Supervised Learning Notes

Contents

Chapter 1: Introduction

1. Statistical learning
   (a) Refers to a vast set of tools for understanding data. Grouped into 2 categories: Supervised vs unsupervised
   (b) Supervised learning builds a statistical model for predicting an output based on one or more inputs. Both inputs and outputs must be given to build the model. Once built the model can predict an output on any input. Target goal is clear. Many ways to quantify accuracy.
      - Want to estimate relationship
      $$Y = f(X) + \epsilon$$
      for X inputs (features), Y output (response), f an unknown function to approximate, and ϵ random error which cannot be explained.
      - Prediction is key, but statistics give certainty.
      - Statistical inference helps understand how X effects Y.
   (c) Unsupervised learning only has inputs, no outputs. The model learns relationships and structure from such data. End goal is more fuzzy. Difficult to measure success.
2. Examples: Supervised vs unsupervised learning
   (a) Supervised learning: House price from size, car MPG from weight, Twitter sentiment, bird identification, student dropout
   (b) Unsupervised learning: Types of students, running gaits, topic modeling, GoT network discovery
   (c) Which is it? Music recommender, self driving car, customer segmentation
3. Statistical learning vs Machine learning (much overlap, different POV)
   (a) Statistical learning
      - Origin: Statistics and classic regression.
      - Goal: Understand the relationship between variables and provide interpretable models by extending classical statistics to flexible, data-driven models.
      - How does X effect Y, and how confident are we in that effect?
   (b) Machine learning
      - Origin: CS and AI
      - Goal: Maximize predictive performance, often on unseen data
      - How well can we predict Y from X, regardless of how and why? Though sometimes how and why.
4. History of statistical learning
   - Early 1800s, least squares and linear regression
   - 1936, linear discriminant analysis
   - 1940s, logistic regression
   - 1970, generalized linear model
   - 1980s, nonlinear models such as trees, generative additive models, neural networks
   - 1990s, SVMs
5. Matrix basics and notation for the text

Chapter 2: Statistical learning

2.1: What is statistical learning?

1. Probability crash course: Discrete case only, concrete example of dice roll.

   (a) Sample space S is the set of all possible outcomes. Die sample space $S = \{1, 2, 3, 4, 5, 6\}$.

   (b) An event $E \subset S$ is a subset of outcomes of the sample space S. Probability of subset of equally likely outcomes is given by counts. That is, for event $E = \{1\}$ is

   $$P(E) = \frac{|E|}{|S|} = \frac{1}{6}.$$

   Probability of event $E = \{2, 4, 6\}$, an even roll is

   $$P(E) = \frac{|E|}{|S|} = \frac{3}{6} = \frac{1}{2}.$$

   (c) Basic rules:
   - $0 \leq P(E) \leq 1$
   - $P(S) = 1$
   - $P(E^c) = 1 - P(E)$
   - $P(A \cup B) = P(A) + P(B) - P(A \cap B)$
   - For independent events, $P(A \cup B) = P(A) + P(B)$

   (d) The conditional probability of event A given B is

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

   which can be seen from the equally likely event case and a Venn diagram.

   $$P(A \mid B) = \frac{|A \cap B|}{|B|} = \frac{|A \cap B|/|S|}{|B|/|S|} = \frac{P(A \cap B)}{P(B)}$$

   For the dice, $P(4|even) = 1/3$.

   (e) Events A and B are independent if

   $$P(A \cap B) = P(A) \cdot P(B),$$

   which can be seen from conditional probability. 2 dice or flipping 2 coins is a better example.

   (f) A discrete random variable X assigns numbers to outcomes. Apparent for dice roll outcome. Could flip 2 coins and counts heads coding outcomes as $\{0, 1, 2\}$. This allows us to enumerate outcomes bringing in use of mathematics and functions.

   (g) The probability mass function $p(x) = P(X = x)$ gives the probability that X has value x, allowing for unequally likely outcomes. Note $\sum_x p(x) = 1$. For the dice, $p(x) = \frac{1}{6}$ for all outcomes.

   (h) The expected value (mean) of discrete random variable X is

   $$\mathbb{E}[X] = \sum_x x\, p(x),$$

   which represents the long-run average outcome. For the die roll, $\mathbb{E}[X] = 3.5$.

(i) The variance of X measures spread around the mean.

where $\mu = \mathbb{E}[X]$ which can be simplified as

$$\text{Var}(X) = \mathbb{E}[(X-\mu)^2] = \sum_x (x-\mu)^2\, p(x) = \cdots = E(X^2) - [E(X)]^2.$$

Variance is nice from a math perspective and good for algebraic properties. For our die, $\text{Var}(X) = 2.916667$ (dice value squared).

(j) The standard deviation is the square root of variance, an alternate measure of spread.

$$\sigma = \text{SD}(X) = \sqrt{\text{Var}(X)}.$$

Standard deviation is messy mathematically, but it is in the same units of X and represents interpretable spread (68-95-99 rule for a normal distribution). For the die, $\sigma = 1.707825$ meaning 66% (2-5) lie within one standard deviation.

(k) Expected value of a function $g(X)$ is

$$\mathbb{E}[g(X)] = \sum_x g(x)\, p(x).$$

2.1: What is statistical learning?

1. Statistical learning terminology and notation (regression case)
   (a) Quantitative output variable Y (dependent variable)
   (b) p input variables $X_1, \ldots, X_p$ (independent variable)
   (c) Writing as a single vector $X = (X_1, \ldots, X_P)$, we aim to identify relationship

   $$Y = f(X) + \epsilon$$

   where f is a fixed but unknown function and ϵ is a random error term independent of X with mean zero.
   (d) Essentially, statistical learning is a set of approaches to estimate f for the purpose of both prediction and inference.
2. Prediction:

- For $\hat{f}$ an known Why estimate of f and any input X, we predict output Y as

$$Y \approx \hat{Y} = \hat{f}(X).$$

- In this way $\hat{f}$ is treated as a black box where we are only concerned with output prediction.
- Predictions have 2 sources of error: reducible and irreducible
- Reducible error comes from $\hat{f} \approx f$ because this can be potentially reduced.
- Irreducible error comes from ϵ because this cannot be reduced due to the inherent $X - Y$ relationship.

