# Opgave 3 Analytics 2019

Naam student: *graag hier invullen*

Data voor deze opgave: **telcotrain.csv** en **telcotest.csv**. Documentatie van de data: **telco.docx**

Je mag voor het maken van confusion matrices de functie die je in de vorige opgave **(bij opgave 2 vraag 1c)** zelf hebt gemaakt, gebruiken om gemakkelijker confusion matrices en dergelijke te berekenen. (Gebruik source("bestand.r")om een script te kunnen laden).

## Inleveren

Inleveren UITSLUITEND via Blackboard. Uiterste datum van inleveren: 17-3-2019.

Inleveren UITSLUITEND in formaat .PDF, .DOC of .DOCX.

Als je met RStudio werkt kan je de .Rmd file als basis gebruiken. Anders mag je de resultaten in dit document opnemen. Vergeet niet je naam in het document te zetten.

## Inleiding

In Opgave 2 heb je deze data ook gebruikt, voor het maken van een logistic regressie-model. Zie het document “Opgave 2 Analytics 2015” voor de details.

Deze opgave ging over een belangrijke vraag uit de Telecomindustrie: hoe houd ik mijn klanten?

In Opgave 3 ga je daarmee verder. Het gemaakte logistic regressiemodel wordt vergeleken met een CART-tree model. Ook ga je Crossvalidation toepassen om de optimale boom te bepalen.

De vraag is dus: hoe groot is de kans op Churn (vertrek) en welke variabelen spelen daarbij wel en niet een rol. Dit gaan we nu doen met de methode “CART trees”.

## Vraag 1

De opgave begint met een herhaling van Opgave 2. (Zie vraag 1 van Opgave 2)   
Maak opnieuw het **optimale** logistic regressie-model zoals eerder bepaald. Dat wil zeggen, met alleen nog significante variabelen.

**1a)** Laad de data (**telcotrain.csv**), verwijder de eerste en derde kolom en bouw het eerder (bij Opgave 2 (vraag 1b)) bepaalde **optimale** logistic regression model. Geef de summary van het model.

**Gevraagde antwoorden:**

Geef alle gebruikte R-commando’s. Geef ook de R-uitvoer.

> setwd("D:/GitHub/School/P2.3/Analytics/opgaven\_data")

> data=read.csv("telcotrain.csv")

> data$X <- NULL

> data$ZIP <- NULL

> model=glm(Churn~IP+VMP+NVM+TDCH+TECH+TNCH+TIC+TICH+CSC, data=data, family="binomial")

> summary(model)

Call:

glm(formula = Churn ~ IP + VMP + NVM + TDCH + TECH + TNCH + TIC +

TICH + CSC, family = "binomial", data = data)

Deviance Residuals:

Min 1Q Median 3Q Max

-2.0729 -0.5127 -0.3380 -0.1951 3.2767

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -8.32520 0.60280 -13.811 < 2e-16 \*\*\*

IPYes 1.99572 0.16883 11.821 < 2e-16 \*\*\*

VMPYes -2.29403 0.65401 -3.508 0.000452 \*\*\*

NVM 0.04633 0.02010 2.305 0.021149 \*

TDCH 0.07904 0.00739 10.695 < 2e-16 \*\*\*

TECH 0.08094 0.01571 5.153 2.56e-07 \*\*\*

TNCH 0.10317 0.02846 3.625 0.000289 \*\*\*

TIC -0.08350 0.02918 -2.861 0.004221 \*\*

TICH 0.32791 0.08663 3.785 0.000154 \*\*\*

CSC 0.51387 0.04493 11.438 < 2e-16 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 2067.9 on 2499 degrees of freedom

Residual deviance: 1617.8 on 2490 degrees of freedom

AIC: 1637.8

Number of Fisher Scoring iterations: 6

**1b)** Gebruikdit model om de waarden van Churn in de **testset** **(telcotest.csv)** te voorspellen. Bij Opgave 2 (vraag 2) heb je ook de **optimale drempelwaarde** voor dit model bepaald.Geef voor *die* drempelwaarde deConfusion Matrix, de accuracy, de specificity en de sensitivity.

**Gevraagde antwoorden:**

Geef alle gebruikte R-commando’s. Geef ook de R-uitvoer. Vermeld de gebruikte drempelwaarde.

> source('D:/GitHub/School/P2.3/Analytics/opgaven\_data/conf.R')

> test=read.csv("telcotest.csv")

> test$X <- NULL

> test$ZIP <- NULL

> conf(test$Churn, 0.6)

n FALSE TRUE

NO 702 10

YES 113 8

[1] "Sensitivity = 0.444444444444444"

[1] "Specificify = 0.861349693251534"

[1] "Accuracy = 0.85234093637455"

**1c)** Maak ook een ROC-curve van het model en bepaal de Area Under Curve (AUC). Voor ROC-curves heb je de package ROCR nodig.

