Data Retrieval:

Two ways of data retrieval have been implemented:

a) A wrapper function that uses HTTP requests and JSON parsing to query and retrieve the data from FRED API. Pandas DataFrame is used to display these datasets.

b) The second implementation of data retrieval uses a third-party python API with the retrieval functionalities defined within a package called full-fred (found on the FRED API Website). Sources: https://fred.stlouisfed.org/docs/api/fred, https://github.com/7astro7/full_fred

Structure of the data:

Structure of a typical datapoint from the API has the following features:

realtime_start: start date of real-time request	*realtime_end*: start date of real-time request	*date*: first date of months until the last real-time updated one	*value*: represents yield in percentage

Data Cleansing:

Among the features above, “realtime_start” and “realtime_end” are redundant features which can be removed and stored as variables. “date” and “value” features are the critical features which are used in the modeling process.

Data obtained from the FRED API follows JSON format, and the contents are text type, which is interpreted as string datatype in Python. Date and value columns need to convert to appropriate datatypes to be used for PCA. We need the dates in DateTime format, and most importantly, the value field must be a numerical field (here, float) as we will be performing PCA, which needs the data points to contain only numerical data.

Data Preprocessing:

From the FRED API, we build a dataset with six features where each feature represents the monthly change in yield over the dates mentioned for each Treasury series, GS1_CHG to GS6_CHG. These features can be isolated and analyzed using Principal Component Analysis (PCA). Further, an iteration of PCA has been implemented to demonstrate Truncated Single Vector Decomposition (tSVD).

Some features have missing values (represented as np.nan in the dataset) these columns require some transformations. The missing values can be Imputed using sklearn SimpleImputer. We will use two impute functions; first, we replace the nan values with zero, then, we replace the nan values with the median values of the columns.

After applying the impute function to the dataset, we will standardize the data using a Standard Scaler. Now, the dataset has observations for all data points, and all the features contain numerical values – this makes the dataset viable for dimensionality reduction using PCA.

Model - Dimensionality Reduction:

The model is based on Principal Component Analysis (PCA) which projects the original features onto a transformed plane using eigenvalues and eigenvectors. These new Principal Components can be used as features. The more features used while deriving the Principal Components, the variance in data is well-explained by them.

Intuitively, we can come up with a threshold for the variance that needs to be explained by the transformed features (say 90-95%). To explore the explained variance by Principal Components, a trial-and-error method is used to reduce the given six features into lower dimensions, starting from projection onto one up to five features. This experiment would help us analyze the different explained variances in each instance. To visualize the results, we can use the "scree plot" that shows the relationship between the number of principal components and the percentage of variance explained by them.

To identify the best features, eigenvalues of the Principal Components are considered to get a clear picture of the correlation between the original feature and the transformed Principal Component. This process allows us to perform feature selection in a better manner.

Conclusion:

The ideal threshold for explained variance is 94% (generally, 95% or in the 90-95%). After running the PCA simulations on the dataset for different numbers of resultant Principal Components, we can observe that a transformation to 2 Principal Component datasets from the original state explains more than 94% of the variance in the data. Therefore, we can use this technique to project these newly transformed points and use them for training our Machine Learning models or Data Analysis. The two original features with a high correlation with the principal components are "GS5_CHG" and "GS1_CHG".

Alternatively, we can use three or more Principal Components as well. Choosing the number of components is based on the amount of space to be saved using dimensionality reduction and the degree of explained variance of the dataset.