- Prediction error in terms of expectation: Treating $f(X)$ and $\hat{f}(X)$ as constant and ϵ as a random variable with mean zero,

$$\begin{aligned}\mathbb{E}(Y-\hat{Y})^2 &= \mathbb{E}(f(X)+\epsilon-\hat{f}(x))^2 \\ &= \mathbb{E}(f(X)-\hat{f}(X))^2 + 2\mathbb{E}(f(X)-\hat{f}(X))\epsilon + \mathbb{E}\epsilon^2 \\ &= (f(X)-\hat{f}(X))^2 + 2(f(X)-\hat{f}(X))\mathbb{E}(\epsilon) + \mathrm{Var}(\epsilon) \\ &= (f(X)-\hat{f}(X))^2 + \mathrm{Var}(\epsilon) \\ &= \text{Reducible error} + \text{Irreducible error}.\end{aligned}$$

  Main property used here is linearity of expectation.

3. Inference:
   - Inference aims to explain the relationship between Y and $X_1, \ldots, X_p$ rather than predict.
4. How to estimate f?
   (a) Use **training data** to estimate f. Training data has independent and dependent variable pairs which are known.

$$\{(x_1, y_1), \ldots, (x_n, y_n)\} \quad \text{where} \quad x_i = (x_{i1}, \ldots, x_{ip})^T$$

   (b) **Parametric methods** assume a model formula with parameters to estimate.
   - Step 1: Assume a functional form of f. For example, linear regression would give

$$f(X) \approx \hat{f}(X) = \beta_0 + \beta_1 X_1 + \ldots \beta_p X_p$$

     where we need to estimate constant coefficients β_i rather than the entire function f.
   - Use the training data to fit the model.

$$Y \approx \beta_0 + \beta_1 X_1 + \ldots \beta_p X_p$$

     via a computational method such as ordinary least squares.
   - Choosing a form of f has drawbacks. Too simple is easy to interpret but may **underfit** the test data resulting in low accuracy. Too complex may have high accuracy but risks **overfitting** the test data and will not generalize well to unseen data.
   - **Non-parametric methods** do not assume a form of f. Instead rules are created which moves f closer to the test data while balancing over and underfitting.
5. Supervised vs unsupervised learning
   (a) Supervised learning uses independent variables (X) to estimate a dependent variable (Y). A model is built using labeled training data (supervised). Goal is to understand the $X-Y$ relationship better.
   (b) Unsupervised learning discovers patterns or structure in data without any labeled outputs (unsupervised). Goal is to better understand data in general.
6. Regression vs classification (both within supervised learning)

2.2: Assessing model accuracy

1. No free lunch theorem: In statistical learning (and ML), no single model works best on all problems. Instead we need to compare model performance and pick the best for the situation.
2. Measuring quality of fit:
   (a) Mean square error (MSE), most common measure:

$$MSE_{train} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

   With this notation, MSE_{train} is measured on train data.
   (b) Test MSE, for unseen test data (x_0, y_0):

$$MSE_{test} = \frac{1}{n}\sum_{i=1}^{n}(y_{0i} - \hat{f}(x_{0i}))^2 = Ave(y_0 - \hat{f}(x_0))^2$$

   (c) Fit balance: "U" shape test data fit curve as model flexibility increases
   - Simple models have both high train error and higher test error (underfitting)
   - Complex models have low train error but higher test error (overfitting)
   - Middle complexity models balance the 2 (just right)
   - The degrees of freedom in $\hat{f}$ determine the model flexibility

   (d) Cross validation is a more robust approach where train data is subsetted into many train-test pairs.
3. Bias-variance tradeoff:
   (a) Expected test MSE can be decomposed into **variance** of $\hat{f}$, squared **bias** of $\hat{f}$, and variance of error ϵ. That is,

$$\mathbb{E}\left[(Y_0 - \hat{f}(x_0))^2\right] = \underbrace{\text{Var}(\hat{f}(x_0))}_{\text{variance}} + \underbrace{\left[\text{Bias}(\hat{f}(x_0))\right]^2}_{\text{squared bias}} + \underbrace{\text{Var}(\varepsilon)}_{\text{irreducible error}}$$

   where

$$\text{Bias} = \mathbb{E}\left[\hat{f}(x_0)\right] - f(x_0).$$

   (b) Proof: Let $Y_0 = f(x_0) + \varepsilon$, where $\mathbb{E}[\varepsilon] = 0$ and $\text{Var}(\varepsilon) = \sigma^2$. We want to compute:

$$\mathbb{E}\left[(Y_0 - \hat{f}(x_0))^2\right] = \mathbb{E}\left[(f(x_0) + \varepsilon - \hat{f}(x_0))^2\right]$$

$$= \mathbb{E}\left[(f(x_0) - \hat{f}(x_0) + \varepsilon)^2\right]$$

$$= \mathbb{E}\left[(f(x_0) - \hat{f}(x_0))^2 + 2(f(x_0) - \hat{f}(x_0))\varepsilon + \varepsilon^2\right]$$

$$= \mathbb{E}\left[(f(x_0) - \hat{f}(x_0))^2\right] + 2\mathbb{E}\left[(f(x_0) - \hat{f}(x_0))\varepsilon\right] + \mathbb{E}\left[\varepsilon^2\right]$$

   Since ε is independent of $\hat{f}(x_0)$ and has mean zero, the cross term vanishes:

$$\mathbb{E}\left[(f(x_0) - \hat{f}(x_0))\varepsilon\right] = 0 \quad \text{and} \quad \mathbb{E}\left[\varepsilon^2\right] = \sigma^2$$

Now decompose the first term using:

$$\mathbb{E}\left[(f(x_0) - \hat{f}(x_0))^2\right] = \Big(\mathbb{E}[\hat{f}(x_0)] - f(x_0)\Big)^2 + \text{Var}(\hat{f}(x_0))$$

Putting it all together:

$$\mathbb{E}\left[(Y_0 - \hat{f}(x_0))^2\right] = \underbrace{\text{Var}(\hat{f}(x_0))}_{\text{variance}} + \underbrace{\Big(\text{Bias}(\hat{f}(x_0))\Big)^2}_{\text{squared bias}} + \underbrace{\sigma^2}_{\text{irreducible error}}$$

(c) To minimize expected test error, we need a statistical learning method that has both low variance and low bias, but never below irreducible error.