**Gevraagde antwoorden:**

Geef alle gebruikte R-commando’s. Geef ook de R-uitvoer, inclusief de plot en de AUC-waarde.

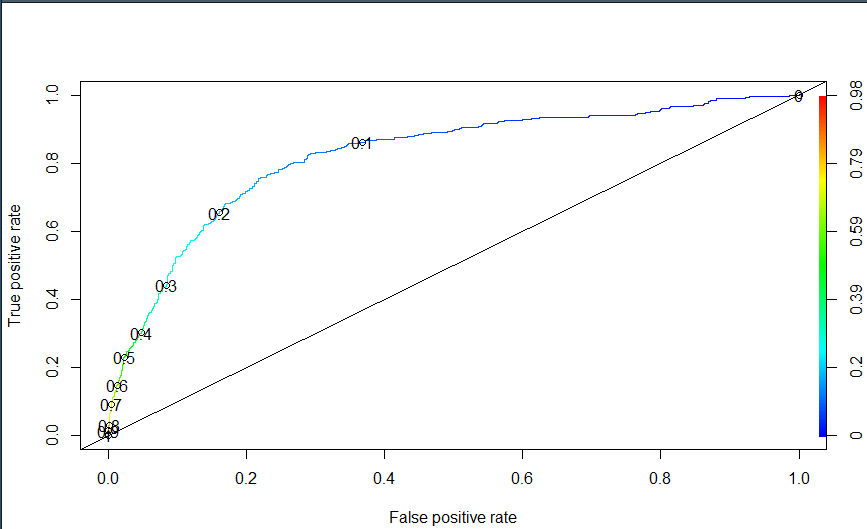
> prediction <- predict(model,type="response")

> ROCRpred = prediction(prediction,data$Churn)

> ROCRperf=performance(ROCRpred,"tpr","fpr")

> plot(ROCRperf,colorize=TRUE,print.cutoffs.at=seq(0,1,0.1))

> abline(0,1)



> auc=as.numeric(performance(ROCRpred, "auc")@y.values)

> auc

[1] 0.8216539

## Vraag 2

Maak nu een CART tree model voor dezelfde **telcotrain** data. Voor CART-trees heb je de packages rpart en rpart.plot nodig. De methodiek is uitgelegd op het college en wordt behandeld in video 4 van de serie Judge Jury and Classifier.

NB: Ook bij deze opgave moet je eerst de eerste en derde kolom uit de data verwijderen.

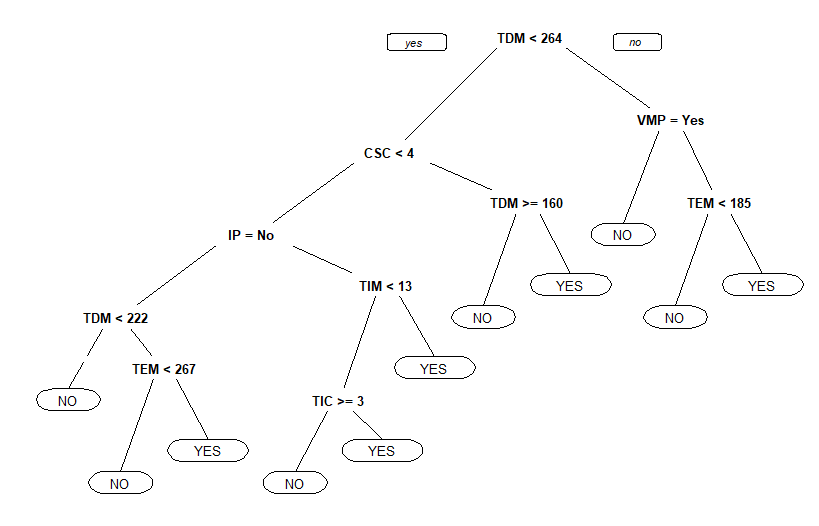
**2a)** Maak op basis van de trainingsdata een CART-tree voor de kans op Churn (afhankelijke variabele) als functie van de overige (onafhankelijke) variabelen. Gebruik een minbucket van 25. Maak een plot van de tree.

**Gevraagde antwoorden:**

Geef alle gebruikte R-commando’s. Geef ook de R-uitvoer en de plot.

