CISC 271 Assignment 2 Selena

Written Report

Singular Values:

Below are the first 10 singular values for both Z1 and Z2:

Z1

sdiag

30x1 double

	1
1	9.8126
2	7.5317
3	4.3269
4	3.9817
5	3.6210
6	2.9567
7	1.7536
8	1.5614
9	1.4806
10	1.3651

Z2

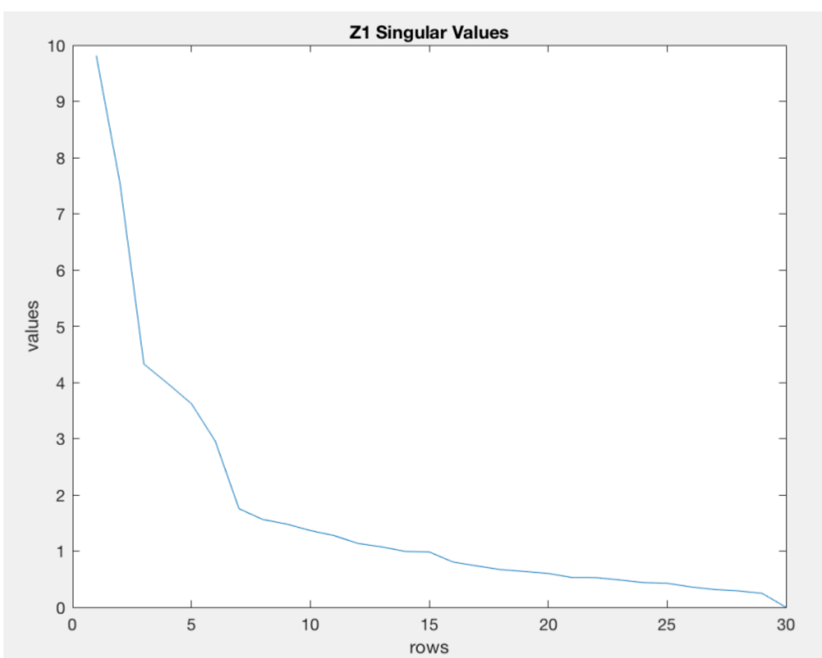
sdiag2

30x1 double

	1
1	6.1623
2	4.9679
3	3.1263
4	2.8470
5	2.1289
6	2.0803
7	1.6676
8	1.4541
9	1.3832
10	1.0824

Picking K Values:

To pick the cut-off for the first "k" components, I plotted the singular values for Z1 and Z2.



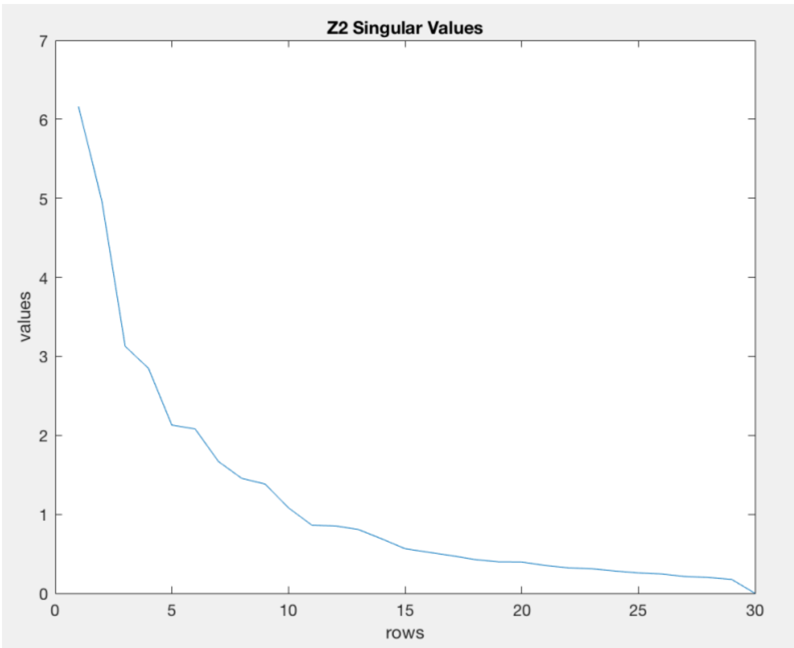
The cut-off “k”, should contain 50% to 60% of the eigenvalues. By looking at the graph, I noticed that most of the changing values were before rows 6-7, since the graph experienced a huge declined up to around rows 6-7 and then declined slowly afterwards.

```
%Sum of the singular values
S = sum(sdiag);

%Matrix with first k
C = sdiag(1:5,1);
Tk = sum(C);

%ratio
P = Tk/S
```

