Becoming a Sommelier with Data Science

Colin McKew

Data Science Final Project

Final Presentation

The Problem and the Goal

- For novice wine drinkers, it can often to be difficult to determine what type of wine is associated with different reviews
- Novice wine drinkers are often unable to determine where the best rated wine is coming from, and also what words may be associated with better quality wine.
- How can novice wine drinker determine the quality and proper price point of a bottle of wine?
  - How do the experts describe highly rated wine?
  - Where is the best wine coming from?
- Goal
  - Properly identify the wine varieties based on expert reviews, without going tasting the wine or becoming a sommelier

Data Set

- Kaggle
  - Data scraped from WineEnthusiast magazine
    - Wine are rated on a scale of 1-100 by expert wine drinkers
    - The following field are available in the data set:
      - *Rating*
      - *Variety*
      - *Description*
      - *Country*
      - *Province*
      - *Region 1*
      - *Region 2*
      - *Winery*
      - *Designation*
      - *Price*
  - Our main field when predicting wine variety will be the description field, containing reviews sommeliers
  - Will also use other field to do exploratory analysis on where and what types of wine are rated the highest

Cleaning the Data

- In exploring the data, there were several duplicates that needed to be removed from the data set
  - Code
    - *data[data.duplicated('description', keep=False)].sort_values('description').head(15)*
    - *df = data.drop_duplicates('description')* – removing the duplicated data based on description
- Also chose to remove fields where there were 'NaN' values
  - Dropped the field 'Region 2' due to missing values
- Using this parsed data, it was now possible to explore the different characteristics of the wine varieties

Exploratory Analysis

- You can see a clear correlation between the price of a certain wine and its assigned rating
- This OLS Regression shows that for every increase in rating, there is a subsequent increase in price of $0.03

OLS Regression Results

Dep. Variable:	points	**R-squared:**	0.200
Model:	OLS	**Adj. R-squared:**	0.200
Method:	Least Squares	**F-statistic:**	2.229e+04
Date:	Tue, 03 Oct 2017	**Prob (F-statistic):**	0.00
Time:	22:26:25	**Log-Likelihood:**	-2.2074e+05
No. Observations:	89105	**AIC:**	4.415e+05
Df Residuals:	89103	**BIC:**	4.415e+05
Df Model:	1		
Covariance Type:	nonrobust		

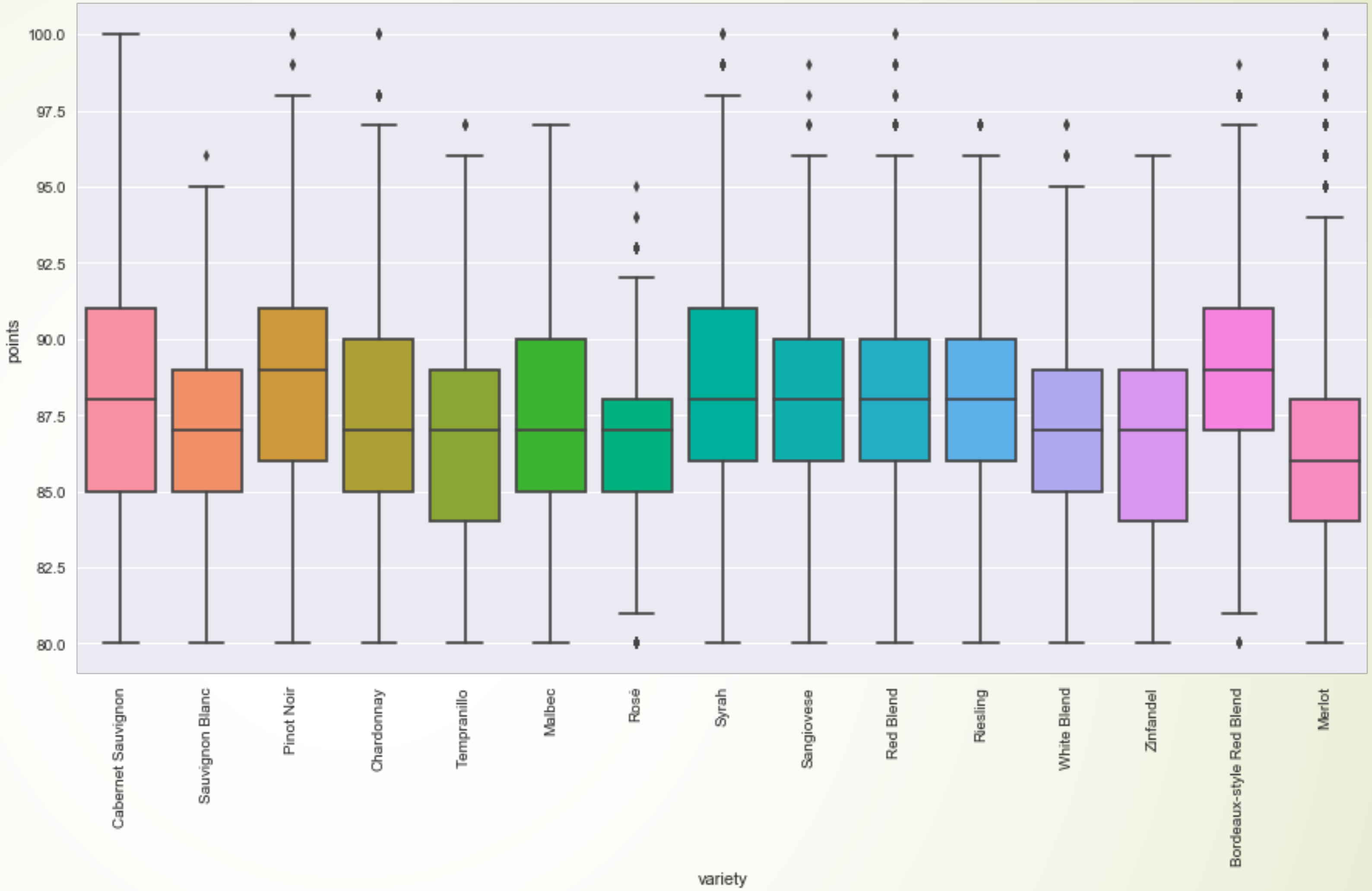
	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	86.5807	0.013	6687.550	0.000	86.555	86.606
price	0.0383	0.000	149.282	0.000	0.038	0.039