(d) Variance refers to how much $\hat{f}$ would change with a different set of training data. Inflexible models have low variance while flexible models have high variance.

(e) Bias refers to error introduced by approximating a real-life problem by a much simpler model. Inflexible models have high bias and flexible models have low bias.

(f) Nice picture in Ng cheatsheet.

4. The classification setting:

(a) y_i is a category now rather than being a quantity, though the bias-variance trade-off still occurs.

(b) We seek to estimate f from training observations $\{(x_1, y_1), \ldots, (x_n, y_n)\}$.

(c) Quantify the accuracy of our estimate $\hat{f}$ via the training error rate:

$$\frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i)$$

where I denotes the indicator function with value 1 if $y_i \neq \hat{y}_i$ and 0 if $y_i = \hat{y}_i$. This is the same as percent correct.

(d) Likewise, test error rate for test observations of the form (x_0, y_0) is:

$$\text{Ave}(I(y_0 \neq \hat{y}_0)$$

(e) The Bayes classifier:

- It is possible to show that test error rate is minimized, on average, by assigning each observation to the most likely class given its predictor values. That is, assign x_0 to class j if

$$Pr(Y = j|X = x_0)$$

is the largest.

- The Bayes decision boundary divides the sample space into one or more classes based on this conditional probability.
- The Bayes classifier produces the lowers possible test error rate given by

$$1 - E(\max_j Pr(Y = j|X))$$

where the expectation averages the probability over all values of X. Being that this rate is less than 1, this is analogous to irreducible error.

- The issue with Bayes classifier is that these conditional probabilities cannot be computed in practice. Instead, this is an unattainable gold standard we compare other methods to.

(f) K-nearest neighbors:

- Idea is to estimate the conditional probability of a Bayes classifier as an average of the K nearest training data.

$$Pr(Y = y_i|X = X_0) = \frac{1}{K}\sum_{i\in N_0} I(y_i = j)$$

Then classify according to the largest of these probabilities over all classes j.

- The choice of K is essential. Small K is overly flexible resulting in low bias but high variance. Large K is less flexible giving high bias and low variance. Later we will have guidance to the best choice of K.

Lab: Intro to R

Homework

Chapter 3: Linear regression

3.1: Simple linear regression

1. We predict a quantitative response Y from a single predictor variable X assuming a linear relationship

$$Y \approx \beta_0 + \beta_1 X$$

where the intercept and slope (β_0, β_1) are learned coefficients or parameters.

2. We use our training data to produce estimates $\hat{\beta}_0, \hat{\beta}_1$ giving the prediction

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x.$$

3. Estimating the coefficients:

(a) Given the training set of n observation pairs

$$(x_1, y_1), \ldots, (x_n, y_n),$$

our goal is to obtain coefficient estimates $\hat{\beta}_0, \hat{\beta}_1$ such that the resulting line is as close to the training data as possible.

(b) There are many ways to measure closeness, but the most natural and common choice is to minimize least squares.

(c) Denoting the ith residual as

$$e_i = y_i - \hat{y}_i$$

where

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 + x_i,$$

define the residual sum of squares as

$$RSS = e_1^2 + \cdots + e_n^2.$$

(d) Using minimization via calculus, we can compute explicitly the least squares coefficients as follows

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sum_{i=1}^{n}(x_i - \overline{x})^2}$$

$$\hat{\beta}_0 = \overline{y} - \hat{\beta}_1 \overline{x}$$

(e) Calculus proof: (also a linear algebra approach)

Model: $y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, \quad i = 1, \ldots, n,$

RSS: $\mathrm{RSS}(\beta_0, \beta_1) = \sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_i)^2.$

$$\frac{\partial \mathrm{RSS}}{\partial \beta_0} = -2\sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_i) = 0 \qquad \Rightarrow \beta_0 = \bar{y} - \beta_1 \bar{x},$$

$$\frac{\partial \mathrm{RSS}}{\partial \beta_1} = -2\sum_{i=1}^{n} x_i(y_i - \beta_0 - \beta_1 x_i) = 0 \qquad \Rightarrow \sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}) = \beta_1 \sum_{i=1}^{n}(x_i - \bar{x})^2.$$

$$\Rightarrow \qquad \hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2},$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}.$$

4. Assessing the accuracy of the coefficient estimates:

(a) Recall we assume a $X - Y$ relationship of the form $Y = f(X) + \epsilon$ where ϵ is a mean-zero random error term. Then, for linear regression this becomes

$$Y = \beta_0 + \beta_1 X + \epsilon$$

where we think of β_0 as the expected value of Y when $X = 0$, and β_1 is the average increase in Y per unit increase in X. We aim to quantify the certainty around these expected values.

(b) Basic example to connect to the notion of bias: Single mean $\hat{\mu} \approx \mu$.

- We say that $\hat{\mu}$ is an unbiased approximation of μ because small samples can either cause $\hat{\mu}$ to be larger or smaller than μ. For a very large sample, it will be exactly the same.
- This is related to the CLT.
- Recall that we define bias as

$$Bias(\hat{\mu}) = E[\hat{\mu}] - \mu$$

meaning that we have an unbiased estimate here because the CLT ensures for large number of samples $E[\hat{\mu}] = \mu$ giving zero bias.

- The standard error of $\hat{\mu}$ has a well known formula:

$$Var(\hat{\mu}) = SE(\hat{\mu})^2 = \frac{\sigma^2}{n}$$

where σ is the standard deviation of each of the realizations of y_i of Y.

- The SE formula tells us the average amount $\hat{\mu}$ differs from μ as well as the effect of the sample size n. As n increases, the SE decreases.

(c) Back to linear regression: SE formulas for the coefficient estimates.

