Statistical Learning, Homework #2

Alberto Catalano - 257816

27/04/2025

Introduction

In this study, a dataset including 442 observations related to **diabetes research** was analyzed. The primary **objective** was to explore associations between disease progression after one year and various clinical predictors. These included age, sex, body mass index (BMI), average blood pressure (BP), triglyceride levels (TG), glycemia levels (GC), and total cholesterol (TC), which was further broken down into low-density lipoproteins (LDL), high-density lipoproteins (HDL), and the ratio of total cholesterol to HDL (TCH). To identify statistically significant predictors of diabetes progression, three tree-based modeling approaches were employed: **decision trees**, **random forests**, and **boosted regression trees**.

These are the R libraries required for this analysis:

```r
library(tidyverse)  #ggplot2 and more
library(tree)  #for fecision trees
library(randomForest)
library(gbm)  #tree boosting
library(caret)
library(conflicted)  #priority for functions with the same name
conflicts_prefer(dplyr::select())
```

Methods

Before proceeding with the analysis, this section provides a concise overview of the statistical methods employed in this study to evaluate and model the progression of diabetes:

- **Cross-validation (CV)**: K-fold cross-validation is employed as the core evaluation strategy. Through this technique, the data is repeatedly partitioned into training and testing sets, to estimate the model generalization performance. CV is also utilized for the selection of optimal tuning parameters (*tree complexity, mtry, n.trees*) to mitigate overfitting.
- **Decision Tree**: This non-parametric supervised learning method is used for regression to predict the diabetes progression outcome. The dataset is recursively partitioned based on predictor variable values, creating a tree-like structure.
- **Tree pruning**: Applied to the initial decision tree, this technique involves the removal of specific branches identified by CV as having limited predictive value on hold-out data. The tree model is simplified; in this way overfitting to the training data is reduced, and generalizability is often improved.
- **Random Forest**: In this ensemble learning method, a multitude of decision trees is constructed. Randomness is introduced in two ways: each tree is built using a different bootstrap samples of the original data (bagging), and only a random subset of features is considered at each potential split. The final prediction is derived by averaging the predictions from all individual trees, typically resulting in enhanced accuracy and stability compared to a single tree.
- **Boosted regression tree (Tree boosting)**: By this ensemble technique, decision trees are built sequentially. Each new tree is specifically trained to focus on correcting the residual errors remaining from the previously constructed trees. Through this iterative refinement process, highly accurate predictive models are often yielded. The optimal number of trees to include is determined using CV.

Data Exploration

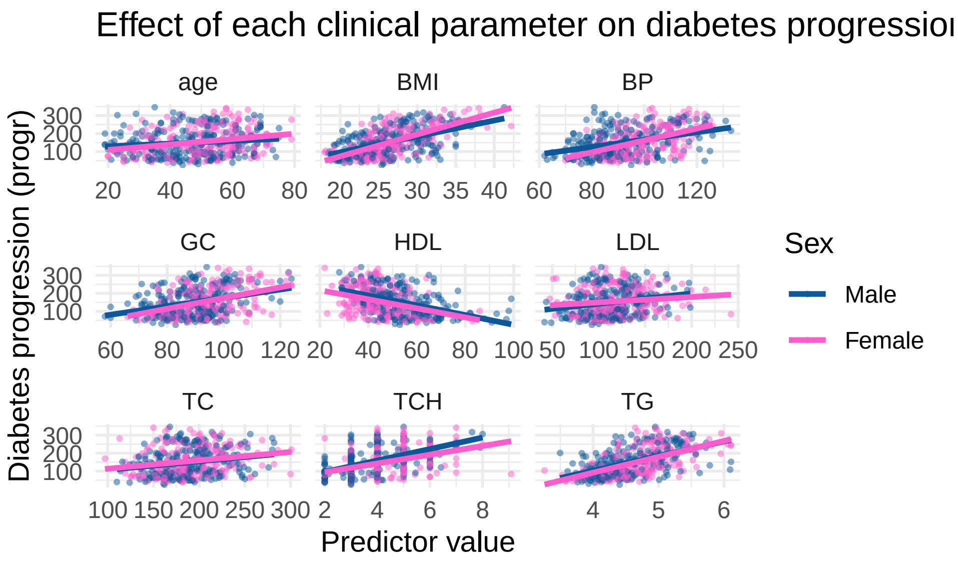
To begin the analysis, the dataset was loaded and duplicated to preserve the original. The summary confirmed correct data types, except for "*sex*" which was factorized. The summary also showed the **absence** of missing values.

```
dataf <- read.csv("db.txt", sep = "\t", header = TRUE)
dataset <- dataf  #copy
dataf$sex <- factor(dataf$sex)
levels(dataf$sex) <- c("Male", "Female")
attach(dataf)
as.tibble(dataf)
```

```
## # A tibble: 442 x 11
##      age sex      BMI    BP    TC   LDL   HDL   TCH    TG    GC progr
##    <int> <fct>  <dbl> <dbl> <int> <dbl> <dbl> <dbl> <dbl> <int> <int>
##  1    59 Female  32.1   101   157  93.2    38  4      4.86    87   151
##  2    48 Male    21.6    87   183 103.     70  3      3.89    69    75
##  3    72 Female  30.5    93   156  93.6    41  4      4.67    85   141
##  4    24 Male    25.3    84   198 131.     40  5      4.89    89   206
##  5    50 Male    23     101   192 125.     52  4      4.29    80   135
##  6    23 Male    22.6    89   139  64.8    61  2      4.19    68    97
##  7    36 Female  22      90   160  99.6    50  3      3.95    82   138
##  8    66 Female  26.2   114   255 185      56  4.55   4.25    92    63
##  9    60 Female  32.1    83   179 119.     42  4      4.48    94   110
## 10    29 Male    30      85   180  93.4    43  4      5.38    88   310
## # i 432 more rows
```

