

Recerca i Avaluació de models de predicció per estimar el Carboni Aeri Total a partir de dades LIDAR a Andorra

Jordi ORDOÑEZ ADELLACH

5/30/2023

1. Dades per elaborar el model

Disposem d'avaluacions de Carboni Aeri Total (CAT) sobre 194 parcel·les repartides pel territori nacional Andorrà. Aquestes parcel·les són circulars, de radis variables i s'ha procedit a la mesura física dels elements vegetals que hi són presents. En funció de les dades alomètriques obtingudes i de característiques de les espècies estudiades, s'ha associat a cada parcel·la un valor CAT.

També disposem de dades LIDAR per cadascuna de les parcel·les citades anteriorment. Associem les dades en un fitxer .csv

```
data <- read.csv("infaCAT.csv", header = TRUE, sep = ";", dec = ",")
data$Habitat <- as.factor(data$Habitat)
head(data)
```

```
##  Parcela  superficie  CAT  Habitat  AR2      ARM      CRR      CV  ELEVMAX  ELEVMEAN
## 1   AD1013    452.3893  73.2      PNX  100  47.2906  0.38448  0.3227  25.0398  14.3990
## 2   AD1014    452.3893  56.6      PNX  100  44.6894  0.43490  0.3487  22.3687  12.6247
## 3   AD1019    452.3893  66.8      PNX  100  53.2872  0.41722  0.3278  21.1379  12.4580
## 4   AD1020   1256.6371  22.0      PNM  100  37.3796  0.28821  0.3398  17.1765   8.9861
## 5   AD1021   1017.8760  44.8      PNX  100  49.3392  0.40099  0.3984  19.6798  10.6481
## 6   AD1022    452.3893  53.3      PNM  100  44.0647  0.31671  0.3253  22.7948  12.0918
##  ELEVMIN     P10     P25     P50     P75     P80     P99  R1C  R2C  R3C  R4C  KURTO
## 1  7.7522  8.5044  9.7717 14.1162 18.2419 19.1920 24.0989 456 142  11   0  1.8893
## 2  5.1257  6.7348  9.5194 11.5017 16.1564 17.1454 21.1619 382 108   9   0  2.0221
## 3  6.2439  7.2180  8.2472 12.8756 15.8313 16.4008 20.2915 434 131  13   0  1.6414
## 4  5.6697  6.3240  6.4983  7.5115 11.7148 12.3757 16.6772 437  74   8   0  2.3740
## 5  4.6022  5.4980  6.7264 10.5084 14.2273 14.9602 18.9800 354  91   9   0  1.7412
## 6  7.1309  7.8195  8.5251 10.8106 15.3181 15.7974 21.7941 441 113   2   0  2.2018
##  SKEWN     LFCC      RMH  P99P75  P75P50  P50P25      STD  FCC      IQ  FR2      FRM
## 1  0.2605  74.87685  0.56375  5.8570  4.1257  4.3445  4.6467 100  8.4702 100 56.5789
## 2  0.2635  76.55311  0.51419  5.0055  4.6547  1.9823  4.4022 100  6.6370 100 54.1885
## 3  0.0535  75.08651  0.60912  4.4602  2.9557  4.6284  4.0841 100  7.5841 100 68.8940
## 4  0.8789  84.20039  0.43731  4.9624  4.2033  1.0132  3.0537 100  5.2166 100 40.2746
## 5  0.2410  77.97357  0.53397  4.7527  3.7189  3.7820  4.2422 100  7.5009 100 61.2994
## 6  0.5755  79.31655  0.47426  6.4760  4.5075  2.2855  3.9336 100  6.7930 100 52.6077
##  TR  TRANMINHT
## 1  609      609
## 2  499      499
## 3  578      578
## 4  519      519
## 5  454      454
## 6  556      556
```

