Cross-Validation

Dr Austin R Brown

Kennesaw State University

Introduction

- Some of today's materials were adapted from materials created by Dr Taasoobshirazi and my former professor, Dr Khalil Shafie (thanks Drs S & T!)

Introduction

- Hopefully throughout this semester, it has become clear that regression analysis has two broad and interrelated end goals: estimating the relationship between some predictors and a response and prediction of the response.

- With respect to estimation, we care about seeing how variables may be related to one another (through our β coefficients) to help us explain why some phenomenon varies.
  - This is the primary focus of the academic research application of regression methods.

- With respect to prediction, we're less interested in the interpretation of β coefficients and more interested in building a model highly capable of accurately predicting our response given some values of our predictors.

Introduction

- If our goal is primarily prediction of future observations, let's think about the steps we've taken thus far in creating linear regression models:

- We have some sample data which includes an outcome variable and some set of predictors, we obtain our least squares estimates, and then we evaluate model fit through residual analysis, assumption checking, and then statistical testing.

- Suppose we end up with a "good" model according everything we've done so far this semester. Do we have any indication that this model will be effective in predicting the outcome if we collect new data?
    - Of course not!!

Introduction

- Ideally, we could collect some new data and use it in our model to predict the outcome and then determine how accurate our predictions are!

- However, many times this isn't feasible for a variety of reasons:
  - Mostly time and money!
  - So what do we do then??

- This is where the concept of *cross-validation* comes into play!

Cross-Validation

- In general, cross-validation refers to any technique where we split our sample data into smaller subsets, estimate model parameters with one subset (i.e., *train* the model) and evaluate the accuracy of predictive capability using another, left-out subset (*testing* the model).

- We generally have three different approaches:
  1. Validation Set Approach
  2. Leave-One-Out (LOOCV)
  3. k-Fold

- Let's consider how we apply each!!

Cross-Validation: Validation Set

- With the validation set approach, we randomly divide the sample data into two parts:
  - A training set (for estimating model parameters)
  - A testing (or validation or hold-out) set (for assessing accuracy of prediction)

- The validation set error rate, typically assessed using MSE or Mean Absolute Error (MAE), is how we determine how “good” our predictions of new data are.

Cross-Validation: Validation Set

- Now the validation set approach is nice because it is conceptually simple and straightforward to implement. But it has two main drawbacks:

- Depending on which observations are included in the training set and which are including in the testing set, the MSE/MAE we obtain can vary wildly!
  - Remember our conversation on influential observations??

- Since statistical methods tend to perform worse when trained on fewer observations (as we have in the training set compared to the full set), this suggests that MSE/MAE may overestimate the overall error rate for the whole model.

- LOOCV and k-Fold aim to solve these problems!

Cross-Validation: LOOCV

- LOOCV is like the validation set method where we split the full dataset into two parts.
- But instead of creating two datasets of comparable size, we use a single observation for the validation set and all other $n-1$ observations for the training set.
    - We calculate MSE for this observation: $(y_1 - \hat{y}_1)^2$.
- This process is repeated for all n observations
    - So every observation gets to be part of the training dataset and every observation gets to be the validation set.
    - We average the n MSE values to arrive at our LOOCV MSE estimate

Cross-Validation: LOOCV

- LOOCV has some substantial advantages over the validation set approach.

- First, it has much less bias of the model MSE as we are using almost as many observations in the training set as exist in the whole dataset.
  - So it doesn’t overestimate MSE to the same extent that the validation set approach does.

- Second, since the validation set approach can give wildly different MSE estimates depending on which observations are split into testing and training, LOOCV will yield nearly identical results no matter how many times you run it.

Cross-Validation: k-Fold

- The primary limitation of LOOCV is that it can be computationally expensive to implement as we are fitting our regression model n times.

- However, since we are estimating MSE n times with almost identical data each time, it isn't hard to see that those n estimates are highly correlated.
  - If we have a high degree of correlation, then just like with multicollinearity, variance increases.

- So while the LOOCV MSE estimate is less biased than the validation set approach, it is more variable (i.e., less precise)
  - Bias-Variance tradeoff yet again!

Cross-Validation: k-Fold

- Well, instead of running the model n times, what if we take the advantages of both the validation set approach and LOOCV and implement them into a single validation technique?
  - This is the aim of k-Fold!
- With k-Fold, we randomly partition observations into k subsets referred to as *folds*, usually choosing $k = 5$ or $k = 10$.
- This first fold is used as the testing set and the remaining $k - 1$ folds are used as the training set.
  - We go through and estimate MSE/MAE as before.

Cross-Validation: k-Fold

- But rather than stopping there, now we allow the second fold to be used as the testing set and all the others to be used as the training set.

- This process continues for all k folds until we have k estimates of MSE/MAE.

- The mean of these estimates serves as our MSE estimate.

Cross-Validation: k-Fold

- k-Fold has to contend with the bias-variance tradeoff as well, but it balances the problems the former two CV methods posses when we use $k = 5$ or $k = 10$.
- Thus, in general, we mostly only work with k-fold CV in real applications.
- Note, every technique we have discussed this semester, and even those we haven't (e.g., regresison trees, classification models, etc.), can use cross-validation.
  - It is an important and valuable skill for the modern data scientist.