Dimensionality Reduction Methods – D212

Gooden, Nina S. Gooden [ID #: 009823504]

Dr. Kesselly Kamara

Dimensionality Reduction Methods – D212

This analysis will explore the medical data of a theoretical real-world organization. By utilizing the given dimension reduction method, I will address the business’ concern about exploring the characteristics of their patients. I will create visualizations, as well as code to support my analysis.

## **A. The Research Question:**

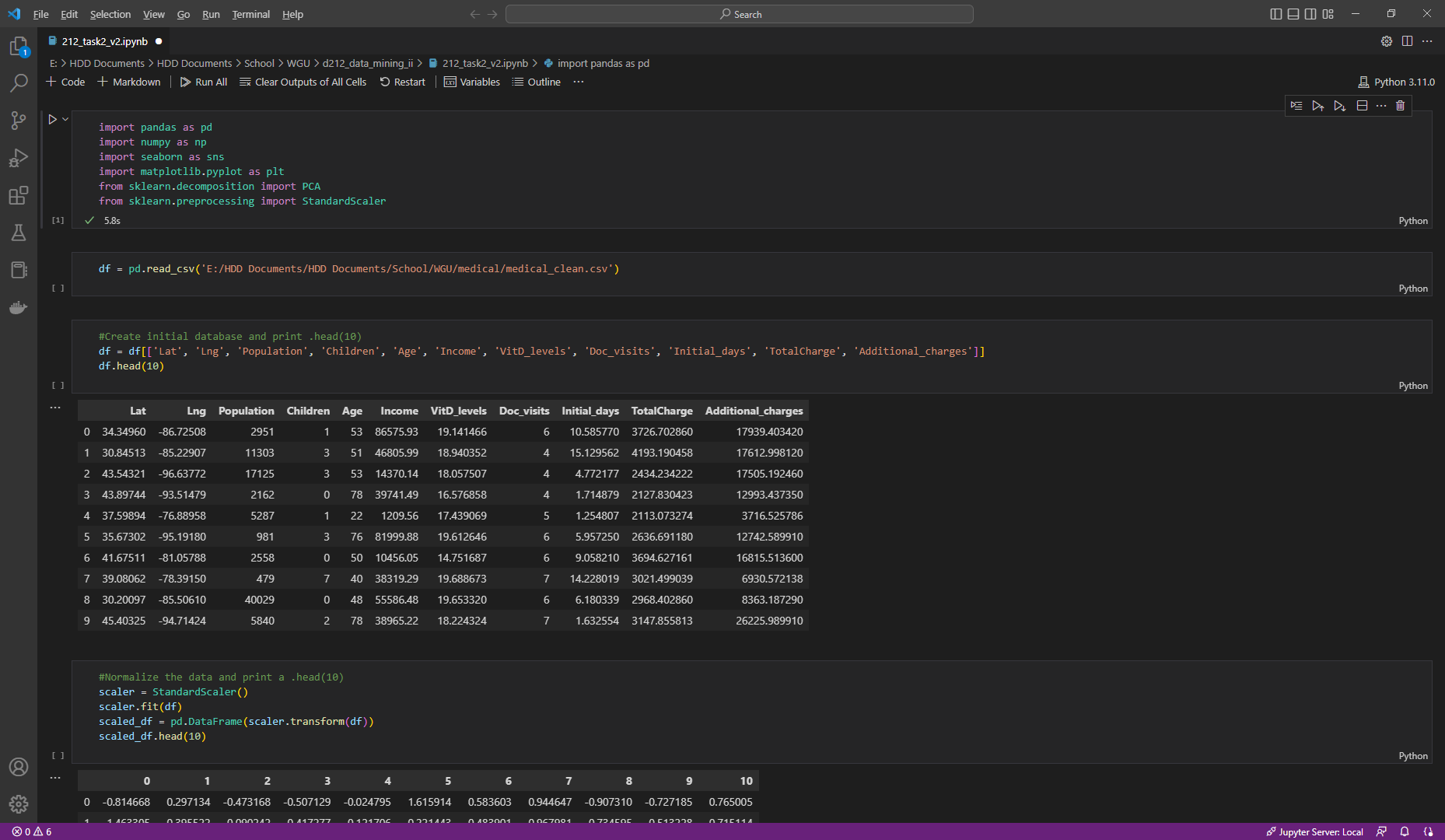
1. Research Question
   1. Which of the listed features of the medical data patient group will return the best metrics for further analysis?
2. Data Analysis Goals and Objective
   1. As principal component analysis is a technique used for “feature extraction” I anticipate that using PCA will take my high dimension dataset of 11 variables and lower that number by at least 40%.

