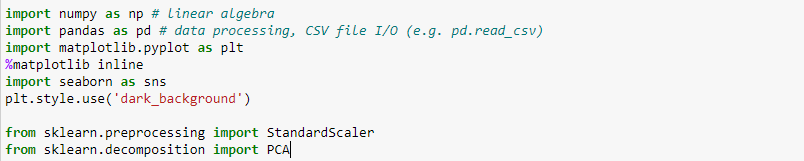
**TPS\_April\_2022\_VR2036\_PCA\_GMM**

# Created by Kaushik Kar

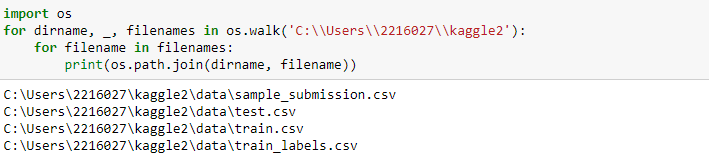
# Employment id- 2216027

1. **Importing the important libraries:**



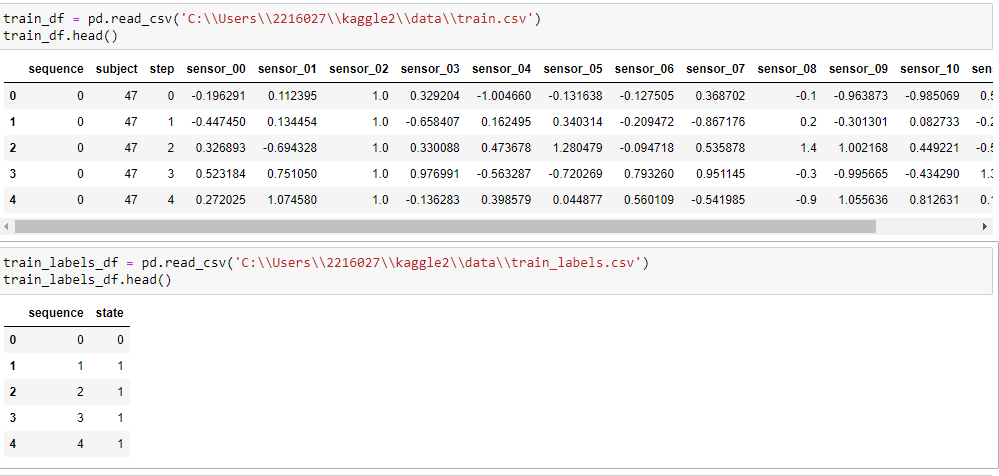
1. NumPy is used for mathematical operations like addition, subtraction, multiplication, division, etc. on arrays and matrices.
2. Pandas provides data structures for efficiently storing and manipulating large datasets, and tools for reading and writing data to and from various file formats, including CSV, Excel, and SQL databases
3. Seaborn is a data visualization library based on Matplotlib which is a plotting library used for creating static, interactive, and animated visualizations in Python.
4. The StandardScaler class from the sklearn.preprocessing module is used for standardizing numerical data. Standardization is a technique used to transform data such that it has zero mean and unit variance.
5. PCA is a dimensionality reduction technique that is used to reduce the number of features in a dataset while retaining most of the information. It does this by finding a new set of orthogonal axes (principal components) that captures the maximum amount of variation in the data.

This below code is using the OS module to walk through a directory tree and print out the path of each file in the tree such as train.csv, train\_labels.csv, test.csv and sample\_submission.csv.



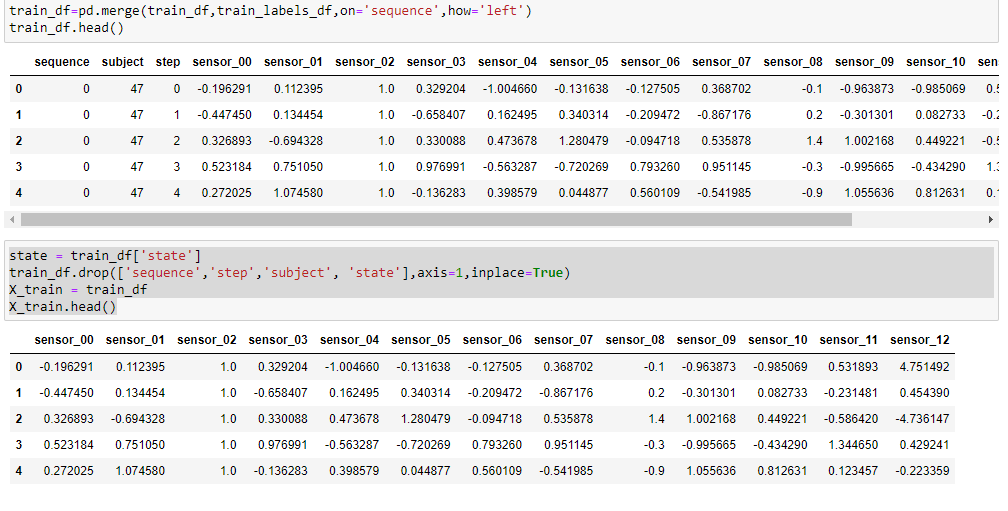
1. **Loading and pre-processing the dataset:**

