IE517 MLF F20

Module 5 Homework (Dimensionality Reduction)

Yaxuan Wang (662869931)

Out[1]: The raw code for this IPython notebook is by default hidden for easier reading. To toggle on/off the raw code, click here.

Out[3]:

	Date	SVENF01	SVENF02	SVENF03	SVENF04	SVENF05	SVENF06	SVENF07	SVENF08
0	5/17/2019	2.1224	2.0266	2.1023	2.2377	2.3790	2.5042	2.6069	2.6885
1	5/16/2019	2.1239	2.0317	2.1096	2.2468	2.3901	2.5171	2.6217	2.7049
2	5/15/2019	2.0874	1.9956	2.0844	2.2289	2.3736	2.4980	2.5984	2.6779
3	5/14/2019	2.1319	2.0559	2.1451	2.2856	2.4257	2.5461	2.6428	2.7188
4	5/13/2019	2.1051	2.0234	2.1180	2.2632	2.4051	2.5248	2.6198	2.6940

5 rows × 32 columns

Part 1: Exploratory Data Analysis

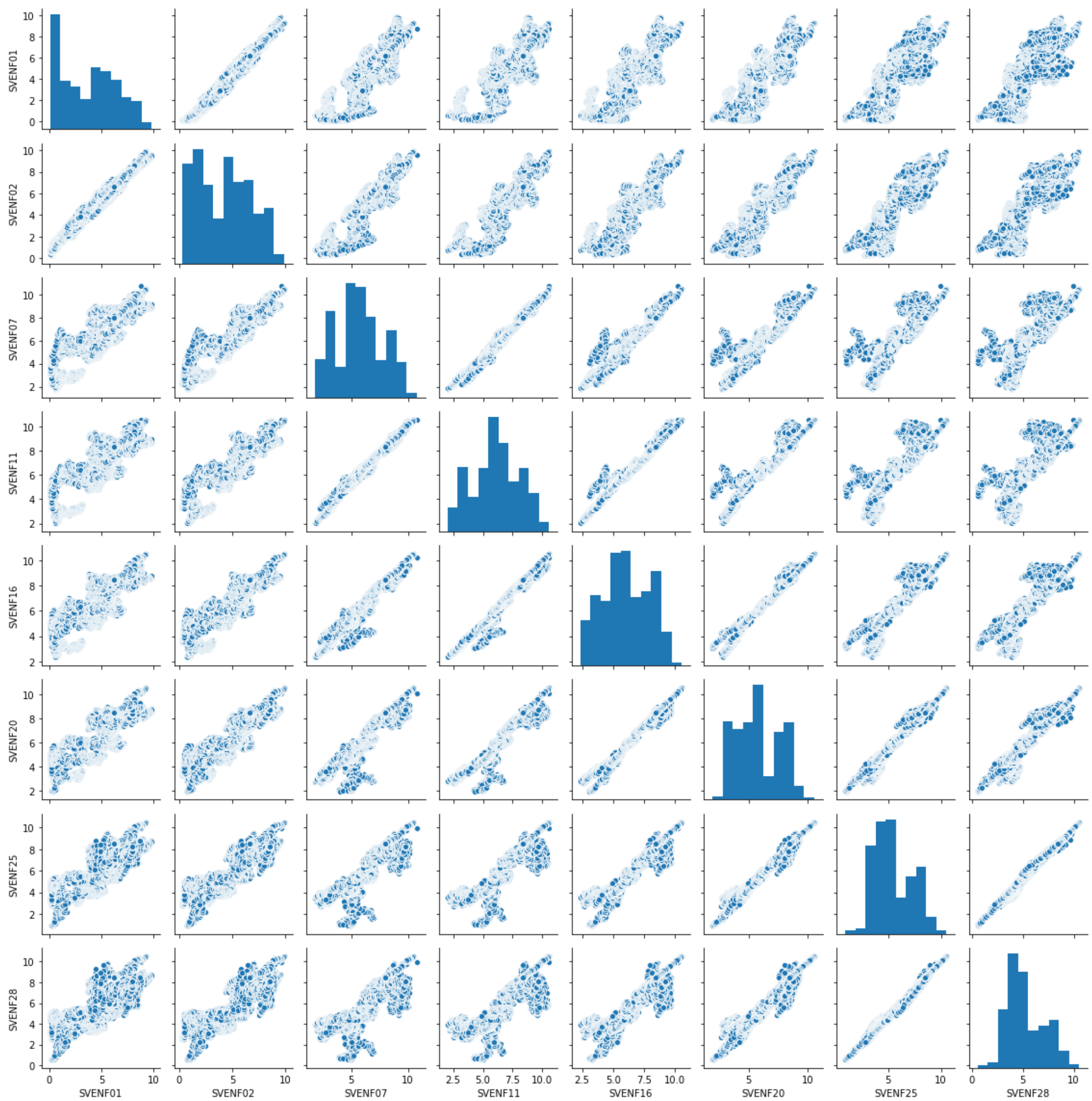
Summary Statistics

Out[4]:

	SVENF01	SVENF02	SVENF03	SVENF04	SVENF05	SVENF06	SVENF
count	8071.000000	8071.000000	8071.000000	8071.000000	8071.000000	8071.000000	8071.0000
mean	3.785311	4.258972	4.669363	5.022430	5.318493	5.559644	5.7500
std	2.648060	2.498137	2.341348	2.221632	2.137801	2.080405	2.0403
min	0.072700	0.327300	0.630300	1.013000	1.424500	1.698200	1.8073
25%	1.144050	1.865600	2.536550	3.023050	3.544700	4.063300	4.4097
50%	3.986500	4.393300	4.505500	4.718900	5.051300	5.394600	5.6637
75%	5.901500	6.221250	6.461300	6.626600	6.779550	6.908050	7.0499
max	9.813800	9.887800	10.145600	10.459900	10.649900	10.741400	10.7663

