# Class 4: Resampling and Boosting

MSA 8150: Machine Learning for Analytics

Alireza Aghasi

Institute for Insight, Georgia State University

# Introduction

# Introduction

- Recall that we fitted out models using training data and were interested in evaluating the performance with respect to independent test data

# Introduction

- Recall that we fitted out models using training data and were interested in evaluating the performance with respect to independent test data
- To produce justifiable model reliability arguments, the test data should not be used in the training

# Introduction

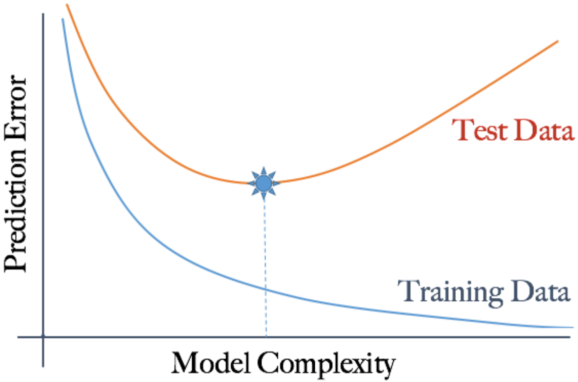
- Recall that we fitted out models using training data and were interested in evaluating the performance with respect to independent test data
- To produce justifiable model reliability arguments, the test data should not be used in the training
- If the model is evaluated against the training data the results can be very distracting

# Introduction

- Recall that we fitted out models using training data and were interested in evaluating the performance with respect to independent test data
- To produce justifiable model reliability arguments, the test data should not be used in the training
- If the model is evaluated against the training data the results can be very distracting
- The training error rate is often quite different from the test error rate, and in particular the former can dramatically underestimate the latter (recall the accuracy vs complexity chart)

# Training vs Test Model Evaluation

Recall this plot from the first session



## Real-World Data Issues and Test Performance

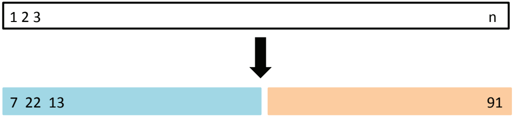
- A good evaluation is possible when a large test set is available

# Real-World Data Issues and Test Performance

- A good evaluation is possible when a large test set is available
- Often such set is not available

# Real-World Data Issues and Test Performance

- A good evaluation is possible when a large test set is available
- Often such set is not available
- We are interested in a class of methods that estimate the test error by holding out a subset of the training observations from the fitting process, and then applying the statistical learning method to those (held out) observations



# Validation Set Approach

– This is the standard approach we have been using so far

# Validation Set Approach

- This is the standard approach we have been using so far
- We randomly divide the available set of samples into two parts: a training set and a validation or hold-out set (sometimes 50%-50% splitting, often 80%-20% splitting)

# Validation Set Approach

- This is the standard approach we have been using so far
- We randomly divide the available set of samples into two parts: a training set and a validation or hold-out set (sometimes 50%-50% splitting, often 80%-20% splitting)
- The model is fit on the training set, and the fitted model is used to predict the responses for the observations in the validation set

# Validation Set Approach

- This is the standard approach we have been using so far
- We randomly divide the available set of samples into two parts: a training set and a validation or hold-out set (sometimes 50%-50% splitting, often 80%-20% splitting)
- The model is fit on the training set, and the fitted model is used to predict the responses for the observations in the validation set
- The error with reference to the hold-out set is an approximation of the test error

# example

- Recall the automobile data: Regressing Mile per Gallon in terms of the Horse Power

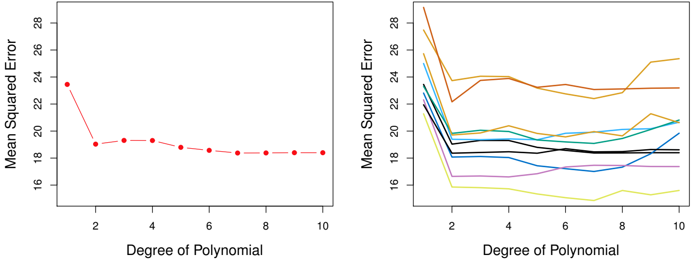
$$\texttt{mpg} = \beta_0 + \sum_{i=1}^{p} \beta_i (\texttt{horsepower})^i$$

## example

- Recall the automobile data: Regressing Mile per Gallon in terms of the Horse Power

$$\texttt{mpg} = \beta_0 + \sum_{i=1}^{p} \beta_i \, (\texttt{horsepower})^i$$

- We randomly split the 392 observations into two sets, a training set containing 196 of the data points, and a validation set containing the remaining 196 observations.



