# An Introduction To Machine Learning

RyeAnne Ricker

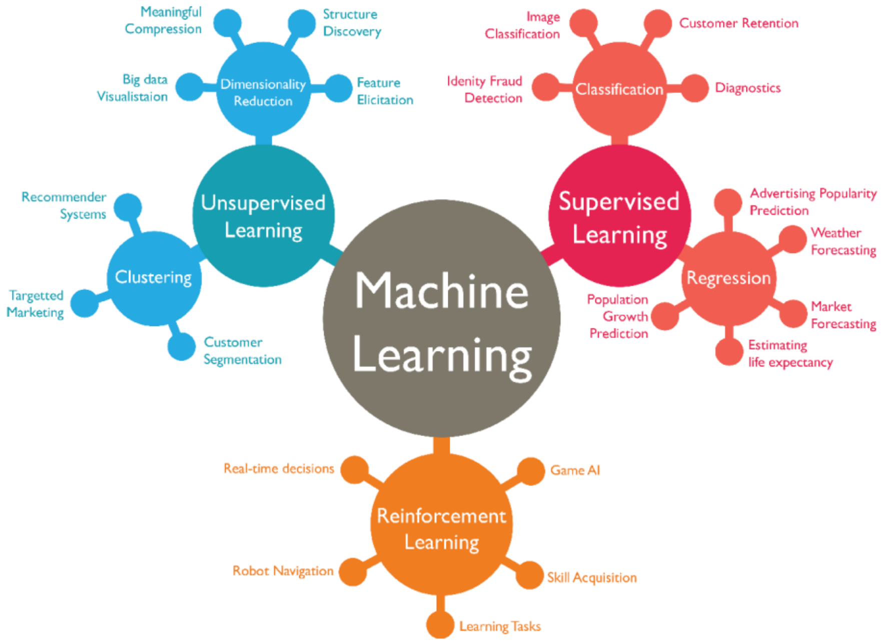
# What is Machine Learning?

- A branch of artificial intelligence (AI)
- Focuses on the use of data and algorithms to imitate the way humans learn, gradually increasing accuracy
- Can be used to learn about the data you have, classify objects, or predict
- ML is basically solving optimization problems

# Major Categories of Machine Learning

Supervised: Maps input variables (X) to an output (Y)

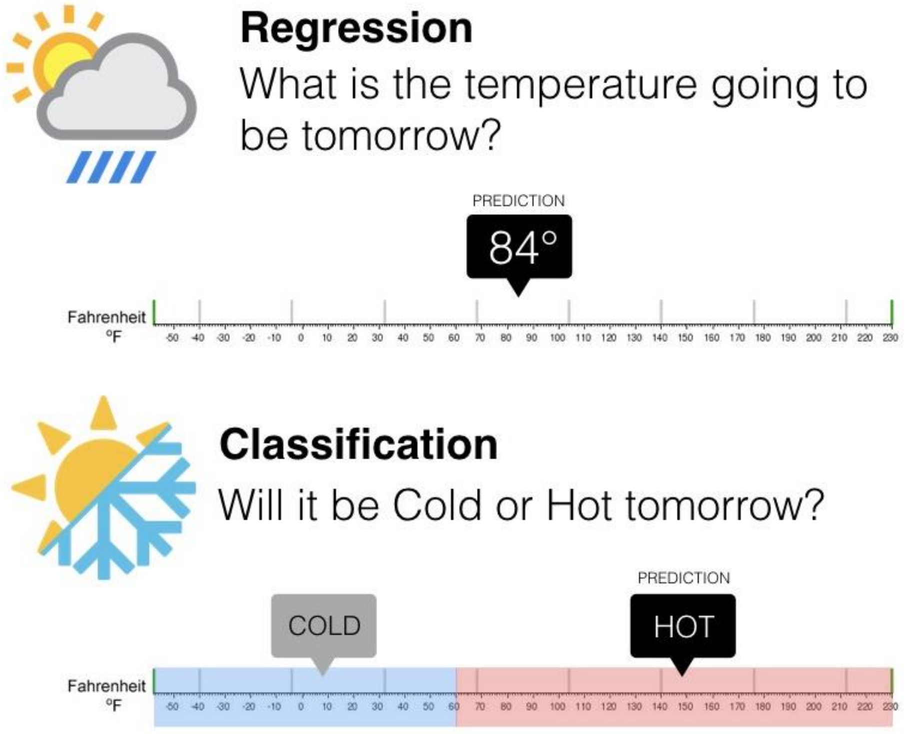
Unsupervised: You have inputs (X) but no output (Y)



# Supervised Learning: Regression vs. Classification

Regression: Aims to predict a continuous quantity

Classification: Aims to predict the group a member belongs to



# Regression

# Linear Regression

## Regressions

| | |
|---|---|
| Simple Linear Regression | $y = b_0 + b_1 x_1$ |
| Multiple Linear Regression | $y = b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_n x_n$ |
| Polynomial Linear Regression | $y = b_0 + b_1 x_1 + b_2 x_1^2 + \ldots + b_n x_1^n$ |