- Formulas can be derived.

$$SE(\hat{\beta}_1)^2 = \frac{\sigma^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$$SE(\hat{\beta}_0)^2 = \sigma^2 \left(\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \right)$$

where $\sigma^2 = Var(\epsilon)$ is the variance of errors, or estimated by $\hat{\sigma}^2 = \frac{1}{n-2}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \frac{RSS}{n-2}$

- Note that $SE(\hat{\beta}_1)$ is smaller when the x_i's are more spread out. Intuitively we have more leverage to estimate the slope in this case.
- Also, $SE(\hat{\beta}_0)$ would be the same as $SE(\hat{\mu})$ if $\overline{x} = 0$ in which case β_0 would equal $\overline{y}$.
- Generally σ is unknown and estimated by the residual standard error (RSE) given above in terms of the residual sum of squares (RSS).
- Standard errors give 95% confidence intervals.

$$\hat{\beta}_1 \pm 2 \cdot SE(\hat{\beta}_1), \quad \hat{\beta}_0 \pm 2 \cdot SE(\hat{\beta}_0)$$

- Standard errors can be used to conduct hypothesis tests.

$$H_0: \text{ There is no relationship between } X \text{ and } Y$$

$$H_a: \text{ There is a relationship between } X \text{ and } Y$$

which translates to

$$H_0 : \beta_1 = 0$$

$$H_a : \beta_1 \neq 0.$$

Note, the null hypothesis translates to $Y = \beta_0 + \epsilon$ which is unrelated to X. We test with a t-statistic as with a single mean via a t-distribution with $n-2$ degrees of freedom.

$$t = \frac{\hat{\beta}_1 - 0}{SE(\hat{\beta}_1}$$

5. Assessing the accuracy of the model:

(a) Here we quantify the extent to which the model fits the data.

(b) Residual standard error (RSE):

- RSE is an estimate of the standard deviation of error term ϵ. That is, the average amount the response will deviate from the true regression line.
- RSE is computed as

$$RSE = \sqrt{\frac{1}{n-2}RSS} = \sqrt{\frac{1}{n-2}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

where RSS denotes the residual sum of squares.

- The units for RSE are the same as with Y making this measure interpretative.

(c) R^2 statistic:

- This measure normalizes to remove units to make fair model comparisons. It takes the form of a proportion of variance explained.

$$R^2 = \frac{TSS - RSS}{TSS} = 1 - \frac{RSS}{TSS}$$

where $TSS = \sum_{i=1}^{n}(y_i - \overline{y})^2$ denotes the total sum of squares.

- TSS is the total variance of Y and can be thought of as the amount of variability prior to preforming regression.
- RSS is the total variance of the regression model.
- $TSS - RSS$ is the remaining variance after regression.
- For the simple linear regression case we have that $R^2 = r^2$ where

$$r = \text{Cor}(X, Y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

though this does not extend to simple linear regression.

3.2: Multiple linear regression

1. Here we consider many predictors X_j of response variable Y.

(a) Instead of many simple linear regression models, combine into one.

$$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \epsilon$$

(b) Here, we interpret β_j as the average effect on Y of one unit increase of X_j while holding all other predictors fixed.

(c) Given estimates $\hat{\beta}_j \approx \beta_j$, we make predictions via

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 X_1 + \ldots \hat{\beta}_p X_p.$$

(d) Estimates $\hat{\beta}_j$ result from minimizing the multiple least squares regression coefficients via

$$RSS = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{n}(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{i1} + \cdots + \hat{\beta}_p x_{ip})^2.$$

(e) Explicit form of each $\hat{\beta}_j$ is messy but can be written in matrix form.

2. Important question 1: Is at least one predictor X_j useful in predicting Y?

(a) Answer via a hypothesis test.

$$H_0 : \beta_1 = \cdots = \beta_p = 0$$

$$H_a : \text{ At least one } \beta_j \neq 0.$$

(b) Conduct the hypothesis test via a F-distribution where the F-statistic is

$$F = \frac{(TSS - RSS)/p}{RSS/(n - p - 1)}$$

where as before $TSS = \sum(y_i - \overline{y})^2$ and $RSS = \sum(y_i - \hat{y}_i)^2$.

(c) If the linear model assumptions are correct, one can show:

$$E(RSS/(n-p-1)) = \sigma^2$$

and provided H_0 is true,

$$E((TSS - RSS)/p) = \sigma^2.$$

Hence no relationship between the response and predictors gives $F = 1$. On the other hand, if H_a is true, $E((TSS - RSS)/p) > \sigma^2$ giving $F > 1$.

(d) t-statistics and individual p-values exist for each β_j coefficient separately.

3. Important question 2: Do all predictors help predict Y, or only some?

(a) Finding the subset of variables associated with the response variable Y is known as variable selection. Many methods here which can produce varying results.

(b) Brute force yields 2^p possibilities, so this is not an option usually.

(c) Common options:

- Forward selection: Start with none, then add one at a time choosing the most significant first.
- Backward selection: Start with all, and remove one at a time choosing least significant first.
- Mixed (stepwise) selection: Combine both, adding and dropping variables iteratively.

(d) Selection criteria include Mallows' Cp, AIC, BIC, and adjusted R^2.

4. Important question 3: How well does the model fit the data?

(a) 2 most common measures are RSE and R^2 (fraction of variance explained).

(b) Residual standard error:

$$RSE = \sqrt{\frac{1}{n-p-1}RSS}, \quad RSS = \sum_{i=1}^{n}(y - \hat{y})^2$$

Note, RSS may decrease when adding more variables, but division with p can cause RSE to increase.

(c) R^2: Simple linear regression has

$$R^2 = \text{Cor}(Y, X)^2$$

while multiple linear regression has

$$R^2 = \text{Cor}(Y, \hat{Y})^2.$$

5. Important question 4: Given a set of predictor values, what response value should we predict, and how accurate is our prediction?

(a) 3 sorts of prediction uncertainty:

- $\hat{\beta}_i$ estimates of β_i (reducible error), can quantify uncertainty with a confidence interval for $\hat{Y}$ relative to $f(X)$.
- Linear model assumption for $f(X)$ (model bias), ignore this one and assume correct.
- ϵ (irreducible error), use prediction intervals to quantify how much Y will vary from $\hat{Y}$.

