Welcome to Module 5's KNeighbors Labs!

In order to complete the KNeighbors labs in this module, please make sure you download and unarchive this .zip file with all the datasets and files necessary.

Lab Assignment 5

Remember that wheat dataset you used while exploring visualizations? It's about to make a comeback! While learning the many classification algorithms we're going to cover in the next few sections, it's a good idea to have a 'benchmark dataset' to come back to, so you can can compare the performance and accuracy of other algorithms.

1. Start by looking through the starter code /Module5/**assignment5.py** and /Module1/Datasets/**wheat.data**
2. Complete the assignment except for the bonus instruction.
3. Try experimenting with other feature scaling methods, in addition to normalize(), to see how they affect the decision boundary.
4. Then, answer the following questions.

Lab Questions

Lab

Lecture: Regression
Quiz

Lab: Regression
Lab

Dive Deeper

▸ 6. Data Modeling II

(2/2 points)

Please enter a numeric value (e.g. 0, 1, 10.5, etc) which correctly answers the question(s) below:

What is the accuracy score of your KNeighbors Classifier when K=9 (Enter as a decimal)?

0.87142857142857144 ✔ **Answer:** 0.871428571429

Decrease K by 1 and record the new accuracy score. Keep doing this until you get down to, and including, K=1. Concerning the scores you saw:

I was able to get an even higher reading, but overfit my data ✔

Answer: I was able to get an even higher reading, but overfit my data

EXPLANATION

Follow the steps in the starter code file. Each 'TODO' should be accomplishable with just 1-4 lines of code. The accuracy value you should be getting if you set the random state properly is 0.8714.

You should also have been able to see that same accuracy score one other time, surprisingly, when K=1.

You have used 3 of 3 submissions

Congratulations on training your computer to identify wheat kernels! As you know, PCA throws away some of your data. Yet, you were able to get the high accuracy level you got in this lab by applying KNeighbors to just **two** principal components! If you're ready for a bonus experiment, remove both the

PCA code as well as the visualization code from the lab. Run the KNeighbors Classifier on your entire X_train dataset and see how it performs compared to the PCA-only version you just completed above. Does it perform better? Or worse?

You can also try properly encoding the wheat_type series as a dummy feature, spanning three columns. If you attempt that, be sure to adjust your `.predict()` and `.score()` methods to fit.