Project 3

By: Richard Foggio, Alex Trainham, and Noah Weingand
CMSC 409 - Fall 2020 - Dr. Manic

Student certification:

Team member 1:
Print Name: ___Noah Weingand_____ Date: __10-19-2020_____
I have contributed by doing the following:

Driver for group programming, developing/programming algorithm with Alex, building function to normalize and add quadratic/cubic columns for squared and cubed hour values, answered questions 2 and 4.

Signed: 

Team member 2:
Print Name: _____Richard Foggio_____ Date: _____10-19-2020_____
I have contributed by doing the following:

Coding algorithm (calculating test error, averaging weights per architecture for testing).
Reviewing/Answering questions.

Signed: 

Team member 3:
Print Name: __Alex Trainham_____ Date: ___10-19-2020_____
I have contributed by doing the following:

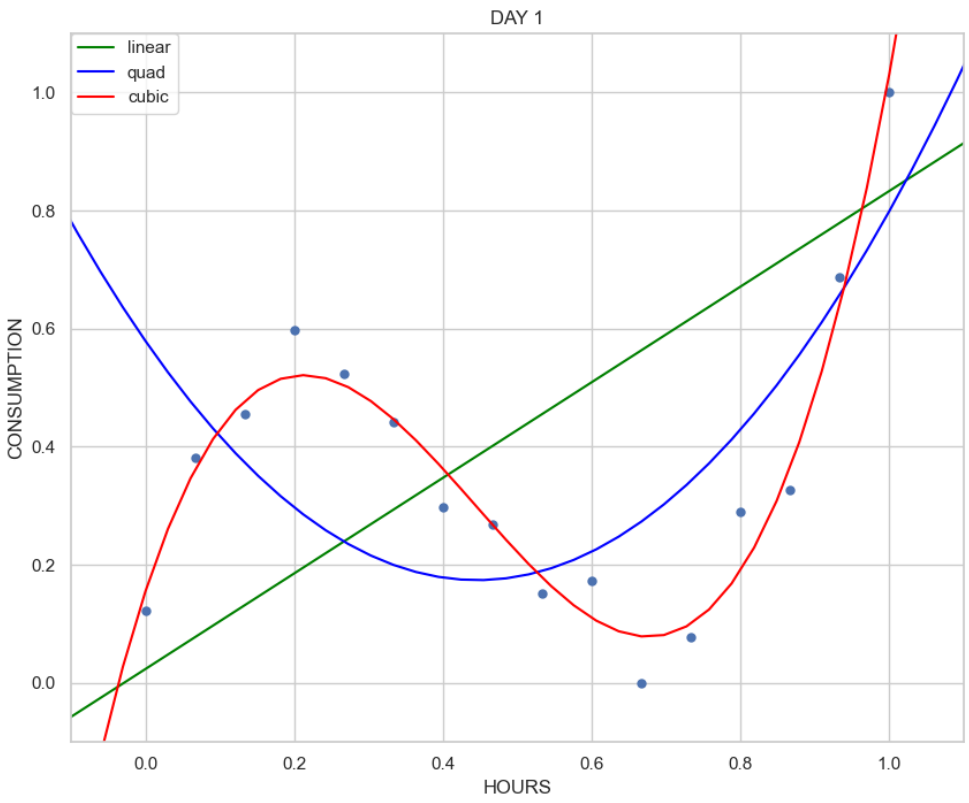
Coding the algorithm (pair programming), writing graphing function, normalizing/preprocessing of the data, and Graphing results. Reviewing and answering questions.

Signed:

1) The inputs to the neuron are the time of day, and the output is the power consumption in kilowatts.

2) Summation, which implies linear activation, because we're predicting values from a given function rather for specific classes (compared to previous assignments). These predicted values can range from -INF to +INF, so there are no hard edge cases like 0 and 1 for classification.

