Daniel Prusinski

412

Appendix 1 for Assignment 3 – Classification Models

Output 1:

print(str(spambase))

'data.frame': 4601 obs. of 60 variables:

$ make : num 0 0.21 0.06 0 0 0 0 0 0.15 0.06 ...

$ address : num 0.64 0.28 0 0 0 0 0 0 0 0.12 ...

$ all : num 0.64 0.5 0.71 0 0 0 0 0 0.46 0.77 ...

$ xd : num 0 0 0 0 0 0 0 0 0 0 ...

$ our : num 0.32 0.14 1.23 0.63 0.63 1.85 1.92 1.88 0.61 0.19 ...

$ over : num 0 0.28 0.19 0 0 0 0 0 0 0.32 ...

$ remove : num 0 0.21 0.19 0.31 0.31 0 0 0 0.3 0.38 ...

$ internet : num 0 0.07 0.12 0.63 0.63 1.85 0 1.88 0 0 ...

$ order : num 0 0 0.64 0.31 0.31 0 0 0 0.92 0.06 ...

$ mail : num 0 0.94 0.25 0.63 0.63 0 0.64 0 0.76 0 ...

$ receive : num 0 0.21 0.38 0.31 0.31 0 0.96 0 0.76 0 ...

$ will : num 0.64 0.79 0.45 0.31 0.31 0 1.28 0 0.92 0.64 ...

$ people : num 0 0.65 0.12 0.31 0.31 0 0 0 0 0.25 ...

$ report : num 0 0.21 0 0 0 0 0 0 0 0 ...

$ addresses : num 0 0.14 1.75 0 0 0 0 0 0 0.12 ...

$ free : num 0.32 0.14 0.06 0.31 0.31 0 0.96 0 0 0 ...

$ business : num 0 0.07 0.06 0 0 0 0 0 0 0 ...

$ email : num 1.29 0.28 1.03 0 0 0 0.32 0 0.15 0.12 ...

$ you : num 1.93 3.47 1.36 3.18 3.18 0 3.85 0 1.23 1.67 ...

$ credit : num 0 0 0.32 0 0 0 0 0 3.53 0.06 ...

$ your : num 0.96 1.59 0.51 0.31 0.31 0 0.64 0 2 0.71 ...

$ font : num 0 0 0 0 0 0 0 0 0 0 ...

$ x000 : num 0 0.43 1.16 0 0 0 0 0 0 0.19 ...

$ money : num 0 0.43 0.06 0 0 0 0 0 0.15 0 ...

$ hp : num 0 0 0 0 0 0 0 0 0 0 ...

$ hpl : num 0 0 0 0 0 0 0 0 0 0 ...

$ george : num 0 0 0 0 0 0 0 0 0 0 ...

$ x650 : num 0 0 0 0 0 0 0 0 0 0 ...

$ lab : num 0 0 0 0 0 0 0 0 0 0 ...

$ labs : num 0 0 0 0 0 0 0 0 0 0 ...

$ telnet : num 0 0 0 0 0 0 0 0 0 0 ...

$ x857 : num 0 0 0 0 0 0 0 0 0 0 ...

$ data : num 0 0 0 0 0 0 0 0 0.15 0 ...

$ x415 : num 0 0 0 0 0 0 0 0 0 0 ...

$ x85 : num 0 0 0 0 0 0 0 0 0 0 ...

$ technology: num 0 0 0 0 0 0 0 0 0 0 ...

$ x1999 : num 0 0.07 0 0 0 0 0 0 0 0 ...

$ parts : num 0 0 0 0 0 0 0 0 0 0 ...

$ pm : num 0 0 0 0 0 0 0 0 0 0 ...

$ direct : num 0 0 0.06 0 0 0 0 0 0 0 ...

$ cs : num 0 0 0 0 0 0 0 0 0 0 ...

$ meeting : num 0 0 0 0 0 0 0 0 0 0 ...

$ original : num 0 0 0.12 0 0 0 0 0 0.3 0 ...

$ project : num 0 0 0 0 0 0 0 0 0 0.06 ...

$ re : num 0 0 0.06 0 0 0 0 0 0 0 ...

$ edu : num 0 0 0.06 0 0 0 0 0 0 0 ...