Before the implementation of the decision tree model, an **exploratory scatterplot analysis** was conducted to visualize the relationship between each predictor and diabetes progession, devided by sex. A linear regression line was included in each panel to highlight trends. From this preliminary analysis, some patterns emerged:

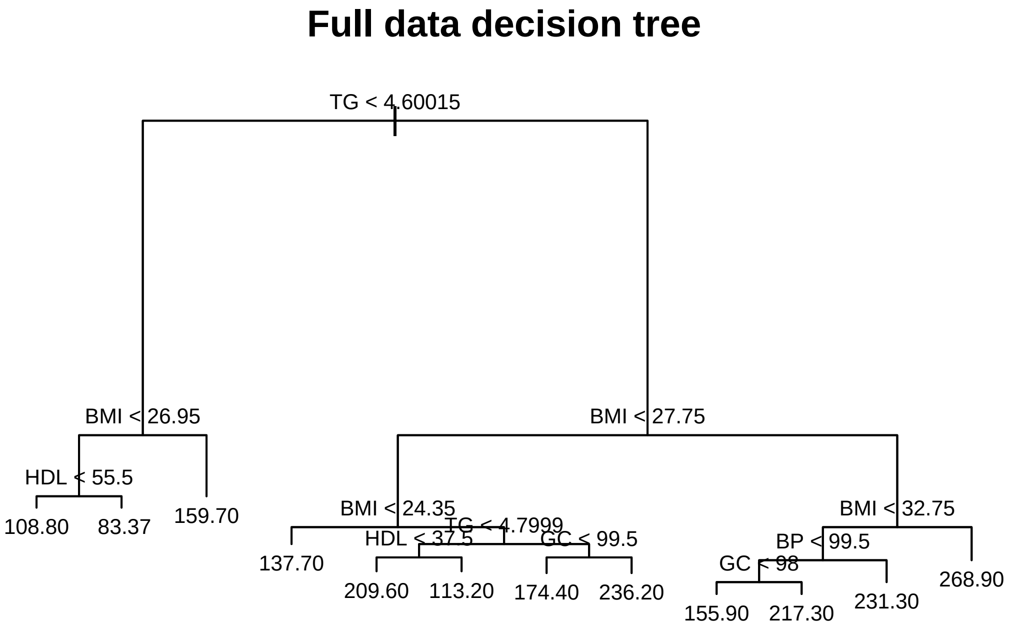
- BMI, glycemia (GC), triglycerides (TG) and total cholesterol ratio (TCH) exhibited strong positive associations with diabetes progression
- HDL showed a strong negative correlation, suggesting a protective effect
- Sex-specific differences in trends were observed for certain predictors