3.3: Other considerations in the regression model

1. Qualitative (categorical) predictors (also known as factors):
   (a) Variables which are not quantities need special encoding.
   (b) 2 levels: Encode one new variable as 0 or 1 (no or yes). This is known as dummy variables. Can also encode as -1 or 1 to give various meanings to intercept β_0 in the case of simple linear regression.
   (c) 3+ (k) levels: Encode $k-1$ dummy variables similar to above. One level becomes the intercept known as the baseline.
2. Extensions of linear models: These are additive (X_j independent of eachother) and linear (Y changes with X_j separately).
   (a) Removing additive assumption uses interaction terms such as product X_1X_2. Interpretation becomes more complicated.
   (b) Removing the linear term allows for other polynomial models. X^2 gives quadratic, X^3 cubic, etc.
3. Potential problems (of linear regression):
   (a) Non-linearity of the predictor-response relationships. Residual plots are useful.
   (b) Correlation of error terms. Especially an issue with time series data, but also when latent groups may occur.
   (c) Non-constant variance of error terms. Data transformation or sample weighting may help.
   (d) Outliers. Detection of outliers is possible, but it is hard to just if data should be removed to improve model fit.
   (e) High-leverage points. Large X values have a relatively large effect on model coefficients. The leverage statistic helps identify these.
   (f) Collinearity. Model accuracy is not impacted, but variable interpretation is muddy. VIF score helps identify collinear variables. Remedy is to remove one or combine into a new variable.

3.4: The marketing plan

Case study summary. Maybe good for class demo.

3.5: Comparison of linear regression with K-nearest neighbors

1. Linear regression is a parametric approach. Parameters $\beta_0, \ldots, \beta_p$ need to be found. There are many advantages here, but the main disadvantage is it makes a strong assumption on the form of $f(X)$.
2. K-nearest neighbors is non parametric. There is no assumed form of $f(X)$ giving much flexibility, but the drawback is interpretability and lack of statistical testing.
3. KNN for regression simply estimates $f(x_0)$ for new test data x_0 by averaging the K training observations which are closest. That is,
$$\hat{f}(x_0) = \frac{1}{K} \sum_{x_i \in N_0} y_i$$
4. Comparison:

Aspect	Linear Regression	KNN Regression
Model type	Parametric; assumes a linear relationship between predictors and response	Non-parametric; no assumption on functional form
Interpretability	High; coefficients have clear meaning	Low; predictions are averages of nearby points
Flexibility	Low; limited to (transformed) linear relationships	High; can capture complex non-linear patterns
Bias–Variance	Lower variance but potentially high bias if model form is wrong	Lower bias for small k but higher variance
Scalability with p	Handles moderate to high p if n is sufficient	Suffers from curse of dimensionality; needs very large n for high p
Assumptions	Linearity, independence, constant variance, normality (for inference)	Only assumes nearby points have similar responses; sensitive to scaling
When to use	Relationship is roughly linear; interpretability important	Relationship is highly non-linear; prediction accuracy prioritized

Table 1: Comparison of Linear Regression and KNN Regression

Model	Bias	Variance
Linear Regression	High if the true relationship is non-linear	Low if p is small and multicollinearity is low
KNN Regression (small k)	Low	High
KNN Regression (large k)	Higher	Lower

Table 2: Bias–variance tradeoff for Linear Regression and KNN Regression

Lab: Linear regression

Homework

Chapter 4: Classification

1. Here we predict a qualitative (categorical) variable rather than a quantitative one. This is known as classification rather than regression.
2. There are many possible classification techniques we will discuss including:
   (a) Logistic regression
   (b) Linear discriminant analysis
   (c) Quadratic discriminant analysis
   (d) Naive Bayes
   (e) K-nearest neighbors
   (f) Start of generalized linear models (Poisson regression)
   (g) More methods in later chapters (generalized additive models, trees, random forests, boosting, and support vector machines

4.1: An overview of classification

1. Just as in regression, our classification models will have a set of training observations $(x_1, y_1), \ldots, (x_n, y_n)$. Only now y_i will be a categorical variable.
2.

4.2: Why not linear regression?

1. Even if we encode random variable Y numerically such as $0, 1, \ldots$ for each category, a regression model would predict values outside the range of Y. Further, ordering categories numerically makes little sense.
2. Instead, we aim to give a meaningful estimate of conditional probability $Pr(Y|X)$.

4.3: Logistic regression

1. The logistic model:
   (a) For simplicity, consider the 2 class case encoded as 0 and 1.
   (b) How to model the relationship between $p(X) = Pr(Y = 1|X)$ and X?
   (c) We want a function $0 \leq p(X) \leq 1$ which makes sense as a probability. Many options.
   (d) The logistic function is most popular:

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}$$

   It has an "s" shape and is symmetric around point $(0, 0.5)$. Also it is defined for any value of X. Extreme values of X give probabilities close to zero. This is a "soft" version of a step function with nice math properties.
   (e) Origins of the logistic curve: For probability p, if you model odds as

$$odds = \frac{p}{1-p},$$

   then the log of odds gives

$$\text{logit}(p) = \ln\left(\frac{p}{1-p}\right).$$

   Now this transformation has range $(-\infty, \infty)$ making it suitable for linear regression.

$$\text{logit}(p) = \beta_0 + \beta_1 x \quad \Rightarrow \quad p = \frac{1}{1 + e^{-(\beta_0+\beta_1 x)}}$$

   which is the logistic function.
   (f) We will fit this logistic curve via maximum likelihood in the next section.
   (g) Considering the rewritten log-odds form

$$\ln\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X$$

   we see a unit increase in X results in changing the log-odds by β_1. Equivalently, the odds changes by e^{β_1}.
2. Estimating the regression coefficients:
   (a) Our estimates $\hat{\beta}_0, \hat{\beta}_1$ of β_0, β_1 are chosen such that they maximize the likelihood function

$$\ell(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i':y_{i'}=0} (1 - p(x_{i'})$$

Mathematically, the log-likelihood function is rather used for numerical optimization.

$$\ln(\ell(\beta)) = \sum_{i=1}^{n} [y_i \ln(p_i) + (1 - y_i) \ln(1 - p_i)]$$

where p_i is the logistic function form.

(b) Intuitively, this amounts to choosing $\hat{\beta}_0, \hat{\beta}_1$ such that plugging these into the model $p(X)$ yields a number close to one for all positive occurrences and 0 otherwise.

(c) Maximum likelihood is a general approach which can be used for many non-linear models where least squares cannot be used. In fact, lease squares is a special case of maximum likelihood.

3. Making predictions: Estimates are easy once $\hat{\beta}$ are found. Same as linear regression only now results are interpreted as probabilities. A probability threshold (such as 0.5) is use to convert probability to category.

