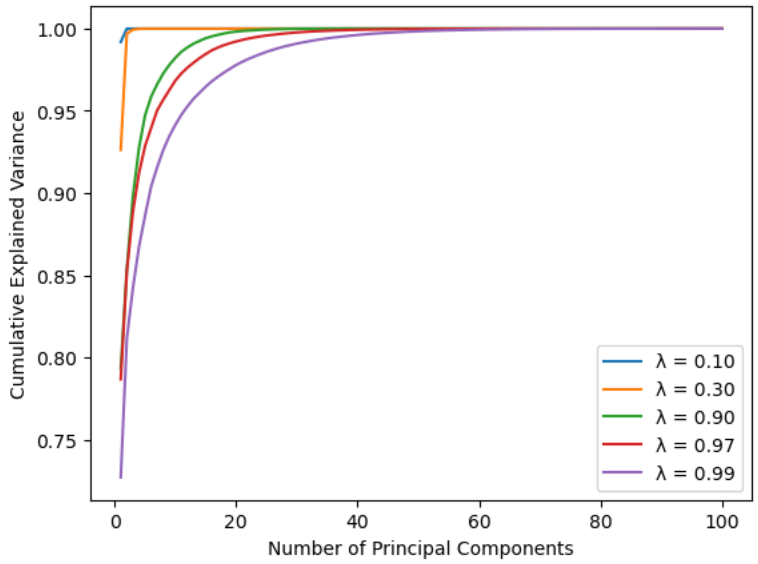
Problem 1:



A smaller Lambda means that older data has more influence on the today's prediction. Thus, the weight decay slower over time. A higher Lambda means that recent data is more heavily used. Thus, the weight decay faster over time.

Problem 2:

Run near_psd ( ), Higham's method the first time:

```
Smallest eigenvalues from near_psd: [-2.46516912e-13]
Smallest eigenvalues from Higham's method: [-0.03156765]
```

Even though the smallest eigenvalue for near_psd ( ) is negative, we can deem it as 0. However, as for Higham's method, the negative number is still significant so we should run the function for multiple times.

```
Smallest 10 eigenvalues from near_psd: [-2.46516912e-13]
Smallest 10 eigenvalues from Higham's method: [-0.00011566]
```

After running the function for multiple times. We got the smallest eigenvalue for Higham's method -0.00011566. Now it is not a significant negative number, and we can deem it as zero. We've confirmed that the matrix is PSD.

Set n = 500:

```
near_psd() runtime: 0.056738 seconds, Frobenius norm: 0.627523
Higham's method runtime: 12.467585 seconds, Frobenius norm: 0.045182
```

The time needed for Higham's method is significantly higher than that of near_psd. However, the Frobenius norm for Higham is smaller.

Set n = 100:

```
near_psd() runtime: 0.038334 seconds, Frobenius norm: 0.272792
Higham's method runtime: 0.603387 seconds, Frobenius norm: 0.043990
```

Though the time needed for Higham is much lower, it is still higher than that of near_psd. The advantage of each of the method is obvious. Though the Frobenius norms for near_psd is higher than that of Higham, near_psd used much shorter time. On the contrary, Higham used longer time, but its Frobenius norms are lower. If one needs quick approximation and can tolerate error, near_psd is a better choice.

Problem 3:

Frobenius results:

Frobenius Norm for direct_pearson_var: 0.0002440625532475145

Frobenius Norm for simulated_pca_50_pearson_var: 10.005871139196792

Frobenius Norm for simulated_pca_75_pearson_var: 4.800534897098526

Frobenius Norm for simulated_pca_100_pearson_var: 2.6163381891237396

Frobenius Norm for direct_pearson_exp_var: 0.0002502644100732166

Frobenius Norm for simulated_pca_50_pearson_exp_var: 10.0010926804298

Frobenius Norm for simulated_pca_75_pearson_exp_var: 4.4897351180556235

Frobenius Norm for simulated_pca_100_pearson_exp_var: 2.223677400974623

Frobenius Norm for direct_exp_corr_var: 0.0002345647192347571

Frobenius Norm for simulated_pca_50_exp_corr_var: 10.016292348037862

Frobenius Norm for simulated_pca_75_exp_corr_var: 3.7521447490490694

Frobenius Norm for simulated_pca_100_exp_corr_var: 1.9904318416598425

Frobenius Norm for direct_exp_corr_exp_var: 0.00020901965481918402

Frobenius Norm for simulated_pca_50_exp_corr_exp_var: 10.005763420205403

Frobenius Norm for simulated_pca_75_exp_corr_exp_var: 3.4418186344831416

Frobenius Norm for simulated_pca_100_exp_corr_exp_var: 1.9885215400068244

Running time:

Runtime for direct_pearson_var: 2.149153232574463 seconds

Runtime for pca_100_pearson_var: 1.8087151050567627 seconds

Runtime for pca_75_pearson_var: 1.75427508354187011 seconds

Runtime for pca_50_pearson_var: 0.285236120223999 seconds

Runtime for direct_pearson_exp_var: 2.110718011856079 seconds

Runtime for pca_100_pearson_exp_var: 1.8234829902648926 seconds

Runtime for pca_75_pearson_exp_var: 1.7464308738708496 seconds

Runtime for pca_50_pearson_exp_var: 0.35396790504455566 seconds

Runtime for direct_exp_corr_var: 2.096648931503296 seconds

Runtime for pca_100_exp_corr_var: 1.7884941101074219 seconds

Runtime for pca_75_exp_corr_var: 1.7541680335998535 seconds

Runtime for pca_50_exp_corr_var: 0.2511558532714844 seconds

Runtime for direct_exp_corr_exp_var: 2.1692512035369873 seconds

Runtime for pca_100_exp_corr_exp_var: 1.7924847602844238 seconds

Runtime for pca_75_exp_corr_exp_var: 1.786980152130127 seconds

Runtime for pca_50_exp_corr_exp_var: 0.4171438217163086 seconds

Findings:

The Frobenius Norm for direct method holds the lowest values but it requires the most amount of time to process. It is reasonable since the direct method does not reduce dimensions of the dataset. As for the PAC method. The Frobenius Norm for PCA with 100% Explained Variance is the lowest among all three, but it does require more time to process. Thus, one can easily observe that as the dimensionality of a dataset reduces, it requires less computation power but will sacrifice the accuracy.