CS540 Spring 2022 Homework 5

Linear Regression on Lake Mendota Ice

The Wisconsin State Climatology Office keeps a record on the number of days Lake Mendota was covered by ice at http://www.aos.wisc.edu/~sco/lakes/Mendota-ice.html.

1 Question 1: Data Curation

As with any real problems, the data is not as clean or as organized as one would like for machine learning. Curate a clean data set starting from year 1855-56 and ending in year 2020-21. We care about the following aspects of the data:

- x, the starting year: for 1855-56, $x = 1855$; for 2017-18, $x = 2017$; and so on.
- y, the number of ice days in that year. For Mendota in 1855-56, $y = 118$; for 2017-18, $y = 94$; and so on.

Some years have multiple freeze thaw cycles such as 2001-02; you should use the aggregated number of days. In the table, this appears as a year followed by a " under it. You'll notice exactly one of the two lines has a number for ice days. That number is the one you'll use as the y for that year. For example, in year 2001-02 your feature will be: $x = 2001, y = 21$. Save your data set as "hw5.csv". We gave you an example toy.csv with the correct format (but the numbers are fake). Your file should follow this standard format for ".csv" files. For example, the first 4 lines of your 'hw5.csv" would be the following:

```
year,days
1855,118
1856,151
1857,121
```

Output: Create (possibly manually) a file named: "hw5.csv", as described above.

2 Question 2: Visualize Data

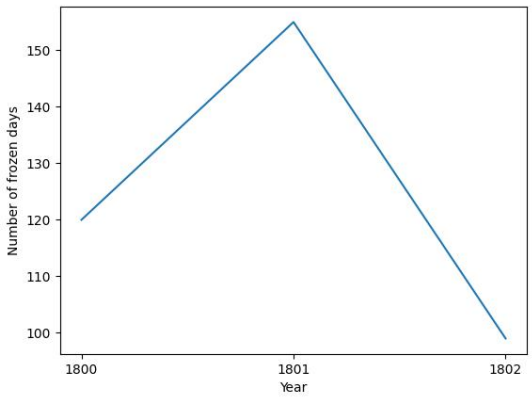
For this and the following questions, you need to write a python program hw5.py. Your code should take as argument the name of the csv file that you want to read (eg. toy.csv or hw5.csv). To get the first argument you can use the following code:

```
import sys
sys.argv[1] #this contains the first argument as string
```

Your hw5.py then needs to produce the outputs in the order described in Question 2 – Question 6.
Plot year vs. ice days from your data set. You should save the plot as a *plot.jpg* . You can save the plot by using the following code:

```
plt.savefig("plot.jpg")
```

For reference, we gave you the output plot.jpg for toy.csv:



Note: You do not need to fully match the plot style, but should have x-axis labels, y-axis labels, and the curve.

Output: Your "hw5.py" would need to produce a "plot.jpg" file.

3 Question 3: Linear Regression

Using the whole data set as the training set, train a linear regression model:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x.$$

Recall, this means finding the closed-form MLE solution for $\beta = (\beta_0, \beta_1)^\top$:

$$\hat{\beta} = \arg\min_{\beta} \frac{1}{n} \sum_{i=1}^{n} (x_i^\top \beta - y_i)^2.$$

To write the solution in matrix form, we first augment the feature vector:

$$\boldsymbol{x}_i = \begin{pmatrix} 1 \\ x_i \end{pmatrix}.$$

Note boldface $\boldsymbol{x}_i$ is a vector while x_i is a scalar. Now organize the features in a $n \times 2$ array, where n is the number of data points (roughly the number of rows in your computed csv).

$$X = \begin{pmatrix} \boldsymbol{x}_1^\top \\ \vdots \\ \boldsymbol{x}_n^\top \end{pmatrix}.$$

And the y values into a vector:

$$Y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}.$$

The MLE solution can be written as

$$\hat{\beta} = \arg\min_{\beta} \|X\beta - Y\|^2.$$

By setting the gradient to zero, we arrive at the closed-form MLE:

$$\hat{\beta} = \left(X^\top X\right)^{-1} X^\top Y.$$

This involves the inverse of $X^\top X$, which for our problem is invertible. Your program should compute $\hat{\beta}$ as specified here. You should break down this process into several steps as follows:

3.1 Q3a:

Represent the data as a matrix X, which will have dimension $n \times 2$. Recall that for each individual data point x_i, you should transform the point into a full feature vector

$$\boldsymbol{x}_i = \begin{pmatrix} 1 \\ x_i \end{pmatrix}$$

Then, you make each feature vector become a row of the overall data matrix X:

$$X = \begin{pmatrix} \boldsymbol{x}_1^\top \\ \vdots \\ \boldsymbol{x}_n^\top \end{pmatrix}$$

Your output for this section is formed by printing out X as follows:

```
print("Q3a:")
print(X)
```

3.2 Q3b:

Next, you need to place all the corresponding y_i values into a vector

$$Y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}$$

Your output for this section is formed by printing out Y as follows:

```
print("Q3b:")
print(Y)
```

3.3 Q3c:

Next, compute the matrix product $Z = X^T X$.
Your output for this section is formed by printing out Z as follows:

```
print("Q3c:")
print(Z)
```

