3.3 Regression

Linear Regression

- **Simple case**: 2 variables
- **Linear equation**: $y = mx + b$, m = slope of the line, b = intersection of the line with the yy axis
- **Estimation of parameters**:
  - The estimation for m should be statistically significantly different from 0
  - The estimation for b may or may bot be statistically significantly different from 0
- Assume errors are independently and identically distributed, with constant variance.

Evaluation of Regression Models

Prediction

Given the value of x, the model estimates the value of y ($y = mx + b$) but the estimate is not perfect

- Error: $\sigma - y$, σ = value estimated, y = true value

Analysis of Evaluation Measures

- Do not use mean error
- **Mean absolute error**: estimates "typical" error
- **Mean squared error**: assigns more weight to larger error (may be dominated by a few cases)
- Values depend on the scale of the target variable

Baseline: trivial model

- Trivial model is assuming the mean value for every instance;
- Regression is only useful if its error is lower than the one obtained with the trivial prediction

Other Algorithms

Nearest Neighbor Algorithm for Regression

- fin kNN

- predict the average of their target values (instead of majority voting)

Decision Trees for Regression

- Train (splitting criterion based on the sum of the variances, instead of gini or entropy)
- Prediction (average of targets in the leaf, instead of majority voting)
- Variants
  - Model trees (using MLR or KNN in the leaves instead of the average)
  - MARS (multivariate adaptive regression splines)

Neural Nets for Regression

- Single output node (predicted y = score)
- Continuous activation function

SVM for Regression

- Margin: minimize the tube "around" the data (instead of maximizing the distance to the closest examples from each class)

Bias vs. Variance

- **Bias**: type of model an algorithm is able to learn given a set of training data; related to hypothesis language (e.g. linear vs. quadratic)

- **Variance**: variation in model an algorithm is able to learn, given different training data (e.g. small changes)

- **Bias-Variance trade-off**: low bias implies high variance and vice-versa

[< Go back]