```
library(dplyr)

rladies_global %>%
  filter(city == 'Bucharest')
```

Introducere în Machine Learning cu Caret

Alexandra Conda

Bună!

Eu sunt o persoană entuziasmată de conceptul de Machine Learning!

Sunt aici pentru că îmi place să lucrez în R și vreau să-mi dezvolt abilitatea de a vorbi despre acesta.

Despre mine, sunt ambițioasă, ador să citesc cărți și sunt pasionată de pictură, comunicare, psihologie și filosofie.

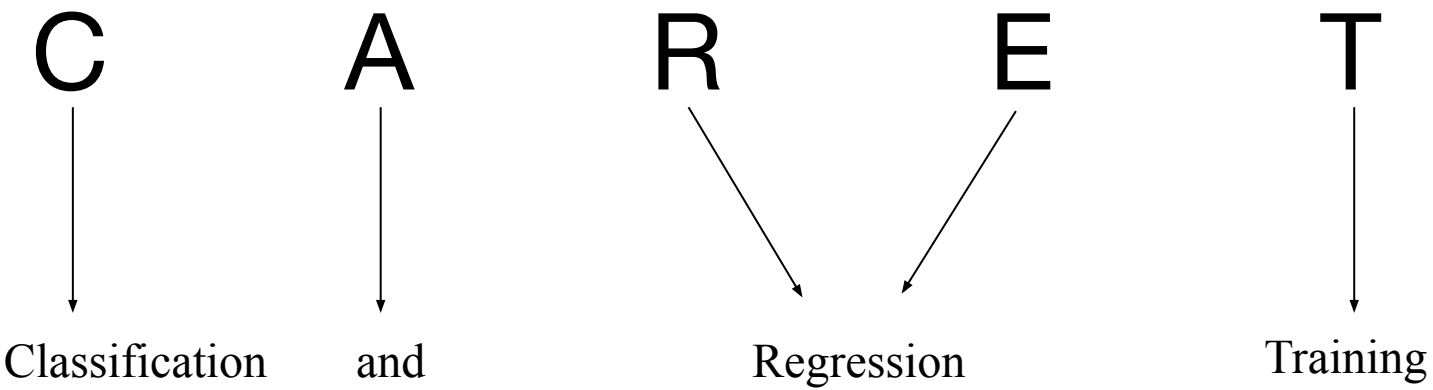
Agenda

- Introducere
- Ce este Caret
- Ce face Caret
- De ce Machine Learning cu Caret
- Pași introductivi
- Definirea obiectivului
- Explorarea datelor
- Feature engineering
- Pregătirea datelor
- Modelare cu Caret
- Concluzii

Obiective

1. În pachetul “Caret” sunt implementați o multitudine de algoritmi, iar aceștia se pot folosi cu ajutorul unei singure funcții.
2. Pentru fiecare pas necesar elaborării unui proiect complet, există în acest pachet o funcție dedicată.
3. Se poate alege cu ușurință cel mai eficient algoritm.

Ce este Caret?



- Utilizat pentru algoritmi de Machine Learning (Învățare Supervizată (Supervised Learning))

Ce face Caret?

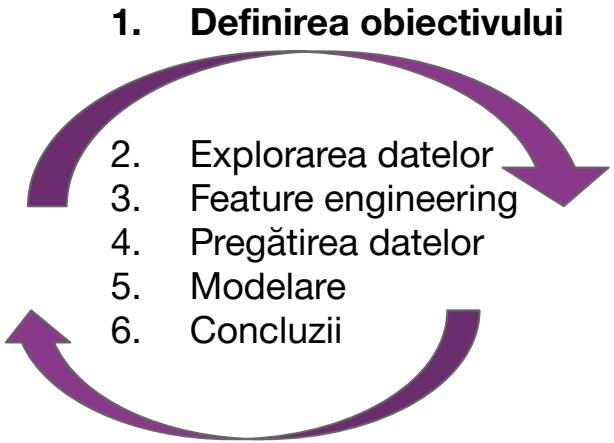
- Eficientizează procesul de creare a modelelor predictive

De ce ML cu Caret?