# Validation Set Cons & Pros

- The procedure is **simple to do** (as we have done so far) and only a subset of the observations those that are included in the training set rather than in the validation set are used to fit the model

# Validation Set Cons & Pros

- The procedure is **simple to do** (as we have done so far) and only a subset of the observations those that are included in the training set rather than in the validation set are used to fit the model
- The validation estimate of the test error can be highly variable, depending on precisely which observations are included in the training set and which observations are included in the validation set

# Validation Set Cons & Pros

- The procedure is **simple to do** (as we have done so far) and only a subset of the observations  those that are included in the training set rather than in the validation set  are used to fit the model
- The validation estimate of the test error can be highly variable, depending on precisely which observations are included in the training set and which observations are included in the validation set
- While the estimated test error vary a lot, **finding information such as model selection is still possible**

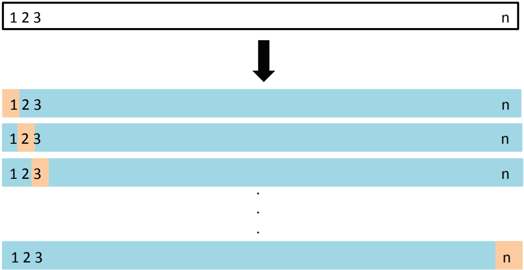
# Validation Set Cons & Pros

- The procedure is **simple to do** (as we have done so far) and only a subset of the observations those that are included in the training set rather than in the validation set are used to fit the model
- The validation estimate of the test error can be highly variable, depending on precisely which observations are included in the training set and which observations are included in the validation set
- While the estimated test error vary a lot, **finding information such as model selection is still possible**
- Since a large portion of the data need to be held aside, the model fits are not accurate enough

# Leave-One Out Cross-Validation (LOOCV)

- We have $n$ data points $(x_1, y_1), \cdots, (x_n, y_n)$, we use $n-1$ for the training and one instance for the test
- Of course a single test point is no where close to the true test error, but this process is repeated $n$ times, every time $n-1$ points used for training and one point left out for the test
- Considering $MSE_1 = (y_1 - \hat{y}_1)^2, \cdots MSE_n = (y_n - \hat{y}_n)^2$, an approximation of the test error is

$$CV_n = \frac{1}{n} \sum_{i=1}^{n} MSE_i$$



# Cons & Pros with LOOCV

- It has a very small bias compared to the validation set approach (it almost uses as much data as possible to fit the model)
- The test error overestimation is less than the validation set approach (because of what we mentioned above)
- Its results are reproducible unlike the validation set approach which uses a random subset of the data for test evaluation
- It can be computationally very expensive (requires running the algorithm $n$ times)
- For linear models there is a shortcut to calculate $CV_n$ that only requires fitting the model once with the entire data (but **this shortcut only applies to linear models**)

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

where $\hat{y}_i$ are the fitted values of the original least squares problem and $h_i$ are only data dependent

# K-Fold Cross Validation

- Widely used approach for estimating test error

# K-Fold Cross Validation

- Widely used approach for estimating test error
- This approach involves **randomly dividing the set of observations into $K$ groups**, or folds, of approximately equal size. The first fold is treated as a validation set, and the method is fit on the remaining $K-1$ folds

# K-Fold Cross Validation

- Widely used approach for estimating test error
- This approach involves **randomly dividing the set of observations into $K$ groups**, or folds, of approximately equal size. The first fold is treated as a validation set, and the method is fit on the remaining $K - 1$ folds
- This is done in turn for each part $k = 1, 2, \cdots, K$, and then the results are combined