Decision Tree

The regression decision tree is first fitted to the **whole dataset**, resulting in an initial tree composed of 12 terminal nodes. The summary statistics also indicate a residual mean deviance of 2674.

```
diab_tree <- tree(progr ~ ., data = dataf)  #fit the tree on the whole data
plot(diab_tree)
text(diab_tree, pretty = 0, cex = 0.7)  #smaller text
title("Full data decision tree")
```

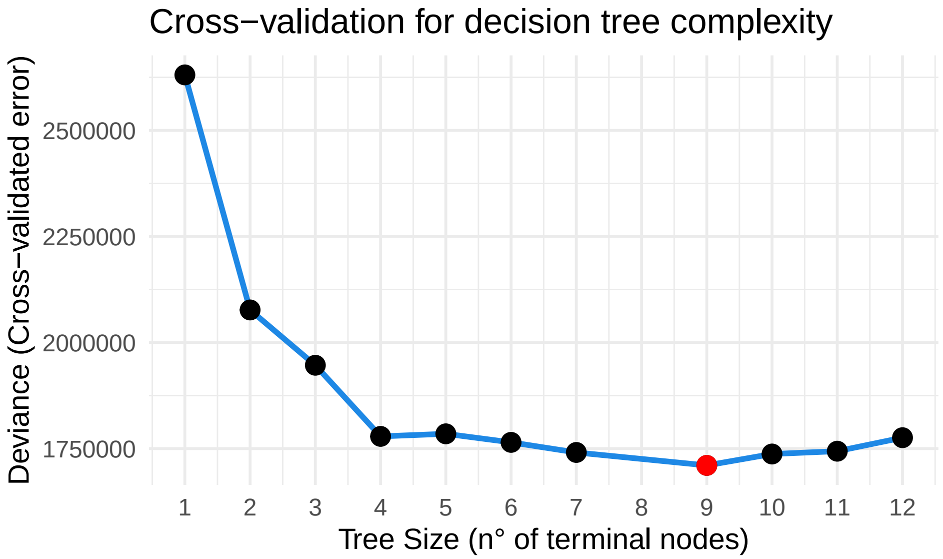


```
summary(diab_tree)
```

```
##
## Regression tree:
## tree(formula = progr ~ ., data = dataf)
## Variables actually used in tree construction:
## [1] "TG"  "BMI" "HDL" "GC"  "BP"
## Number of terminal nodes:  12
## Residual mean deviance:  2674 = 1150000 / 430
## Distribution of residuals:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -140.900  -35.830   -4.805    0.000   33.540  154.100
```

Subsequently, cross-validation of the tree was performed to decide tree complexity. The plot provided below indicates that the optimal number of terminal leaves is 9 (lowest deviance).

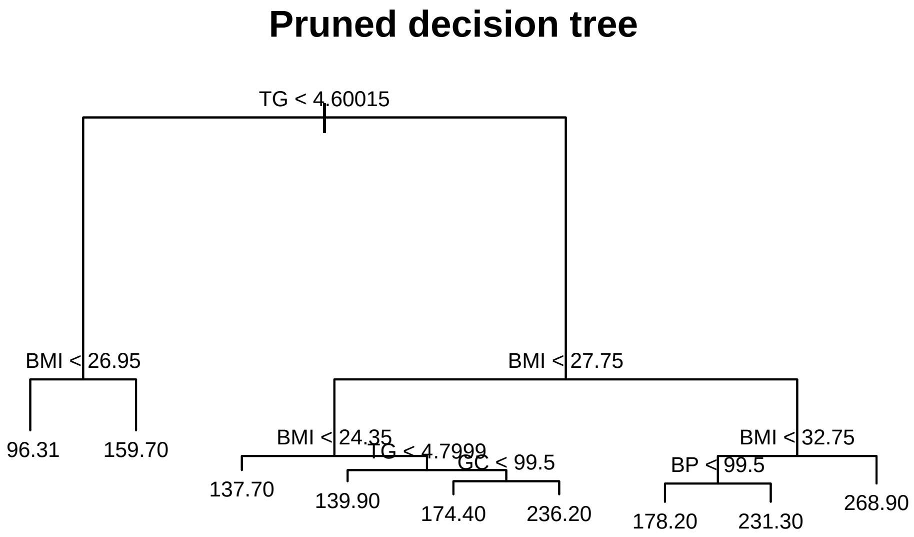
```
set.seed(123)  #for reproducibility
cv_diab_tree <- cv.tree(diab_tree)  #cv
```



Based on the cross-validation results, the tree was **pruned**. Examination of the summary statistics reveals that the residual mean deviance increased to 2864, exceeding the value obtained with the unpruned tree. This increase occurs because the pruned tree possesses a simpler structure, which results in a slightly less precise fit to the training data compared to the potentially overfit, larger, unpruned tree.

