Make Better Wine

•••

With Machine Learning

My Project

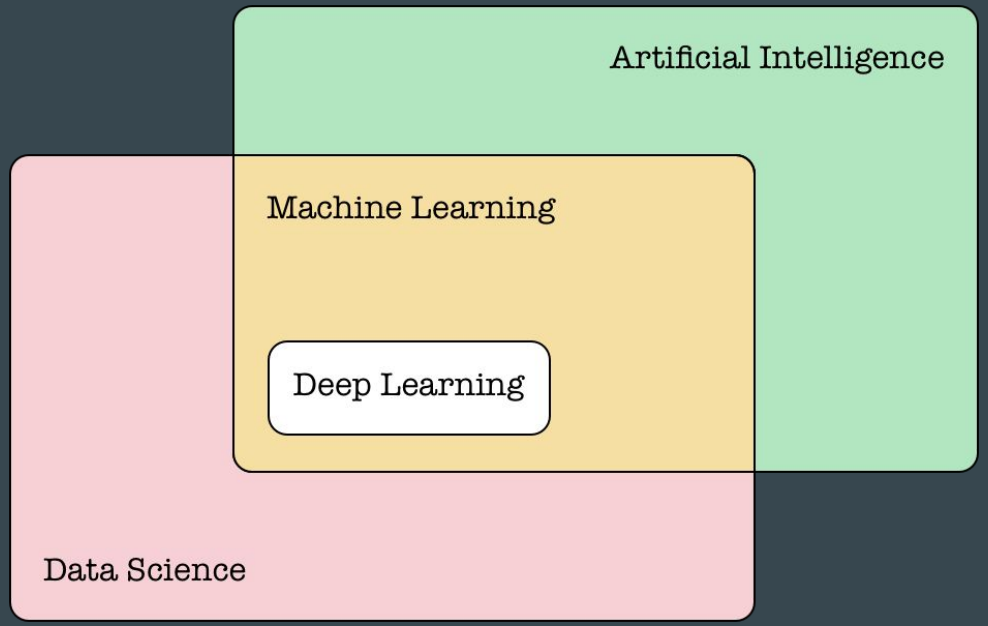
Question:
Can I predict what the wine quality rating will be at the review phase?

The Data:
A data set of red and white wines with various chemical features and a quality label is considered.

fixed acidity, volatile acidity, citric acid, residual sugar, chloride, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, alcohol, quality

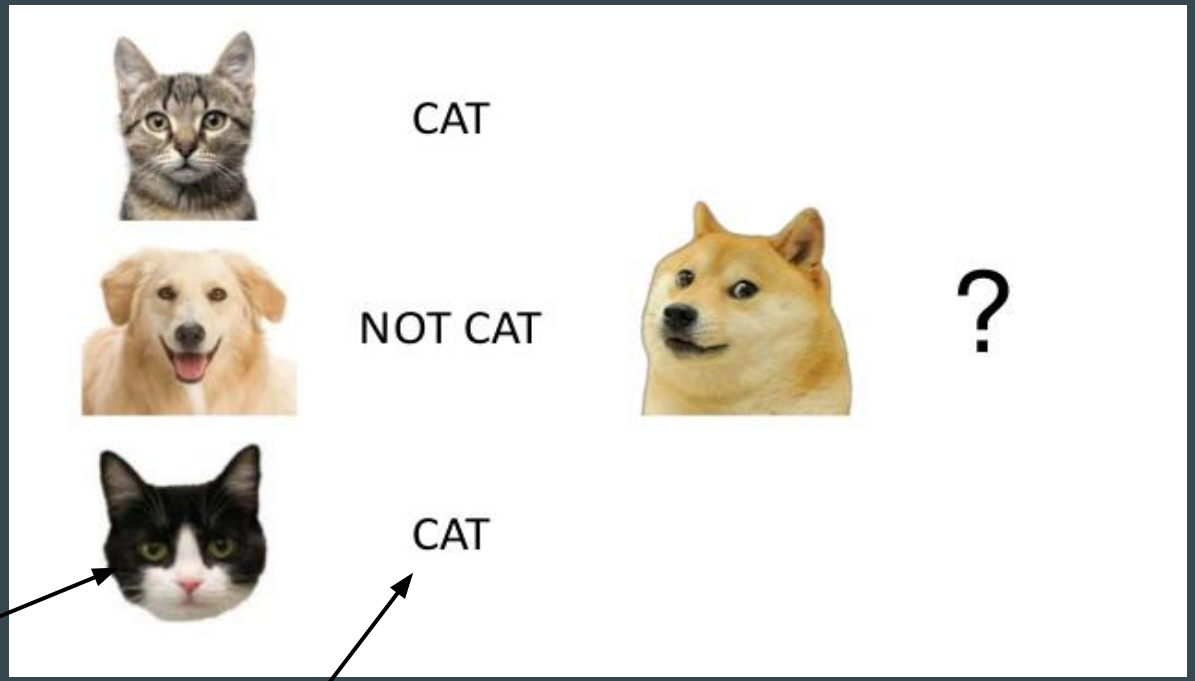
What is Machine Learning?

The ability for a computer to learn without being explicitly programmed via large datasets and fancy math.



- AI produces behaviors.
- ML produces predictions.
- DS produces insights.