# Ridge & Lasso

- Extension of Linear Regression
- Places a penalty on less important features
- Ridge Regression: Shrinks coefficients of the unimportant features
- Lasso Regression: Pushes the coefficients of unimportant features to “0”

$$L = \sum(y - \hat{y})^2 + \alpha \sum |m|$$

Sum of Squared Residuals

Penalty

# Classification Models

# Logistic Regression

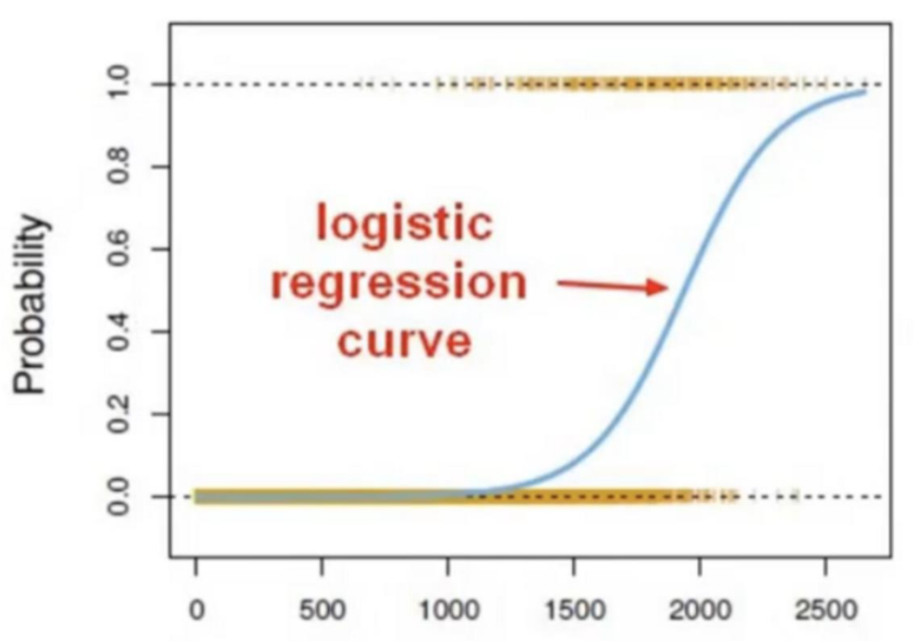
Similar to linear regression (quantitative variables in, quantitative variables out) w/ 2 key differences:

- You want to *classify*
- You have bound the range of output to between 0 and 1

$$\ln\left(\frac{P}{1-P}\right) = b_0 + b_1 x$$

$$\frac{P}{1-P} = e^{b_0 + b_1 x}$$

$$P = \frac{e^{b_0 + b_1 x}}{1 + e^{b_0 + b_1 x}}$$




Example: Predict whether a political candidate will win or lose

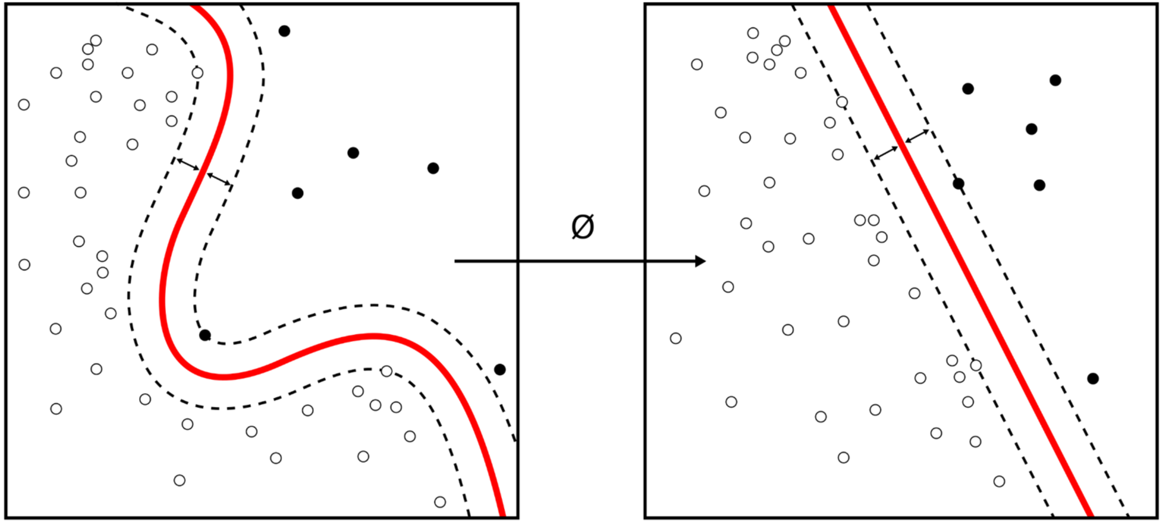
Outcome: Win/Lose (binary 0/1)