# K-Fold Cross Validation

- Widely used approach for estimating test error
- This approach involves **randomly dividing the set of observations into $K$ groups**, or folds, of approximately equal size. The first fold is treated as a validation set, and the method is fit on the remaining $K-1$ folds
- This is done in turn for each part $k = 1, 2, \cdots, K$, and then the results are combined
- Estimates can be used to select best model, and to give an idea of the test error of the final chosen model

$$CV_K = \frac{1}{K} \sum_{k=1}^{K} MSE_k$$

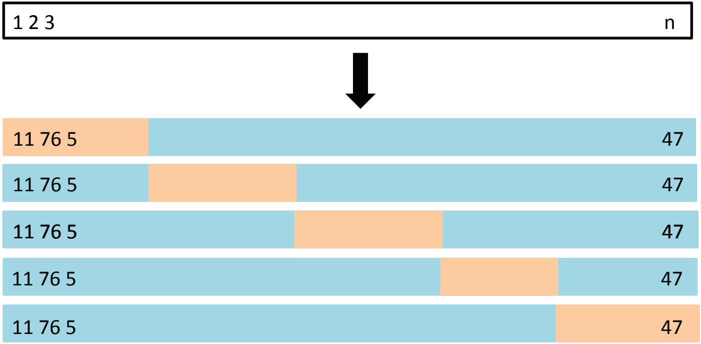
# K-Fold Cross Validation

- Widely used approach for estimating test error
- This approach involves **randomly dividing the set of observations into $K$ groups**, or folds, of approximately equal size. The first fold is treated as a validation set, and the method is fit on the remaining $K-1$ folds
- This is done in turn for each part $k = 1, 2, \cdots, K$, and then the results are combined
- Estimates can be used to select best model, and to give an idea of the test error of the final chosen model

$$CV_K = \frac{1}{K}\sum_{k=1}^{K} MSE_k$$

- Often $K = 5$ or $K = 10$ is what is considered in application

# K-Fold Cross Validation



# K-Fold Cross Validation and LOOCV

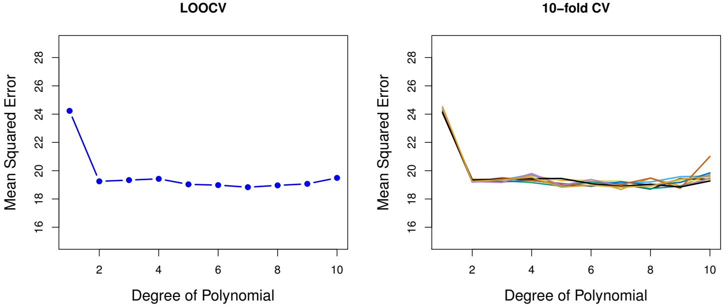
– LOOCV is a special case of $K$-fold CV for $K = n$

# K-Fold Cross Validation and LOOCV

- LOOCV is a special case of $K$-fold CV for $K = n$
- In general $K$-fold CV is much cheaper than LOOCV because it only requires $K$ model fits vs $n$ model fits

# K-Fold Cross Validation and LOOCV

- LOOCV is a special case of $K$-fold CV for $K = n$
- In general $K$-fold CV is much cheaper than LOOCV because it only requires $K$ model fits vs $n$ model fits
- For model selection, $K$-fold CV often gives us similar outcomes at a much lower computational cost



# K-Fold Cross Validation and LOOCV

– Aside form the computational issues, even surprisingly K-Fold CV produces better test estimates than the LOOCV

# K-Fold Cross Validation and LOOCV

- Aside form the computational issues, even surprisingly K-Fold CV produces better test estimates than the LOOCV
- LOOCV has a lower bias compared to the K-fold CV, since it uses more data to fit the model

