STA 4241 Lecture, Week 6

Overview of what we will cover

- Cross validation
  - Estimating test set error rates
  - Choosing tuning parameter values
- The bootstrap
  - Creating confidence intervals or estimating standard errors

Resampling approaches

- Resampling deals with repeatedly drawing samples or subsets of the training data and estimating a chosen model for each data set
- These are extremely powerful tools in statistics
  - Modern computing power makes them easy to implement
  - Can solve problems that would otherwise be difficult
- Nearly every statistical project I've been involved in utilizes resampling methods
  - Ubiquitous in statistics

Resampling approaches

- We typically use resampling approaches when we can't solve things analytically
- There are many reasons people use resampling, but there are two hugely important ones we will see this week
  - Model selection / tuning parameter selection
  - Estimating the uncertainty of parameter estimates
- In many problems there are no closed form solutions to these issues
  - Resampling allows us to approximate unknown quantities of interest

Resampling approaches

- Throughout class, we have seen methods that have tuning parameters
  - K in the KNN approach
  - The budget for support vector machines
  - Choice of kernel for support vector machines
  - Degree of polynomial in a regression model
- We've seen that our results can be very sensitive to these choices
- We need an approach to choosing these parameters that works in many situations

Resampling approaches

- Our goal has always been reducing the test set error rates or testing MSE
- If we knew the testing MSE, we could choose the tuning parameter that minimizes the error rate
- Obviously we never have the testing error rates
  - But we can estimate them!
- Resampling is used to estimate the testing error rates

Resampling approaches

- While estimating the testing error rates is nice in its own right, a more important consequence is that we can choose a tuning parameter that minimizes our estimated test set MSE
- This provides an automated choice of tuning parameter
  - No subjectivity
  - No prior knowledge needed
- Greatly improves the usefulness and widespread applicability of methods that have tuning parameters
- Nearly all new machine learning or complex algorithms have tuning parameters that need to be chosen

Resampling approaches

- Another crucial statistical issue that utilizes resampling is understanding uncertainty
- How variable is an estimate of a prediction or an unknown parameter
  - Standard error of an estimator
  - Construction of confidence intervals
- In many cases, this can be done analytically
  - The basis of nearly all of STA 4322

Resampling approaches

- As we progress into more complex approaches (like those seen in this class), constructing confidence intervals becomes more difficult
- What do we do if an analytic expression for a standard error doesn't exist?
- Resampling can be used to estimate standard errors or construct confidence intervals
    - Without knowing distribution of data
    - Less reliance on asymptotic approximations (big sample sizes)

Cross-validation

- The first resampling approach we will discuss is called cross-validation (CV)
- Every approach we have (or will) consider in this class can utilize CV
- The main purpose of CV is to choose tuning parameters
  - Estimates test set error
  - Minimize this error as a function of tuning parameters

Cross-validation

- Why do we need resampling to do this?
- If we have a designated testing data set then we can simply evaluate performance on that data
  - Frequently not available
- Could also evaluate our model on our training data
  - Severely under-estimates testing error rates
  - Leads to overfit models
  - Incorrect tuning parameter choices

Cross-validation

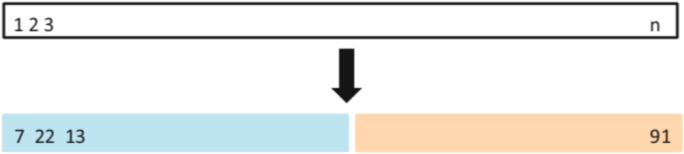
- The main idea behind CV is to leave out or hold out a portion of the data
- We now have the data split into two parts
  - Training data
  - Validation data / testing data
- We fit the model to the subset of the data that is to be used for training
- Evaluate how well it predicts on the subset of data that we held out

Cross-validation

- The most natural way to do this is split the data in half
  - The book calls this the validation set approach
- Randomly choose half of the data to be training and half to be validation
- Estimate the testing MSE as the MSE of your predictions on the validation data

Cross-validation

- Here is a visual illustration of this approach to cross validation
- Fit the model on the blue data and assess performance on the orange



James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning. New York: springer.