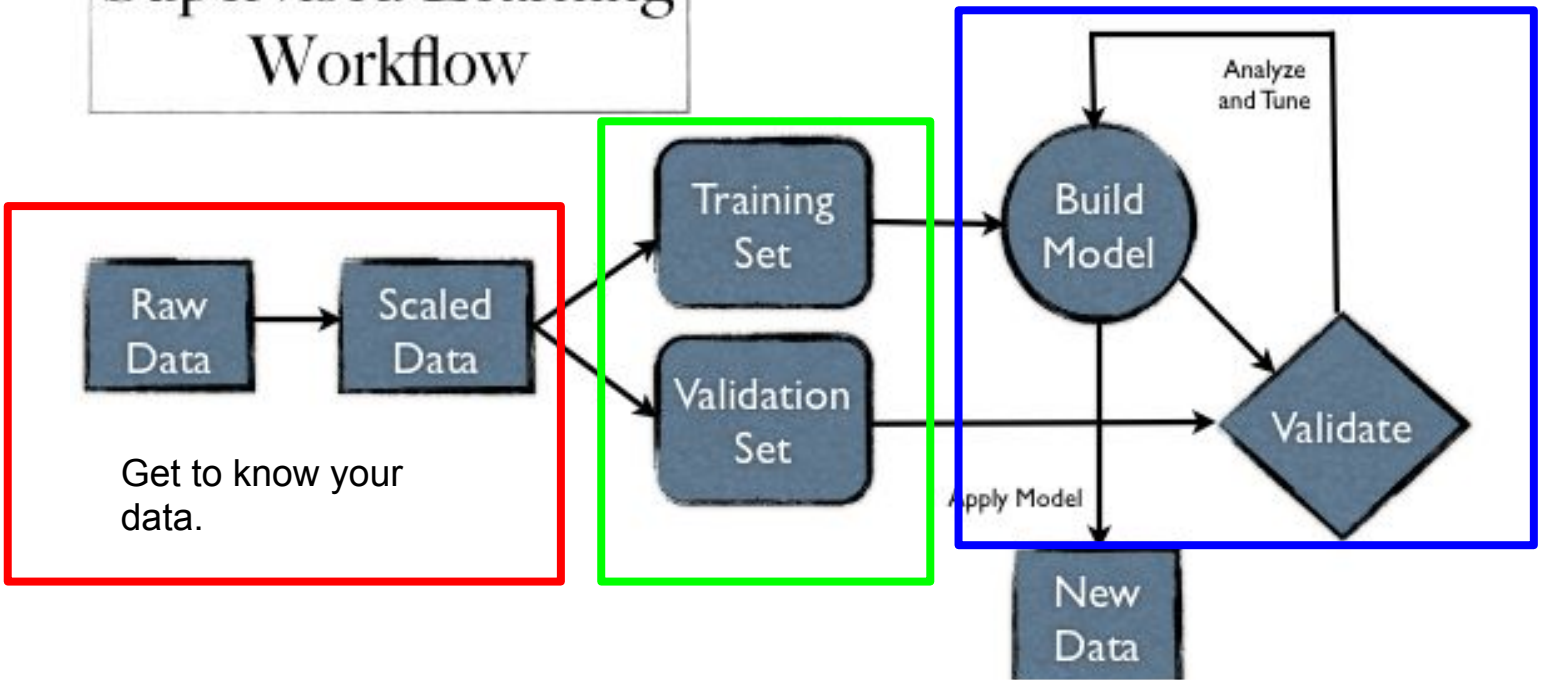
What is Supervised Learning?



