Principle Components [PCA] and Stochastic Neighbor Embedding [t-SNE]

MSDS 411

Joel Riesen

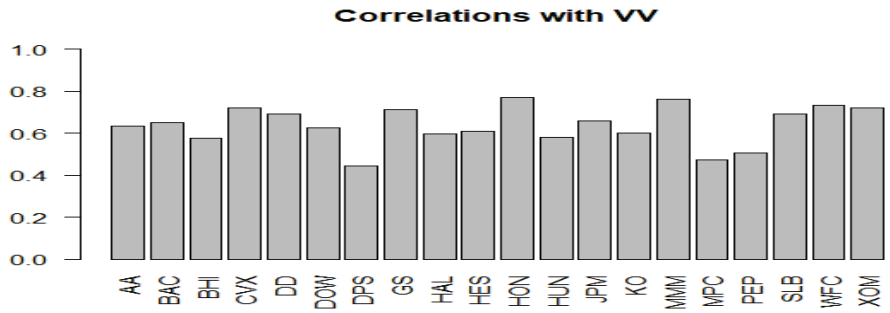
The Data:

The data for this report is the closing stock prices of 20 stocks and a large-cap index fund from Vanguard (ticker: VV). The dates used for the data set were 2012 to the end of 2013. The tickers are AA, BAC, BHI, CVX, DD, DOW, DPS, GS, HAL, HES, HON, HUN, JPM, KO, MMM, MPC, PEP, SLB, WFC, XOM. We will use the information from the individual stocks to explain the variations in the returns of the Vanguard index. The VV is designed to track the performance of the CRSP US Large Cap Index, a broadly diversified index of large US companies representing approximately the top 85% of the US market capitalization. (prospectus)

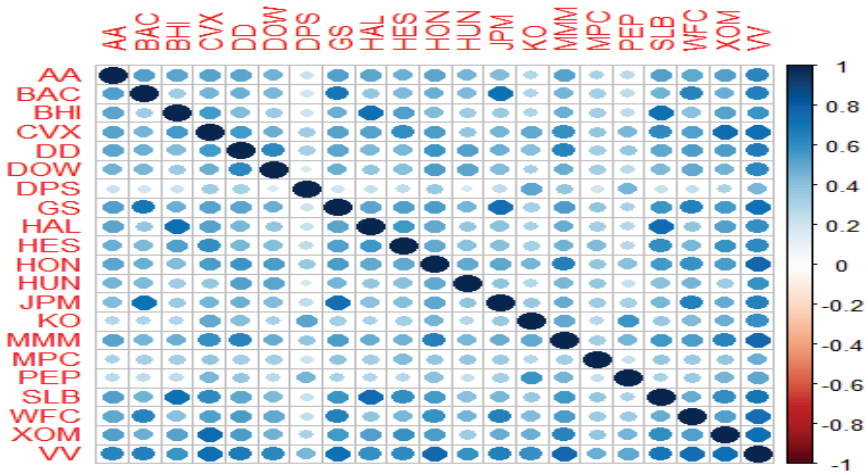
Correlations of the data:

Logging the returns of all the stocks to alleviate some of the noise/variations and making the returns similar. Looking at the returns vs. From the correlation plot it seems that the vast majority of the stocks are over 60%. With the highest correlated stock being MMM at 76% and the lowest being DPS at 44%.