Inputs: Money spent, time spent campaigning, and poll ratings

# Support Vector Machine

- Plots samples to points in space
- Separates classes with a line

    -Works well with non-linear as well by changing the kernel



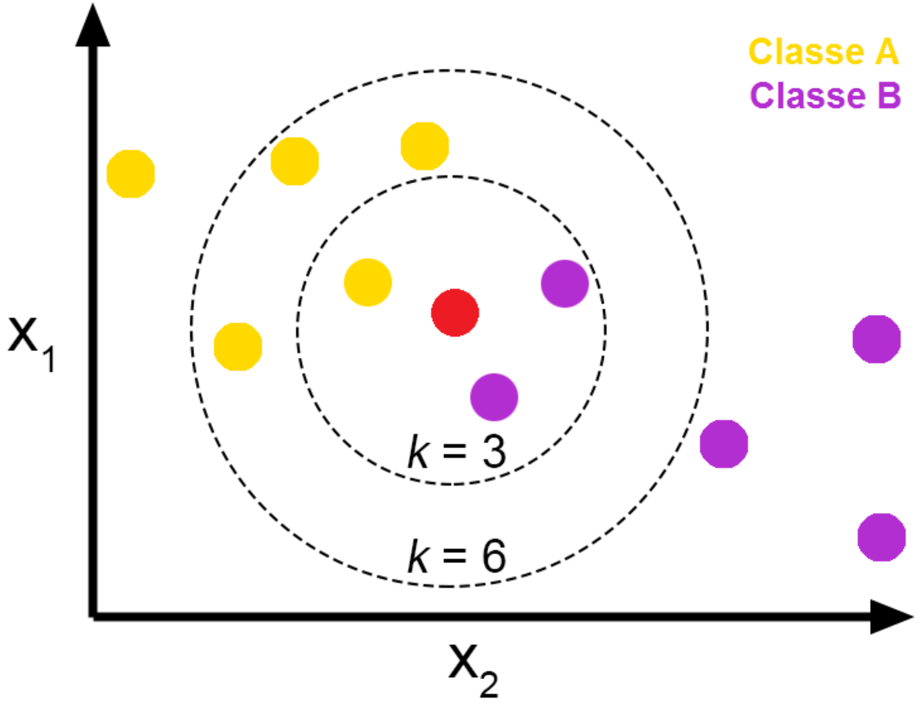
# K–Nearest Neighbors

- Plots samples to a point in space
- Counts the neighbors closest to it to determine which class if belongs to

Example:

For k = 3 Nearest Neighbors, Sample is classified as Class B

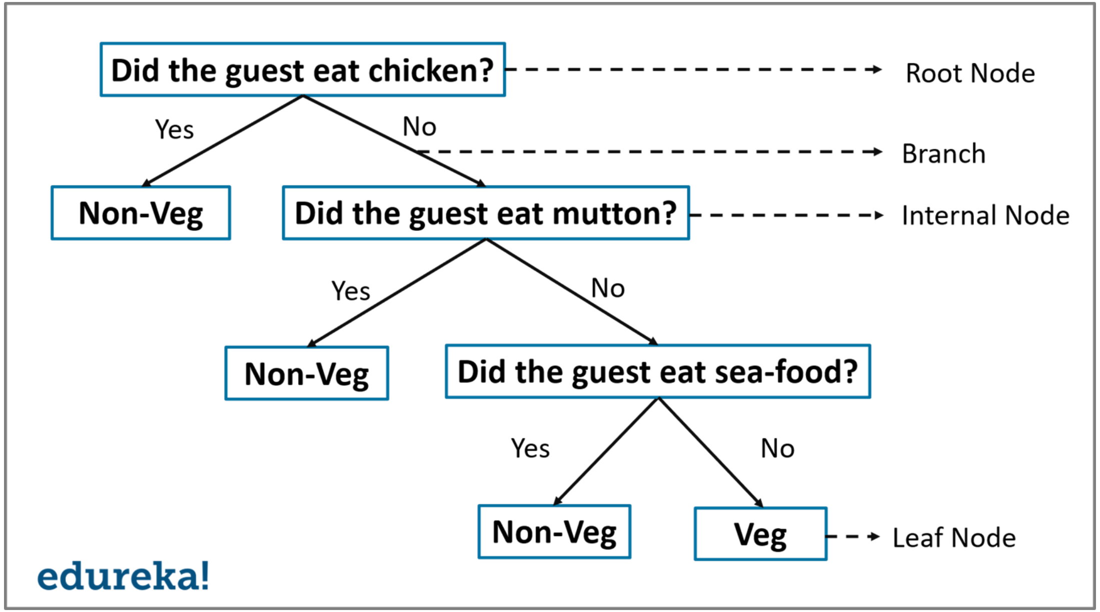
For k = 6 Nearest Neighbors, Sample is classified as Class A



# Decision Tree

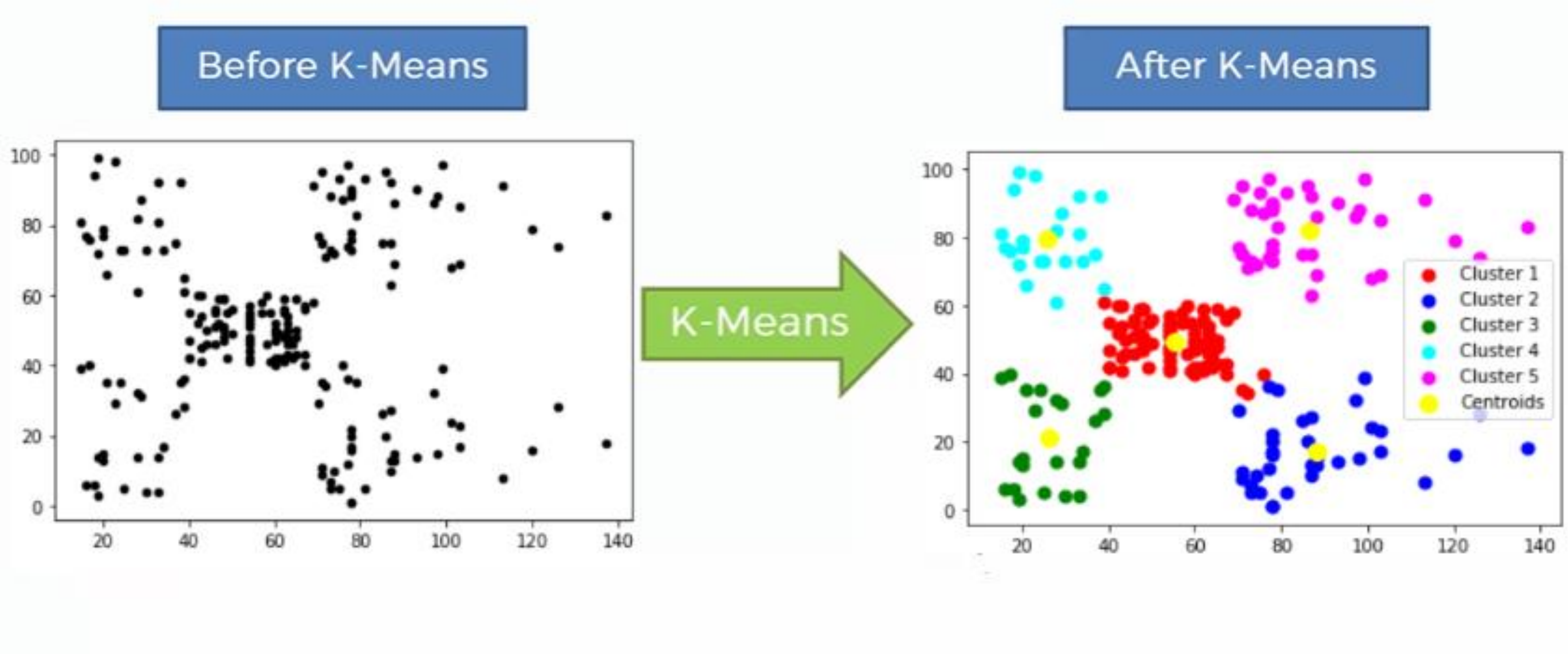
- Generally, the most important features are those used in the splits

