Úvod do strojového učení v systému R
(bh-eHW3)

Jakub Genči

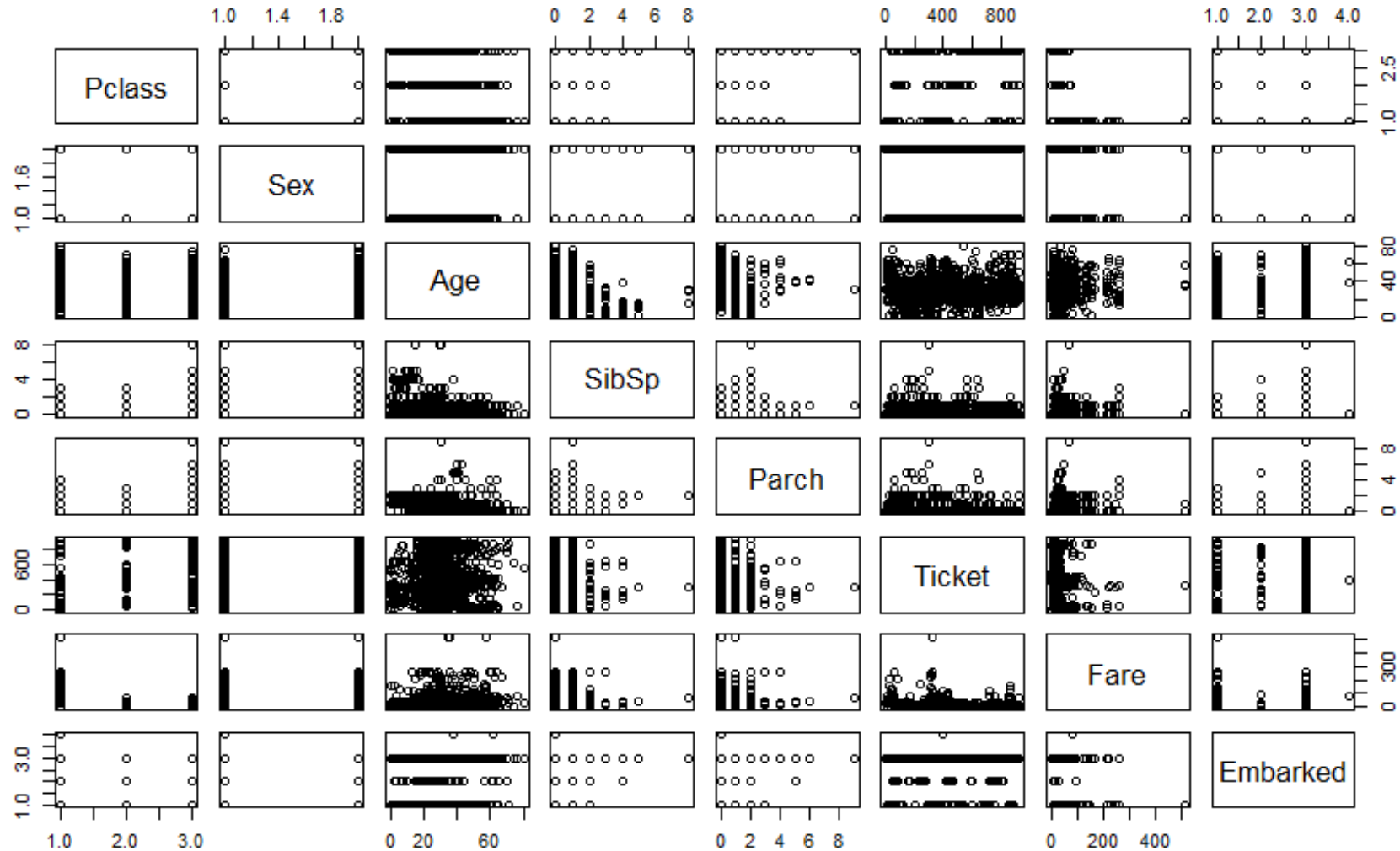
8.4.2022

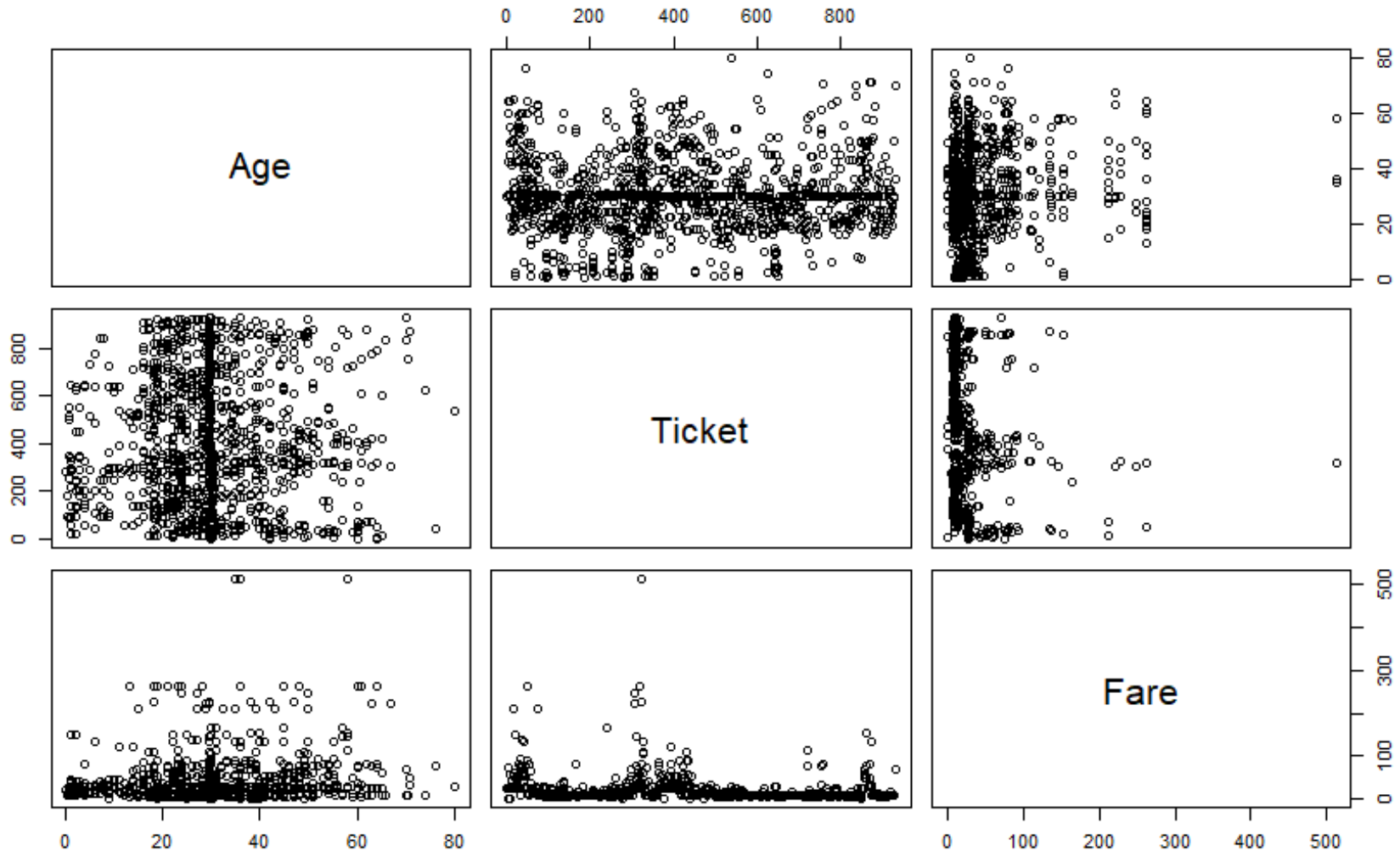
Úloha 1

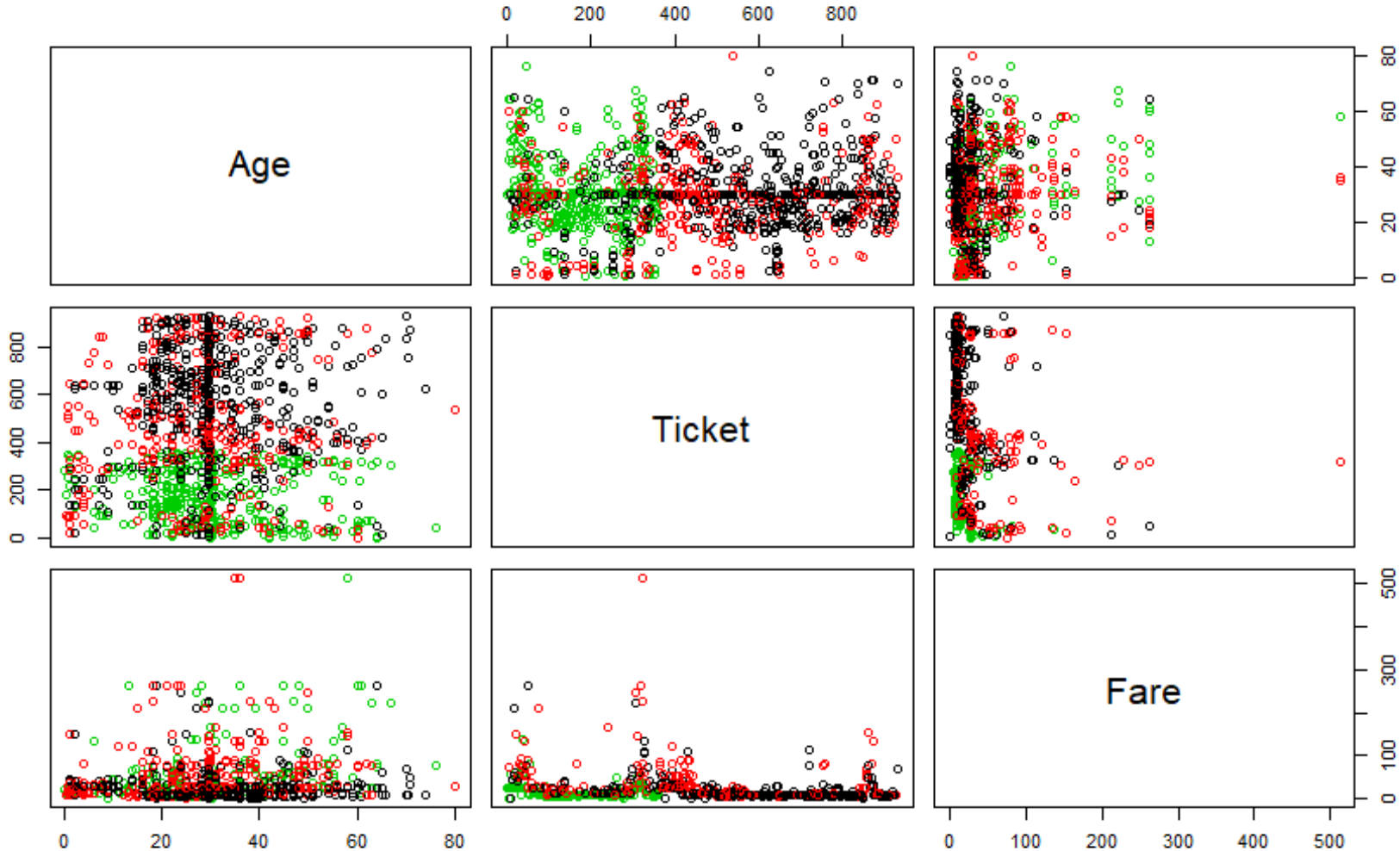
1. Load the Titanic data sets, both the train and test set, and merge them into a single data set. Explore this set graphically using tools of your choice. Create some plots highlighting the relationships among the attributes. Comment on your findings.

- Train dataset – 891 pozorovaní, 12 atribútov
- Test dataset – 418 pozorovaní, 11 atribútov

- Atribút ‘Cabin’ odstránený, chýbajúce hodnoty pre vek nahradené priemerom


Pclass
Sex
Age
SibSp
Parch
Ticket
Fare
Embarked


Age
Ticket
Fare
0
200
400
600
800
0
20
40
60
80
0
100
200
300
400
500


Age
Ticket
Fare
0
200
400
600
800
0
20
40
60
80
0
100
200
300
400
500

Úloha 2

2. Load the Titanic `train` data set and split it into a training set and test set in 90:10 ratio. Using the training data set fit logistic regression models with `Survived` as a target binary attribute. Experiment with different subsets of the given features. Do not forget to handle the missing values using a reasonable method. Evaluate your models on the test data set using the measures Accuracy, Precision, Recall, and F-measure.