To be sure of this, I used the $p = t_k/t_n$ equation provided in the course notes. In other words, I used the sum of the first k rows (t_k) divided by the sum of the singular values (t_n). This gave me a value bigger than 0.60 or 60%, so I reduced it until the proportion of coverage (‘p’) reached a number between 0.5 and 0.6. In my case, k = 5 gave me p = 0.57.




```
%Sum of the singular values
S2 = sum(sdiag2);

%Matrix with first k
C2 = sdiag2(1:5,1);
Tk2 = sum(C2);

%ratio
P2 = Tk2/S2
```

For the k value in Z2, I did the same thing. By looking at this graph from the singular values of Z2 I noticed that like the one before, most of the changing values were before row 6-9. To check this, I also used the sum of the first 7 rows divided by the sum of the singular values of Z2. The number was greater than 60%. So I reduced k until I got a p value of 0.54.

Data Reconstruction:

```
new_data =Z1(:,a);
approxnum = 5;
[approxcomp, approxvec]=pcaapprox(new_data,approxnum,meanvec,uvecmat);
```

To reconstruct one of the given signals, I chose the first column of Z1 and used that as the new_data, which is the signal to be approximated as a column vector. I then used the k value I found for Z1 as the approxnum, the number of approximation.

```
[sdiag, meanvec, uvecmat] = pcaprelim(Z1);

[sdiag2, meanvec2, uvecmat2]=pcaprelim1(Z2);
```

By running the PCA, I got the meanvec and the uvecmat for both Z1 and Z2.

```
% Set up the initial and return values
diffvec = new_data - meanvec;
approxcomp = zeros(approxnum, 1);
approxvec =  meanvec;
```

In the given code, we first find the difference vector by subtracting a column of the original data by the mean vector.

```
% Loop through the eigenvectors, finding the components% and building the approximation
for i=1:approxnum
    uvec = uvecmat(:,i);
    beta = dot(diffvec, uvec);
    approxcomp(i,1) = beta;
    approxvec = approxvec + beta*uvec;
end
```

The uvec is equal to the first left singular column vector of the PCA.
Beta would be the dot product between the difference vector and the uvec that I just found.
Then the beta is multiplied with uvec and is added to the approxvec. By doing this approxnum ('k') number of times, we keep on adding values of beta*uvec to approxvec. As a result, approxvec would get more accurate and closer to the original data that we provided.

Then, I can use the approxvec and the original data to compare the errors between them.

Measuring Errors:

```
%compute error
  RMSE(a)=sqrt(mean((approxvec-new_data).^2));
```

To measure the error, I used the RMSE (root-mean-squared error) equation. This will measure the difference between the approxvec vector created and the original column (new_data) provided.
This gave me the RMSE for 1 column.

```
for a = 1:30
    new_data =Z1(:,a);
    approxnum = 5;
    [approxcomp, approxvec]=pcaapprox(new_data,approxnum,meanvec,uvecmat);

%compute error
    RMSE(a)=sqrt(mean((approxvec-new_data).^2));

end
```