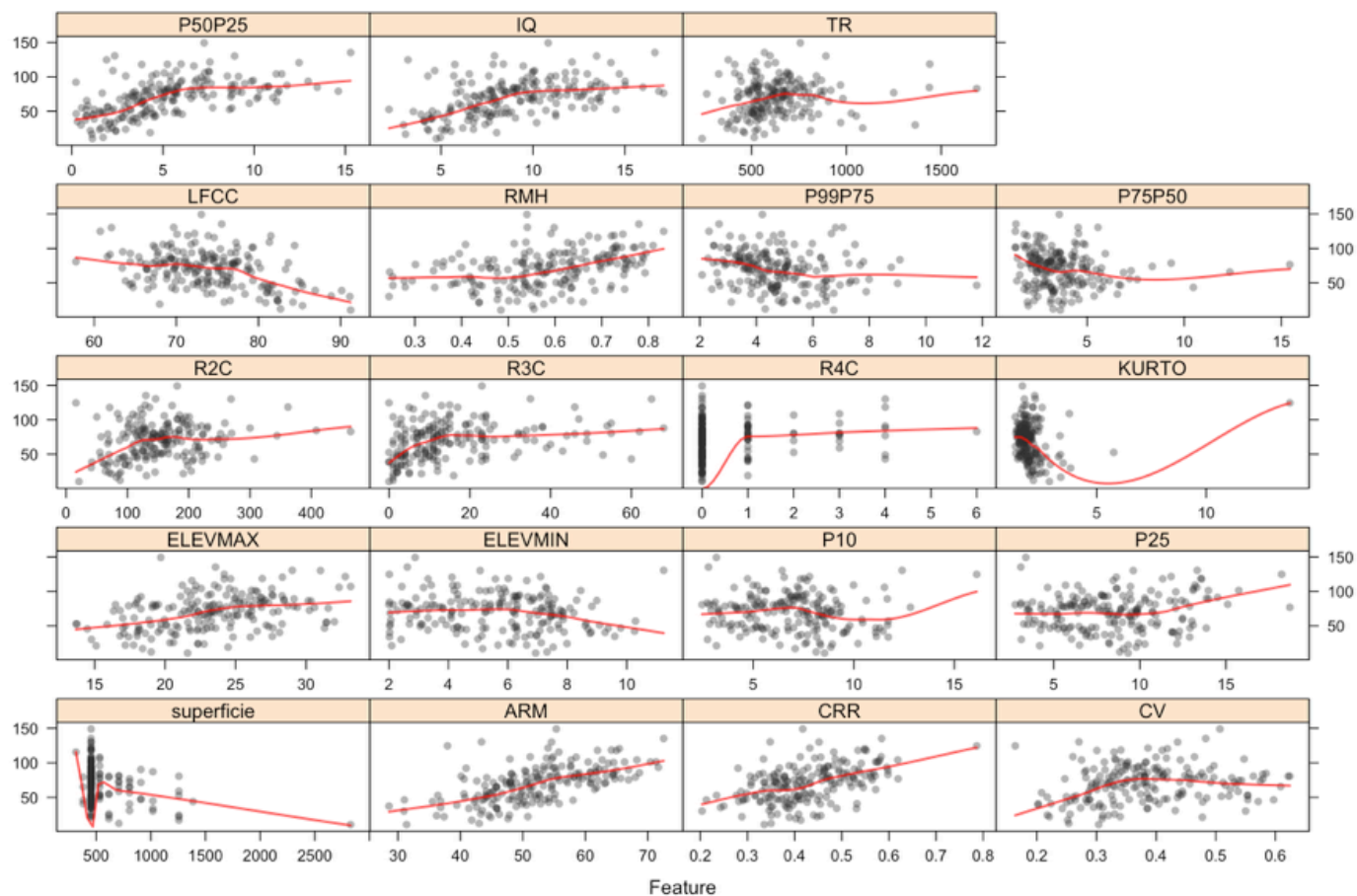
Disposem de 194 observacions de 35 variables entre les quals, la identificació de la parcel·la, el CAT associat a la parcel·la, el Habitat associat a la parcel·la, que no prové de les dades LIDAR i 32 variables obtingudes a partir de les observacions LIDAR.

2. Selecció de variables

Dins de les variables per entrenar el model, no ens servirà la identificació de la parcel·la, ni tampoc ens seran d'utilitat variables que tinguin una variabilitat propera a 0 o variables fortament correlacionades amb altres.

Procedirem a identificar i filtrar les variables amb variabilitat propera a 0 i després buscarem i eliminarem variables exoplicatives correlacionades amb un coeficient de correlació superior a 0,9 respecte a altres, excloient l'habitat d'aquest procediment per ser una variable no numérica.

```
# Near Zero Variable variables identifying and excluding
nzv <- nearZeroVar(data, saveMetrics = TRUE)
filtereddata <- data[, !nzv$nzv]
## Highly correlated variables identifying and excluding
numericdata <- filtereddata[, -c(1, 3, 4)]
dataCor <- cor(numericdata)
highlyCordata <- findCorrelation(dataCor, cutoff = .9)
filtereddata2 <- numericdata[, -highlyCordata]
```



CAT vs Variables

S'observa com la variable CAT té dependència respecte a les variables seleccionades.

