Middle Tennessee State University
Department of Mathematical Sciences

DATA 3550–
Applied Predictive Modeling

Instructor: Dr. Ramchandra Rimal

Chapter 3: Overfitting and Model Tuning

Chapter 2: Overfitting and Model Tuning

Objectives:

1. **Understand what overfitting and underfitting means.**
2. **Understand the parameter tuning strategy and importance**
3. **Understand the resampling techniques and their importance**
4. **Learn the principles and steps for validating a predictive model**

Over-Fitting and Model Tuning:

- Many modeling techniques can **learn the structure of a set of data so well** that when the model is applied to the data on which the model was built, it **correctly predicts every sample**

- In addition to learning the **general patterns** in the data, the model has also learned the **characteristics of each sample's unique noise**

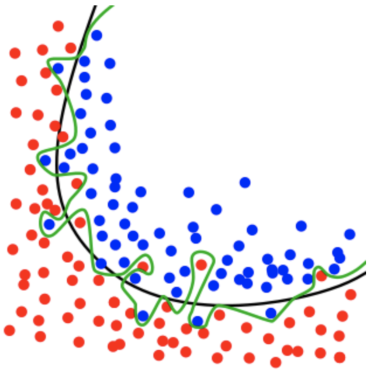
- This type of model is said to be over-fit and will usually have **poor accuracy** when predicting a new sample

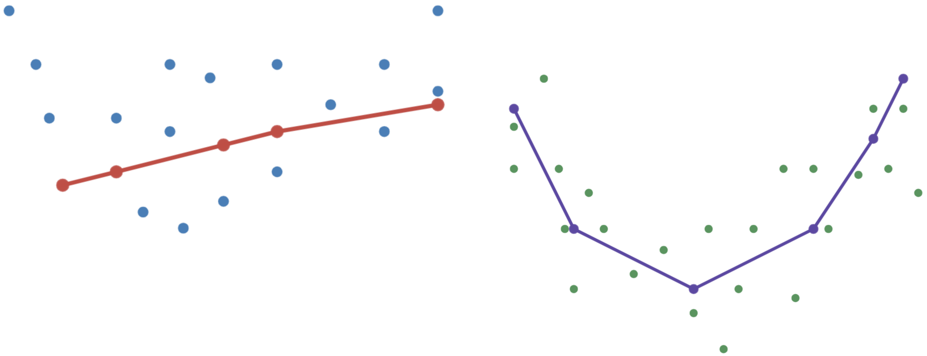
- Thus, **overfitting is the situation where a model fits very well to the current data but fails when predicting new samples**

Table 1: **Underfitting vs Overfitting**

Data Split	Underfitting	Overfitting
Training data	**High Errors** on training set (Model is too simple)	**Low errors** on training set (Model is too complex)
Test Data	**High Errors** on test data	**High errors** on test data

Model Overfitting, Underfitting and Good Fitting





https://en.wikipedia.org/wiki/Overfitting

- Over-fitting is a **concern for any predictive model** regardless of field of research

- We will describe **strategies** that enable us to have confidence that the model we build will predict new samples with a similar degree of accuracy on the set of data for which the model was evaluated

Assumption : data quality is sufficient and that it is representative of the entire sample population

- Traditionally, the best model building is achieved by **splitting the existing data into training and test sets**. The training set is used to **build and tune** the model and the test set is used to **estimate the model's predictive performance**.

-Modern approaches to model building **split the data into multiple training and testing sets**, which have been shown to often find more optimal tuning parameters and **give a more accurate representation** of the model's predictive performance

- A **tuning parameter** is a model parameter for which there is no analytical formula available to calculate an appropriate value
- Several predictive models we are going to discuss have at least one tuning parameter

- Since many of these parameters **control the complexity** of the model, **poor choices** for the values can result in over-fitting

- A **general approach to search for the best parameters** that can be applied to almost any model is given by the following chart:

Parameter Tuning Process

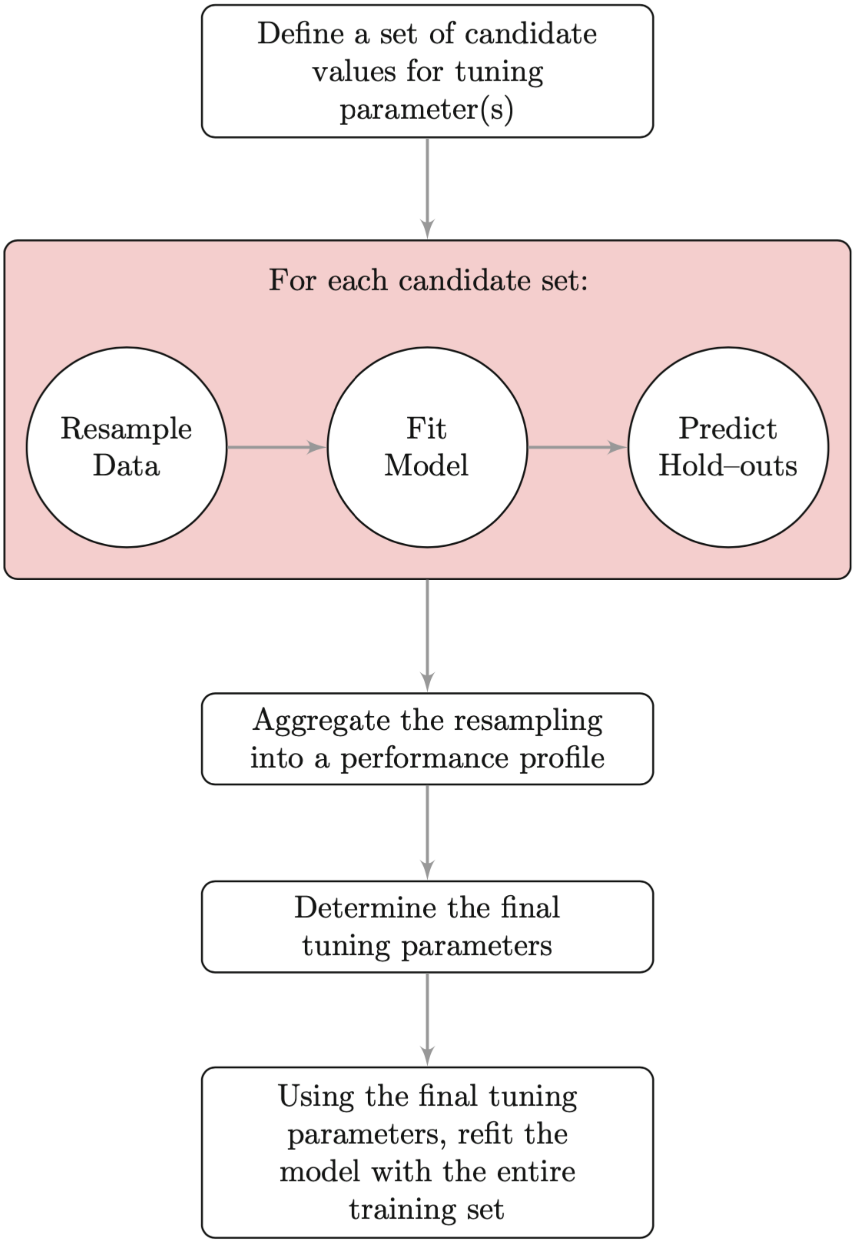

Define a set of candidate values for tuning parameter(s)
For each candidate set:
Resample Data
Fit Model
Predict Hold–outs
Aggregate the resampling into a performance profile
Determine the final tuning parameters
Using the final tuning parameters, refit the model with the entire training set

Figure 1: Source: Applied Predictive Modeling by Kuhn and Johnson

Data Splitting: Which samples will be used to evaluate performance ?

