R Exercise: Ensemble Regression

Pilsung Kang

School of Industrial Management Engineering

Korea University

R Exercise: Ensemble Regression

- Data Set: Toyota Corolla Used Car Price Prediction

Variable	Description	Variable	Description
		Guarantee_Period	Guarantee period in months
		ABS	Anti-Lock Brake System (Yes=1, No=0)
Price	Offer Price in EUROs	Airbag_1	Driver_Airbag (Yes=1, No=0)
Age_08_04	Age in months as in August 2004	Airbag_2	Passenger Airbag (Yes=1, No=0)
Mfg_Month	Manufacturing month (1-12)	Airco	Airconditioning (Yes=1, No=0)
Mfg_Year	Manufacturing Year	Automatic_airco	Automatic Airconditioning (Yes=1, No=0)
KM	Accumulated Kilometers on odometer	Boardcomputer	Boardcomputer (Yes=1, No=0)
Fuel_Type	Fuel Type (Petrol, Diesel, CNG)	CD_Player	CD Player (Yes=1, No=0)
HP	Horse Power	Central_Lock	Central Lock (Yes=1, No=0)
Met_Color	Metallic Color? (Yes=1, No=0)	Powered_Windows	Powered Windows (Yes=1, No=0)
Automatic	Automatic ((Yes=1, No=0)	Power_Steering	Power Steering (Yes=1, No=0)
CC	Cylinder Volume in cubic centimeters	Radio	Radio (Yes=1, No=0)
Doors	Number of doors	Mistlamps	Mistlamps (Yes=1, No=0)
Cylinders	Number of cylinders	Sport_Model	Sport Model (Yes=1, No=0)
Gears	Number of gear positions	Backseat_Divider	Backseat Divider (Yes=1, No=0)
Quarterly_Tax	Quarterly road tax in EUROs	Metallic_Rim	Metallic Rim (Yes=1, No=0)
Weight	Weight in Kilograms	Radio_cassette	Radio Cassette (Yes=1, No=0)
Mfr_Guarantee	Within Manufacturer's Guarantee period (Yes=1, No=0)	Parking_Assistant	Parking assistance system (Yes=1, No=0)
BOVAG_Guarantee	BOVAG (Dutch dealer network) Guarantee (Yes=1, No=0)	Tow_Bar	Tow Bar (Yes=1, No=0)

R Exercise: Ensemble Regression

- Purpose
  - ✓ Compare the regression performances of single models and ensemble models
    - Single classifier: MLR with forward variable selection, ANN
    - Ensemble classifier: Bagging with ANN, Random Forests, Gradient Boosting Machine (GBM) with Stump Tree
  - ✓ Experimental Settings
    - Use the same dataset for training and test
    - Use the best parameter found in the previous R exercise for ANN
    - Use the same parameter for Bagging with ANN

R Exercise: Ensemble Regression

- Create a performance evaluation function

```
# Part 1: Regression with Single Model ----------------------------------------
# Performance Evaluation Function ---------------------------------------------
perf_eval <- function(target, haty){
    # Mean squared error (MSE)
    MSE <- mean((target - haty)^2)
    # Root mean squared error (RMSE)
    RMSE <- sqrt(MSE)
    # Mean absolute error
    MAE <- mean(abs(target-haty))
    # Mean absolute percentage error
    MAPE <- mean(abs((target-haty)/target))
    return(c(MSE, RMSE, MAE, MAPE))
}

perf_table <- matrix(0, nrow = 5, ncol = 4)
rownames(perf_table) <- c("MLR", "ANN", "Bagging ANN", "GBM", "Random Forests")
colnames(perf_table) <- c("MSE", "RMSE", "MAE", "MAPE")
```

R Exercise: Ensemble Regression

- Load the dataset and randomly split the dataset into training (70%) and test (30%)

```
# Read data file
corolla <- read.csv("Toyota_Corolla.csv")

# Split the data into the training/validation sets
set.seed(12345)
trn_idx <- sample(1:dim(corolla)[1], round(0.7*dim(corolla)[1]))
trn_data <- corolla[trn_idx,]
tst_data <- corolla[-trn_idx,]
```

