SPRAWOZDANIE

Klasyfikacja

Krzyszczuk Michał

7 stycznia 2018

1(Wczytanie Danych):

```
dane <- iris
dane$Species<- as.character(dane$Species)
for(j in c(101:150)) dane$Species[j] <- "versicolor"
dane$Species <- as.factor(dane$Species)
zbiory <- split(dane, sample(rep(1:2,c(100,50))))
zbior_uczacy <- zbiory$'1'
zbior_testowy <- zbiory$'2'
```

2(Regresja logistyczna):

```
#Przeprowadzam regresje logistyczna
regresjalog <- glm(Species ~Sepal.Length +Sepal.Width +Petal.Length +Petal.Width, data = zbior_uczacy, f
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

Wykonuje przewidywanie dla zbioru testowego

```
przewidywanie <- predict(regresjalog,newdata =zbior_testowy)
for (i in 1:length(przewidywanie)) {
  if(przewidywanie[i]>0.5)przewidywanie[i]=1
  else
    przewidywanie[i]=0
}
table(zbior_testowy$Sepal.Length)
```

```
##
## 4.6 4.8 4.9   5 5.1 5.4 5.5 5.6 5.7 5.8 5.9   6 6.1 6.2 6.3 6.4 6.5 6.7
##   2   3   1   2   3   4   2   2   3   2   2   2   1   2   4   4   2   2
## 6.8 6.9   7 7.1 7.7
##   3   1   1   1   1
```

Pomimo wielokrotnych prób stworzenia macierzy pomyłek przy użyciu funkcji ***confusionMatrix()*** z pakietu caret nie udało się zrealizować tego zadania. Powyższa funkcja oblicza wszystkie parametry konieczne i niezbędne do oceny jakości klasyfikacji(ACC,TNR,P-value,...). Otrzymywane raporty błędów: * The data must contain some levels that overlap the reference. * The data cannot have more levels than the reference. Pomimo wielu prób rozwiązania problemu na forach statystycznych oraz programistycznych nie udało się rozwiązać tego zagadnienia przy użyciu tego narzędzia.

```
cm <-table(przewidywanie)
cm
```

```
## przewidywanie
##  0  1
## 13 37
```

```
cat("Accuracy: ",round(sum(diag(cm))/sum(cm),3),"\n")#wynik wskazuje że metoda raczej błędna
```