- Se poate determina modelul optim în cel mai scurt timp posibil

Să începem!

Pași introductivi



Definirea obiectivului pentru ML

- Scopul predicției este de a determina care dintre cele două tipuri de suc de portocale va fi cumpărat de către clienți.
- Identificăm tipul problemei: o problema de clasificare
- Algoritm principal: Multivariate Adaptive Regression Splines (MARS)
- Date numerice și categoriale

Se importă datele

```
Dataset_caret<-read.csv('https://raw.githubusercontent.com/selva86/datasets/master/orange_juice_withmissing.csv')
```

Explorarea datelor

- Descrierea variabilelor

```
'data.frame':   1070 obs. of  18 variables:
 $ Purchase      : Factor w/ 2 levels "CH","MM": 1 1 1 2 1 1 1 1 1 1 ...
 $ WeekofPurchase: int  237 239 245 227 228 230 232 234 235 238 ...
 $ StoreID       : int  1 1 1 1 7 7 7 7 7 7 ...
 $ PriceCH       : num  1.75 1.75 1.86 1.69 1.69 1.69 1.69 1.75 1.75 1.75 ...
 $ PriceMM       : num  1.99 1.99 2.09 1.69 1.69 1.99 1.99 1.99 1.99 1.99 ...
 $ DiscCH        : num  0 0 0.17 0 0 0 0 0 0 0 ...
 $ DiscMM        : num  0 0.3 0 0 0 0 0.4 0.4 0.4 0.4 ...
 $ SpecialCH     : int  0 0 0 0 0 0 1 1 0 0 ...
 $ SpecialMM     : int  0 1 0 0 0 1 1 0 0 0 ...
 $ LoyalCH       : num  0.5 0.6 0.68 0.4 0.957 ...
 $ SalePriceMM   : num  1.99 1.69 2.09 1.69 1.69 1.99 1.59 1.59 1.59 1.59 ...
 $ SalePriceCH   : num  1.75 1.75 1.69 1.69 1.69 1.69 1.69 1.75 1.75 1.75 ...
 $ PriceDiff     : num  0.24 -0.06 0.4 0 0 0.3 -0.1 -0.16 -0.16 -0.16 ...
 $ Store7        : Factor w/ 2 levels "No","Yes": 1 1 1 1 2 2 2 2 2 2 ...
 $ PctDiscMM     : num  0 0.151 0 0 0 ...
 $ PctDiscCH     : num  0 0 0.0914 0 0 ...
 $ ListPriceDiff : num  0.24 0.24 0.23 0 0 0.3 0.3 0.24 0.24 0.24 ...
 $ STORE         : int  1 1 1 1 0 0 0 0 0 0 ...
```

Realizat in R

- Împărțirea datelor (trainingData, testData) -> createDataPartition()

