Lab 9 - zadanie domowe

Mateusz Kapusta

2022-05-21

Zadanie nr 2

```
data(iris)
w<-sample.split(iris$Species,SplitRatio = 1/2)
tren<-iris[w,]
test<-iris[!w,]
tren["czysetoza"]=(tren$Species=="setosa")
tren["czyversi"]=(tren$Species=="versicolor")
tren["czyvirginica"]=(tren$Species=="virginica")
m1<-glm(czysetoza~Sepal.Length,family=binomial,data=tren)
m2<-glm(czyversi~Sepal.Length,family=binomial,data=tren)
m3<-glm(czyvirginica~Sepal.Length,family=binomial,data=tren)
d<-data.frame("set"=predict(m1,type="response",test),
              "versi"=predict(m2,type="response",test),
              "virginica"=predict(m3,type="response",test))
pred<-apply(d, MARGIN=1,FUN=which.max)
true<-sapply(test$Species,as.numeric)
t<-table('Predicted'=pred,'True'=true)
print(t)
```

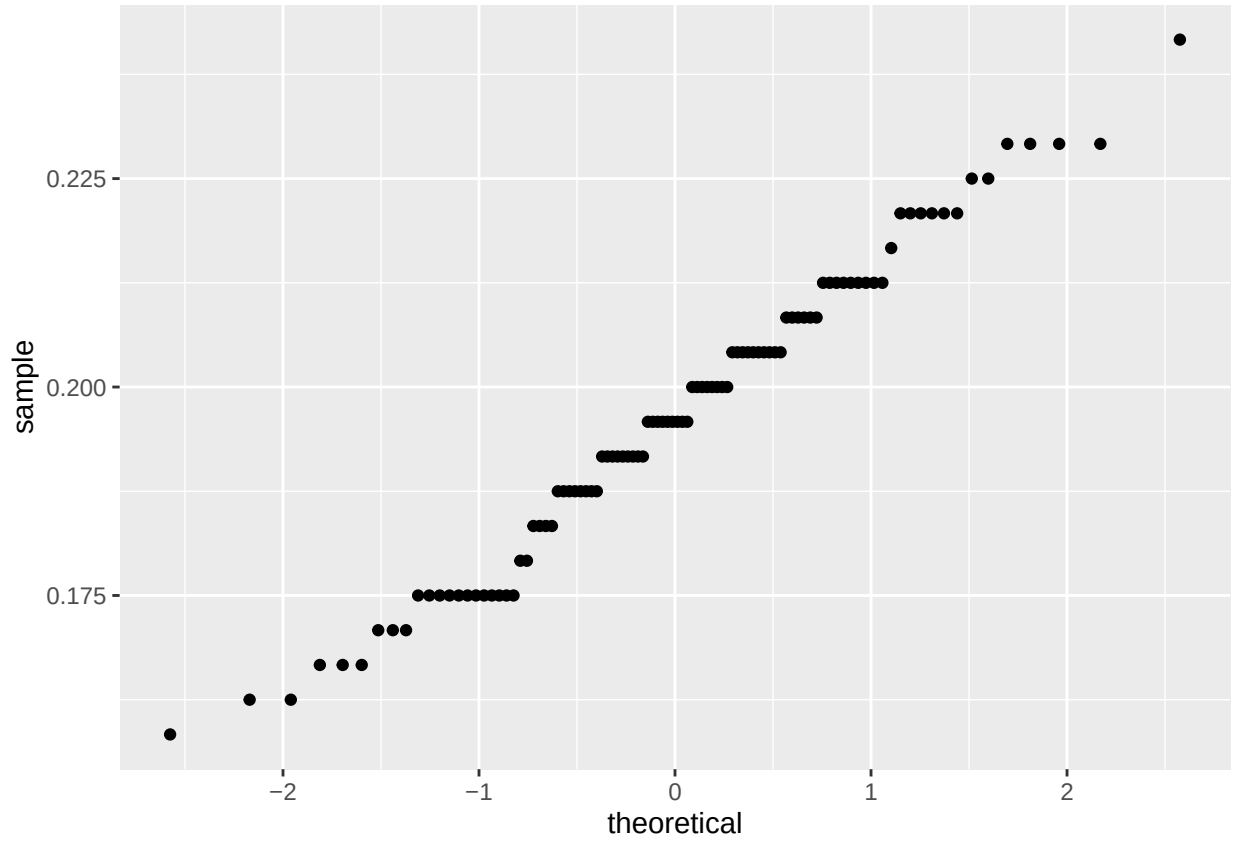
```
##          True
## Predicted  1  2  3
##         1 21  3  1
##         2  4 13  3
##         3  0  9 21
```