Graph 1



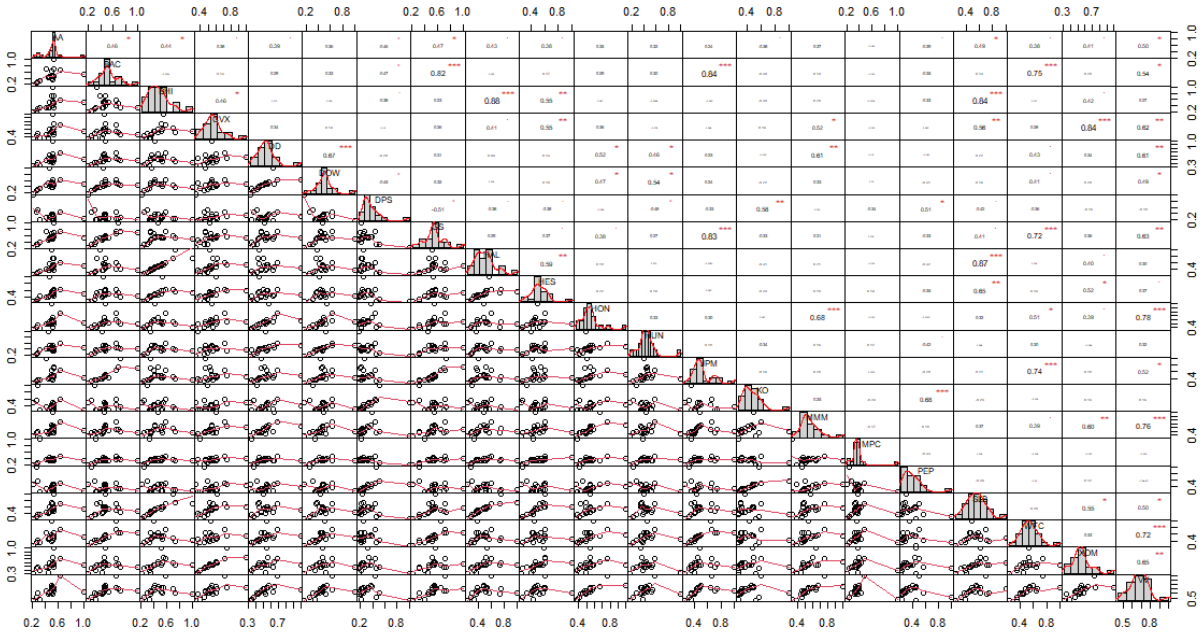
Graph 2



The Corrplot graph not only shows the relationship with VV but also with each other. Like taking a look at JPM, you can see that it's correlated with other financials such as BAC, WFC, and GS.

It uses the Performance Analytics package from R. The absolute value of the correlation and the results of the cor. Test as stars. On the bottom is the bivariate scatter plots with a fitted line. Again, it shows all the correlations, but the graph also shows how messy it is. It seems that BHI, HAL, and SLB are highly correlated with each other. (VV is in the bottom right corner)

Graph 3



Looking at the correlation graphs, stocks that look to have a low VIF are MPC, DPS, and PEP. The stocks that I would say have a high VIF would be HON, MMM, and WFC.

VIF

Since a lot of the stocks are in the same fields and are competitors, they are correlated. Additionally, VV uses the CRSP US Large Cap Index, and most of these companies are large-cap. I tried to get ahold of the data from 2012 to see what weightings these companies had, but I could not get it. Looking at the VIF's for the stocks, the more robust correlated stock in the financials and oil have the higher values. With a few being over three, we need to watch out for multicolarity.