```
## Accuracy:  1
```

3(Regresja logistyczna + PCA)

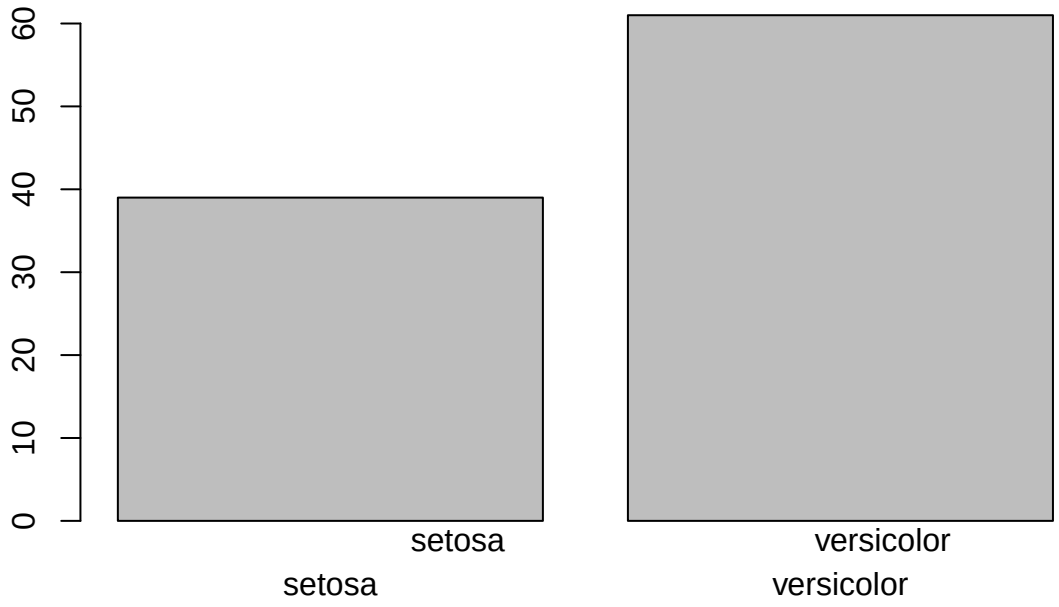
```
new_iris <-zbiory$`1`
new_iris$Species <-NULL
princ <-prcomp(as.matrix(new_iris))
species1 <-zbiory$`1`$Species
var1 <-data.frame(species1,princ$x)
model2 <-glm(species1 ~PC1 +PC2 +PC3 +PC4, data <-var1, family ='binomial',control = list(maxit = 50))
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
new_iris_testing <-zbiory$`2`
new_iris_testing$Species <-NULL
princ2 <-prcomp(as.matrix(new_iris_testing))
species1  <-zbiory$`2`$Species
var2 <-data.frame(species1,princ2$x)
pcapred <-predict(princ, zbiory$`2`)
var3 <-data.frame(pcapred,zbiory$`2`$Species)
prediction1 <-predict(model2, var3)
```

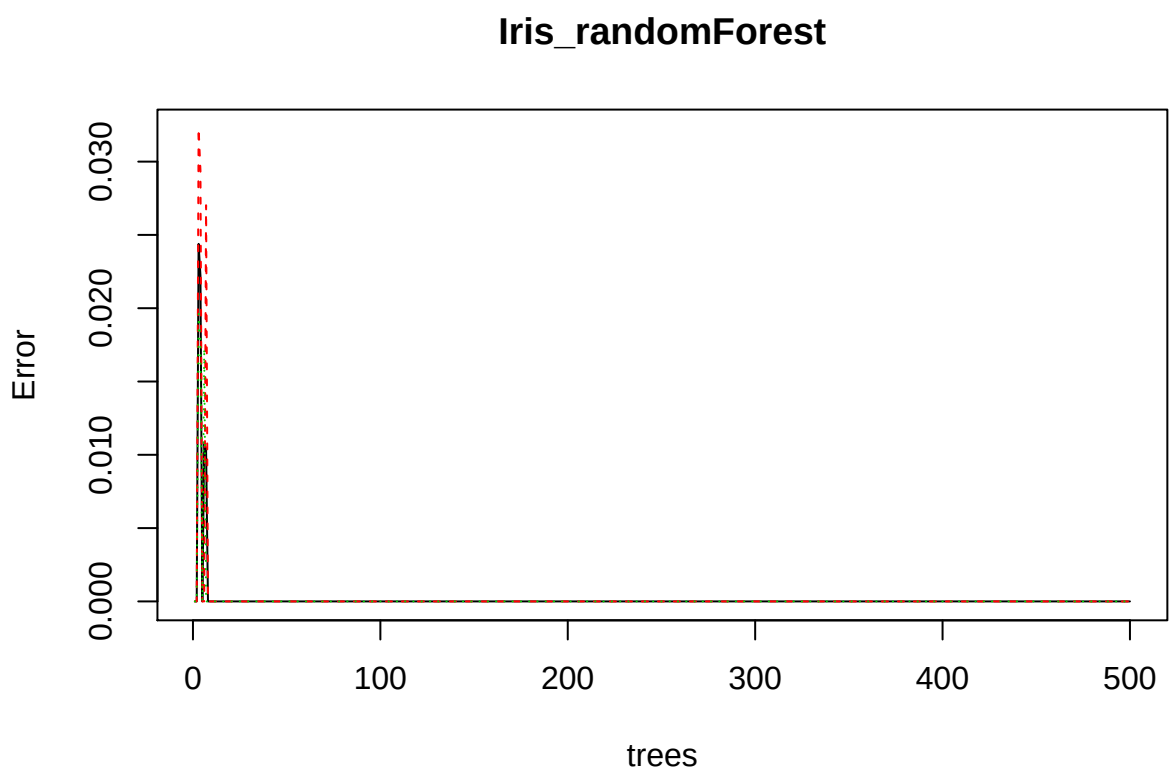
4(Drzewo)

```
library(rpart)
tree_data_iris<-split(dane, sample(rep(1:2,c(100,50))))
tree <-tree(Species ~., data=tree_data_iris$`1`)
prediction_tree <-predict(tree,tree_data_iris$'2')
plot(tree$y)
text(tree)
```



5(Las losowy)

```
Iris_randomForest <- randomForest(Species ~. , data=tree_data_iris$`1`, method="class")
prediction_Iris_randomForest <- predict(Iris_randomForest, tree_data_iris$`1`, type = "class")
plot(Iris_randomForest,type="l")
```




```
#confusionMatrix(prediction_Iris_randomForest, tree_data_iris$`1`$Sepal.Length)
```

```
library(e1071)
```

```
## Warning: package 'e1071' was built under R version 3.4.3
```

```
naive <-naiveBayes(Species ~., data=tree_data_iris$`1`)
prediction_naiveBayes <-predict(naive,tree_data_iris$'2')
table( tree_data_iris$`2`$Species,prediction_naiveBayes)
```

```
##              prediction_naiveBayes
##               setosa versicolor
##    setosa         11          0
##    versicolor      0         39
```