Decision tree(Regression)

Classification tree vs regression tree(discrete structure)

Random forest

1-pick random k point from training set

2-build decision tree associated with these points

3-choose number Nth tree to build and repeat 1&2

4- for new point predict Y value and assign to point the new average across all Y

Learning Decision Trees: A Hierarchical data structure that represents data

Generalization Vs overfitng??

LD3 Algorithm(greedy heuristics):

Data processed in Batch, build decision tree recursively



Inforamtion Gain



Logistics



**true positives (TP):** These are cases in which

we predicted yes (they have the disease), and

they do have the disease.

**true negatives (TN):** We predicted no, and

they don't have the disease.

**false positives (FP):** We predicted yes, but

they don't actually have the disease. (Also

known as a "Type I error.")

**false negatives (FN):** We predicted no, but

they actually do have the disease. (Also knownas a "Type II error.")

\**Accuracy:** Overall, how often is the classi\_er

correct?

(TP+TN)/total = (100+50)/165 = 0.91

**Misclassi\_cation Rate:** Overall, how often is

it wrong?

(FP+FN)/total = (10+5)/165 = 0.09

equivalent to 1 minus Accuracy

also known as "Error Rate"

**True Positive Rate:** When it's actually yes,

how often does it predict yes?

TP/actual yes = 100/105 = 0.95

also known as "Sensitivity" or "Recall"

**False Positive Rate:** When it's actually no,

how often does it predict yes?

FP/actual no = 10/60 = 0.17

**True Negative Rate:** When it's actually no,

how often does it predict no?

TN/actual no = 50/60 = 0.83

equivalent to 1 minus False Positive

Rate

also known as "Speci\_city"

**Precision:** When it predicts yes, how often is

it correct?

TP/predicted yes = 100/110 = 0.91

**Prevalence:** How often does the yes

condition actually occur in our sample?

actual yes/total = 105/165 = 0.64

A couple other terms are also worth mentioning:

**Null Error Rate:** This is how often you would

be wrong if you always predicted the majority

class. (In our example, the null error rate

Logistics



Recursive Feature Elimination (RFE) is based on the idea to

repeatedly construct a model and choose either the best or

worst performing feature, setting the feature aside and then

repeating the process with the rest of the features.

This process is applied until all features in the dataset are

exhausted. The goal of RFE is to select features

KKN Classification



Naïve Bayes



*P*( A | B) = *P* ( B | A ) *P* (A)

*P*(B)

NEURAL NETWORK: perceptron

gradient descent

C = ½( (Y’-Y)^2)

Stochastic Include sum