✓ Use the same index for training/test data split for all models

R Exercise: Ensemble Regression

- Single model 1: MLR

```
# Model 1: Multiple Linear Regression with forward variable selection------------
# Variable selection: Forward selection
# Upperbound formula
tmp_x <- paste(colnames(trn_data)[-1], collapse=" + ")
tmp_xy <- paste("Price ~ ", tmp_x, collapse = "")
as.formula(tmp_xy)

MLR_model <- step(lm(Price ~ 1, data = trn_data),
                  scope = list(upper = as.formula(tmp_xy), lower = Price ~ 1),
                  direction="forward", trace=1)

summary(MLR_model)
```

✓ Use step( ) function to conduct forward variable selection

R Exercise: Ensemble Regression

- Single model 1: MLR

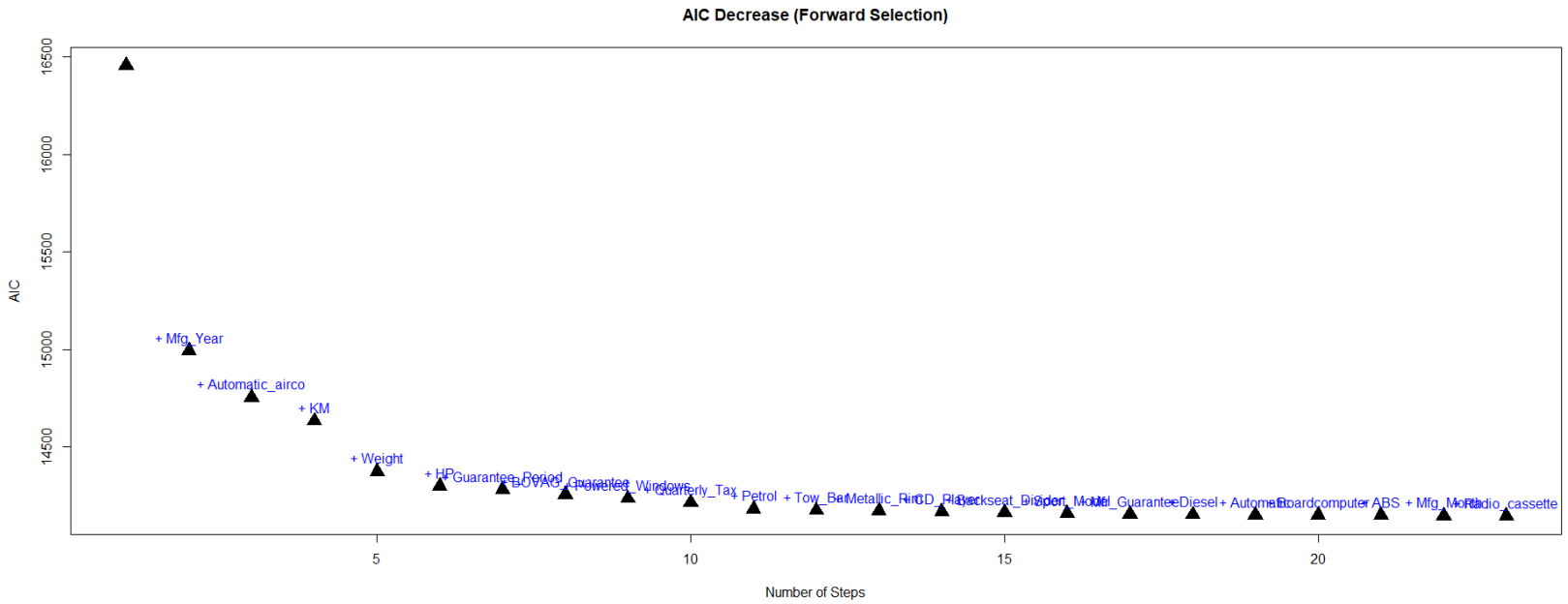
```
Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)     -2.870e+06  9.915e+04 -28.949  < 2e-16 ***
Mfg_Year         1.432e+03  4.975e+01  28.776  < 2e-16 ***
Automatic_airco  2.216e+03  1.975e+02  11.222  < 2e-16 ***
KM              -1.717e-02  1.341e-03 -12.803  < 2e-16 ***
Weight           1.205e+01  1.296e+00   9.299  < 2e-16 ***
HP               2.060e+01  3.424e+00   6.018 2.49e-09 ***
Guarantee_Period 7.961e+01  1.500e+01   5.309 1.36e-07 ***
BOVAG_Guarantee  4.878e+02  1.351e+02   3.611  0.00032 ***
Powered_Windows  3.223e+02  8.139e+01   3.960 8.05e-05 ***
Quarterly_Tax    1.517e+01  1.849e+00   8.208 7.03e-16 ***
Petrol           2.314e+03  3.759e+02   6.156 1.09e-09 ***
Tow_Bar         -2.181e+02  8.284e+01  -2.633  0.00859 **
Metallic_Rim     2.415e+02  9.668e+01   2.498  0.01267 *
CD_Player        2.482e+02  1.067e+02   2.327  0.02015 *
Backseat_Divider -3.576e+02  1.238e+02  -2.888  0.00397 **
Sport_Model      2.364e+02  9.092e+01   2.600  0.00947 **
Mfr_Guarantee    1.956e+02  7.801e+01   2.507  0.01233 *
Diesel           9.928e+02  3.677e+02   2.700  0.00705 **
Automatic        2.889e+02  1.530e+02   1.889  0.05924 .
Boardcomputer   -2.784e+02  1.272e+02  -2.189  0.02884 *
ABS             -2.111e+02  1.048e+02  -2.015  0.04423 *
Mfg_Month        2.344e+01  1.098e+01   2.135  0.03297 *
Radio_cassette  -2.059e+02  1.083e+02  -1.902  0.05751 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1126 on 982 degrees of freedom
Multiple R-squared:  0.904,     Adjusted R-squared:  0.9018
F-statistic: 420.2 on 22 and 982 DF,  p-value: < 2.2e-16
```

