Recall or Sensitivity 1

True positive rate

TP / (TP + FN) = Positives Caught / All Positives

Out of all the (few) positive cases, how many did I find. This is about *catching* positives.

Precision 2

TP / (TP + FP) = Positives caught relative to false positives

Out of all cases I predicted as positive, how many times was I right? This is *sureness* about positives.

Specificity 3

True negative rate

TN / (TN + FP) = Negatives Caught / All Negatives

The proportion of negatives that are correctly identified.

Accuracy 4

Percent correct predictions of all predictions

(TP + TN) / (TP + TN + FP + FN)

(true positive + true negative) / (total population)

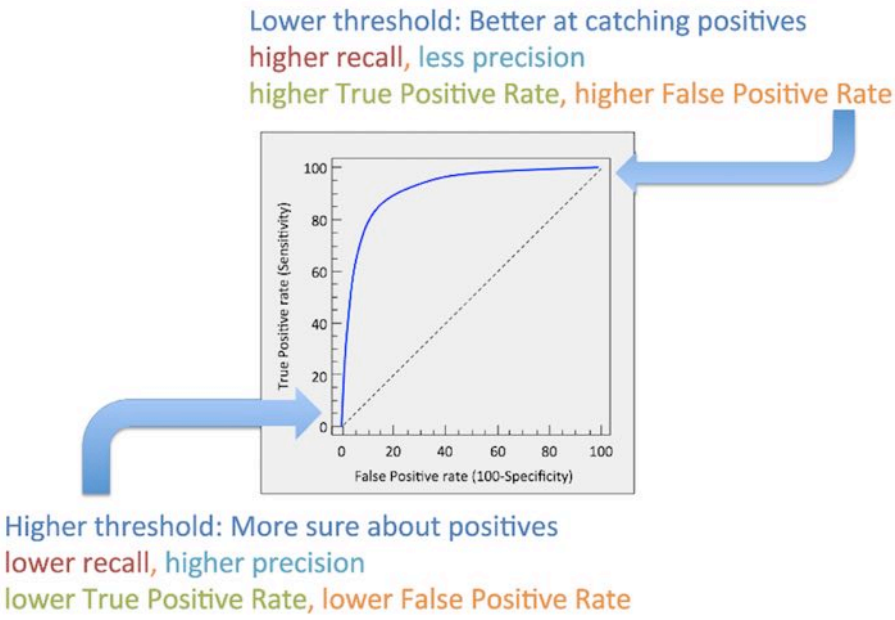
predicted / actual

F1 5

2*Recall*Precision / (Recall+Precision)

Harmonic mean of recall (catching positives) and precision (sureness of positives)

ROC Curve 6



Bayes theorem 7

Expresses the probability of A, conditioned on B.

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

posterior — likelihood — prior — evidence

Bootstrapping 8

Sampling from a dataset with replacement, to create a "new" dataset. Used in random forest.

Bagging 9

Bootstrap + Aggregating. Combining the predictions of decision trees, each of which was trained on a bootstrapped dataset

Random forest 10

Uses bagging (bootstrap + aggregating), but introduces more randomness by using a random subset of features for each tree. Excess trees will not overfit.

Boosting 11

Fits successive trees to residuals with incorrect predictions upweighted based on smooth loss function. Successive trees scaled by learning rate lambda < 1.0. Fits to entire dataset. Can overfit.

Maximum likelihood estimation 12

Select the parameter that makes the observed data "most likely,", i.e. maximizes the probability of obtaining the data at hand. To solve:

1. Write Likelihood = product of probabilities
2. Take log and simplify
3. Take derivative wrt parameter, set to zero, solve for parameter

Bias-variance tradeoff 13

The **bias** is error from erroneous assumptions in the learning algorithm. High bias can cause an algorithm to miss the relevant relations between features and target outputs (**underfitting**).

The **variance** is error from sensitivity to small fluctuations in the training set. High variance can cause an algorithm to model the random noise in the training data, rather than the intended outputs (**overfitting**).

Need to manage both so that model generalizes well.

Lasso regression 14

Adds the L1 norm (magnitude of coefficients) to the linear regression cost function, for regularization. Tends to produce sparse solutions, with many coefficients zeroed.

Ridge regression 15

Adds the L2 norm (sum of the squares of coefficients) to the linear regression cost function, for regularization. Tends to produce dense solutions, and smoother functions.

ElasticNet 16

Combines lasso and ridge: both the L1 and L2 norm are included in the linear regression cost function.

Regularization 17

Reduces overfitting by favoring simpler models over complex ones, usually by controlling the size of the parameters used.

Binary classification outcomes 18

True Negative case was negative and predicted negative

True Positive case was positive and predicted positive

False Negative case was positive but predicted negative

False Positive case was negative but predicted positive

p-value 19

The p-value for each term tests the null hypothesis that the coefficient is equal to zero (no effect). A low p-value (< 0.05) indicates that you can reject the null hypothesis.

SVM (Support Vector Machine) 20

Linear classifier for binary classification.

Finds optimal separating hyperplane that has maximum margin. Margin = distance between closest data points to the separator, margin is "no man's land," no data point inside margin.

Logistic regression 21

Classification algorithm which uses the logistic function whose values are constrained to [0,1].

K-nearest neighbor 22

Classification algorithm that predicts based on the most common class among K-nearest nodes.

K-means clustering 23

Randomly initialize cluster centers or K-means++, with each point belonging to the nearest center. Move centers to each cluster mean; reassign points. Continue until centers stop moving.

K-means++ is an initialization method that creates cluster centers with prob increasing with distance^2 to first point. Still random, but ensures centers are spaced out.

Generalized Linear Models 24

Response variable y, explanatory variables x1 x2 x3 ..., link function g

$$s = \sqrt{\frac{\sum (x - \bar{x})^2}{n - 1}}$$