

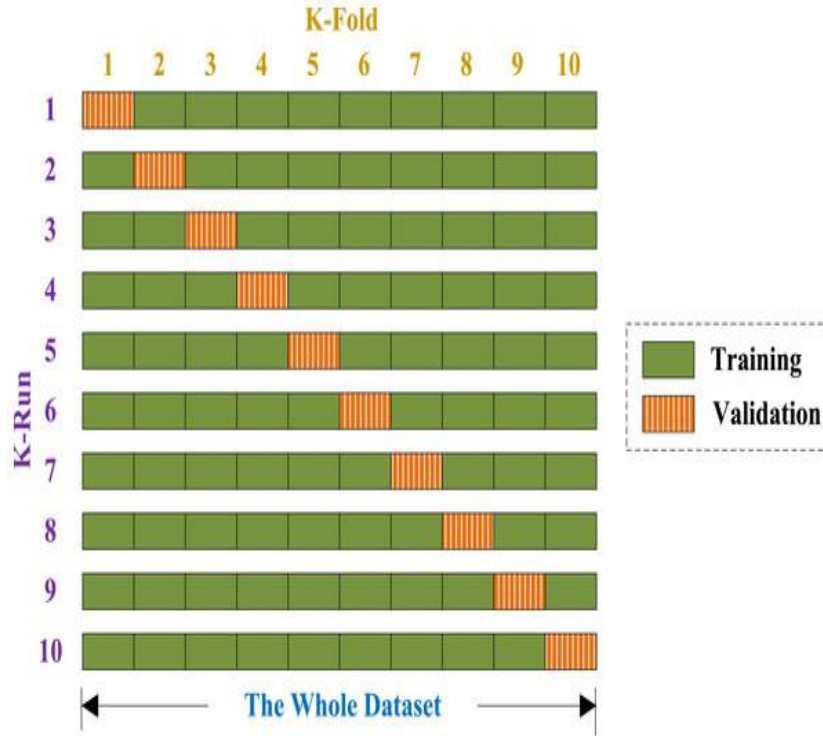
Multiple Linear Regression

Cross Validation - II

Contents

1. K-Fold Cross Validation
2. Repeated K-Fold Cross Validation
3. Leave One Out Cross Validation
4. Resampling (Bootstrap Method)

K-Fold Cross Validation



- In k-fold cross-validation the data is first partitioned into k equally (or nearly equally) sized segments or folds
- Then k iterations of training and testing are performed such that each time one fold is kept aside for testing and model is developed using k-1 folds

- Model performance measure is aggregate measure based on above iterations

K-Fold Cross Validation in R

#Creating 'k' Folds

```
kfolds<-trainControl(method="cv",number=4)
```

- *trainControl() controls the computational nuances of the train function.*
- *method="cv" tells R to use Cross Validation method.*
- *number= specifies the number of folds.*

#Testing

```
model<-  
train(claimamt~vehage+CC+Length,data=motor,method="lm",trControl=k  
folds)  
model
```

- *train() fits predictive models over different tuning parameters.*
- *It performs a number of classification and regression routines, fits each model and calculates a resampling based performance measure.*
- *method=lm fits a linear regression*
- *trControl= specifies the train function.*

K-Fold Cross Validation in R

Output

Linear Regression

1000 samples
3 predictor

No pre-processing

Resampling: Cross-Validated (4 fold)

Summary of sample sizes: 751, 749, 750, 750

Resampling results:

RMSE	Rsquared	MAE
11445.19	0.7319599	9004.326

Interpretation :

- ▣ R^2 of the original model is 73.19%
- ▣ RMSE for the original is model is 11444.51
- ▣ Comparing the RMSE values, we can say that the model is stable

Repeated K-Fold Cross Validation

- As the name suggests, repeated k-fold cross validation technique **undertakes cross validation and repeats the process m-number of times**
- This ensures that more robust measure of model performance is generated
- **K-fold** is repeated m times with different randomization in each repetition

For instance,

- Five repeats of 10-fold cross validation will generate 50 total resamples.
- These results are again averaged to produce a single estimate
- This is not the same as 50-fold cross validation

Repeated K-Fold Cross Validation in R

#Creating 'k' Folds and 'm' repeats

```
kfolds<-trainControl(method="repeatedcv",number=4,repeats=5)
```

- *trainControl()* control the computational nuances of the train function.
- *method="repeatedcv"* tells R to use Repeated Cross Validation method.
- *number=* specifies the number of folds.
- *repeats=* specifies the number of repeats.

#Testing

```
model<-  
train(claimamt~vehage+CC+Length,data=motor,method="lm",trControl=k  
folds)
```

```
model
```

Repeated K-Fold Cross Validation in R

Output

```
Linear Regression

1000 samples
  3 predictor

No pre-processing
Resampling: Cross-Validated (4 fold, repeated 5 times)
Summary of sample sizes: 749, 749, 751, 751, 749, 751, ...
Resampling results:

    RMSE      Rsquared    MAE
11527.31  0.7305732  9043.351
```

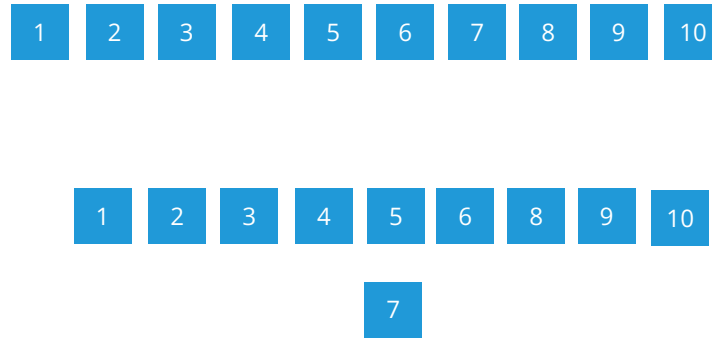
Interpretation :

- ▣ R^2 of the original model is 73.19%
- ▣ RMSE for the original is 11444.51
- ▣ RMSE values of the cross validated model indicates stability.