- The **training data set** is the general term for the samples used to **create the model**, while the **test or validation data set** is used to **measure performance**

- **Ideally, the model should be evaluated on samples that were not used to build or fine-tune the model, so that they provide an unbiased sense of model effectiveness**

- When a large amount of data is at hand, **a set of samples can be set aside** to evaluate the final model

- When the **sample size is not large**, a strong case can be made that a **test set should be avoided** because **every sample may be needed for model building**. In addition, the **size of the test set may not have sufficient power or precision** to make reasonable judgements

- **Resampling methods**, such as cross-validation, can be used to produce appropriate estimates of model performance using the training set

The following **methods for splitting the samples** can be used if a test set is deemed necessary:

Nonrandom Approaches: When is the nonrandom approach to splitting the data is appropriate?

Examples:

(1) If a model was being used to **predict patient outcomes**, the model may be **created using certain patient sets (e.g., from the same clinical site or disease stage)**, and then **tested on a different sample population** to understand how well the model generalizes

(2) If a model was being used for **spam filtering**; it is more **important for the model to catch the new spamming techniques** rather than prior spamming schemes

Random Approaches: When is the random approach to splitting the data is appropriate?

In most cases, there is the **desire to make the training and test sets as homogeneous as possible**, so we use random sampling.

Simple random sample: randomly choose certain percentage for training and remaining for test

Disadvantage:

- **Does not control for any of the data attributes**, such as the percentage of data in the classes

- When one class has a **disproportionately small frequency** compared to the others, there is a chance that the **distribution of the outcomes may be substantially different** between the training and test sets

Stratified random sampling: apply random sampling within subgroups (such as the classes)

Advantage:
-There is a higher likelihood that the **outcome distributions will match**
When the **outcome is a number**; the **numeric values are broken into similar groups** (e.g., low, medium, and high) and the **randomization is executed within these groups**

We can **split the data on the basis of the predictor values**

Maximum dissimilarity sampling: Introduced by Willett (1999) and Clark (1997)
- Maximum dissimilarity sampling is a method used in sampling from a population to select a representative subset of elements from it. The goal is to maximize the dissimilarity between the selected elements, with the idea that this will result in a **more diverse and representative sample.** This can be particularly useful in cases where the **population is highly heterogeneous**, as it helps ensure that all subgroups are represented in the sample. The method works by iteratively selecting elements in a way that minimizes the similarity to previously selected elements.

- There are **various ways** to measure dissimilarity between two samples
-The **simplest one is to use the distance** between the predictor values for two samples

- **Smaller distances implies proximity**; **larger distances implies the dissimilarity**

How to use dissimilarity as a tool for data splitting?

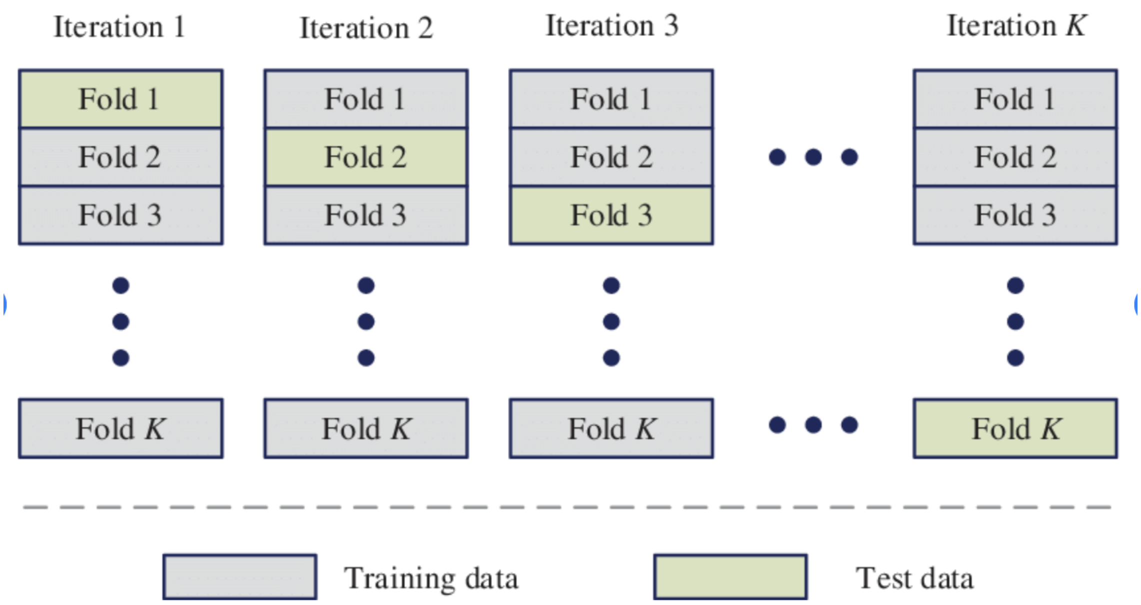
- Suppose the test set is **initialized with a single sample**. The **dissimilarity between this initial sample and the unallocated samples can be calculated**
- The unallocated sample that is **most dissimilar** would then be added to the test set
- To allocate more samples to the test set, a method is needed to determine the **dissimilarities between groups of points** (i.e., the two in the test set and the unallocated points)
- One approach is to use the **average or minimum of the dissimilarities**. For example, to measure the dissimilarities between the two samples in the test set and a single unallocated point, we can determine the two dissimilarities and average them. The third point added to the test set would be chosen as having the maximum average dissimilarity to the existing set
- This process would **continue until the targeted test set size is achieved**

