# Football attendance in Portugal

*Lúcia Moreira, Irene Azzini*

*01/05/2019*

# Summary

Football attendance is a difficult regression problem, but an important one for planning football events. A dataset from the main Portuguese football league was given to find the best model that predicts the attendance of a match given the host, the visitor, the date and match performance. The accumulated points in a current league were calculated and the data set was also enriched with the capacity of the stadium of the host, the number of the football club members, for the host and visitor, and the time of the match. This assignment compares regression trees, bagging, random forest, boosting and kernel-based approaches (support vector machines) to address the problem. The best model for this regression exercise was a trees-based approach either coupled with bagging or random forest as these models present the lower MSE test. The most relevant variables to evaluate the football attendance showed to be the number of club members, both for visitor and host, and stadium capacity.

# Dataset

The dataset provides the attendance per match for six seasons. It contains date, teams, results at half time and full time for all the matches in the competitions. Dataset was enriched with the time of the match [1] (**timeofthematch**), the capacity of the host stadiums [2] (**StadiumCapacity**) and club members of the host and visitor [3] (**ClubMembersHost**, **ClubMembersVisitor**). We have introduced also two dummmies binary variables, the first take into account if the match was performed during the weekend (dummy equal to 1) or during a weekday (**dummiesday**), the second one contains the information if the time of the match was in the evening (dummy equal to 1) or in the afternoon (**dummiestime**). We have considered 18:30h as a limit between afternoon and evening. A variable with the accumulated points in the season was calculated from the results of the match and is related to the performance of the club in the season (**AccpointsHFT**, **AccpointsVFT**). A variable **Jornada** was also used as a discretized variable that groups expected weather at the match, season of the year and timeline of the league in a given year. League usually begins in August and finishes in May. This way, initial Jornada numbers occur in Summer and the league finishes in Spring. Higher numbers attributed for Jornada are related to final matches of the league in that year. A discrete variable (**changeRes**) accounting if the result of the match kept the same (0), or changed for favouruing the host (+1), or desfavouring (-1) the host was also used in model evaluation. Only for the SVM classification approach, several dummied were created according to the EM analysis performed. In this case, the dummies replaced the continous logarithm of the variables Attendance, Stadium Capacity and Club Members.

# Pre-analysis of the data

*Fig.1* gives a general snapshot of the attendance to the Portuguese football league. Data shows that nearly 65% of the all attendance population in the data set (6 seasons) are attendees when 1st - Benfica, 2nd- FCPorto and 3rd- Sporting are playing as Hosts. When 4th- Victoria de Guimarães and 5th- SC Braga are included attendance population percentage increases to 80 %. Attendees in all other matches are much smaller compared to this 5 ones (ca. 20%).

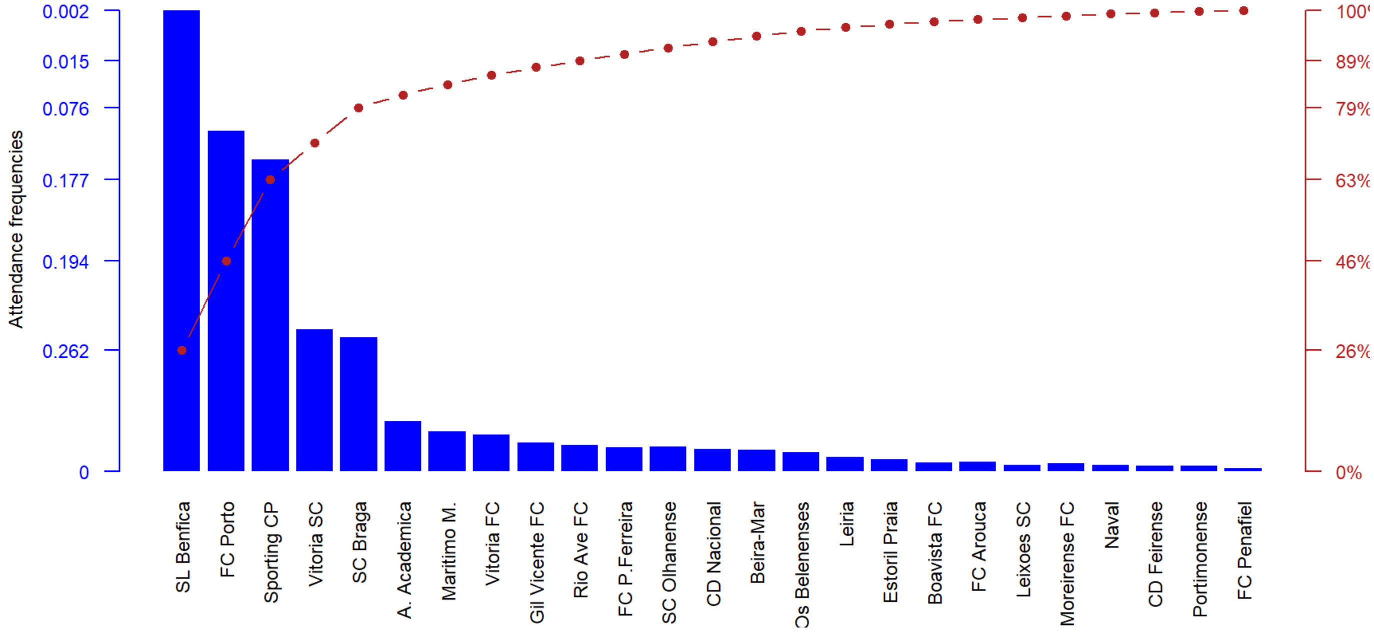

Figure 1 : Pareto for Attendance

*Fig.2* shows the average of attendance by Jornada (Round) for the 6 seasons. As explained above this discrete variable accounts for expected weather at the time of match, season of the year and timeline of the league in a given year. League usually begins in August and finishes in May. First Jornada numbers occur in Summer and the league finishes in Spring. Higher numbers attributed for Jornada are related to final matches of the league in that year. There are alternate attendance numbers at begining of the season that decreases up to middle season (probably related to winter, colder months) and afterwards starts increasing and attaining high attendance rates due to the approaching of the final decision of the season. Only one Season accounts for 34 Jornadas while the others only 31. This way, the average of attendance is lower for the last Jornadas of season 2013/2014.

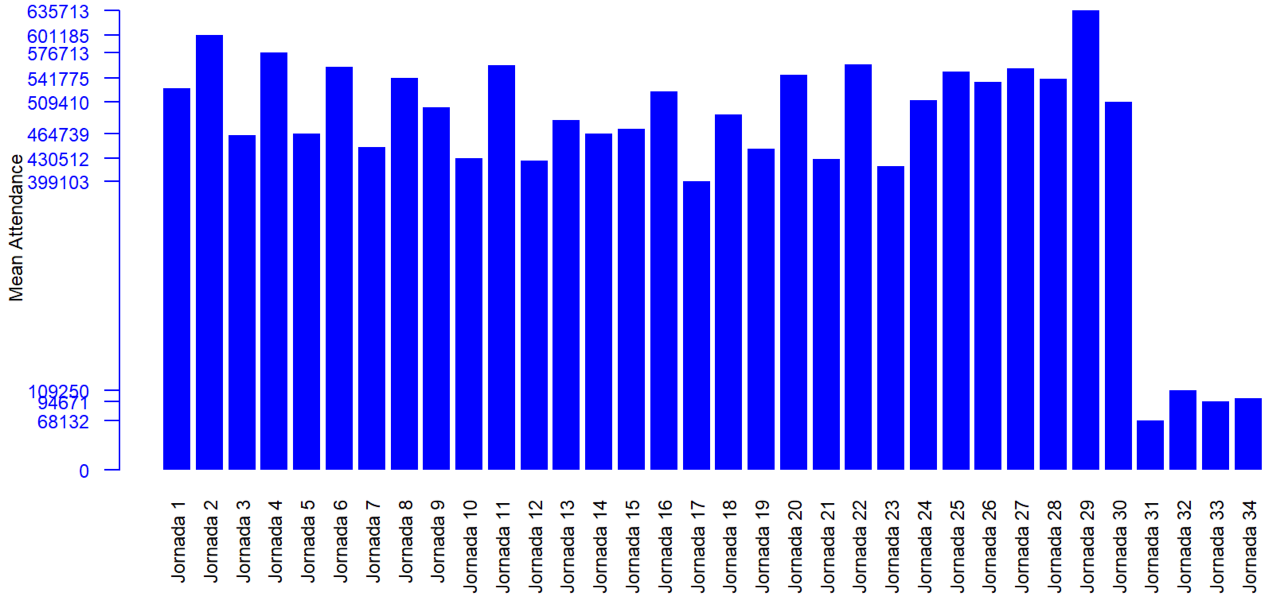

Figure 2 : Averaged attendance by Round

*Fig. 3* shows the distribution of the time of the match according to the Host club. It can be seen that the five first clubs with the highest attendance all occur in average at evening time (correlating with ca. 80% of total attendance to the stadium) (above the red line). Few questions arise: Is this a time more convenient to the public? Or is it more related to TV broadcasting? Will our dataset help to infer about this subject?