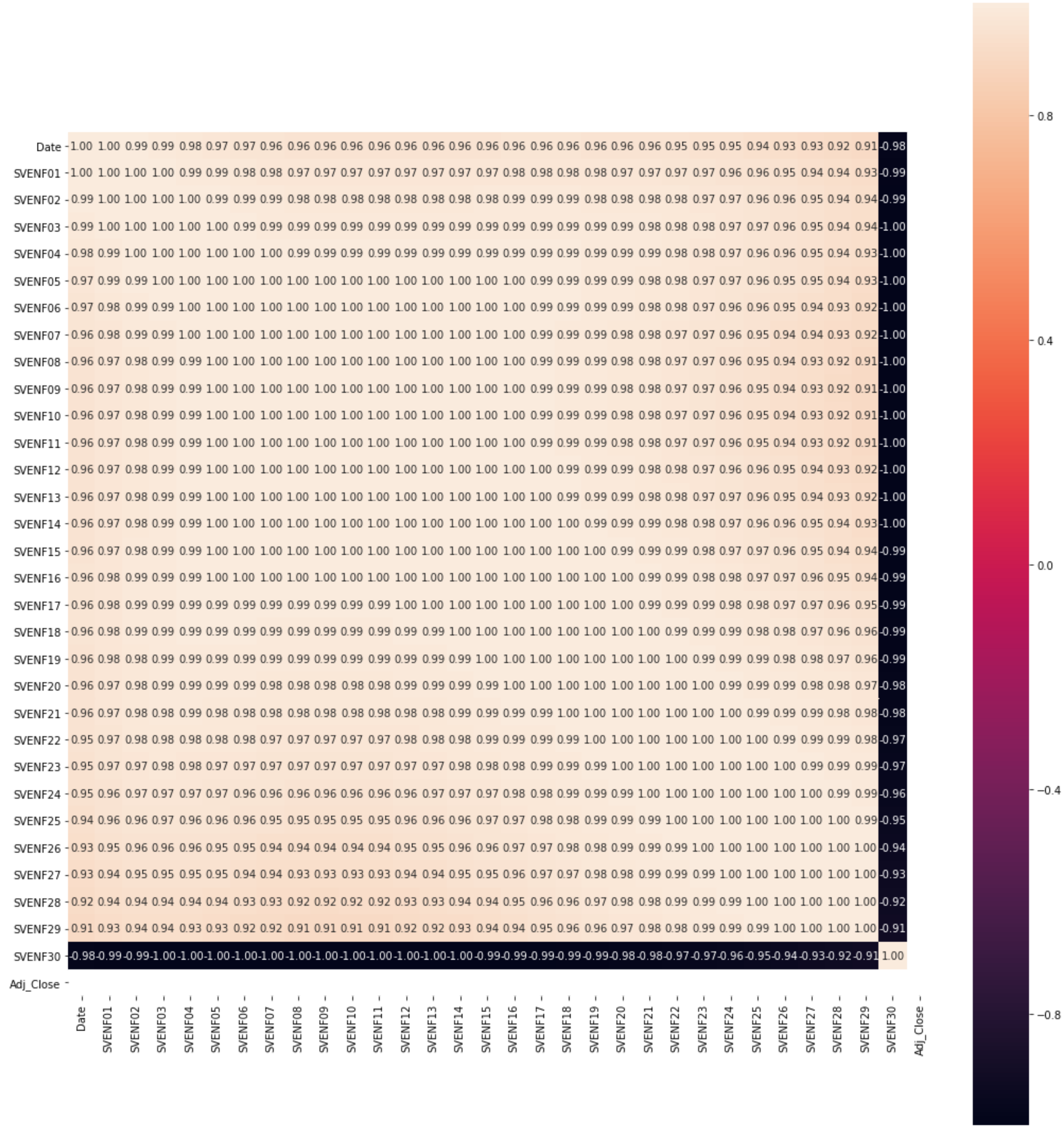
8 rows × 31 columns

Scatterplot matrix



As we can see from the scatter plot, there exists linear relationship among those variables, although some have noises. For example, apparently, the `SVENF25` has positive linear association with `SVENF28`.

Heatmap/Correlation Matrix



From the heatmap, there exists strong positive correlation among those `SVENF` variables.

Split Data into Training and Testing Sets

Now in order to test and compare our model performance in next part, we are spliting data into training and test sets. I use 85% of the data for the training set and the rest of 15% as our testing set.

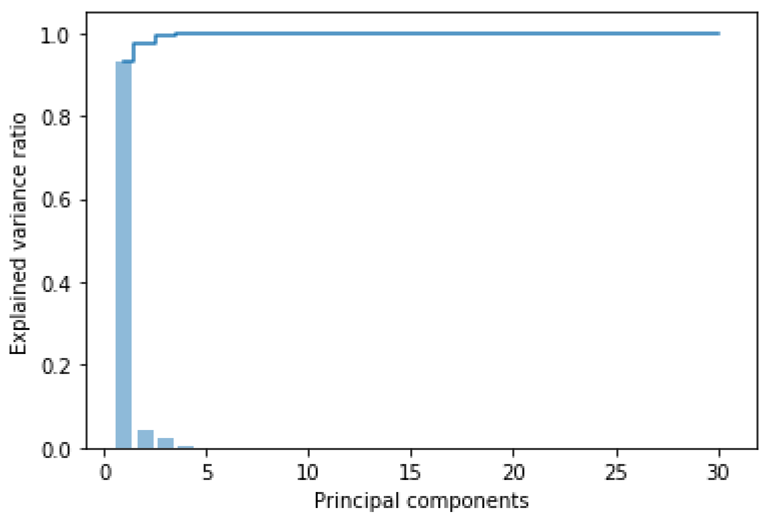
Standardized Data

In order to perform PCA in the latter section, we need to normalize our data since PCA model is very senstive the feature scaling.