- Nepracujeme s test datasetom (nemá atribút ‘Survived’), ale iba s trénovacím, ktorý rozdelíme

- Chýbajúce hodnoty sme ošetrili v prvej úlohe

```r
# Q2
# Test set doesn't have 'Survived' data, therefore we can't evaluate the classifier on it
titanic_train2 = titanic_train

# convert survived into factor => survived = 1
titanic_train2$Survived = factor(titanic_train2$Survived, levels = c(0,1))

# Setting seed just for reproducibility
set.seed(42)
indices = sample(nrow(titanic_train))
split_index = round(0.9 * nrow(titanic_train))  # just to simplify
train_train = titanic_train2[indices[1:split_index],]
train_test = titanic_train2[indices[(split_index + 1):nrow(titanic_train2)],]

# Generating the model
m1 = glm(Survived ~ Pclass + Sex + Age + SibSp + Parch + Fare + Embarked,
         data = train_train, family = binomial(link = "logit"))
# Getting from predictions (log odds) to confusion matrix
p1 = predict.glm(m1, train_test, type = "response")
y1 = ifelse(p1 > 0.5, 1, 0)
cm1 = table(train_test[,2], y1)
```

```
  y1
    0  1
0 45  8
1 16 20
```

```
Accuracy of the model 1 is 0.7303371
Precision of the model 1 is 0.7377049
Recall of the model 1 is 0.8490566
F-measure of the model 1 is 0.7894737
```

Úloha 3

3. Load the Movie data set and split it into a train set and test set in 90:10 ratio. Using the train set fit logistic regression models with `rating` as a target categorical attribute having 5 different values. Use `one-to-all` method for multi-class classification. Experiment with different subsets of the given features. Evaluate your models on the test set using the measures Accuracy, Precision, Recall, and F-measure.

- Movie dataset – 100 000 pozorovaní, 33 atribútov
- Cieľový atribút (rating) má hodnoty od 1 do 5

- One-to-all klasifikátor

```
# splitting into train and test set
set.seed(42)
indices = sample(nrow(movies))
split_index = round(0.9 * nrow(movies))  # just to simplify
movies_train = movies[indices[1:split_index],]
movies_test = movies[indices[(split_index + 1):nrow(movies)],]

# Splitting training set into 5 subsets with each having rating as a 2 level factor
m_tr1 = movies_train
m_tr1$rating = factor(m_tr1$rating, levels = c(1,2,3,4,5), labels = c(1,0,0,0,0))

m_tr2 = movies_train
m_tr2$rating = factor(m_tr2$rating, levels = c(1,2,3,4,5), labels = c(0,1,0,0,0))

m_tr3 = movies_train
m_tr3$rating = factor(m_tr3$rating, levels = c(1,2,3,4,5), labels = c(0,0,1,0,0))

m_tr4 = movies_train
m_tr4$rating = factor(m_tr4$rating, levels = c(1,2,3,4,5), labels = c(0,0,0,1,0))

m_tr5 = movies_train
m_tr5$rating = factor(m_tr5$rating, levels = c(1,2,3,4,5), labels = c(0,0,0,0,1))

# Fitting models on the train set
m1_1 = glm(rating ~ . - timestamp - title - release_date - imdb_url,
           data = m_tr1, family = binomial(link = "logit"))
```

```
Error: cannot allocate vector of size 3.4 Gb
```

- Odstránenie zbytočných(?) stĺpcov

```
movies = subset(movies, select = -zip)
movies = subset(movies, select = -timestamp)
movies = subset(movies, select = -title)
movies = subset(movies, select = -release_date)
movies = subset(movies, select = -imdb_url)
movies = subset(movies, select = -directors)
movies = subset(movies, select = -writers)
movies = subset(movies, select = -stars)
```

- Confusion matrix

```
 classifications
    1   2   3   4   5
1 126 112 107 143 111
2 224 230 235 245 210
3 508 541 523 554 534
4 711 685 726 721 666
5 413 422 412 412 429
```

- Ako vyhodnotiť klasifikátor?

```
# Function for computing evaluation parameters
get_stats = function(cm, wanted_rating){
  TP = cm[wanted_rating, wanted_rating]
  TN = sum(cm[-wanted_rating, -wanted_rating])
  FP = sum(cm[,wanted_rating]) - TP
  FN = sum(cm[wanted_rating,]) - TP

  p = TP/(TP+FP)
  r = TP/(TP+FN)

  cat("Accuracy of the model", wanted_rating  ,"is", (TP+TN)/(sum(cm)) ,"\n")
  cat("Precision of the model", wanted_rating  ,"is", p, "\n")
  cat("Recall of the model", wanted_rating  ,"is", r, "\n")
  cat("F-measure of the model", wanted_rating  ,"is", 2*((p * r)/(p + r)),"\n")
  cat("\n")
}
```

