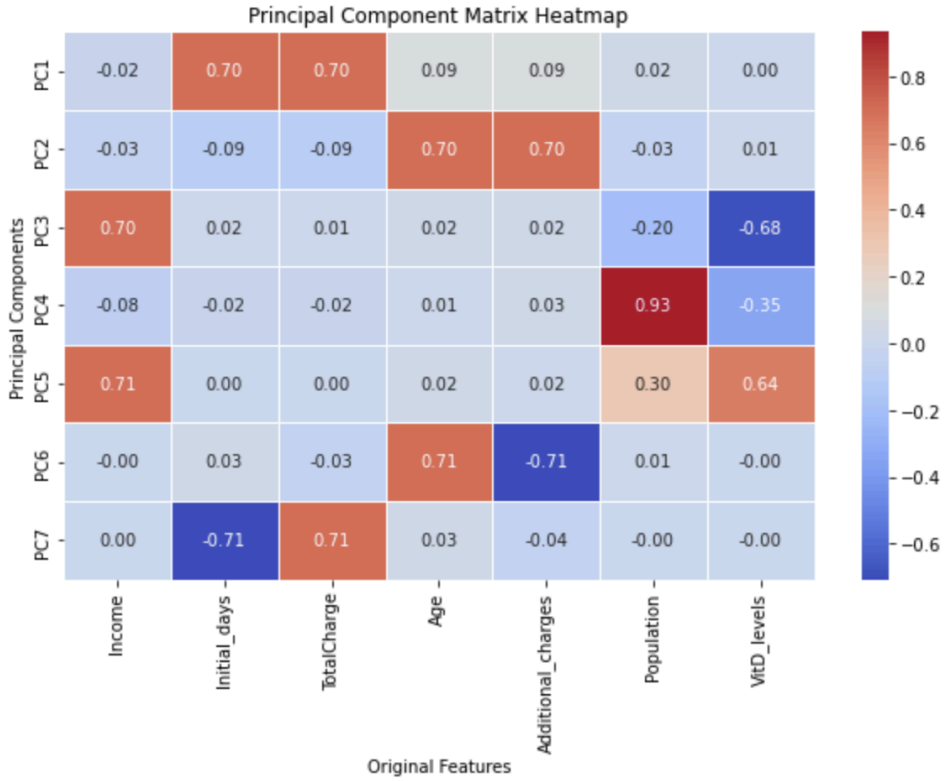

Figure 2: A heatmap representing the matrix of all principal components.

D2: IDENTIFICATION OF THE TOTAL NUMBER OF COMPONENTS

The total number of principal components equals the number of features, so there are seven principal components. The optimal number of components is three because the scree plot shows a plateau in explained variance after PC3, as shown in Figure 3. According to the Kaiser criterion, only principal components with an eigenvalue of at least one should be retained [7]. Eigenvalues quantify the spread or variance of the data along the axes defined by the principal components, indicating how much of the data's variance is explained by each component [8]. As shown in Figure 4, only the first three components meet this threshold. Therefore, three principal components are the optimal number for explaining the majority of the variance.

