WineReview

Lacey Field

Client Problem

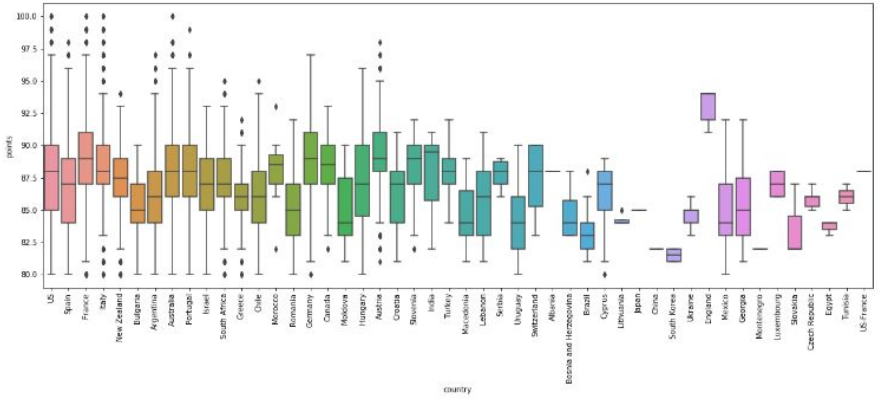
How is this data useful?

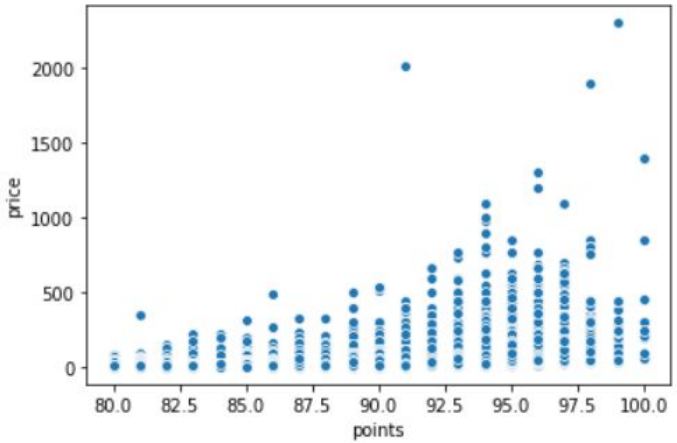
Goal is to review trends various trends such as descriptions and price/ points as well as trends between price and points

Useful to wine sellers to boost sales - can change their stocks based on higher points/ price

WineReview Data

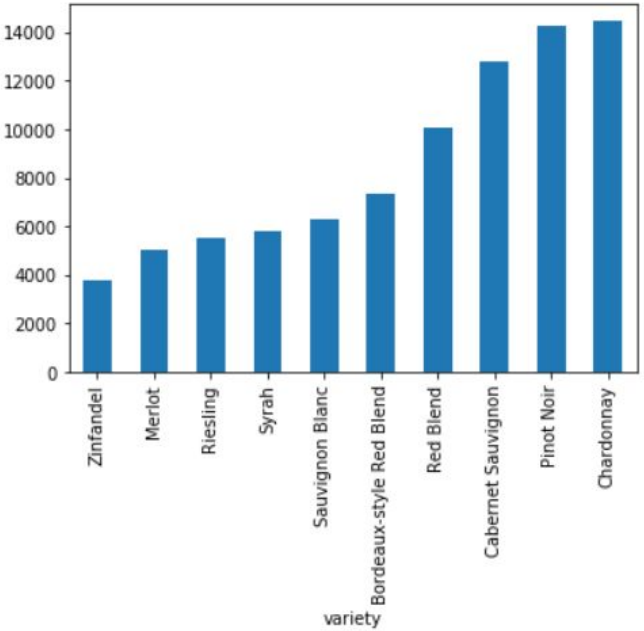
- Sourced from WineEnthusiast via Kaggle
  - Contained over 130K wines
- Data fairly clean but performed wrangling
  - Dropped unnecessary columns
  - Treated/ removed NAN values
  - Observed outliers and removed as needed
- EDA on data
  - Graphed top countries and wine varieties
  - Ranges of prices/ points, varieties
  - Point ranges by country
  - Compared price and prints to see correlation