- 22 variables are selected from a total of 35 variables
- Adjusted R^2: 0.9018 – the 90.18% of variance is explained by the regression model
- There is a strong linear relationship between the input variables and the target variable

R Exercise: Ensemble Regression

- Single model 1: MLR

```
# Show the selected variable in each step
MLR_model$anova$Step
MLR_model$anova$AIC

# AIC decrease according the the selected variable
plot(MLR_model$anova$AIC, pch = 17, cex=2, main = "AIC Decrease (Forward
Selection)", xlab = "Number of Steps", ylab = "AIC")

text(MLR_model$anova$AIC, MLR_model$anova$Step, cex=1, pos=3, col="blue")
```



R Exercise: Ensemble Regression

- Single model 1: MLR

```
MLR_haty <- predict(MLR_model, newdata = tst_data)

perf_table[1,] <- perf_eval(tst_data$Price, MLR_haty)
perf_table
```

	MSE	RMSE	MAE	AMPE
MLR	1,539,502	1240.77	880.86	8.90%
ANN				
Bagging ANN				
GBM				
Random Forests				

R Exercise: Ensemble Regression

- Single model 2: ANN

```
# Model 2: Artificial Neural Network ----------------------------------------
# nnet package install
install.packages("nnet", dependencies = TRUE)
library(nnet)
corolla_input <- corolla[,-1]
corolla_target <- corolla[,1]

# Data normalization
corolla_input_scaled <- scale(corolla_input, center = TRUE, scale = TRUE)

# Input/Target configuration
ANN_trn_input <- corolla_input_scaled[trn_idx,]
ANN_trn_target <- corolla_target[trn_idx]
ANN_tst_input <- corolla_input_scaled[-trn_idx,]
ANN_tst_target <- corolla_target[-trn_idx]
```