Cross-validation

- In principle, this should provide an estimate of the test set error rate
- There are two main problems with this approach
  - There is variability in which half of the data you choose
    - Different results for different splits
  - This will over estimate the testing error rate
- Let's look at this first issue in more detail

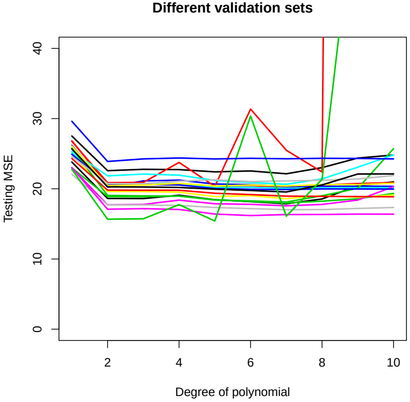
Cross-validation

- The auto data set from the book is available in the ISLR package in R
- The goal is to predict mpg using horsepower
- The data consists of 392 observations
  - Randomly split into 196 training and 196 testing data points
- Will perform polynomial regression

$$E(mpg|\text{horsepower}) = \beta_0 + \sum_{j=1}^{d} \beta_j \text{horsepower}^j$$

Cross-validation

- Will do this for a set of d values to vary model flexibility
  - Tuning parameter for the model



Cross-validation

- There is substantial variability in the error rate estimates
  - Big shifts up or down
  - Erratic points
- More importantly, the d that minimizes the testing MSE varies by validation set
  - One data set suggested $d = 2$ while another suggested $d = 10$!
- Not ideal if tuning parameter choice depends on this random process

Cross-validation

- Another issue is that we are over-estimating the true testing MSE
- Remember the formula for testing MSE

$$E[(Y_0 - \widehat{f}(\boldsymbol{X}_0))^2] = \text{Var}(\widehat{f}(\boldsymbol{X}_0)) + [\text{Bias}(\widehat{f}(\boldsymbol{X}_0))]^2 + \text{Var}(\epsilon)$$

- Generally speaking $\text{Var}(\widehat{f}(\boldsymbol{X}_0))$ is on the order of $1/n$
- In our estimates of the testing error rates, we're using half the data and therefore increasing this component of the error by a factor of 2

Cross-validation

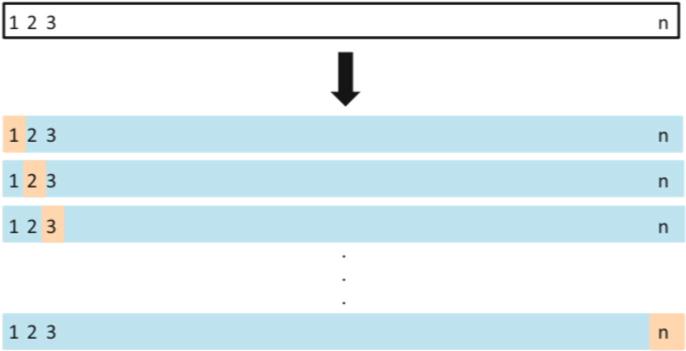
- This is only an issue if we're interested in estimating the out of sample testing performance
- If we're interested in tuning parameter estimation it isn't necessarily a huge concern
- Our error estimates might be too high, but as long as the same tuning parameter is chosen, it doesn't matter

Cross-validation

- Leave one out cross validation (LOOCV) aims to address these issues
- Instead of separating the data into two parts of size $n/2$ we split the data into $n-1$ training samples and 1 validation sample
- Fit the model on the $n-1$ training points
  - Variance is now approximately the correct order
- Do this for all n possible validation points

Cross-validation

- Here is a visual illustration of LOOCV



James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning. New York: springer.