Part 2: Perform a PCA on the Treasury Yield dataset

Explained Variance Ratio for All Components

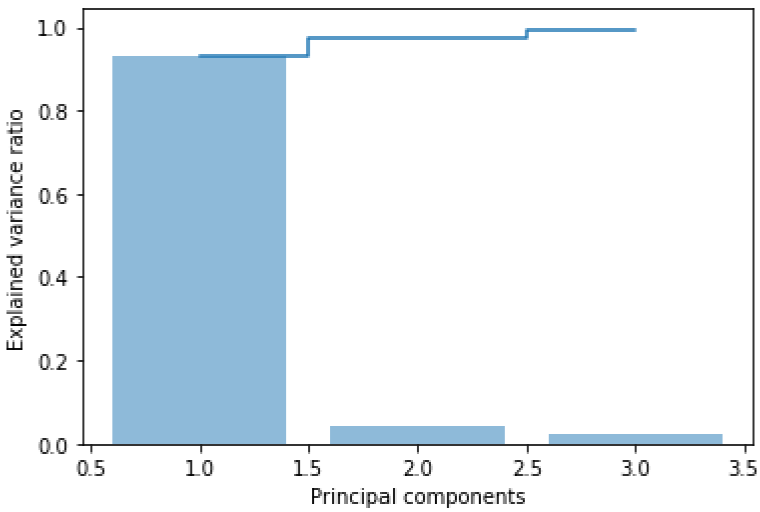
```
The explained variance ratio for all components are :
 [9.31796975e-01 4.07650756e-02 2.18438668e-02 4.85133058e-03
 6.63341250e-04 6.87379176e-05 9.48168675e-06 1.09876995e-06
 8.50910960e-08 6.74428451e-09 4.72621857e-10 3.80161775e-11
 9.30609065e-12 8.71811396e-12 8.59362280e-12 8.53519254e-12
 8.41818231e-12 8.32407041e-12 8.16475927e-12 8.12446450e-12
 7.94366171e-12 7.84486451e-12 7.70300119e-12 7.56475430e-12
 7.49445649e-12 7.39336209e-12 7.21415235e-12 6.93802195e-12
 6.58019099e-12 6.26023985e-12]
```



From above, the first feature `SVENF01` has 0.93 varaince on the PCA model, which is a lot. And the second feature `SVENF02` has 0.04 explaiend variance ratio. Only the first feature are informative in this case. This is due to hifh correlation among attributes, which is verfied in heatmap.

Explained Variance Ratio for n_components=3

```
The explained variance ratio for 3 components are :
 [0.93179697 0.04076508 0.02184387]
```



If we fit the PCA model on the first three principal components, we will have the first PC has up to 93% explained variance ratio, and the rest two has very small explained varaince ratio. This corresponds to the PCA model we fit on the all components. The first PC is most informative and important.

The cumulative explained varaince will reach approximately to 1 after the 3 PC.

Part 3: Linear regression v. SVM regressor - baseline

Part 3.1: Linear Regression on Original Dataset

The Fitted Model

```
[  -4.83843828   53.15886154 -249.77609515  590.39765971 -686.96356431
  228.09887769  289.2937007  -302.62642323  -44.31624559  320.69207747
 -288.36927381  200.16366115   -0.89256855  -86.9401334   -96.64031266
   -7.50513023 -302.47703104  216.50764238  136.90241245  133.63875552
  562.97736489 -387.63320904  176.17955175 -418.55197044 -795.41172645
  238.76730551  102.69781344  839.17533861  -80.32795403 -336.50681657]
```

The above are the coefficients for this linear regression model. The first number -4.83843828 is our y-intercept when all x's are zero. The second number 53.15886154 are the coefficient for the explanatory variable `SVENF25` . And the rest numbers are coefficients for each features.

