ECE4721J - Homework 5

Methods and Tools for Big Data

Kexuan Huang 518370910126

July 20, 2022

Ex.1 Numerical stability

1. In the big data context, how beneficial would it be to increase precision? For instance what would be the gain of using double instead of float, or move on with multi-precision?[1]

The precision can be controlled to avoid accumulated errors. With a double, as the magnitude of the value increases, its precision decreases and this can introduce significant error into the result, which can increase the stability of methods in big data analysis.

2. Generate 100 random 1000×100 matrices X and measure the total time needed for MATLAB to compute:

MATLAB:

```
X = randi([0 100000], 1000, 100);

tic
svd(X);
toc

tic
svd(X');
toc

tic
eig(X * X');
toc

tic
eig(X' * X);
toc
```

Results:

```
Elapsed time is 0.00286484 seconds.
Elapsed time is 0.00371003 seconds.
Elapsed time is 0.0701489 seconds.
Elapsed time is 0.000877857 seconds.
```

[1] stackoverflow

3. Calculations

Let

$$X = \begin{pmatrix} -9 & 11 & -21 & 63 & -252 \\ 70 & -69 & 141 & -421 & 1684 \\ -575 & 575 & -1149 & 3451 & -13801 \\ 3891 & -3891 & 7782 & -23345 & 93365 \\ 1024 & -1024 & 2048 & -6144 & 24572 \end{pmatrix}$$

a) Use MATLAB to determine the eigenvalues of $X + \delta X$, where δX represents a small random perturbation on X. Study the variations over 1000 tests.

MATLAB:

```
X = [-9 11 -21 63 -252;
    70 -69 141 -421 1684;
    -575 575 -1149 3451 -13801;
    3891 -3891 7782 -23345 93365;
    1024 -1024 2048 -6144 24572];

err_eig = zeros(5, 1);
eig_X = eig(X);

for i = 1:1000
    dX = eps(X);
    err_eig = err_eig + eig(X + dX) - eig_X;
end

err_eig
```

Results:

```
err_eig =

   7.731538279038739e-01 - 2.561862243142342e+00i
   7.731538279038739e-01 + 2.561862243142342e+00i
   2.718376927461463e+00 + 0.000000000000000e+00i
  -2.132342289641740e+00 - 1.508562891852383e+00i
  -2.132342289641740e+00 + 1.508562891852383e+00i
```

b) Use MATLAB to determine the singular values of $X + \delta X$, where δX represents a small random perturbation on X. Study the variations over 1000 tests.

MATLAB:

```
sum_dX = zeros(5, 5);
err_sin = zeros(5, 1);
svd_X = svd(X);

for i = 1:1000
    dX = eps(X);
    err_sin = err_sin + svd(X + dX) - svd_X;
end

err_sin
```

Results:

```
err_sin =

   1.455191522836685e-08
   9.305889392408062e-10
   9.237055564881302e-11
   3.752553823233029e-11
   1.691492729682813e-10
```

4. In light of the lectures content, explain your observations.

SVD is much more stable than calculating eigenvalues, but might be a little slower to compute in MATLAB.

Ex.2 Simple SVD calculations

Without using a computer, find the singular values of the matrix

$$X = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 0 & 1 & 2 \end{pmatrix}.$$

$$X^T X = \begin{pmatrix} 1 & 5 & 9 \\ 2 & 6 & 0 \\ 3 & 7 & 1 \\ 4 & 8 & 2 \end{pmatrix} \times \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 0 & 1 & 2 \end{pmatrix} = \begin{pmatrix} 107 & 32 & 47 & 62 \\ 32 & 40 & 48 & 56 \\ 47 & 48 & 59 & 70 \\ 62 & 56 & 70 & 84 \end{pmatrix}$$

$$\det\left(X^T X - \lambda I\right) = \det \begin{pmatrix} 107-\lambda & 32 & 47 & 62 \\ 32 & 40-\lambda & 48 & 56 \\ 47 & 48 & 59-\lambda & 70 \\ 62 & 56 & 70 & 84-\lambda \end{pmatrix} = 0$$

$$\lambda^4 - 290\lambda^3 + 12840\lambda^2 - 9600\lambda = 0$$

$$\begin{cases} \lambda_1 \approx 0.76 \\ \lambda_2 \approx 53.54 \\ \lambda_3 \approx 235.70 \end{cases}$$

$$\begin{cases} \sigma_1 \approx 0.87 \\ \sigma_2 \approx 7.32 \\ \sigma_3 \approx 15.35 \end{cases}$$

Ex.3 PCA in Spark

Please refer to `ex3.py`

1. Explain how PCA can be of any help to Krystor?[2]

PCA:

- Provides the best "perspective" that emphasises similarities and differences in the data
- This new perspective combines the original "characteristics" in order to best summarize the data

As a result, PCA can help Krystor quickly targeting the columns x (sensors) which are most related to the last columns y (hourly electric consumption)

2. How many columns of sensors1.csv are necessary to explain 90% of the data? Let n be that number.

```
column 0: 0.56
column 1: 0.28
column 2: 0.15
```

So $n = 3$.

3. Construct the linear model $y = \sum_{i=1}^{n} \beta_i p_i + x_0 + \varepsilon$, where, $\varepsilon \sim \text{Normal}(0, 1)$, x_0 is the intercept at $x = 0$, and $(p_i)_{1 \le i \le n}$ are the first n principal components of sensors1.csv.

The linear model is constructed as follow:

```
Coefficient: [5.85535522 7.84716081 6.86019518]
Intercept: 734.0304003124222
R_square: 0.9999719057230781
```

[2] Lecture Slide

4. Help Krystor determine whether sensors2.csv also contains the output of the sensors in the electric circuit of Reapor Rich's new cinema.

```
column 0: 0.52
column 1: 0.31
column 2: 0.16
Coefficient: [5.87137867 7.82779001 6.8058421 ]
Intercept: 734.3335911223433
R_square: 0.999965257449195
```

Yes, we found column 1~3 as the principal columns with R square value of 0.99, so sensors2.csv also contains the output of the sensors.