# K-Fold Cross Validation and LOOCV

- Aside form the computational issues, even surprisingly K-Fold CV produces better test estimates than the LOOCV
- LOOCV has a lower bias compared to the K-fold CV, since it uses more data to fit the model
- But K-fold CV has a lower variance compared to the LOOCV, since LOOCV is the sum of $n$ highly correlated random variables while the correlation between the MSEs in K-fold CV is lower, recall

$$var(X+Y) = var(X) + var(Y) + 2cov(X,Y)$$

# CV & Classification

– We divide the data into K roughly equal-sized index sets $C_1, \cdots, C_K$

# CV & Classification

- We divide the data into K roughly equal-sized index sets $C_1, \cdots, C_K$
- Compute

$$CV_K = \frac{1}{K} \sum_{k=1}^{K} Err_k$$

where

$$Err_k = \frac{1}{\#\text{ elements in } C_k} \sum_{i \in C_k} I(y_i \neq \hat{y}_i)$$

# Bootstrap

# Bootstrap

- The bootstrap is a flexible and very powerful statistical tool that can be used to quantify the uncertainty associated with a given estimator or statistical learning method

# Bootstrap

- The bootstrap is a flexible and very powerful statistical tool that can be used to quantify the uncertainty associated with a given estimator or statistical learning method
- It can provide an estimate of the standard error of a coefficient, or a **confidence interval for that coefficient**, regardless of how complex the derivation of that coefficient is

# Bootstrap via an Example

- Lets explain bootstrap via an example

# Bootstrap via an Example

- Lets explain bootstrap via an example
- Suppose that we wish to invest a fixed sum of money in two financial assets that yield returns of $X$ and $Y$ (random quantities)

# Bootstrap via an Example

- Lets explain bootstrap via an example
- Suppose that we wish to invest a fixed sum of money in two financial assets that yield returns of $X$ and $Y$ (random quantities)
- We will invest $\alpha$ shares in $X$, and will invest the remaining $1 - \alpha$ in $Y$

# Bootstrap via an Example

- Lets explain bootstrap via an example
- Suppose that we wish to invest a fixed sum of money in two financial assets that yield returns of $X$ and $Y$ (random quantities)
- We will invest $\alpha$ shares in $X$, and will invest the remaining $1 - \alpha$ in $Y$
- To minimize the risk, we want to minimize $var(\alpha X + (1 - \alpha)Y)$

# Bootstrap via an Example

- Lets explain bootstrap via an example
- Suppose that we wish to invest a fixed sum of money in two financial assets that yield returns of $X$ and $Y$ (random quantities)
- We will invest $\alpha$ shares in $X$, and will invest the remaining $1-\alpha$ in $Y$
- To minimize the risk, we want to minimize $var(\alpha X + (1-\alpha)Y)$
- We can show that (in the class we do it) that the minimizer is

$$\alpha = \frac{var(Y) - cov(X,Y)}{var(X) + var(Y) - 2cov(X,Y)}$$

# Bootstrap via an Example

- In real-world we do not know $var(X), var(Y), cov(X, Y)$

# Bootstrap via an Example

- In real-world we do not know $var(X), var(Y), cov(X, Y)$
- Suppose we are given a data set containing pairs of $X$ and $Y$. We can estimate $var(X), var(Y), cov(X, Y)$ from the sample set and get an estimate $\hat{\alpha}$ for the optimal share

# Bootstrap via an Example

- In real-world we do not know $var(X), var(Y), cov(X, Y)$
- Suppose we are given a data set containing pairs of $X$ and $Y$. We can estimate $var(X), var(Y), cov(X, Y)$ from the sample set and get an estimate $\hat{\alpha}$ for the optimal share
- Ideally we can generate these sample sets many times, and estimate an $\hat{\alpha}$ for each and look into the histogram

# Bootstrap via an Example

- In real-world we do not know $var(X), var(Y), cov(X, Y)$
- Suppose we are given a data set containing pairs of $X$ and $Y$. We can estimate $var(X), var(Y), cov(X, Y)$ from the sample set and get an estimate $\hat{\alpha}$ for the optimal share
- Ideally we can generate these sample sets many times, and estimate an $\hat{\alpha}$ for each and look into the histogram
- However in a real-world we only have one sample set to use