Omnibus:	26783.717	**Durbin-Watson:**	0.411
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	759914.709
Skew:	-0.851	**Prob(JB):**	0.00
Kurtosis:	17.205	**Cond. No.**	67.8

Exploratory Analysis (cont.)

country	price
US-France	50.000000
England	47.500000
Hungary	47.166667
France	44.910644
Germany	42.537787

- Top 5 average wine prices in the data set



- Top wine varieties – looks as if Pinot Noir has the highest average point score

Further Manipulating Data for Model

- Noticing that there were a great deal of wine varieties with limited data points, chose to use the top 15 varieties for the model:
  - *toplist = ['Pinot Noir', 'Chardonnay', 'Cabernet Sauvignon', 'Red Blend', 'Sauvignon Blanc', 'Syrah', 'Riesling', 'Bordeaux-style Red Blend', 'Merlot', 'Zinfandel']*
- Remove the varieties from description in order to solely use the descriptive words for prediction
  - Set the top 15 varieties to a unique list, then remove from the data set for clean descriptions
    - Left with a data set of 51,352 line items

Label Encoder

- Utilized a LabelEncoder to assign a numeric value to the top 15 wine varieties
- Now, able to use this assigned value to predict the variety of wine based on its assigned numeric value

Variety	Label Encoder
Cabernet Sauvignon	1
Sauvignon Blanc	7
Pinot Noir	4
Pinot Noir	4
Pinot Noir	4