Data

Label

Supervised Learning Workflow



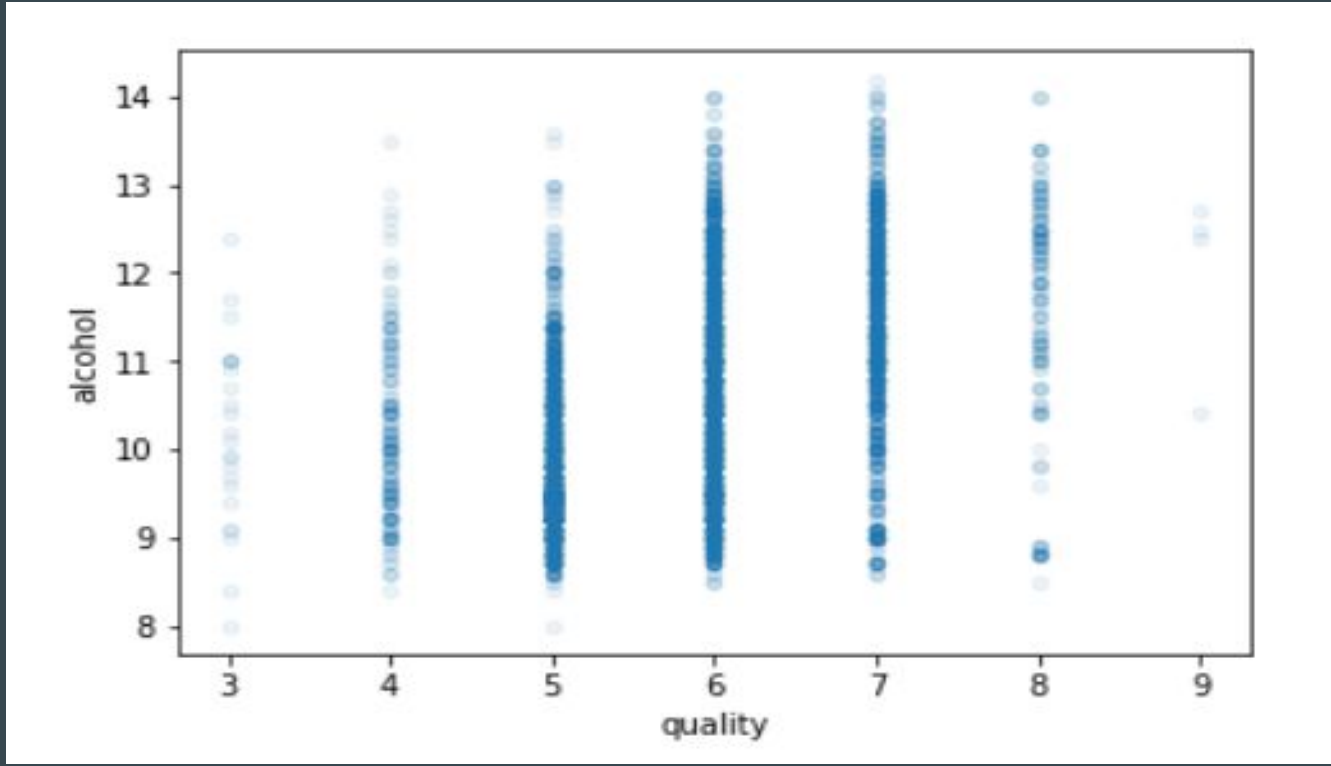
Supervised Learning Workflow

Get to know your data.

Correlations and Data Visualization

Quality VS:

alcohol 0.448437
citric acid 0.084433
pH 0.020651
density -0.301925

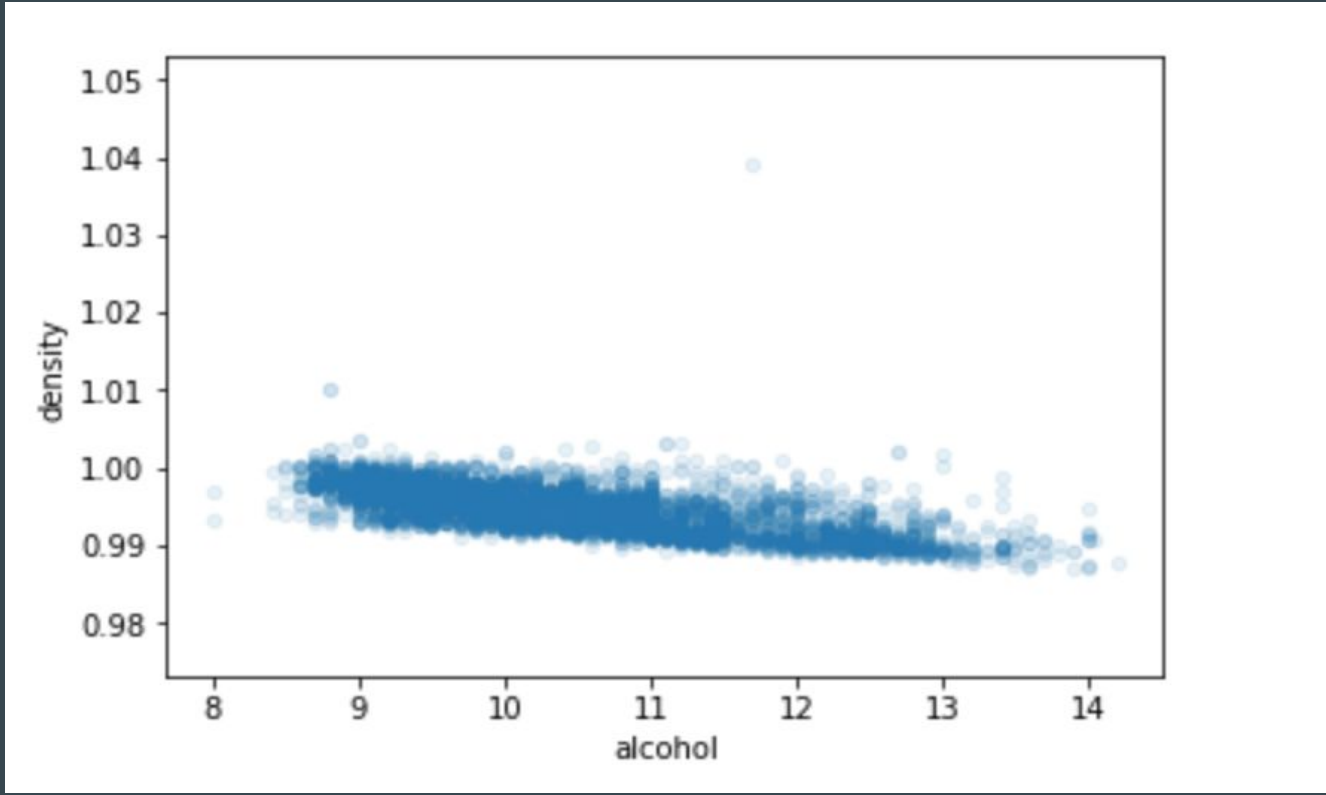


* correlation implies (not causation)