- Upgrades:
  - Random Forest: Builds many independent trees from subsets of data and combines them
  - Boosted Models: Builds a tree, learns what it did “wrong”, builds a better one, repeat



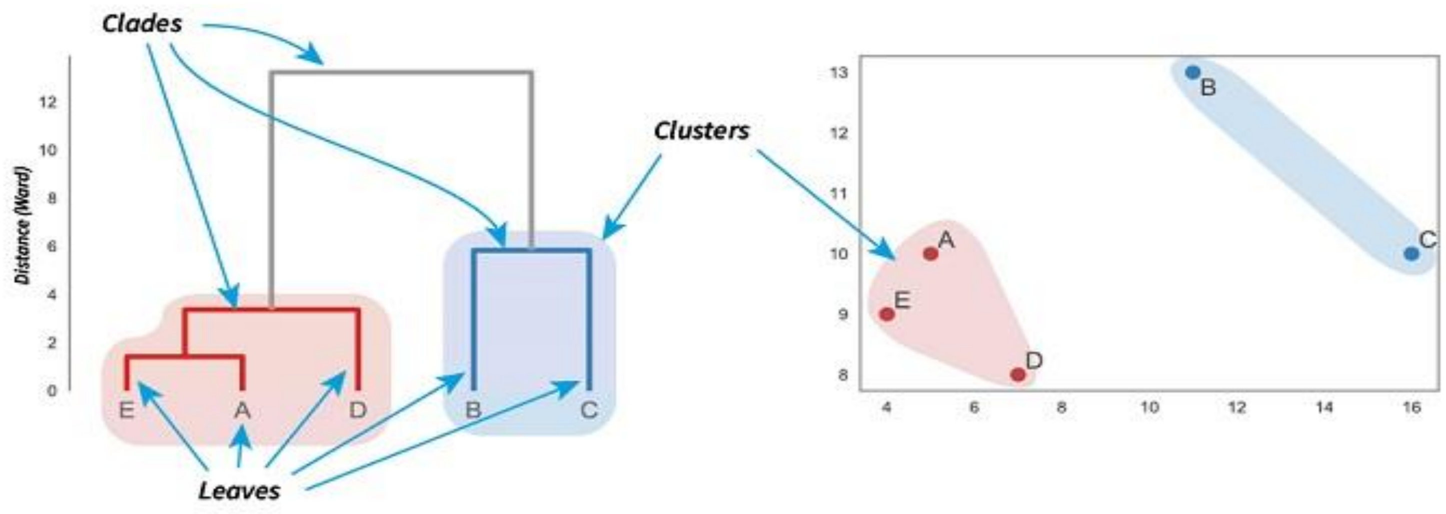
# Unsupervised Learning

# K Means Clustering



# Hierarchical Clustering

Components of dendrogram and corresponding data shown on a scatter plot



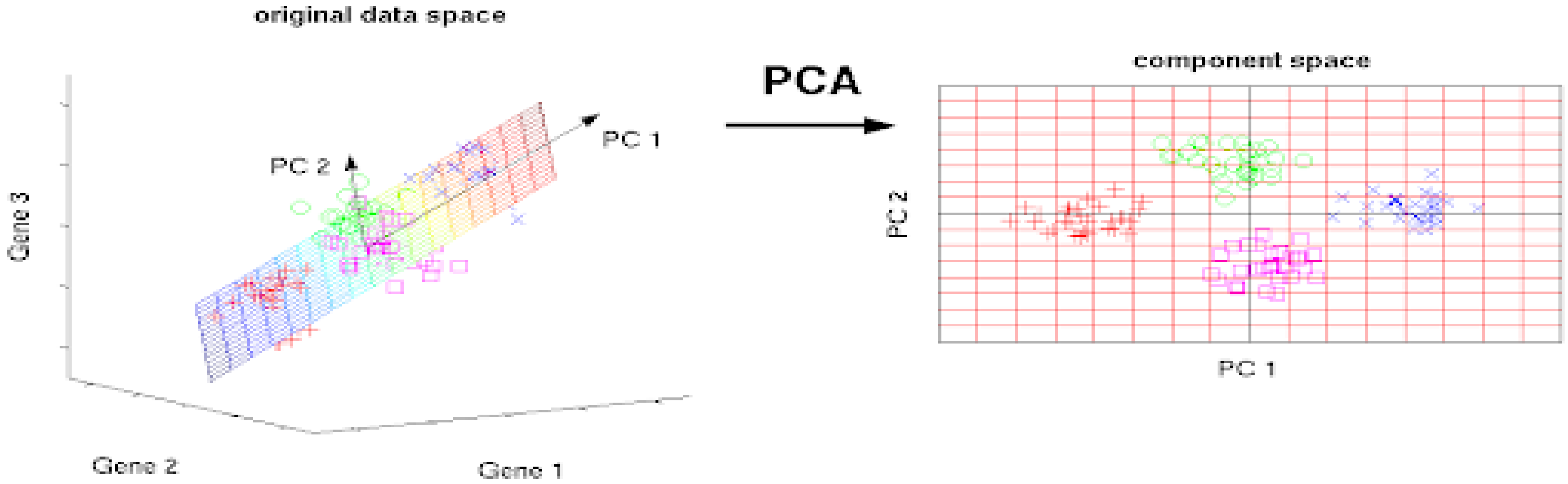

Hierarchical clustering dendrogram (Ward distance)

Scatter plot of same data with points colored by hierarchical clustering

# Principal Component Analysis

Goal: Reduce dimensionality while still keeping most of the information of the original data set