Based on the pruned decision tree, Triglycerides (TG) were identified as the primary splitting variable for predicting diabetes progression. Body Mass Index (BMI) served as the key secondary separator. Further stratification within branches was achieved using Glucose (GC) and Blood Pressure (BP) at specific BMI thresholds.

```
optimal_size_diab_tree <- cv_diab_tree$size[which.min(cv_diab_tree$dev)]   #9
pruned_diab_tree <- prune.tree(diab_tree, best = optimal_size_diab_tree)

plot(pruned_diab_tree)
text(pruned_diab_tree, pretty = 0, cex = 0.7)
title("Pruned decision tree")
```



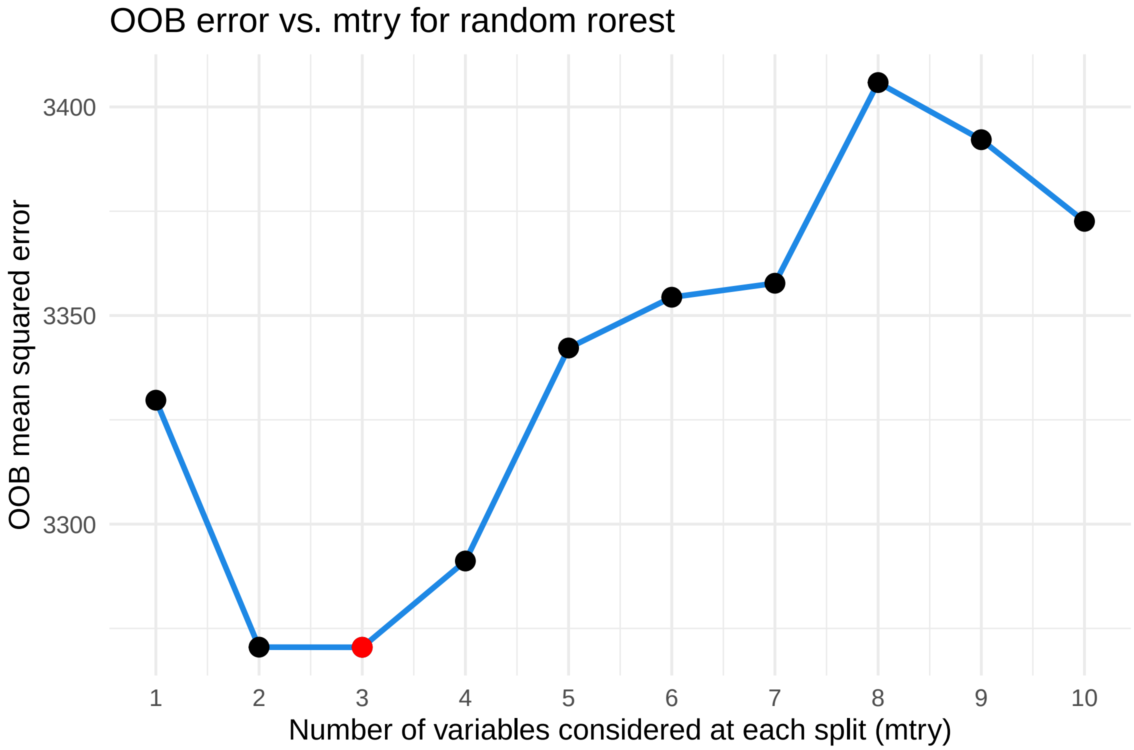
```
summary(pruned_diab_tree)
```

```
##
## Regression tree:
## snip.tree(tree = diab_tree, nodes = c(4L, 28L, 26L))
## Variables actually used in tree construction:
## [1] "TG"  "BMI" "GC"  "BP"
## Number of terminal nodes:  9
## Residual mean deviance:  2864 = 1240000 / 433
## Distribution of residuals:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -140.900  -37.310   -6.325    0.000   34.150  156.700
```