## **B. Justification for Technique**

1. Principal component analysis is a dimensionality-reduction method used to reduce large data sets into smaller, more manageable packages. It does this by shifting the data, while maintaining the variation present in the dataset. PCA transforms the data into a new form (principal components), while keeping the original variation steady. This allows users to visualize the data on a smaller, more manageable scale. This practice is useful because in doing so, much of the “noise” data can be dropped for simplicity.
   1. **Mathematically, principal component analysis (PCA) accomplishes the aforementioned dimensionality reduction by identifying datasets with numerical variables and keeping linear combinations between the columns in a larger matrix X with a maximum variance.  
      After preparing data in the manner mentioned in the previous entry (transformed feature set, based on variance, removing those with 0 variance and/or minimizing distance between data points to shift toward eigenvector ), PCA forms linear combinations of said variables.  
      In order to identify the first principal component, PCA considers the value with the maximum variance within the variables, as well as its linear combinations. This consideration is subject to certain parameters, such as the sum of the coefficients squared must equal 1 in order to provide a distinctive answer.  
      The second principal component is the variable that encompasses as much of the remaining variation as possible but the correlation of between the first and second principal components much be 0. As such, the sums of the squared coefficients must still equal 1 and the components must not be correlated.  
      Following principal components follow a similar roadmap. They are the next, highest variable that encompasses as much of the leftover variation as possible, while not being correlated to the other components.  
      In completing this task until the remaining variance is either accounted for completely or there are no additional noncorrelated features, PCA allows users to identify components that are specific to the needs of the data to combat overfitting. (Kathuria, 2021)**
   2. The expected outcome for this analysis is: I anticipate that using PCA will take my high dimension dataset of 11 variables and lower that number by at least 40%, to 7 or less.
2. Principal Component Analysis assumption summary (Vadapalli, 2022):
   1. Assumes that the principal component(s) with high variance must be more valuable than those with lower variance, which will be designated as “noise.”

## **C. Data Preperation**

1. Identify the continuous variables:
   1. 'Lat', 'Lng', 'Population', 'Children', 'Age', 'Income', 'VitD\_levels', 'Doc\_visits', 'Initial\_days', 'TotalCharge', 'Additional\_charges'



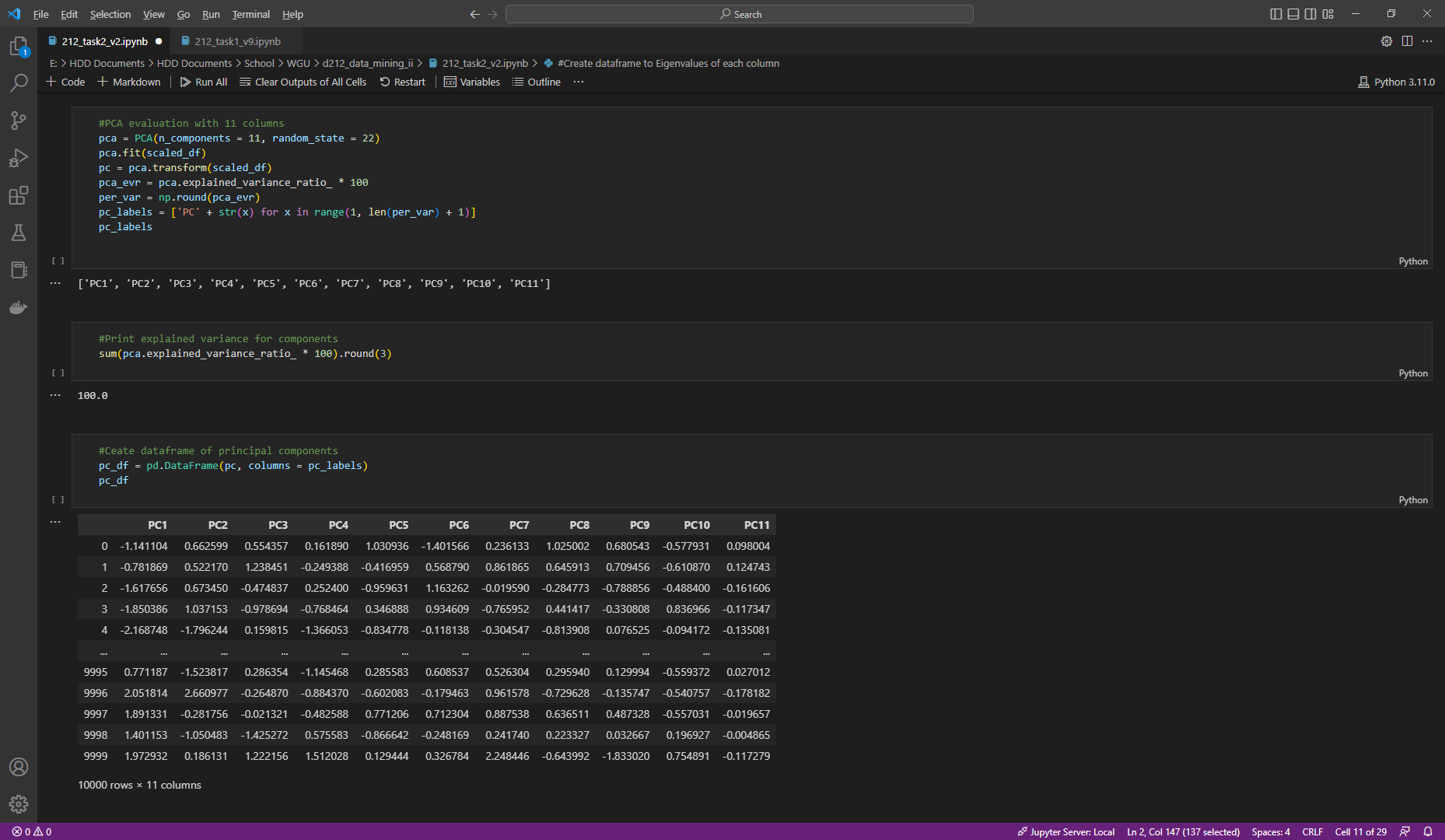
1. Next, I normalized the data and saved the updated DataFrame.