Performance Metrices by MSE

```
The MSE train: 0.603, test: 0.612
```

We can see that the Mean Squared Error (MSE) for the training set is 0.603, whilte the one for the testing set is 0.612, which is slightly bigger. That is acceptable.

Performance Metrices by Coefficient of Determination

```
The coefficient of determination train: 0.902, test: 0.904
```

The R^2 for both training set and test set are very high. The Linear regression model fits the original data very well.

Part 3.2: Linear Regression on PCA Transformed Dataset

```
[-0.42364406 -0.48646442  0.26559618]
```

The above are the coefficients for this linear regression model fitted on the PCA transformed data. The first number -0.42364406 is the coefficient for the first principal component. The second number -0.48646442 is the coefficient for the second principal component. And 0.26559618 is the coefficient for the third principal component

Performance Metrices by MSE

```
The MSE train: 0.820, test: 0.854
```

The Mean Squared Error (MSE) for the training set is 0.820, whilte the one for the testing set is 0.854, which is slightly bigger.

Performance Metrices by Coefficient of Determination

```
The coefficient of determination train: 0.867, test: 0.866
```

The R^2 for both training set and test set are high, around 0.87. The Linear regression model fits the original data very well. Also, the R^2 for the testing dataset is as the same high as the one for training dataset, so there is no overfitting or underfitting problems.

Part 3.3: SVM on Original Dataset

Performance Metrices by MSE

```
The MSE train: 0.071, test: 0.070
```

The MSE for both training and testing dataset are very small. They are much smaller than the ones for linear regression models.

Performance Metrices by Coefficient of Determination

```
The coefficient of determination train: 0.988, test: 0.989
```

Both R^2 are very closed to 1. And the R^2 for testing dataset is even bigger, which is very accurate for prediction.

Part 3.4: SVM on PCA Transformed Dataset

Performance Metrices by MSE

```
The MSE train: 0.136, test: 0.143
```

Performance Metrices by Coefficient of Determination

```
The coefficient of determination train: 0.978, test: 0.978
```

Part 4: Conclusions

Out[25]:

MSE	Experiment 1 (Treasury Yields)			
	Linear		SVM	
Baseline (all attributes)	Train Acc:	0.603	Train Acc:	0.071
	Test Acc:	0.612	Test Acc:	0.070
PCA transform (3 PCs)	Train Acc:	0.820	Train Acc:	0.136
	Test Acc:	0.854	Test Acc:	0.143

Choose MSE as our criteria:

On the untransformed data, SVM gives relatively lower MSE, so SVM is better for all attributes. On the PCA transformed data, the SVM also provides much lower MSE. Therefore, regardless transformed data or orginial data, SVM perfroms much better.

Out[26]:

R^2	Experiment 1 (Treasury Yields)			
	Linear		SVM	
Baseline (all attributes)	Train Acc:	0.902	Train Acc:	0.988
	Test Acc:	0.904	Test Acc:	0.989
PCA transform (3 PCs)	Train Acc:	0.867	Train Acc:	0.978
	Test Acc:	0.866	Test Acc:	0.978

Choose R^2 as our criteria:

On the untransformed data, SVM gives a bit higher R^2 than linear regression. However, both models gives the R^2 above 0.90, which means both models has fit the data very well. SVM is slight better. On the PCA transformed data, the SVM also provides a bit higher MSE. Therefore, overall SVM perfroms better.

None of the transformation leads to the best performance increase. This might be because the PCA tranformation lose some important information.

The SVM models has lower training time.

Part 5: Appendix

Link to github repo:

[https://github.com/yaxuanw3/IE517_F20_HW5 (https://github.com/yaxuanw3/IE517_F20_HW5)](https://github.com/yaxuanw3/IE517_F20_HW5)

```
My name is {Yaxuan Wang}
My NetID is: {662869931}
I hereby certify that I have read the University policy on Academic Integrity
and that I am not in violation.
```