Machine Learning

1. Supervised
   1. Regression
      1. Linear Models
         1. Basic
         2. Regularized
            1. Lasso (L1)
            2. Ridge (L2)
            3. Elastic Net
   2. Classification
      1. KNN ("K Nearest Neighbors)
      2. Logistic Regression
         1. Basic (Binary)
         2. Multinomial
2. Unsupervised
   1. Clustering
      1. Density Models (DB Scans)
      2. Centroid Models (K Means)
      3. Connectivity Models (Hierarchical)
         1. Decision Trees
         2. Random Forests
      4. Distribution Models

Supervised: Regression: Linear Models: Basic

Def: Explanation of a continuous variable given a series of independent variables

The simplest version is just a line of best fit: y = mx + b. Explains the relationship between x and y, giving starting point b and explanation power of m.

Linear regression works best when:

* ‣The data is normally distributed (but doesn’t have to be)
* ‣X’s significantly explain y (have low p-values)
* ‣X’s are independent of each other (low multicollinearity)
* ‣Resulting values pass linear assumption (depends upon problem) 🡨["The relationship is actually linear."]

If data is not normally distributed, we could introduce bias.

General format for sklearn model classes and methods

* *# generate an instance of an estimator class*estimator = base\_models.AnySKLearnObject()
* *# fit your data*estimator.fit(X, y)
* *# score it with the default scoring method (recommended to use the metrics module in the future)*  
  estimator.score(X, y)
* *# predict a new set of data*estimator.predict(new\_X)
* *# transform a new X if changes were made to the original X while fitting*estimator.transform(new\_X)

Seaborn has a linear model plot

sns.lmplot(x, y, data, hue=None, col=None, row=None, palette=None, col\_wrap=None, size=5, aspect=1, markers='o', sharex=True, sharey=True, hue\_order=None, col\_order=None, row\_order=None, legend=True, legend\_out=True, x\_estimator=None, x\_bins=None, x\_ci='ci', scatter=True, fit\_reg=True, ci=95, n\_boot=1000, units=None, order=1, logistic=False, lowess=False, robust=False, logx=False, x\_partial=None, y\_partial=None, truncate=False, x\_jitter=None, y\_jitter=None, scatter\_kws=None, line\_kws=None)

example: sns.lmplot('bodywt', 'brainwt', mammals)

Supervised: Regression: Linear Models: Regularized

Regularization = a technique to improve the generalizability of a learned model / to reduce the possibility of overfitting the model.

Methods:

1. Lasso [least absolute shrinkage /selection operator] (L1)
   1. Method that performs variable selection and regularization
   2. Weights
2. Ridge (L2)
3. Elastic Net

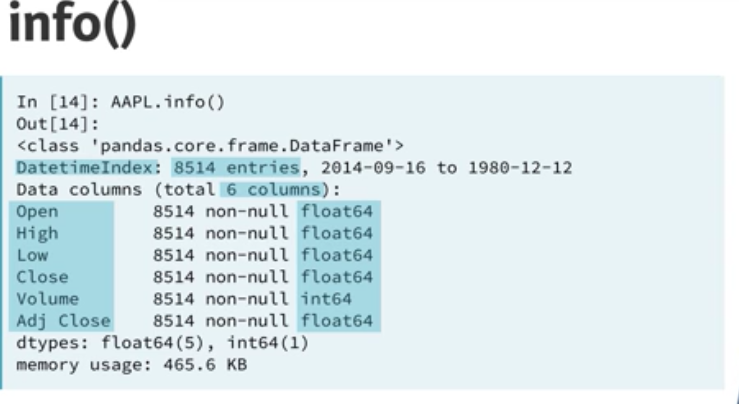
CROSS VALIDATION

k-fold cross validation

* Split the data into k groups
* Train the model on all segments except one
* Test model performance on the remaining set
* If k = 5, split the data into five segments and generate five models.

Useful Pandas DataFrame methods

.info()



Broadcasting is assigning values to cells in DF?