Leave One Out Cross Validation (LOOCV)

- **LOOCV** is a special case of k-fold cross validation where **k equals the sample size (n)**
- Each time one observation is kept aside and the model is developed on the remaining data.
- The left out observation is predicted using the model.
- This process is repeated n times
- RMSE is computed based on these predicted residuals

- Sample size is 10 and one observations (say 7) is chosen to be kept aside
- The model is developed on the new sample with $n=9$ and observation 7 is predicted



LOOCV in R

#Creating 'k' Folds

```
kfolds<-trainControl(method="LOOCV")
```

- *trainControl()* control the computational nuances of the train function.
- *method="LOOCV"* tells R to use Leave One Out Cross Validation process.
- *No other arguments need to be included in the command.*

#Testing

```
model<-  
train(claimamt~vehage+CC+Length,data=motor,method="lm",trControl=k  
folds)  
model
```

LOOCV in R

Output

```
Linear Regression
```

```
1000 samples  
  3 predictor
```

```
No pre-processing
```

```
Resampling: Leave-One-Out Cross-Validation
```

```
Summary of sample sizes: 999, 999, 999, 999, 999, 999, ...
```

```
Resampling results:
```

RMSE	Rsquared	MAE
11515.85	0.7294088	9039.582

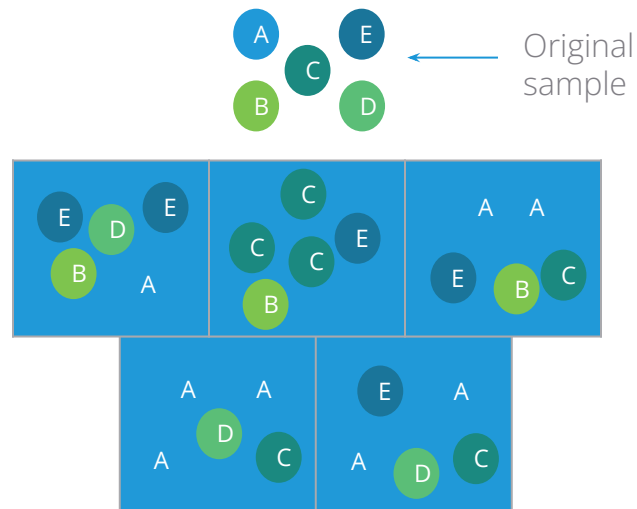
Interpretation :

- ▣ R^2 of the original model is 73.19%
- ▣ Not much difference in RMSE values compared to that of the original model
- ▣ Thus model can be considered stable

Resampling Validation (Bootstrapping)

- Bootstrapping is a technique by which a **random sample with replacement** is **repeatedly drawn from the original sample** and the size of the random sample is **the same as original** (Therefore having some observations appear more than once)
- Bootstrapping essentially allows us to simulate repeated statistical experiments
- Generally, 100-500 bootstrap samples are considered adequate for evaluating precision of a model

- Suppose our sample size is 5
- Number of bootstrap samples is 5
- 5 samples of size 5 are drawn
- Model is fit on each of the samples and average of all results is considered to measure model performance



Bootstrapping in R

#Creating 'k' Folds

```
kfolds<-trainControl(method="boot")
```

- *trainControl()* control the computational nuances of the train function.
- *method="boot"* tells R to use bootstrapping approach to resampling.

#Testing

```
model<-  
train(claimamt~vehage+CC+Length,data=motor,method="lm",trControl=k  
folds)  
model
```

Bootstrapping in R

Output

```
Linear Regression
```

```
1000 samples
```

```
  3 predictor
```

```
No pre-processing
```

```
Resampling: Bootstrapped (25 reps)
```

```
Summary of sample sizes: 1000, 1000, 1000, 1000, 1000, 1000, ...
```

```
Resampling results:
```

RMSE	Rsquared	MAE
11400	0.7354236	8967.069

Interpretation :

- ▣ R^2 of the original model is 73.19%
- ▣ RMSE for the original is 11444.51
- ▣ Comparing the RMSE values of the cross validated model, we can say model is stable.

Quick recap

This session illustrated five most important **model validation techniques**:

K-Fold Cross Validation

- Data is first partitioned into k equally (or nearly equally) sized segments or folds
- Then k iterations of training and testing are performed such that each time one fold is kept aside for testing and model is developed using $k-1$ folds

Repeated K-Fold Cross Validation

- This is an extension of k -fold method wherein the process is repeated m number of times

Quick Recap

Leave One Out Cross Validation (LOOCV)

- Special case of k-fold cross validation where **k** equals the sample size (**n**), observation number **i** is kept aside and the model is developed on remaining data after which error is calculated
- This process is repeated **n** times, for all **i**'s
- RMSE is computed based on these predicted residuals

Resampling Cross Validation (Bootstrapping)

- Random sample with replacement is repeatedly drawn from the original sample and the size of the random sample is same as original
- Model is fit on each sample and average results are considered for measuring model validity

Validation in R

- Package **caret** has all functions needed to carry out different types of validation
- **createDataPartition()**, **trainControl()** and **train()** are the most important functions
- Depending on the validation technique, **method=** needs to be specified in **trainControl()**