R Notebook

Parametros:

Measure = Accuracy
Columns = sampling, weight_space, underbagging, learner
Performance = tuning_measure
Filter keys = NULL
Filter values = NULL

```r
library("scmamp")
library(dplyr)
```

Tratamento dos dados

Carregando data set compilado

```r
ds = read.csv("/home/rodrigo/Dropbox/UNICAMP/IC/estudo_cost_learning/SummaryResults/summary_compilation_
```

```r
ds = filter(ds, learner != "classif.rusboost")
summary(ds)
```

```
##                  learner      weight_space
##  classif.ksvm       :17100   Mode :logical
##  classif.randomForest:17100  FALSE:41040
##  classif.rusboost    :    0  TRUE :10260
##  classif.xgboost     :17100  NA's :0
##
##
##
##                                measure        sampling       underbagging
##  Accuracy                       :10260   ADASYN:10260   Mode :logical
##  Area under the curve           :10260   FALSE :30780   FALSE:41040
##  F1 measure                     :10260   SMOTE :10260   TRUE :10260
##  G-mean                         :10260                  NA's :0
##  Matthews correlation coefficient:10260
##
##
##  tuning_measure     holdout_measure    holdout_measure_residual
##  Min.   :-0.1277   Min.   :-0.2120   Min.   :-0.4658
##  1st Qu.: 0.6911   1st Qu.: 0.4001   1st Qu.: 0.1994
##  Median : 0.9700   Median : 0.8571   Median : 0.5581
##  Mean   : 0.7903   Mean   : 0.6718   Mean   : 0.5298
##  3rd Qu.: 0.9975   3rd Qu.: 0.9900   3rd Qu.: 0.8755
##  Max.   : 1.0000   Max.   : 1.0000   Max.   : 1.0000
##  NA's   :1077      NA's   :1077      NA's   :1077
##  iteration_count              dataset         imba.rate
##  Min.   :1        abalone         :  900   Min.   :0.0010
##  1st Qu.:1        adult           :  900   1st Qu.:0.0100
##  Median :2        bank            :  900   Median :0.0300
##  Mean   :2        car             :  900   Mean   :0.0286
```

1

```
## 3rd Qu.:3        cardiotocography-10clases:  900   3rd Qu.:0.0500
## Max.   :3        cardiotocography-3clases :  900   Max.   :0.0500
## NA's   :1077     (Other)                   :45900
```

Filtrando pela metrica

```r
ds = filter(ds, measure == params$measure)
```

Filtrando o data set

```r
if(params$filter_keys != 'NULL' && !is.null(params$filter_keys)){
  ds = filter_at(ds, .vars = params$filter_keys, .vars_predicate = any_vars(. == params$filter_values))
}

summary(ds)
```

```
##                   learner       weight_space
##  classif.ksvm        :3420    Mode :logical
##  classif.randomForest:3420    FALSE:8208
##  classif.rusboost    :   0    TRUE :2052
##  classif.xgboost     :3420    NA's :0
##
##
##
##                                   measure          sampling     underbagging
##  Accuracy                        :10260    ADASYN:2052    Mode :logical
##  Area under the curve            :    0    FALSE :6156    FALSE:8208
##  F1 measure                      :    0    SMOTE :2052    TRUE :2052
##  G-mean                          :    0                   NA's :0
##  Matthews correlation coefficient:    0
##
##
##  tuning_measure     holdout_measure    holdout_measure_residual
##  Min.   :0.09041    Min.   :0.01517    Min.   :0.0346
##  1st Qu.:0.96185    1st Qu.:0.95349    1st Qu.:0.3809
##  Median :0.98796    Median :0.98113    Median :0.7239
##  Mean   :0.95509    Mean   :0.94933    Mean   :0.6600
##  3rd Qu.:0.99669    3rd Qu.:0.99347    3rd Qu.:0.9428
##  Max.   :1.00000    Max.   :1.00000    Max.   :1.0000
##  NA's   :204        NA's   :204        NA's   :204
##  iteration_count                     dataset      imba.rate
##  Min.   :1      abalone                 : 180   Min.   :0.0010
##  1st Qu.:1      adult                   : 180   1st Qu.:0.0100
##  Median :2      bank                    : 180   Median :0.0300
##  Mean   :2      car                     : 180   Mean   :0.0286
##  3rd Qu.:3      cardiotocography-10clases: 180   3rd Qu.:0.0500
##  Max.   :3      cardiotocography-3clases : 180   Max.   :0.0500
##  NA's   :204    (Other)                  :9180
```