> tree=rpart(Churn~.,data=data,method="class", control=rpart.control(minbucket = 25))

> prp(tree)



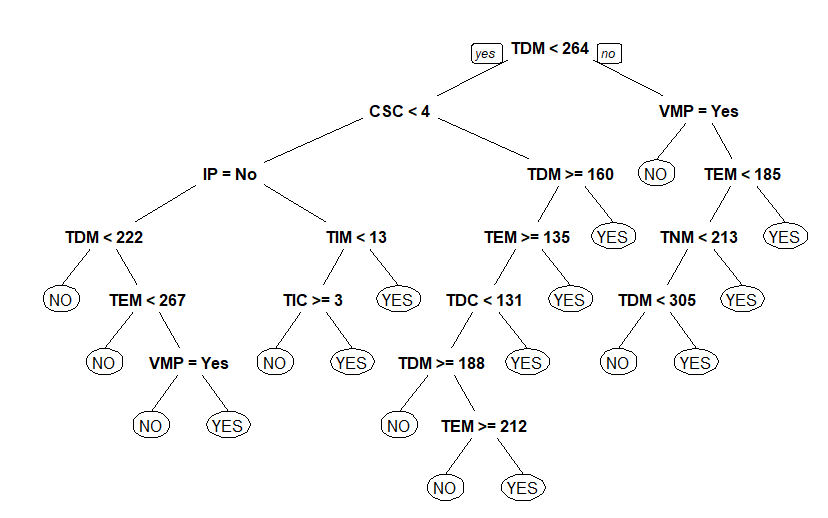
**2b)** Bereken ook de tree voor een minbucket van 5 en een voor een minbucket van 100. Druk ook daarvan de tree af.

**Gevraagde antwoorden:**

Geef alle gebruikte R-commando’s. Geef ook de R-uitvoer en de plots.

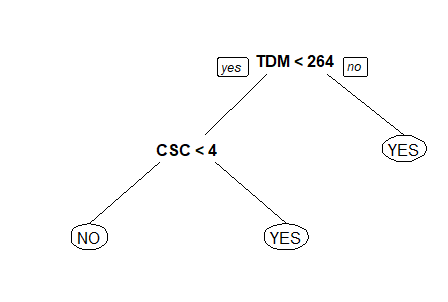
> tree=rpart(Churn~.,data=data,method="class", control=rpart.control(minbucket = 5))

> prp(tree)



> tree=rpart(Churn~.,data=data,method="class", control=rpart.control(minbucket = 100))

> prp(tree)



**2c)** Wat is het effect van de verschillende minbucket parameters?

**Gevraagde antwoorden:**

Beantwoord de vraag.

Hoe lager je de minbucket maakt hoe nauwkeuriger het model.

## Vraag 3

Valideer het CART-model dat bij Vraag 2a) gemaakt is. Gebruik daarvoor dit model om de waarden van Churn in de **testset** **(telcotest.csv)** te voorspellen.

NB: Ook bij deze opgave moet je eerst de eerste en derde kolom uit de data verwijderen.

**3a)** Geef de Confusion Matrix, de accuracy, de specificity en de sensitivity.

**Gevraagde antwoorden:**

Geef alle gebruikte R-commando’s. Geef ook de R-uitvoer.

> conf(test$Churn, 0.6)

n FALSE TRUE

NO 698 14

YES 30 91

[1] "Sensitivity = 0.866666666666667"

[1] "Specificify = 0.958791208791209"

[1] "Accuracy = 0.947178871548619"

**3b)** Maak ook een ROC-curve van het model en bepaal de Area Under Curve (AUC). Voor ROC-curves heb je de package ROCR nodig.

**Gevraagde antwoorden:**

Geef alle gebruikte R-commando’s. Geef ook de R-uitvoer en de plots.

> predictTreeROC=predict(tree,newdata=test)

> treeROC=prediction(predictTreeROC[,2], test$Churn)

> treePerf=performance(treeROC,"tpr","fpr")