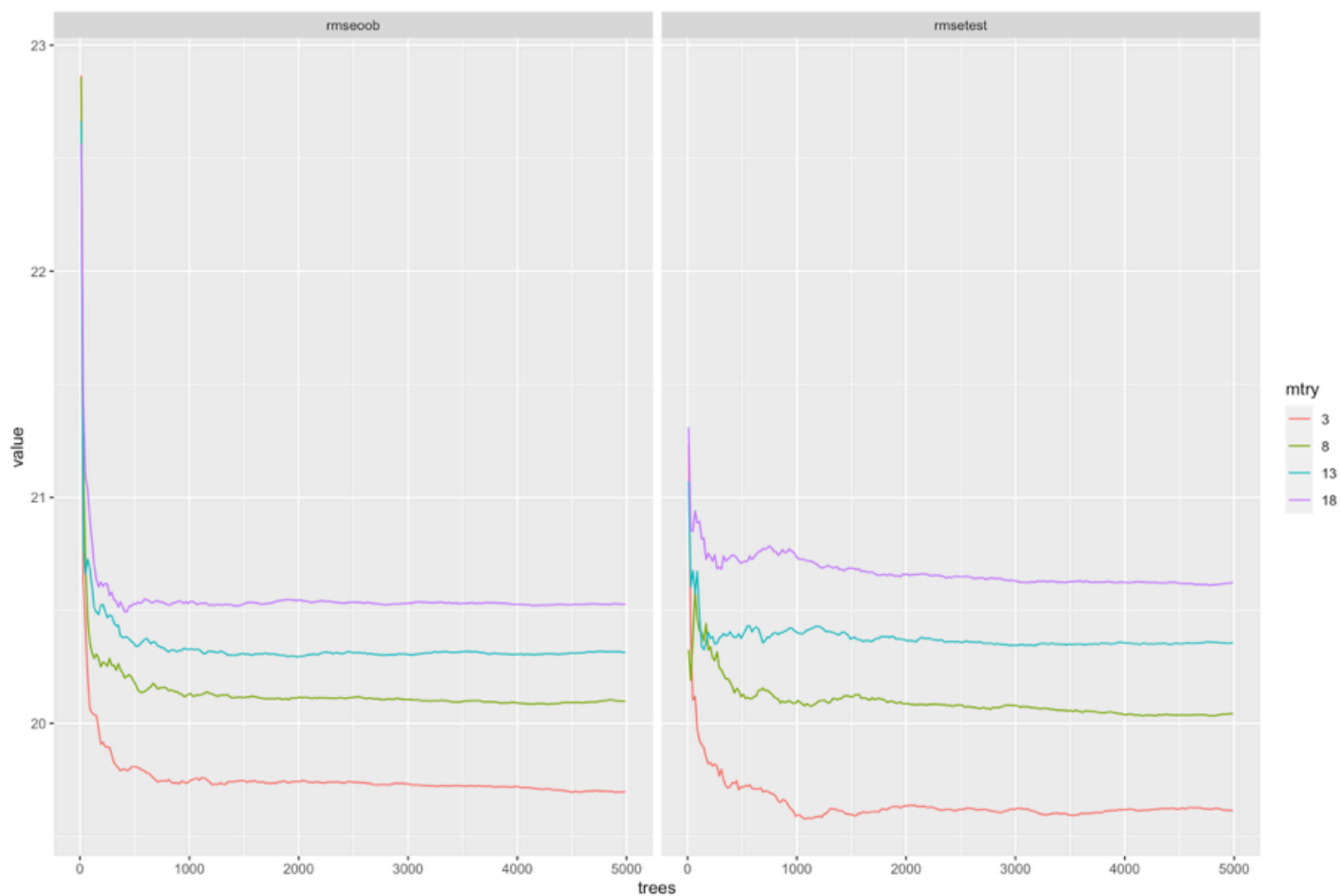
3. Entrenament del model

Seleccionarem un Training set a partir del 70% del total de les dades i deixarem el 30% de les dades restants en el Test set que només utilitzarem després d'haver seleccionat hiperàrmetres i entrenat el model amb aquests hiperparametres, per avaluar el nostre model.

```
set.seed(3456)
data2 <- cbind(CAT = data$CAT, filteredata2, Habitat = data$Habitat)
trainIndex <- createDataPartition(data2$CAT,
  p = .7,
  list = FALSE,
  times = 1
)

Train <- data2[trainIndex, ]
Test <- data2[-trainIndex, ]
```

Per escollir els hiperparametres `n tree` (nombre d'arbres del model) i `m try` (nombre de variables que escollirem a l'atzar per efectuar cada divisió dins dels arbres), procedirem per `CrossValidation` sobre 6 blocs sobre el nostre Training set.



