Homework 2 - Classifying Wine Quality
DS6040 Bayesian Machine Learning
(Summer 2022, Instructor: Taylor Brown)

2022-06-18

1 The Problem

This assignment will let you practice applying Bayesian classifiers, specifically *linear discriminant analysis* and *quadratic discriminant analysis*, and examining the effect of different prior distributions for class membership. You'll be analyzing a data on various chemical properties of white wine, and working to classify these into three quality categories.
What you will need:

- `whitewine-training-ds6040.csv` - The training dataset, found on the Collab Site
  - Wine data consists of 12 variables. The variable of interest is "wine_quality", which takes values "A", "C" and "F" for "excellent", "average", and "poor," respectively. All other features are continuously distributed.
- `whitewine-testing-ds6040.csv` - The testing dataset, found on the Collab Site
- `BayesClassifiers.ipynb` - Jupyter notebook with LDA and QDA classification functions.

Prepare your own Jupyter Notebook for submission. You may discuss this assignment with other students in the class, but you must submit your own answers to the questions below. **Include an honor pledge with your submission.**

2 Questions

1. **Linear Discriminant Analysis (25 Points)** After loading your training data, fit LDA classifiers (using all 11 continuous features to predict wine quality) under the following scenarios
   (a) Non-informative (flat) priors on wine quality
   (b) Priors that reflect the observed proportion of wines at different quality levels.
   (c) Priors that reflect the notion that most wines are awful, some wines are average, and few wines are good (your choice for specific values.)
   (d) Priors that somebody with terrible taste in wine would use (i.e. most wines are good, few wines are bad or average).

   In each case, **calculate the overall mis-classification rate and present a cross-tabs table showing which categories are being classified correctly vs. incorrectly in your training dataset.** (Note, do not present pair-plots here, the dimensionality is too high.)

   Next, for each prior, apply your LDA model to the testing dataset, and present the mis-classification rate and cross-tabs.

   Discuss the performance of your LDA models under your various choices of priors. How does the performance change when we start testing our models on the testing data?

2. (25 Points) Fit LDA models for each combination of 3 features (loop over all combinations of 3 features, there will be 165 combinations). Use flat priors on wine quality.
   (a) For each model, extract the overall miss-classification rate for both the training and the testing dataset. Which combination of three features provides the lowest miss-classification rate for the testing and training datasets? Are they the same or different (between the training/testing)? For the best performing models (for training and testing), using the functions provided, provide pair-plots for mis-classification.
   (b) Using priors that reflect the observed proportion of wine quality in the training dataset, identify the combination of three features that provide the lowest mis-classification rate. Do the best models differ from when you used flat priors?

3. **Quadratic Discriminant Analysis (25 Points)** Now, fit QDA classifiers (using all 11 continuous features to predict wine quality) under the following scenarios (NOTE: This problem is identical to the previous, except now you are using QDA rather than LDA. Copy-paste your code accordingly.)

   (a) Non-informative (flat) priors on wine quality

   (b) Priors that reflect the observed proportion of wines at different quality levels.

   (c) Priors that reflect the notion that most wines are awful, some wines are average, and few wines are good (your choice for specific values.)

   (d) Priors that somebody with terrible taste in wine would use (i.e. most wines are good, few wines are bad or average).

   In each case, **calculate the overall mis-classification rate and present a cross-tabs table showing which categories are being classified correctly vs. incorrectly in your training dataset.** (Note, do not present pair-plots here, the dimensionality is too high.)

   Next, for each prior, apply your QDA model to the testing dataset, and present the mis-classification rate and cross-tabs.

   Discuss the performance of your QDA models under your various choices of priors. How does the performance change when we start testing our models on the testing data?

4. (25 Points) Fit QDA models for each combination of 3 features (loop over all combinations of 3 features, there will be 165 combinations). Use flat priors on wine quality.

   (a) For each model, extract the overall miss-classification rate for both the training and the testing dataset. Which combination of three features provides the lowest miss-classification rate for the testing and training datasets? Are they the same or different (between the training/testing)? For the best performing models (for training and testing), using the functions provided, provide pair-plots for mis-classification.

   (b) Using priors that reflect the observed proportion of wine quality in the training dataset, identify the combination of three features that provide the lowest mis-classification rate. Do the best models differ from when you used flat priors?

5. **(EXTRA CREDIT 15 Points)** A common approach in ML is to use ensemble methods. Ensemble methods are ones that use multiple slightly different models to each "vote" on the classification of an observation (random forests are an example of an ensemble method). In this extra credit problem you are doing a quick and dirty ensemble method using the code you wrote before.

   (a) (10 Points) For each of your 3 feature QDA models, extract the overall mis-classification rate, and MAP prediction (there are class functions for each...).
   First, calculate the "ensemble" predictions for your training and testing set (separately) by finding, for each observation, the most frequent quality. Calculate the overall mis-classification rate and classification cross-tabs using this ensemble prediction (for both training and testing data). How do they compare to the 11 feature QDA, and your best 3 feature QDAs?

   (b) (5 Points) Now, reweight your ensemble predictions by performing the following steps:

   - Take your overall classification rate (1- training mis-classification) for each 3 feature model, and normalize so it sums to 1 across all 3 feature models.
   - Now, for each of your 3 feature models, create $N \times 3$ classification matrix, where N is the number of observations in that dataset (testing/training) and entries in that matrix are 0/1 indicating which quality category is the MAP estimate for an observation. (e.g, each row looks like, for example $[0, 1, 0]$)
   - For each 3-feature model, multiply your newly created classification matrix by the model's normalized classification probability.
   - Average your weighted classification matrices across all 3-feature models to obtain the ensemble classification matrix:
   $$C_{Ensemble} = \frac{\sum_{i \in \text{3-feature models}} C_i * p_i}{165} \tag{1}$$
   where p_i is the normalized classification rate for variable combination i.
   - The values in $C_{Ensemble}$ can be interpreted as classification probabilities.

   (c) (5 Points) What is the mis-classification rate of this new ensemble classifier for both training and testing datasets? Cross-tabs? How does performance compare with the unweighted ensemble classifier. **How can our use of the normalized classification rates be interpreted (Hint: The original, unweighted ensemble classifier assumed that each 3 feature model had equal prior probabilities).**