$ table : num 0 0 0 0 0 0 0 0 0 0 ...

$ conference: num 0 0 0 0 0 0 0 0 0 0 ...

$ x. : num 0 0 0.01 0 0 0 0 0 0 0.04 ...

$ x.. : num 0 0.132 0.143 0.137 0.135 0.223 0.054 0.206 0.271 0.03 ...

$ x...1 : num 0 0 0 0 0 0 0 0 0 0 ...

$ x..1 : num 0.778 0.372 0.276 0.137 0.135 0 0.164 0 0.181 0.244 ...

$ x..2 : num 0 0.18 0.184 0 0 0 0.054 0 0.203 0.081 ...

$ x..3 : num 0 0.048 0.01 0 0 0 0 0 0.022 0 ...

$ crla : num 3.76 5.11 9.82 3.54 3.54 ...

$ crll : int 61 101 485 40 40 15 4 11 445 43 ...

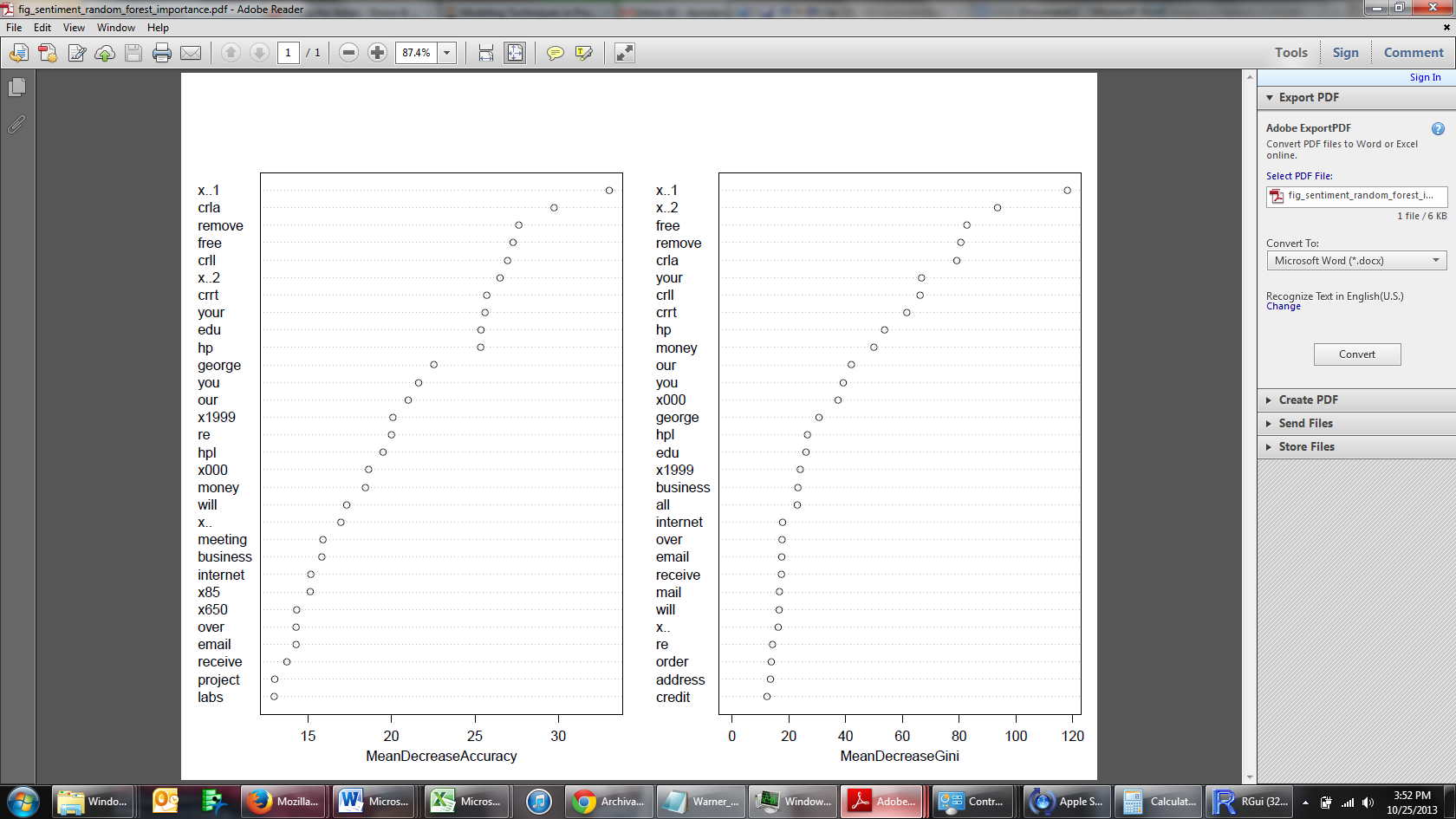
$ crrt : int 278 1028 2259 191 191 54 112 49 1257 749 ...