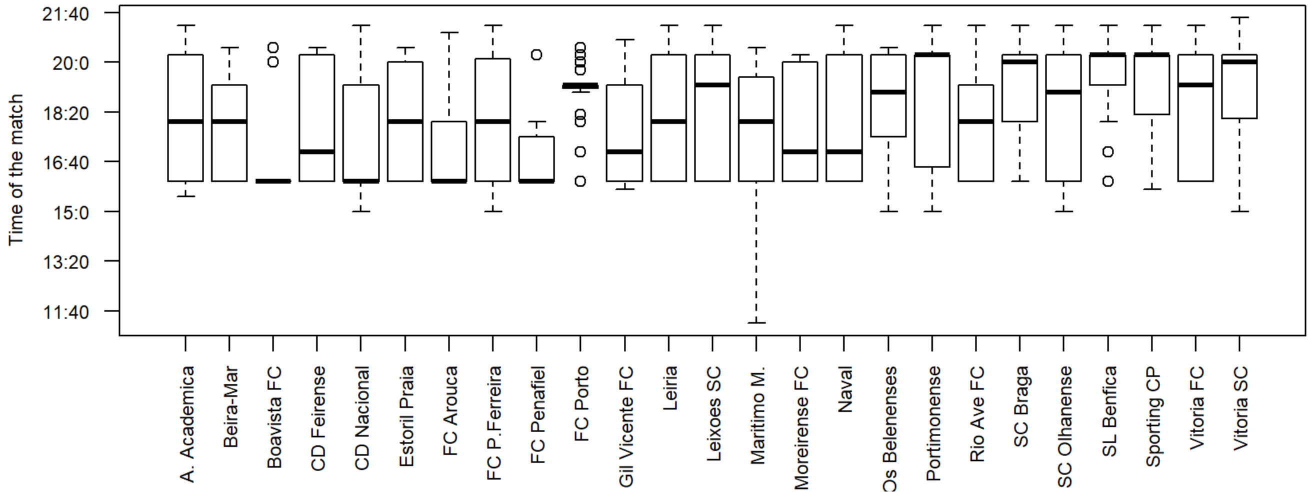

Figure 3 : Distribution of the time of the match by host club

*Fig. 4* shows the distribution of the days of the week that matches take place. Most of the matches happen at weekends with a minor number taking place at Fridays and Mondays. Other days of the week are quite rare.

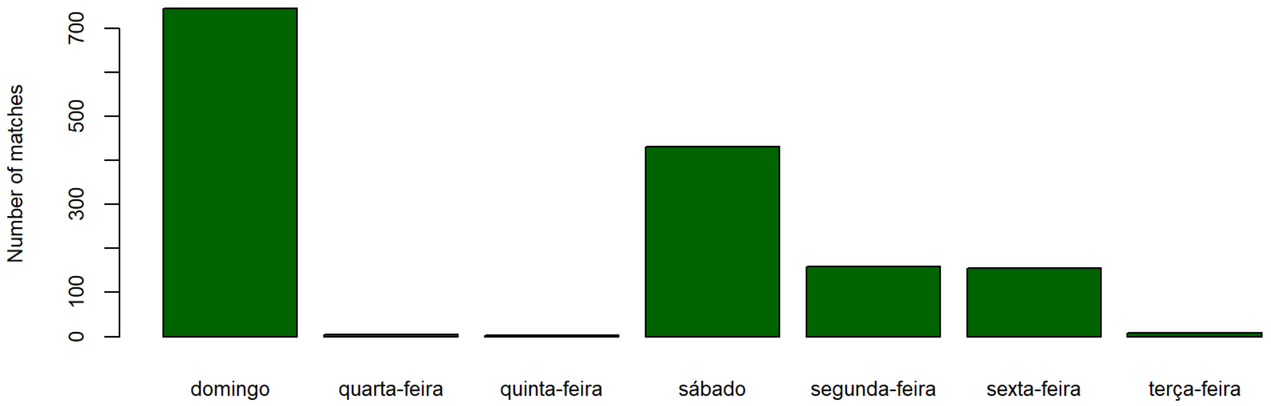

Figure 4 : Distribution of attendance by day of the week for the match

*Fig. 5* shows the average accumulated points from the six seasons for each of the football clubs. A higher mean in the boxplot shows a higher position in the season classification. Clubs with higher accumulated points are FC Porto, SL Benfica, SC Braga and Sporting CP. During the six seasons in the present dataset FCPorto or SL Benfica were the winners because those clubs have the highest accumulated data points by a significant amount.

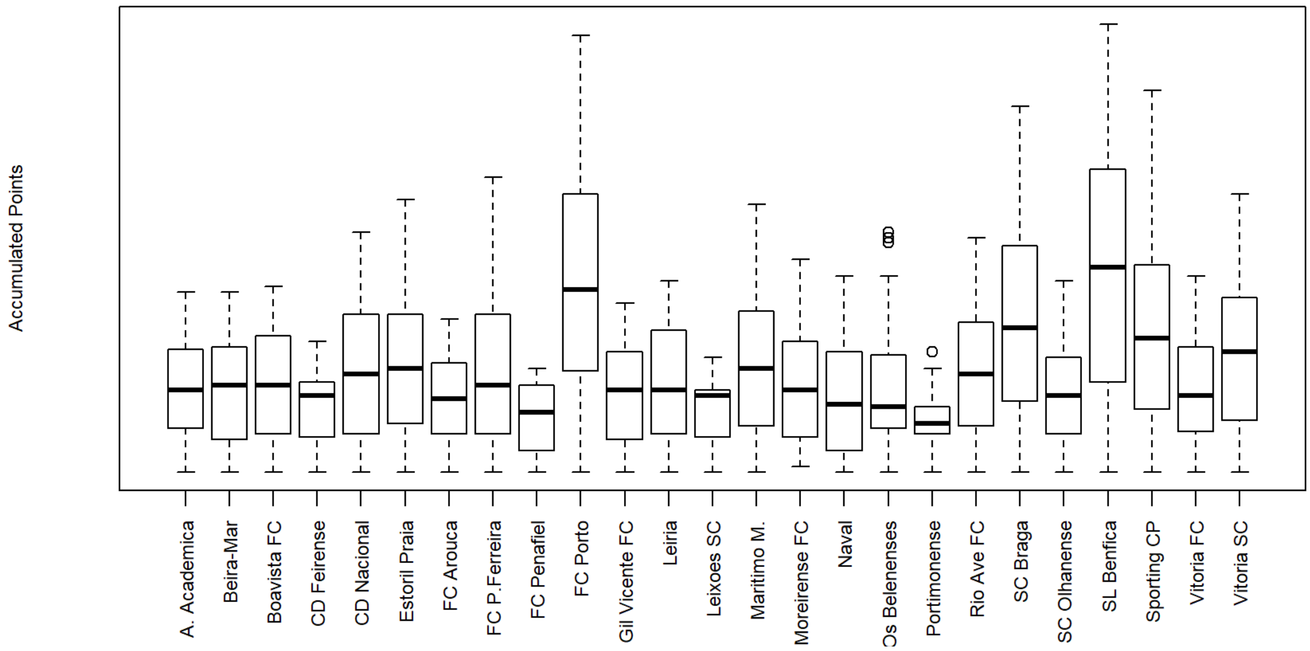

Figure 5 : Average accumulated points for each football team

# EM analysis

Mixture models are useful tools for density estimation where the Gaussian mixture model is by far the most popular. The parameters of the model are fit by maximum likelihood, using the **Expectation Maximization** (EM) algorithm. This algorithm is a popular tool for simplifying difficult maximum likelihood problems. The distribution of the attendance response variable was studied by EM to evaluate the presence of a mixed population. Analysis indicated the existence of 4 normal distributions of the logarithm of the attendance with equal variance. The same procedure was used for host capacity and club members. For a classification approach using SVM, all the continuous variables were discretized according to the normal distributions uncovered by this EM analysis. Study of the not logarithmized values for the continuous variables gave a poor adjustment to the EM analysis (variance of the normal population subsets was very high). Discrete variables could not be analyzed by EM once is best suitable for continuous variables. We would like to model the density of the **logAttendance** variable, and due to the apparent possible bi-modality (*Fig. 6*), a simple Gaussian distribution would not be appropriate. *Fig. 6A* (left) presents the density estimation via Gaussian finite mixture modeling using Mclust E (univariate, equal variance) showing a 5 component population in the logAttendance data. In order to decrease the number of components for better discretization in the classification SVM setting, only 4 components were considered.*Fig.6B* (right) shows the components found by EM with 4 groups (the lowest attendance component was grouped with the second lowest attendance).