Computando as médias das iteracoes

```r
ds = group_by(ds, learner , weight_space , measure , sampling , underbagging , dataset , imba.rate)
ds = summarise(ds, tuning_measure = mean(tuning_measure), holdout_measure = mean(holdout_measure),
            holdout_measure_residual = mean(holdout_measure_residual))

ds = as.data.frame(ds)
```

Criando dataframe

```r
# Dividindo o ds em n, um para cada técnica
splited_df = ds %>% group_by_at(.vars = params$columns) %>% do(vals = as.data.frame(.)) %>% select(vals

# Juntando cada uma das partes horizontalmente em um data set
df_tec_wide = do.call("cbind", splited_df)

# Renomeando duplicacao de nomes
colnames(df_tec_wide) = make.unique(colnames(df_tec_wide))

# Selecionando apenas as medidas da performance escolhida
df_tec_wide_residual = select(df_tec_wide, matches(paste("^", params$performance, "$|", params$performan

# Renomeando colunas
new_names = NULL
for(i in (1:length(splited_df))){
  id = toString(sapply(splited_df[[i]][1, params$columns], as.character))
  new_names = c(new_names, id)
}
colnames(df_tec_wide_residual) = new_names

# Verificando a dimensao do df
dim(df_tec_wide_residual)
```

```
## [1] 228  15
```

```r
# Renomeando a variavel
df = df_tec_wide_residual

summary(df)
```

```
##  ADASYN, FALSE, FALSE, classif.ksvm
##  Min.   :0.7408
##  1st Qu.:0.9775
##  Median :0.9947
##  Mean   :0.9798
##  3rd Qu.:0.9988
##  Max.   :1.0000
##  NA's   :7
##  ADASYN, FALSE, FALSE, classif.randomForest
##  Min.   :0.7035
##  1st Qu.:0.9856
##  Median :0.9967
##  Mean   :0.9856
##  3rd Qu.:0.9992
##  Max.   :1.0000
##  NA's   :26
##  ADASYN, FALSE, FALSE, classif.xgboost FALSE, FALSE, FALSE, classif.ksvm
##  Min.   :0.7120                         Min.   :0.9264
##  1st Qu.:0.9849                         1st Qu.:0.9609
##  Median :0.9948                         Median :0.9766
##  Mean   :0.9850                         Mean   :0.9751
##  3rd Qu.:0.9987                         3rd Qu.:0.9901
##  Max.   :1.0000                         Max.   :1.0000
##
```

```
##  FALSE, FALSE, FALSE, classif.randomForest
##  Min.   :0.9442
##  1st Qu.:0.9702
##  Median :0.9897
##  Mean   :0.9821
##  3rd Qu.:0.9939
##  Max.   :1.0000
##  NA's   :6
##  FALSE, FALSE, FALSE, classif.xgboost FALSE, FALSE, TRUE, classif.ksvm
##  Min.   :0.9440                        Min.   :0.2896
##  1st Qu.:0.9716                        1st Qu.:0.7244
##  Median :0.9898                        Median :0.9320
##  Mean   :0.9827                        Mean   :0.8491
##  3rd Qu.:0.9951                        3rd Qu.:0.9836
##  Max.   :1.0000                        Max.   :1.0000
##
##  FALSE, FALSE, TRUE, classif.randomForest
##  Min.   :0.4222
##  1st Qu.:0.7803
##  Median :0.8951
##  Mean   :0.8564
##  3rd Qu.:0.9651
##  Max.   :1.0000
##  NA's   :5
##  FALSE, FALSE, TRUE, classif.xgboost FALSE, TRUE, FALSE, classif.ksvm
##  Min.   :0.3600                       Min.   :0.9346
##  1st Qu.:0.7856                       1st Qu.:0.9611
##  Median :0.8871                       Median :0.9761
##  Mean   :0.8459                       Mean   :0.9752
##  3rd Qu.:0.9508                       3rd Qu.:0.9902
##  Max.   :1.0000                       Max.   :1.0000
##
##  FALSE, TRUE, FALSE, classif.randomForest
##  Min.   :0.9442
##  1st Qu.:0.9700
##  Median :0.9897
##  Mean   :0.9820
##  3rd Qu.:0.9932
##  Max.   :1.0000
##  NA's   :6
##  FALSE, TRUE, FALSE, classif.xgboost SMOTE, FALSE, FALSE, classif.ksvm
##  Min.   :0.9440                       Min.   :0.7394
##  1st Qu.:0.9720                       1st Qu.:0.9780
##  Median :0.9898                       Median :0.9952
##  Mean   :0.9827                       Mean   :0.9795
##  3rd Qu.:0.9948                       3rd Qu.:0.9994
##  Max.   :1.0000                       Max.   :1.0000
##
##  SMOTE, FALSE, FALSE, classif.randomForest
##  Min.   :0.7259
##  1st Qu.:0.9852
##  Median :0.9964
##  Mean   :0.9860
##  3rd Qu.:0.9994
```
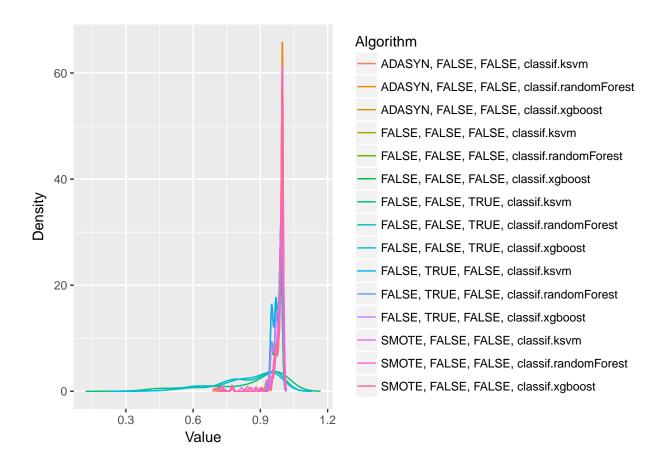
```
## Max.    :1.0000
## NA's    :18
## SMOTE, FALSE, FALSE, classif.xgboost
## Min.    :0.6952
## 1st Qu.:0.9850
## Median :0.9949
## Mean    :0.9851
## 3rd Qu.:0.9987
## Max.    :1.0000
##
```

