Nama : Moch. Izza Zulfikar S.

Nim : 221810422

No Absen : 19

Dosen : Ibnu Santoso, SST., MT.

Mata Kuliah : Data Mining dan Knowledge Management

Hari/tanggal : Selasa/27 Oktober 2020

“Saya menyatakan bahwa ujian ini saya kerjakan dengan jujur sesuai kemampuan sendiri dan tidak mengutip sebagian atau seluruh pekerjaan orang lain. Jika suatu saat ditemukan saya melanggar ketentuan ujian, saya siap menerima konsekuensi yang berlaku.”

Moch. Izza Zulfikar S

UTS DMKM

Moch Izza Z S (221810422)

27/10/2020

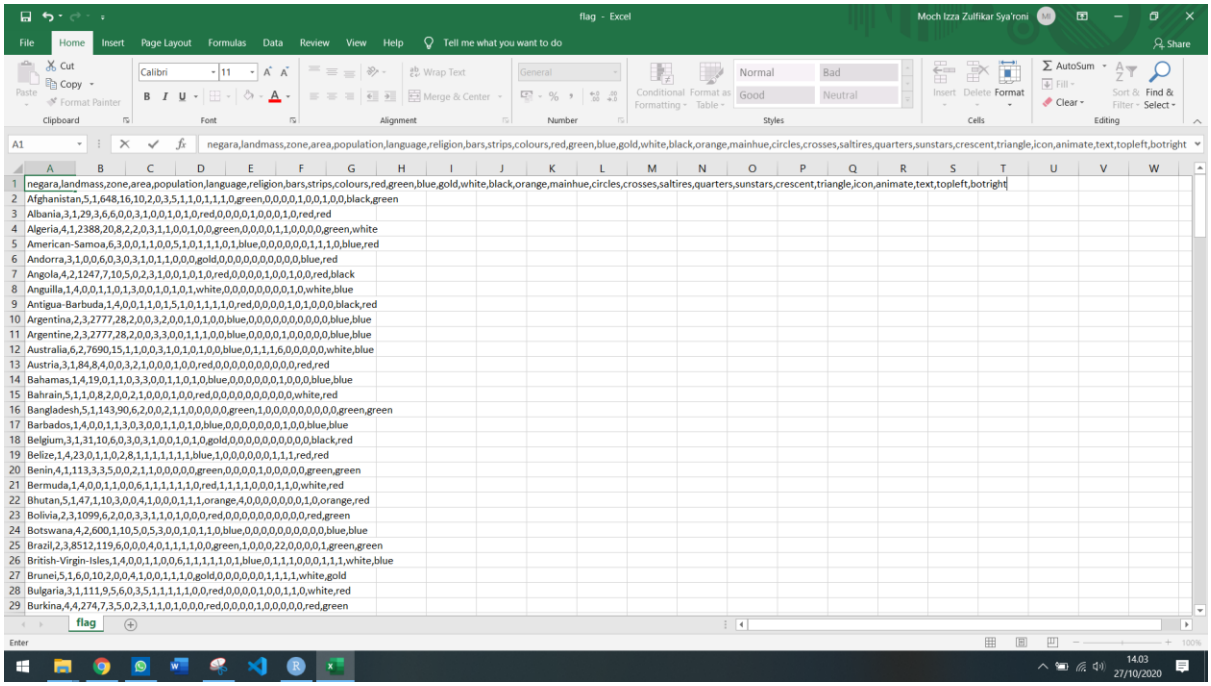
“party” akan digunakan untuk membuat visualisasi dari decision tree.

“psych” akan digunakan untuk melihat korelasi antar variabel.