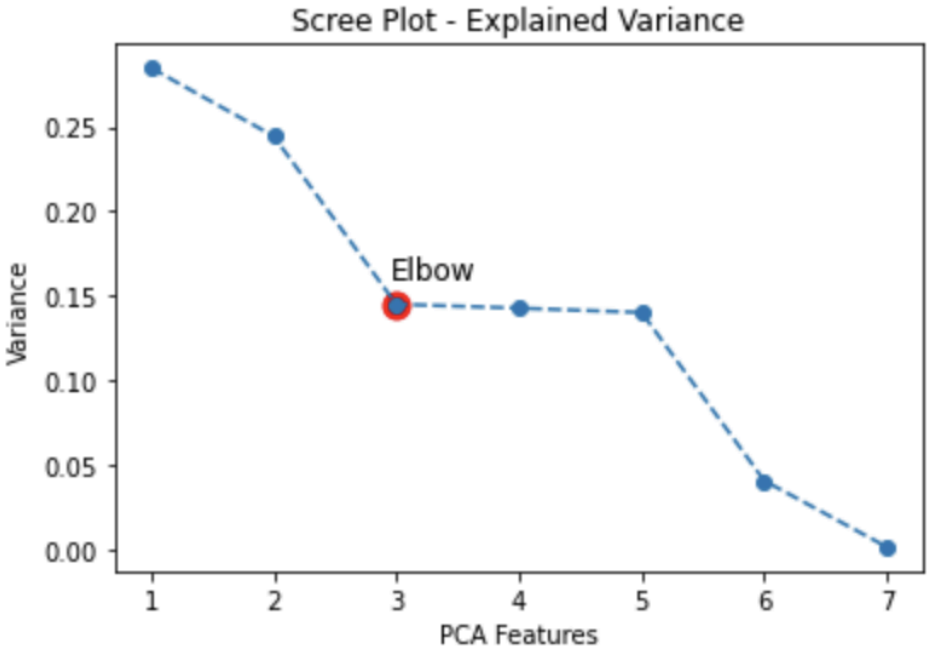

Figure 3: The scree plot showing the explained variance for each principal component, with an elbow observed at the third principal component.

D3: VARIANCE OF EACH COMPONENT

The following figure contains the variance of each component, and the eigenvalue is equivalent to the variance.

```
Eigenvalues: [('PC1', 1.9936), ('PC2', 1.7117), ('PC3', 1.016),
('PC4', 0.9999), ('PC5', 0.982), ('PC6', 0.286), ('PC7', 0.0117)]
```

Figure 4: Shows the eigenvalues of the principal components, with only the first three components exceeding the threshold of one.

D4: TOTAL VARIANCE CAPTURED BY COMPONENTS

The total variance captured by all seven principal components is approximately 7.0009, while the variance captured by the optimal three components is approximately 4.7214.

D5: SUMMARY OF DATA ANALYSIS

The PCA resulted in seven principal components where the first three components captured a majority of the explained variance. Three models were tested on different subsets of the training data to evaluate the effectiveness of dimensionality reduction: a base model using all continuous variables, a PCA model using the first three components, and a PC1 model using only the first principal component.

The first performance metric evaluated is accuracy, which quantifies how correct the model's predictions are. However, accuracy can be misleading when the target variable is imbalanced, and this was the case in this analysis [9]. The dataset was imbalanced regarding

readmission status, with approximately 6,500 non-readmitted patients compared to 3,500 readmitted patients. The accuracy scores for the models are as follows: baseline model 98.05%, PCA model 97.65%, and PC1 model 95.70%. Despite the reduction in training data, the decrease in accuracy between the models is minimal, with a small gap of approximately 2.35% across all scores. This difference could be considered negligible if accuracy is not the primary metric being prioritized.

The AUC score measures how effectively a model ranks predictions to distinguish between the positive and negative classes [10]. In other words, it signifies how well a model ranks the likely outcomes. All three models achieved an identical AUC score of 0.9988, indicating they are equally capable of ranking the probabilities of different outcomes regardless of the subset of training data used. The consistent AUC score across models suggests that the smallest data subset, PC1, contains the essential information needed to distinguish between readmission status.

The final performance metric assessed was the recall score, which was prioritized over accuracy due to the imbalance in the target observations. The recall maximizes the identification of true positives, and this can potentially increase the number of false positives [11]. In this analysis, a true positive refers to a readmitted patient, and maximizing true positives is crucial because there are few in this imbalanced dataset. A goal of the analysis is to identify the factors influencing readmission rates because the Centers for Medicare & Medicaid Services (CMS) penalize hospitals with high readmission rates to encourage initiatives that lower readmissions and improve patient outcomes. Therefore, identifying as many patients at risk of readmission as possible is prioritized to ensure they receive proper care, reduce their risk, and potentially avoid fines from CMS.

It is likely that the original features with high loadings in PC1 effectively explain the variance of the target variable, which is why the other components were not necessary for achieving a good performance. However, it should be noted that this result occurred because the features with the greatest variance happened to be correlated with the target variable. It is possible that other components with less variance could have also achieved good performance if the features with high loadings in those components were correlated with the target, or readmission status in this case.