Verificando a média de cada coluna selecionada

```r
for(i in (1:dim(df)[2])){
  #print(df[,i])
  print(paste("Media da coluna ", colnames(df)[i], " = ", mean(df[,i], na.rm = TRUE), sep=""))
}
```

```
## [1] "Media da coluna ADASYN, FALSE, FALSE, classif.ksvm = 0.979794708481056"
## [1] "Media da coluna ADASYN, FALSE, FALSE, classif.randomForest = 0.985572124397908"
## [1] "Media da coluna ADASYN, FALSE, FALSE, classif.xgboost = 0.984951544919685"
## [1] "Media da coluna FALSE, FALSE, FALSE, classif.ksvm = 0.975133744239946"
## [1] "Media da coluna FALSE, FALSE, FALSE, classif.randomForest = 0.982104793887645"
## [1] "Media da coluna FALSE, FALSE, FALSE, classif.xgboost = 0.982731622083143"
## [1] "Media da coluna FALSE, FALSE, TRUE, classif.ksvm = 0.84906001351275"
## [1] "Media da coluna FALSE, FALSE, TRUE, classif.randomForest = 0.856448743496149"
## [1] "Media da coluna FALSE, FALSE, TRUE, classif.xgboost = 0.845946456366123"
## [1] "Media da coluna FALSE, TRUE, FALSE, classif.ksvm = 0.975176544455616"
## [1] "Media da coluna FALSE, TRUE, FALSE, classif.randomForest = 0.981998634989551"
## [1] "Media da coluna FALSE, TRUE, FALSE, classif.xgboost = 0.98271382723725"
## [1] "Media da coluna SMOTE, FALSE, FALSE, classif.ksvm = 0.979458695568733"
## [1] "Media da coluna SMOTE, FALSE, FALSE, classif.randomForest = 0.986035995107693"
## [1] "Media da coluna SMOTE, FALSE, FALSE, classif.xgboost = 0.985137394866884"
```

Fazendo teste de normalidade

```r
plotDensities(data = na.omit(df))
```

Testando as diferencas

```
friedmanTest(df)
```

```
##
##  Friedman's rank sum test
##
## data:  df
## Friedman's chi-squared = 1669.7, df = 14, p-value < 2.2e-16
```

