Classification
Supervised Learning

- **Classification** is a supervised learning problem.
- Given a labeled set of training data $D = \{(x_i, y_i) \mid i = 1, \ldots, n\}$, where each $x_i$ is an instance (i.e., a data point) and each $y_i$ is a label, the goal of a supervised learner is to learn a function from instances to labels, so that it can appropriately label new instances that are not a part of the training data.
Classification

- **Regression** is also a supervised learning problem
- The only difference is: the labels in a regression problem are quantitative, while the labels in classification are categorical
- Consequently, the goal of a classifier is to classify!
  - i.e., assign a category to each observation
- There are many, many, many classification algorithms
  - [https://en.wikipedia.org/wiki/Category:Classification_algorithms](https://en.wikipedia.org/wiki/Category:Classification_algorithms)
- Why? Because there is no “best” algorithm
Evaluation Metrics

The results of a binary classification task can be depicted in matrix form, in a confusion matrix:

- **Accuracy** is the ratio of correct classifications to all classifications: \( \frac{TP + TN}{P + N} \)
- **Error** is the ratio of incorrect classifications to all classifications: \( \frac{FP + FN}{P + N} \)
Evaluation Metrics

The results of a binary classification task can be depicted in matrix form, in a confusion matrix.

- **Sensitivity** is the true positive rate: proportion of positives correctly identified as such (TP / P’)
- **Specificity** is the true negative rate: proportion of negatives correctly identified as such (TN / N’)
- **Precision** is the proportion of true positives among all predicted positives (TP / (TP + FP))
- **Recall** is the proportion of true positives among all actual positive (TP / (TP + FN))
Overfitting

- Problem: Models are always biased towards training data
- A model overfits when it “memorizes” the training data
- Overfit models cannot generalize to test data
- Solution: Use test data to evaluate models to mitigate the risk that they overfit
Evaluating Classifiers

- Partition our training data into a large training set and smaller testing set
- Train a classifier on the training data
- Test accuracy on the testing data
Cross validation

- **KEY IDEA: Partition multiple times**
  - If you want to partition your data 10 times, create 10 folds, and then use each fold as a test set, and the rest of the data as a training set
  - Average accuracy across all partitions to approximate model accuracy

- **This is cross validation**
  - Typical to use $k$ partitions for $k$-fold cross validation (usually $k = 10$)
  - Data are often shuffled first (e.g., if they were compiled by different sources)