Statistical Learning

Problem Description

- Given a set of data and set of hypotheses
- Predict next result based on past data
- Each hypothesis gives a chance of producing different data

Bayes' Theorem:

$$P(H \mid D) = \frac{P(H)P(D \mid H)}{P(D)}$$

Problem Description

- Given a set of data and set of hypotheses
- Predict next result based on past data
- Each hypothesis gives a chance of producing different data

Bayes' Theorem:

$$P(H \mid D) = \frac{P(H)P(D \mid H)}{P(D)}$$

Bayes' Theorem

$$P(H \mid D) = \frac{P(H)P(D \mid H)}{P(D)}$$

$$\begin{aligned} P(H) &= 0.0001 \\ P(D \mid H) &= 0.99 \\ P(D) &= 0.01 \end{aligned}$$

$$P(H \mid D) =$$

If you don't have $P(D)$:

$$P(D) = P(D \mid H)P(H) + P(D \mid \neg H)P(\neg H)$$

Bayes' Theorem

$$P(H \mid D) = \frac{P(H)P(D \mid H)}{P(D)}$$

$$\begin{aligned} P(H) &= 0.0001 \\ P(D \mid H) &= 0.99 \\ P(D) &= 0.01 \end{aligned}$$

$$P(H \mid D) =$$

If you don't have $P(D)$:

$$P(D) = P(D \mid H)P(H) + P(D \mid \neg H)P(\neg H)$$

Naive Bayesian Model

Finding the probability that hypothesis H is correct given data D:

$$\begin{aligned} P(D \mid H) &= P(d_0, \ldots, d_n \mid H) \\ &= \prod_i P(d_i \mid H) \end{aligned}$$

Bayesian Learning

Make a prediction based on all hypotheses:

$$P(X \mid d) = \sum_i P(X \mid h_i) P(h_i \mid d)$$

Variations on Bayesian Learning:

- Pick the most likely hypothesis to make predictions – *maximum a posteriori*
- Penalize complicated hypotheses – useful with many H
- Assume each H has equal likelihood: predict with *maximum-likelihood*

Ensemble Learning

Description

- Use multiple instances of a learning model
- Combine predictions with voting or average
- Reduces bias, reduces variance
- Good for distributed computing
- Flavors of ensemble learning:
  - Bagging
  - Stacking
  - Boosting

Bagging

- Generate k smaller (overlapping) training sets
- Run learning model on all k training sets
- Prediction is based on the k results
  - Highest votes in classification
  - Average of all predictions in regression
- Higher likelihood that most models will be correct

Stacking

- Train entirely different learning models on same data
- Mark the training data with each model's prediction
- Use a learning model to predict based on augmented training data (input data + other models' output)
- Reduces bias

Boosting

- Gives each input example a weight w_i, initialized to 1
- Runs a learning model
- Gives higher weights to all examples that the model gets wrong
- Runs the same learning model with new weights
- Repeat until happy
- Give each hypothesis a weight in order to determine ensemble predictions