Resampling Techniques for Estimating Model Performance

k-Fold Cross-Validation Illustration:

How the k-fold cross-validation is performed?

- Randomly partition the data into k sets of roughly equal size
- **Fit the model using the all samples except the first sub-**



K-fold cross-validation method.

Figure 2: Source: Mingchao Li Researchgate

set (called the first fold). The **held-out samples (the first fold) are predicted** by this model and used to estimate performance measures

- The first subset is returned to the training set and **procedure repeats with the second subset held out**, and so on

- The k **resampled estimates of performance** are summarized (usually with the mean and standard error) and used **to understand the relationship between the tuning parameter(s) and model utility**

- The choice of k is usually 5 or 10, but there is **no formal rule**

5-Fold Cross-Validation Illustration:

A slight variant of this method is to select the k partitions using stratified random sampling that makes the folds balanced with respect to

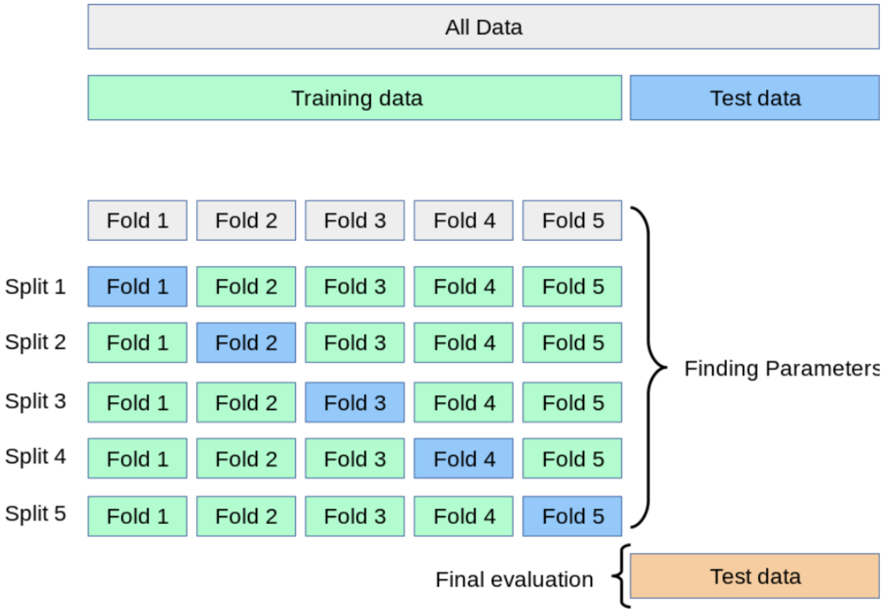


Figure 3: Source: scikit-learn

the outcome

Leave-one-out cross-validation (LOOCV): Special case of k-fold cross-validation where k is the number of samples.

