CSci 343 Fundamentals of Data Science
Challenge 5

Submission Window Opens:
Friday, November 15, 2019

Points Available:
200 XP for a working demonstration
50 XP for readable & understandable code

Objectives:

- Learn the basics of Linear & Polynomial Regression
- Have fun!

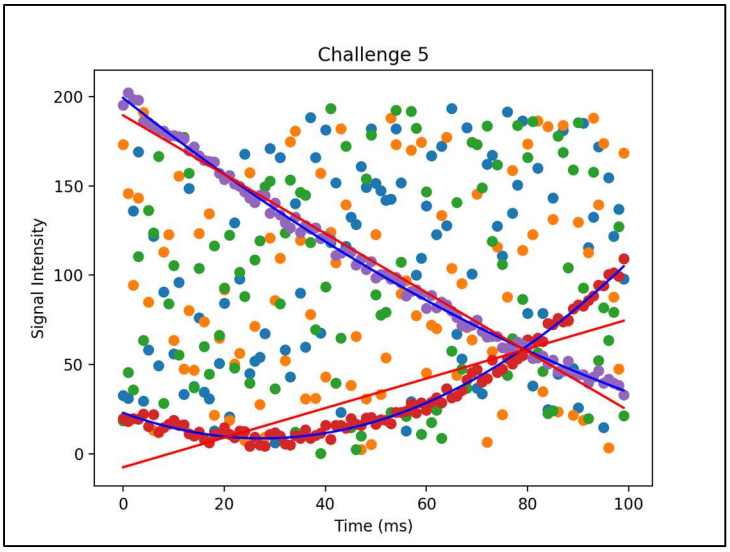
Assignment:

Remember back at the beginning of the semester when life was simple and the plots were easy? Let's revisit that time but bring our newfound knowledge of linear and polynomial regression with us! Your task will be to take data similar to that used in Challenge 1 and create a predictive model of the signals hidden in the data.

You have been contracted by SETI to analyze radio telescope data. They have sent you a 100ms dataset containing signal intensity data from five unique radio frequencies. You must determine if there are signals hidden in the data (Spoiler: There are! Specifically two!) You must find these using your program. Once you've found the signals, you must plot the raw data as a scatter plot and then draw a line plot of both linear and polynomial regression lines on top (see figure below). For linear regression, you must write the code (from scratch!) to determine your $y=mx+b$ formula. For your polynomial regression, you will need to use NumPy's `polyfit` and `polyval` functions (you CANNOT use this for your linear regression).

On the class website, you'll find a CSV file for Challenge 5 that has three columns: sample time in milliseconds, radio frequency in MHz, and signal intensity. Similar to your previous challenge, you must first find out which radio frequencies are showing a structured signal and which are just random noise. However, this time *your program* has to figure out which is which! To do this, you will need to check each unique frequency using some sort of test (time/signal correlation might be a good place to look, but there are other ways too). **You cannot hard-code your**

frequencies into your program. Your program must be able to start with the raw data and end with a plot showing both a polynomial (as a blue line) and linear regression line (as a red line) for each signal.



Submission Instructions:

1. Demo your ***working*** code to the class TA before uploading it to Blackboard. You cannot proceed to step 2 before doing this.
2. Once your code is working and you've demoed it to the TA, upload ***all your code*** to Blackboard as a single ZIP file. Name your ZIP file *spiritAnimal*.zip, where *spiritAnimal* is your class user ID (not your webID or ID number). Be sure to name your main source file *spiritAnimal.py*. In a comment at the top of the file, include the following information. Spirit Animal User ID, Date the file was last edited, Challenge Number, and cite any sources that you used as a reference for code, data, and content (including title and URL).