Homework 3

Jacob Sachs

26 April 2013

Problem 1: Power Method Eigendecomposition

- **Accuracy Testing**
- **All Eigenvectors:**

Matrix: $\begin{pmatrix} 5 & -2 \\ -2 & 8 \end{pmatrix}$ U with $k = 2$: $\begin{pmatrix} -0.4472 & 0.8944 \\ 0.8944 & 0.4472 \end{pmatrix}$

Are these eigenvectors?

$$\begin{pmatrix} 5 & -2 \\ -2 & 8 \end{pmatrix} \begin{pmatrix} -0.4472 \\ 0.8944 \end{pmatrix} = \begin{pmatrix} -4.0248 \\ 8.0496 \end{pmatrix} = 9 * \begin{pmatrix} -0.4472 \\ 0.8944 \end{pmatrix}$$

$$\begin{pmatrix} 5 & -2 \\ -2 & 8 \end{pmatrix} \begin{pmatrix} 0.8944 \\ 0.4472 \end{pmatrix} = \begin{pmatrix} 3.5776 \\ 1.7888 \end{pmatrix} = 4 * \begin{pmatrix} 0.8944 \\ 0.4472 \end{pmatrix}$$

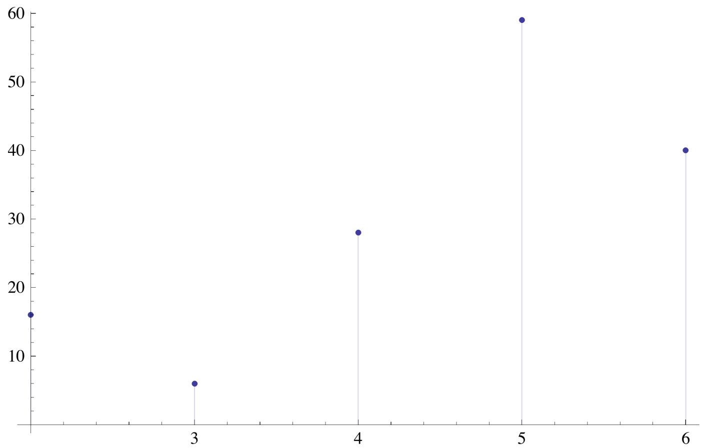
- **Bigger Matrices:**

$$\begin{pmatrix} 1 & 2 & 3 & 6 \\ 2 & 5 & 4 & -2 \\ 3 & 4 & 7 & 8 \\ 6 & -2 & 8 & -3 \end{pmatrix} \begin{pmatrix} 0.3958 \\ 0.2975 \\ 0.7510 \\ 0.4370 \end{pmatrix} = \begin{pmatrix} 5.8658 \\ 4.4091 \\ 11.1304 \\ 6.4768 \end{pmatrix} = 14.82 * \begin{pmatrix} 0.3958 \\ 0.2975 \\ 0.7510 \\ 0.4370 \end{pmatrix}$$

Complexity Testing

Matrix Size: $\epsilon = 0.00001$

$n = 2$: iterations = 16
$n = 3$: iterations = 6
$n = 4$: iterations = 28
$n = 5$: iterations = 59
$n = 6$: iterations = 40

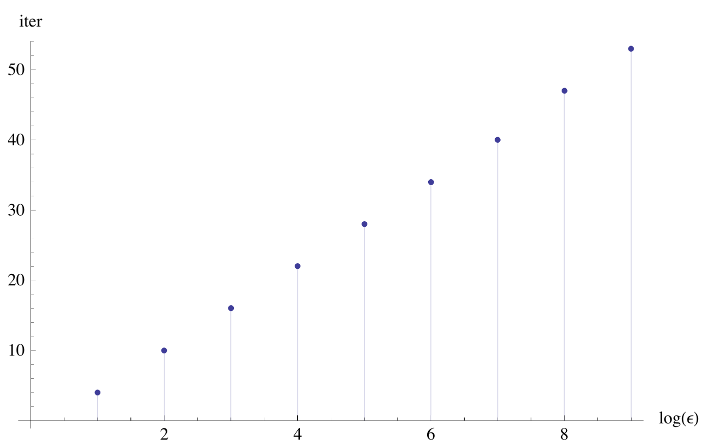


While the data isn't great, there appears to be a linear relation between iterations and matrix size. However, it could be n^2, since we do a matrix multiplication with every iteration, but this is a question related solely to iterations, not total complexity.