Random forest

Following the pruned tree, a random forest model was fitted to the dataset, as previously introduced. The **mtry parameter**, which represents the number of variables randomly sampled as candidates at each split, required tuning. This tuning was performed using cross-validation, evaluating the **Out-of-Bag (OOB) error** for different mtry values. The plot below illustrates the relationship between mtry and the corresponding OOB error. The same number can also be obtained by the standard formula: *mtry = n° predictors / 3.*

```
set.seed(123)
n_pred <- ncol(dataf) - 1
mtry_vals <- 1:n_pred  #mtry tuning
oob_errors <- numeric(length(mtry_vals))

for (i in mtry_vals) {
    # forest creation
    diab_rf_temp <- randomForest(progr ~ ., data = dataf, mtry = i, ntree = 250,
        importance = FALSE)
    oob_errors[i] <- diab_rf_temp$mse[diab_rf_temp$ntree]
}

best_mtry <- which.min(oob_errors)  #3
```



Based on the results shown in the plot, the final random forest model was fitted utilizing this optimal mtry value. For this final model, the number of trees was set to 500, a value frequently employed in practice for robust model generation. The percentage of data variance explained by the model is 45.88%.

```
set.seed(123)
diab_random_forest <- randomForest(progr ~ ., data = dataf, mtry = best_mtry, ntree = 500,
    importance = TRUE)
diab_random_forest  #summary of final rf model
```

```
##
## Call:
##  randomForest(formula = progr ~ ., data = dataf, mtry = best_mtry,      ntree = 500, importance = TRUE)
##                Type of random forest: regression
##                      Number of trees: 500
## No. of variables tried at each split: 3
##
##           Mean of squared residuals: 3209.478
##                     % Var explained: 45.88
```

Following the generation of the final random forest model, variable importance was assessed. Although several methods exist for representing it, an **ordered table** format based on the percentage increase in Mean Squared Error (%IncMSE) was chosen for presentation. According to this metric, the largest increases in MSE are associated with the variables TG and BMI, indicating their high predictive importance (both exhibiting %IncMSE values exceeding 30).

```
importance_rf <- importance(diab_random_forest)
knitr::kable(importance_rf[order(importance_rf[, "%IncMSE"], decreasing = TRUE),
    ])
```

	%IncMSE	IncNodePurity
TG	35.097372	550228.60
BMI	33.640745	563230.43
BP	17.292044	299767.00
TCH	11.928423	181484.33
HDL	9.450693	210197.98
GC	8.106341	207275.44
LDL	8.080853	158275.73
sex	7.935234	32914.08
TC	7.356326	145918.83
age	2.041153	143317.46

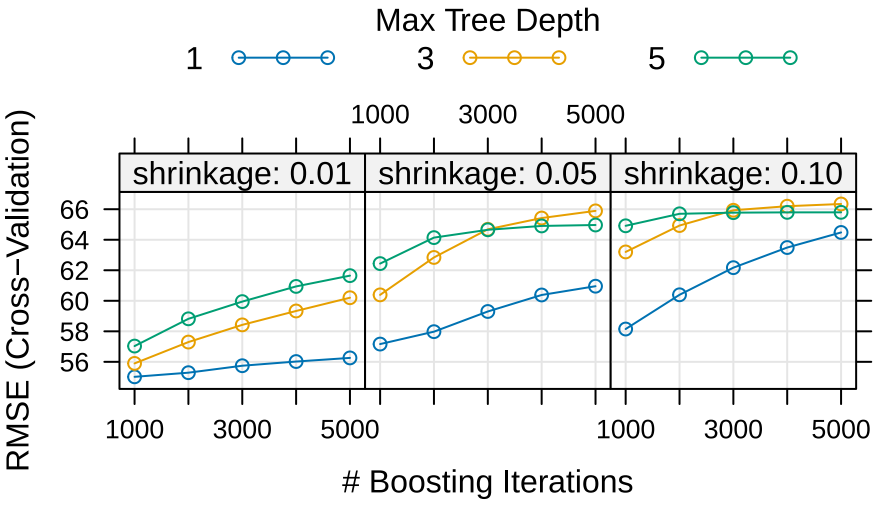
```
# varImpPlot(diab_random_forest, main = 'Variable Importance (Random Forest)'
```

Boosted regression tree

The final model implemented was a **boosted regression tree**. Firstly, cross validation was used to tune key parameters such as the number of trees, interaction depth and shrinkage, for each parameter values with lowest RMSE were selected.