R Exercise: Ensemble Regression

- Single model 2: ANN

```
# Trainin ANN
ANN_model <- nnet(ANN_trn_input, ANN_trn_target, size = 10,
                  linout = TRUE, decay = 5e-4, maxit = 500)

# Performance evaluation
ANN_haty <- predict(ANN_model, ANN_tst_input)
perf_table[2,] <- perf_eval(tst_data$Price, ANN_haty)
perf_table
```

✓ Set the number of hidden nodes to 10 (it can be optimized through cross validation process)

✓ Note that "linout = TRUE" option is used for regression

	MSE	RMSE	MAE	AMPE
MLR	1,539,502	1240.77	880.86	8.90%
ANN	3,286,328	1,812.82	1,288.35	12.62%
Bagging ANN				
GBM				
Random Forests				

R Exercise: Ensemble Regression

- Ensemble model 1: Bagging ANN

```
# Bagging Training
ptm <- proc.time()
Bagging_ANN_model <- avNNet(ANN_trn_input, ANN_trn_target, size = 10,
                            decay = 5e-4, linout = TRUE, repeats = 100, bag = TRUE,
                            allowParallel = TRUE, trace = TRUE)
Bagging_Time <- proc.time() - ptm
Bagging_Time
```

✓ Set the same number of hidden nodes (10) with the single model

✓ Use 100 bootstraps

	MSE	RMSE	MAE	AMPE
MLR	1,539,502	1240.77	880.86	8.90%
ANN	3,286,328	1,812.82	1,288.35	12.62%
Bagging ANN	2,810,978	1,676.60	1,137.86	10.83%
GBM				
Random Forests				

R Exercise: Ensemble Regression

- Ensemble model 2: GBM

```
# Training the GBM
ptm <- proc.time()
GBM_model <- gbm.fit(trn_data[,-1], trn_data[,1], distribution = "gaussian",
                     n.trees = 1000, shrinkage = 0.02, bag.fraction = 0.8,
                     nTrain = 1000)
GBM_Time <- proc.time() - ptm
GBM_Time
summary(GBM_model)
```

✓ Use [distribution = "gaussian"] option for regression

R Exercise: Ensemble Regression

- Ensemble model 2: GBM

```
> summary(GBM_model)
                             var      rel.inf
Age_08_04               Age_08_04 65.963054014
Mfg_Year                 Mfg_Year 17.164856889
Weight                     Weight  4.432563011
KM                             KM  3.865313552
Automatic_airco   Automatic_airco  2.580794435
HP                             HP  2.320149525
Quarterly_Tax       Quarterly_Tax  1.159423763
Airco                       Airco  0.949990225
Mfr_Guarantee       Mfr_Guarantee  0.304297238
Mistlamps               Mistlamps  0.258035552
BOVAG_Guarantee   BOVAG_Guarantee  0.240231857
Guarantee_Period Guarantee_Period  0.228343372
Automatic               Automatic  0.157408595
Powered_Windows   Powered_Windows  0.145473497
Doors                       Doors  0.119571882
Metallic_Rim         Metallic_Rim  0.066544933
cc                             cc  0.017055262
CD_Player               CD_Player  0.011439660
Sport_Model           Sport_Model  0.011229340
Mfg_Month               Mfg_Month  0.004223397
Met_Color               Met_Color  0.000000000
Gears                       Gears  0.000000000
ABS                           ABS  0.000000000
Airbag_1                 Airbag_1  0.000000000
Airbag_2                 Airbag_2  0.000000000
Boardcomputer       Boardcomputer  0.000000000
Central_Lock         Central_Lock  0.000000000
Power_Steering     Power_Steering  0.000000000
Radio                       Radio  0.000000000
Backseat_Divider Backseat_Divider  0.000000000
Radio_cassette     Radio_cassette  0.000000000
Tow_Bar                   Tow_Bar  0.000000000
Petrol                     Petrol  0.000000000
Diesel                     Diesel  0.000000000
CNG                           CNG  0.000000000
```