Correlations and Data Visualization

Alcohol VS:

pH 0.115036
density -0.683560
residual sugar -0.365590
Sulfur dioxide -0.273108

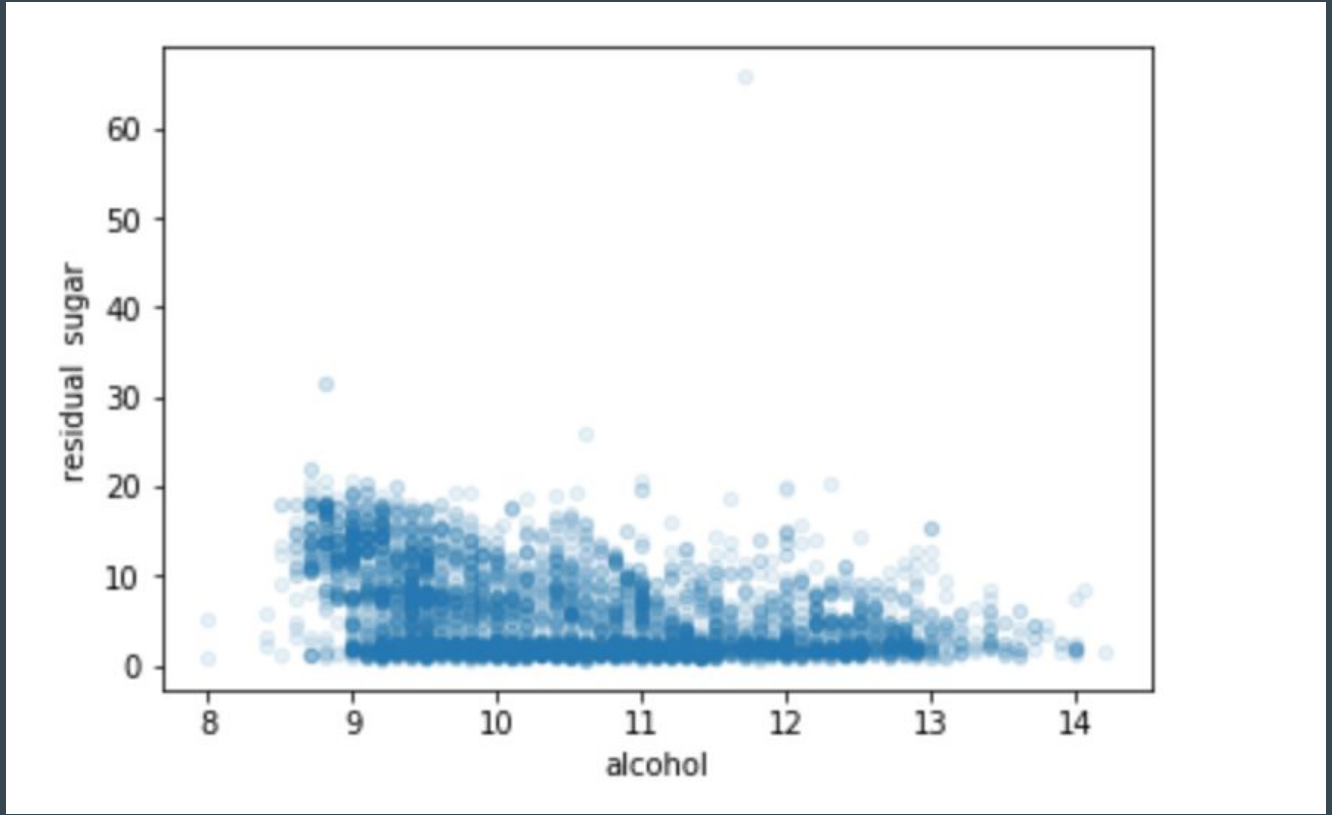


* correlation implies (not causation)

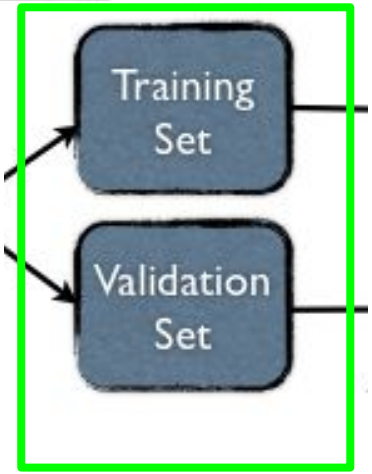
Correlations and Data Visualization

Alcohol VS:

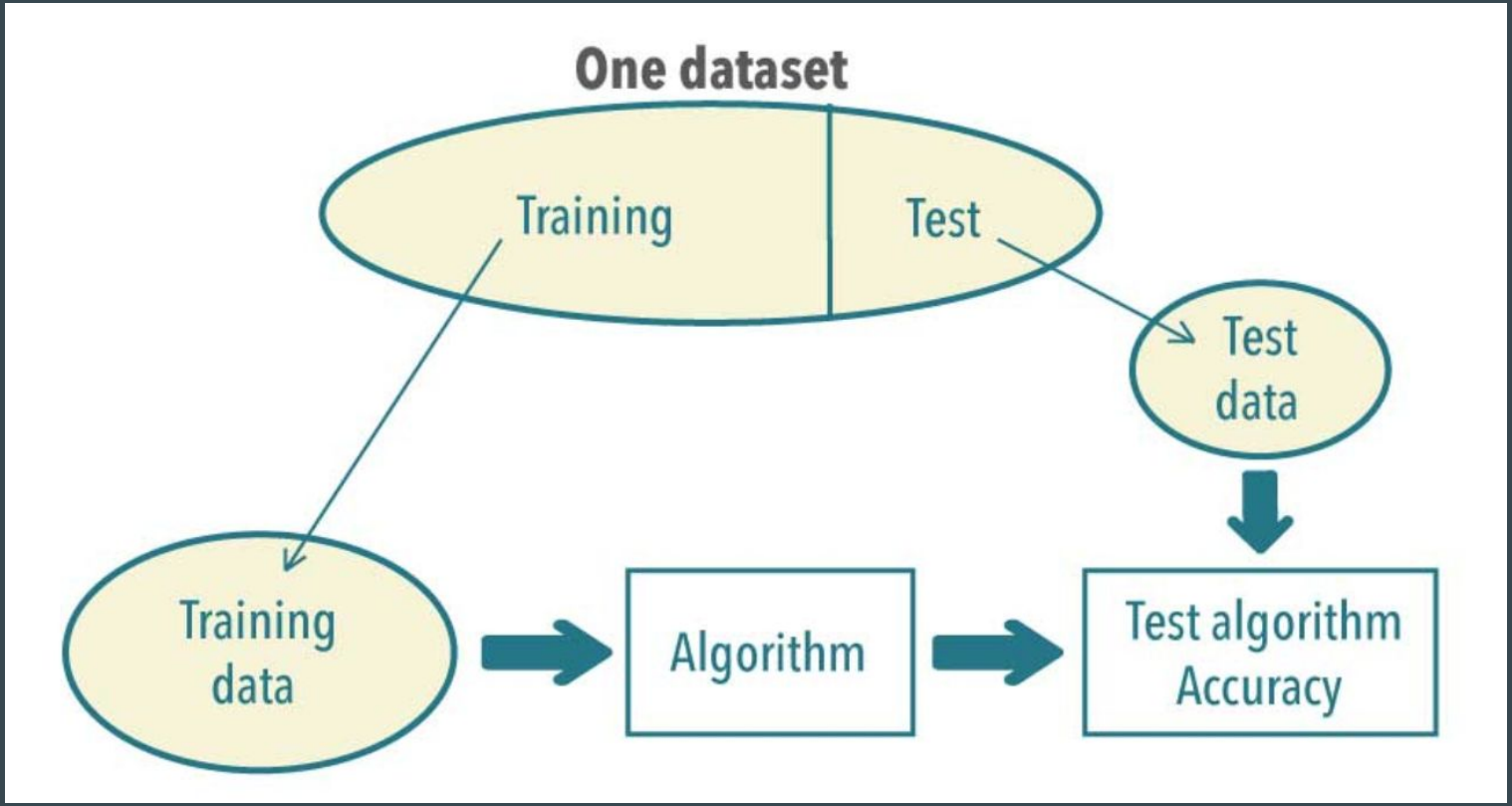
pH 0.115036
density -0.683560
residual sugar -0.365590
Sulfur dioxide -0.273108