Statistici descriptive cu “skimr” -> skim() -> train data

```
-- Data Summary ------------------------
                           Values
Name                       trainData
Number of rows             857
Number of columns          18
_______________________
Column type frequency:
  factor                   2
  numeric                  16
________________________
Group variables            None

-- Variable type: factor -------------------------------------------------------
  skim_variable n_missing complete_rate ordered n_unique top_counts
1 Purchase              0             1 FALSE          2 CH: 523, MM: 334
2 Store7                0             1 FALSE          2 No: 576, Yes: 281

-- Variable type: numeric ------------------------------------------------------
   skim_variable  n_missing complete_rate     mean      sd         p0    p25     p50     p75    p100 hist
 1 WeekofPurchase         0         1     254.     15.6   227          239     257     268     278
 2 StoreID                1         0.999   3.95    2.29    1            2       3       7       7
 3 PriceCH                1         0.999   1.87    0.102   1.69         1.79    1.86    1.99    2.09
 4 PriceMM                1         0.999   2.08    0.136   1.69         1.99    2.09    2.18    2.29
 5 DiscCH                 2         0.998   0.0541  0.120   0            0       0       0       0.5
 6 DiscMM                 3         0.996   0.121   0.208   0            0       0       0.2     0.8
 7 SpecialCH              2         0.998   0.156   0.363   0            0       0       0       1
 8 SpecialMM              4         0.995   0.150   0.357   0            0       0       0       1
 9 LoyalCH                5         0.994   0.564   0.312   0.000011     0.32    0.595   0.853   1.000
10 SalePriceMM            3         0.996   1.96    0.247   1.19         1.69    2.09    2.13    2.29
11 SalePriceCH            1         0.999   1.81    0.145   1.39         1.75    1.86    1.89    2.09
12 PriceDiff              1         0.999   0.150   0.266  -0.67         0       0.23    0.32    0.64
13 PctDiscMM              2         0.998   0.0581  0.0988  0            0       0       0.113   0.402
14 PctDiscCH              2         0.998   0.0285  0.0635  0            0       0       0       0.253
15 ListPriceDiff          0         1       0.217   0.109   0            0.13    0.24    0.3     0.44
16 STORE                  2         0.998   1.66    1.44    0            0       2       3       4
```