R Exercise: Ensemble Regression

- Ensemble model 2: GBM

```
# Training the GBM
ptm <- proc.time()
GBM_model <- gbm.fit(trn_data[,-1], trn_data[,1], distribution = "gaussian",
                     n.trees = 1000, shrinkage = 0.02, bag.fraction = 0.8,
                     nTrain = 1000)
GBM_Time <- proc.time() - ptm
GBM_Time
summary(GBM_model)
```

	MSE	RMSE	MAE	AMPE
MLR	1,539,502	1240.77	880.86	8.90%
ANN	3,286,328	1,812.82	1,288.35	12.62%
Bagging ANN	2,810,978	1,676.60	1,137.86	10.83%
GBM	2,353,836	1,534.22	1,088.47	11.12%
Random Forests				

R Exercise: Ensemble Regression

- Ensemble model 3: Random Forest

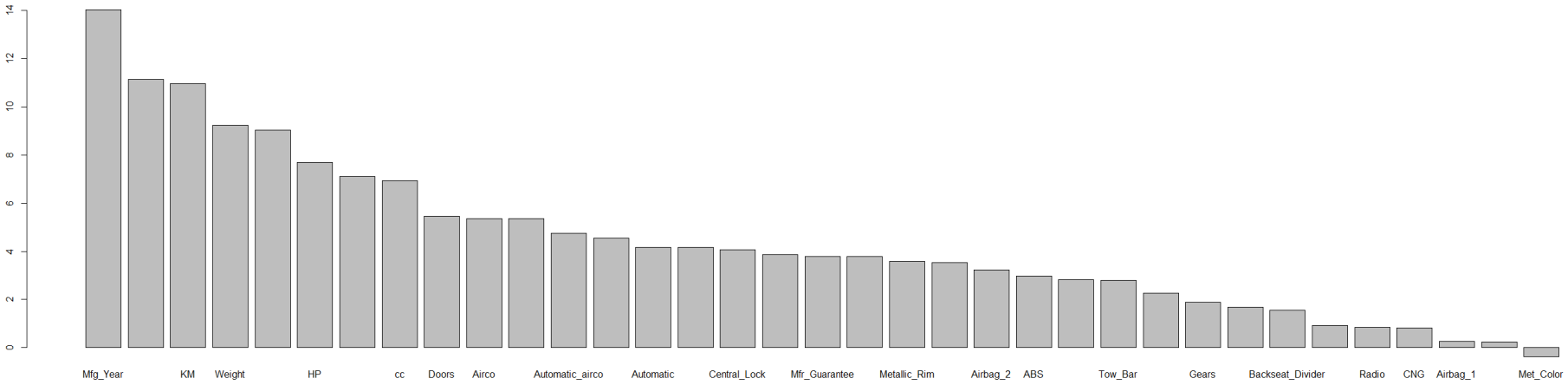
```
# Training the Random Forest
ptm <- proc.time()
RF_model <- randomForest(trn_data[,-1], trn_data[,1], ntree = 100,
                         importance = TRUE, do.trace = TRUE)
RF_Time <- proc.time() - ptm
RF_Time

# Check the result
print(RF_model)
plot(RF_model)

# Variable importance
Var_imp <- importance(RF_model)
barplot(Var_imp[order(Var_imp[,1], decreasing = TRUE),1])
```

✓ randomForest( ) function

- If the variable type of the second argument is factor: classification model
- If the variable type of the second argument is numeric: regression model