Epsilon: $n = 4$

$\epsilon = 0.1$: iterations = 4
$\epsilon = 0.01$: iterations = 10
$\epsilon = 0.001$: iterations = 16
$\epsilon = 0.0001$: iterations = 22
$\epsilon = 0.00001$: iterations = 28
$\epsilon = 0.000001$: iterations = 34
$\epsilon = 0.0000001$: iterations = 40
$\epsilon = 0.00000001$: iterations = 47
$\epsilon = 0.000000001$: iterations = 53

Out[4]=



My guess is that the number of iterations is $O(n \log(\epsilon))$

Problem 2: SVD and Frobenius Norm

▪ Code Test

Matrix: $\begin{pmatrix} 2 & 4 \\ 1 & 3 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$, $k = 2$, $\epsilon = 0.0000001$

$$U = \begin{pmatrix} 0.8174 & -0.576 \\ 0.576 & 0.8174 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, S = \begin{pmatrix} 5.465 & 0 \\ 0 & 0.366 \end{pmatrix}, V^T = \begin{pmatrix} 0.40455 & 0.9145 \\ 0.9145 & -0.40456 \end{pmatrix}$$

$$\text{Reconstructed } A = \begin{pmatrix} 1.61437 & 4.17044 \\ 1.54705 & 2.75767 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

▪ Frobenius Norm

k, ∈ = 0.000001	Frobenius Norm
1	43.4974624568
2	76.9731978598
3	73.3288842462
4	73.3288842469
5	73.3288499536

There does not appear to be any clear relation, though perhaps a linear one in a better example matrix.

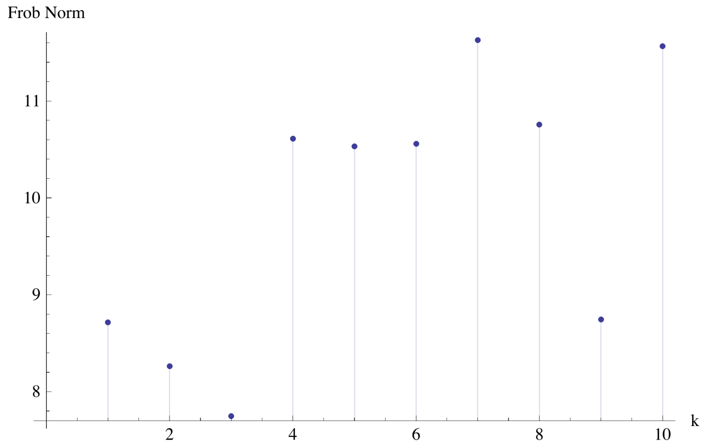
Problem 3: SVD with Missing Entries

▪ Function of *k*

$p = 0.9$

In[5]:=
```
ListPlot[{{1, 8.71542560456}, {2, 8.26163926493}, {3, 7.74686568125}, {4, 10.6093608423},
  {5, 10.530231617}, {6, 10.5577753014}, {7, 11.627392635}, {8, 10.7571033345},
  {9, 8.74500384174}, {10, 11.5636121233}}, Filling → Axis, AxesLabel → {"k", "Frob Norm"}]
```

Out[5]=

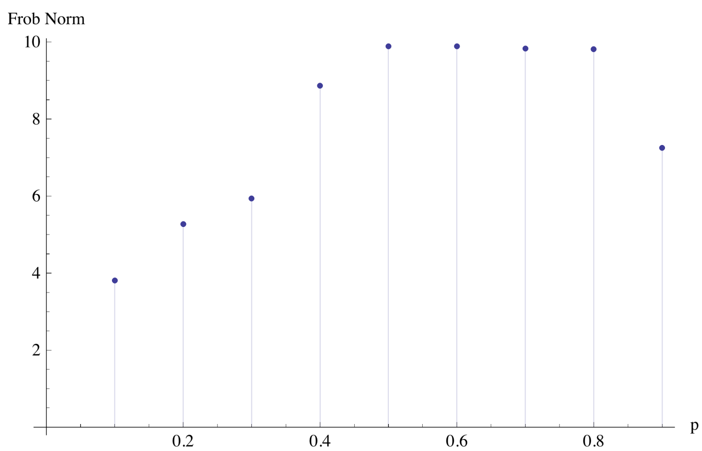


Function of *p*

$k = 4$

In[6]:=
```
ListPlot[{{.1, 3.80610291185}, {.2, 5.26913207621}, {.3, 5.94108022057},
  {.4, 8.86373438169}, {.5, 9.89066379246}, {.6, 9.88893903494}, {.7, 9.82577846779},
  {.8, 9.80918130332}, {.9, 7.24801256161}}, Filling → Axis, AxesLabel → {"p", "Frob Norm"}]
```

Out[6]=



Problem 4: Latent Semantic Analysis

Reconstruction Accuracy

As stated in the README, my code will not converge for the movie matrix.

Genre Prediction