COGS 109: Assignment #1

Due on Thursday, October 8, 2015

Tu, Zhuowen 2pm

Kyle Lee
A01614951

Problem 1

Find/define/design your problem and briefly describe
https : //snap.stanford.edu/data/amazon − meta.html
Do consumers care about reviews more on certain types of products? For example, do people look more at reviews on electronics over clothing brands? Are people more critical of certain types of products and how does it relate to the cost of the good? There are many problems we can derive from this dataset. With so much raw data, we can extrapolate many details such as customer trends, customer behavior in the decision making process due to each product having complementary goods as well as competitive goods.

Problem 2

Describe your data.
There are many dimensions of data to this dataset. For example, each item is divided into 6 characteristics such as product ID, Amazon ID Number, title, group, sales rank, similar products, product categories, and reviews. In the Reviews category, the data can be further broken down by the time of a review, total number of votes, total number of helpfulness votes, time, user id, and ratings. They were extracted from a subset of products collected from different categories through Amazon. The dataset is not split into training and testing. There is not a validation set. There are 548,552 products and over 7 million reviews.

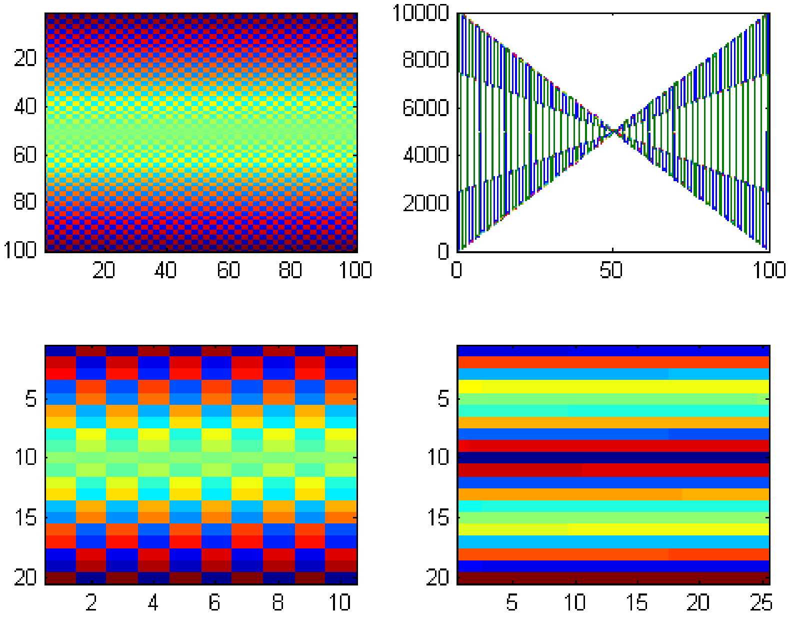
Problem 3

What kind of conclusion you plan to draw from this study?
I hypothesize that people will tend to look at the number of reviews over the number of helpful reviews. We can base this off the total number of reviews and see a correlation of helpful reviews as to being a considerable factor in one's purchasing decision. We can also pick up the trend of certain products over time and see how the introduction of the Kindle for example can affect purchases. Since music is included in the data set, we can also analyze the trend of music over 2004-2005.

Problem 4

What kind of techniques you expect to use to solve your problem?
I expect to be using heavy non-linear regression because there are so many products in one category. Also, k-clustering will be very useful in breaking down any product by grouping certain categories together. Linear regression can be involved as we can determine how the number of reviews can relate to the sales rank of the product. We can go so far as the bootstrap the data if my laptop cannot process all this data. We can also use some Hadoop-R optimization in order to look at big data in a more efficient scale.

Problem 5



```
% Initialize variables
jumps = 0;
% Start for loop
for i= 0:33
    a = 2+3*i;
    % If a doesnt get too big, add to jumps
    if a<100
        % Update
        jumps = jumps + a;
    end
end
jumps

%2
jumps2=[2:3:100]
total = sum(jumps2)

%3
popCanStock = 40;
controllerCount =2;
for iGamer = 1:5
    controllerCount = controllerCount+1;
    popCanStock = popCanStock - 1;
    for iAfternoon = 1:3
        popCanStock = popCanStock - 2;
    end
end

popCanStock
```

Problem 5 continued on next page. . .

```
controllerCount

%4
% Part a
myMagic = magic(100);
% Part b
subplot(2,2,1)
subplot(2,2,2)
subplot(2,2,3)
subplot(2,2,4)
colormap(jet)
% Part c
 subplot(2,2,1)
imagesc(myMagic)
% Part d
subplot(2,2,3)
imagesc(myMagic(5:5:100,10:10:100))
% Part e
subplot(2,2,2)
plot(myMagic)
% Part f
matrixA = myMagic(10:10:100,4:4:100);
matrixB = myMagic(10:10:100,2:4:100);
% Note that we concatenated vertically
matrixConcatenated = [matrixA; matrixB]
subplot(2,2,4);
imagesc(matrixConcatenated);
```