Cross-validation

- Our estimate of the testing error rate is simply

$$\frac{1}{n}\sum_{i=1}^{n}\mathrm{MSE}_i$$

where MSE_i is the MSE for validation point i

$$\mathrm{MSE}_i = (Y_i - \widehat{Y}_i)^2$$

and $\widehat{Y}_i$ is based on the model fit on all data except data point i

Cross-validation

- LOOCV solves two of the problems from the validation set approach
  - Solution is no longer random
  - The estimate of the test set error is not overly biased due to the sample size used for model fitting
- One drawback of LOOCV is computation time
  - Need to fit n models instead of one
  - Certain methods are very slow computationally

Cross-validation

- A rather interesting result is that for least squares regression, LOOCV can be written as

$$\frac{1}{n}\sum_{i=1}^{n}\left(\frac{Y_i - \widehat{Y}_i}{1 - h_i}\right)^2$$

  where $\widehat{Y}_i$ is an estimate of the fit from the model with all of the data

- This means we only need to fit one model!

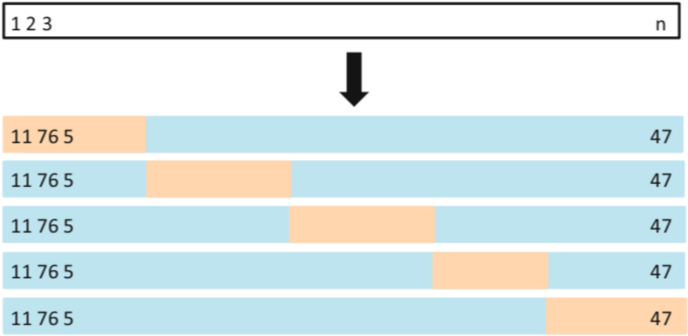
- $1/n < h_i < 1$ is the leverage and is a measure of how much that data point influences the model fit
  - Points with higher leverage need their training error inflated more

Cross-validation

- Most models do not permit such a nice representation for LOOCV
- Most models require n models to be fit
- k-fold cross validation provides an alternative
  - Middle ground between validation set approach and LOOCV
- k-fold cross validation involves splitting the data into k groups
- Fit data on $k - 1$ groups and validate on remaining group of data

Cross-validation

- Visualization of k-fold cross validation



James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning. New York: springer.

Cross-validation

- Our estimate of the testing MSE is therefore

$$\frac{1}{k}\sum_{i=1}^{k}\text{MSE}_i$$

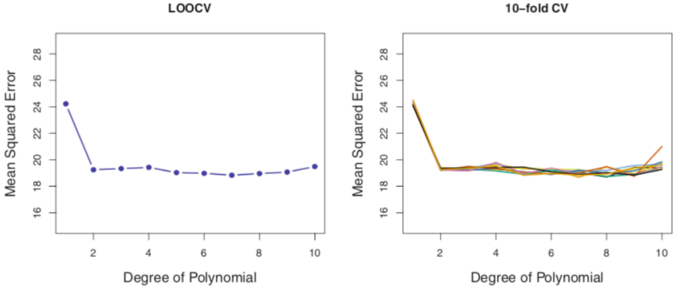
  where MSE_i is the MSE on the i^{th} validation group

- There is variability in how we split the data into k groups
  - Much less variability than the validation set approach

- Only need to fit the model k times

Cross-validation

- Let's see how LOOCV and k-fold CV work on the auto data
- We see some variability in the 10-fold cross validation estimates but it is very small
- LOOCV and 10-fold lead to similar estimates here



James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning. New York: springer.

Cross-validation

- How do we choose k in k-fold CV?
  - LOOCV is a special case with $k = n$
- Computation time is not the only concern
- There is a bias-variance trade-off that comes with the choice of k
- Interestingly $k < n$ can give more accurate estimates than LOOCV

Cross-validation

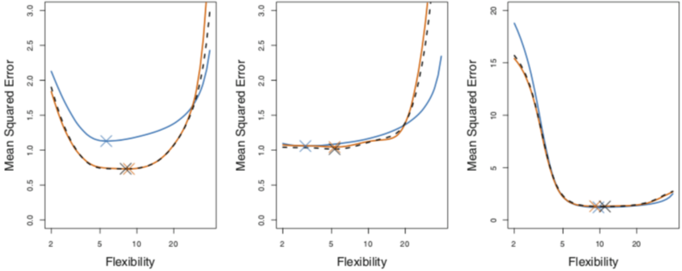
- We know the validation set approach gives us very biased estimates of the testing MSE
- LOOCV on the other hand gives nearly unbiased estimates
- k-fold CV lies somewhere in the middle
  - Bias is generally low and closer to LOOCV
- We don't only care about bias
  - What about variance?