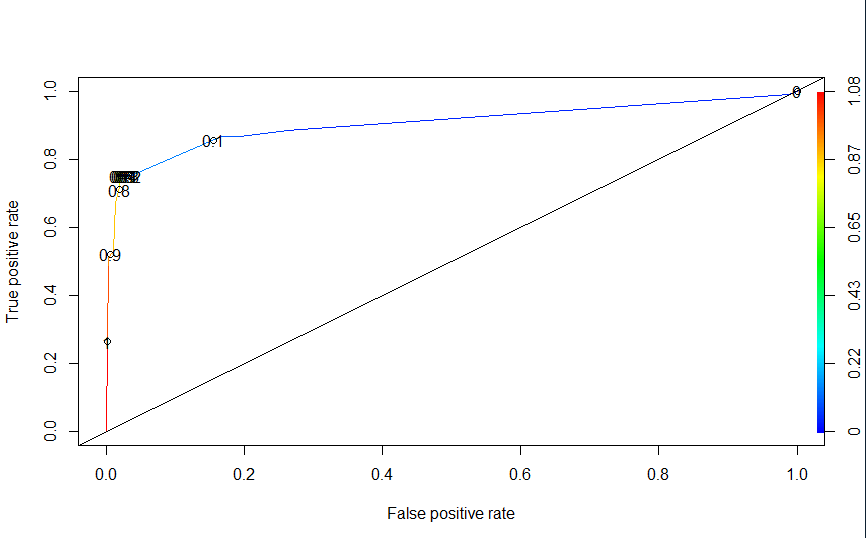
> plot(treePerf, colorize=TRUE,print.cutoffs.at=seq(0,1,0.1))

> abline(0,1)

> AUC=as.numeric(performance(treeROC, "auc")@y.values)

> AUC

[1] 0.9052256



**3c)** Vergelijk de resultaten van dit CART-tree model met het logistic regression model uit vraag 1. Welke is het beste en waarom?

**Gevraagde antwoorden:**

Beantwoord de vraag.

Uit de resutlaten blijkt dat de cart tree veel accurater resultaten geeft. Dit omdat de beste drempelwaarde hoger en rechter ligt

## Vraag 4

Gebruik Cross-Validation om de optimale CART-tree voor deze data te kunnen bepalen. Hiervoor heb je de packages caret en e1071 nodig. De methodiek is uitgelegd op het college en wordt behandeld in video 6 van de serie Judge Jury and Classifier.

**4a)** Bepaal door middel van Cross-Validation de optimale waarde voor de complexity-parameter (cp).

**Gevraagde antwoorden:**

Geef alle gebruikte R-commando’s. Geef ook de R-uitvoer.

> fitControl=trainControl(method="cv", number = 10)

> cartGrid=expand.grid(.cp=(1:50)\*0.01)

> cv=train(Churn~.,data=data,method="rpart", trControl=fitControl,tuneGrid=cartGrid)

> cv

CART

2500 samples

17 predictor

2 classes: 'NO', 'YES'

No pre-processing

Resampling: Cross-Validated (10 fold)

Summary of sample sizes: 2250, 2251, 2250, 2250, 2249, 2249, ...

Resampling results across tuning parameters:

cp Accuracy Kappa

0.01 0.9384044 0.72743162

0.02 0.9324044 0.69829383

0.03 0.9248043 0.65556899

0.04 0.9248043 0.65627910

0.05 0.9128139 0.58258258

0.06 0.8932234 0.44157230

0.07 0.8932234 0.44157230

0.08 0.8840217 0.38152711

0.09 0.8572007 0.12423735

0.10 0.8544023 0.03384586

0.11 0.8552023 0.00000000

0.12 0.8552023 0.00000000

0.13 0.8552023 0.00000000

0.14 0.8552023 0.00000000

0.15 0.8552023 0.00000000

0.16 0.8552023 0.00000000

0.17 0.8552023 0.00000000

0.18 0.8552023 0.00000000

0.19 0.8552023 0.00000000

0.20 0.8552023 0.00000000

0.21 0.8552023 0.00000000

0.22 0.8552023 0.00000000

0.23 0.8552023 0.00000000

0.24 0.8552023 0.00000000

0.25 0.8552023 0.00000000

0.26 0.8552023 0.00000000

0.27 0.8552023 0.00000000

0.28 0.8552023 0.00000000

0.29 0.8552023 0.00000000

0.30 0.8552023 0.00000000

0.31 0.8552023 0.00000000

0.32 0.8552023 0.00000000

0.33 0.8552023 0.00000000

0.34 0.8552023 0.00000000

0.35 0.8552023 0.00000000

0.36 0.8552023 0.00000000

0.37 0.8552023 0.00000000

0.38 0.8552023 0.00000000

0.39 0.8552023 0.00000000

0.40 0.8552023 0.00000000

0.41 0.8552023 0.00000000

0.42 0.8552023 0.00000000

0.43 0.8552023 0.00000000

0.44 0.8552023 0.00000000

0.45 0.8552023 0.00000000

0.46 0.8552023 0.00000000

0.47 0.8552023 0.00000000

0.48 0.8552023 0.00000000

0.49 0.8552023 0.00000000

0.50 0.8552023 0.00000000

Accuracy was used to select the optimal model using the largest value.