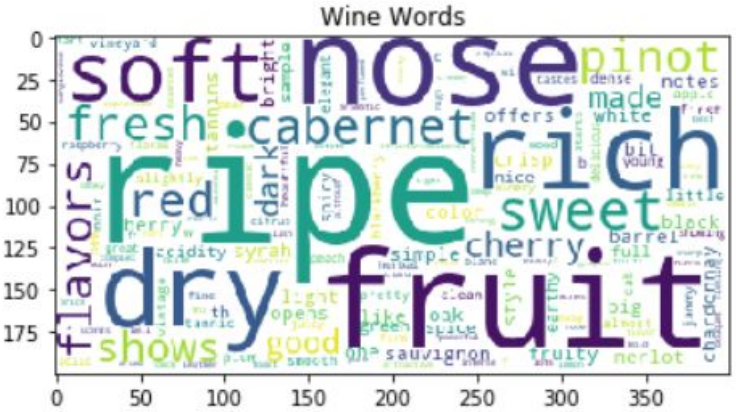




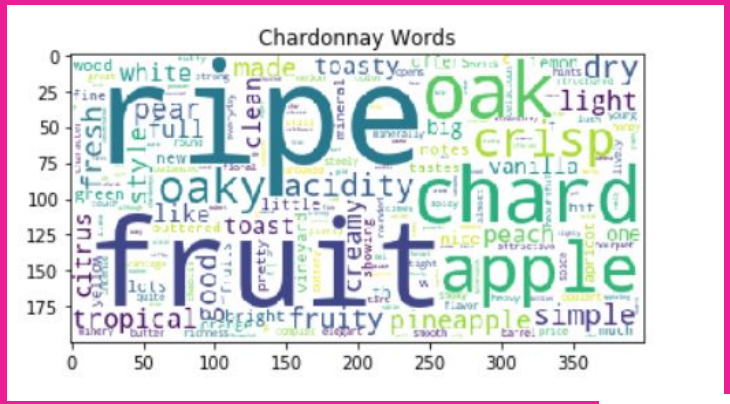
Bag of words

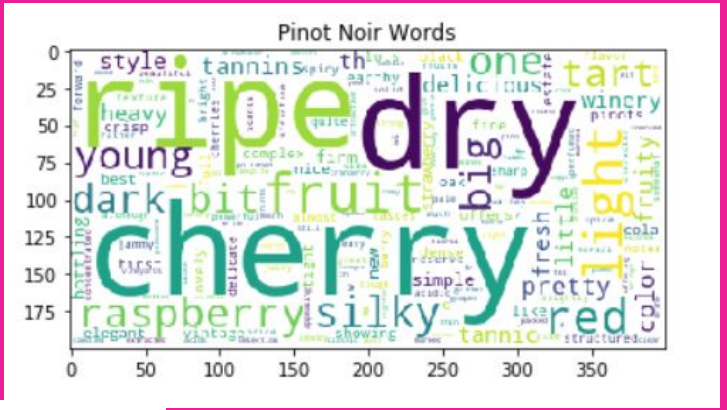
- Used Word Tokenize to split the description column into words and remove stop words
- Used Counter to create a list of top words
- Used word cloud to visualize these top words
- Repeated steps for top wine varieties to see differences with descriptions