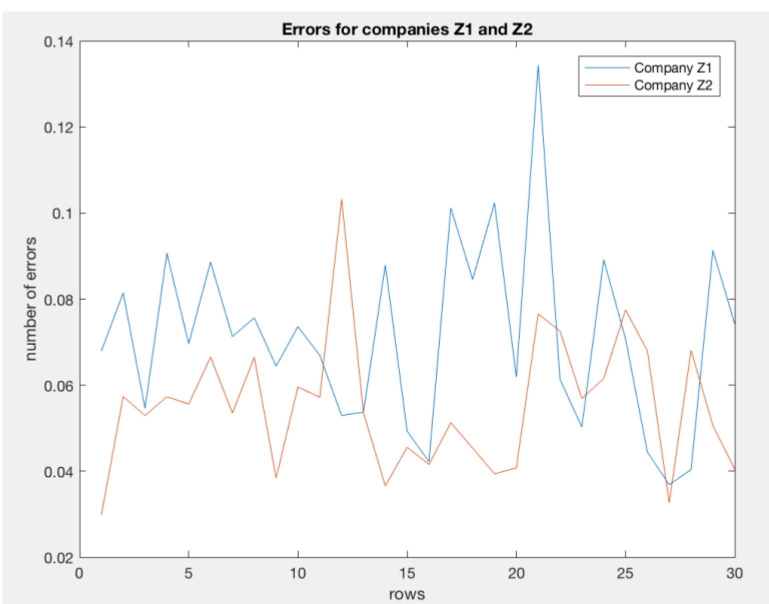
By looping through this section of code 30 times (the total number of column in Z1), I measured the error for each column and then put them into a RMSE matrix.

Analysis:

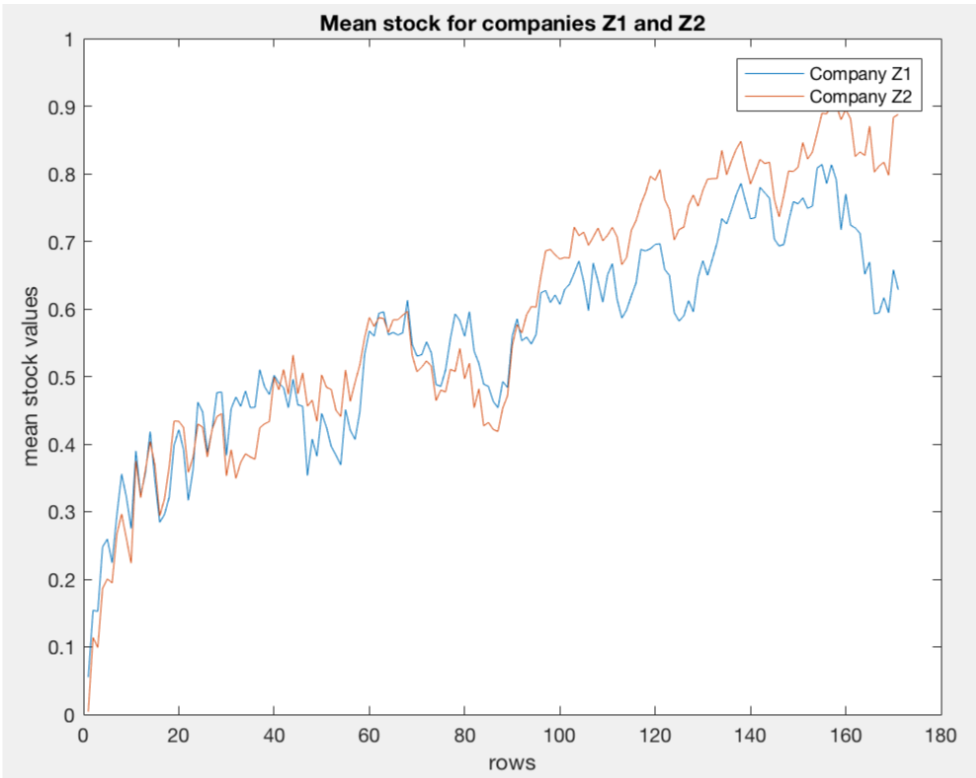
In this case, we have 2 group of companies. One group are relatively large companies and the other group have relatively small companies.

Large caps grow more slowly on average and tend to be less volatile, while small caps grow more quickly and are more volatile.

Volatility refers to the amount of uncertainty or risk about the size of changes in a security's value. In other words, the higher the volatility, there's a higher risk of security.



Based on the error graph of both companies, we can see that company Z1 has bigger number of errors rate than Company Z2. Since Company Z1 has more variance of error rates and higher error rates, Z1 is the smaller company. Company Z2 has less error rates and the errors have less variance, so Z2 is the bigger company.



Also, by looking at the mean stock values from the graph, we can see that Company Z2 has higher stock values than Company Z1. This can also be used to indicate that Company Z2 is the larger company and Company Z1 is the smaller company.