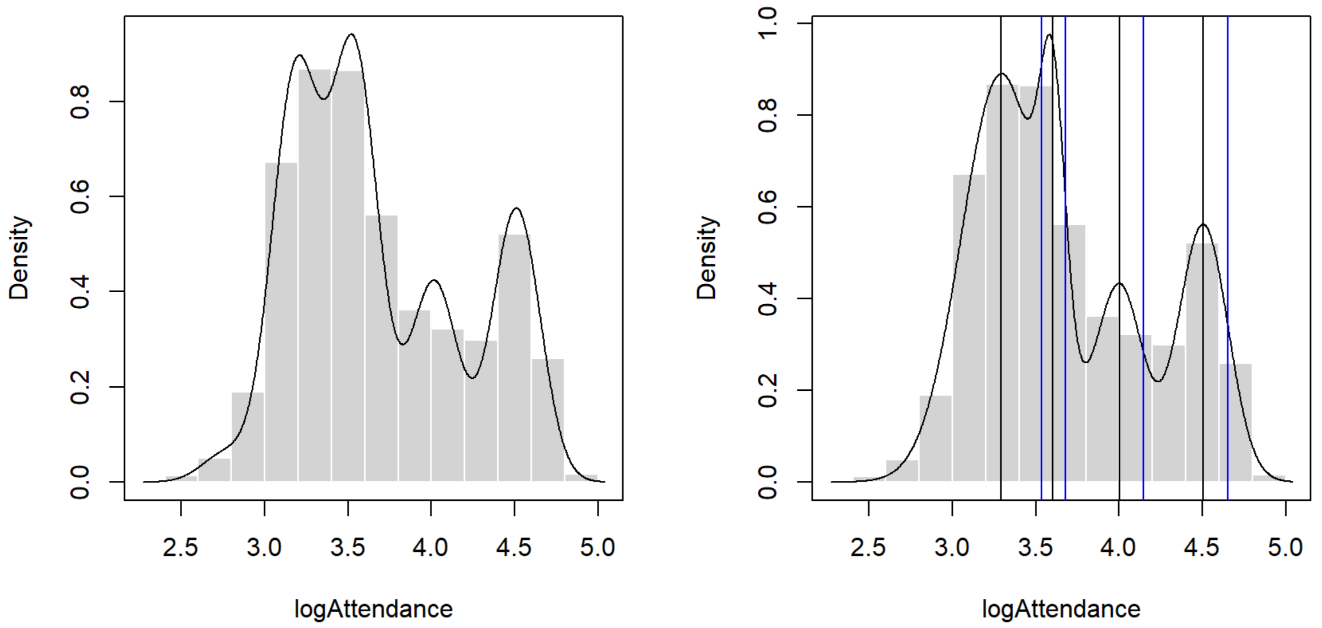

Figure 6 : EM without predefinition of the number of populations in logAttendance data (left-6A). Predefining 4 groups (blue line shows mean of each component (black) and plus one standard variation that was used for limiting of each of the groups (right-6B)

We would like to model the density of the **logStadCapacity** variable.*Fig.7A* (left) shows the density estimation via a Gaussian finite mixture modeling (univariate, unequal variance) with 6 components. At a logStadiumCapacity of 4.7 two populations were found with nearly same mean, so we decreased to only four groups in order to simplify discretization in the classification SVM approach.*Fig.7B* (right) shows the 4 components found by EM analysis.

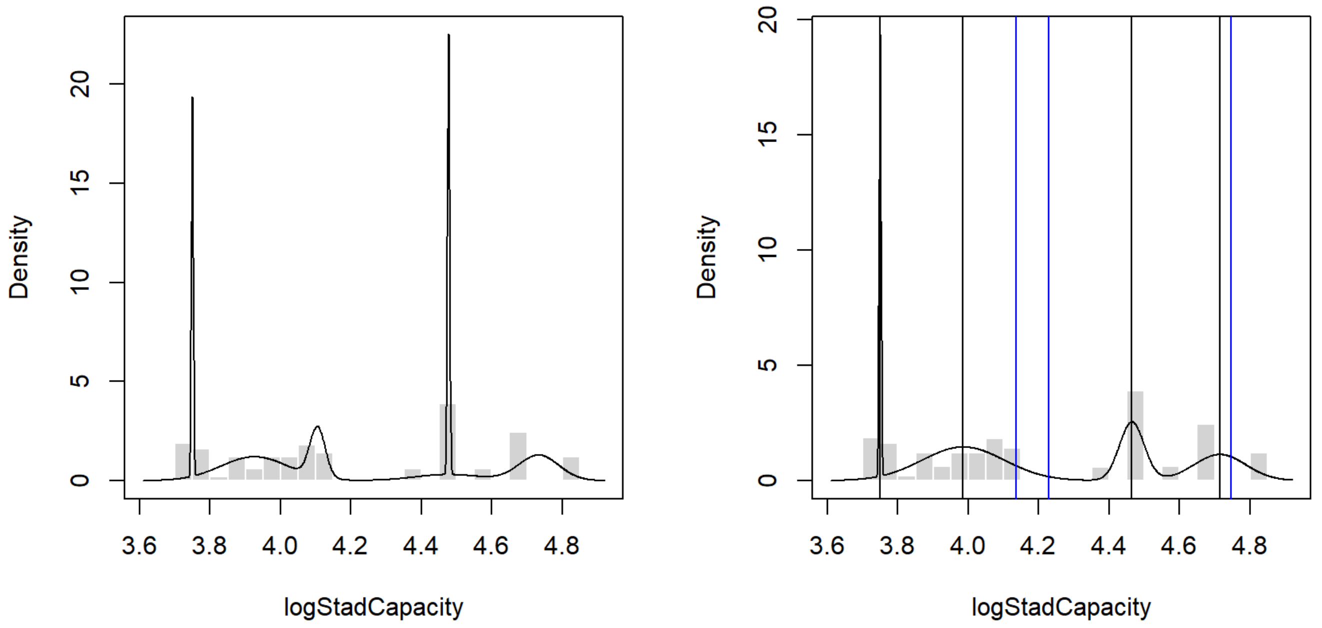

Figure 7 : EM analysis without predefinition of the number of populations in logStadiumCapacity data (left-7A). Predefining 4 groups (blue line shows mean of each component (black vertical) line plus two standard variations that was used for limiting each of the groups (right-7B)

The same way, *Fig.8A* (left) shows the density estimation for **logClubMembers** variable via Gaussian finite mixture modeling (univariate, equal variance) finding 9 components. In order to simplify the number of components, EM analysis on this variable was performed in second approach with only 6 components (*Fig.8B*(right)).

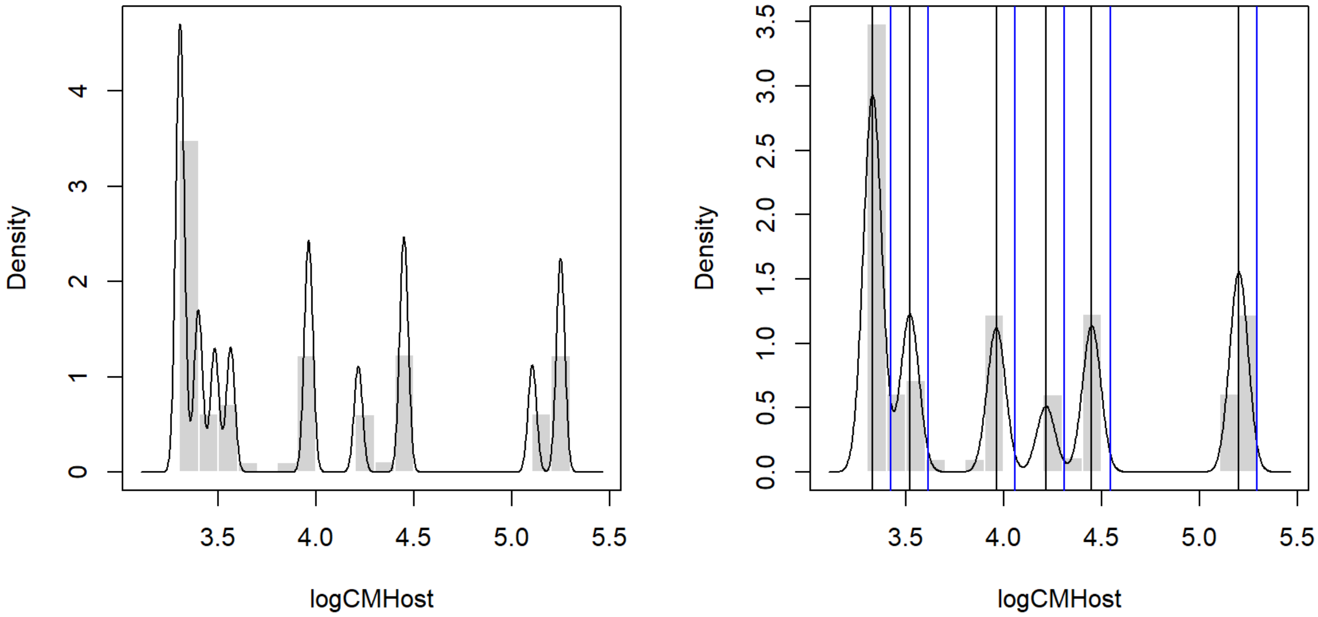

Figure 8 : EM analysis without predefinition of the number of populations in logClubMembers data(8A-left). Predefining 6 groups (blue line shows mean of each component (black vertical) line plus two standard variations that was used for limiting each of the groups (8B-right)

# Regression models

For our work we have compared the following models: **linear regression**, **trees**, **bagging**, **random forest**, **boosting** and **SVM**. For all of these methods we have calculated the ***Mean Squared Error*** *(MSE) in order to choose which is the best model for doing a regression on the logAttendance.* **Accuracy** was calculated for the classification SVM approach.

