Intro to Stat Learning

Y = f(x) + ξ

**Error:**

irreversible and reversible errors.

We’ll use reversible errors

Goal:

predict and/or inference

predict = accurate model

inference = interpret coefficients, now output.

Single parameter inference.

training data = data to teach, get ḟ.

Goal: Apply a statistical learning method to the training data

to estimate unknown function ḟ : Y~ ḟ (x)

Methods: parametric / Non-parametric

Parametric:

1. Assume the model
2. 2. Fit the model

We estimate a set of parameters instead of finding ḟ (x)

Overfitting – too flexible model that follows noise too closely

Non-parametric – seek an estimate ḟ (x) that gets close to the data points without being too rough.

By avoiding assumption of a particular form ḟ, they have potential to accurately fit a wider range of possible shapes of ḟ.

form = thin-plate spline.

Disadvantage: for better fitting and accuracy it needs very large number of observations.

Smoothness – need to choose overfitting level (get rougher fit)

*Correct amount of smoothness.*

Interpetability VS flexibility trade-off:

In general, flexibility increasing, interpretability decreasing (1\x)

Choice:

1. Inference VS Accuracy
2. Too much calculation time, costs

Overfitting paradox:

to get most accuracy 🡪 the most flexible model 🡪 overfitting 🡪 the less flexible model.

Unsupervised learning:

cluster analysis, for 2 variables we can visualize but for p-var p\*(p-1) variants 🡺 automated clustering methods.

Semi-supervised – only part of responses is available (ie, too expensive to get responses).

Quantitative – regression problems.

Categorical – classification problems.

Logistical regression – often an classification but based on quantitative probability calculation.

Categorical can be properly coded.

**MSE** = SUM(1:n) **(yº - ḟ(xº))^2** / n

MSE on unseen data we want:

**MSE(test)** = Ave **(yº - ḟ(xº))^2**)🡺 **min**

U-shape of MSE for test. MSE decreases with flexibility and then begin to increase due to overfitting.

Cross-validation – if no test data.

**E((yº - ḟ(xº))^2) = Var (ḟ(xº)) + [Bias(ḟ(xº))]^2 + Var(ξ)**

bias – error from erroneous assumptions.

variance – error from sensitivity to small fluctuations m the training set.

The higher bias – worse relevant output

The higher variance – overfitting.

Bias-variance trade-off

variance – fchanges if we chance datasets

bias – approximation of much simpler problem, the error between a model and the real data.

In complicated cases a linear regression result in high bias.

At first, bias decrease faster, then at some point change of bias is very small but variance begins to grow (overfitting).

**Classification error:** E = 1/n\*SUM(1:n) I(y(i) !=f(x(i))

For test: **Ave(I(Y!=Yo)) 🡺 min.**

Bayes Classifier: Ave 🡺 min if it assigns each observation to the most likely class, given its predictor values.

**E🡪0 when P ( Y=j| X = xº) 🡪 max.**

conditional probability.

Bayes Decision Boundary : bound between region where

**P(Y = 1 | X = xº) = 0.5**

**P(Y = 0 | X = xº) = 0.5**

Bayes error rate: 1 – max(j) Pr (Y=j | X = xº)

**1 – E ( max(by j) Pr (Y=j|X))**

Bayes error rate is analogous to irreducible error.

NB: With no Y= yº no output we can NOT calculate Bayes classifier but stil Bayes classifier is a golden standard to compare other methods.

K-nearest neighbors [KNN] classifier – estimation of comditional distribution Y with highest estimated probability.

1. Get K points closest (\*) to xº and calculated Pr (Y = j | X = xº) = 1\k SUM (No) I (y(i) = j)

(\*) iff metrics: Euclid – square, Manhattan --abs, SQRT(g) ()^g

No connection between training and test errors: overfitting train error = 0, test overfitted – increased significantly.