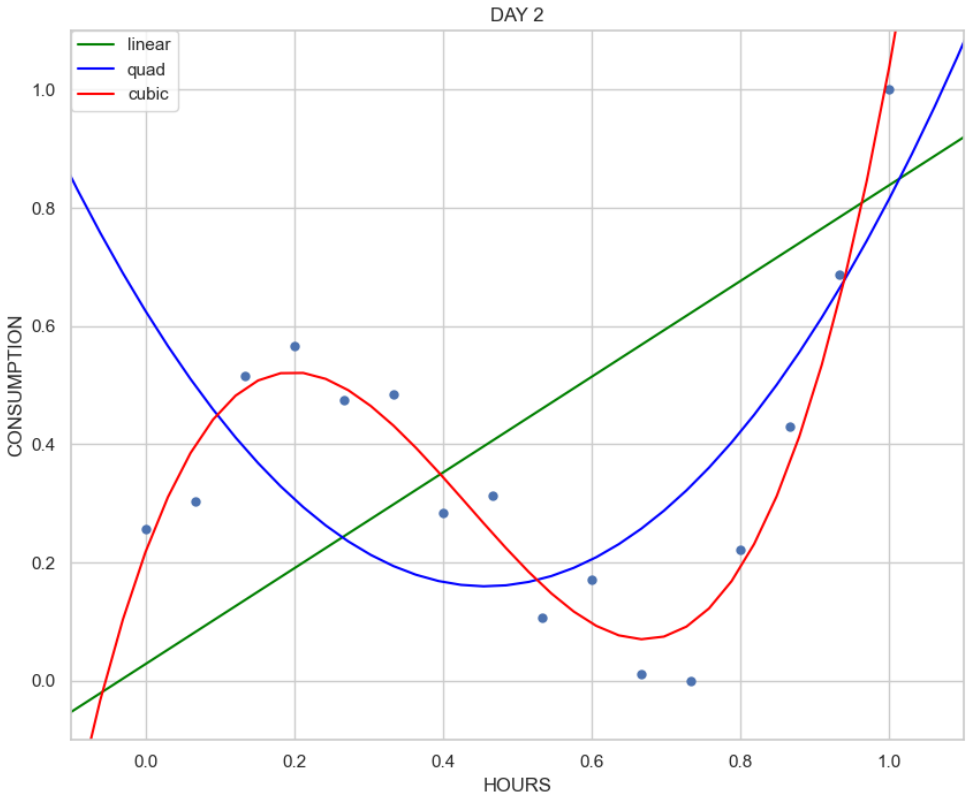
3) Training and Testing Graphs/Data:
    a) Training



Day 1 Training Errors:

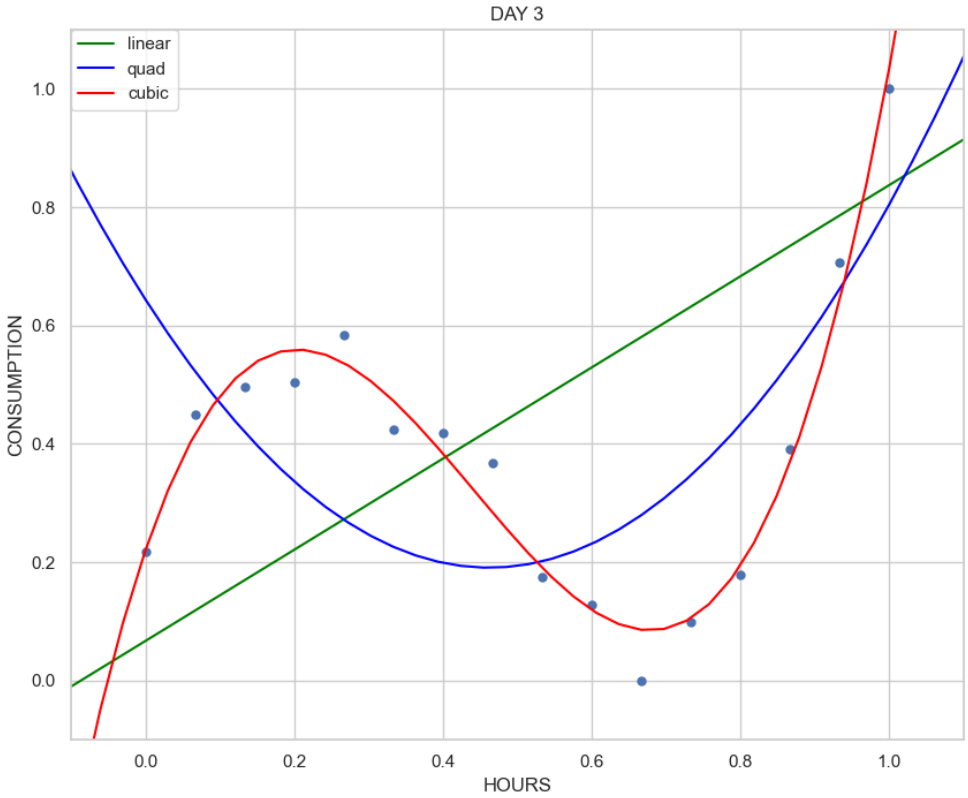
[TRAIN ERROR: DAY 1 Linear] 0.962380

[TRAIN ERROR: DAY 1 Quadratic] 0.787838
[TRAIN ERROR: DAY 1 Cubic] 0.042372



Day 2 Training Errors:

