CS251 HW 5 | Mon Mar 11, 2019 | Week 6

Name:

Notes:

- Feel free to work with a friend on this assignment. However, be sure that you understand all the steps.
- The plots that you create should be a good test to make sure that your code is correct.
- Please turn in your code, along with your answers to the following questions, and screenshots of your plots.

Question 1

a) Write code to determine linear regression coefficients for the data points (1, 1), (2, 2), (4, 2), and (5, 3) using the line equation $y = c_0 x + c_1$ model. **Use the normal equations directly, NOT `scipy.stats.linregress` or another high-level function.** The function for the matrix inverse `np.linalg.inv` is helpful.

b) Write code to compute the R^2 value **from scratch** (i.e. do numpy mathematical operations on your A, c, y, don't rely on a library function specifically designed to give you an R^2 value). Is the quality of the fit good or bad?

c) Create a plot using `pyplot` in `matplotlib` with:

- an informative title
- x and y axes labels ('x' and 'y' labels are fine)
- the data points
- the regression curve

See code from class and appendix below for tips.

Question 2

a) Explicitly write out the independent variable matrix A and dependent variable vector y corresponding to the following data fit with the linear regression model $y = \frac{c_0}{x} + c_1 x$. *Note: this model does NOT contain an intercept term.*

x	y
0.955	5.722
1.380	4.812
1.854	4.727
2.093	4.850
2.674	5.011
3.006	5.253
3.255	5.617
3.940	6.282
4.060	6.255

b) Write code to determine the regression coefficients. **Use the normal equations directly, NOT `scipy.stats.linregress` or another high-level function.** The function for the matrix inverse `np.linalg.inv` is helpful.

c) Write code to compute the R^2 value **from scratch** (i.e. do numpy mathematical operations on your A, c, y, don't rely on a library function specifically designed to give you an R^2 value). Is the quality of the fit good or bad?

d) Create a plot using `pyplot` in `matplotlib` with:

- an informative title
- x and y axes labels ('x' and 'y' labels are fine)
- the data points
- the regression curve

Appendix: Plotting tips

- Lab machines in Davis 102 should have `matplotlib` installed. If you want to install it on your own machine and you followed the instructions on the CS251 website to install `numpy` and `scipy`, all you should need to do (on Mac) is run `python3 -m pip install matplotlib` in the Terminal.
- Use `lecture_13_bayesian_updating.py` as your template to plot data.
- `plt.plot(xVals, yVals, 'o')` will plot the (x, y) ordered pairs without connecting them with a continuous curve.
- `plt.plot(xVals, yVals)` will connect the points with a curve (NOT the regression curve!).
- `plt.show()` keeps the plot window up (like `turtle.exitonclick()` for x-CS151ers)
- Issuing two `plt.plot` commands before `plt.show()` will superimpose elements from the two plots into a single plot.