With the help of pandas library, we can read and upload the data in csv form. we can display the first five rows of the data



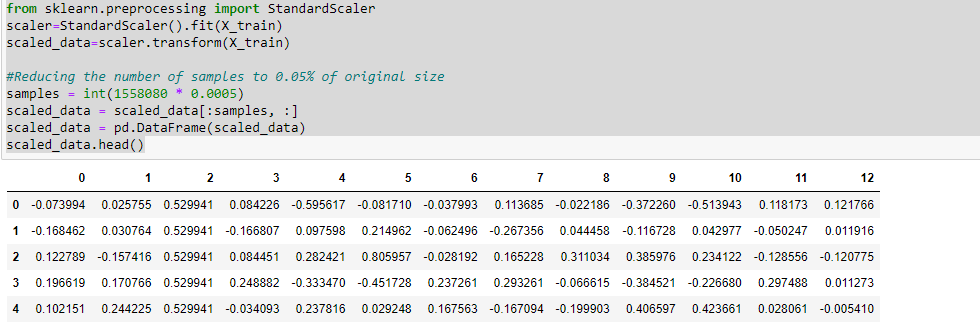
Uploaded train.csv and train.labels.csv and merge both the dataset and make a new dataset called train\_df.

Then extracts the state column from the train\_df DataFrame and assigns it to the state variable, drops the sequence, step, subject, and state columns from the train\_df DataFrame using the drop method, assigns the remaining columns of the train\_df DataFrame to the X\_train variable, prints the first few rows of the X\_train DataFrame using the head method.



1. **Normalizing the Dataset:**

Normalizing a dataset is a type of data preprocessing that scales the values of the features so that they fall within a specified range, typically between 0 and 1. Normalization is often used in machine learning to standardize the data and to make it easier to compare features that have different scales or units.

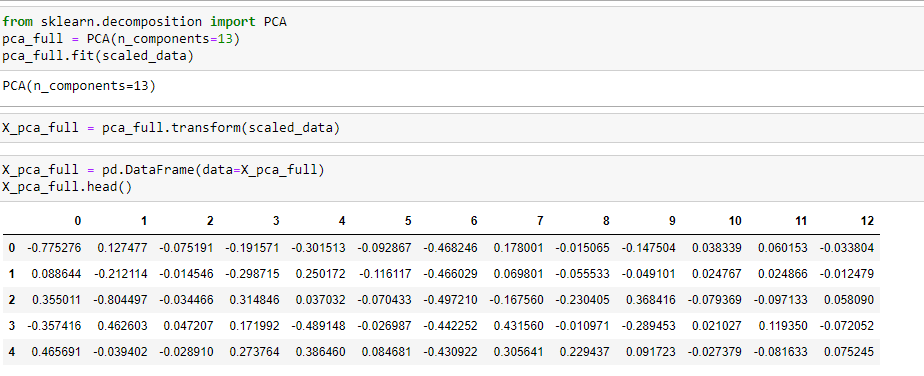


This code creates an instance of the StandardScaler class, and then uses its fit method to compute the mean and standard deviation of each feature in the X\_train DataFrame. The resulting scaler object is then used to transform the data using the transform method, which standardizes each feature by subtracting its mean and dividing by its standard deviation.

This code computes the number of samples to keep after downsampling the dataset by multiplying the original number of samples (1558080) by 0.0005 (0.05%). The resulting samples variable is an integer value that is used to slice the scaled\_data variable to keep only the first samples rows. Finally, the resulting data is converted to a pandas DataFrame using the pd.DataFrame function, and the first few rows are printed using the head method.

1. **PCA(Principal Component Analysis):**

For the decomposition part we need to install Principal Component Analysis (PCA) from sklearn on the feature-scaled data in the DataFrame X. PCA is a technique used for dimensionality reduction, where many input features are transformed into a smaller number of principal components (linear combinations of the original features), while retaining most of the variation in the data.