Testando as diferencas par a par

```
test <- nemenyiTest (df, alpha=0.05)
abs(test$diff.matrix) > test$statistic
```

```
##        ADASYN, FALSE, FALSE, classif.ksvm
## [1,]                             FALSE
## [2,]                             FALSE
## [3,]                             FALSE
## [4,]                              TRUE
## [5,]                              TRUE
## [6,]                              TRUE
## [7,]                              TRUE
```

```
##  [8,]                                        TRUE
##  [9,]                                        TRUE
## [10,]                                        TRUE
## [11,]                                        TRUE
## [12,]                                        TRUE
## [13,]                                       FALSE
## [14,]                                       FALSE
## [15,]                                       FALSE
##       ADASYN, FALSE, FALSE, classif.randomForest
##  [1,]                                          FALSE
##  [2,]                                          FALSE
##  [3,]                                          FALSE
##  [4,]                                           TRUE
##  [5,]                                           TRUE
##  [6,]                                           TRUE
##  [7,]                                           TRUE
##  [8,]                                           TRUE
##  [9,]                                           TRUE
## [10,]                                           TRUE
## [11,]                                           TRUE
## [12,]                                           TRUE
## [13,]                                          FALSE
## [14,]                                          FALSE
## [15,]                                          FALSE
##       ADASYN, FALSE, FALSE, classif.xgboost
##  [1,]                                      FALSE
##  [2,]                                      FALSE
##  [3,]                                      FALSE
##  [4,]                                       TRUE
##  [5,]                                       TRUE
##  [6,]                                       TRUE
##  [7,]                                       TRUE
##  [8,]                                       TRUE
##  [9,]                                       TRUE
## [10,]                                       TRUE
## [11,]                                       TRUE
## [12,]                                       TRUE
## [13,]                                      FALSE
## [14,]                                      FALSE
## [15,]                                      FALSE
##       FALSE, FALSE, FALSE, classif.ksvm
##  [1,]                                   TRUE
##  [2,]                                   TRUE
##  [3,]                                   TRUE
##  [4,]                                  FALSE
##  [5,]                                   TRUE
##  [6,]                                   TRUE
##  [7,]                                   TRUE
##  [8,]                                   TRUE
##  [9,]                                   TRUE
## [10,]                                  FALSE
## [11,]                                   TRUE
## [12,]                                   TRUE
## [13,]                                   TRUE
```

```
## [14,]                                 TRUE
## [15,]                                 TRUE
##       FALSE, FALSE, FALSE, classif.randomForest
##  [1,]                                  TRUE
##  [2,]                                  TRUE
##  [3,]                                  TRUE
##  [4,]                                  TRUE
##  [5,]                                 FALSE
##  [6,]                                 FALSE
##  [7,]                                  TRUE
##  [8,]                                  TRUE
##  [9,]                                  TRUE
## [10,]                                  TRUE
## [11,]                                 FALSE
## [12,]                                 FALSE
## [13,]                                  TRUE
## [14,]                                  TRUE
## [15,]                                  TRUE
##       FALSE, FALSE, FALSE, classif.xgboost
##  [1,]                                 TRUE
##  [2,]                                 TRUE
##  [3,]                                 TRUE
##  [4,]                                 TRUE
##  [5,]                                FALSE
##  [6,]                                FALSE
##  [7,]                                 TRUE
##  [8,]                                 TRUE
##  [9,]                                 TRUE
## [10,]                                 TRUE
## [11,]                                FALSE
## [12,]                                FALSE
## [13,]                                 TRUE
## [14,]                                 TRUE
## [15,]                                 TRUE
##       FALSE, FALSE, TRUE, classif.ksvm
##  [1,]                              TRUE
##  [2,]                              TRUE
##  [3,]                              TRUE
##  [4,]                              TRUE
##  [5,]                              TRUE
##  [6,]                              TRUE
##  [7,]                             FALSE
##  [8,]                             FALSE
##  [9,]                             FALSE
## [10,]                              TRUE
## [11,]                              TRUE
## [12,]                              TRUE
## [13,]                              TRUE
## [14,]                              TRUE
## [15,]                              TRUE
##       FALSE, FALSE, TRUE, classif.randomForest
##  [1,]                                  TRUE
##  [2,]                                  TRUE
##  [3,]                                  TRUE
```