“caret” digunakan untuk membuat confusion matriks dan melihat akurasi model.

```
library(caret)
library(party)
library(psych)
```

Data diunduh dan kemudian diberikan nama variabel supaya bisa dilihat dengan rapi



Pemberian nama variabel pada data *flag.data* kemudian di save ke dalam bentuk *flag.csv*

Nama variabel diambil dari Data Set Description

Definisi data berupa :

1. name: Name of the country concerned
2. landmass: 1=N.America, 2=S.America, 3=Europe, 4=Africa, 4=Asia, 6=Oceania
3. zone: Geographic quadrant, based on Greenwich and the Equator; 1=NE, 2=SE, 3=SW, 4=NW
4. area: in thousands of square km

5. population: in round millions
6. language: 1=English, 2=Spanish, 3=French, 4=German, 5=Slavic, 6=Other Indo-European, 7=Chinese, 8=Arabic, 9=Japanese/Turkish/Finnish/Magyar, 10=Others
7. religion: 0=Catholic, 1=Other Christian, 2=Muslim, 3=Buddhist, 4=Hindu, 5=Ethnic, 6=Marxist, 7=Others
8. bars: Number of vertical bars in the flag
9. stripes: Number of horizontal stripes in the flag
10. colours: Number of different colours in the flag
11. red: 0 if red absent, 1 if red present in the flag
12. green: same for green
13. blue: same for blue
14. gold: same for gold (also yellow)
15. white: same for white
16. black: same for black
17. orange: same for orange (also brown)
18. mainhue: predominant colour in the flag (tie-breaks decided by taking the topmost hue, if that fails then the most central hue, and if that fails the leftmost hue)
19. circles: Number of circles in the flag
20. crosses: Number of (upright) crosses
21. saltires: Number of diagonal crosses
22. quarters: Number of quartered sections
23. sunstars: Number of sun or star symbols
24. crescent: 1 if a crescent moon symbol present, else 0
25. triangle: 1 if any triangles present, 0 otherwise
26. icon: 1 if an inanimate image present (e.g., a boat), otherwise 0
27. animate: 1 if an animate image (e.g., an eagle, a tree, a human hand) present, 0 otherwise
28. text: 1 if any letters or writing on the flag (e.g., a motto or slogan), 0 otherwise
29. topleft: colour in the top-left corner (moving right to decide tie-breaks)
30. botright: Colour in the bottom-left corner (moving left to decide tie-breaks)

```
dataset <- read.csv("E:/UTS/DMKM/DATA/flag.csv", sep = ",")
head(dataset)
```

```
##           negara landmass zone area population language religion bars s
trips
## 1    Afghanistan        5    1  648         16       10        2    0
3
## 2        Albania        3    1   29          3        6        6    0
0
## 3        Algeria        4    1 2388         20        8        2    2
0
## 4 American-Samoa        6    3    0          0        1        1    0
0
## 5        Andorra        3    1    0          0        6        0    3
0
## 6         Angola        4    2 1247          7       10        5    0
2
##   colours red green blue gold white black orange mainhue circles crosse
s
## 1       5   1     1    0    1     1     1      0   green       0
0
## 2       3   1     0    0    1     0     1      0     red       0
0
## 3       3   1     1    0    0     1     0      0   green       0
0
## 4       5   1     0    1    1     1     0      1    blue       0
0
## 5       3   1     0    1    1     0     0      0    gold       0
0
## 6       3   1     0    0    1     0     1      0     red       0
0
##   saltires quarters sunstars crescent triangle icon animate text toplef
t
## 1        0        0        1        0        0    1       0    0   blac
k
## 2        0        0        1        0        0    0       1    0     re
d
## 3        0        0        1        1        0    0       0    0   gree
n
## 4        0        0        0        0        1    1       1    0    blu
e
## 5        0        0        0        0        0    0       0    0    blu
e
## 6        0        0        1        0        0    1       0    0     re
d
##   botright
## 1    green
## 2      red
## 3    white
## 4      red
## 5      red
## 6    black
```

Data terlihat bagus,

Kemudian masuk ke dalam pengonversian data menjadi bentuk *factor*

```r
for(i in names(dataset)){
  dataset[,i]= as.factor(dataset[,i])
}
#untuk memastikan, dilihat struktur datanya sudah benar atau belum
str(dataset)
```