# Split of the dataset:

We have divided the dataset in train and test, using 70% of the variables chosen randomly as train, and the remainder as test. To make our study as proper as possible we removed two rows corresponding to missing values for the Attendance (lines 935 and 989). The variables that we have used in this work to predict logAttendance are: logCMHost, logCVisitor, logStadCapacity, dummiesday, pointsHFT, pointsVFT, Jornada, AccpointsHFT, AccpointsVFT, changeRes, dummiestime, dummiesSC, dummiesCMH and dummiesCMV. We already have explained the meaning of all these variables. For what concern the variables StadiumCapacity, ClubMembersHost and ClubMembersVisitor as predictors, we have used their log10 version in the regression setting and the dummies version, in this case obtained thanks to EM, in the SVM classification setting.

# Linear models:

The first model that we have used to predict logAttendance was **linear model**, to fit this model we have used the function *glm* in order to be as general as possible.

Using all the variables to make prediction we have noticed that almost all the variables are significant with exception for pointsHFT, pointsVFT, AccPointsVFT and changeRes. Also dummiestime does not seem very significant (p-value 0.017) but we have decided to keep it in the regression model. So we have redone the linear model eliminating one at a time the variables that do not seem significant.

The summary of the final model is:

```
##
## Call:
## glm(formula = logAttendance ~ logCMHost + logCMVisitor + logStadCapacity +
##     dummiesday + Jornada + AccpointsHFT + dummiestime, data = train2_orig)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -0.93439  -0.12545   0.01851   0.14681   0.71600
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -0.4515538  0.1206494  -3.743 0.000192 ***
## logCMHost        0.3371270  0.0185427  18.181  < 2e-16 ***
## logCMVisitor     0.1685773  0.0110369  15.274  < 2e-16 ***
## logStadCapacity  0.4858656  0.0336400  14.443  < 2e-16 ***
## dummiesday       0.0986568  0.0199013   4.957 8.34e-07 ***
## Jornada         -0.0153953  0.0015454  -9.962  < 2e-16 ***
## AccpointsHFT     0.0107477  0.0009621  11.171  < 2e-16 ***
## dummiestime      0.0442953  0.0180987   2.447 0.014552 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.05587035)
##
##     Null deviance: 298.302  on 1051  degrees of freedom
## Residual deviance:  58.329  on 1044  degrees of freedom
## AIC: -39.311
##
## Number of Fisher Scoring iterations: 2
```

So the final model has as predictors: **logCMHost**, **logCMVisitor**, **logStadCapacity**, **dummiesday**, **Jornada**, **AccpointsHFT** and **dummiestime**. The first six variables are very significant (p-value low). It is possible to notice that the club members for both, host and visitor, has a positive influence on the attendance, the hosts have more weight and this has sense because usually the most of the people that see the game at the stadium belong to the host team. We also expected the fact that the capacity of the stadium is significant, the biggest is the stadium the more people can attend the game. For what concerns the dummies day, we have already seen that during the weekend the mean attendance is bigger with respect to the weekdays; this also holds for the dummiestime, indeed usually during the weekend the team plays later than during the weekdays. From this model, we can also see that as the number of accumulated points for host team increases the attendance at the stadium also increases. We interpreted it as the more a team is doing well in the competition the more people are interested in going to see the match.

Using this model for doing prediction we obtain:

| **MSE** | |
|---|---|
| MSE.train | 0.0555 |
| MSE.test | 0.0581 |

Table 1 : MSE for linear regression

# Trees:

The second method that we have tried was **trees**, and then improved with **bagging**, **random forest** and **boosting**.

By applying trees to our dataset, it can be noticed (*Fig.9A*(left)) that logCMHost is the most important variable in the evaluation of logAttendance. The relevant variables are only 3: **logCMHost**, **logCMVisitor**, **logStadCapacity**. If the amount of club members for the host is bigger than ca. 28369 the only other variable that influences the attendance is the capacity of the stadium (limit ~ 38928). Otherwise logCMVisitor, logCMHost and logStadCapacity influence the logAttendance. For example, the less attendance (around 1606) is obtained when the club members for the visitor is below 61745, for the host is below 9487 and the stadium capacity is smaller than 11900. This agrees with what we expected, as already said, most of the people that are going to see a match belongs to the host team. The stadium capacity is obviously a limit for the attendance at a match and also the number of people that belongs to the club members of the visitor team have influence because probably they will participate in all the games. (Obs. we speak about the real value and not the logaritmic ones to make the things as interpretable as possible).

The summary of the tree is:

```
##
## Regression tree:
## tree(formula = logAttendance ~ ., data = train2_orig)
## Variables actually used in tree construction:
## [1] "logCMHost"       "logStadCapacity" "logCMVisitor"
## Number of terminal nodes:  8
## Residual mean deviance:  0.04108 = 42.88 / 1044
## Distribution of residuals:
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -0.714100 -0.121100 -0.004421  0.000000  0.117500  0.724100
```

To see if it is possible to reduce the size of the tree, that originally is equal to 8, we have done trees with *Cross Validation* with 10 folds, and we have seen (*Fig.10*) that it is possible to use a tree of size 5 (shown in *Fig.9B*(right)) without incrasing a lot the deviance (sum of squared errors for the tree), in this case the predictor used in the construction were only logCMHost and logCMVisitor.

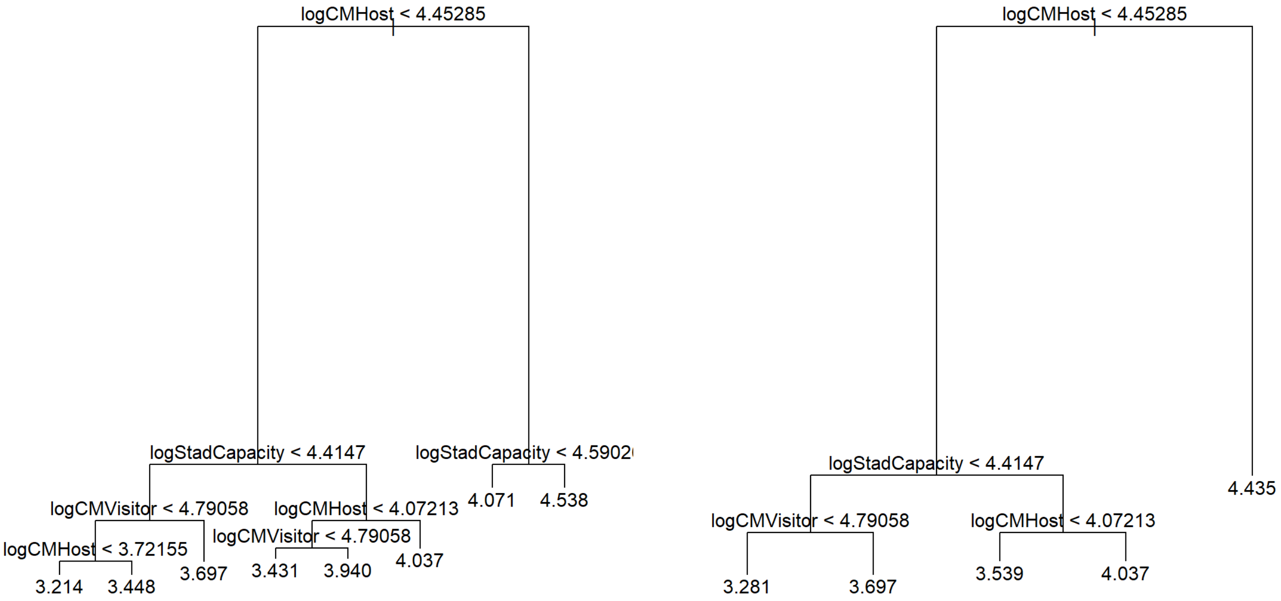

Figure 9 : Trees unpruned (9A-left) and with pruned(5) (9B-right)

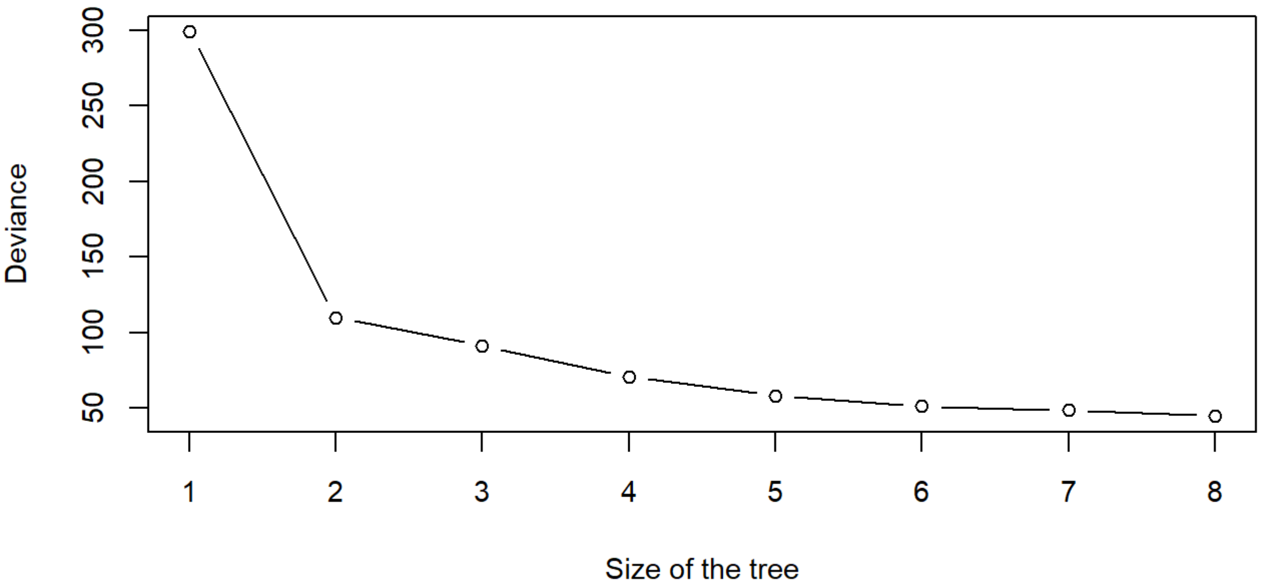

Figure 10 : Deviance against size of the tree

Making prediction with the unpruned tree we obtain the following MSE for train and test.

| **MSE** | |
|---|---|
| MSE.train | 0.0407 |
| MSE.test | 0.0421 |

Table 2 : MSE for unpruned trees

Since MSE is lower with trees than with linear regression we can infer that models like tree represent better our type of problem. So probably the relationship between the predictors and the response is predominantly non-linear. So we had to prefer to fit our model like $\hat{f} = \sum_{m=1}^{M} c_m I_{x \in R_m}$, where $R_1, \ldots, R_m$ is the partition of the preditors space.