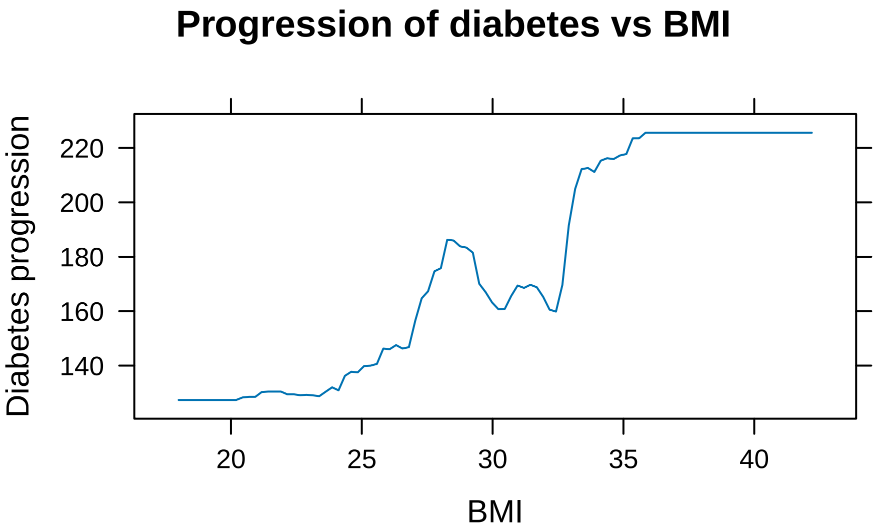
```
set.seed(123)
boosted_control <- trainControl(method = "cv", number = 10)
boosted_par_tuning <- expand.grid(
  n.trees = c(1000,2000,3000,4000,5000), #5000
  interaction.depth = c(1,3,5), #1
  shrinkage = c(0.01,0.05,0.1), #0.01
  n.minobsinnode = 10)
```

```
diab_gbm_tuned <- train(
  progr ~ .,  data = dataf, method = "gbm", distribution = "gaussian",
  trControl = boosted_control,
  tuneGrid = boosted_par_tuning, verbose = F)

plot(diab_gbm_tuned)
```



The provided partial dependence plot illustrates the estimated effect of BMI on predicted diabetes progression, holding all other variables constant. Generally, diabetes progression tends to **increase** with higher BMI. The progression shows a sharp rise around a BMI of 33, before reaching a plateau at higher BMI.

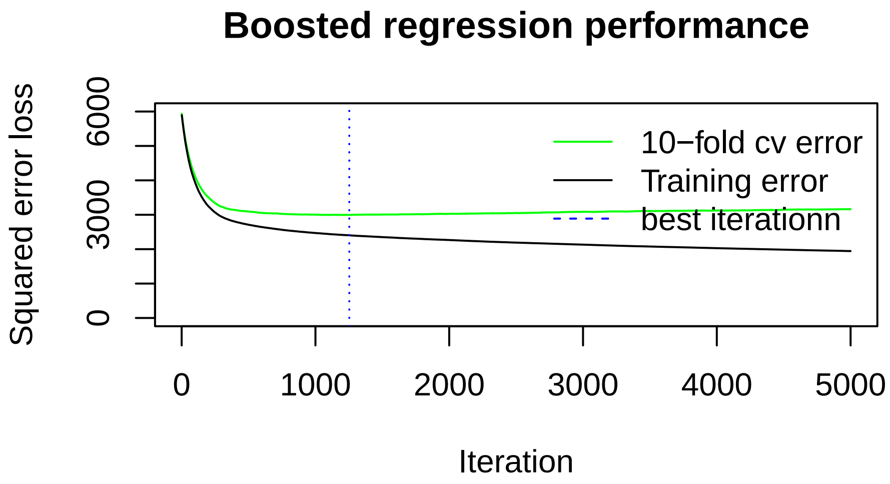
```
set.seed(123)
diab_gbm <- gbm(progr ~ ., data = dataf, distribution = "gaussian", n.trees = 5000,
    interaction.depth = 1, shrinkage = 0.01, cv.folds = 10, verbose = FALSE)

plot(diab_gbm, i = "BMI", ylab = "Diabetes progression", main = "Progression of diabetes vs BMI")
```



This plot displays the training error and 10-fold cross-validation error against the number of boosting iterations for the model. While training error decreases continually, the CV error reaches a minimum at approximately **1253 iterations** (dashed line), indicating the optimal number of trees before overfitting begins.