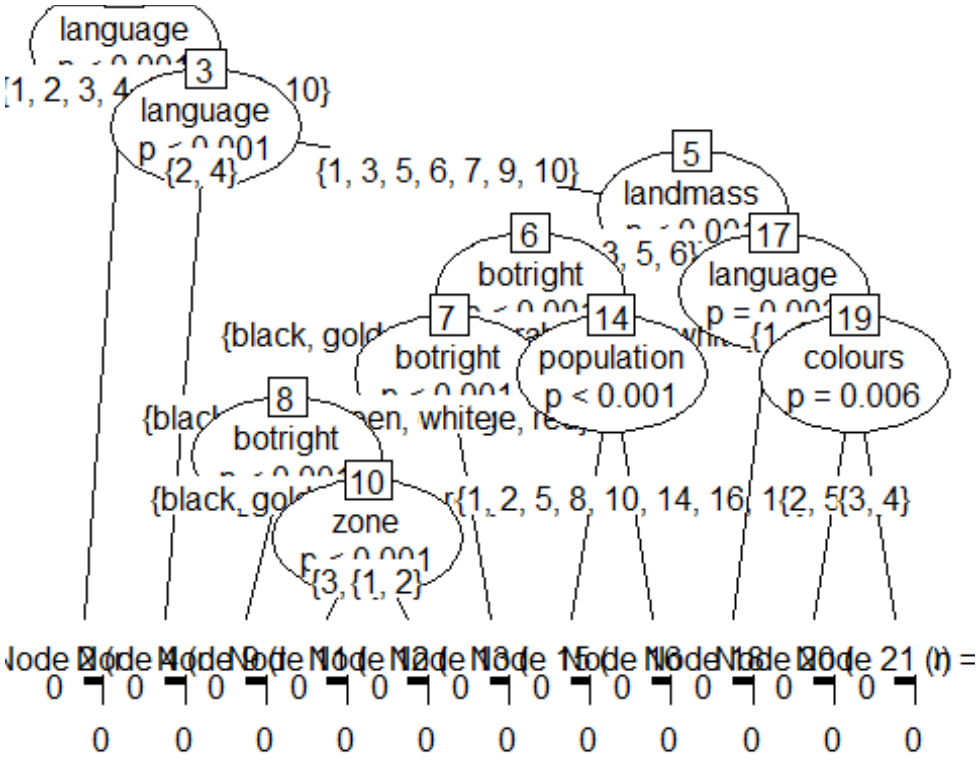
```
## 'data.frame':    194 obs. of  30 variables:
##  $ negara    : Factor w/ 194 levels "Afghanistan",..: 1 2 3 4 5 6 7 8 9
10 ...
##  $ landmass  : Factor w/ 6 levels "1","2","3","4",..: 5 3 4 6 3 4 1 1 2
2 ...
##  $ zone      : Factor w/ 4 levels "1","2","3","4": 1 1 1 3 1 2 4 4 3 3
...
##  $ area      : Factor w/ 136 levels "0","1","2","3",..: 98 21 127 1 1 1
16 1 1 129 129 ...
##  $ population: Factor w/ 48 levels "0","1","2","3",..: 17 4 20 1 1 8 1
1 23 23 ...
##  $ language  : Factor w/ 10 levels "1","2","3","4",..: 10 6 8 1 6 10 1
1 2 2 ...
##  $ religion  : Factor w/ 8 levels "0","1","2","3",..: 3 7 3 2 1 6 2 2 1
1 ...
##  $ bars      : Factor w/ 5 levels "0","1","2","3",..: 1 1 3 1 4 1 1 1 1
1 ...
##  $ strips    : Factor w/ 12 levels "0","1","2","3",..: 4 1 1 1 1 3 2 2
4 4 ...
##  $ colours   : Factor w/ 8 levels "1","2","3","4",..: 5 3 3 5 3 3 3 5 2
3 ...
##  $ red       : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 1 2 1 1 ...
##  $ green     : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 1 1 1 ...
##  $ blue      : Factor w/ 2 levels "0","1": 1 1 1 2 2 1 2 2 2 2 ...
##  $ gold      : Factor w/ 2 levels "0","1": 2 2 1 2 2 2 1 2 1 2 ...