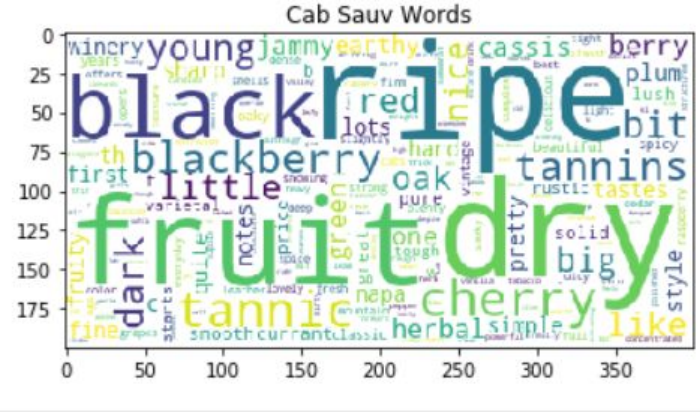




Words Split by Variety





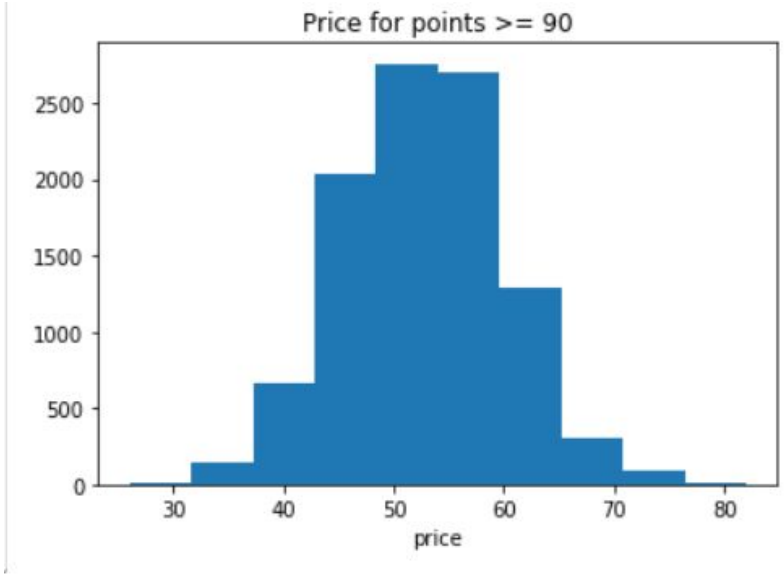


Word Preprocessing

- Used stemming and lemmatization on tokens to clean data
- Used RegexpTokenizer and lambda functions to create a column of stemmed word tokens to be used for modeling during ML section

```
stopword_list = stopwords.words('english')
ps = PorterStemmer()
wine_descriptions = wine_descriptions.apply(lambda elem: [word for word in elem if not word in stopword_list])
wine_descriptions = wine_descriptions.apply(lambda elem: [ps.stem(word) for word in elem])
wine['description_cleaned'] = wine_descriptions.apply(lambda elem: ' '.join(elem))
```

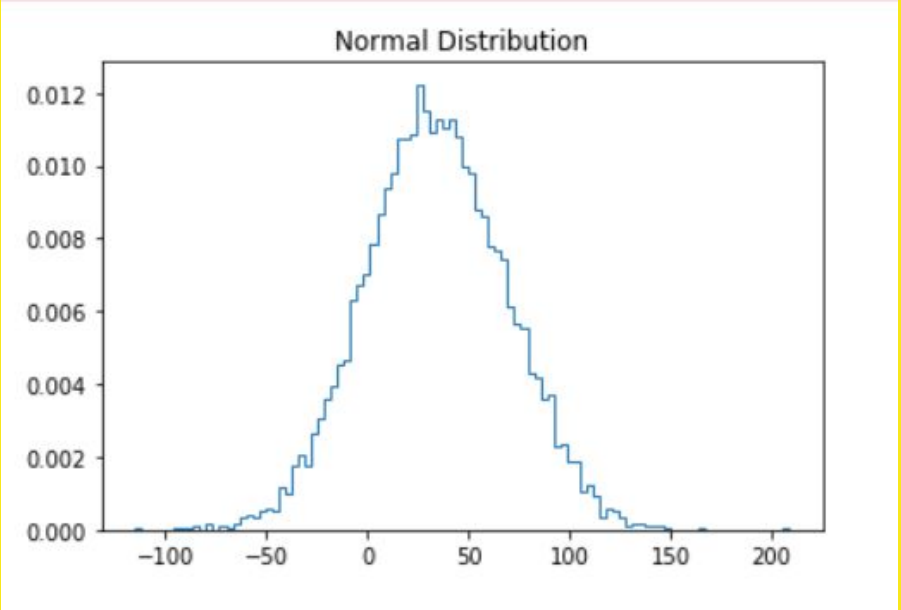
Inferential Statistics



- Calculated covariance between price and points - showed moderately correlated
- Ran linregress on points and price for summary stats - saw r-value was 0.4511
  - Also calculated pearson and spearman coefficient and saw strong results at 0.45 and 0.58
- Ran Poisson distribution of price and points - predicted mean was very similar to data with mean of 33
- Ran second Poisson distribution for points >= 90 and saw mean was higher at 52