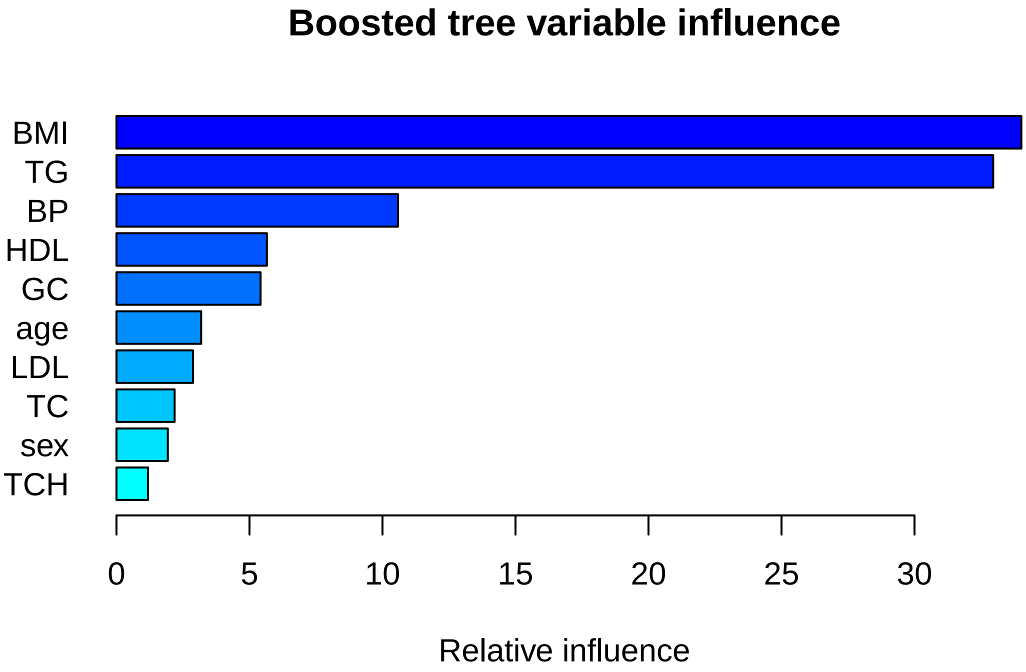
```
set.seed(123)
best_treenum_cv <- gbm.perf(diab_gbm, method = "cv", plot.it = FALSE)  #1253
best_treenum_cv
```

```
## [1] 1253
```



Following the determination of the optimal number of trees, variable influence was examined. The summary indicates that TG and BMI are again identified as the most influential predictors.

```
summary(diab_gbm, n.trees = best_treenum_cv, main = "Boosted tree variable influence",
    las = 1)
```



```
##     var   rel.inf
## BMI BMI 34.017988
## TG   TG 32.958332
## BP   BP 10.583864
## HDL HDL  5.654691
## GC   GC  5.417244
## age age  3.185046
## LDL LDL  2.877446
## TC   TC  2.186832
## sex sex  1.930619
## TCH TCH  1.187937
```