In conclusion, while the baseline model achieved the highest performance across all metrics, the choice of the best model depends on the priorities of the network. If maximizing recall is the primary goal, the PCA model would be the best option because it achieved the highest recall and the same AUC score as the base model while reducing the dimensionality of the dataset. Conversely, if prioritizing efficiency is more important, the PC1 model would be the better choice. It performed similarly to the other two models while reducing the training data to its lowest possible dimensionality. In either case, both models performed on par with the baseline model, making the PCA a successful approach.

E: SOURCES FOR THIRD-PARTY CODE

Pandas Documentation, "pandas.set_option," *Pandas*, [Online]. Available: https://pandas.pydata.org/docs/reference/api/pandas.set_option.html. [Accessed: Jan. 10, 2025].

Scikit-learn Documentation, "sklearn.decomposition.PCA," *Scikit-learn*, [Online]. Available: https://scikit-learn.org/1.5/modules/generated/sklearn.decomposition.PCA.html. [Accessed: Jan. 11, 2025].

Seaborn Documentation, "seaborn.heatmap," *Seaborn*, [Online]. Available: https://seaborn.pydata.org/generated/seaborn.heatmap.html. [Accessed: Jan. 11, 2025].

F: SOURCES

Reference List

[1] GeeksforGeeks, "Principal Component Analysis (PCA)," *GeeksforGeeks*, [Online]. Available: https://www.geeksforgeeks.org/principal-component-analysis-pca/. [Accessed: Jan. 10, 2025].

[2] Scikit-learn, "Scaling importance," *Scikit-learn*, [Online]. Available: https://scikit-learn.org/1.1/auto_examples/preprocessing/plot_scaling_importance.html. [Accessed: Jan. 10, 2025].

[3] A. Kim, "Why is the second principal component orthogonal to the first one?" *Medium*, 27-Oct-2020. [Online]. Available: https://medium.com/intuitionmath/why-is-the-second-principal-component-orthogonal-to-the-first-one-d453c9fd97ca. [Accessed: Jan. 10, 2025].

[4] GraphPad Software, "PCA: Process, Eigenvalue, and Eigenvector," *GraphPad*, 2025. [Online]. Available: https://www.graphpad.com/guides/prism/latest/statistics/stat_pca_process_eigenvalue_eigenvector.htm. [Accessed: Jan. 11, 2025].

[5] B. Boehmke, "Principal Component Analysis (PCA)," *Hands-On Machine Learning*, 2025. [Online]. Available: https://bradleyboehmke.github.io/HOML/pca.html. [Accessed: Jan. 11, 2025].

[6] R. Gopinath, "Key Statistics Terms: 5. Standardization and Normalization," *Medium*, 2021. [Online]. Available: https://medium.com/@mail2rajivgopinath/key-statistics-terms-5-standardization-and-normalization-7a0123d60fa5. [Accessed: Jan. 11, 2025].

[7] R Core Team, "KGC function," *CRAN*, 2024. [Online]. Available: https://search.r-project.org/CRAN/refmans/EFAtools/html/KGC.html. [Accessed: Jan. 11, 2025].

[8] S. Cansiz, "Covariance matrix," *Built In*, 27-Sep-2021. [Online]. Available: https://builtin.com/data-science/covariance-matrix. [Accessed: Jan. 11, 2025].

[9] J. Brownlee, "Failure of accuracy for imbalanced class distributions," *Machine Learning Mastery*, 2019. [Online]. Available: https://machinelearningmastery.com/failure-of-accuracy-for-imbalanced-class-distributions/. [Accessed: Jan. 12, 2025].

[10] A. Bhandari, "AUC-ROC curve in machine learning," *Analytics Vidhya*, 2020. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/06/auc-roc-curve-machine-learning/. [Accessed: Jan. 12, 2025].

[11] T. Keldenich, "Recall, precision, and F1 score: A simple metric explanation in machine learning," *Inside Machine Learning*, 2021. [Online]. Available: https://inside-machinelearning.com/en/recall-precision-f1-score-simple-metric-explanation-machine-learning/. [Accessed: Jan. 12, 2025].