**Predicting Wine Quality: A Conundrum**

Would you like some cheese with that?

ST 599 Statistical Computing and Big Data-Project 3

June 2, 2014

Matt Edwards, Song Hoa Choi, Gina Shellhammer, Kalbi Zongo

**Introduction**

The goal of our project is to predict the blind taster quality score of a wine based on chemical tests, using the “Wine Quality” data from UCI Machine Learning Repository, <https://archive.ics.uci.edu/ml/datasets.html>. The response variable is the taster quality scale with eleven explanatory variables from various phytochemicals in wine. The explanatory variables include: fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, ph, sulphates, and alcohol. There are two datasets, 4898 white and 1599 red vinho verde wine samples from Northern Portugal, we concentrated on the white wine data. {{NOTE: We can probably cut down/out}} the list of variables.

The taster quality is a discrete scale ranging from 0 to 10, with 0 indicating ‘very bad’ and 10 indicating ‘very excellent’. The median of taster quality in white wine is 6 (n = 2198) with no graded as 0, 1, 2, or 10 (Figure1).

**Description of the machine learning method**

To start, we constructed training and testing set by using stratified sampling with the quality variable defining the strata. 37.5% of items in each stratum were randomly selected to be in the testing set and remaining 62.5% were the training set. The same training & testing sets were used for each analysis method.

*K*-Nearest Neighbor Regression

*K*-nearest neighbor (KNN) is “a supervised learning algorithm where the result of new instance query is classified based on majority of *k*-nearest neighbor category” (Teknomo, n.s.). The KNN regression is using some measures of distance to find the nearest neighbors in datasets. Order examples by increasing distance, and then find “optimal” number k of nearest neighbors. And, calculated an inverse distance weighted average with the k-nearest multivariate neighbors. In order to obtain the result, we used fit function from rminer package in R, which offers many regression types.

Ordinal Regression

Ordinal regression is one of the general linear models and its formula is similar to logistic binomial regression. It also called “Ordered logistic” or “probit” regression. This is to predict the ordinal categorical dependent variable – scale with the explanatory variables. Because taster quality scale is ordinal categorical variable, ordinal regression is reasonable to predict the taster quality from various phytochemicals. This is also to determine which the significant effect of various phytochemicals on the taster quality scale. Proportional odds were used that coefficients stay the same, and the intercept value changes. All explanatory variables have the same weight for all categories. We picked the one with the highest probability after put them in possible categories. Model selection using backward and forward was conducted to find the good-fit model. “ordinal” package installed in R.

Multiclass Classification

The classification approach is multiclass classification algorithm call One vs. All.  The algorithm trains logistic regression parameters for each class--it computes the probability of the class. Additionally, it is to predict the class for a new observation, the algorithm picks the class with highest probability. We used three data sets such as training set, cross-validation set, and test set given the amount of data in training set. 11 predictors might not be sufficient to better-fit data because it is possible high bias or under fitting. For preventing high bias, fitted 4 degree polynomial regression to raise all predictors to power 2, 3, and 4. This give us model with 44 predictors. Downside of this is over fitting (high variance). To prevent over fitting, we applied a shrink (penalty) parameter lambda to reduce the effect of each predictor. Model selection is conducted to find the best model with using I couple of lambda values and compute training set error along with cross validation error.

Random Assignment

Since most of the wines (75%) were rated with a quality score of 5 or 6, our "baseline" prediction was the simple random assignment of each wine to one of those two categories.

**Summary findings**

K-Nearest Neighbor Regression (Figure 2)

* Overall 59.6% success rate
* No properly allocated 3s or 9s

Ordinal Regression (Figure 3)

* Overall 53.3% success rate
* No properly allocated 3s, 8s, or 9s

Classification (Figure 4)

The best model (likely not to over fit nor under fit) is the one with lowest cross-validation error- corresponding to lambda=0.5. plotted errors vs. lambda . Using parameters corresponding to lambda=.5, I made prediction of wine quality in test set. the test set accuracy is between 54 and 56% (because my 3 data set are randomly split, so are not same from one run to another) at my last run I got 54.38% test set accuracy and 55.83% training set accuracy. Attached is validation (model selection) curve.

* 54.4% success rate
* No properly allocated 3s or 9s

Random Assignment (Figure 5)

**Discussion including assumptions/limitations**

Assumptions:

* Regular logistic regression: individual logistic regression is independent meaning that the probability of all categories does not sum to one. Also, the category with higher probability is more likely of occur that other categories.
* Multicolinearity is not an issue with prediction.
* Ordinal regression:‘ordering of categories’ has some repercussions.
* K-Neighbors: options available for the “search method” for KNN algorithm were not explored. This changes how the hyper-parameters of the algorithm are tuned.
* Cross validation was not explored.

Limitations:

* K-Neighbors: if category distribution is skewed, larger categories can dominates, which is what we see in our result.
* Regression does not always scale well, adding covariates can bog down the nuber of comparisons, especially with model selection.
* Random or stratified sampling of data to get a reasonable set size could help.
* When two or more categories have the same probability of success, then the approach will just pick one. Aiddiotionaly, the algorithm is computationally expensive but run in about 3mns scalability is an issue for the algorithm.

**References**

Ordinal Regression. Retrieved May 27, 2014 from <http://www.norusis.com/pdf/ASPC_v13.pdf>

Herbrich,R., Graepel, T., & Obermayer, K. (1999). Regression models for ordinal data: A machine learning approach. Retrieved May 29, 2014 from

<http://research.microsoft.com/apps/pubs/default.aspx?id=65632>

Teknomo, K. (n.s.). What is K nearest Neighbors Algorithm? Retrieved May 30, 2014 from

http://people.revoledu.com/kardi/tutorial/KNN/What-is-K-Nearest-Neighbor-Algorithm.html

**Appendix**

**Figure 1. Red wine taste quality histogram**

**Macintosh HD:Users:choiso:Downloads:white_hist.pdf**

**Figure 2. K- Nearest Neighbors Regression**

**Mac HD:Users:choiso:wine-st599:images:KNNRegression_Results.pdf**

**Figure 3. Ordinal Regression**

**Mac HD:Users:choiso:wine-st599:images:OrdinalRegression_Results.pdf**

**Figure 4. Classification**

**Mac HD:Users:choiso:wine-st599:images:RandomPrediction.pdf**

**Figure 5. Random Randomness**

**Mac HD:Users:choiso:wine-st599:images:Classification_Results.pdf**