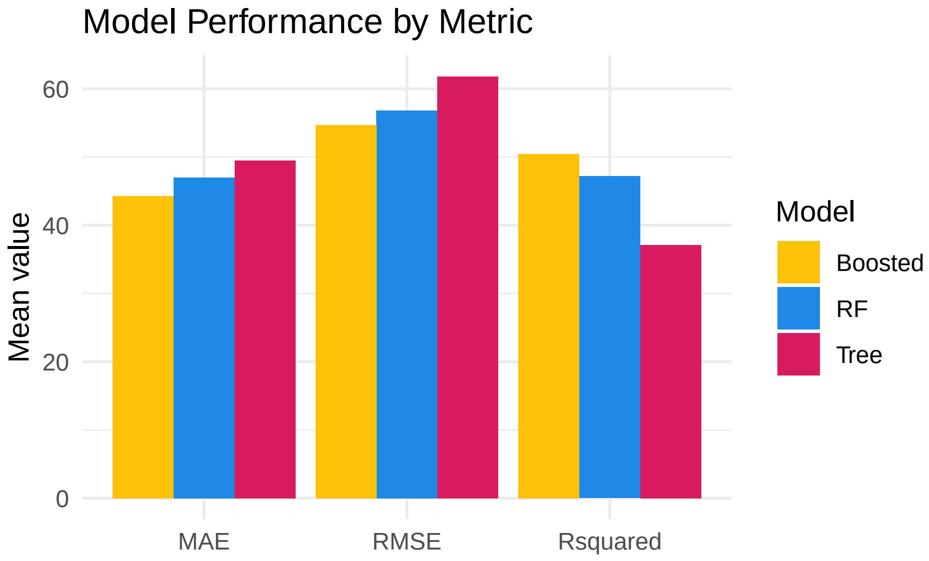
Performance of the three models

Model performance was compared using a 10-fold nested cross-validation approach. The outer 10 folds were utilized for evaluating the final predictive performance. Within each outer training fold, model complexity wes re-optimized independently: internal cross-validation was employed for decision trees and boosting models, while the Out-of-Bag error was used for random forests. This nested methodology ensures an unbiased performance assessment by preventing the outer test data from influencing complexity selection.

```
set.seed(123)
ctrl <- trainControl(method = "cv", number = 10)  #10-fold cv

ctrl_diab_pruned_tree <- train(progr ~ ., data = dataf, method = "rpart", trControl = ctrl,
    tuneLength = 10)
ctrl_diab_random_forest <- train(progr ~ ., data = dataf, method = "rf", trControl = ctrl,
    tuneLength = 5, importance = T)
ctrl_diab_boosted <- train(progr ~ ., data = dataf, method = "gbm", trControl = ctrl,
    verbose = FALSE, tuneLength = 5)

results <- resamples(list(Tree = ctrl_diab_pruned_tree, RF = ctrl_diab_random_forest,
    Boosted = ctrl_diab_boosted))
summary(results)
```

```
##
## Call:
## summary.resamples(object = results)
##
## Models: Tree, RF, Boosted
## Number of resamples: 10
##
## MAE
##             Min.  1st Qu.   Median     Mean  3rd Qu.     Max. NA's
## Tree    38.54036 49.19738 50.61963 49.48875 51.46885 56.09596    0
## RF      41.94933 43.14121 47.93592 46.96492 49.39536 53.75345    0
## Boosted 37.82140 38.60741 46.18319 44.30927 48.37411 50.93345    0
##
## RMSE
##             Min.  1st Qu.   Median     Mean  3rd Qu.     Max. NA's
## Tree    50.56258 60.52763 62.39552 61.79852 64.65509 67.01670    0
## RF      50.28694 52.27309 57.78621 56.80963 61.18088 62.50274    0
## Boosted 46.56856 48.56826 55.73216 54.68502 60.18369 62.59798    0
##
## Rsquared
##              Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## Tree    0.2756637 0.3130744 0.3577871 0.3708766 0.3902898 0.5885822    0
## RF      0.3052472 0.4125886 0.5051372 0.4717574 0.5333699 0.5495263    0
## Boosted 0.3426313 0.3717247 0.5217970 0.5043640 0.6292599 0.6555594    0
```

It can be observed from the plot that the boosted model generally exhibit lower error metrics (MAE, RMSE) and higher R-squared values compared to the pruned tree and random forest models.



Conclusions

In this analysis, three tree-based regression methods: pruned decision trees, random forests, and boosted trees were employed to predict diabetes progression based on clinical data. A 10-fold cross-validation procedure was utilized for evaluation and incorporating parameter re-optimization within each training fold to facilitate unbiased performance comparisons.

While the pruned decision tree provided a simple and interpretable structure, its predictive accuracy was lower compared to the ensemble methods. Random forests improved performance by leveraging bootstrap aggregation and feature randomness, while **boosted trees achieved superior overall predictive performance** according to the evaluation metrics. Boosting' sequential learning and residual correction mechanism enabled the capture of complex, non-linear relationships within the data, resulting in the lowest error metrics (MAE and RMSE) and the highest R-squared values among the models evaluated.

Regarding variable importance, all models consistently identified **BMI** and **triglyceride levels (TG)** as the most influential predictors of diabetes progression. These were followed by blood pressure (BP) and glycemia (GC), indicating the importance of metabolic and cardiovascular indicators in disease development. Notably, HDL demonstrated a potentially protective association, aligning with clinical understanding of its role. Overall, while decision trees offer advantages in interpretability ensemble methods, particularly gradient boosting, provide superior predictive accuracy and potentially deeper insights into the factors driving disease progression.