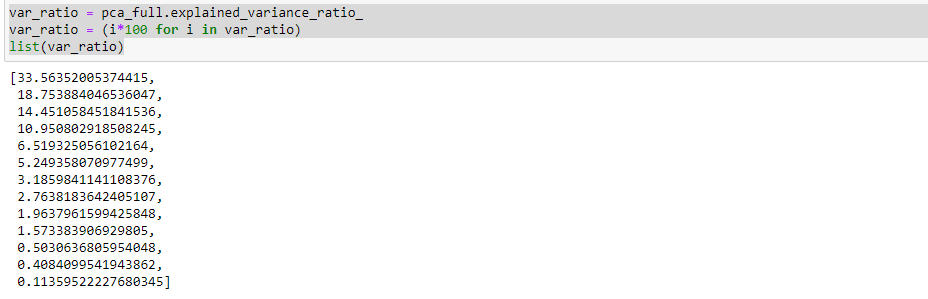


This code creates an instance of the PCA class with n\_components set to 13, which means that the output of PCA will consist of 13 principal components. The fit method of the PCA class is then used to fit the model to the standardized data in scaled\_data, which calculates the principal components based on the correlation structure of the data.

This code uses the transform method of the PCA class to transform the standardized data in scaled\_data into the space defined by the principal components. The resulting X\_pca\_full variable is a numpy array that contains the transformed data. The numpy array is then converted to a pandas DataFrame using the pd.DataFrame function, and the first few rows are printed using the head method.

1. **Scatter Plot :** Explains the %-of-variance each feature brings out. Number of features can be selected based on the top features causing 80%-90% variance in the dataset.



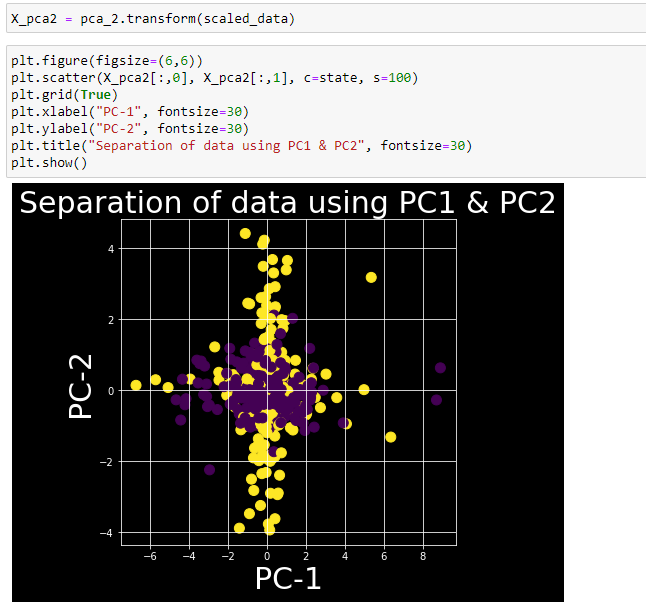


Var\_ratio is basically explained the variance from each principal component gives an indication of how much information is retained by each component and can be used to determine the optimal number of principal components to keep.

Here we also have the list of var\_ratio

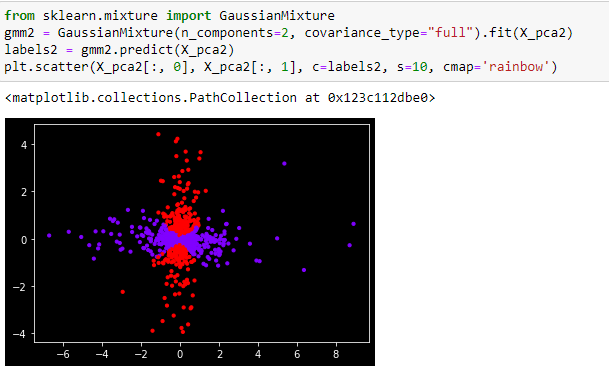
Selecting top-5 features explains approximately 84% of the total variance.

For visualization purposes, selecting only 2 of the best features. Together they explain roughly 52% of the variance.



1. **Gaussian Mixture Models (GMM):**

Gaussian Mixture Models (GMM) is a probabilistic model that assumes that the observed data is generated from a mixture of several Gaussian distributions. The goal of GMM is to estimate the parameters of these Gaussian distributions, including their means, variances, and mixing coefficients, such that the model can be used to generate new data that resembles the observed data.



**Comparing the Covariance Type:**