A picture containing graphical user interface

Description automatically generated

## **D. Analysis**

1. Determine the matrix of the principal components
   1. To find the matrix, I started with the evaluation of the 11 columns.

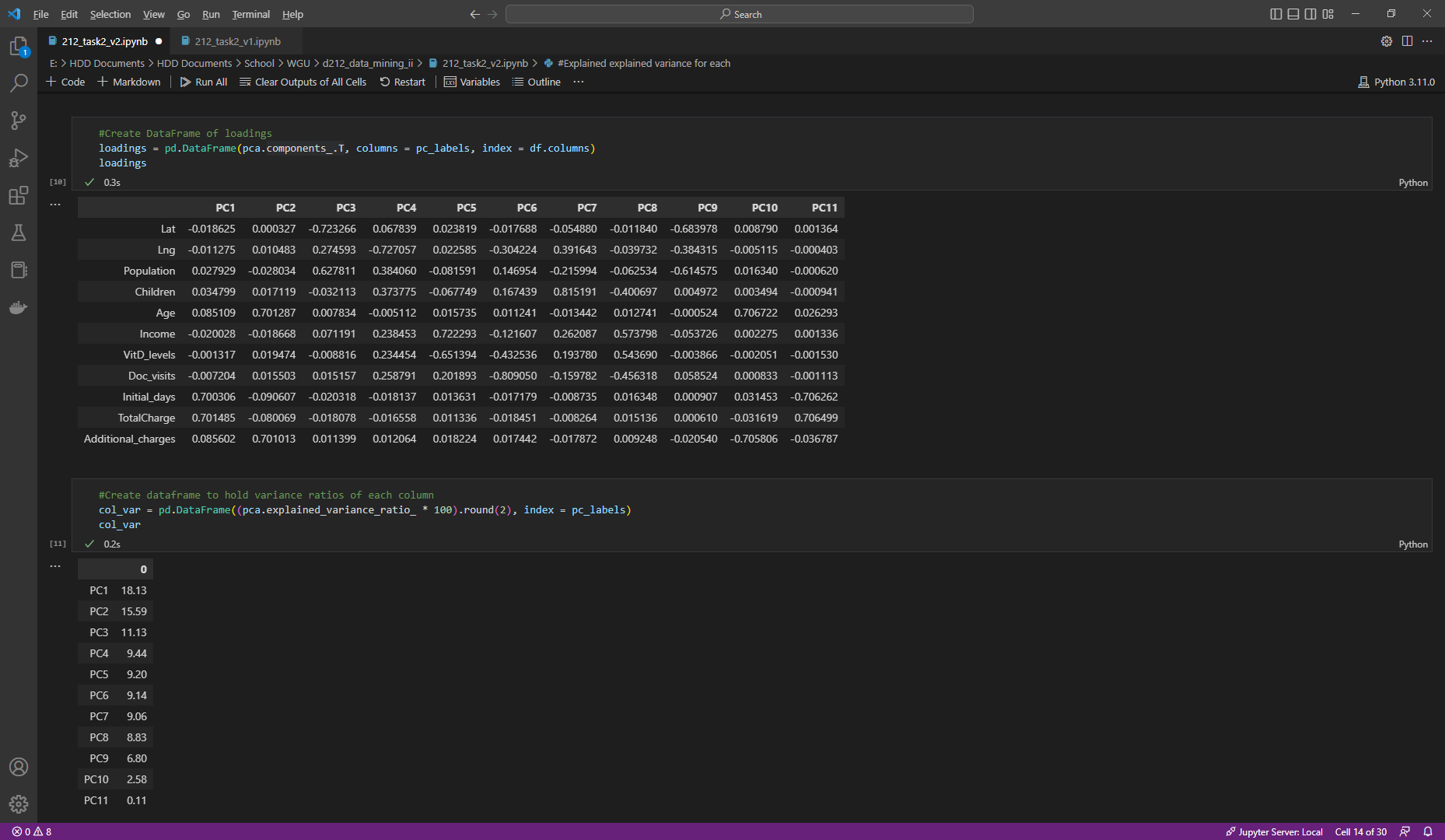


* 1. Next, I created a DataFrame for the principal components.

A screenshot of a computer

Description automatically generated

* 1. While I was doing that, I also created a DataFrame for the load factors.



* 1. Which allowed me to create the initial component matrix

Graphical user interface

Description automatically generated

1. Identify the *total* number of principal components:
   1. I also created DataFrames for the variance ratios and the Eigenvalues of each column.

A screenshot of a computer

Description automatically generated with medium confidence

* 1. I did a quick sanity check to make sure my variance sum equaled out to 100 and that the EVR list matched the DataFrame I had created.

A screenshot of a computer

Description automatically generated with medium confidence

* 1. I used the .cumsum function to calculate the cumulative sum of the Eigenvalues.

A screenshot of a computer

Description automatically generated with medium confidence

* 1. I created a scree plot for the principal components. This graph shows the cumulative sum produced by the .cumsum function I used in the previous step. Based on the **Kaiser criterion**, as well as the eigen\_df Dataframe, I looked for values greater than 1 and the threshold point for 80%. It looks like 6 or 7 would be my optimum principal components.

Graphical user interface, website

Description automatically generated

1. Identify the variance for *each* of the principal components:
   1. I updated my model to reflect the identified principal components

A screenshot of a computer

Description automatically generated

* 1. And created a loadings DataFrame for the updated values.

Graphical user interface, text

Description automatically generated

* 1. I then created DataFrames for the updated variance ratios and eigenvalues for each column.

A screenshot of a computer

Description automatically generated with medium confidence

1. Identify the *total* variance of the principal components
   1. And lastly, I ran the previous calculations again for the updated/reduced components.