Liczba jeden odpowiada gatunkowi Setosa, dwa Versicolor a trzy Virginica. Prezycja klasyfikatora to 0.84, 0.52, 0.84. Widzimy, że nalepiej klasyfikujemy gatunek Virginica. Ogólne accuracy to 0.7333333.

Zadanie nr 3

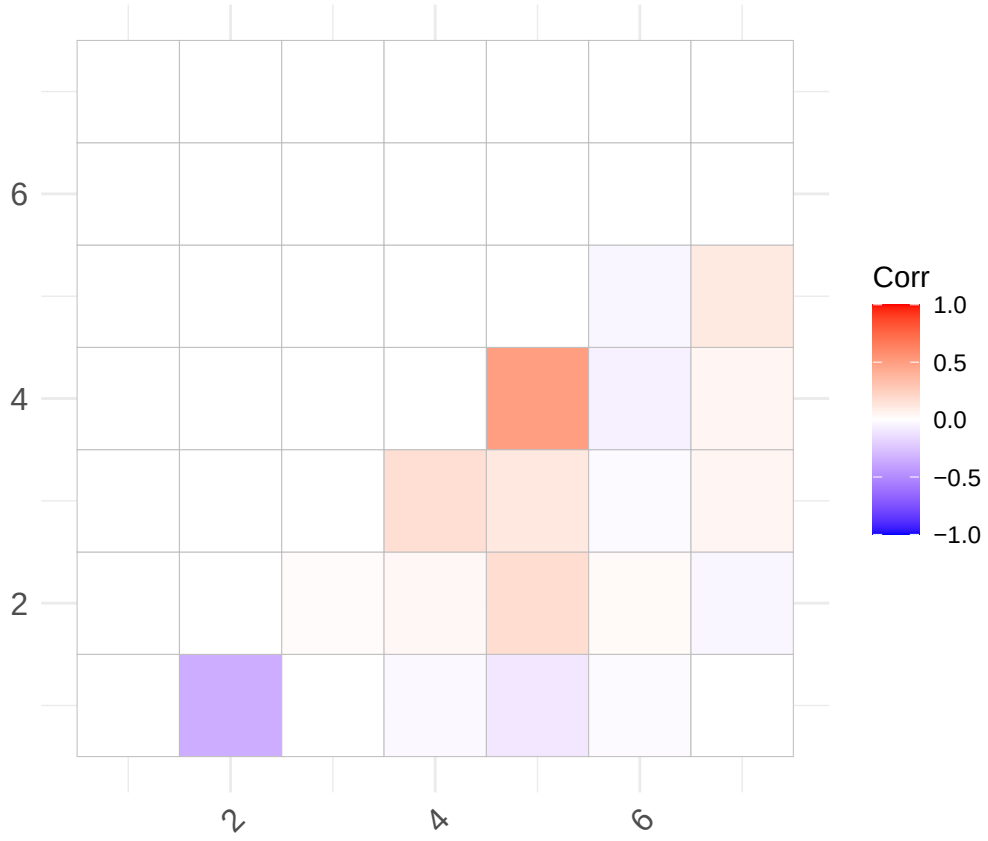
```
dane <- read.table("pozyczka.csv", sep=";", header=T)
pred<-numeric(100)
for (i in 1:100)
{
  train_ind <- sample(seq_len(nrow(dane)), size = nrow(dane)/2)
  train<-dane[train_ind,]
  test<-dane[-train_ind,]
  m<-lda(pozyczka~.,data=train,type="response")
  predy<-as.numeric(predict(m,test)$class)-1
  pred[i]<-mean((predy-test$pozyczka)^2)
```

```
}
d<-data.frame("data"=pred,"p"=1:100)
ggplot(d,aes(sample=data))+stat_qq()
```