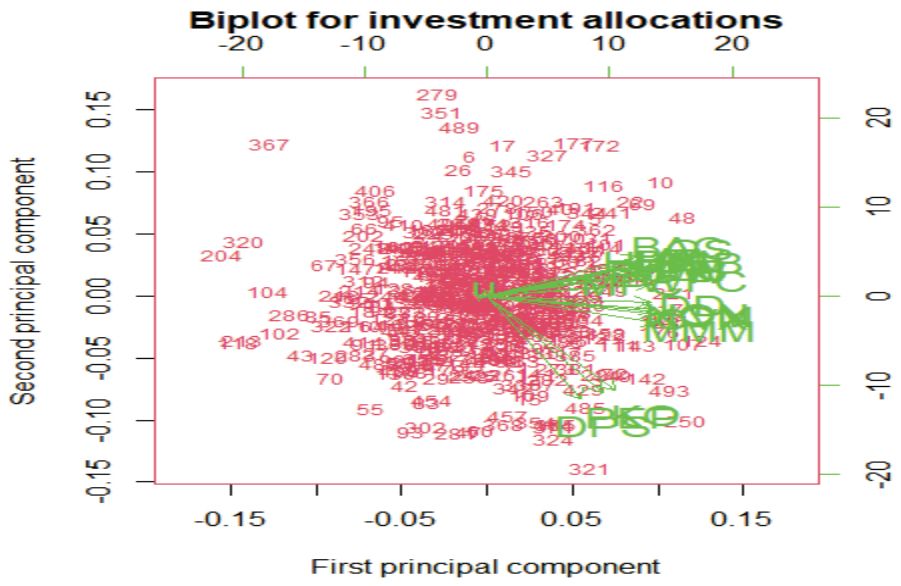
Table 1

BAC	GS	JPM	WFC	BHI	CVX	DD	DOW	DPS
2.558097	3.190808	2.844537	2.528808	2.603510	2.909686	2.432674	1.961953	1.524399
HAL	HES	HON	HUN	KO	MMM	MPC	PEP	SLB
2.902240	2.095666	2.447013	1.721319	1.967512	2.670404	1.376185	1.719788	3.257595
XOM								
2.924084								

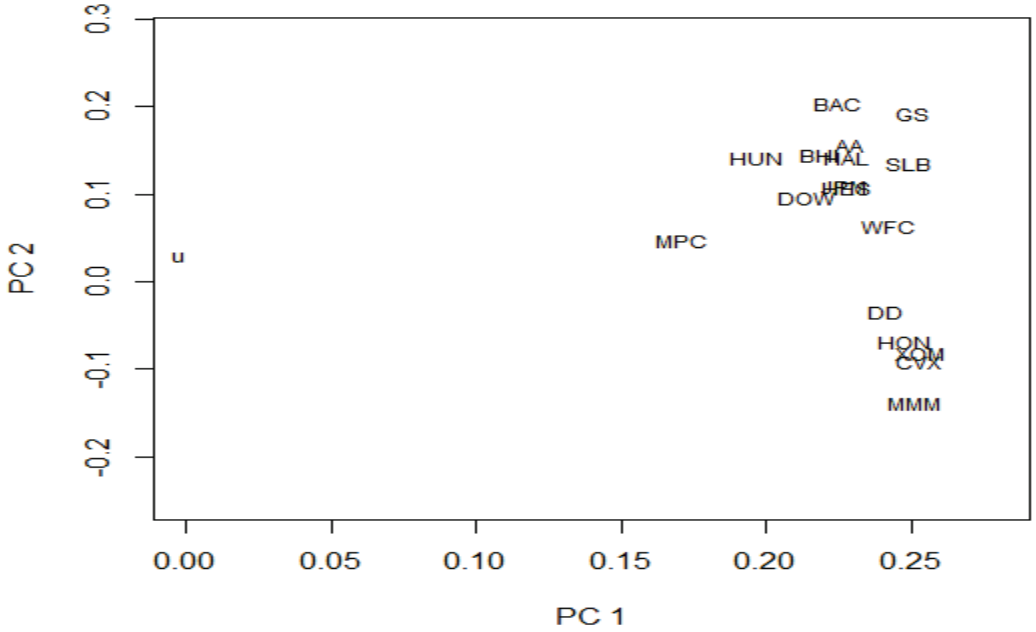
PCA to Help

Using the princomp function in R see the clusters of the stocks. No real surprises