One of the problems of using tree models to make predictions is that decision trees suffer from high variance, indeed with different splitting of the training data the results could be quite different. To solve this problem we have used three different methods: bagging, random forest and boosting. For all this models we have used CV with 10 folds in order to find the best hyperparameters.

# Bagging:

The first method that we have tried is **bagging**, this method in order to reduce the variance returns as result the average of the outcomes obtained using bootstrap on our training data. In practice, it constructs B regression trees from the same sample (bootstrapped) and makes the average of the resulting predictions. $\hat{f}_{bag} = \frac{1}{B} \sum_{b=1}^{B} \hat{f}^{*}_{b}(x)$ where, $B$ is the number of regression trees that we construct. So, the subsample obtained using bootstrap $\hat{f}^{*}_{b}(x)$ is the result obtained applying the method on the b-th bootstrapped training set.

To make this method as expressive as possible we run it with three different values for the variables corresponding to the number of trees, $ntrees = [100, 500, 800]$ and we have discovered that the best value is obtained for $ntrees = 500$.

The results obtained applying bagging are really better with respect to the ones obtained only with trees, indeed MSE is greatly reduced.

| **MSE** | |
|---|---|
| MSE.train | 0.0053 |
| MSE.test | 0.0298 |

Table 3 : MSE for bagging with $ntree = 500$

The summary of this model is:

```
## 
## Call:
##  randomForest(formula = logAttendance ~ ., data = train2_orig,      mtry = 11, ntree = tre
evalues[pos.bagg], importance = TRUE) 
##                Type of random forest: regression
##                      Number of trees: 800
## No. of variables tried at each split: 11
## 
##           Mean of squared residuals: 0.02720856
##                     % Var explained: 90.4
```

In *Fig.11* we report the plot of the importance of the variables, this confirms that the important variables are **logCMHost**, **logCMVisitor** and **logStadCapacity**.

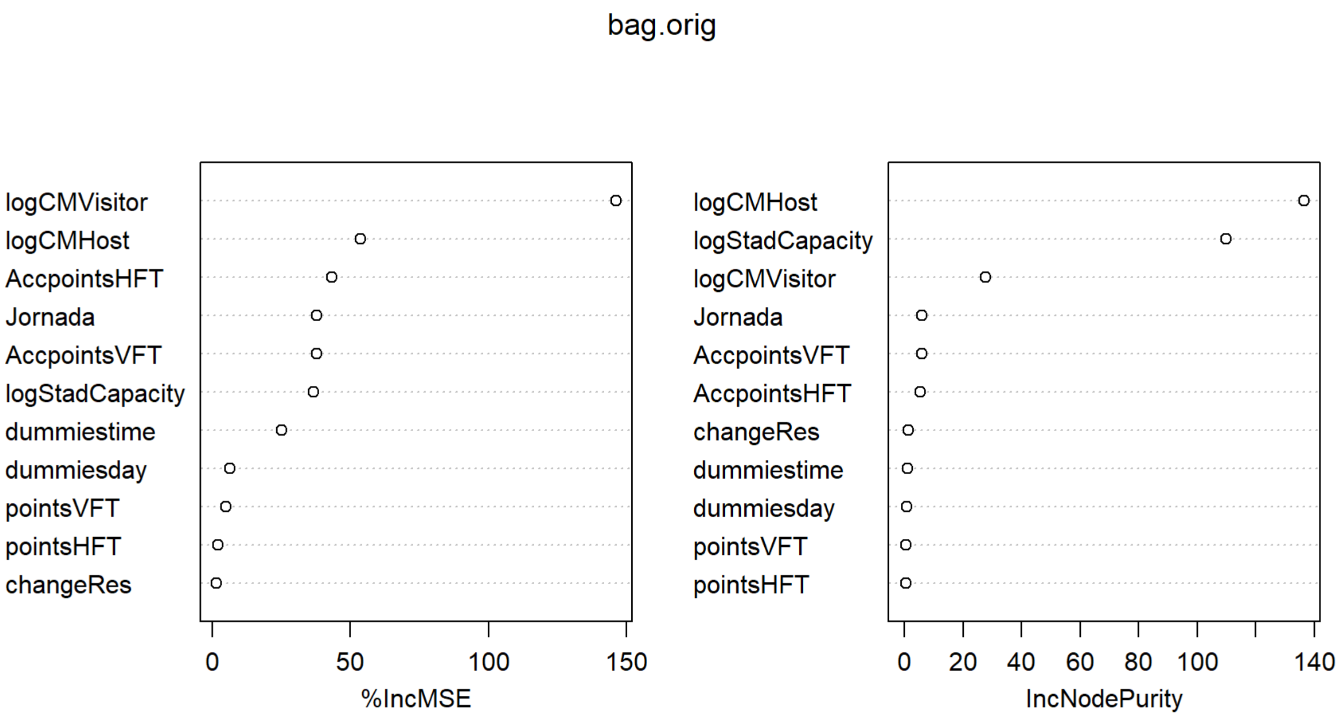

Figure 11 : Importance of variables

*%IncMSE* is based upon the mean decrease of accuracy in predictions on the Out of Bag samples when a given variable is excluded from the model. *IncNodePurity* is a measure of the total decrease in node impurity that results from splits over that variable, averaged over all trees.

So we can deduce that bagging improves trees-based model approach, the problem of using this methods is that the results are more difficult to be interpreted.

# Random Forest:

A way to improve bagging is use *Random Forest*, the idea of these two algorithms is the same, build a number of decision trees on bootstrapped training samples, but in this case each time that a decision trees is built, it is considered a split in the tree (a random sample of $m$ predictors is choosen as split candidates from the full set of parameters). To choose the most expressive model we test how was MSE considering $m = [3, 4] andntrees = [100, 500, 800]$ and we have obtained that the highest expressivness is obtained with $m = 4, ntrees = 500$.

The summmary of the model obtained is:

```
##
## Call:
##  randomForest(formula = logAttendance ~ ., data = train2_orig,      mtry = mvalues[pos.R
F], ntree = treevaluesRF[pos.RF], importance = TRUE)
##                Type of random forest: regression
##                      Number of trees: 500
## No. of variables tried at each split: 4
##
##           Mean of squared residuals: 0.02852714
##                     % Var explained: 89.94
```

The MSE are:

| MSE | |
|---|---|
| MSE.train | 0.0066 |

| MSE | |
|---|---|
| MSE.test | 0.0295 |

Table 4 : MSE for random forest with $m = 4, ntrees = 500$

So the results obtained using Random Forest are quite similar with the ones collected using bagging, this probably because none of the predictors is most relevant with respect to the other and so also bagging can detect well the regression model. From *Fig.12* we can see that the important variables are also logCMHost, logCMVisitor and logStadiumCapacity.