# Bootstrap via an Example

- In real-world we do not know $var(X), var(Y), cov(X, Y)$
- Suppose we are given a data set containing pairs of $X$ and $Y$. We can estimate $var(X), var(Y), cov(X, Y)$ from the sample set and get an estimate $\hat{\alpha}$ for the optimal share
- Ideally we can generate these sample sets many times, and estimate an $\hat{\alpha}$ for each and look into the histogram
- However in a real-world we only have one sample set to use
- Bootstrap yet allows us to generate good estimates of $\alpha$ **using only one sample set!**

# Bootstrap via an Example

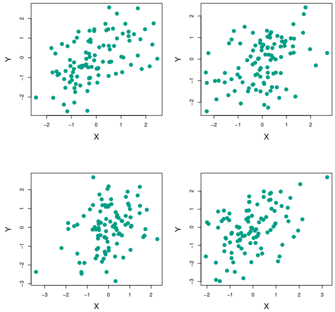
To see how nicely Bootstrap works, lets compare its outcome with the case that $\alpha$ is generated from many synthetic sample generations

- We generate 1000 sample sets each containing 100 pairs of $X, Y$

# Bootstrap via an Example

To see how nicely Bootstrap works, lets compare its outcome with the case that $\alpha$ is generated from many synthetic sample generations

- We generate 1000 sample sets each containing 100 pairs of $X, Y$
- For the synthetic data generated $var(X) = 1, var(Y) = 1.25$ and $cov(X, Y) = 0.5$ which yield an optimal value of $\alpha = 0.6$



# Bootstrap via an Example

– To get the left panel we generate 1000 synthetic sample sets, for each obtain $\hat{\alpha}$ and plot the histogram and calculate

$$\bar{\alpha} = \frac{1}{1000}\sum_{i=1}^{1000}\hat{\alpha}_i, \quad SE(\alpha) = \sqrt{\frac{1}{999}\sum_{i=1}^{1000}(\hat{\alpha}_i - \bar{\alpha})^2}$$

# Bootstrap via an Example

- To get the left panel we generate 1000 synthetic sample sets, for each obtain $\hat{\alpha}$ and plot the histogram and calculate

$$\bar{\alpha} = \frac{1}{1000}\sum_{i=1}^{1000}\hat{\alpha}_i, \quad SE(\alpha) = \sqrt{\frac{1}{999}\sum_{i=1}^{1000}(\hat{\alpha}_i - \bar{\alpha})^2}$$

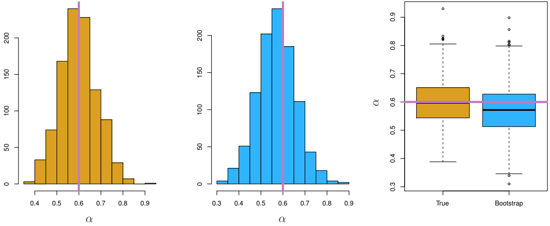
- For the bootstrap we only use one of the sample sets and regenerate new sample set by sampling with replacement

# Bootstrap via an Example

- To get the left panel we generate 1000 synthetic sample sets, for each obtain $\hat{\alpha}$ and plot the histogram and calculate

$$\bar{\alpha} = \frac{1}{1000}\sum_{i=1}^{1000}\hat{\alpha}_i, \quad SE(\alpha) = \sqrt{\frac{1}{999}\sum_{i=1}^{1000}(\hat{\alpha}_i - \bar{\alpha})^2}$$

- For the bootstrap we only use one of the sample sets and regenerate new sample set by sampling with replacement
- Surprisingly, the results are very close



## Bootstrap General Framework

- Suppose a black-box calculates $\hat{\alpha}$ from a sample set, e.g., a coefficient in linear or logistic regression

# Bootstrap General Framework

- Suppose a black-box calculates $\hat{\alpha}$ from a sample set, e.g., a coefficient in linear or logistic regression
- We are interested in estimating the variability of $\hat{\alpha}$ without examining many new sample sets