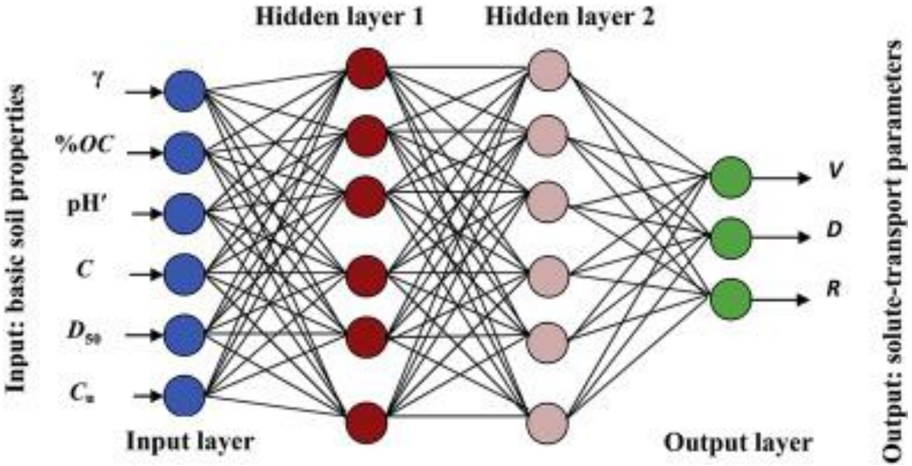
- A Principal Component is a linear combination of all or most of the initial features
- A Principal Component is *much* less interpretable than original features



# Deep Learning

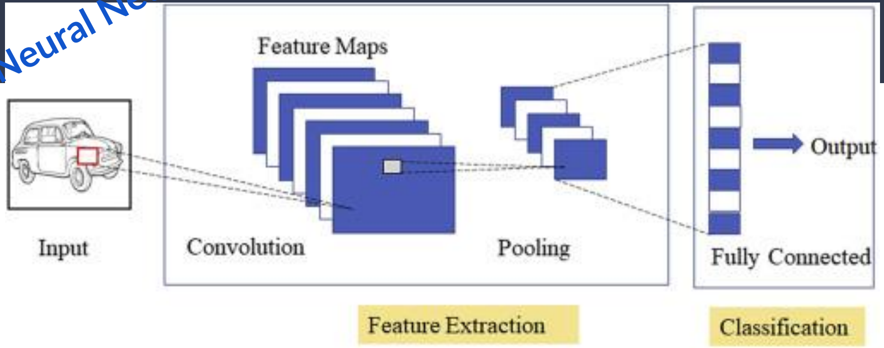
# What is Deep Learning

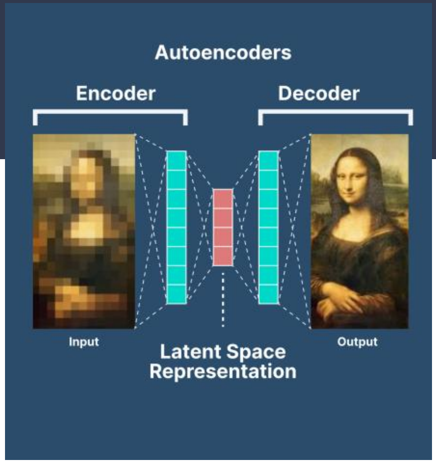
- Can perform supervised or unsupervised learning
- Formed of interconnected neurons grouped into 3 layers:
    - Input Layer: Receives input data
    - Hidden Layer: Performs mathematical computations on inputs
        - “Deep” refers to having more than one hidden layer
    - Output Layer:  Returns the result
- **Each layer transforms the data, determining its features**