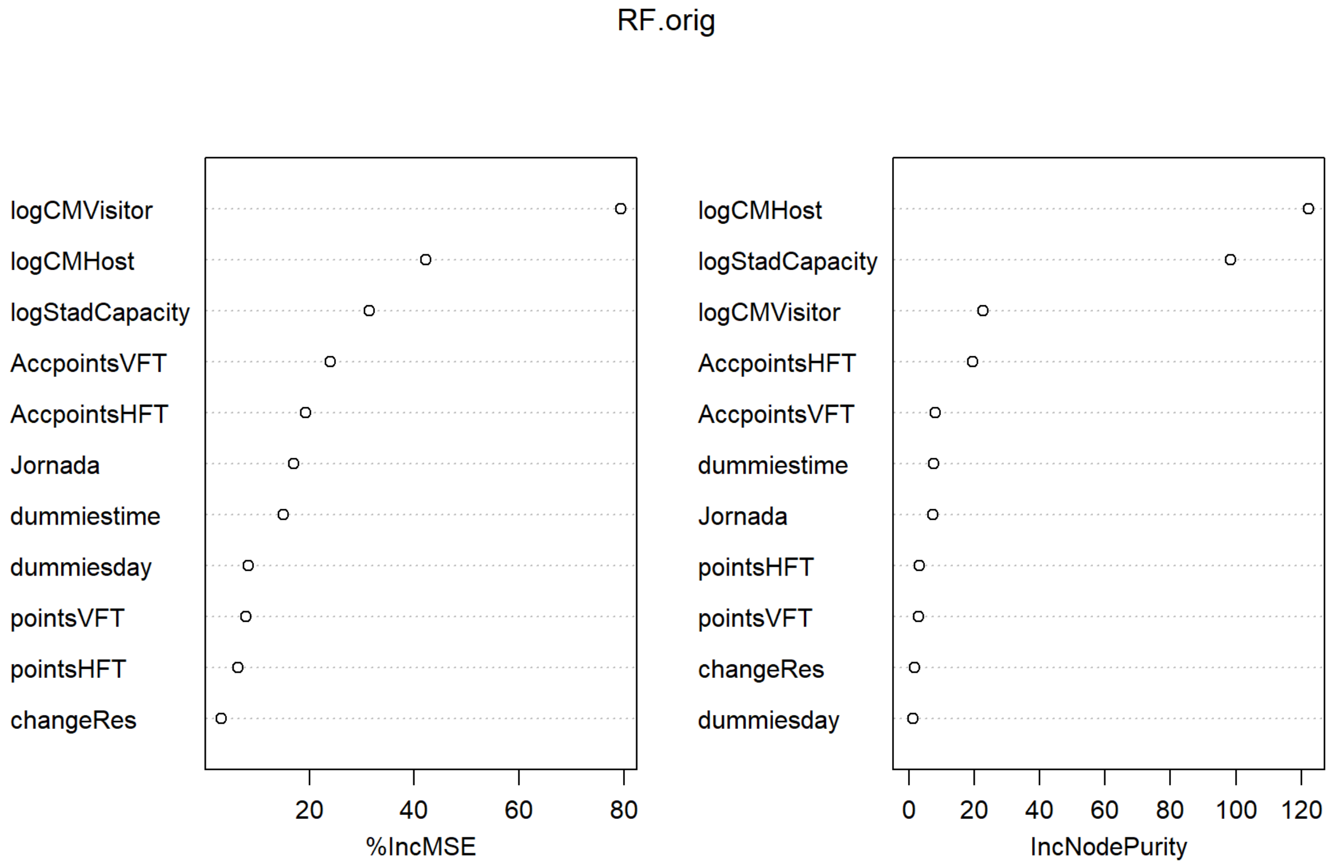

Figure 12: Importance of variables

# Boosting:

The difference introduced by *boosting* is that the trees in this case are grown sequentially using the information obtained from the trees obtained previously.

To study the expressivness of boosting we have trained it with different values for its tuning parameters, in particular we have used $\lambda = [0.01, 0.001], depthmaxforeachtrees = [1, 2], ntrees = [100, 500, 800]$ and we have obtained that the best model is for $\lambda = 0.01, ntrees = 800, depthmaxforeachtrees = 2.$ With these values we have observed that the important predictors are also logCMHost, logStadCapaciy and logCMVisitor, followed by Joranda and AccPoitnsHost (see summary below).

```
##                           var     rel.inf
## logCMHost             logCMHost 44.73683790
## logStadCapacity logStadCapacity 42.74019687
## logCMVisitor       logCMVisitor 10.17158682
## Jornada                 Jornada  1.10466306
## AccpointsHFT       AccpointsHFT  0.66589458
## dummiesday           dummiesday  0.28005772
## AccpointsVFT       AccpointsVFT  0.15087070
## changeRes             changeRes  0.11353694
## pointsHFT             pointsHFT  0.01675718
## pointsVFT             pointsVFT  0.01126637
## dummiestime         dummiestime  0.00833186
```

The MSE obtained with this method are:

| MSE | |
|---|---|
| MSE.train | 0.0304 |
| MSE.test | 0.0334 |

Table 5 : MSE for boosting with $\lambda = 0.01, ntrees = 800, depthmaxforeachtrees = 2$

From the MSE values in *table 5* we can deduce that boosting improves the result for trees but not as random forest or bagging do.

## Tree-based methods confrontation

In order to visualize the results obtained with trees, bagging, random forest and boosting and to see which are the most expressive methods, we present, in *figure 13*, the values of MSE for test for different tree sizes (from 1 to 2000) with the parameters that we have found to be the best for the different models.

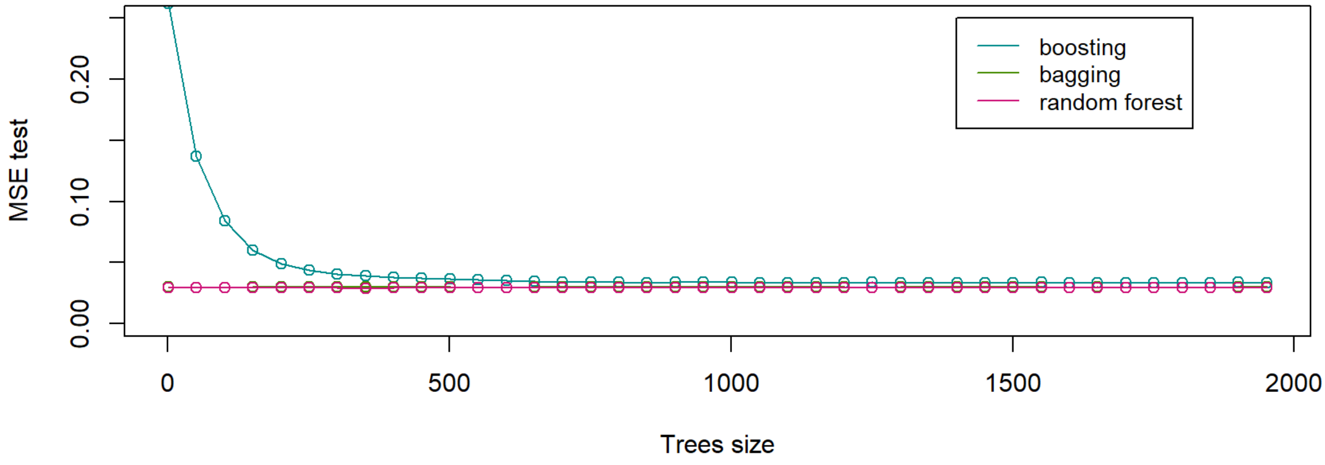

Figure 13 : MSE test errors confrontation

# Supporting Vector Machine Regression

Regression of the logarithm of Attendance was performed as function of the predictors: **log of Club members**, for Host and Visitor, **log of the Stadium Capacity**, **Accumulated points** in the season before the match, and **Jornada** (Round) number and two more logical dummies variables of the **Time** and **Date** of the match accounting if time of the match was in the afternoon or at the evening while the other accounting if the game was played at weekend or during a weekday. **Linear**, **polynomial**, **radial** and **sigmoid kernels** were used with a *10-fold cross validation*(CV) for determination of the best hyperparameters for each kernel-based method.

*Fig. 14 to 17* show the MSE for each of the kernels tested presenting the mean least squared errors for the hyperparameter obtained by CV.