- Larger values of k results the modeling more computationally burdensome
- LOOCV is most **computationally expensive** because it requires as many model fits as data points and each model fit uses a subset that is nearly the same size of the training set
- Molinaro (2005) found that leave-one-out and k =10-fold cross-validation yielded similar results, indicating that $k = 10$ **is more attractive** from the perspective of computational efficiency

Generalized Cross-Validation:

Repeated Training/Test Splits: a.k.a. "leave-group-out cross- validation" or "Monte Carlo cross-validation

The Bootstrap:

- A **bootstrap sample** is a **random sample of the data taken with replacement**
- In Bootstrap sampling, after a data point is selected for the subset, it is still available for further selection

-The **bootstrap sample is of the same size as the original data set**. Hence, **some samples will be represented multiple times** in the bootstrap sample while others will not be selected at all

- The **samples not selected** are usually referred to as the **out-of-bag" samples**
- For a given iteration of bootstrap resampling, a **model is built on the selected samples** and is used to **predict the out-of-bag samples**

Choosing Final Tuning Parameters

- Once model performance has been quantified across sets of tuning parameters, the simplest approach is to pick the settings associated with the numerically best performance estimates

- In general, it may be a good idea to **favor simpler models** over more **complex models**
- Choosing the tuning parameters based on the numerically optimal value may lead to models that are overly complicated

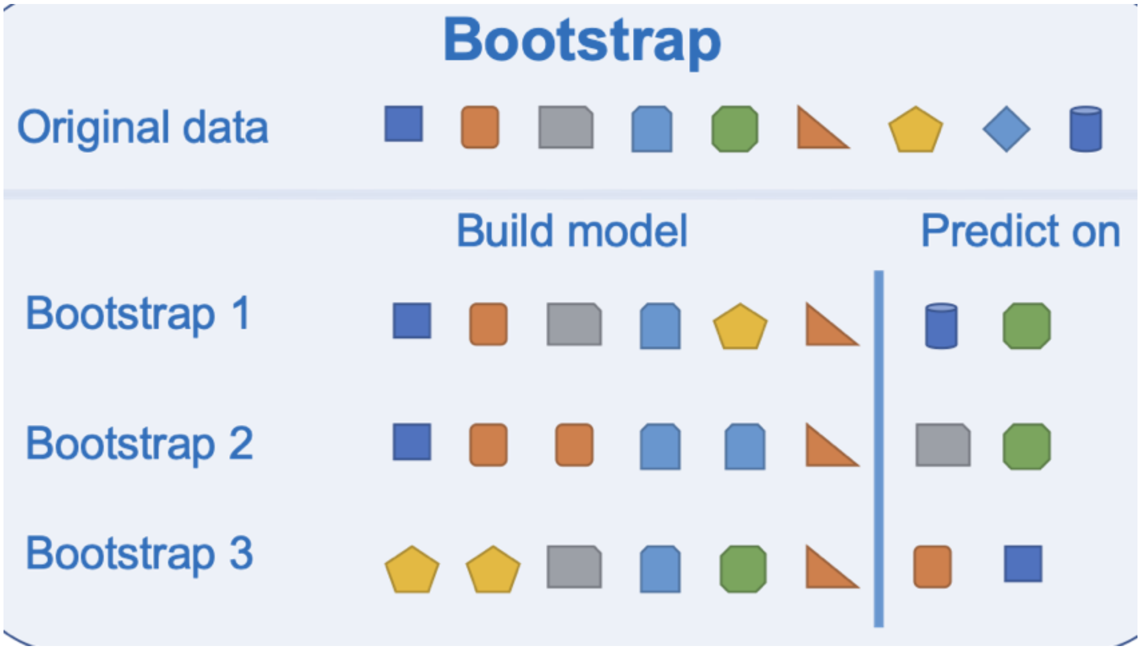


Figure 4: Source: Article Machine learning orientation by Uwe Sterr

How Bootstrapping Works?