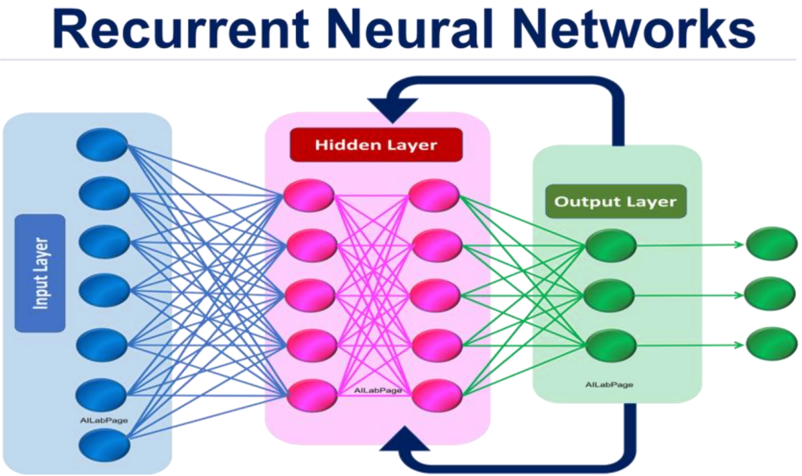


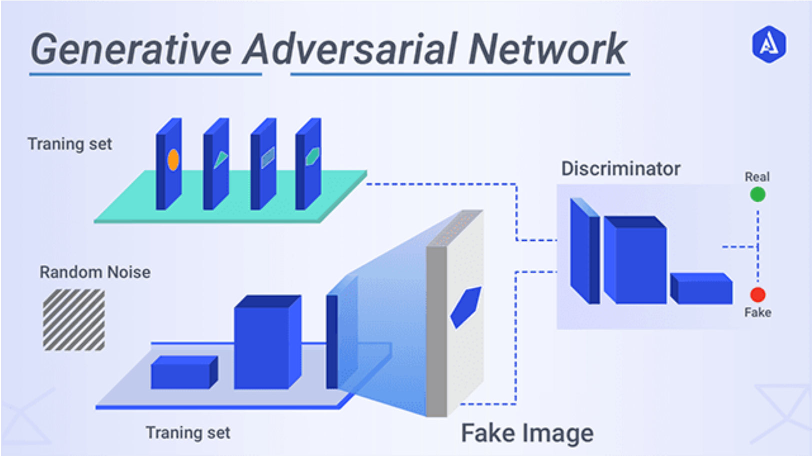
# Commonly Used Neural Network Types

Convolution Neural Network



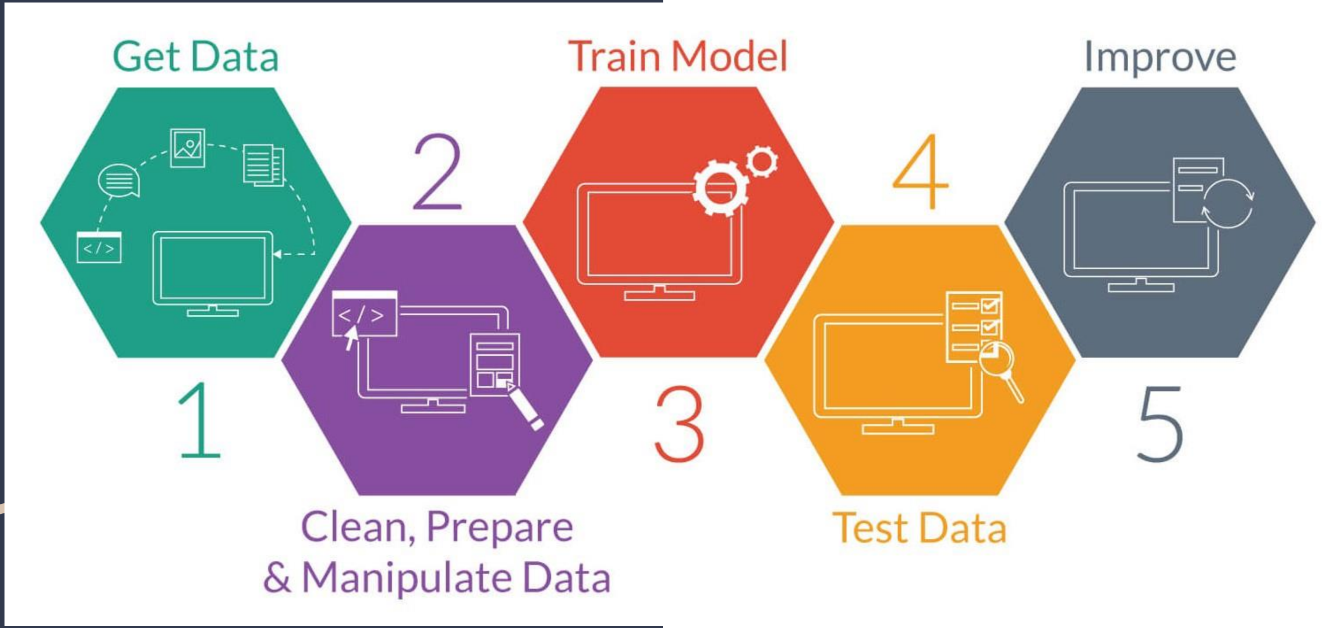



# Model Building & Performance

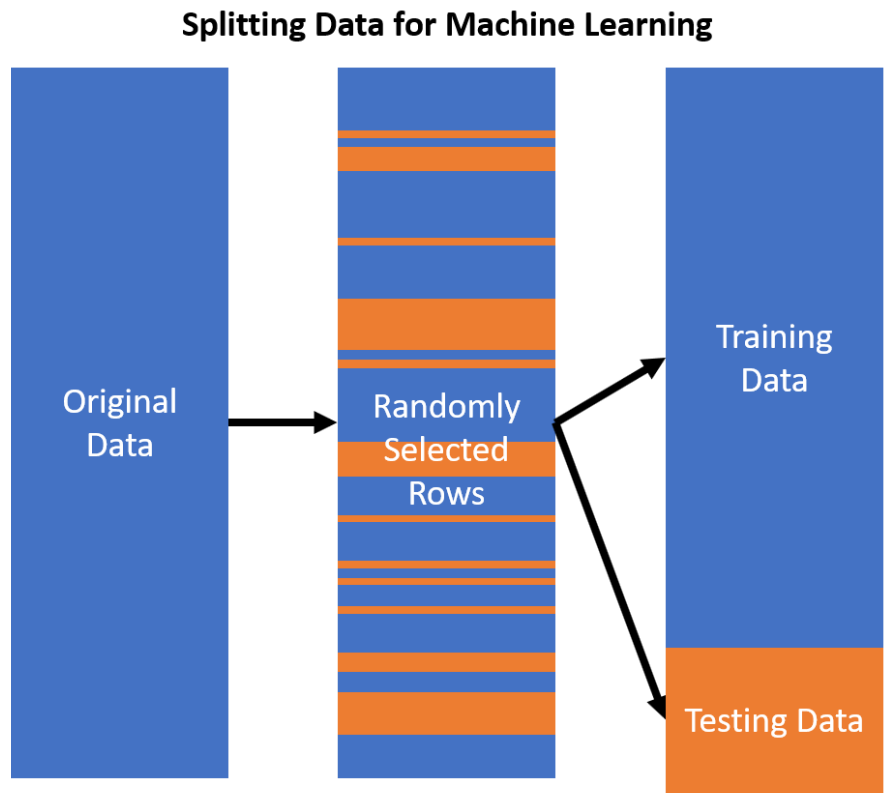
# The Machine Learning Process



# Training and Testing Sets

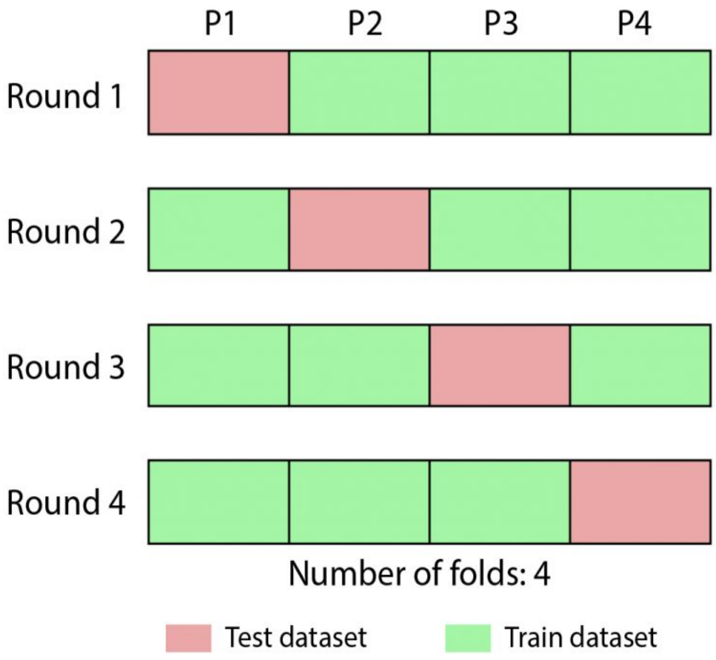
Training Set: Used to develop model

Test Set: Used to test the models accuracy



# Cross Validation

- Allows us to train and test/validate on all our data
- Gets more metrics: 92.0, **44.0**, 91.5, and 92.5
- Aids in parameter tuning



# Regression Model Accuracy

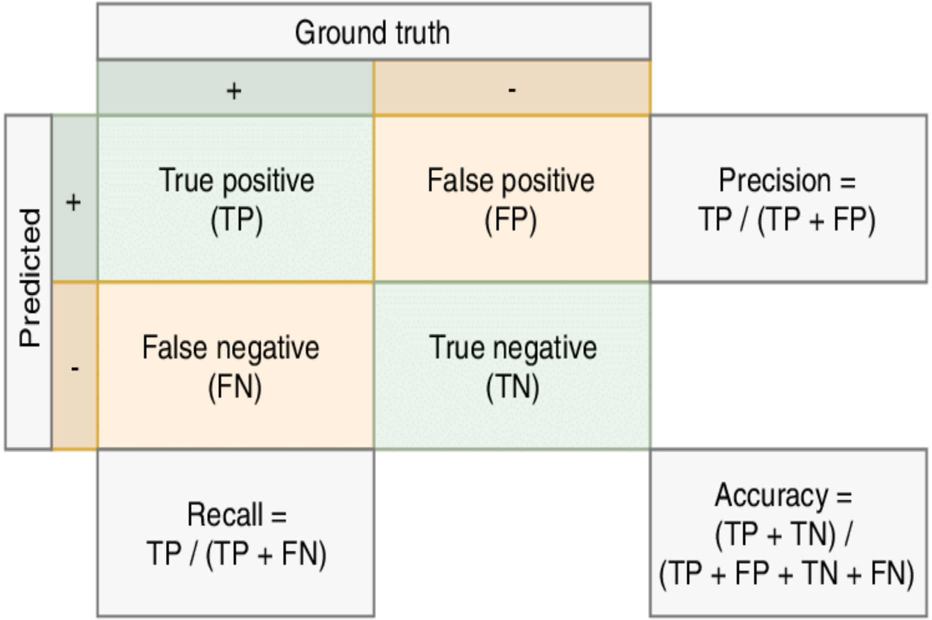
1) Mean Squared Error (MSE) - punishes models with larger errors
   - If you are accustomed to seeing an $R^2$ value, this is simply the standardized version of MSE
1) Root Mean Squared Error (RMSE) - more understandable (dollars vs dollars squared)
2) Mean Absolute Error (MAE) - Easy to understand; units are interpretable and error is linear
3) Adjusted $R^2$, AIC, BIC, Mallows Cp
   a) More robust
      i) Adding features will always decrease RMSE and increase $R^2$
   b) All unbiased estimates of the MSE

# Classification Model Accuracy

Precision: Out of all the positives predicted, how many are actually positive

Recall/Sensitivity: How good a test is at actually detecting positive cases

Specificity: Out of all the people that don't have the disease, how many got negative results