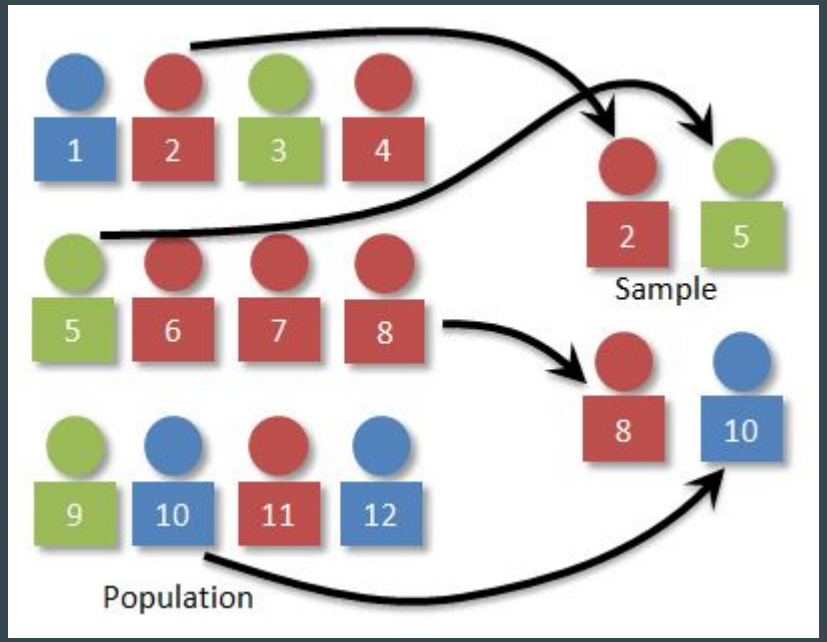
Supervised Learning Workflow



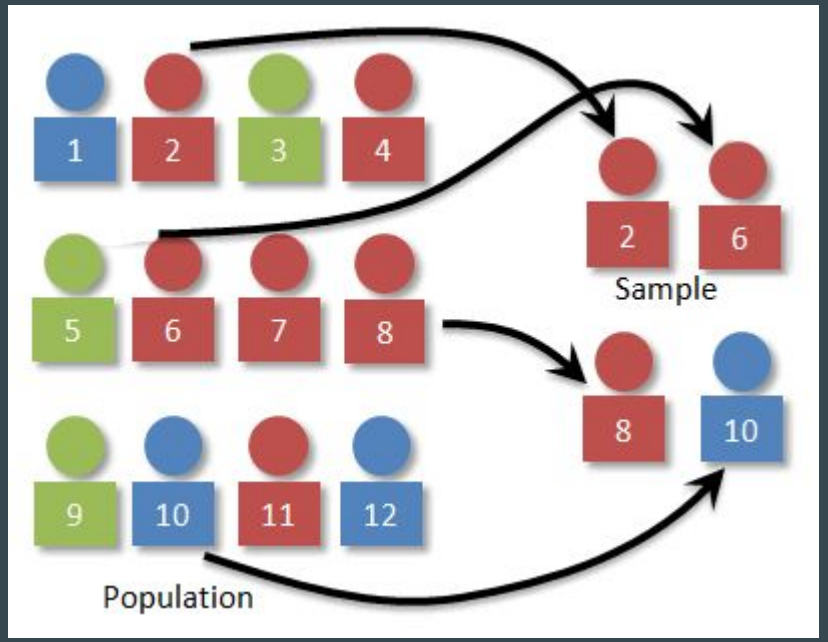
Training vs Test (Validation) Data



Sampling and Bias

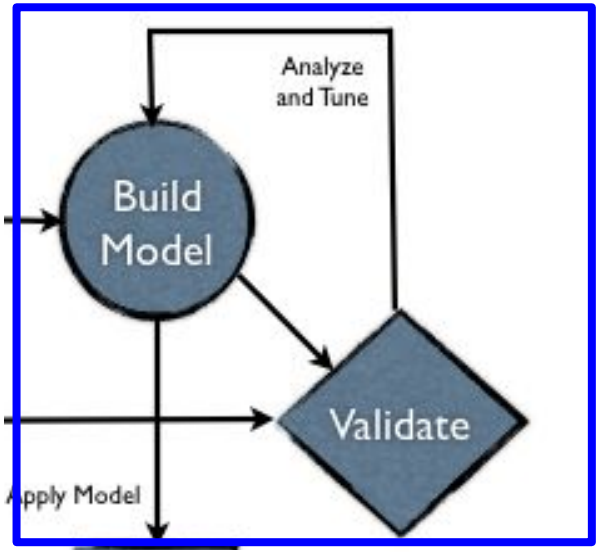


Good Representation

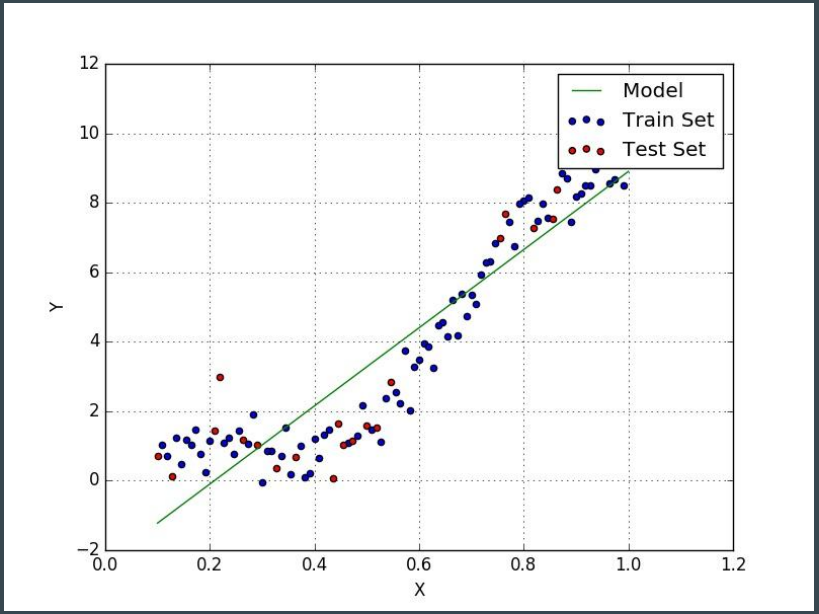


Bad Representation

Supervised Learning Workflow



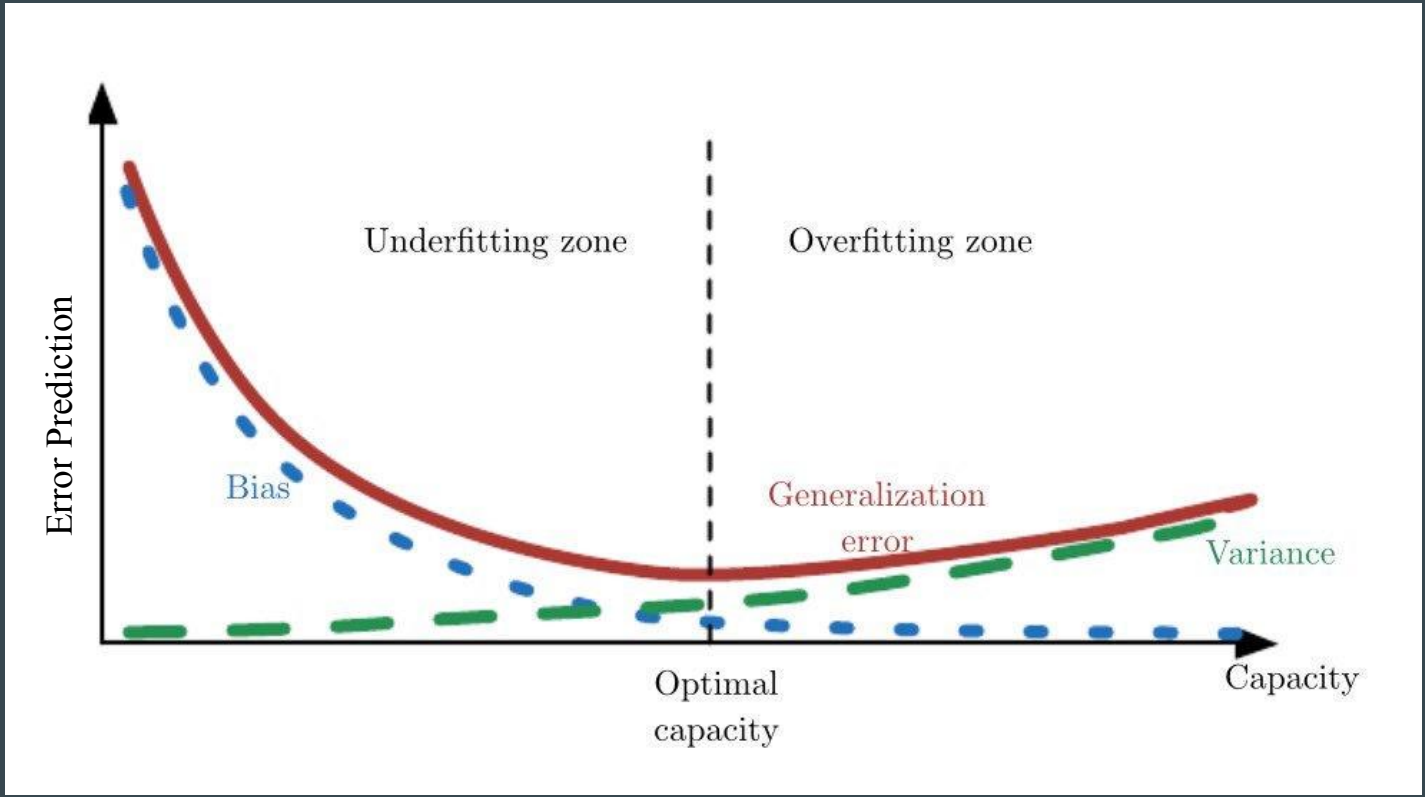
Linear Regression



If the goal is prediction, or forecasting, or error reduction, linear regression can be used to fit a predictive model to an observed data set of y and X values. After developing such a model, if an additional value of X is then given without its accompanying value of y, the fitted model can be used to make a prediction of the value of y.

When a value is being predicted, supervised learning is called regression.

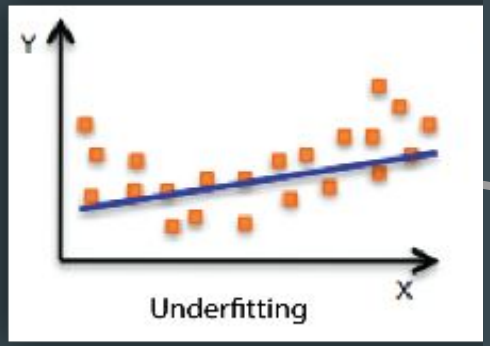
Regression Error (RSME):

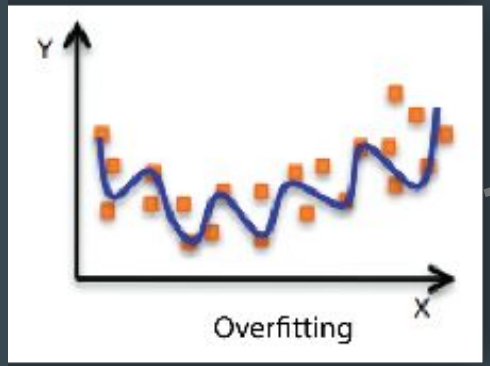


Train the Model

What do these errors mean?

```
In [13]: from sklearn.linear_model import LinearRegression
         from sklearn.metrics import mean_squared_error
         import numpy as np

         lin_reg = LinearRegression()
         lin_reg.fit(wine_data, wine_data_labels)
         wine_predictions = lin_reg.predict(wine_data)
         lin_mse = mean_squared_error(wine_data_labels, wine_predictions)
         lin_rmse = np.sqrt(lin_mse)
         lin_rmse

Out[13]: 0.7318414929843039

In [14]: # this seems promising but did it overfit?
         from sklearn.tree import DecisionTreeRegressor
         tree_reg = DecisionTreeRegressor()
         tree_reg.fit(wine_data, wine_data_labels)
         wine_tree_predictions = tree_reg.predict(wine_data)
         tree_mse = mean_squared_error(wine_data_labels, wine_tree_predictions)
         tree_rsme = np.sqrt(tree_mse)
         tree_rsme

Out[14]: 0.0

In [15]: # looks like the decision tree did better but this one has a small error
         from sklearn.ensemble import RandomForestRegressor
         forest_reg = RandomForestRegressor()
         forest_reg.fit(wine_data, wine_data_labels)
         wine_random_predictions = forest_reg.predict(wine_data)
         forest_mse = mean_squared_error(wine_data_labels, wine_random_predictions)
         forest_rsme = np.sqrt(forest_mse)
         forest_rsme

Out[15]: 0.27432987254150021
```