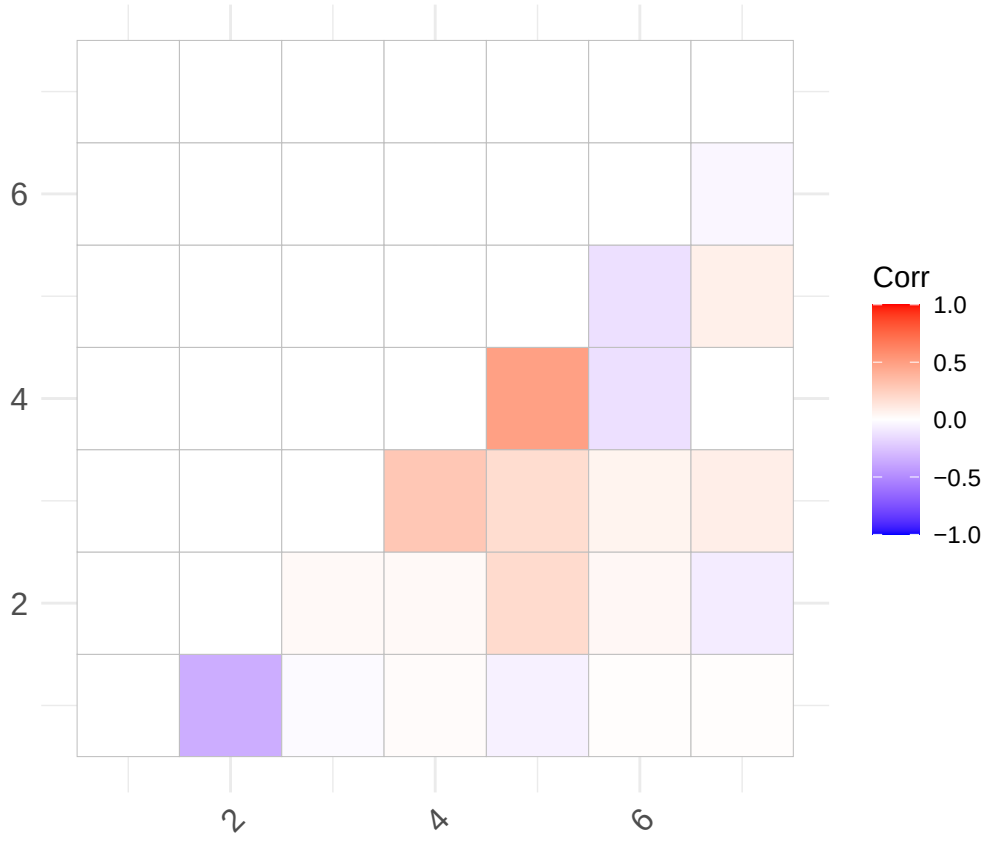


Widzimy, że rozkład błędów pochodzi z grubsza z rozkładu normalnego jak oczekiwaliśmy. Teraz polcizmy korelację predykatorów. Poniższy fragment kodu liczy macierz korelacji a następnie przedstawia nam je na wykresie w koljeności korelacja dla wszystkich danych, dla tych co spłacili pożyczkę i dla tych co tego nie zrobili.