A screenshot of a computer

Description automatically generated

1. Summarize the results
   1. While including all 11 of the principal components resulted in an explained variance ratio of 100, our research goal was to get to at most 7 but as few components as possible. Using the Kaiser criterion, we identified 6 and 7 components to be our best bet for reduction, and so we were able to bring that to the following correlations from our matrix:
      1. PC1: Initial\_days and TotalCharge at .7 each
      2. PC2: Age and Additional\_charges at 0.7 each
      3. PC3: Lat and Population at -.72 and .63 respectively
      4. PC4: Lng at -.73
      5. PC5: Income and VitD\_levels at .72 and -.65 respectively
      6. PC6: Doc\_visits at -.81
   2. Our updated model with 6 prinicpal components has a total cumulative explained variance of a little under .73, which is respectable. Given the options provided from the Kaiser criterion, I am confident that raising that number to 7 would get us over the .8 threshold.

Graphical user interface

Description automatically generated

## **E. Reference web sources**

Pramoditha, R. (2021, December 15). *Principal Component Analysis (PCA) with Scikit-learn.* Medium. https://towardsdatascience.com/principal-component-analysis-pca-with-scikit-learn-1e84a0c731b0

*sklearn.decomposition.PCA*. (n.d.). Scikit-learn. https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html

Raschka, S. (2015, January 27). Principal Component Analysis. Sebastian Raschka, PhD. https://sebastianraschka.com/Articles/2015\_pca\_in\_3\_steps.html

Mangale, S. (2021, December 15). Scree Plot - SANCHITA MANGALE. Medium. https://sanchitamangale12.medium.com/scree-plot-733ed72c8608

## **F. Acknowledge sources**

Vadapalli, P. (2022, November 22). PCA in Machine Learning: Assumptions, Steps to Apply & Applications. upGrad Blog. https://www.upgrad.com/blog/pca-in-machine-learning/

Brems, M. (2022, January 26). *A One-Stop Shop for Principal Component Analysis - Towards Data Science*. Medium. https://towardsdatascience.com/a-one-stop-shop-for-principal-component-analysis-5582fb7e0a9c

Kathuria, C. (2021, December 13). *How exactly does PCA work?* - Towards Data Science. Medium. https://towardsdatascience.com/how-exactly-does-pca-work-5c342c3077fe