$ classdigit: Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...

$ class : Factor w/ 2 levels "email","spam": 2 2 2 2 2 2 2 2 2 2 ...

$ Group : Factor w/ 2 levels "TRAIN","TEST": 2 1 2 1 2 2 1 2 2 2 ...

0utput 2



Output 3

Call:

randomForest(formula = spam.classification.model, data = spambase.train, mtry = 3, importance = TRUE, na.action = na.omit)

Type of random forest: classification

Number of trees: 500

No. of variables tried at each split: 3

OOB estimate of error rate: 5.22%

Confusion matrix:

email spam class.error

email 1787 57 0.03091106

spam 103 1120 0.08421913

Output 4: Training Data

Reference

Prediction email spam

email 1839 37

spam 5 1186

Accuracy : 0.9863

95% CI : (0.9815, 0.9901)

No Information Rate : 0.6012

P-Value [Acc > NIR] : < 2.2e-16

Kappa : 0.9713

Mcnemar's Test P-Value : 1.724e-06

Sensitivity : 0.9973

Specificity : 0.9697

Pos Pred Value : 0.9803

Neg Pred Value : 0.9958

Prevalence : 0.6012

Detection Rate : 0.5996

Detection Prevalence : 0.6117

'Positive' Class : email

Output 5: Testing Data

Confusion Matrix and Statistics

Reference

Prediction email spam

email 917 54

spam 27 536

Accuracy : 0.9472

95% CI : (0.9348, 0.9578)

No Information Rate : 0.6154

P-Value [Acc > NIR] : < 2.2e-16

Kappa : 0.8875

Mcnemar's Test P-Value : 0.003866

Sensitivity : 0.9714

Specificity : 0.9085

Pos Pred Value : 0.9444

Neg Pred Value : 0.9520

Prevalence : 0.6154

Detection Rate : 0.5978

Detection Prevalence : 0.6330

'Positive' Class : email

Output 7: Neural Network

confusionMatrix(data = spambase.train$nnet,

+ reference = spambase.train$class)

Confusion Matrix and Statistics

Reference

Prediction email spam

email 1824 56

spam 48 1139

Accuracy : 0.9661

95% CI : (0.9591, 0.9722)

No Information Rate : 0.6104

P-Value [Acc > NIR] : <2e-16

Kappa : 0.9286

Mcnemar's Test P-Value : 0.4925

Sensitivity : 0.9744

Specificity : 0.9531

Pos Pred Value : 0.9702

Neg Pred Value : 0.9596

Prevalence : 0.6104

Detection Rate : 0.5947

Detection Prevalence : 0.6130

'Positive' Class : email

confusionMatrix(data = spambase.test$pred.rf,

+ reference = spambase.test$class)

Confusion Matrix and Statistics

Reference

Prediction email spam

email 875 69

spam 41 549

Accuracy : 0.9283

95% CI : (0.9142, 0.9407)

No Information Rate : 0.5971

P-Value [Acc > NIR] : < 2e-16

Kappa : 0.8499

Mcnemar's Test P-Value : 0.01004

Sensitivity : 0.9552

Specificity : 0.8883

Pos Pred Value : 0.9269

Neg Pred Value : 0.9305

Prevalence : 0.5971

Detection Rate : 0.5704

Detection Prevalence : 0.6154

'Positive' Class : email

Output 8: Support Vector Machine

Training Data

confusionMatrix(data = spambase.train$svm,

+ reference = spambase.train$class)