4. Multiple logistic regression:

(a) Extension is straightforward:

$$p(X) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1 + e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}} \quad \Rightarrow \quad \ln\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

(b) Again, maximum likelihood is used to estimate parameters.

5. Multinomial logistic regression: Extend to the $K > 2$ classifier.

(a) Select the Kth class as the baseline case. Then our new model is

$$Pr(Y = k|X = x) = \frac{e^{\beta_{k0}+\beta_{k1}x_1+\ldots\beta_{kp}x_p}}{1 + \sum_{l=1}^{K-1} e^{\beta_{l0}+\beta_{l1}x_1+\cdots+\beta_{lp}x_p}}$$

for $k = 1, 2, \ldots, k - 1$, and

$$Pr(Y = K|X = x) = \frac{1}{1 + \sum_{l=1}^{K} e^{\beta_{l0}+\beta_{l1}x_1+\cdots+\beta_{lp}x_p}}.$$

(b) One can show that

$$\ln\left(\frac{Pr(Y = k|X = x)}{Pr(Y = K|X = x)}\right) = \beta_{k0} + \beta_{k1}x_1 + \ldots \beta_{kp}x_p$$

for $k = 1, 2, \ldots, K - 1$.

(c) This shows the log odds between any pair of classes is linear in the features. The odds in this case are any class compared to the Kth one.

(d) An alternative coding for multinomial logistic regression is known as the softmax coding. It is effectively equivalent to the above (same fitted values, log odds, key model outputs, etc). Instead we have no baseline class and one more β coefficient to estimate.

$$Pr(Y = k|X = x) = \frac{e^{\beta_{k0}+\beta_{k1}x_1+\cdots+\beta_{kp}x_p}}{\sum_{l=1}^{K} e^{\beta_{l0}+\beta_{l1}x_1+\cdots+\beta_{lp}x_p}}.$$

$$\ln\left(\frac{Pr(Y = k|X = x)}{Pr(Y = k'|X = x)}\right) = (\beta_{k0} - \beta_{k'0}) + (\beta_{k1} - \beta_{k'1})x_1 + \ldots (\beta_{p0} - \beta_{p'0})x_p$$

4.4: Generative models for classification

1. Logistic regression directly models the conditional probability $Pr(Y = k|X = x)$ via the sigmoid function. This is a modeling of a conditional distribution.
2. An alternate approach instead models the distribution of predictors X separately in each response class Y, then uses Baye's theorem to flip these around into estimates for $Pr(Y = k|X = x)$.
   (a) When distributions are normal, this aligns very closely with logistic regression.
   (b) This approach is called "generative" because they learn the data generating process for each class, not just a decision boundary.
   (c) Synthetic data for class can be generated from the learned distributions.
3. Why bother having another model when we already have logistic regression?
   (a) When classes are substantially separate, logistic regression is unstable. This approach solves this.
   (b) For small sample and normal distributions of X by classes Y, this approach can be more accurate.
4. Approach: Suppose Y has K classes.
   (a) Let $\pi_k = Pr(Y = k)$ denote the prior probability that an observation comes from class k.
   (b) Let $f_k(X) = Pr(X|Y = k)$ denote the density function of X for an observation that comes from the kth class. That is, $f_k(x)$ is large if there is a high probability that an observation in the kth class has $X \approx x$, and small otherwise.
   (c) Then, Bayes' theorem states
   $$p_k(x) = Pr(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}$$
   where $p_k(x)$ is the posterior probability that an observation $X = x$ belongs to the kth class. Note also that the denominator is expanded form of $Pr(X = x)$ by classes of Y as the total law of probability.
   (d) Estimating π_k is easy, just fraction of training observations in each class.
   (e) Estimating density function $f_k(x)$ is much more challenging, leading to discriminant analysis here. Once we can estimate $f_k(x)$, this approach because an approximation of the Bayes classifier described in Chapter 2.
5. Linear discriminant analysis: for $p = 1$, one predictor case
   (a) Assume $f_k(x)$ is a normal distribution (AKA Gaussian).
   $$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{1}{2\sigma_k^2}(x - \mu_k)^2\right)$$
   where μ_k and σ_k are the mean and variance parameters for the kth class.
   (b) Assumed a shared variance across all classes $\sigma = \sigma_1 = \cdots = \sigma_K$.
   (c) Pluggin into $p_k(x)$, we have
   $$p_k(x) = Pr(Y = k|X = x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma_k^2}(x - \mu_k)^2\right)}{\sum_{l=1}^{K} \pi_l \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x - \mu_l)^2\right)}$$

(d) Assign $X = x$ the class for which $p_k(x)$ is largest. Taking the natural log of $p_k(x)$, this equates to maximizing

$$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma_2} + \log(\pi_k).$$

Note this is a linear function in x.

(e) For instance, if $K = 2$ and $\pi_1 = \pi_2$, then the Bayes classifier assigns observation to class 1 if $2x(\mu_1 - \mu_2) > \mu_1^2 - \mu_2^2$ and to class 2 otherwise. The Bayes decision boundary is the point for which $\delta_1 = \delta_2$. This becomes

$$x = \frac{\mu_1^2 - \mu_2^2}{2(\mu_1 - \mu_2)} = \frac{\mu_1 + \mu_2}{2}.$$

(f) Assuming normal class distributions, we still need an estimate for μ_k and σ_k. Linear discriminant analysis (LDA) does this from the training set.

$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} x_i$$

$$\hat{\sigma} = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2$$

(g) In the end, LDA assigns the class k to $X = x$ for which

$$\hat{\delta}_k(x) = x \cdot \frac{\hat{\mu}_k}{\hat{\sigma}^2} - \frac{\hat{\mu}_k^2}{2\hat{sigma}^2} + \ln(\hat{\pi}_k)$$

is maximized. Not again the linear function of x.

6. Linear discriminant analysis for $p > 1$:

(a) Extend LDA to the case of multiple predictors, $X = (X_1, \ldots, X_p)$, by drawing from a multivariate Gaussian.