Realizat in R

Feature engineering si pregatirea datelor

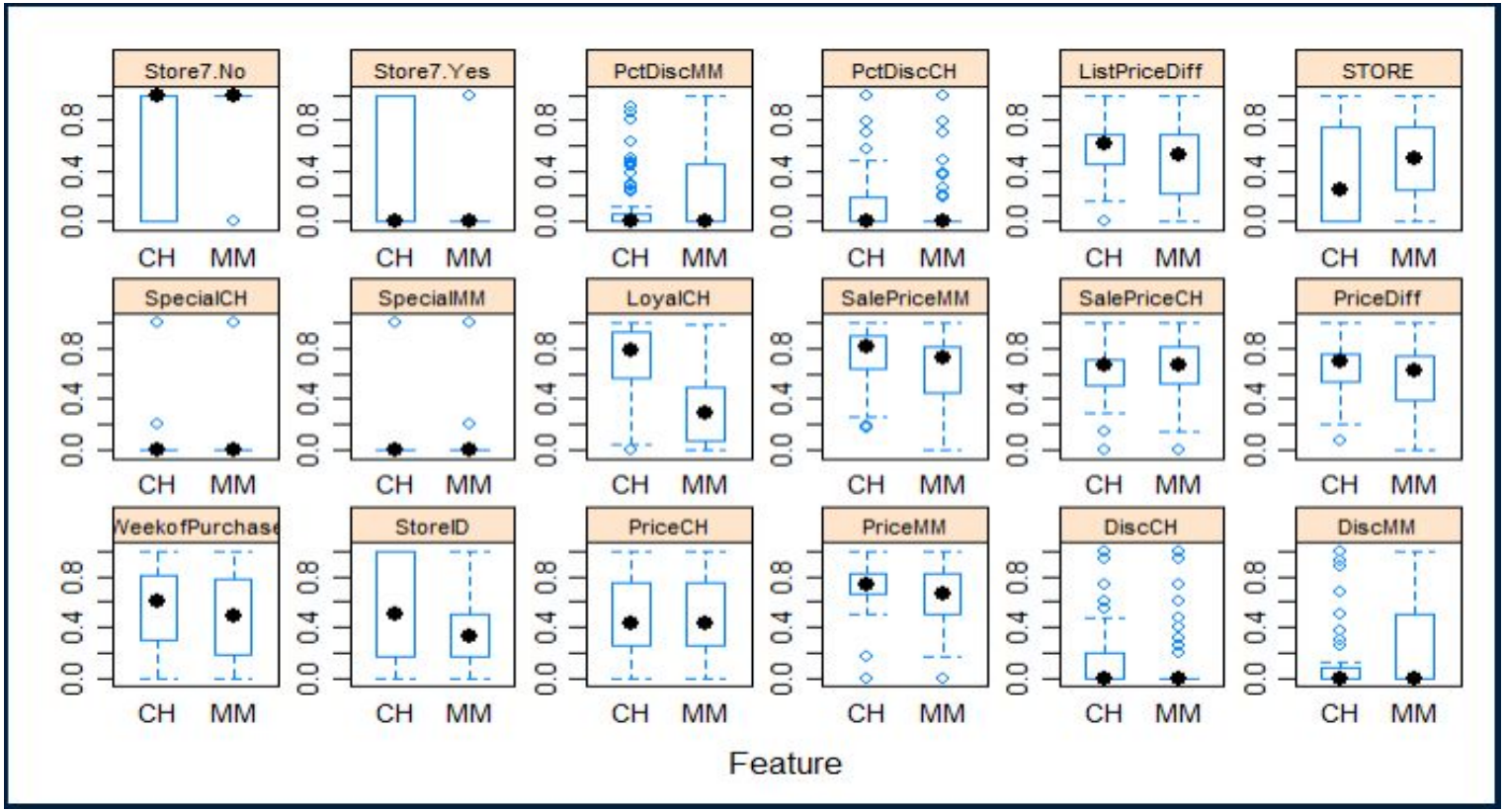
Feature engineering

- Tratare valori lipsă -> preProcess() -> algoritmul K-Nearest-Neighbors
- Creare variabilă dummy -> dummyVars() (pentru variabila Purchase)

Pregatirea datasetului

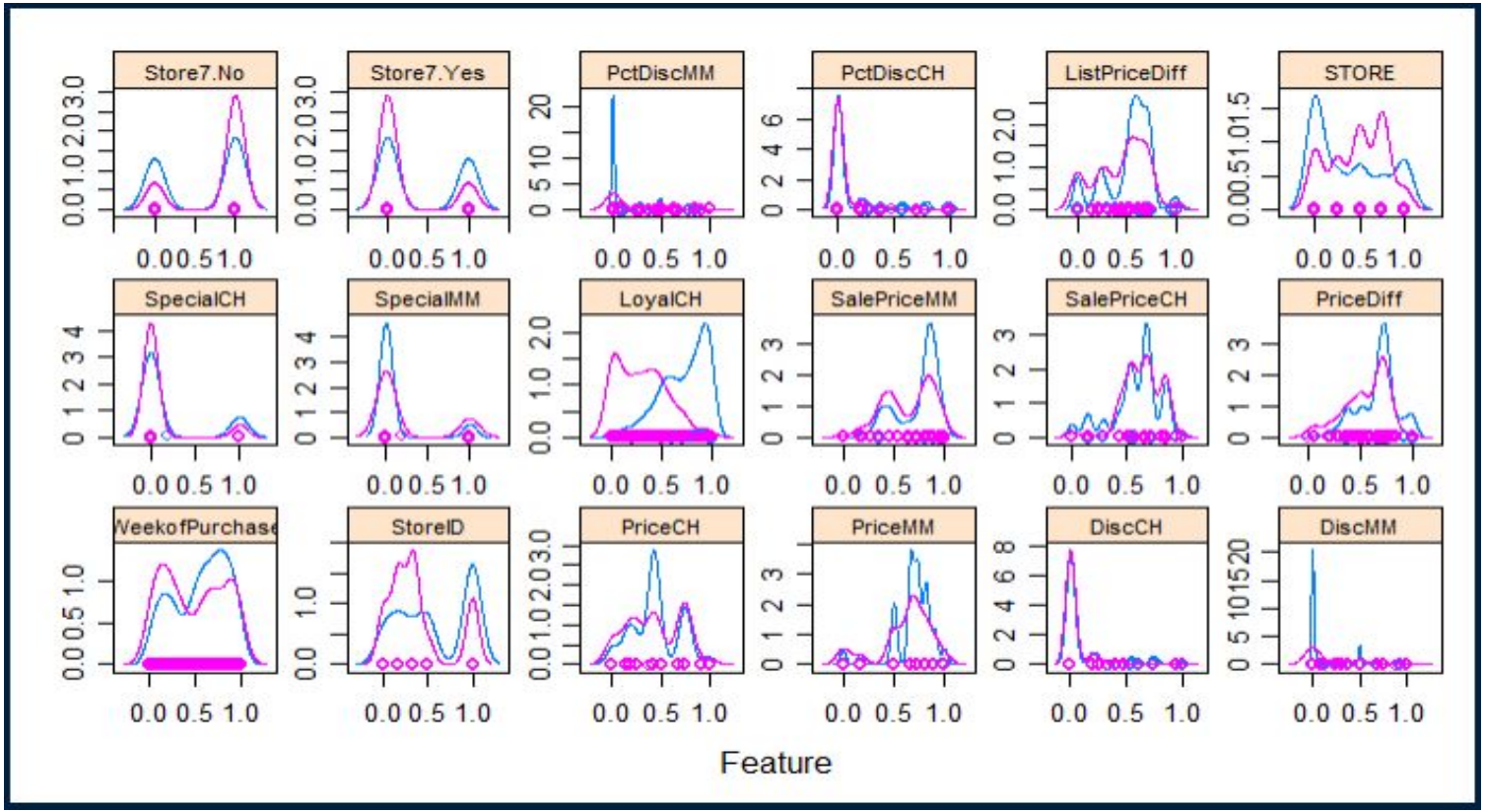
- Normalizare date -> preProcess() ->metoda range