R Exercise: Ensemble Regression

- Ensemble model 3: Random Forest
  - ✓ Variable importance



R Exercise: Ensemble Regression

- Ensemble model 3: Random Forest
  - ✓ Variable importance of RF and GBM

```
> summary(GBM_model)
                             var      rel.inf
Age_08_04               Age_08_04 59.617320526
Mfg_Year                 Mfg_Year 23.432509028
Weight                     Weight  4.466906834
KM                             KM  3.927265378
Automatic_airco   Automatic_airco  2.683857864
HP                             HP  2.272759360
Quarterly_Tax       Quarterly_Tax  1.160952140
Airco                       Airco  0.848673170
Mistlamps               Mistlamps  0.324353196
Mfr_Guarantee       Mfr_Guarantee  0.267628115
BOVAG_Guarantee   BOVAG_Guarantee  0.246253885
Guarantee_Period Guarantee_Period  0.242808358
Automatic               Automatic  0.161310896
Doors                       Doors  0.118199214
Powered_Windows   Powered_Windows  0.109775320
Metallic_Rim         Metallic_Rim  0.072000314
cc                             cc  0.024786843
CD_Player             CD_Player  0.015270228
Sport_Model         Sport_Model  0.005054391
Mfg_Month             Mfg_Month  0.002314941
Met_Color             Met_Color  0.000000000
Gears                     Gears  0.000000000
ABS                         ABS  0.000000000
Airbag_1               Airbag_1  0.000000000
Airbag_2               Airbag_2  0.000000000
Boardcomputer     Boardcomputer  0.000000000
Central_Lock       Central_Lock  0.000000000
Power_Steering   Power_Steering  0.000000000
Radio                     Radio  0.000000000
Backseat_Divider Backseat_Divider  0.000000000
Radio_cassette   Radio_cassette  0.000000000
Tow_Bar                 Tow_Bar  0.000000000
Petrol                   Petrol  0.000000000
Diesel                   Diesel  0.000000000
CNG                         CNG  0.000000000
```

```
> Var_imp
                  %IncMSE IncNodePurity
Age_08_04       11.1504523    3707150467
Mfg_Month        2.2755854      98079113
Mfg_Year        14.0241485    4331523755
KM              10.9540156     827446727
HP               7.6907742     205065089
Met_Color       -0.3599107      27382527
Automatic        4.1801747      14215308
cc               6.9311401     122906195
Doors            5.4717214      42427626
Gears            1.8847491       8419435
Quarterly_Tax    9.0277874     229836967
Weight           9.2322720     687946223
Mfr_Guarantee    3.7905260      37281923
BOVAG_Guarantee  1.6816841      27582053
Guarantee_Period 2.8339027      22958574
ABS              2.9816172      26164959
Airbag_1         0.2547517       4980504
Airbag_2         3.2369906      24120154
Airco            5.3700727      73424028
Automatic_airco  4.7557439     672751814
Boardcomputer    7.1153288    1390684441
CD_Player        4.1612230     107575452
Central_Lock     4.0662990      55075686
Powered_Windows  3.8707544      76151464
Power_Steering   0.9105631       2609527
Radio            0.8541061      12091741
Mistlamps        4.5580650      39070876
Sport_Model      5.3639339     139673995
Backseat_Divider 1.5663182      15473928
Metallic_Rim     3.5787926      37510212
Radio_cassette   0.2463033      13283659
Tow_Bar          2.7859789      25821047
Petrol           3.5219052      24975174
Diesel           3.7901764      20481394
CNG              0.8096385       4278186
```

R Exercise: Ensemble Regression

- Ensemble model 3: Random Forest

```
# Prediction
RF_haty <- predict(RF_model, newdata = tst_data[,-1], type = "response")
perf_table[5,] <- perf_eval(tst_data$Price, RF_haty)
perf_table
```

	MSE	RMSE	MAE	AMPE
MLR	1,539,502	1240.77	880.86	8.90%
ANN	3,286,328	1,812.82	1,288.35	12.62%
Bagging ANN	2,810,978	1,676.60	1,137.86	10.83%
GBM	2,353,836	1,534.22	1,088.47	11.12%
Random Forests	1,079,293	1,038.89	774.27	7.95%


ANY
questions?