task3_nzv

Code

nzv

Hide

```
#nearZeroVar() with saveMetrics = TRUE returns an object containing a table including: f
requency ratio, percentage unique, zero variance and near zero variance

nzvMetrics <- nearZeroVar(iphone_smallMatrix, saveMetrics = TRUE)
nzvMetrics
```

	freqRatio <dbl>	percentUnique <dbl>	zeroVar <lgl>	nzv <lgl>
iphone	5.041322	0.20812457	FALSE	FALSE
samsunggalaxy	14.127336	0.05395822	FALSE	FALSE
sonyxperia	44.170732	0.03854159	FALSE	TRUE
nokialumina	497.884615	0.02312495	FALSE	TRUE
htcphone	11.439614	0.06937486	FALSE	FALSE
ios	27.735294	0.04624990	FALSE	TRUE
googleandroid	61.247573	0.04624990	FALSE	TRUE
iphonecampos	10.524697	0.23124952	FALSE	FALSE
samsungcampos	93.625000	0.08479149	FALSE	TRUE
sonycampos	348.729730	0.05395822	FALSE	TRUE

1-10 of 59 rows Previous **1** 2 3 4 5 6 Next

Hide

```
# returns column 2, iphonecamunc, same as nvzMetrics
# nearZeroVar() with saveMetrics = FALSE returns an vector
nzv <- nearZeroVar(iphone_smallMatrix, saveMetrics = FALSE)
nzv
```

```
 [1]  3  4  6  7  9 10 11 12 13 14 15 16 17 19 20 21 22 24 25 26 27 29 30 31 32 34 35 36
37 39 40
[32] 41 42 44 45 46 47 49 50 51 52 53 54 55 56 57 58
```

Hide

```
# create a new data set and remove near zero variance features
iphoneDFBigNZV <- iphone_smallMatrix[,-nzv]
str(iphoneDFBigNZV)
```

```
'data.frame':   12973 obs. of  12 variables:
 $ iphone         : int  1 1 1 1 1 41 1 1 1 1 ...
 $ samsunggalaxy  : int  0 0 0 0 0 0 0 0 0 0 ...
 $ htcphone       : int  0 0 0 0 0 0 0 0 0 0 ...
 $ iphonecampos   : int  0 0 0 0 0 1 1 0 0 0 ...
 $ iphonecamunc   : int  0 0 0 0 0 7 1 0 0 0 ...
 $ iphonedispos   : int  0 0 0 0 0 1 13 0 0 0 ...
 $ iphonedisneg   : int  0 0 0 0 0 3 10 0 0 0 ...
 $ iphonedisunc   : int  0 0 0 0 0 4 9 0 0 0 ...
 $ iphoneperpos   : int  0 1 0 1 1 0 5 3 0 0 ...
 $ iphoneperneg   : int  0 0 0 0 0 0 4 1 0 0 ...
 $ iphoneperunc   : int  0 0 0 1 0 0 5 0 0 0 ...
 $ iphonesentiment: int  0 0 0 0 0 4 4 0 0 0 ...
```

Model Building

Hide

```
# create 10-fold cross validation fitcontrol
fitControl <- trainControl(method = "cv", number = 10)
```

Model of dataframe without highly correlated variables

Hide

```
# convert variable types, categorical
iphoneDFBigNZV$iphonesentiment <- as.factor(iphoneDFBigNZV$iphonesentiment)
```

Train and Test Set:

Hide

```
# Create Train and Test Set for iphoneDFBig
# create 75% sample of row indices
in_training <-createDataPartition(iphoneDFBigNZV$iphonesentiment, p = .7, list = FALSE)
# create 75% sample of data and save it to trainData
trainData_iphoneDFBigNZV <- iphoneDFBigNZV[in_training, ]
 # create 25% sample of data and save it to test_data
testData_iphoneDFBigNZV <- iphoneDFBigNZV[-in_training, ]
# verify split percentages
nrow(trainData_iphoneDFBigNZV) / nrow(iphoneDFBigNZV)
```

```
[1] 0.7001465
```

Hide

```
#c5
c5_iphoneDFBigNZV <- train(iphonesentiment ~., data = trainData_iphoneDFBigNZV, method =
"C5.0",
                trControl = fitControl)
```

Hide

```
# randomforest
rf_iphoneDFBigNZV <- train(iphonesentiment ~., data = trainData_iphoneDFBigNZV, method =
"rf",
                trControl = fitControl)
```

We will run just c5 and random forest, since those two did the best on the dataset by itself as well as on the dataset with highly correlated variables removed.

Hide

```
# svm   (kernlab)
#svm_iphoneDFBigNZV <- train(iphonesentiment ~., data = trainData_iphoneDFBigNZV, method
= "svmLinear",
#                trControl = fitControl)
```

Hide

```
# kknn
#kknn_iphoneDFBigNZV <- train(iphonesentiment ~., data = trainData_iphoneDFBigNZV, metho
d = "kknn",
#                trControl = fitControl)
```

Hide

```
# gbm
#gbm_iphoneDFBigNZV <- train(iphonesentiment ~., data = trainData_iphoneDFBigNZV, method
= "gbm",
#                trControl = fitControl)
```

Compare Accuracy on Prediction Results:

Hide

```
#c5
prediction_c5_iphoneDFBigNZV <- predict(c5_iphoneDFBigNZV, testData_iphoneDFBigNZV)
postResample(prediction_c5_iphoneDFBigNZV, testData_iphoneDFBigNZV$iphonesentiment)
```

```
 Accuracy     Kappa
0.7575835 0.5247553
```

Hide

```
#randomforest
prediction_rf_iphoneDFBigNZV  <- predict(rf_iphoneDFBigNZV, testData_iphoneDFBigNZV)
postResample(prediction_rf_iphoneDFBigNZV, testData_iphoneDFBigNZV$iphonesentiment)
```

```
 Accuracy     Kappa
0.7632391 0.5340252
```

Hide

```


#svm
#prediction_svm_iphoneDFBigNZV  <- predict(svm_iphoneDFBigNZV, testData_iphoneDFBigNZV)
#postResample(prediction_svm_iphoneDFBigNZV, testData_iphoneDFBigNZV$iphonesentiment)
# kknn
#prediction_kknn_iphoneDFBigNZV  <- predict(kknn_iphoneDFBigNZV, testData_iphoneDFBigNZ
V)
#postResample(prediction_kknn_iphoneDFBigNZV, testData_iphoneDFBigNZV$iphonesentiment)
```

Hide

```
modelData_iphoneDFBigNZV <- resamples(list(C50 = c5_iphoneDFBigNZV, randomForest = rf_ip
honeDFBigNZV))
```

Hide

```
#, svMLinear = svm_iphoneDFBigNZV,kknn = kknn_iphoneDFBigNZV
```

Hide

```
summary(modelData_iphoneDFBigNZV)
```

```


Call:
summary.resamples(object = modelData_iphoneDFBigNZV)

Models: C50, randomForest
Number of resamples: 10

Accuracy
                  Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
C50          0.7348735 0.7514448 0.7546849 0.7554771 0.7618761 0.7720264    0
randomForest 0.7400881 0.7524079 0.7625074 0.7593243 0.7671350 0.7742291    0

Kappa
                  Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
C50          0.4734614 0.5105674 0.5215799 0.5206099 0.5355079 0.5556722    0
randomForest 0.4816587 0.5069005 0.5333482 0.5256350 0.5446478 0.5631583    0
```