IE41201 - AI For Finance
Assignment 1
Interest Risk Modeling Via PCA

Student name: Nguyen Minh Duc
Student ID: 20202026

1 PCA Analysis

In this first problem, we are asked to fill in the blanks to complete the `PCA_solver` function. The answer could be seen below:

```python
import scipy

def PCA_solver(df, num_reconstruct):

    dataset = df.copy()

    # Fill in the code!
    # Step 1   : Compute mean-centered data matr
    mean_centered = dataset - dataset.mean()
    # Step 2-1 : Compute Covariance matrix
    covariance_matrix = mean_centered.cov()
    # Step 2-2 : Use the function scipy.linalg.eigh
    eigenvals, eigenvecs = scipy.linalg.eigh(covariance_matrix)

    eigenvecs = eigenvecs[:, np.argsort(eigenvals)[::-1]]
    eigenvals = eigenvals[np.argsort(eigenvals)[::-1]]
    eigenvecs = eigenvecs[:, :num_reconstruct]

    return np.dot(eigenvecs.T, df.T).T, eigenvals, eigenvecs
```

Running the above function obtains the expected result. Please refer to Figure 1 for visualization.

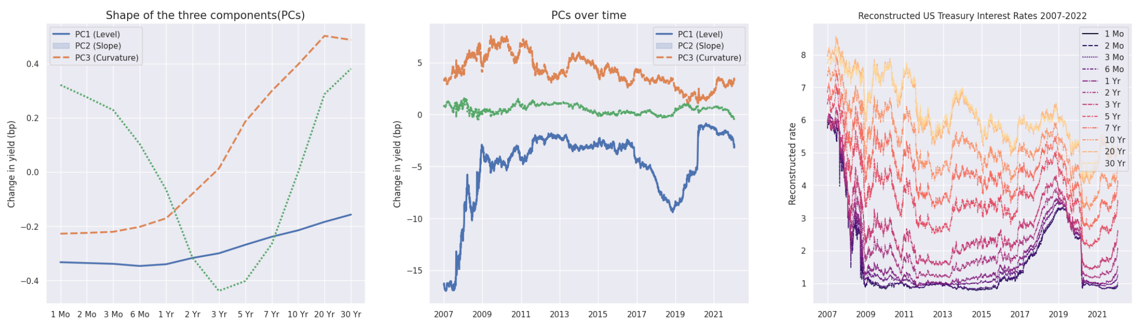

Figure 1: Principle Components Analysis

2 Different Dimension Reduction Algorithm

In this second problem, we are asked to experiment with another dimension reduction algorithm and compare the result with principle component analysis. I decide to choose t-distributed stochastic neighbor embedding (t-SNE) [6] among those algorithms that `sklearn` package [5] supports because of its strengths and popularity. The implementation of the algorithm is trivial thanks to the API provided by `sklearn`.

```
from sklearn.manifold import TSNE

def dimension_reduction(df, num_reconstruct):

    dataset = df.copy()

    fa = TSNE(n_components=num_reconstruct)
    embedded = fa.fit_transform(dataset)

    return embedded
```

The function takes the `bond_data.csv` data as the input and produces the embedded vectors in the `num_reconstruct`-dimensional space for each day in the data. Here, I use the default parameters given by the package. Please refer to Figure 2 for the comparison.

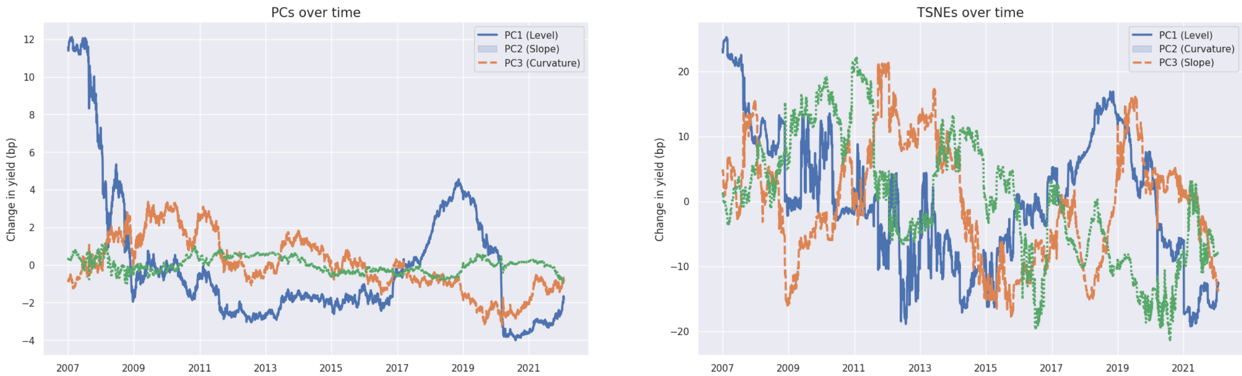

Figure 2: PCA vs t-distributed stochastic neighbor embedding

One could easily observe a significant variance from the t-sne method, compared to a reasonably stable trend from PCA. Does this mean those two results are totally not related to each other? Further investigation by computing their correlation coefficients, their relationship is discovered. Please refer to Figure 3 for more details.