##  $ white     : Factor w/ 2 levels "0","1": 2 1 2 2 1 1 2 2 2 2 ...
##  $ black     : Factor w/ 2 levels "0","1": 2 2 1 1 1 2 1 2 1 1 ...
##  $ orange    : Factor w/ 2 levels "0","1": 1 1 1 2 1 1 2 1 1 1 ...
##  $ mainhue   : Factor w/ 8 levels "black","blue",..: 5 7 5 2 4 7 8 7 2
2 ...
##  $ circles   : Factor w/ 4 levels "0","1","2","4": 1 1 1 1 1 1 1 1 1 1
...
##  $ crosses   : Factor w/ 3 levels "0","1","2": 1 1 1 1 1 1 1 1 1 1 ...
##  $ saltires  : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ quarters  : Factor w/ 3 levels "0","1","4": 1 1 1 1 1 1 1 1 1 1 ...
##  $ sunstars  : Factor w/ 14 levels "0","1","2","3",..: 2 2 2 1 1 2 1 2
1 2 ...
##  $ crescent  : Factor w/ 2 levels "0","1": 1 1 2 1 1 1 1 1 1 1 ...
##  $ triangle  : Factor w/ 2 levels "0","1": 1 1 1 2 1 1 1 2 1 1 ...
##  $ icon      : Factor w/ 2 levels "0","1": 2 1 1 2 1 2 1 1 1 1 ...
##  $ animate   : Factor w/ 2 levels "0","1": 1 2 1 2 1 1 2 1 1 1 ...
##  $ text      : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ topleft   : Factor w/ 7 levels "black","blue",..: 1 6 4 2 2 6 7 1 2
2 ...
##  $ botright  : Factor w/ 8 levels "black","blue",..: 5 7 8 7 7 1 2 7 2
2 ...
```