- Výsledky

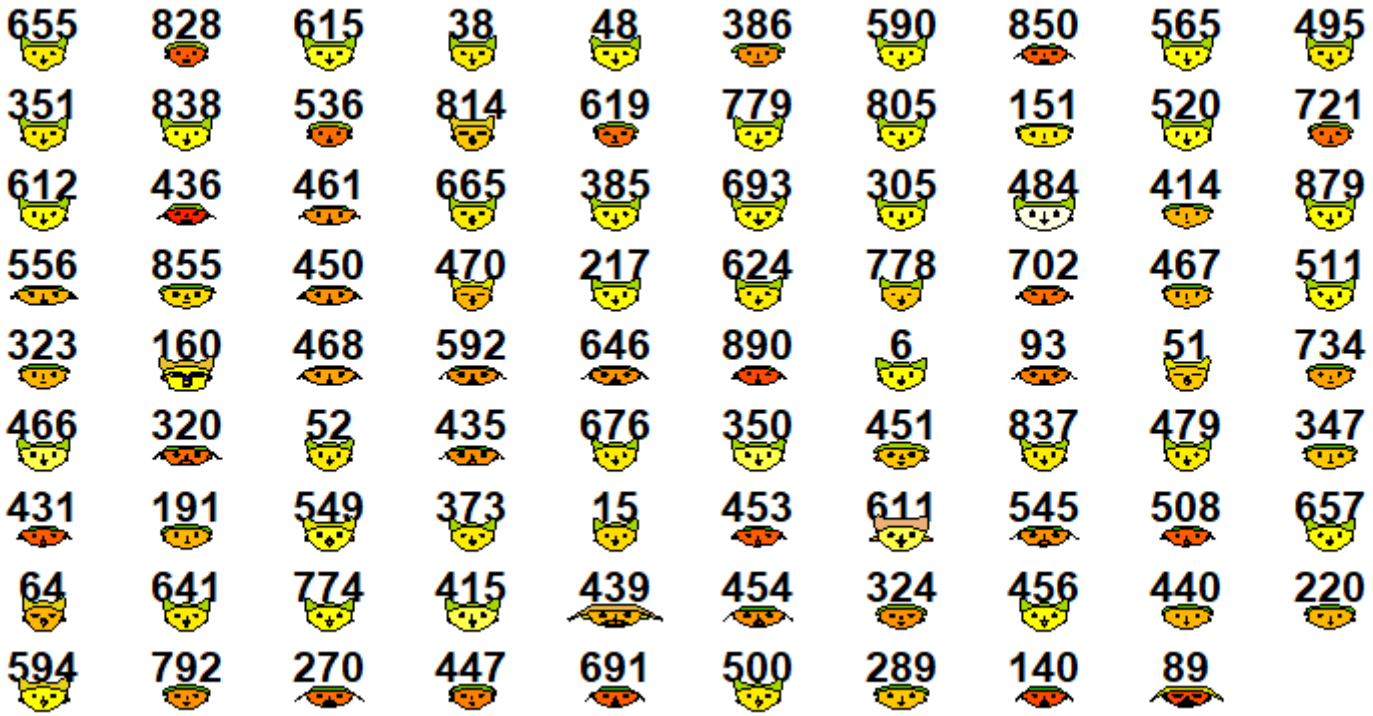
```
Accuracy of the model 1 is 0.7671
Precision of the model 1 is 0.06357215
Recall of the model 1 is 0.2103506
F-measure of the model 1 is 0.09763657

Accuracy of the model 2 is 0.7326
Precision of the model 2 is 0.1155779
Recall of the model 2 is 0.201049
F-measure of the model 2 is 0.1467773

Accuracy of the model 3 is 0.6383
Precision of the model 3 is 0.2611083
Recall of the model 3 is 0.1966165
F-measure of the model 3 is 0.2243191

Accuracy of the model 4 is 0.5858
Precision of the model 4 is 0.3474699
Recall of the model 4 is 0.2054716
F-measure of the model 4 is 0.2582378

Accuracy of the model 5 is 0.682
Precision of the model 5 is 0.22
Recall of the model 5 is 0.2054598
F-measure of the model 5 is 0.2124814
```

Chcete si to skúsiť sami?

- Zadanie a odkazy k datasetom
  - https://ufal.mff.cuni.cz/~hladka/2022/docs/bh-ehw3.pdf

- Môj kód
  - https://github.com/GenciJakub/MLinR

Ďakujem za pozornosť



- faces(train_test[c(3,6,7,8,10)])