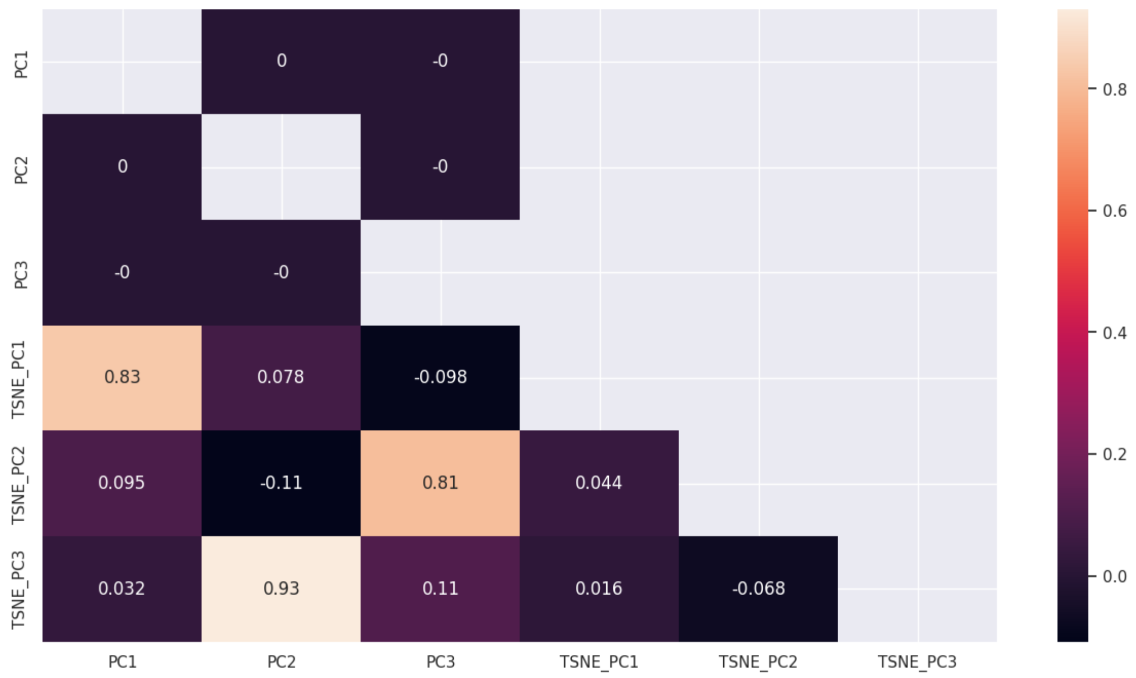

Figure 3: Correlation between PCA and t-sne

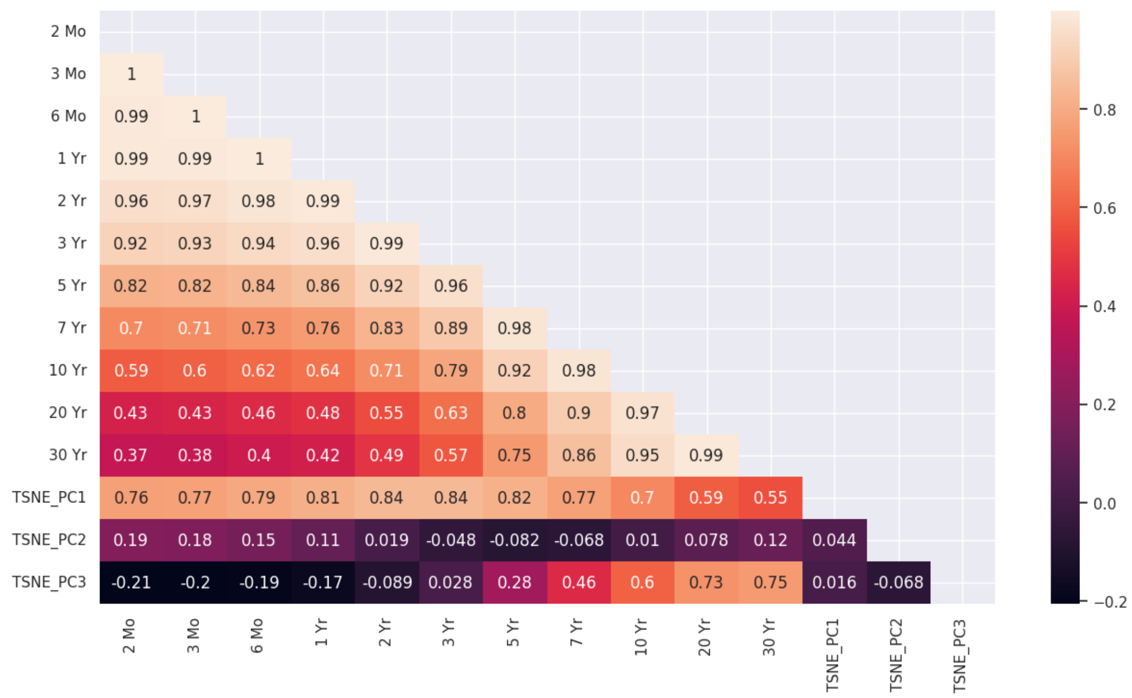

Figure 4: Correlation between t-sne and real data

Let `TSNE_PC1`, `TSNE_PC2`, and `TSNE_PC3` be the three components obtained from the t-sne process. To my surprise, they show strong correlations with the components obtained from PCA. In particular, they correspond to `PC1` (level), `PC3` (slope), and `PC2` (curvature) respectively. Therefore, even though components from t-sne stochastic behaviors with high variance, still represent the same type of components as PCA to some extent. One possible explanation to why t-sne exercises this kind of behavior but PCA does not is that t-sne is a *non-linear* dimensional reduction technique [8] while PCA is a *linear* one since it only consists of linear operations [3]. In both techniques, the "level" component seems to be the most important factor when it comes to bond analysis since the trend of those bonds follows closely with the trend of the component. Interestingly, the ordering of the other two components switches in t-sne. As one could see from Figure 4, the "level" component is useful for short-term bonds while the "slope" component is capable of capturing patterns in long-term bonds. Unfortunately, the third component, i.e., "curvature" does not seem to carry much information about the bonds' structure.

3 Applications Of Dimensional Reduction In Finance

In this section, we are asked to extend the project with an introduction to other dimensional reduction techniques that have applications in finance. In the tutorial session, the TA already introduced a machine-learning approach using a linear autoencoder and showed that this approach is equivalent to PCA [4]. Another similar method is partial least squares [7], which is also a linear dimensional reduction algorithm, and it shows the same interpretation of the bonds' structure as PCA. Applying the technique to our `bond_data.csv`, the correlation between it and PCA and the real data could be obtained in Figure 5, and Figure 6 respectively.