Lalu memecah data training 80% dan data test 20%

```
set.seed(123)
sampel <- sample(2,nrow(dataset),replace = T, prob = c(0.8,0.2))
training <- dataset[sampel==1, ]
testing <- dataset[sampel==2, ]

#melihat distribusi data
print(paste("Jumlah training data :", nrow(training)))

## [1] "Jumlah training data : 158"

print(paste("Jumlah testing data :", nrow(testing)))

## [1] "Jumlah testing data : 36"
```

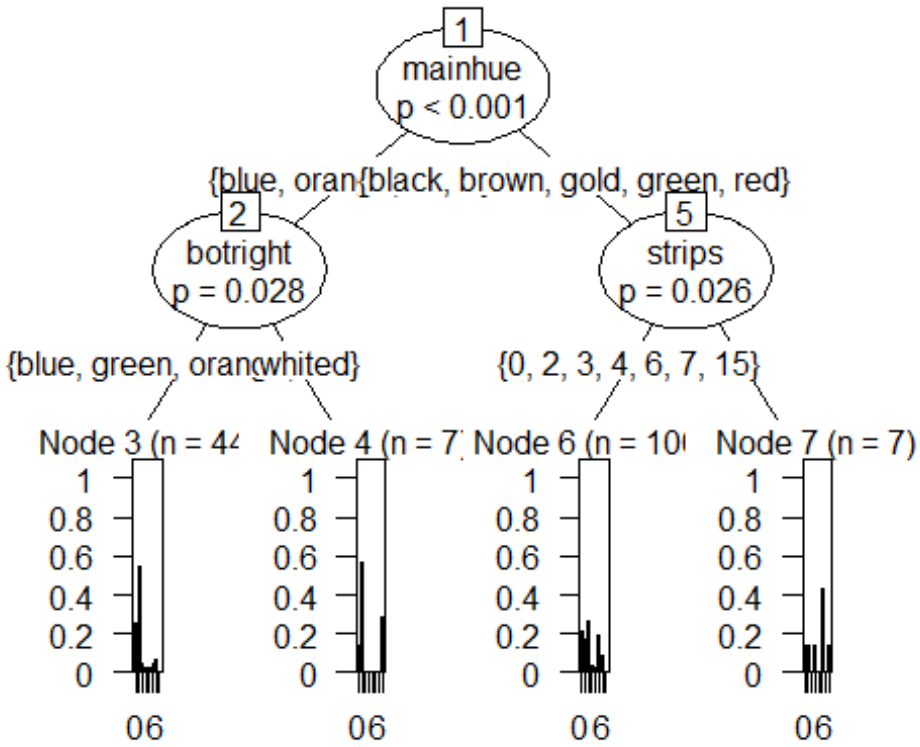
Langkah selanjutnya membuat model dengan memprediksi religion nya berdasarkan seluruh data

```
dtree <- ctree(religion~., data=training)
plot(dtree)
```



Karena soal yang diminta adalah Klasifikasi/Prediksi agama dari bendera negara, maka variabel yang tidak dibutuhkan, tidak perlu diikutkan

```
truetree <- ctree(religion~bars+strips+colours+red+green+blue+gold+white+b
lack+orange+mainhue+circles+crosses+saltires+quarters+sunstars+crescent+tr
iangle+icon+animate+text+topleft+botright, data=training)
plot(truetree)
```



Hal terakhir yang diperlukan adalah membuat Confusion Matrix

```
prediksi <- predict(truetree, testing)
confusionMatrix(table(prediksi,testing$religion))
```

```
## Confusion Matrix and Statistics
##
##
## prediksi 0 1 2 3 4 5 6 7
##        0 0 0 0 0 0 0 0 0
##        1 3 6 1 2 0 1 1 1
##        2 3 8 6 0 0 2 2 0
##        3 0 0 0 0 0 0 0 0
##        4 0 0 0 0 0 0 0 0
##        5 0 0 0 0 0 0 0 0
##        6 0 0 0 0 0 0 0 0
##        7 0 0 0 0 0 0 0 0
##
## Overall Statistics
##
##                Accuracy : 0.3333
##                  95% CI : (0.1856, 0.5097)
##     No Information Rate : 0.3889
##     P-Value [Acc > NIR] : 0.8025
##
##                   Kappa : 0.0799
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
```

```
##                      Class: 0 Class: 1 Class: 2 Class: 3 Class: 4 Class
: 5
## Sensitivity            0.0000   0.4286   0.8571  0.00000       NA  0.00
000
## Specificity            1.0000   0.5909   0.4828  1.00000        1  1.00
000
## Pos Pred Value            NaN   0.4000   0.2857      NaN       NA
NaN
## Neg Pred Value         0.8333   0.6190   0.9333  0.94444       NA  0.91
667
## Prevalence             0.1667   0.3889   0.1944  0.05556        0  0.08
333
## Detection Rate         0.0000   0.1667   0.1667  0.00000        0  0.00
000
## Detection Prevalence   0.0000   0.4167   0.5833  0.00000        0  0.00
000
## Balanced Accuracy      0.5000   0.5097   0.6700  0.50000       NA  0.50
000
##                      Class: 6 Class: 7
## Sensitivity           0.00000  0.00000
## Specificity           1.00000  1.00000
## Pos Pred Value            NaN      NaN
## Neg Pred Value        0.91667  0.97222
## Prevalence            0.08333  0.02778
## Detection Rate        0.00000  0.00000
## Detection Prevalence  0.00000  0.00000
## Balanced Accuracy     0.50000  0.50000
```

Intepretasi :