In [13]:
```
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
import numpy as np

lin_reg = LinearRegression()
lin_reg.fit(wine_data, wine_data_labels)
wine_predictions = lin_reg.predict(wine_data)
lin_mse = mean_squared_error(wine_data_labels, wine_predictions)
lin_rmse = np.sqrt(lin_mse)
lin_rmse
```

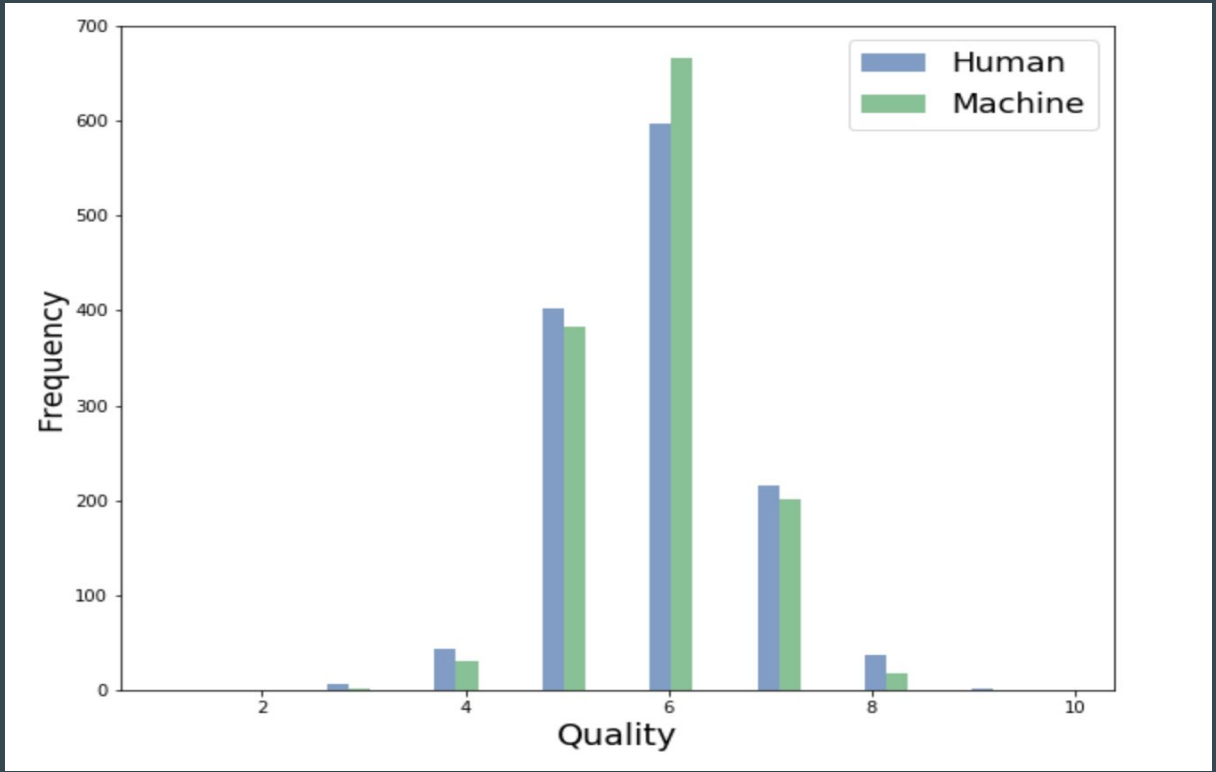
Out[13]: 0.7318414929843039

In [14]:
```
# this seems promising but did it overfit?
from sklearn.tree import DecisionTreeRegressor
tree_reg = DecisionTreeRegressor()
tree_reg.fit(wine_data, wine_data_labels)
wine_tree_predictions = tree_reg.predict(wine_data)
tree_mse = mean_squared_error(wine_data_labels, wine_tree_predictions)
tree_rsme = np.sqrt(tree_mse)
tree_rsme
```

Out[14]: 0.0

In [15]:
```
# looks like the decision tree did better but this one has a small error
from sklearn.ensemble import RandomForestRegressor
forest_reg = RandomForestRegressor()
forest_reg.fit(wine_data, wine_data_labels)
wine_random_predictions = forest_reg.predict(wine_data)
forest_mse = mean_squared_error(wine_data_labels, wine_random_predictions)
forest_rsme = np.sqrt(forest_mse)
forest_rsme
```

Out[15]: 0.27432987254150021



	Human	Machine
0	8.0	7.3
1	5.0	5.0
2	7.0	6.9
3	6.0	5.7
4	6.0	5.7
5	6.0	6.4
6	5.0	5.2
7	6.0	6.0
8	5.0	5.1
9	7.0	6.8
10	5.0	5.2
11	5.0	5.2
12	7.0	7.0
13	5.0	5.3
14	7.0	6.8
15	6.0	5.5
16	5.0	5.0
17	6.0	5.8
18	7.0	6.9
19	6.0	5.7
20	5.0	5.1
21	6.0	6.2

Final error: .302

```
data["Human"].mean()
```

```
5.833846153846154
```

```
data["Machine"].mean()
```

```
5.845230769230775
```

What did I learn from this project?

- The Machine Learning (Supervised Learning) process.
- Basic data visualization to understand the dataset.
- Linear Regression and Regression Error.
- How to present a difficult topic :)