For part 1, printing out jumps will get:

```
jumps =

        1650
```

For part 2, printing out the total sum using vectors will get:

```
jumps2 =

  Columns 1 through 22

     2     5     8    11    14    17    20    23    26    29    32    35    38    41
44    47    50    53    56    59    62    65

  Columns 23 through 33

    68    71    74    77    80    83    86    89    92    95    98


total =

        1650
```

For part 3, printing out popCanStock and controllerCount will get:

```
popCanStock =

     5


controllerCount =

     7
```

For part 4a-4e, we look at the plot and the code above.
For part 4f, printing out the matrix concatenated with matrixA and matrixB vertically will get:

```
matrixConcatenated =

  Columns 1 through 10

         904         908         912         916         920         924         928
932          936         940
        8097        8093        8089        8085        8081        8077        8073
8069        8065        8061
        2904        2908        2912        2916        2920        2924        2928
2932        2936        2940
        6097        6093        6089        6085        6081        6077        6073
6069        6065        6061
        4904        4908        4912        4916        4920        4924        4928
4932        4936        4940
        4097        4093        4089        4085        4081        4077        4073
4069        4065        4061
        6904        6908        6912        6916        6920        6924        6928
6932        6936        6940
        2097        2093        2089        2085        2081        2077        2073
2069        2065        2061
        8904        8908        8912        8916        8920        8924        8928
8932        8936        8940
          97          93          89          85          81          77          73
69            65          61
        9099        9095        9091        9087        9083        9079        9075
9071        9067        9063
        1902        1906        1910        1914        1918        1922        1926
1930        1934        1938
        7099        7095        7091        7087        7083        7079        7075
7071        7067        7063
        3902        3906        3910        3914        3918        3922        3926
3930        3934        3938
        5099        5095        5091        5087        5083        5079        5075
5071        5067        5063
        5902        5906        5910        5914        5918        5922        5926
5930        5934        5938
        3099        3095        3091        3087        3083        3079        3075
3071        3067        3063
        7902        7906        7910        7914        7918        7922        7926
7930        7934        7938
        1099        1095        1091        1087        1083        1079        1075
1071        1067        1063
        9902        9906        9910        9914        9918        9922        9926
9930        9934        9938
```

```

  Columns 11 through 20

         944         948         952         956         960         964         968
972         976         980
        8057        8053        8049        8045        8041        8037        8033
8029        8025        8021
        2944        2948        2952        2956        2960        2964        2968
2972        2976        2980
        6057        6053        6049        6045        6041        6037        6033
6029        6025        6021
        4944        4948        4952        4956        4960        4964        4968
4972        4976        4980
        4057        4053        4049        4045        4041        4037        4033
4029        4025        4021
        6944        6948        6952        6956        6960        6964        6968
6972        6976        6980
        2057        2053        2049        2045        2041        2037        2033
2029        2025        2021
        8944        8948        8952        8956        8960        8964        8968
8972        8976        8980
          57          53          49          45          41          37          33
29          25          21
        9059        9055        9051        9047        9043        9039        9035
9031        9027        9023
        1942        1946        1950        1954        1958        1962        1966
1970        1974        1978
        7059        7055        7051        7047        7043        7039        7035
7031        7027        7023
        3942        3946        3950        3954        3958        3962        3966
3970        3974        3978
        5059        5055        5051        5047        5043        5039        5035
5031        5027        5023
        5942        5946        5950        5954        5958        5962        5966
5970        5974        5978
        3059        3055        3051        3047        3043        3039        3035
3031        3027        3023
        7942        7946        7950        7954        7958        7962        7966
7970        7974        7978
        1059        1055        1051        1047        1043        1039        1035
1031        1027        1023
        9942        9946        9950        9954        9958        9962        9966
9970        9974        9978

  Columns 21 through 25

         984         988         992         996        1000
        8017        8013        8009        8005        8001
        2984        2988        2992        2996        3000
        6017        6013        6009        6005        6001
        4984        4988        4992        4996        5000
        4017        4013        4009        4005        4001
        6984        6988        6992        6996        7000
```