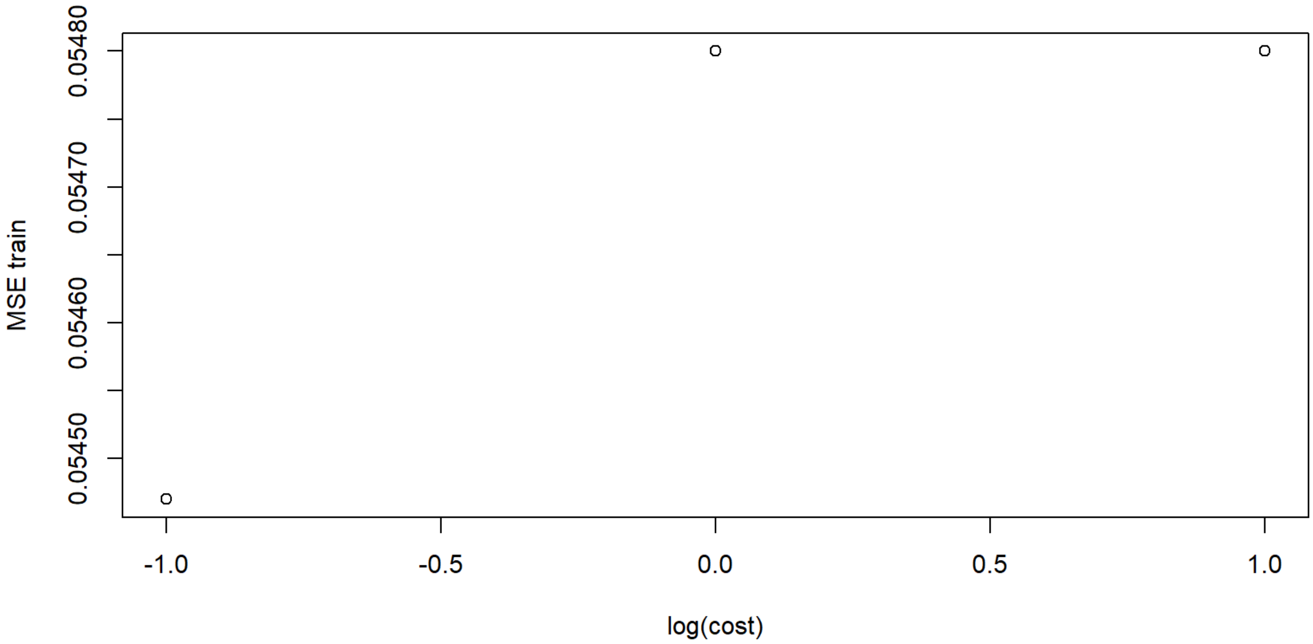

Figure 14 : Linear kernel regression SVM

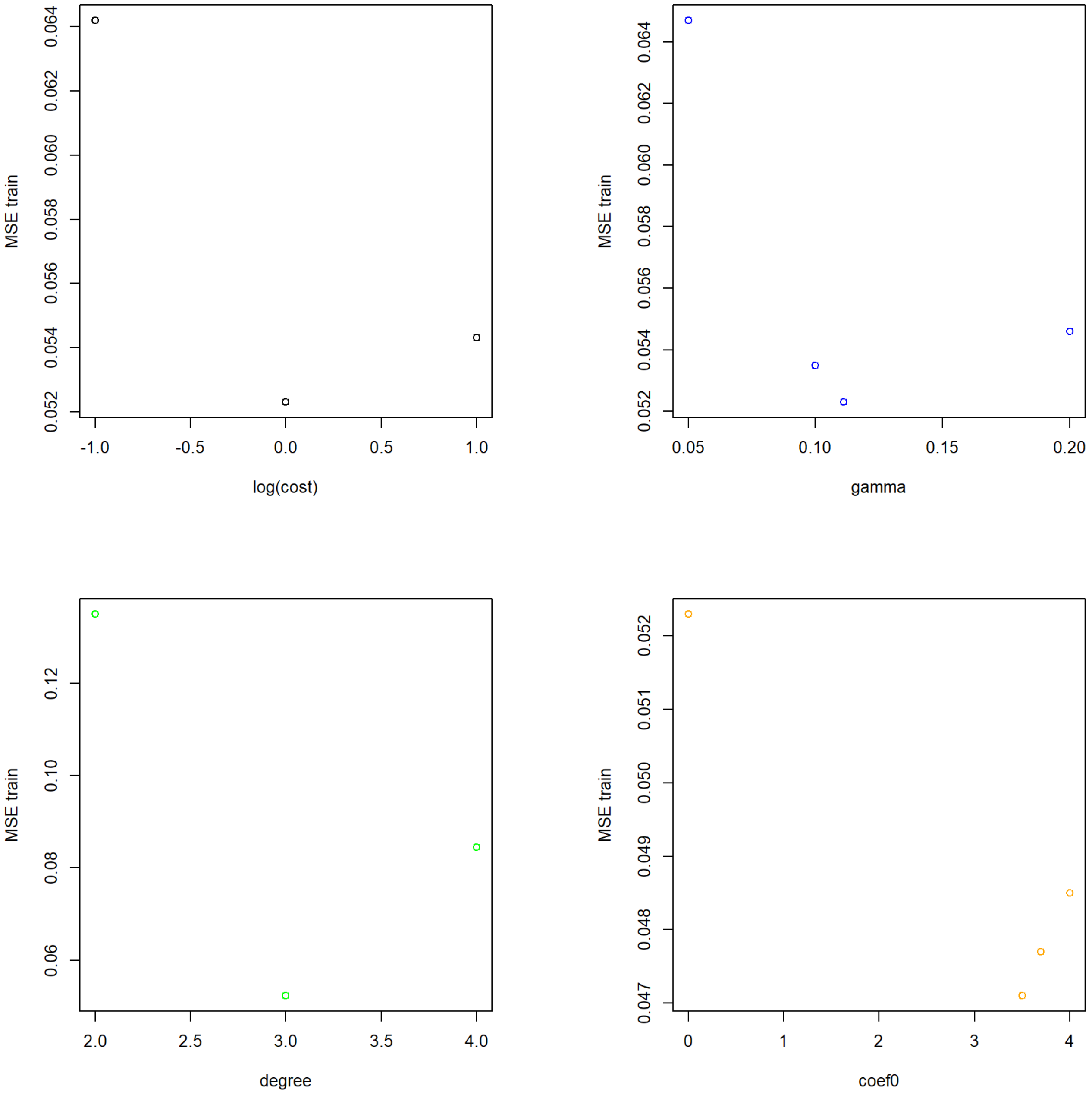

Figure 15 : Polynomial kernel regression SVM

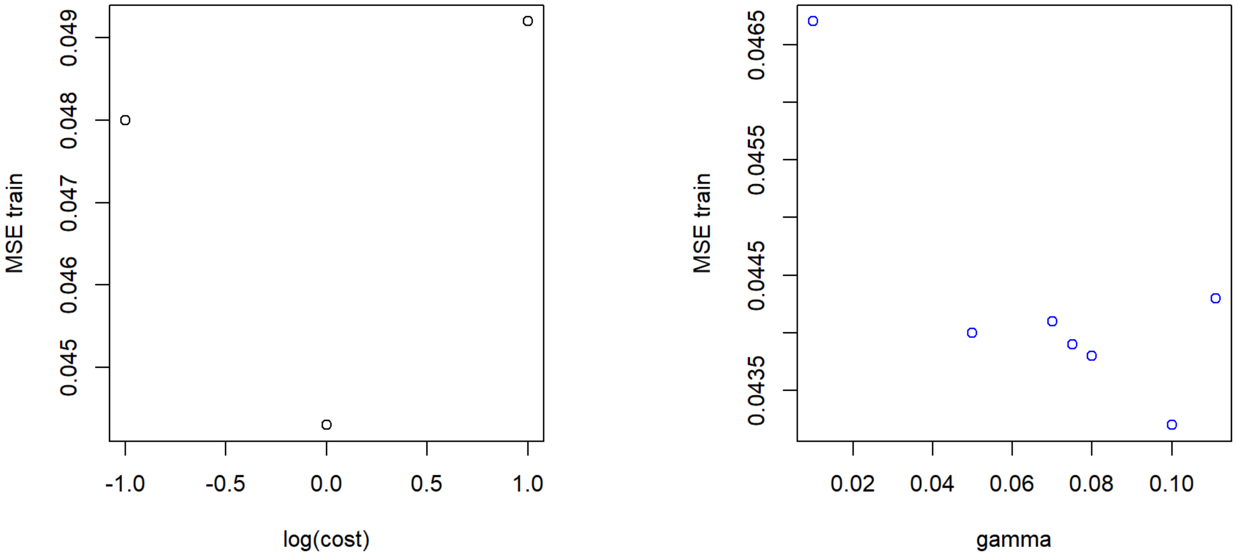

Figure 16 : Radial kernel regression SVM

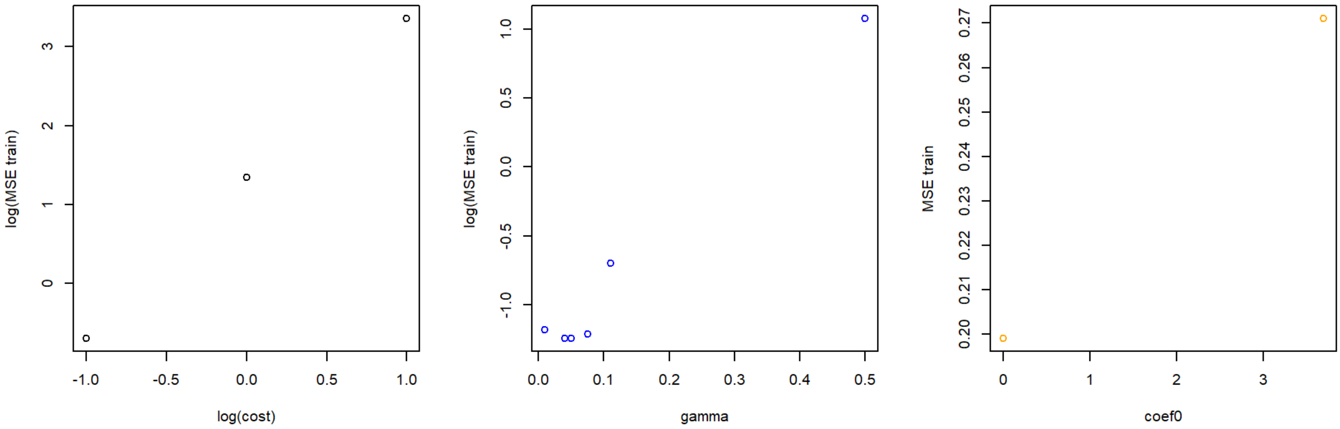

Figure 17 : Sigmoid kernel regression SVM

## Bagging on SVM regression

**Bagging** was used for determination of the confidence limits (with $95\%$ confidence) of the train and test errors, in order to give a better insight on the decision of the best kernel to be used (*Fig.18*). *Fig.18* show that the best kernel to be chosen for the SVM regression would be the *Gaussian kernel* once presents the lower test and train errors.