Cross-validation

- LOOCV has higher variance than k-fold CV for $k < n$
- Bias-variance trade-off when choosing k
- LOOCV has higher variance because of correlation in the data
  - All predictions are made from a model fit on $n - 1$ data points
  - These models are extremely similar to each other because they're fit on almost the same data
  - This leads to higher, positive correlation between predictions
  - Averaging positively correlated variables leads to a higher variance
- Generally people choose $k = 5$ or $k = 10$

Cross-validation

- The book shows these CV estimates for 3 different simulated examples
  - Blue line is true testing MSE
  - Orange line is the 10-fold estimate
  - Dashed line is the LOOCV estimate



James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning. New York: springer.

Cross-validation

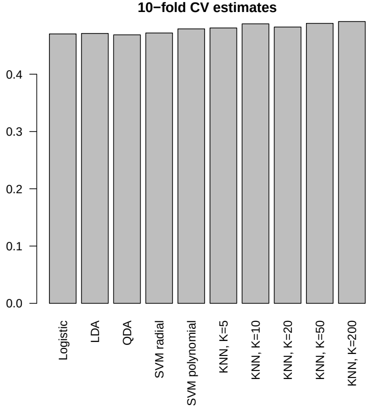
- The top left panel shows a situation where the estimates were below the truth, but still led to a good estimate of the tuning parameter
  - In this case, tuning parameter selection is the goal of CV
- In the other two data sets, the CV approaches estimated the true error quite well
- 10-fold and LOOCV led to similar results in all three cases

Cross-validation

- So far we have only discussed testing MSE for quantitative responses
- Applying these ideas to classification problems is nearly identical
- Simply replace $(Y_i - \widehat{Y}_i)^2$ with $1(Y_i \neq \widehat{Y}_i)$
- All other ideas about using k-fold cross validation or LOOCV apply directly

Cross-validation

- I applied 10-fold cross validation to the stock market data to see which approach is best
- QDA is the best according to 10-fold CV, though differences are very small



Cross-validation

- The great part about CV is that it can be used to make nearly any decision that goes into a model
  - Degree of nonlinearity
  - Which variables to include
  - Which kernel is best for an SVM
  - Number of neighbors in KNN
- Provides a principled and automated way to select tuning parameters in complex models

Bootstrap

- Now we will discuss another important resampling technique called the bootstrap
- The bootstrap is generally used to create confidence intervals or estimate standard errors of statistics
  - More generally, we want to understand uncertainty
- In some cases, standard errors can be calculated analytically
  - Linear regression, many others
- But in many complex approaches, the variance of the sampling distribution is hard to find

Bootstrap

- We generally use the bootstrap for estimating uncertainty in two scenarios
  - Derivation of standard errors is difficult or impossible
  - Don't want to assume anything about the distribution of the data
- The bootstrap is used universally due to its simplicity and broad applicability
- Nearly all approaches can be combined with the bootstrap
  - We will briefly discuss situations where it can not be used

Bootstrap

- Suppose we observe n data points from an unknown distribution F, i.e.

$$X_i \sim F$$

- Let's suppose we have a statistic $\widehat{\theta}$ that is a function of the data
- Ideally we would know $SE(\widehat{\theta})$ or we would know the sampling distribution of $\widehat{\theta}$
    - This would allow us to create confidence intervals