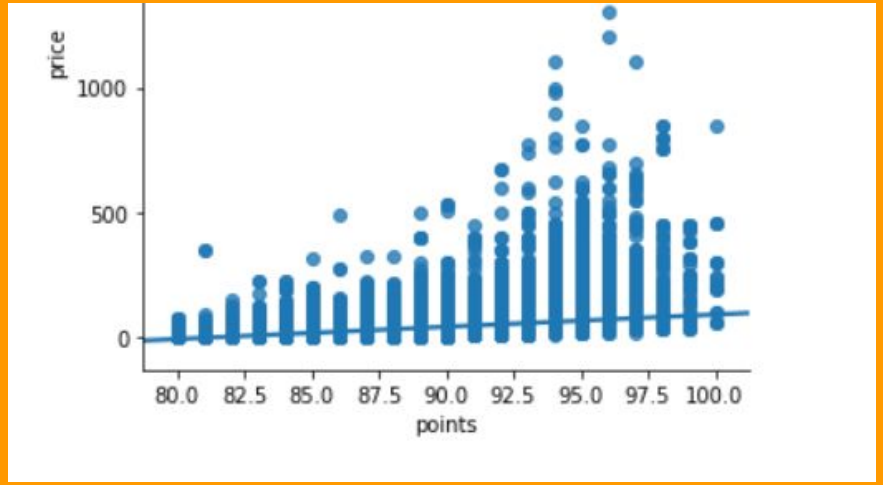
Points and Price

- Ran a random normal distribution to view PMF for points and price - saw numbers were ranging from -100 - 200 due to outliers
- Removed and re-ran to get tighter range but still had high STD



ANOVA Testing

- Performed a series of ANOVA testing for different variables to compare R**2 and p values. Variables tested:
  - Price and Variety
  - Points and Province
  - Points and Variety
  - Price and Points
- Saw that most variables hovered around R**2 of 0.10 with p-value of 0.00
- Price and points had strongest R**2 with 0.495



Pearson Correlation: (0.4511200330666678, 0.0)

OLS Regression Results

Dep. Variable:	points	R-squared (uncentered):	0.495
Model:	OLS	Adj. R-squared (uncentered):	0.495
Method:	Least Squares	F-statistic:	1.482e+05
Date:	Sat, 14 Mar 2020	Prob (F-statistic):	0.00

TF-IDF Model

- Wanted to use TF-IDF to create a more enhanced bag of words
- Used TF-IDF vectorizer on cleaned description column and transformed data
- Enumerated over matrix of tokens and TF-IDF rating to assign the term with the rank
- Sorted this list to obtain top words based on TF-IDF rank
- Can see top words are more descriptive with fruit, finish, cherri, tannin making the top ten

	term	rank
21212	wine	6607.974903
7037	flavor	6081.029707
7492	fruit	5704.858899
114	acid	4220.534530
6929	finish	4132.519846
3593	cherri	4066.382630
18772	tannin	3682.020426
5705	dri	3639.322179
969	aroma	3569.457087
15781	ripe	3425.437289

ML Modeling NB

- Train, test split with Multinomial NB
  - Used countvectorizer on wine description and split into training and test data
  - Wine variety and description yielded 0.534 accuracy
  - Wine variety and description cleaned with pre-processing yielded 0.5399 accuracy
- Other factors compared with this model:
  - Points and description cleaned yielded 0.284 accuracy
  - Points and count yielded 0.139
  - Points and province yielded 0.148
- Strongest relationship is between variety and description

```
X_train, X_test, y_train, y_test = train_test_split(wine['description_cleaned'], y, test_size=0.33, random_state=53)
count_vectorizer = CountVectorizer(stop_words='english')
count_train = count_vectorizer.fit_transform(X_train.values)
count_test = count_vectorizer.transform(X_test.values)
```

```
nb_classifier = MultinomialNB()
nb_classifier.fit(count_train, y_train)
pred = nb_classifier.predict(count_test)
metrics.accuracy_score(y_test, pred)
```