3.4 Q3d:

Next, compute the inverse of $X^T X$, which we call I.
Your output for this section is formed by printing out I as follows:

```
print("Q3d:")
print(I)
```

3.5 Q3e:

Next, compute what we call the *pseudo-inverse* of X, which we call PI. Mathematically, $PI = (X^T X)^{-1} X^T$.
Your output for this section is formed by printing out PI as follows:

```
print("Q3e:")
print(PI)
```

3.6 Q3f:

Lastly, compute $\hat{\beta}$ using the results from the previous parts. Recall,

$$\hat{\beta} = \left(X^\top X\right)^{-1} X^\top Y.$$

Your output for this section is formed by printing out PI as follows:

```
print("Q3f:")
print(hat_beta)
```

Q3 Summary

Output: Your program should output the matrices $X, Y, X^\top X, (X^\top X)^{-1}, (X^\top X)^{-1}X^\top$ and β. If you do each part correctly, your code will print out the answers in the following format, where each variable will be replaced with the actual computed value:
Q3a:
X
Q3b:
Y
Q3c:
$X^\top X$
Q3d:
$(X^\top X)^{-1}$
Q3e:
$(X^\top X)^{-1}X^\top$
Q3f:
$\hat{\beta}$

We gave you the complete sample output for toy.csv at the end of this file. You should test your code on this file.

4 Question 4: Prediction

Using your $\hat{\beta}$, predict the number of ice days for this winter 2021-22. Equivalently, we have a test item $x_{test} = 2021$ and you should predict

$$\hat{y}_{test} = \hat{\beta}_0 + \hat{\beta}_1 x_{test}.$$

The Wisconsin State Climatology Office is likely to release the official 2021-22 ice days sometime in March 2022; at that time you may want to check how close your prediction was.

Output: Print the following similar to in Q3 but without the new line:
Q4: $\hat{y}_{test}$ You can do this with the following code:

```
print("Q4: " + str(y_test))
```

You will use the same formatting to print the answers to the remaining questions

5 Question 5: Model Interpretation

(a) What is the sign of your $\hat{\beta}_1$? Print a symbol where the symbol should be either $>, <, =$ depending if the sign is positive, negative or zero.
(b) Interpret, in English, the meaning of this sign for Mendota ice. Print a short answer explanation

Output: Your program should print the following lines:
Q5a: Symbol
Q5b: Short Answer

6 Question 6: Model Limitation

(a) Given your MLE $\hat{\beta}$, predict the year x^* by which Lake Mendota will no longer freeze. That is,

$$0 = \hat{\beta}_0 + \hat{\beta}_1 x^*.$$

Note x^* will in general be a real number instead of an integer.
(b) Discuss whether x^* is a compelling prediction based on the trends in the data, and why.

Output: Your program should print the following lines:
Q6a: x^*
Q6b: Answer
Where Answer should be replaced with your answer for part b.

Summary

You will create two files named **hw5.csv** and **hw5.py** that you will zip into one zip file `hw5_<netid>.zip`.

hw5.csv This will be a csv file containing two column headers: "year" and "days" in that order. Then, each row will contain the corresponding x and y defined in Question 1 in the same order as they appear in the original Mendota data set starting with year $1855 - 56$.

hw5.py This will be a python file that produces a plot for Q2 (save it do not plt.show() it!) and then that prints out the answers to the following questions:

Q3a: Compute X

Q3b: Compute Y

Q3c: Compute $X^T X$

Q3d: Compute $(X^T X)^{-1}$

Q3e: Compute $(X^T X)^{-1} X^T$

Q3f: Compute $\hat{\beta}$

Q4: Compute a prediction for year $x_{test} = 2021$ using your linear regression model.

Q5a: Compute the sign of $\hat{\beta}_1$

Q5b: Explain what this sign could mean.

Q6a: Solve the equation $0 = \hat{\beta}_0 + \hat{\beta}_1 x^*$ for x^*.

Q6b: Discuss whether this x^* makes sense given what we see in the data trends.

Submission Details

- Please submit your files in a zip file named `hw5_<netid>.zip`
- Inside your zip file, there should be have two files named **hw5.csv** and **hw5.py**.
- All code should be contained in functions or under a `if __name__=="__main__":`
- Be sure to remove all debugging output before submission.

Example Input/Output

toy.csv:

```
year,days
1800,120
1801,155
1802,99
```

Run parameters

```
./python3 hw5.py toy.csv
```

Output: (note plot.jpg should be produced but is not shown here)

```
Q3a:
[[   1 1800]
[   1 1801]
[   1 1802]]
Q3b:
[120 155  99]
Q3c:
[[      3    5403]
[   5403 9730805]]
Q3d:
[[ 1.62180083e+06 -9.00500000e+02]
[-9.00500000e+02  5.00000000e-01]]
Q3e:
[[ 9.00833334e+02  3.33333333e-01 -9.00166667e+02]
[-5.00000000e-01  0.00000000e+00  5.00000000e-01]]
Q3f:
[ 1.90351667e+04 -1.05000000e+01]
Q4: -2185.3333340429162
Q5a: <
Q5b: Answer
Q6a: 1812.8730158716883
Q6b: Answer
```