Springboard Project 11.1 (Logistic Regression)

Using Heights and Weights data

1. *Create a scatter plot of weight vs. height. Then, color the points differently by gender.*

   After importing all necessary packages, including Seaborn, I set Seaborn preferences that would create a viewer-friendly scatter plot. Next, I imported the heights-and-weights data, plotted it, and added appropriate labels and a title.

   To add different colors for male and female observations, I grouped the data by gender. Next, I plotted the groups accordingly and added a legend, title, and axis labels.

2. *For each C, create a logistic-regression model with that value of C. Then, find the average score for this model using the cv_score function only on the training set (Xlr, ylr). Pick the C with the highest average score.*

   After importing all needed packages and functions, I read in the data for heights and weights and split it into training and test sets. I then set up the cv_score function and iterated it over the given values of C (a regularization parameter) such that it would give the lowest C with the highest score.

3. *Use the C you obtained from the procedure earlier and train a Logistic Regression on the training data. Calculate the accuracy on the test data.*

   I ran a logistic regression with the best C from Exercise 2, fit the model on the training data, and finally checked that model on the test data. The C value (0.1) actually did a little better on the test data. Since the difference in scores was less than 1, I don't think this is a real problem. I would not expect the scores to be equal, but I would hope for them to be fairly close. Otherwise, there could be problems with model generalization or overfitting.

   The aim of cross-validation is balancing bias and variance to create an appropriate model. The value of C introduces an amount of bias with the goal of reducing variance, but the amount of variance reduced should be markedly higher than the amount of bias introduced; otherwise, the C is too high and the trade-off is not worthwhile.

4. *Use scikit-learn's GridSearchCV tool to perform cross validation and grid search. Does it give you the same best value of C? How does this model you've obtained perform on the test set?*

   After creating a grid using the given values of C, I performed a grid search using logistic regression with the training data. A different C-value (.001) from the one from Exercise 2 was selected, but the score changed by only .0002 points. Last, I checked the model on the test data and found a score of .9256 – a jump of just .0004 from the cv_search procedure.