Graph 4



Graph 5

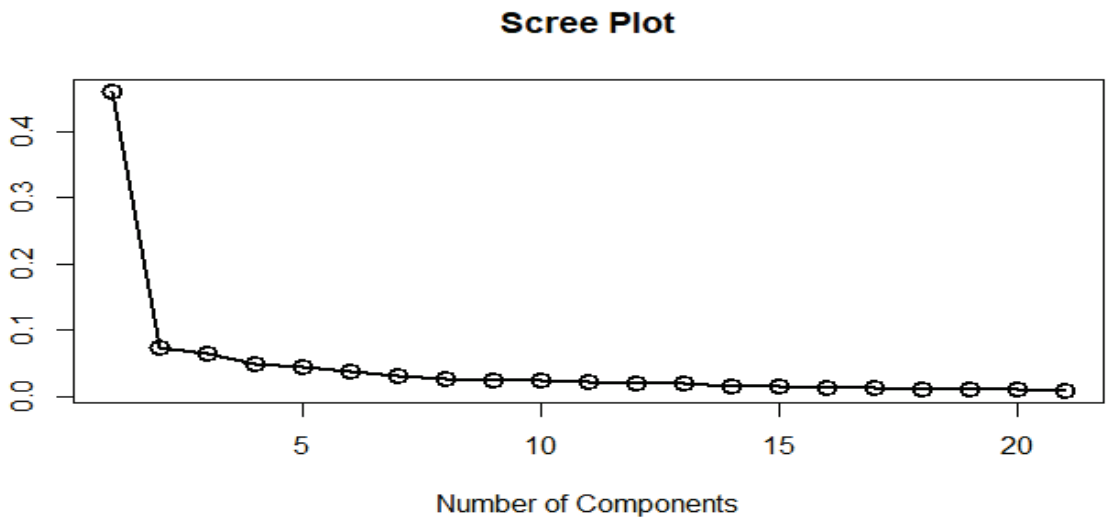


The second graph is the same but without all the background. This shows how specific stocks group cluster together

Reduction

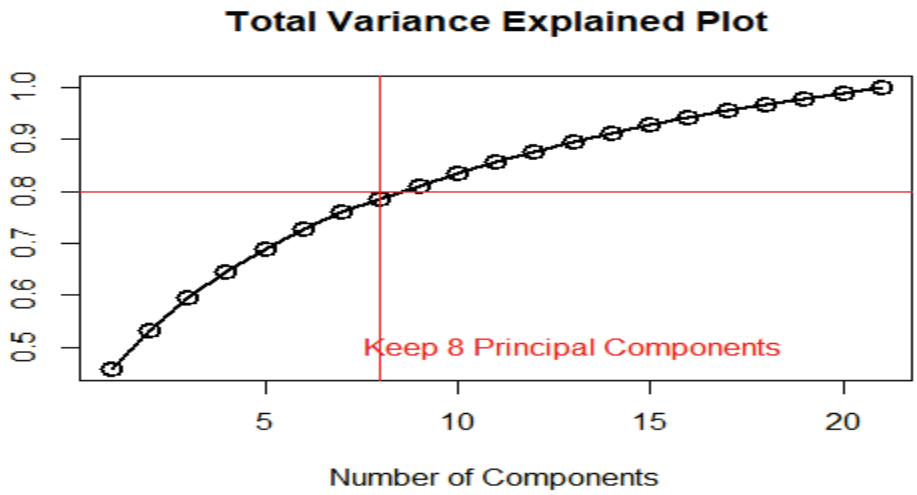
The princomp function uses the eigenvalues and eigenvectors for measuring the variance. Using that information, we should be able to narrow the stock list. The princomp(PC) loadings show the eigenvector information. We are looking for larger values which we can see using a scree plot

Graph 6



From this scree plot, you can tell that a few of them are very close to zero. We need to make these numbers "bigger," which we can do using the cumsum function, getting the comulative proportions of variance. From doing that, we get the following graph to help us. Showing us that we should keep 6 PC.

Graph 7



Predictive Modeling

Using the entire data set, we computed and scored the PC. Then we split the data into training and test set. Then we fit a linear model on the first eight principal components. When checking the model using the VIF function, it show all eight components are around one now since being fit.

Table 2

Comp.1	Comp.2	Comp.3	Comp.4	Comp.5	Comp.6	Comp.7	Comp.8
1.006147	1.007050	1.007509	1.015089	1.013446	1.009119	1.011988	1.014580

Comparing Models

We fit model1 using the tickers VV ~ GS+DD+DOW+HON+HUN+JPM+KO+MMM+XOM and for model2 we used all VV ~ BAC + GS + JPM + WFC + BHI + CVX + DD + DOW + DPS + HAL + HES + HON + HUN + KO + MMM + MPC + PEP + SLB + XOM. Then we also used the pca models as the third model.

The MAE for the three models are very similar

Table 3

model2.mae.test = 0.002240339,
model1.mae.test = 0.002345785
pca1.mae.test = 0.002273255

.

Since the MAE is the models' errors, I would have to go with the smallest number. Which, in this case, would be model2, which is the model that contains all the stocks. Model2 had more stocks helping to spread out the errors. The model pca1 using only eight stocks was only 0.0003 higher. This is why I would have to say that pca1 was more efficient.

Unsupervised to Supervised

With the information we gained and our selection of the eight components to keep, we can also verify our decisions with a supervised approach using variable selection. Using the AIC function, it chooses ten stocks to keep.