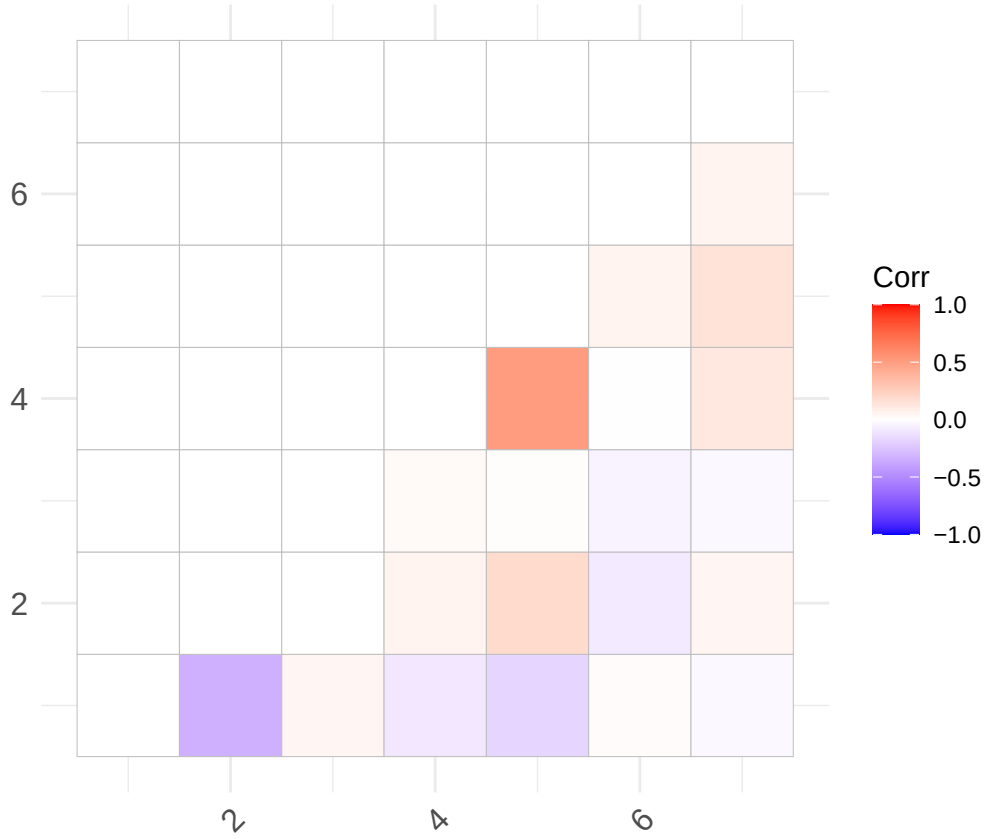
```
mat<-matrix(0,7,7)
mat1<-matrix(0,7,7)
mat2<-matrix(0,7,7)
for (i in 1:7)
{
  for (j in 1:7)
  {
    if (i>j)
    {
      mat[i,j]<-cor(dane[,i],dane[,j])
      mat1[i,j]<-cor(dane[dane$pozyczka==1,i],dane[dane$pozyczka==1,j])
      mat2[i,j]<-cor(dane[dane$pozyczka==0,i],dane[dane$pozyczka==0,j])
    }
  }
}
ggcorrplot(mat)
```




```
ggcorrplot(mat1)
```




```
ggcorrplot(mat2)
```



Ze względu na rózne macierzy kowariancji najprawdopodobniej qda będzie lepsza metodą.

```
train_ind <- sample(seq_len(nrow(dane)), size = nrow(dane)/2)
  train<-dane[train_ind,]
  test<-dane[-train_ind,]
  m<-qda(pozyczka~.,data=train,type="response")
  predy<-as.numeric(predict(m,test)$class)-1
  pred_test<-mean((predy-test$pozyczka)^2)
  predy_train<-as.numeric(predict(m,train)$class)-1
  pred_train<-mean((predy_train-train$pozyczka)^2)
```

Błędy średniokwadratowe dla zbioru treningowego i testowego to kolejno 0.2125 praz 0.1708333