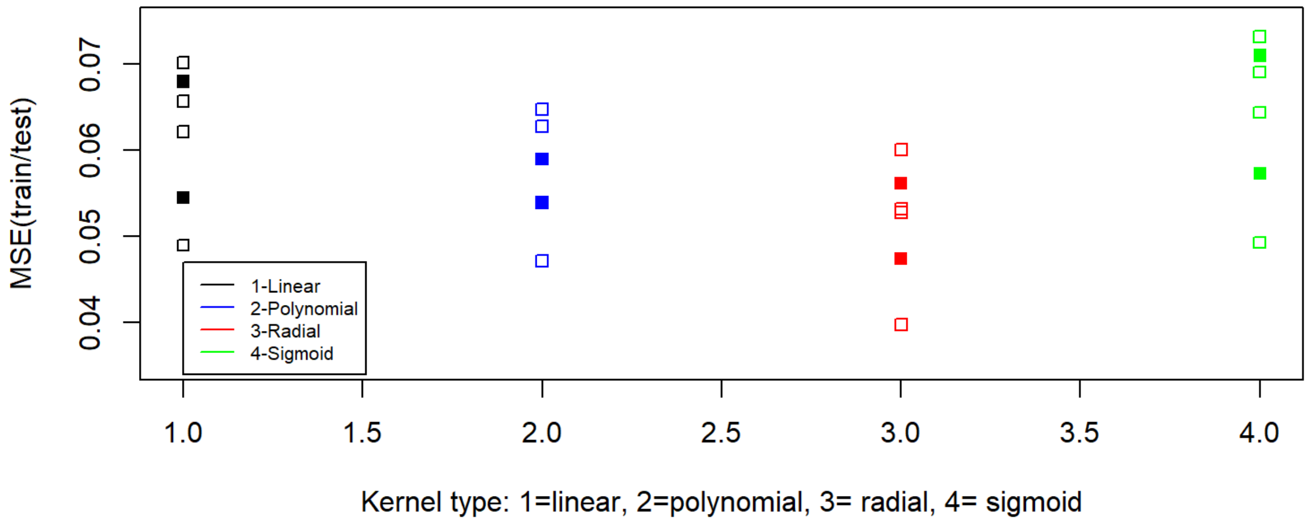

Figure 18 : Train and test errors with confidence limits obtained by bagging of SVM regression. 1=linear, 2=polynomial, 3=radial, 4=sigmoid. Full symbols: mean values, open symbols: 95 % confidence values. Lower values: train MSE; upper values: MSE test

| Kernel | Train MSE | Test MSE |
|---|---|---|
| Linear | 0.054 | 0.068 |
| Polynomial | 0.054 | 0.059 |
| Radial | 0.047 | 0.056 |
| Sigmoid | 0.057 | 0.071 |

Table 6 : Train and test errors for the different kernels (summary) in SVM

# Classification approach with SVM

Using the created discretized dummies, a classification approach for the logAttendance was also considered using SVM. LogAttendance was divided into 4 classes, according to:

| Dummie value | Attendance interval | number of cases |
|---|---|---|
| 0 | [0, 3409.094] | 714 |
| 1 | (3409.094, 4760.91] | 188 |
| 2 | (4760.91, 13917.68] | 253 |
| 3 | (13917.68, 64103] | 351 |

Table 7 : Explanation of the dummie variable of logAttendance

The same procedure was followed for Stadium capacity and Club members variables according to EM analysis. Several kernels were also tested. Best train accuracy obtained was ca. $78\%$, with the Gaussian kernel. *Table 8* shows the confusion matrix for the train and *Table 9* for test classifier. The lower and higher attendances are better classified in the train sample. Attendance dummie variable with value of 1 is the one with the highest error in classification probably because this population is closely intermixed with 0 (see EM analysis, *Fig.6B*). Due to this poorer result we did not try another classification approach.

| | **0** | **1** | **2** | **3** |
|---|---|---|---|---|
| 0 | 470 | 4 | 16 | 0 |
| 1 | 73 | 34 | 21 | 0 |
| 2 | 30 | 4 | 150 | 0 |
| 3 | 0 | 1 | 12 | 235 |

Table 9 : Confusion matrix for the train data set with radial kernel SVM classifier

| | **0** | **1** | **2** | **3** |
|---|---|---|---|---|
| 0 | 470 | 4 | 16 | 0 |
| 1 | 73 | 34 | 21 | 0 |
| 2 | 30 | 4 | 150 | 0 |
| 3 | 0 | 1 | 12 | 235 |

Table 8 : Confusion matrix for the test data set with radial kernel SVM classifier

Regarding expressiveness of SVM, there is a trade-off between a higher complexity SVM model which may over-fit the data and the use of a larger margin (cost) which will incorrectly classify some of the training data in the interest of better generalization. This way, the number of support vectors can range from very few to every single data point if you completely over-fit your data. SVMs can then be a very expressive model, indeed in the regression exercise ca. $80-85\%$ of the training data is a support vector (800-850 supporting vectors out of ca. 1050 training samples) for all the kernels tested meaning that, apart from $20\%$, every point is a support vector and we maybe be nearly overfitting the data and we may have a poor generalization of the model (low variance but high bias). In the classification approach, ca. $50\%$ of the training data is a support vector, implying less overfitting.

# Conclusion:

In this exercise we have compared regression trees, bagging, random forest, boosting and kernel-based approaches (support vector machines) to address the prediction of the football attendance of a match given the host, the visitor, the date and match performance. The accumulated points in a current league were calculated and the data set was also enriched with the capacity of the stadium of the host, the number of the football club members for the host and visitor and the time of the match. *Fig.19* summarizes the regression study taking into account the test error. We can conclude that the more appropriate model to choose would be a trees-based approach either with bagging or random forest. The most relevant variables showed to be the number of club members, both for visitor and host, and stadium capacity.

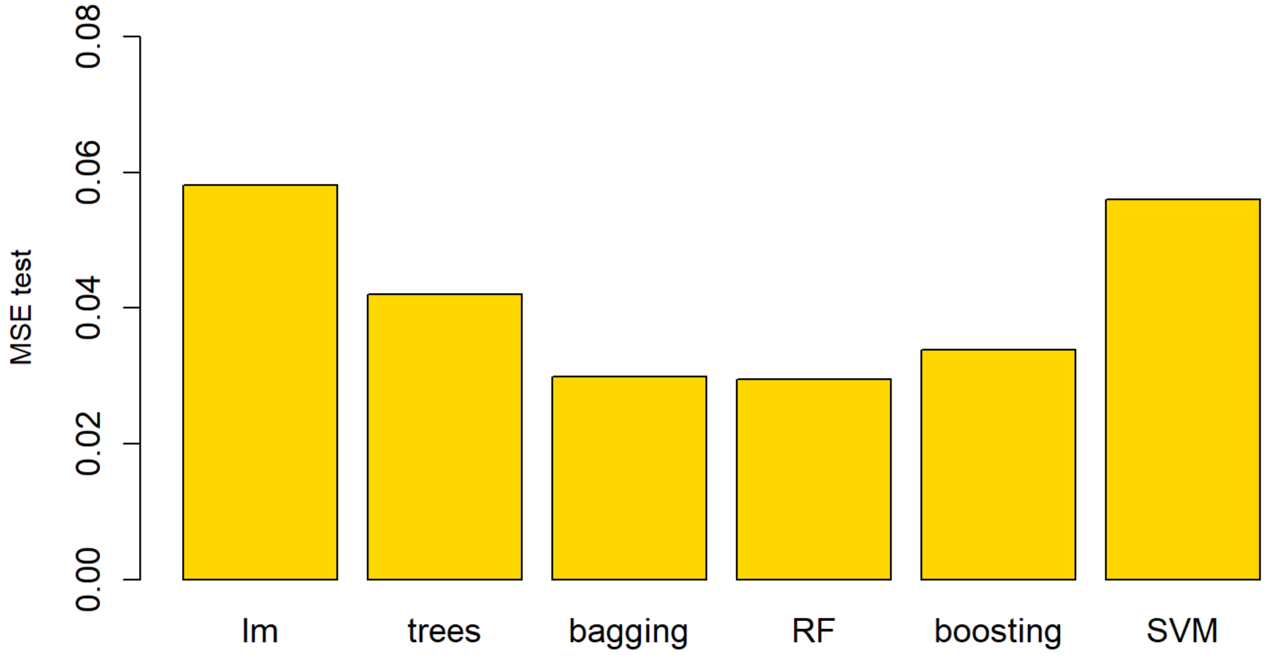

Figure 19 : MSE test errors confrontation

# References:

[1]- Dataset was enriched with time of the match available at: https://www.zerozero.pt/edition_matches.php?fase_in=22682&equipa=0&id_edicao=8838&filtro (https://www.zerozero.pt/edition_matches.php?fase_in=22682&equipa=0&id_edicao=8838&filtro)=&page=1&op=, for the different season. Time of the math when FCP Porto playing was not available. Authors assumed after 18:30. A dummie created with these times considered a value of 0 if time of the match before 18:30 and 1 if after.

[2]- Stadium Capacity in Portugal wer obtained from: https://pt.wikipedia.org/wiki/Lista_de_est%C3%A1dios_de_futebol_de_Portugal (https://pt.wikipedia.org/wiki/Lista_de_est%C3%A1dios_de_futebol_de_Portugal)

[3] - Clube members were mostly obtained from: https://www.transfermarkt.pt/primeira-liga/daten/wettbewerb/PO1 (https://www.transfermarkt.pt/primeira-liga/daten/wettbewerb/PO1), clubs with membership number not available were considered a value of 2000 for the number of members.