- BoxPlot -> featurePlot()



Realizat in R

- Density Plot -> featurePlot()



Realizat in R

Modelare în Caret

- Selecția caracteristicilor folosind eliminarea recursivă a funcțiilor

rfe() si rfControl()

Variables <S3: AsIs>	Accuracy <S3: AsIs>	Kappa <S3: AsIs>	AccuracySD <S3: AsIs>	KappaSD <S3: AsIs>	Selected <S3: AsIs>
1	0.7440	0.4563	0.04010	0.08519	
2	0.8147	0.6074	0.03647	0.07789	
3	0.8205	0.6189	0.04137	0.08774	*
4	0.8035	0.5852	0.04663	0.09682	
5	0.8024	0.5818	0.04475	0.09458	
10	0.8038	0.5838	0.04380	0.09314	
15	0.8077	0.5896	0.04221	0.09025	
18	0.8054	0.5853	0.03991	0.08606	

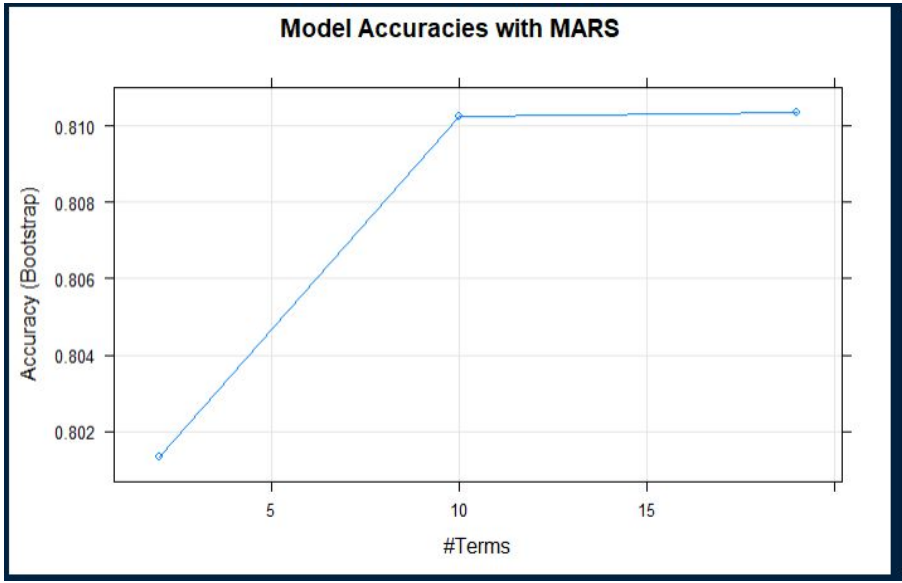
Realizat in R

- Multivariate Adaptive Regression Splines (MARS)