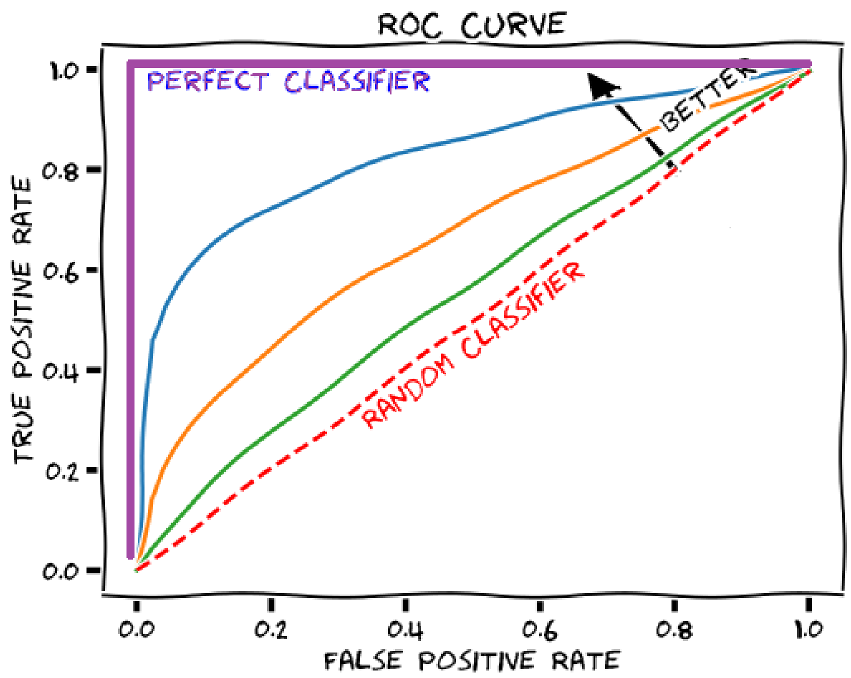


# What if your data is very unbalanced?

**AUC: Area Under Receiver Operating Curve (ROC)**

$$\text{TPR (sensitivity)} = \frac{TP}{TP + FN}$$

$$\text{FPR (1-specificity)} = \frac{FP}{TN + FP}$$



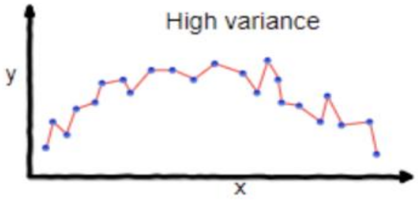
# The Bias/Variance Tradeoff

Variance: the variability of model prediction , i.e. how much the model will change if we change the training data set

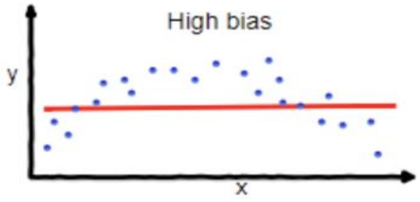
High Variance = Very complex model

Bias:  the difference between the average prediction of our model and the correct value we are trying to predict

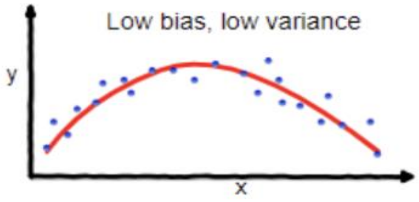
High bias =  oversimplified model




**overfitting**

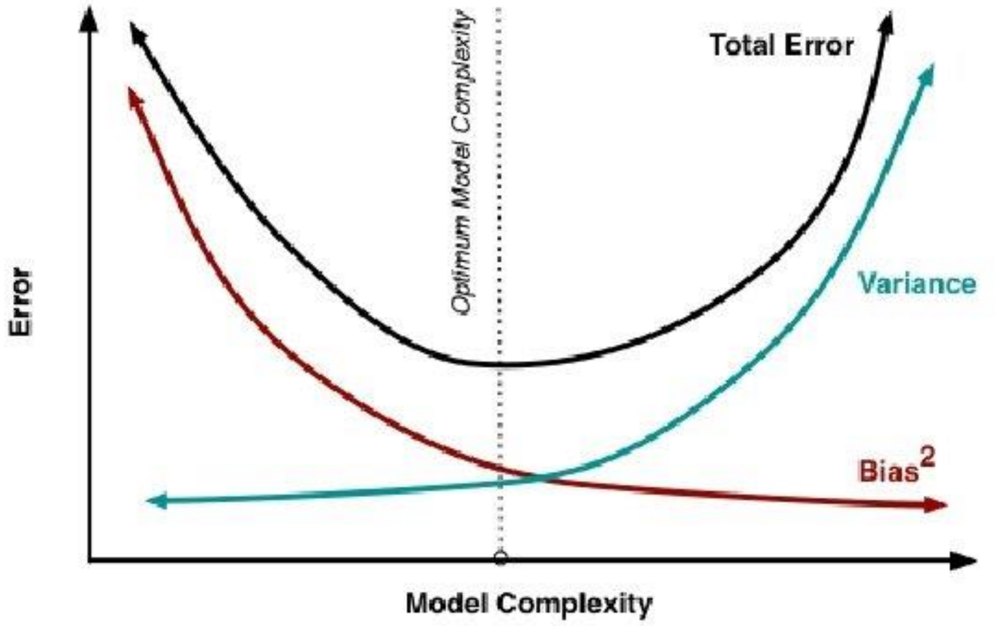



**underfitting**




**Good balance**

# Bias Variance Plot




$$Err(x) = \text{Bias}^2 + \text{Variance} + \text{Irreducible Error}$$

Image from http://scott.fortmann-roe.com/docs/BiasVariance.html

# The End :)