Confusion Matrix and Statistics

Reference

Prediction email spam

email 1790 107

spam 82 1088

Accuracy : 0.9384

95% CI : (0.9293, 0.9466)

No Information Rate : 0.6104

P-Value [Acc > NIR] : < 2e-16

Kappa : 0.8699

Mcnemar's Test P-Value : 0.08086

Sensitivity : 0.9562

Specificity : 0.9105

Pos Pred Value : 0.9436

Neg Pred Value : 0.9299

Prevalence : 0.6104

Detection Rate : 0.5836

Detection Prevalence : 0.6185

'Positive' Class : email

Testing Data

confusionMatrix(data = spambase.test$pred.rf,

+ reference = spambase.test$class)

Confusion Matrix and Statistics

Reference

Prediction email spam

email 875 69

spam 41 549

Accuracy : 0.9283

95% CI : (0.9142, 0.9407)

No Information Rate : 0.5971

P-Value [Acc > NIR] : < 2e-16

Kappa : 0.8499

Mcnemar's Test P-Value : 0.01004

Sensitivity : 0.9552

Specificity : 0.8883

Pos Pred Value : 0.9269

Neg Pred Value : 0.9305

Prevalence : 0.5971

Detection Rate : 0.5704

Detection Prevalence : 0.6154

'Positive' Class : email

Output 9: Stepwise Logistic Regression

Training Data

confusionMatrix(data = spambase.train$pred.lr,

+ reference = spambase.train$class)

Confusion Matrix and Statistics

Reference

Prediction email spam

email 1791 120

spam 81 1075

Accuracy : 0.9345

95% CI : (0.9251, 0.943)

No Information Rate : 0.6104

P-Value [Acc > NIR] : < 2.2e-16

Kappa : 0.8614

Mcnemar's Test P-Value : 0.007355

Sensitivity : 0.9567

Specificity : 0.8996

Pos Pred Value : 0.9372

Neg Pred Value : 0.9299

Prevalence : 0.6104

Detection Rate : 0.5840

Detection Prevalence : 0.6231

'Positive' Class : email

Testing Data

confusionMatrix(data = spambase.test$pred.lr,

+ reference = spambase.test$class, positive = "email")

Confusion Matrix and Statistics

Reference

Prediction email spam

email 875 78

spam 41 540

Accuracy : 0.9224

95% CI : (0.9079, 0.9353)

No Information Rate : 0.5971

P-Value [Acc > NIR] : < 2.2e-16

Kappa : 0.8372

Mcnemar's Test P-Value : 0.0009665

Sensitivity : 0.9552

Specificity : 0.8738

Pos Pred Value : 0.9182

Neg Pred Value : 0.9294

Prevalence : 0.5971

Detection Rate : 0.5704

Detection Prevalence : 0.6213

'Positive' Class : email

Output 10: Naive Bayes

Training Data

confusionMatrix(data = spambase.train$naivebayes\_class,

+ reference = spambase.train$class)

Confusion Matrix and Statistics

Reference

Prediction email spam

email 1046 69

spam 826 1126

Accuracy : 0.7082

95% CI : (0.6917, 0.7242)

No Information Rate : 0.6104

P-Value [Acc > NIR] : < 2.2e-16

Kappa : 0.4495

Mcnemar's Test P-Value : < 2.2e-16

Sensitivity : 0.5588

Specificity : 0.9423

Pos Pred Value : 0.9381

Neg Pred Value : 0.5768

Prevalence : 0.6104

Detection Rate : 0.3410

Detection Prevalence : 0.3635

'Positive' Class : email

Testing Data Naive Bayes

confusionMatrix(data = spambase.test$pred.rf,

+ reference = spambase.test$class)

Confusion Matrix and Statistics

Reference

Prediction email spam

email 495 45

spam 421 573

Accuracy : 0.6962

95% CI : (0.6725, 0.7192)

No Information Rate : 0.5971

P-Value [Acc > NIR] : 5.19e-16

Kappa : 0.4255

Mcnemar's Test P-Value : < 2.2e-16

Sensitivity : 0.5404

Specificity : 0.9272

Pos Pred Value : 0.9167

Neg Pred Value : 0.5765

Prevalence : 0.5971

Detection Rate : 0.3227

Detection Prevalence : 0.3520

'Positive' Class : email