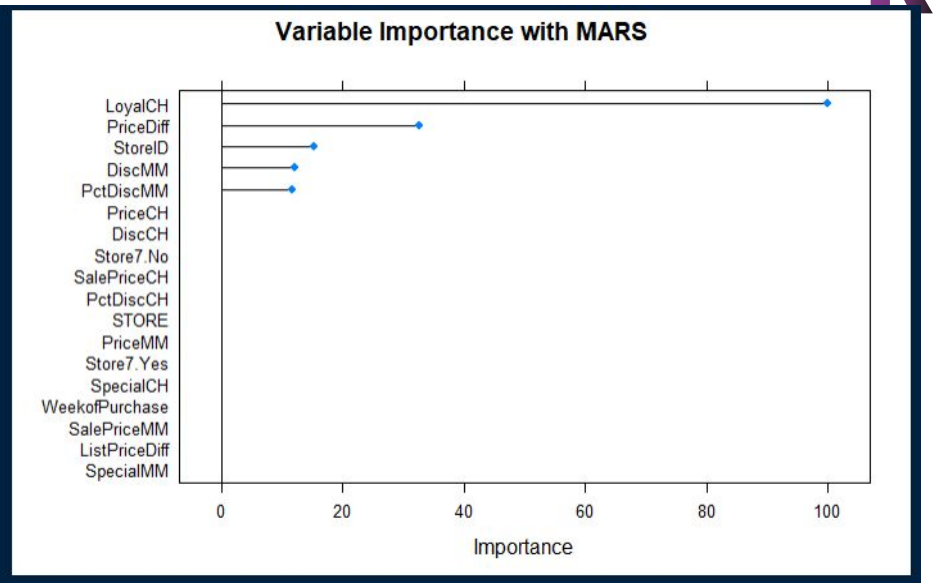
```
857 samples
 18 predictor
  2 classes: 'CH', 'MM'

No pre-processing
Resampling: Bootstrapped (25 reps)
Summary of sample sizes: 857, 857, 857, 857, 857, 857, ...
Resampling results across tuning parameters:

  nprune  Accuracy   Kappa
   2      0.8013184  0.5746285
  10      0.8102610  0.5987447
  19      0.8103685  0.5986923

Tuning parameter 'degree' was held constant at a value of 1
Accuracy was used to select the optimal model using the largest value.
The final values used for the model were nprune = 19 and degree = 1.
```

Realizat in R

plot()



varImp()



Realizat in R

- Realizăm aceeași pași pentru test data - >predict()
- Se face predicția pe test data ->predict()

```
[1] CH CH CH CH CH MM
Levels: CH MM
```

- Matricea de confuzie

```
Confusion Matrix and Statistics

          Reference
Prediction  CH  MM
        CH 113  21
        MM  17  62

               Accuracy : 0.8216
                 95% CI : (0.7635, 0.8705)
    No Information Rate : 0.6103
    P-Value [Acc > NIR] : 2.139e-11

                  Kappa : 0.6216

 Mcnemar's Test P-Value : 0.6265

            Sensitivity : 0.7470
            Specificity : 0.8692
         Pos Pred Value : 0.7848
         Neg Pred Value : 0.8433
              Precision : 0.7848
                 Recall : 0.7470
                     F1 : 0.7654
             Prevalence : 0.3897
         Detection Rate : 0.2911
   Detection Prevalence : 0.3709
      Balanced Accuracy : 0.8081

       'Positive' Class : MM
```