8

```
##  [4,]                                           TRUE
##  [5,]                                           TRUE
##  [6,]                                           TRUE
##  [7,]                                          FALSE
##  [8,]                                          FALSE
##  [9,]                                          FALSE
## [10,]                                           TRUE
## [11,]                                           TRUE
## [12,]                                           TRUE
## [13,]                                           TRUE
## [14,]                                           TRUE
## [15,]                                           TRUE
##       FALSE, FALSE, TRUE, classif.xgboost FALSE, TRUE, FALSE, classif.ksvm
##  [1,]                                TRUE                             TRUE
##  [2,]                                TRUE                             TRUE
##  [3,]                                TRUE                             TRUE
##  [4,]                                TRUE                            FALSE
##  [5,]                                TRUE                             TRUE
##  [6,]                                TRUE                             TRUE
##  [7,]                               FALSE                             TRUE
##  [8,]                               FALSE                             TRUE
##  [9,]                               FALSE                             TRUE
## [10,]                                TRUE                            FALSE
## [11,]                                TRUE                             TRUE
## [12,]                                TRUE                             TRUE
## [13,]                                TRUE                             TRUE
## [14,]                                TRUE                             TRUE
## [15,]                                TRUE                             TRUE
##       FALSE, TRUE, FALSE, classif.randomForest
##  [1,]                                   TRUE
##  [2,]                                   TRUE
##  [3,]                                   TRUE
##  [4,]                                   TRUE
##  [5,]                                  FALSE
##  [6,]                                  FALSE
##  [7,]                                   TRUE
##  [8,]                                   TRUE
##  [9,]                                   TRUE
## [10,]                                   TRUE
## [11,]                                  FALSE
## [12,]                                  FALSE
## [13,]                                   TRUE
## [14,]                                   TRUE
## [15,]                                   TRUE
##       FALSE, TRUE, FALSE, classif.xgboost
##  [1,]                                TRUE
##  [2,]                                TRUE
##  [3,]                                TRUE
##  [4,]                                TRUE
##  [5,]                               FALSE
##  [6,]                               FALSE
##  [7,]                                TRUE
##  [8,]                                TRUE
##  [9,]                                TRUE
```

```
## [10,]                                           TRUE
## [11,]                                          FALSE
## [12,]                                          FALSE
## [13,]                                           TRUE
## [14,]                                           TRUE
## [15,]                                           TRUE
##          SMOTE, FALSE, FALSE, classif.ksvm
##  [1,]                                          FALSE
##  [2,]                                          FALSE
##  [3,]                                          FALSE
##  [4,]                                           TRUE
##  [5,]                                           TRUE
##  [6,]                                           TRUE
##  [7,]                                           TRUE
##  [8,]                                           TRUE
##  [9,]                                           TRUE
## [10,]                                           TRUE
## [11,]                                           TRUE
## [12,]                                           TRUE
## [13,]                                          FALSE
## [14,]                                          FALSE
## [15,]                                          FALSE
##          SMOTE, FALSE, FALSE, classif.randomForest
##  [1,]                                                FALSE
##  [2,]                                                FALSE
##  [3,]                                                FALSE
##  [4,]                                                 TRUE
##  [5,]                                                 TRUE
##  [6,]                                                 TRUE
##  [7,]                                                 TRUE
##  [8,]                                                 TRUE
##  [9,]                                                 TRUE
## [10,]                                                 TRUE
## [11,]                                                 TRUE
## [12,]                                                 TRUE
## [13,]                                                FALSE
## [14,]                                                FALSE
## [15,]                                                FALSE
##          SMOTE, FALSE, FALSE, classif.xgboost
##  [1,]                                            FALSE
##  [2,]                                            FALSE
##  [3,]                                            FALSE
##  [4,]                                             TRUE
##  [5,]                                             TRUE
##  [6,]                                             TRUE
##  [7,]                                             TRUE
##  [8,]                                             TRUE
##  [9,]                                             TRUE
## [10,]                                             TRUE
## [11,]                                             TRUE
## [12,]                                             TRUE
## [13,]                                            FALSE
## [14,]                                            FALSE
## [15,]                                            FALSE
```

Plotando grafico de Critical Diference

```
result = tryCatch({
    plotCD(df, alpha=0.05, cex = 0.35)
}, error = function(e) {})
```