Problem 5 continued on next page. . .

```
        2017        2013        2009        2005        2001
        8984        8988        8992        8996        9000
          17          13           9           5           1
        9019        9015        9011        9007        9003
        1982        1986        1990        1994        1998
        7019        7015        7011        7007        7003
        3982        3986        3990        3994        3998
        5019        5015        5011        5007        5003
        5982        5986        5990        5994        5998
        3019        3015        3011        3007        3003
        7982        7986        7990        7994        7998
        1019        1015        1011        1007        1003
        9982        9986        9990        9994        9998
```

Note that formatting from MatLab to LATEX will cause the concatenated matrix to look messy. For further verification, I posted below the values of matrixA and matrixB:

```
>> matrixA

matrixA =

  Columns 1 through 10

         904         908         912         916         920         924         928
932          936         940
        8097        8093        8089        8085        8081        8077        8073
8069        8065        8061
        2904        2908        2912        2916        2920        2924        2928
2932        2936        2940
        6097        6093        6089        6085        6081        6077        6073
6069        6065        6061
        4904        4908        4912        4916        4920        4924        4928
4932        4936        4940
        4097        4093        4089        4085        4081        4077        4073
4069        4065        4061
        6904        6908        6912        6916        6920        6924        6928
6932        6936        6940
        2097        2093        2089        2085        2081        2077        2073
2069        2065        2061
        8904        8908        8912        8916        8920        8924        8928
8932        8936        8940
          97          93          89          85          81          77          73
69          65          61

  Columns 11 through 20

         944         948         952         956         960         964         968
972          976         980
        8057        8053        8049        8045        8041        8037        8033
8029        8025        8021
        2944        2948        2952        2956        2960        2964        2968
2972        2976        2980
        6057        6053        6049        6045        6041        6037        6033
6029        6025        6021
```

```
        4944        4948        4952        4956        4960        4964        4968
4972        4976        4980
        4057        4053        4049        4045        4041        4037        4033
4029        4025        4021
        6944        6948        6952        6956        6960        6964        6968
6972        6976        6980
        2057        2053        2049        2045        2041        2037        2033
2029        2025        2021
        8944        8948        8952        8956        8960        8964        8968
8972        8976        8980
          57          53          49          45          41          37          33
29          25          21

  Columns 21 through 25

         984         988         992         996        1000
        8017        8013        8009        8005        8001
        2984        2988        2992        2996        3000
        6017        6013        6009        6005        6001
        4984        4988        4992        4996        5000
        4017        4013        4009        4005        4001
        6984        6988        6992        6996        7000
        2017        2013        2009        2005        2001
        8984        8988        8992        8996        9000
          17          13           9           5           1

>> matrixB

matrixB =

  Columns 1 through 10

        9099        9095        9091        9087        9083        9079        9075
9071        9067        9063
        1902        1906        1910        1914        1918        1922        1926
1930        1934        1938
        7099        7095        7091        7087        7083        7079        7075
7071        7067        7063
        3902        3906        3910        3914        3918        3922        3926
3930        3934        3938
        5099        5095        5091        5087        5083        5079        5075
5071        5067        5063
        5902        5906        5910        5914        5918        5922        5926
5930        5934        5938
        3099        3095        3091        3087        3083        3079        3075
3071        3067        3063
        7902        7906        7910        7914        7918        7922        7926
7930        7934        7938
        1099        1095        1091        1087        1083        1079        1075
1071        1067        1063
        9902        9906        9910        9914        9918        9922        9926
9930        9934        9938
```

Problem 5 continued on next page...

```
 Columns 11 through 20

    9059  9055  9051  9047  9043  9039  9035
9031  9027  9023
    1942  1946  1950  1954  1958  1962  1966
1970  1974  1978
    7059  7055  7051  7047  7043  7039  7035
7031  7027  7023
    3942  3946  3950  3954  3958  3962  3966
3970  3974  3978
    5059  5055  5051  5047  5043  5039  5035
5031  5027  5023
    5942  5946  5950  5954  5958  5962  5966
5970  5974  5978
    3059  3055  3051  3047  3043  3039  3035
3031  3027  3023
    7942  7946  7950  7954  7958  7962  7966
7970  7974  7978
    1059  1055  1051  1047  1043  1039  1035
1031  1027  1023
    9942  9946  9950  9954  9958  9962  9966
9970  9974  9978

 Columns 21 through 25

    9019  9015  9011  9007  9003
    1982  1986  1990  1994  1998
    7019  7015  7011  7007  7003
    3982  3986  3990  3994  3998
    5019  5015  5011  5007  5003
    5982  5986  5990  5994  5998
    3019  3015  3011  3007  3003
    7982  7986  7990  7994  7998
    1019  1015  1011  1007  1003
    9982  9986  9990  9994  9998
```