Quiz 2

Load the Alzheimer's disease data using the commands:



1

2

library(AppliedPredictiveModeling)

data(AlzheimerDisease)

Which of the following commands will create non-overlapping training and test sets with about 50% of the observations assigned to each?





1

2

3

4

adData = data.frame(diagnosis,predictors)

train = createDataPartition(diagnosis, p = 0.50,list=FALSE)

test = createDataPartition(diagnosis, p = 0.50,list=FALSE)





1

2

3

4

adData = data.frame(predictors)

trainIndex = createDataPartition(diagnosis,p=0.5,list=FALSE)

training = adData[trainIndex,]

testing = adData[-trainIndex,]





1

2

3

4

adData = data.frame(diagnosis,predictors)

trainIndex = createDataPartition(diagnosis, p = 0.50)

training = adData[trainIndex,]

testing = adData[-trainIndex,]





1

2

3

4

adData = data.frame(diagnosis,predictors)

trainIndex = createDataPartition(diagnosis, p = 0.50,list=FALSE)

training = adData[trainIndex,]

testing = adData[-trainIndex,]

**Correct**

Incorrect

0 / 1 points

2.

Load the cement data using the commands:



1

2

3

4

5

6

7

library(AppliedPredictiveModeling)

data(concrete)

library(caret)

set.seed(1000)

inTrain = createDataPartition(mixtures$CompressiveStrength, p = 3/4)[[1]]

training = mixtures[ inTrain,]

testing = mixtures[-inTrain,]

Make a plot of the outcome (CompressiveStrength) versus the index of the samples. Color by each of the variables in the data set (you may find the cut2() function in the Hmisc package useful for turning continuous covariates into factors). What do you notice in these plots?



There is a non-random pattern in the plot of the outcome versus index that does not appear to be perfectly explained by any predictor suggesting a variable may be missing.



There is a non-random pattern in the plot of the outcome versus index.



There is a non-random pattern in the plot of the outcome versus index that is perfectly explained by the Age variable.



There is a non-random pattern in the plot of the outcome versus index that is perfectly explained by the FlyAsh variable.

**This should not be selected**

Incorrect

0 / 1 points

3.

Load the cement data using the commands:



1

2

3

4

5

6

7

library(AppliedPredictiveModeling)

data(concrete)

library(caret)

set.seed(1000)

inTrain = createDataPartition(mixtures$CompressiveStrength, p = 3/4)[[1]]

training = mixtures[ inTrain,]

testing = mixtures[-inTrain,]

Make a histogram and confirm the SuperPlasticizer variable is skewed. Normally you might use the log transform to try to make the data more symmetric. Why would that be a poor choice for this variable?



The SuperPlasticizer data include negative values so the log transform can not be performed.



The log transform does not reduce the skewness of the non-zero values of SuperPlasticizer



There are values of zero so when you take the log() transform those values will be -Inf.



The log transform produces negative values which can not be used by some classifiers.

**This should not be selected**

Correct

1 / 1 points

4.

Load the Alzheimer's disease data u