Bootstrap

- Remember that if we want to see the sampling distribution of a statistic we can simply follow these steps:
    1. Draw n data points from F
    2. Calculate $\widehat{\theta}$ based on these n data points
    3. Repeat this process a large number of times
- We can't do this for one main reason
    - We don't know what F is
- If we knew F we could perform these steps and would have a perfect understanding of the uncertainty in our estimator

Bootstrap

- The main idea of the bootstrap is to estimate F with the empirical distribution of the data, denoted by $\widehat{F}_n$
- $\widehat{F}_n$ is a discrete distribution that assigns probability $1/n$ to each data point observed in your sample
- Instead of drawing n samples from F, we draw n samples from $\widehat{F}_n$
    - Sample n data points from your data, with replacement

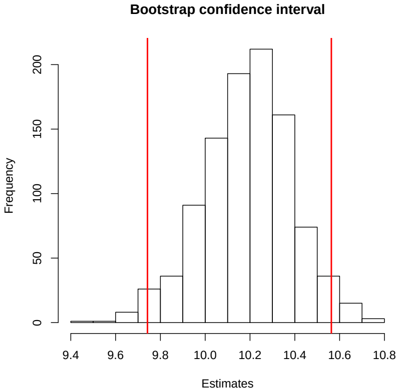
Bootstrap

- The steps of the bootstrap are as follows
  1. Sample n data points with replacement from your original data. Call these $X_i^{(b)}$
  2. Calculate $\widehat{\theta}^{(b)}$ based on $X_i^{(b)}$ for $i = 1, \dots, n$
  3. Repeat steps 1 and 2 for $b = 1, \dots, B$ where B is large

- Once we have these B estimates, there are many ways to proceed with inference

Bootstrap

- The most straightforward approach is called the percentile method
- Construct a confidence interval as $(q_{\alpha/2}, q_{1-\alpha/2})$
  - $q_{\alpha/2}$ is the $\alpha/2$ quantile of the bootstrap samples



- This works well if your bootstrap samples are centered around $\widehat{\theta}$
- If your bootstrap samples are not centered around $\widehat{\theta}$, you can do the following
  1. Find $q_{\alpha/2}$ and $q_{1-\alpha/2}$ as before
  2. Set your confidence interval as $(2\widehat{\theta} - q_{1-\alpha/2}, 2\widehat{\theta} - q_{\alpha/2})$
- This should give better confidence intervals if the bootstrap estimates are not centered at $\widehat{\theta}$

Bootstrap

- We can also calculate a standard error estimate from the bootstrap samples
- The bootstrap estimate of the standard error for $\widehat{\theta}$ is

$$\widehat{SE}(\widehat{\theta}) = \sqrt{\frac{1}{B-1}\sum_{b=1}^{B}\left(\widehat{\theta}^{(b)} - \frac{1}{B}\sum_{b=1}^{B}\widehat{\theta}^{(b)}\right)^2}$$

- We can then proceed with inference as usual

$$CI(\theta) = (\widehat{\theta} - K \times \widehat{SE}(\widehat{\theta}), \widehat{\theta} + K \times \widehat{SE}(\widehat{\theta}))$$

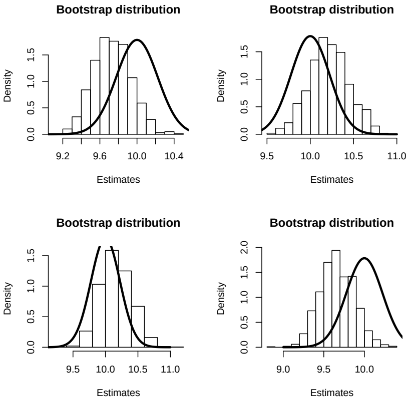
- This assumes that our test statistic is symmetric
- If our statistic follows a normal distribution, then $K = 1.96$
  - 1.96 is the 0.975 quantile of the normal distribution
- There are other approaches to inference with the bootstrap that can alleviate issues stemming from small sample sizes, bias, or skewedness
  - Studentized bootstrap interval
  - Bias-corrected and accelerated bootstrap
  - Others