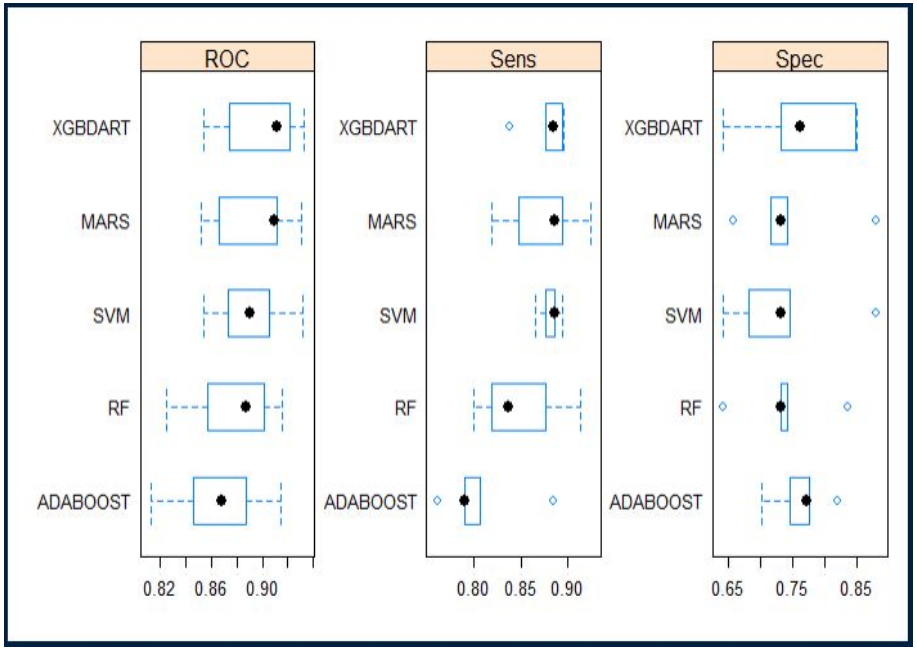
Realizat in R

Evaluarea celui mai bun model

- AdaBoost Classification Trees
- Random Forest
- eXtreme Gradient Boosting
- Support Vector Machines with Radial Basis

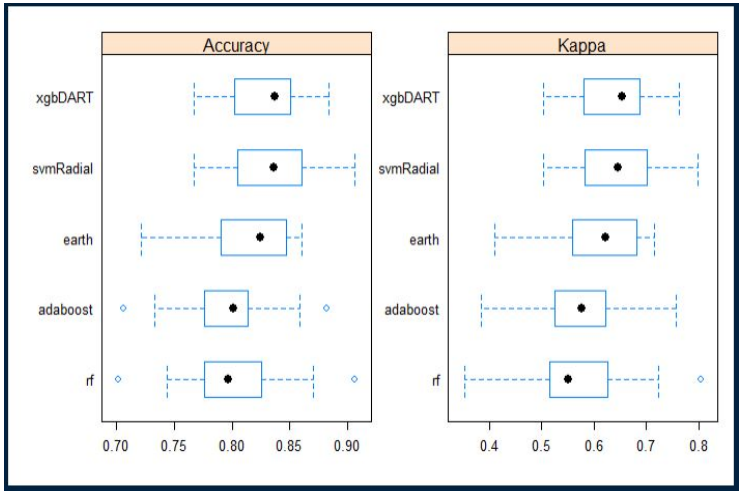
Comparația modelelor se face:

-> resample()



Realizat in R

- Asamblarea predicțiilor -> caretEnsemble()



Realizat in R

```
A glm ensemble of 2 base models: rf, adaboost, earth, xgbDART, svmRadial

Ensemble results:
Generalized Linear Model

2571 samples
   5 predictor
   2 classes: 'CH', 'MM'

No pre-processing
Resampling: Cross-Validated (10 fold, repeated 3 times)
Summary of sample sizes: 2314, 2314, 2314, 2314, 2313, 2313, ...
Resampling results:

  Accuracy   Kappa
  0.8334093  0.6454668
```

Realizat in R

- Combinarea predicțiilor pentru a forma o predicție finală -> caretStack()

Concluzii

- Este un pachet ce simplifică mult sintaxa programului
- Nu necesită cunoștiințe avansate de ML
- Alege modelul optim și creează rapid predicții

MULȚUMESC!