Transforming Final Data for Model

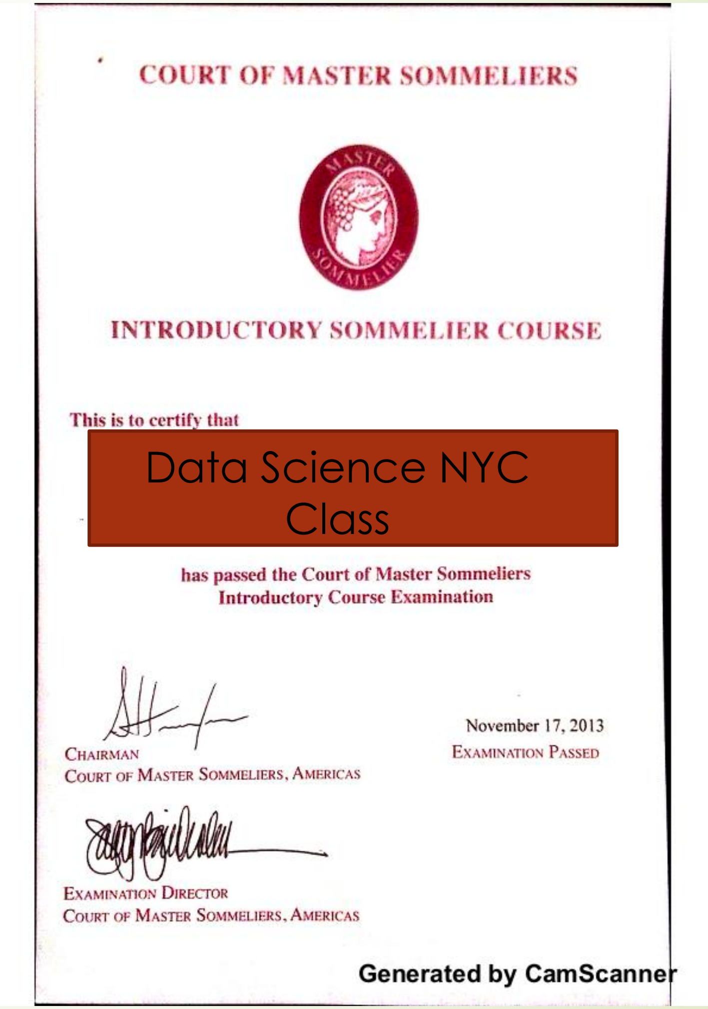
- First, needed to remove the "stop words" and transform the 'Description' field into a list of features
  - *from sklearn.feature_extraction.text import TfidfVectorizertfidf = TfidfVectorizer(min_df=5, max_features=100, strip_accents='unicode',lowercase =True, analyzer='word', token_pattern=r'\w+', use_idf=True, smooth_idf=True, sublinear_tf=True, stop_words = 'english').fit(subdata["description"])*
- Have a list of 'features' from the description to use in predicting wine variety
  - *u'red', u'rich', u'ripe', u's', u'shows', u'smoky', u'smooth', u'soft', u'spice', u'spicy', u'structure', u'style', u'sweet', u'syrah', u't', u'tannic', u'tannins', u'tart', u'texture', u'toast', u'tobacco', u'touch', u'vanilla', u'vineyard', u'white', u'wine', u'wood', u'years']*

Using Cross Validation (train and test split)

- Splitting the data set to train the model
  - y value will be the 'encoded variety' value assigned to the top 15 wine varieties
  - x value till be the vector of descriptive words created with the previous vectorizer
- Split the 20% of the data using cross validation into our train and testing variables; then imported and utilized 'xgboost' to assign learning rate for model
  - *test_size = 0.2y = subdata_2['encoded_winevariety']X = subdata_2.drop(['encoded_winevariety','variety'], axis=1)X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=test_size, random_state=seed)import xgboost as xgbclf = xgb.XGBClassifier(max_depth=3, n_estimators=300, learning_rate=0.05)*
- *Quick note on xgboost*
  - *Basically a function that uses 'boosted trees' to use additive training to optimize tree splits*

Model Results

- Running the cross validation and fitting the model
  - Checking the accuracy between the two train and test splits
  - *print('Accuracy Score:',accuracy_score(y_pred, y_test)*100,"%")*
- The accuracy score shows that the cross validation model predicted the correct encoded variety at a rate of 65%
  - Not perfect, but better than going through the process of actually becoming a true sommelier


COURT OF MASTER SOMMELIERS

INTRODUCTORY SOMMELIER COURSE

This is to certify that

Data Science NYC Class

has passed the Court of Master Sommeliers
Introductory Course Examination

CHAIRMAN
COURT OF MASTER SOMMELIERS, AMERICAS

November 17, 2013
EXAMINATION PASSED

EXAMINATION DIRECTOR
COURT OF MASTER SOMMELIERS, AMERICAS

Generated by CamScanner

Conclusion and Next Steps

- In conclusion, the model did not display a particularly high accuracy rates in terms of predicting the proper variety
  - Difficult to use a multitude of similar words to predict the proper value
  - Ability to utlize a variety of natural language processing techniques, outside of xgboost and decision trees, to explore if the hit rate would be higher
- Further steps
  - Could take this model further, and add variety of pricing data, to begin to predict what words are associated with the highest priced wines
  - Also, could utilize a Word2Vec to explore which words have the most similarity to one another
    - Similar to our exercise with StumbleUpon data to see which words are most similar across a number of reviews