Bootstrap

- Let's first apply the bootstrap in a simple example where we know the sampling distribution
- $X_i \sim \mathcal{N}(\mu, 1)$ and we want to estimate μ with $\overline{X}$
- We know that $\overline{X} \sim \mathcal{N}(\mu, 1/n)$
- let's apply the bootstrap and see how well it approximates this known sampling distribution

Bootstrap

- Here are four bootstrap distributions where $n = 20$
- True sampling distribution is denoted by the solid line



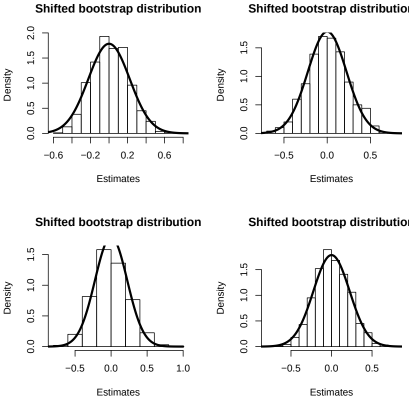
Bootstrap

- You might be thinking that the bootstrap histograms don't closely match the true sampling distribution
- This is because the mean of the bootstrap histograms is $\overline{X}$ and not μ
  - Inherent randomness in $\overline{X}$
- Importantly, however, the spread of the distributions does seem to closely match the sampling distribution spread

Bootstrap

- We are not using the bootstrap for point estimation
- We are using it for uncertainty estimation!
  - The spread is what we care about
- The bootstrap is based on the idea that the distribution of $\widehat{\theta} - \theta$ is well approximated by $\widehat{\theta}^{(b)} - \widehat{\theta}$
  - How far off is the estimate from the truth

- Let's return to the normal means example
- Now we shift the true distribution by μ and the bootstrap distributions by $\overline{X}$



Bootstrap

- The bootstrap is doing a remarkable job at estimating the uncertainty in the sampling distribution!
- Importantly, the bootstrap assumed no knowledge about the distribution of the data
- By simply resampling the data and re-estimating the test statistic, we can accurately capture the uncertainty of $\widehat{\theta}$

Bootstrap

- The previous example was a case where the bootstrap was not needed
  - We knew distribution of $\overline{X}$
- There are many situations where this won't be the case
- Suppose we want to estimate the residual variance in a linear regression model, σ^2
- What is the distribution of $\widehat{\sigma}^2$?
  - I'm not sure, so let's use the bootstrap!

Bootstrap

- Remember from lecture 2 that our estimate of the variance is given by

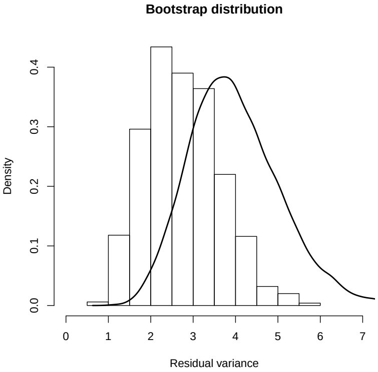
$$\frac{1}{n-p-1}\sum_{i=1}^{n}(Y_i - \widehat{Y}_i)^2$$

- We can create B bootsrapped datasets denoted by $(\boldsymbol{X}_i^{(b)}, Y_i^{(b)})$
- Each time calculate

$$\frac{1}{n-p-1}\sum_{i=1}^{n}(Y_i^{(b)} - \widehat{Y}_i^{(b)})^2$$

Bootstrap

- Let's compare the bootstrap distribution to the true sampling distribution
- Note that I can obtain the true sampling distribution empirically because I know F, the distribution that governs the data
  - Solid line is true distribution



Bootstrap

- Again we see that the bootstrap spread looks very similar to the true sampling distribution spread
- The bootstrap distribution is shifted from the truth due to randomness in any one individual data set
- If we were to shift both distributions they would appear similar as was the case for $\overline{X}$