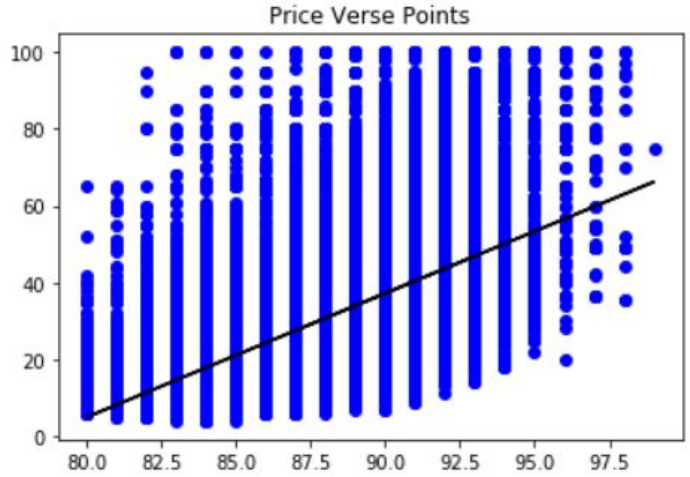
ML Modeling NB Cont.

- Used same model to compare to compare variables against price
  - Price and description yielded .162 accuracy
  - Removed price outliers and focused on top threes wines, accuracy improved to .185
- Price and country yielded .07 accuracy
- Price and province yielded accuracy .10

```
wine_update2 = wine.set_index('variety').loc[['Chardonnay', 'Pinot Noir', 'Cabernet Sauvignon']]
wine_update2 = wine_update2.reset_index()
y = wine_update2['price_int']
X_train, X_test, y_train, y_test = train_test_split(wine_update2['description_cleaned'], y, test_size=0.33, random_state=53)
count_vectorizer = CountVectorizer(stop_words='english')
count_train = count_vectorizer.fit_transform(X_train.values)
count_test = count_vectorizer.transform(X_test.values)
```

Linear Regression

- Since points and price had a strong pearson coefficient, modeled with linear regression
- Used scaling and preprocessing on data and modeled with linear regression. Yielded accuracy score of 0.197
- Removed price outliers over 500 and modeled, yielded score of .295




```
x = wine6['points']
y = wine6['price']
x = x.values.reshape(-1,1)
y = y.values.reshape(-1,1)
x = preprocessing.scale(x)
X_train, X_test, y_train, y_test = train_test_split(x, y, random_state=42)
scaler = preprocessing.StandardScaler().fit(X_train)
X_train = scaler.transform(X_train)
X_test = scaler.transform(X_test)
LN = LinearRegression()
LN.fit(X_train, y_train)
y_pred = (LN.predict(X_test))
r2_score(y_test, y_pred)
```

ML Modeling RF

- Used RandomForestRegressor with CountVectorizer on points and description - accuracy score was .70
- Also compared price and description with RF model, accuracy score was 0.30
- Used RF model with TF-IDFVectorizer with points and price against description
  - Points and description had score of .707
  - Price and description had score of .296
- Removed price outliers and modeled improved to .547

```
wine_update = wine[wine.price <=100]
X_desc = wine_update['description_cleaned']
y = wine_update['price_int'].values
count_vectorizer = CountVectorizer(max_features = 1000)
count_vectorizer.fit(X_desc)
X_count = count_vectorizer.transform(X_desc)
X_train, X_test, y_train, y_test = train_test_split(X_count, y, test_size=0.33, random_state=12)
rf = RandomForestRegressor()
rf.fit(X_train, y_train)
y_pred = rf.predict(X_test)
rf.score(X_test, y_test)
```

ML Takeaways

Fairly strong relationship with points and description/price - would recommend stocking higher point wines to boost sales/ satisfaction

Words for the top three wine varieties are most descriptive and specific and would recommend using those lists if focus is on description

Top Trends

Top trends noted during modeling with score:

price/points: 0.30

variety/ description: 0.54

points/ description: 0.71

price/ description: 0.55

Recommendations/ Follow-ups

- Would be more insightful to have actual reviews from customers of wine types
- Could perform sentiment analysis to obtain words that have positive customer responses
- This would also help to have an understanding of what varieties and types are being sold more often to improve the models

Summary

- **Data Wrangling**
  Removed outliers/unnecessary data and treated Nan values. Looked at beginning trends such as points and prices ranges. Created a bag of words for top wine varieties to compare words for trends

- **EDA**
  Explored data via Anova testing/ Pearson correlations/ Poisson distributions and noticed moderate relationship between price and points

- **Machine Learning**
  Modeled numerous features against price and points such as description and country using NB. Also modeled using RF and saw large improvement in model score. Price and points modeled very well against description. Used the TF-IDF model to create more enhanced bag of words but found wine varietal specific bags were most useful