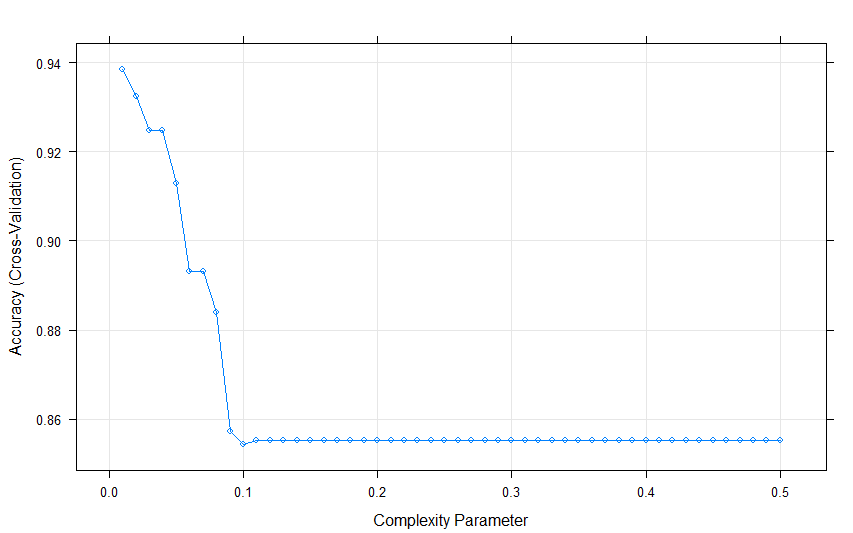
The final value used for the model was cp = 0.01.

**4b)** Maak een plot van de Cross-Validation data. Valt je nog iets op?

**Gevraagde antwoorden:**

Geef alle gebruikte R-commando’s. Geef ook de plot. Beantwoord de vraag.

Op een drietal waardes maakt de plot een raar sprongetje, verder daalt de accuracy vanaf cp=0.1 niet meer.



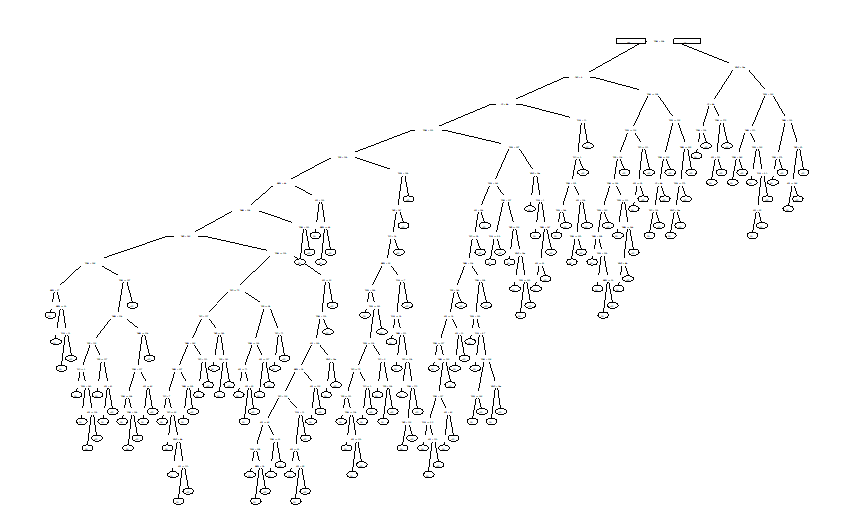
**4c)** Bereken de CART-tree met de berekende cp-waarde. Maak een plot van de tree.

**Gevraagde antwoorden:**

Geef alle gebruikte R-commando’s. Geef ook de R-uitvoer en de plots.

> tree2=rpart(Churn~.,data=data,method="class",control=rpart.control(cp=0,01))

> prp(tree2)



**4d)** Gebruik ook dit model om de waarden van Churn in de **testset** **(telcotest.csv)** te voorspellen. Geef weer de Confusion Matrix, de accuracy, de specificity en de sensitivity.

**Gevraagde antwoorden:**

Geef alle gebruikte R-commando’s. Geef ook de R-uitvoer en de plots.

> conf(test$Churn, 0.6)

n FALSE TRUE

NO 675 37

YES 33 88

[1] "Sensitivity = 0.704"

[1] "Specificify = 0.953389830508475"

[1] "Accuracy = 0.915966386554622"

**4e)** Vergelijk dit ‘optimale’ model met het ‘ standaard’model dat je bij vraag 3 hebt gemaakt. Wat is je conclusie?

**Gevraagde antwoorden:**

Beantwoord de vraag.

De accuracy bij de tweede versie is slightly hoger, dit is een goed ding.