Table 4

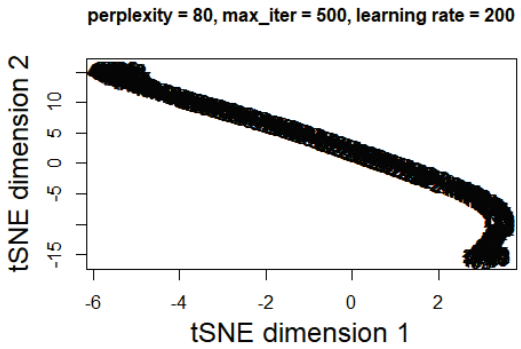
model2.mae.test = 0.002240339
model1.mae.test = 0.002345785
pca1.mae.test = 0.002273255
backward.mae.train = 0.001929819
backward.mae.test = 0.002214971

This information is showing the supervised model has a lower error rate. The train data set having under .002 is great.

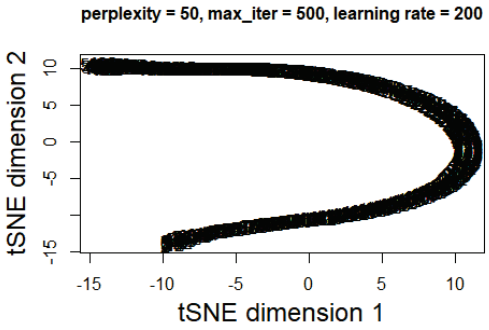
t – SNE

My R skills are lacking in trying this part of the excise. To get the columns associated with their respected industry is impossible for me. I have tried lists, group_by, assigning them directly, etc. Still no luck. I was able to train the dataset, and the plots look interesting. But not being able to see the grouping makes it hard to gauge.

Graph 8

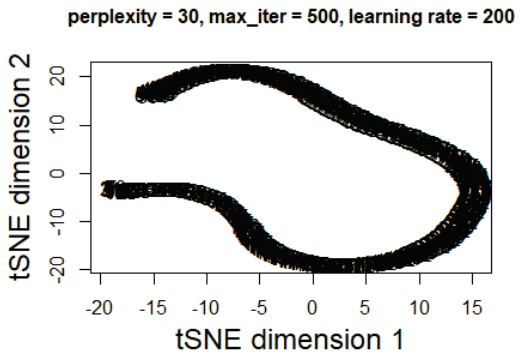


Graph 9



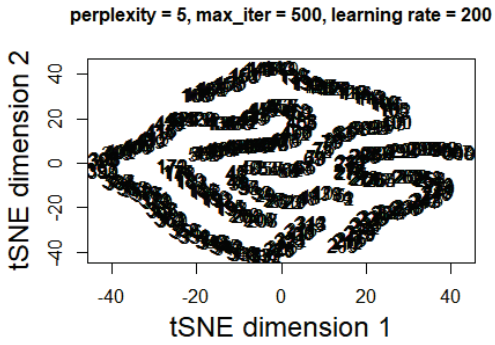
Looks like the continuation from the plot before.

Graph 10



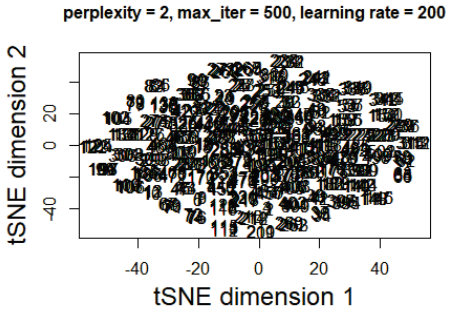
And even more.

Graph 11



I would like to see the grouping on this one.

Graph 12



<u>Reflection</u>

I like R, and my preference is R, but the vast majority of what I need and use in python. I do not have a math or coding background, and my day jobs have been very operational for the past 20 years. I like puzzles and fixing things. So I do not get a lot of opportunities to code using R or python at my job. Assignments like these that I find interesting get frustrating when I want to dig deeper or change one thing, and the models fall apart.

I have been involved in the investment world for many years. I wanted to dig into this; one of the biggest things that I was trying to show is the clustering of the stocks in the industry. Since this was a clone of the Large-cap CRSP index, it would have been interesting to see what percentage these stocks represented in the CRSP index during that year.