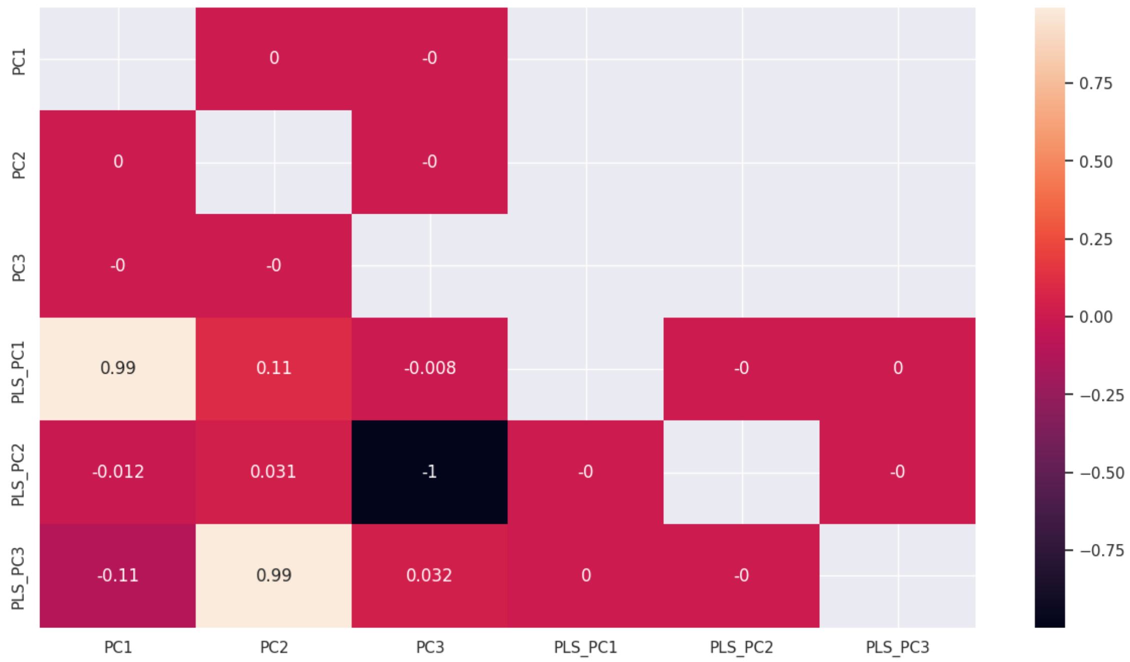

Figure 5: Partial Least Squares vs. Principle Components Analysis

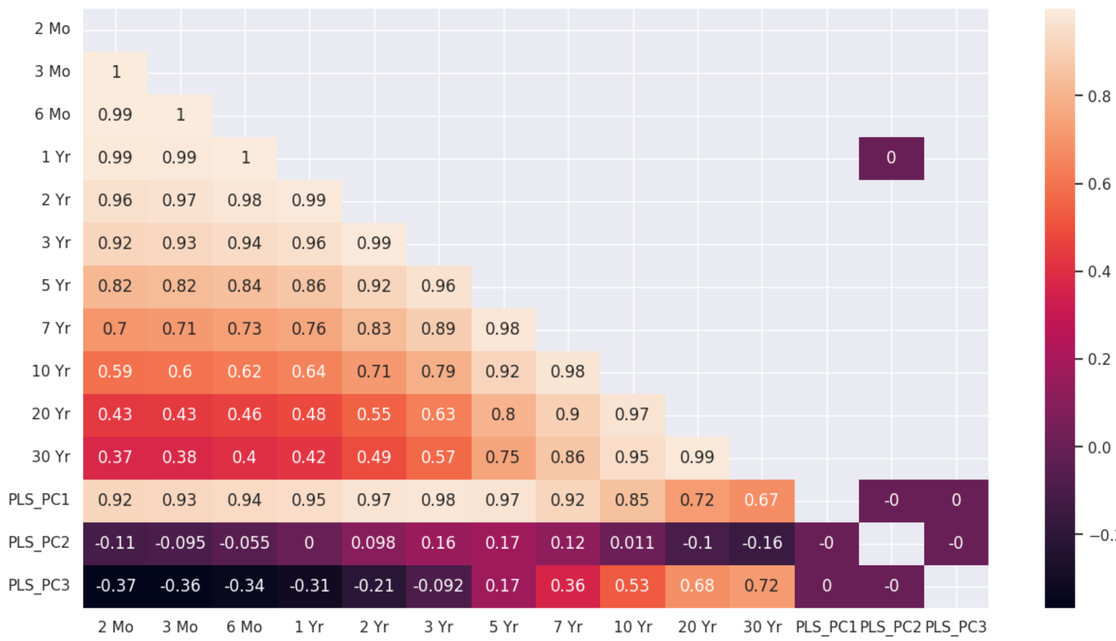

Figure 6: Correlation between Partial Least Squares and real data

To extend this project, I decided to search for other dimensional reduction techniques that could be applied to the field of Finance. Upon my research, I came across an interesting article: "*Dimension and variance reduction for Monte Carlo methods for high-dimensional models in finance*" [1]. The paper proposed **drMC**, a novel dimensional reduction technique for analyzing pricing problems under N-dimensional one-way coupled model in finance that is built upon the conditional Monte Carlo technique and a closed-form solution to a partial differential equation (PDE). The approach reduces the dimension from N down to 1, which results in a significant variance reduction. The PDEs that they used is

$$dS(t) = rS(t)dt + \sqrt{v(t)}S(t)dW_S(t), \tag{1}$$

$$dv(t) = \kappa_v\left(\overline{v} - v(t)\right)dt + \sigma_v\sqrt{v(t)}dW_v(t), \tag{2}$$

where r, S, and v denote the constant interest rate, the asset price, and its instantaneous variance respectively. From this, they proved its closed-form solution to the system of equations, after which they apply the technique to calibrate cross-currency models [2] to market data.

References

[1] Duy-Minh Dang, Kenneth R. Jackson, and Mohammadreza Mohammadi. Dimension and variance reduction for monte carlo methods for high-dimensional models in finance. *Applied Mathematical Finance*, 22:522 – 552, 2015.

[2] Lech A. Grzelak and Cornelis W. Oosterlee. On cross-currency models with stochastic volatility and correlated interest rates. *Applied Mathematical Finance*, 19:1 – 35, 2010.

[3] Yoontae Hwang. Common factors affecting bond returns via PCA. UNIST IE412 AI for Finance practice session 1 tutorial, 2023.

[4] Elad Plaut. From principal subspaces to principal components with linear autoencoders. *ArXiv*, abs/1804.10253, 2018.

[5] scikit learn. Machine Learning package in Python. https://scikit-learn.org/stable/.

[6] Laurens van der Maaten and Geoffrey E. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

[7] Wikipedia. Partial least squares regression. https://en.wikipedia.org/wiki/Partial_least_squares_regression.

[8] Wikipedia. t-distributed stochastic neighbor embedding. https://en.wikipedia.org/wiki/T-distributed_stochastic_neighbor_embedding.