Bootstrap

- One important question is how many bootstrap samples, B need to be taken
- If using the percentile method to constructing intervals, a larger B is recommended
  - At least 1000
- If only using the bootstrap to estimate a standard error, it might be ok to use less
  - Around 100
- Unless computation time is a big concern, simply use a large number, over 1000

Other bootstraps

- There are many modifications to the bootstrap all based on the same idea
    - Resample to approximate the true sampling distribution
- The most common such approach is the parametric bootstrap
- Assume the data come from a distribution $F(\theta)$
    - Parameters θ fully characterize the distribution
    - Imagine F is a normal distribution with parameters $\theta = (\mu, \sigma^2)$

Other bootstraps

- The parametric bootstrap first estimates $\widehat{\theta}$ from the data
- Then creates bootstrapped data sets by drawing data sets of size n from $F(\widehat{\theta})$
- All remaining steps are the same as the standard nonparametric bootstrap
- Works better than the nonparametric bootstrap in some situations
  - Relies on assuming the correct parametric form for F!
  - Nonparametric bootstrap makes no such assumptions

Correlated data

- If the individual observations are correlated, then applying the standard bootstrap won't work
- In order to approximate the true sampling distribution, the resampled data sets must have the same correlation structure
- In clustered data settings, we can bootstrap the clusters instead of the individuals
  - Maintains correlation inside of clusters

Auto data

- Let's look again at the Auto data set and fit a linear regression relating mpg to horsepower

$$E(mpg|\text{horsepower}) = \beta_0 + \beta_1 \text{horsepower}$$

- Our interest will lie in the standard errors of β_0 and β_1
- We will compare two approaches to calculating standard errors
  - Analytic expressions from lecture 2
  - Bootstrap estimates of standard error

Auto data

- If we fit the model in R, it will give us the analytic standard errors

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) 39.935861   0.717499   55.66   <2e-16 ***
horsepower  -0.157845   0.006446  -24.49   <2e-16 ***
```

- We obtain $\widehat{SE}(\widehat{\beta}_0) = 0.717$ and $\widehat{SE}(\widehat{\beta}_1) = 0.0064$
- Now let's try the bootstrap and see what we get

Auto data

- When we apply the bootstrap we get $\widehat{SE}(\widehat{\beta}_0) = 0.855$ and $\widehat{SE}(\widehat{\beta}_1) = 0.0074$
- These aren't that similar!
- This may seem like a problem with the bootstrap, but in fact it is a problem with the analytic expressions
- The theoretical standard errors rely on certain assumptions about the linear model being correctly specified
  - The bootstrap does not

Auto data

- There is nonlinearity in the model, as we saw in the cross-validation section of the notes
- Instead, let's fit a quadratic model to the data and compare the standard errors

	Analytic	Bootstrap
$\widehat{SE}(\widehat{\beta}_0)$	1.8004	2.0904
$\widehat{SE}(\widehat{\beta}_1)$	0.0311	0.0333
$\widehat{SE}(\widehat{\beta}_2)$	0.0001	0.0001

Auto data

- These are much more similar as the assumptions for the analytic standard errors are more reasonable in this setting
- Another subtle reason for the discrepancy is that the analytic standard errors assume X is fixed, while the bootstrap accounts for uncertainty in the covariates as well
  - Parametric bootstrap can be used to target the distribution of the coefficients given X
  - Parametric bootstrap also assumes the linear model is correct

	Analytic	Bootstrap	Parametric Bootstrap
$\widehat{SE}(\widehat{\beta}_0)$	1.8004	2.0904	1.8163
$\widehat{SE}(\widehat{\beta}_1)$	0.0311	0.0333	0.0314
$\widehat{SE}(\widehat{\beta}_2)$	0.0001	0.0001	0.0001

Bootstrap summary

- The bootstrap is an incredible approach that is widely applicable to many situations
- It is not an all encompassing fix to creating confidence intervals
- There are situations where the bootstrap can fail
- If an estimator is not sufficiently smooth (don't worry about what this means), the bootstrap can fail
  - LASSO estimates
  - Tree-based estimates
- In most standard settings, however, it can be applied and works remarkably well