Jatin Bharwani (jsb399)
AEM 4070 Final Project Proposal
Wine Not?

Section 1: Introduction

One of the most popular classes offered at Cornell is HADM 4300, which is known as Introduction to Wines. In this class, students go over different regions from South Africa to Oregon and try different grape varietals from cabernet sauvignon to muscadet. Almost every adult will have a glass of wine at some point. Whether it's impressing a date or just having a casual glass of wine with dinner, knowing what kind of wine to drink is a life skill. Despite the usefulness of wine knowledge, not everyone has the time to learn about wine regions, varietals, and chemical composition. In addition, trying many wines to figure out which ones are good and bad can get costly fast.

Our solution to this problem is a strictly machine learning approach. We would like to see if machine learning can provide an average individual insight on whether or not a wine might be worth buying. In order to do this, we will build a web application for the user to run queries so that they can find wines without needing specific knowledge on the location, varietal, or vintage.

Section 2: Data

Kaggle provides us with two very interesting datasets. The first one shown in figure 1 contains data scraped from Wine Enthusiast. It provides us with some quantitative data like price and points, but it is mostly qualitative. This data is interesting in that it gives a sommelier's description if the wine in a taste test along with key factors when choosing wine, like variety and country. In addition to this data, we have a strictly quantitative dataset shown in figure 2. This

dataset contains the chemical makeup of each of the wines as well as a quality rating. Note that these two datasets are disjoint meaning we cannot merge them. This behooves us to look at this wine recommending system in multiple ways.

country	description	designation	points	price	province	region_1	region_2	variety	winery

Figure 1

Fixed acidity	volatile. acidity	Citric acid	residual sugar	chlorides	Free sulfur dioxide	Total sulfur dioxide	density	pH	sulphates	alcohol	quality
7.4	0.7	0	1.9	0.076	11	34	0.9978	3.51	0.56	9.4	5
7.8	0.88	0	2.6	0.098	25	67	0.9968	3.2	0.68	9.8	5

Figure 2

Section 3: Methodology

With this combination of data we have can ask several questions. Is there a chemical formula for the highest quality wine? Given the components of wine, can we predict the quality of the wine? Given the wine specifications in Figure 2 and a description, can we predict the points the wine would receive? Given the other wine specifications, can we predict a description and a price? These questions all involve the word predict meaning that we will take a machine learning approach.

I am familiar with working in python using jupyter notebook, pandas, numpy, and sklearn. These tools will allow me to develop a workbook and develop a machine learning approach to answer some of these questions. If there is more time after analysis, I will build a web app in order to give recommendations to users based on information that they can input. The

first step to any machine learning strategy is splitting the dataset into training and validation sets. We will take a standard 75% for training and 25% for testing for each of the two datasets. The training set is used to build our model while validation allows us to optimize hyperparameters of the model.

Given that we have a quantitative dataset (figure 2), we will be able to use models like logistic regression and neural networks immediately in order to predict quality rating. We plan on using K-nearest neighbors as a baseline algorithm. For the dataset in figure 1, we need more advanced techniques in order to handle our textual data. The first step will be to encode regions, varietal, designations as numbers based on a consistent mapping. For example, Italy maps to 1, Spain maps to 2, etc. Then we will need to encode the descriptions as vectors as well. This means that we take a vector with size equal to length of our vocabulary(all words seen in descriptions), and for each word we put its count at the index for the word. This provides us with a sparse vector so we may need to reduce the dimensionality by using singular value decomposition, but it will allow us to use logistic regression and neural networks for analysis.

Section 4: Reference

Personal Background:

I am actually a TA for machine learning and I have taken natural language processing, language and information, and artificial intelligence. I have also taken a class on data visualizations.

Datasets:

https://www.kaggle.com/zynicide/wine-reviews

https://www.kaggle.com/danielpanizzo/wine-quality/data