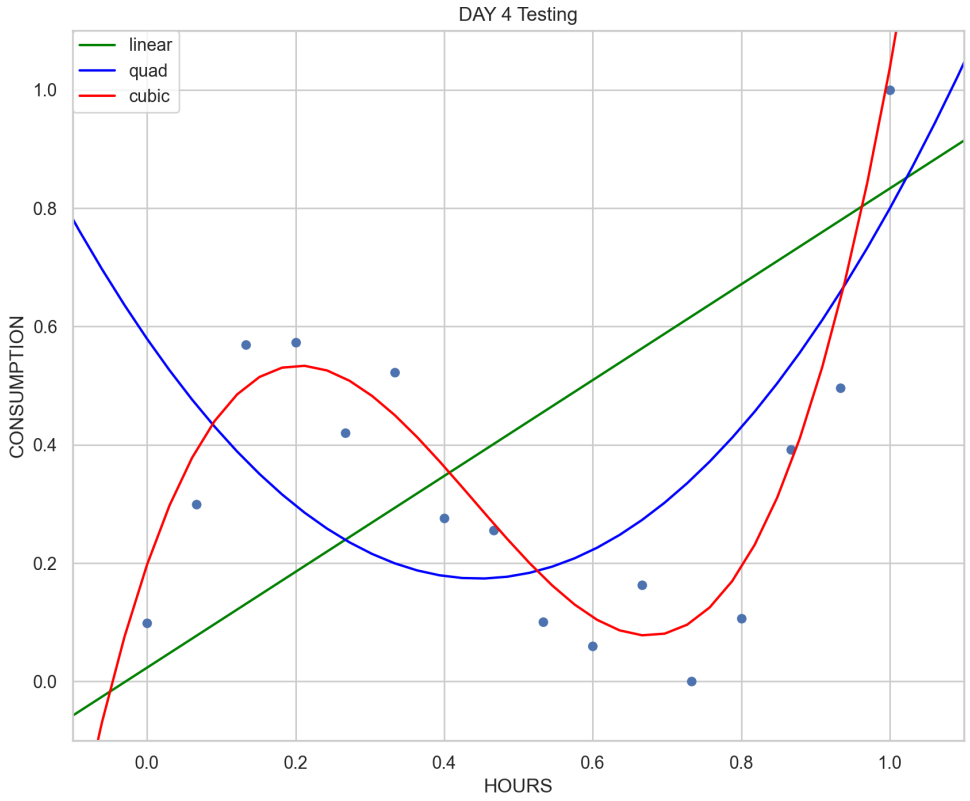
[TRAIN ERROR: DAY 2 Linear] 0.999108
[TRAIN ERROR: DAY 2 Quadratic] 0.744746
[TRAIN ERROR: DAY 2 Cubic] 0.057503



Day 3 Training Errors:

[TRAIN ERROR: DAY 3 Linear] 0.959683
[TRAIN ERROR: DAY 3 Quadratic] 0.768641
[TRAIN ERROR: DAY 3 Cubic] 0.034100

b) Testing



Day 4 Testing Errors:

[TEST ERROR Linear] 1.918881
[TEST ERROR Quadratic] 0.896685
[TEST ERROR Cubic] 0.102937

4) Number of iterations: 50000
Learning rate:

Architecture 1 (Linear) - 0.3
Architecture 2 (Quadratic) - 0.1
Architecture 3 (Cubic) - 0.05

We chose the learning rate for each architecture to match the quality of model it would build. Since the model for a linear graph wouldn't result in a great prediction, we increased the learning rate so it could learn more efficiently without worrying about slight nuanced changes in the graph. However, since quadratic and cubic functions would be more fitting for the purposes

of modeling our specific data, the learning rate was reduced coupled with a large number of iterations to give the models time to learn, and learn effectively. A lower learning rate and increased iterations resulted in more accurate graphs than did a higher learning rate with less iterations.

Preprocessing:

1. Normalized data
    - We normalized the data so that it was more comparable and to eliminate the differences between units of measurement across each data set.
2. Added bias
    - We added a “bias” for an additional constant input valued at 1 to train with. This connects all non-constant inputs and helps fit the prediction.
3. Added columns for each architecture
    - We added columns so that we could identify the different values for graphing. Linear just had “hours”, “consumption” and “bias” while quadratic had those plus an additional column for “hours squared”. The cubic architecture also included an “hours cubed” column on top of all the columns the others had as well.
4. Averaged weights from Days 1, 2 & 3
    a. After training on days 1, 2, and 3, the weights for each respective model (linear, quadratic, cubic) were averaged to produce final weights that were used for testing. Since the neuron was trained on each day individually, averaging the output weights to create a final line for testing incorporates data learned from each day.