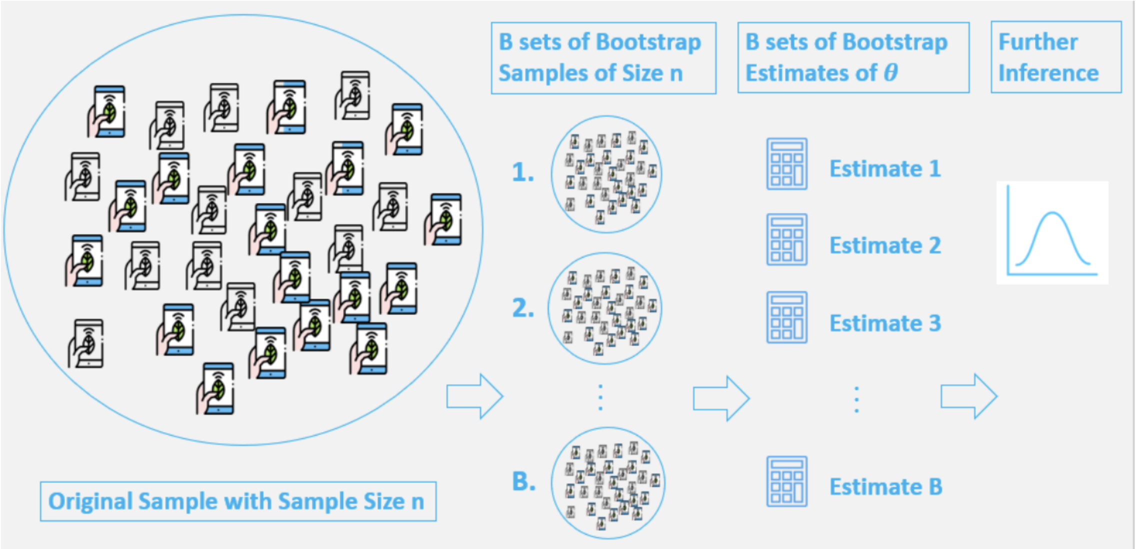

B sets of Bootstrap Samples of Size n
B sets of Bootstrap Estimates of θ
Further Inference
1.
2.
B.
Estimate 1
Estimate 2
Estimate 3
Estimate B
Original Sample with Sample Size n

Figure 5: Source: Article by Lorna Yen

Two Schemes for choosing less complex models:

One-standard error method for choosing simpler models:

- Find the **numerically optimal value** and its **corresponding standard error**
- Look for the **simplest model** whose performance is **within a single standard error** of the numerically best value

Choose a simpler model that is within a certain tolerance of the numerically best value:

For example, in the figure below, the best accuracy value across the profile was 75 % corresponding to the cost of 16. If a 4 % loss in accuracy was acceptable as a trade-off for a simpler model, accuracy values greater than 72% would be acceptable. For the profile in the figure, a cost value of 1 would be chosen using this approach. However, the **apparent accuracy rate (accuracy obtained by predicting the training samples)** shows the model improvs as the cost is increasing.

Recommendations for Data Splitting

- A **test set** is a single evaluation of the model and has **limited ability** to characterize the uncertainty in the results
- Proportionally **large test sets** divide the data in a way that **increases bias** in the performance estimates

With small sample sizes:

-The model may **need every possible data point** to adequately determine model values

Choosing simple vs complex models?