(b) Review of multivariate Gaussian:

- Assume each predictor follows a one-dimensional normal distribution with some correlation between each pair of predictors.
- If $p = 2$, we have a surface in 3D. The height of this surface at a point represents the probability that both X_1 and X_2 fall in a small region around that point.
- Notation: A p-dimensional random variable X has a multivariate Gaussian distribution is written $Z \sim N(\mu, \Sigma)$ where $E(X) = \mu$ is the mean of X (a vector of p components), and $Cov(X) = \Sigma$ is the $p \times p$ covariance matrix of X.
- Formally, the multivariate Gaussian density is

$$f(x) = \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)$$

- The kth classis drawn from its own multivariate Gaussian $N(\mu_k, \Sigma)$ where the covariance matrix is still common to each class.
- The density function becomes $f_k(X = x)$ and we assign $X = x$ a class for which

$$\delta_k(x) = x^T\Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k + \ln(\pi_k)$$

is largest.

4.5: A comparison of classification models

1. An analytical comparison: LDA vs QDA vs naive Bayes vs logistic regression

   (a) Consider these approaches in a setting with K classes, so that we assign an observation to the class that maximizes $Pr(Y = k|X = x)$. We set K as the baseline class and assign an observation to the class that maximizes

   $$\ln\left(\frac{Pr(Y = k|X = x)}{Pr(Y = K|X = x)}\right)$$

   for $k = 1, \ldots, K$. Examine this form for each method.

   (b) Method comparison:

   - Logistic regression

   $$\ln\left(\frac{Pr(Y = k|X = x)}{Pr(Y = K|X = x)}\right) = \beta_{k0} + \sum_{j=1}^{p} \beta_{kj} x_j$$

   - LDA

   $$\ln\left(\frac{Pr(Y = k|X = x)}{Pr(Y = K|X = x)}\right) = a_k + \sum_{j=1}^{p} b_{kj} x_j$$

   where a_k and b_k depend on π_k, μ_k, and Σ. Note this is a linear form just as in logistic regression

   - QDA

   $$\ln\left(\frac{Pr(Y = k|X = x)}{Pr(Y = K|X = x)}\right) = a_k + \sum_{j=1}^{p} b_{kj} x_j + \sum_{j=1}^{p} \sum_{l=1}^{p} c_{kjl} x_j x_l$$

   similar to LDA but with the quadratic interaction terms

   - Naive Bayes

   $$\ln\left(\frac{Pr(Y = k|X = x)}{Pr(Y = K|X = x)}\right) = \beta_{k0} + \sum_{j=1}^{p} \beta_{kj} x_j$$

4.6: Generalized linear models

Lab: Classification methods

Homework

Chapter 5: Resampling methods

1. 2 common resampling methods:

   (a) Cross-validation: estimate test error to evaluate performance (model assessment), or select model flexibility (model selection)

   (b) Bootstrapping: Measure accuracy of a parameter estimate or model

5.1: Cross-validation

1. Usually, test data sets are not available. We only have access to a single training set. Unfortunately, training error often dramatically underestimates test error. So, this must be remedied.
2. Here we estimate the test error by making use of a hold-out of the training data. Approaches discussed will apply to both regression and classification problems.
3. The validation set approach:
   (a) Randomly divide the training set into 2 parts: Training set and validation (hold-out) set. We fit the model on the training set and assess accuracy on the validation set as a way to approximate test error. In the end though, we still use the full data for training our final model.
   (b) Repeating this random validation set process many times gives many approximations of test error.
   (c) Drawbacks:
   - Validation estimates of test error can vary a lot.
   - Because we reduce the training set size, our model will perform worse than it would if we trained on the full data. For that reason, validation error will underestimate the true training error.
4. Leave-one-out cross-validation (LOOCV):
   (a) Reserve a single observation for the validation set. Repeat for all possible validation sets.
   (b) The test MSE is then estimated by the average of all model errors.

$$CV_{(n)} = \frac{1}{n}\sum_{i=1}^{n} MSE_i, \quad MSE_i = (y_i - \hat{y}_i)^2$$

   Note that each MSE is from a different model here.
   (c) Advantages:
   - LOOCV has less bias and tends not to overestimate the true test error.
   - Individual MSE will vary less because each model uses most of the observations as training data.
   (d) A major disadvantage is LOOCV is computationally expensive. In the case of least squares linear or polynomial regression, there is a nice result.

$$CV_{(n)} = \frac{1}{n}\sum_{i=1}^{n} \frac{(y_i - \hat{y}_i)^2}{1 - h_i}$$

   where $\hat{y}_i$ is the prediction of the full model trained on all data, and h_i is the leverage term from page 99. This is almost the ordinary MSE with correction $1/n < h_i < 1$, which reflects the amount an observation influences its own fit.
5. k-fold cross validation:
   (a) Randomly divide the set of observations into k groups (or folds). Use the first fold as validation, the rest as training. MSE_1 is the resulting test accuracy. Repeat for all k folds. Then,

$$CV_{(k)} = \frac{1}{k}\sum_{i=1}^{k} MSE_i.$$

(b) LOOCV is a special case of k-fold cross validation, though the most computational.

(c) There is a bias-variance tradeoff in the choice of k.

(d) Goal of the flexibility MSE curves is either to (1) find the minimum to identify model flexibility, or (2) estimate test error.

6. Bias-variance trade-off for k-fold cross validation:

   (a) Aside from efficiency, k fold cross validation has another advantage over LOOCV: bias-variance trade-off. With bias reduction, LOOCV has the win because models are trained with most of that data. But, for variance, k-fold CV has the lower result because its models are less correlate to to eachother.

7. Cross-validation on classification problems:

   (a) CV extends naturally here as you would expect. LOOCV error rate takes the form

$$CV_{(n)} = \frac{1}{n}\sum_{i=1}^{n} Err_i$$

   where $Err_i = I(y_i \neq \hat{y}_i)$.

5.2: The bootstrap

1. For models like linear regression, one can derive standard error (SE) formulas for confidence interval and hypothesis tests. Alternatively, bootstrapping can be used to estimate SE. This can be applied to complex models where SE formulas aren't available.

Lab: Cross-validation and the bootstrap

Homework

Chapter 6: Linear model selection and regularization

1. The standard linear model for regression is

$$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \epsilon$$

   which is typically fitted via least squares. We will keep this linear framework for now because of its many advantages (simplicity, interpretability, etc), but we will improve by considering alternate fitting techniques.