# Bootstrap General Framework

- Suppose a black-box calculates $\hat{\alpha}$ from a sample set, e.g., a coefficient in linear or logistic regression
- We are interested in estimating the variability of $\hat{\alpha}$ without examining many new sample sets
- Denoting the first bootstrap data set by $Z^1$, we use $Z^1$ to produce a new bootstrap estimate for $\alpha$, which we call $\hat{\alpha}^1$

# Bootstrap General Framework

- Suppose a black-box calculates $\hat{\alpha}$ from a sample set, e.g., a coefficient in linear or logistic regression
- We are interested in estimating the variability of $\hat{\alpha}$ without examining many new sample sets
- Denoting the first bootstrap data set by $Z^1$, we use $Z^1$ to produce a new bootstrap estimate for $\alpha$, which we call $\hat{\alpha}^1$
- This procedure is repeated $B$ (say 100 or 1000) times, in order to produce $B$ different bootstrap data sets, $Z^1, Z^2, \cdots, Z^B$, and the corresponding $\alpha$ estimates, $\hat{\alpha}^1, \cdots, \hat{\alpha}^B$

# Bootstrap General Framework

- Suppose a black-box calculates $\hat{\alpha}$ from a sample set, e.g., a coefficient in linear or logistic regression
- We are interested in estimating the variability of $\hat{\alpha}$ without examining many new sample sets
- Denoting the first bootstrap data set by $Z^1$, we use $Z^1$ to produce a new bootstrap estimate for $\alpha$, which we call $\hat{\alpha}^1$
- This procedure is repeated $B$ (say 100 or 1000) times, in order to produce $B$ different bootstrap data sets, $Z^1, Z^2, \cdots, Z^B$, and the corresponding $\alpha$ estimates, $\hat{\alpha}^1, \cdots, \hat{\alpha}^B$
- We estimate the standard error of these bootstrap estimates using the formula

$$SE_B(\alpha) = \sqrt{\frac{1}{B-1}\sum_{i=1}^{B}(\hat{\alpha}^i - \bar{\hat{\alpha}})^2} \quad \text{where} \quad \bar{\hat{\alpha}} = \frac{1}{B}\sum_{i=1}^{B}\hat{\alpha}^i$$

## Bootstrap General Framework

- Suppose a black-box calculates $\hat{\alpha}$ from a sample set, e.g., a coefficient in linear or logistic regression
- We are interested in estimating the variability of $\hat{\alpha}$ without examining many new sample sets
- Denoting the first bootstrap data set by $Z^1$, we use $Z^1$ to produce a new bootstrap estimate for $\alpha$, which we call $\hat{\alpha}^1$
- This procedure is repeated $B$ (say 100 or 1000) times, in order to produce $B$ different bootstrap data sets, $Z^1, Z^2, \cdots, Z^B$, and the corresponding $\alpha$ estimates, $\hat{\alpha}^1, \cdots, \hat{\alpha}^B$
- We estimate the standard error of these bootstrap estimates using the formula

$$SE_B(\alpha) = \sqrt{\frac{1}{B-1}\sum_{i=1}^{B}(\hat{\alpha}^i - \bar{\hat{\alpha}})^2} \quad \text{where} \quad \bar{\hat{\alpha}} = \frac{1}{B}\sum_{i=1}^{B}\hat{\alpha}^i$$

- This serves as an estimate of the standard error of $\alpha$ estimated from the original data set!

# Programming Exercise

In the remainder of the session we go through some programming exercise over the following topics

- Linear models, data splitting, confidence intervals, etc
- Some classification examples
- Some cross validation and bootstrap exercise

**Questions?**

# References


- https://www.alsharif.info/iom530, 2013.
- J. Friedman, T. Hastie, and R. Tibshirani.
  **The elements of statistical learning.**
  Springer series in statistics, 2nd edition, 2009.
- G. James, D. Witten, T. Hastie, and R. Tibshirani.
  **https://lagunita.stanford.edu/c4x/HumanitiesScience/StatLearning/asset/cv_boot.pdf, 2013.**
- G. James, D. Witten, T. Hastie, and R. Tibshirani.
  **An introduction to statistical learning: with applications in R, volume 112.**
  Springer, 2013.