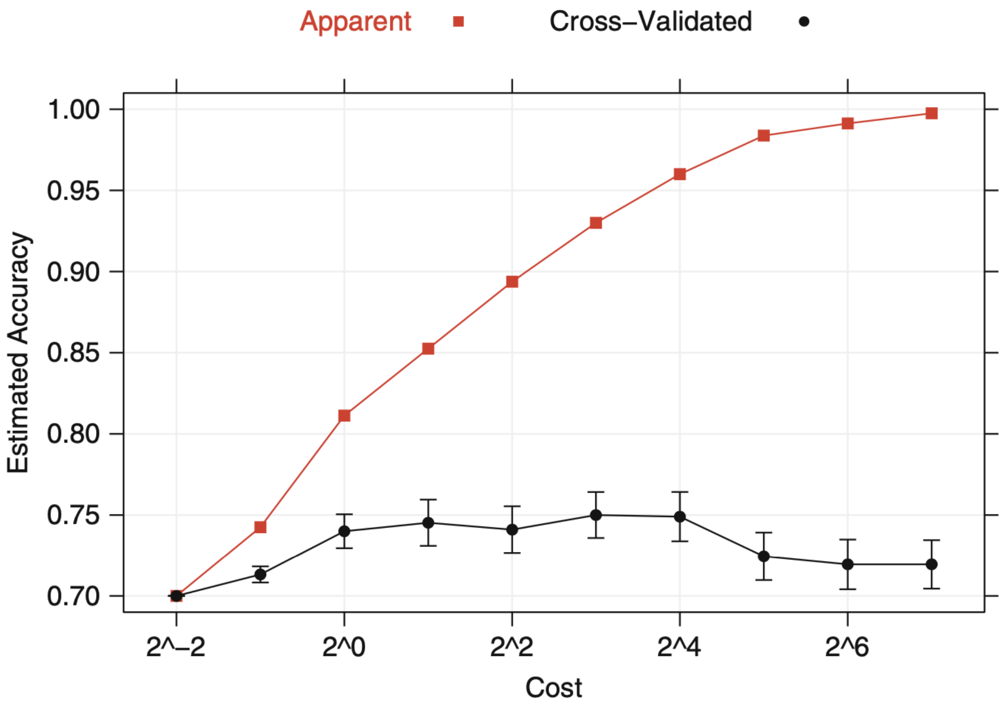


Figure 6: Source: Applied Predictive Modeling by Kuhn and Johnson

-The **uncertainty of the test set** can be considerably large to the point where **different test sets may produce very different results**

- **Resampling methods** can produce reasonable predictions of how well the model will perform on future samples

- **No resampling method is uniformly better than another; the choice should be made while considering several factors**

- If the **samples size is small**, **repeated 10-fold cross-validation is recommended** for several reasons: the bias and variance properties are good and, given the sample size, the computational costs are not large

- If the goal is to **choose between models**, as opposed to getting the best indicator of performance, a strong case can be made for using one of the **bootstrap procedures** since these have very **low variance**

- For **large sample sizes**, the differences between resampling methods become less pronounced, and **computational efficiency increases in importance**. Here, **simple 10-fold cross-validation** should provide acceptable variance, low bias, and is relatively quick to compute

Strategies For Making Selection Between Models

How do we choose between multiple models?

Once the settings for the tuning parameters have been determined for each model, the following scheme can be applied:
- Start with **several models** that are the **least interpretable** and **most flexible**
- Investigate **simpler models** that are more understandable
- Consider using the **simplest model** that **reasonably approximates the performance of the more complex methods**

In many cases, a range of models shows equivalent performance so we can weight the benefits of different methodologies (e.g., computational complexity, easy of prediction, interpretability)

For example:

- A **nonlinear support vector machine or random forest** model might have superior accuracy, but the complexity and scope of the prediction equation may prohibit exporting the prediction equation to a production system.

- **Logistic regression** is a more simplistic technique than the nonlinear support vector machine model for estimating a classification boundary. It has no tuning parameters and its prediction equation is simple and easy to implement using most software

-In this case **if the results of both models are similar then we choose the simpler model**.

Hothorn et al. (2005) and Eugster et al. (2008) describe statistical methods for comparing methodologies based on resampling results

- Since the accuracies were measured using **identically resampled data sets**, statistical methods for **paired comparisons can be used to determine if the differences between models are statistically significant**

-A **paired t-test** can be used to evaluate the hypothesis that the models have equivalent accuracies (on average) or, analogously, that the mean difference in accuracy for the resampled data sets is zero

References

[1] Kuhn, Max., and Kjell Johnson. Applied Predictive Modeling. New York: Springer, 2013

[2] https://towardsdatascience.com/an-introduction-to-the-bootstrap-method-58bcb51b4d60

[3] http://machinelearningintro.uwesterr.de

[4] Hastie, Trevor, Tibshirani, Robert and Friedman, Jerome. The Elements of Statistical Learning. New York, NY, USA: Springer New York Inc., 2001.