2. Advantages of alternative fitting:

   (a) Better predictive accuracy: When $n >> p$, least squares estimates have low variance and hence perform better on test observations. If n is not much bigger than p, there can be a lot of variability, resulting in overfitting, and poor accuracy. Our strategy will be to constrain (or shrink) the estimated coefficients.

   (b) Better model interpretability: When there are many variables, some may be irrelevant leading to unnecessary complexity. We remove such variables.

3. Summary of options:

   (a) Subset selection: Identify a best subset of p predictors and remove rest.

   (b) Shrinkage: Keep all predictors, but shrink the coefficients towards zero to remove variance.

   (c) Dimension reduction: Project p variables onto a M-dimensional subspace and use these instead.

6.1: Subset selection

1. Best subset selection:
   (a) Fit separate least squares regressions for each possible combinations of the p predictors. Note, there are 2^p possibilities.
   (b) Algorithm: $p+1$ possible model options.
   - Let M_0 denote the null model with no predictors.
   - For $k = 1, 2, \ldots, p$: Fit all $\binom{p}{k}$ models with k predictors. Pick the best, call it M_k using smallest RSS (or other option).
   - Select a single best model from $M_0, \ldots, M_p$ using prediction error on a validation set (or other options).
2. Stepwise selection:
   (a) Best subset selection is not feasible computationally for large p. There are also statistical issues. The larger the search space, the higher the chance of finding models that look good on the training data even though they will suffer on the test data (overfitting and high variance of coefficient estimates).
   (b) Forward stepwise selection:
   - Start with no predictors, add best predictors (most model improvement) one at a time until all predictors are in the model.
   - Reduction to $1 + \sum_{k=1}^{p-1}(p-k) = 1 + p(p+1)/2$ models.
   - Choose from all such models via cross validation.
   (c) Backward stepwise selection:
   - Start with all predictors, remove the least best predictor (least model improvement) one at a time until all predictors are in the model.
   - Reduction to $1 + \sum_{k=1}^{p-1}(p-k) = 1 + p(p+1)/2$ models.
   - Choose from all such models via cross validation.
   (d) Hybrid: Combine the 2.
   -
3. Choosing the optimal model:
   (a) How to tell which model is best? Previously, we said the best model has the smallest RSS and the largest R^2, since these qualities are related to training error. Instead, we wish to choose a model with low test error. Because training error can be a poor estimate of test error, we need a new approach.
   (b) 2 common approaches:
   - Indirectly estimate test error by making and adjustment to the training error to account for the bias due to overfitting.
   - Directly estimate the test error using either a validation set or cross validation approach.
   (c) C_p, AIC, BIC, and adjusted R^2: Adjusting the training error for the model size.
   - C_p: Estimate test MSE (recall, $MSE = RSS/n$) as

$$C_p = \frac{1}{n}(RSS + 2d\hat{sigma}^2).$$

   where $\hat{sigma}^2$ is an estimate of the variance of the error ϵ associated with each response measurement, typically using the full model containing all predictors.

- Essentially, the C_p statistic adds a penalty of $2d\hat{\sigma}^2$ to the training RSS to adjust for the fact that the training error tends to underestimate the test error. This penalty increases as the number of predictors in the model increases balancing a corresponding decrease in RSS. Once can show C_p is an unbiased estimate of the test MSE.
- AIC (Akaike information criterion):

(d) AIC (Akaike information criterion):

- AIC is defined for a large class of models fit by maximum likelihood.
- In our case, it is given by

$$AIC = \frac{1}{n}(RSS + 2d\hat{sigma}^2)$$

where for simplicity we have omitted irrelevant constants.

- Once can see AIC and C_p are proportional to eachother.
- Choose lowest AIC.

(e) BIC (Bayes information criterion):

- BIC is derived from a Bayesian point of view, but ends up looking similar to C_p and AIC.
- For the least squares model with d predictors, BIC is given up to irrelevant constants as

$$BIC = \frac{1}{n}(RSS + \ln(n)d\hat{\sigma}^2$$

- BIC places a higher penalty for models with many variables compared to C_p, resulting in the selection of smaller models.
- Choose lowest BIC.

(f) Adjusted R^2:

- Good for comparing models with different numbers of variables.
- Recall,

$$R^2 = 1 - \frac{RSS}{TSS}, \quad TSS = \sum (y_i - \overline{y})^2, \quad \text{the total sum of squares}$$

The R^2 always increases as more variables are added.

- For a least squares model with d variables,

$$\text{Adjusted } R^2 = 1 - \frac{RSS/(n-d-1)}{TSS/(n-1)}$$

- Large value of adjusted R^2 indicates a model with small test error.
- Intution is that once all of the correct variables have been included in the model, adding additional noise variables leads to only a very small decrease in RSS and the increase in d will drag down the overall value.

(g) Validation and cross-validation:

- As an alternative to these metrics, we can directly estimate the test error via validation and CV.
- Advantage is we directly estimate the test error and make fewer assumptions about the underlying model.
- The only drawback is computational efficiency.
- Can select a model using the on-standard-error rule: calculate the stard error of the estimated test MSE for each model size, then select the smallest model for which the estimated test error is within one standard error of the lowest point on the curve allowing for choice of the simplest model.

6.2: Shrinkage methods

1. Ridge regression:

   (a)

2. The lasso:

   (a)

6.3: Dimension reduction methods

6.4: Considerations in higher dimensions

Lab: Linear models and regularization methods

Homework

Chapter 7: Moving beyond linearity

7.1: Polynomial regression

7.2: Step functions

7.3: Basis functions

7.4: Regression splines

7.5: Smoothing splines

7.6: Local regression

7.7: Generalized additive models

Lab: Non-linear modeling

Homework

Chapter 8: Tree-based methods

8.1: Basics of decision trees

8.2: Bagging, random forests, boosting, and Bayesian additive regression trees

Lab: Decision trees

Homework

Chapter 9: Support vector machines

9.1: Maximal margin classifier

9.2: Support vector classifiers

9.3: Support vector machines

9.4: SVMs with more than two classes

9.5: Relationship to logistic regression

Lab: Support Vector Machines

Homework

Chapter 10: Deep learning

10.1: Single layer neural networks

10.2: Multilayer neural networks

10.3: Convolution neural networks

10.4: Document classification

10.5: Recurrent neural networks