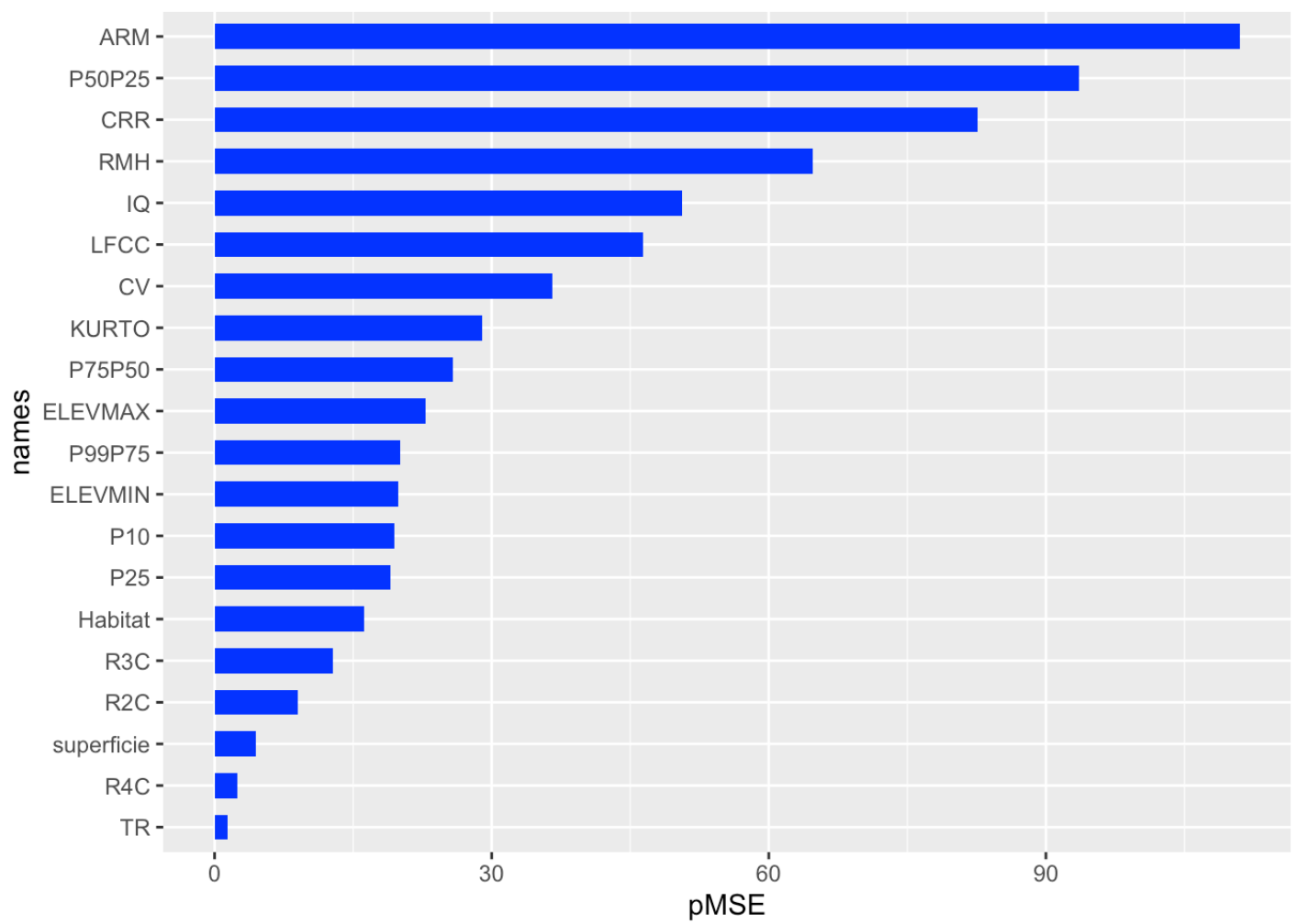
rmse oob and test

Podem observar per diferents valors de *mtry* les mitjanes dels rmse obtinguts amb els 6 crossvalidation sobre els out-of-bag elements i sobre els tests respecte a cada selecció de blocs. Veiem una bona estabilització dels errors amb una utilització dels 2500 arbres, pel que centrarem la recerca del *mtry* optim repertint el test anterior sobre valors de *mtry* propers a 3.



rmse oob and test

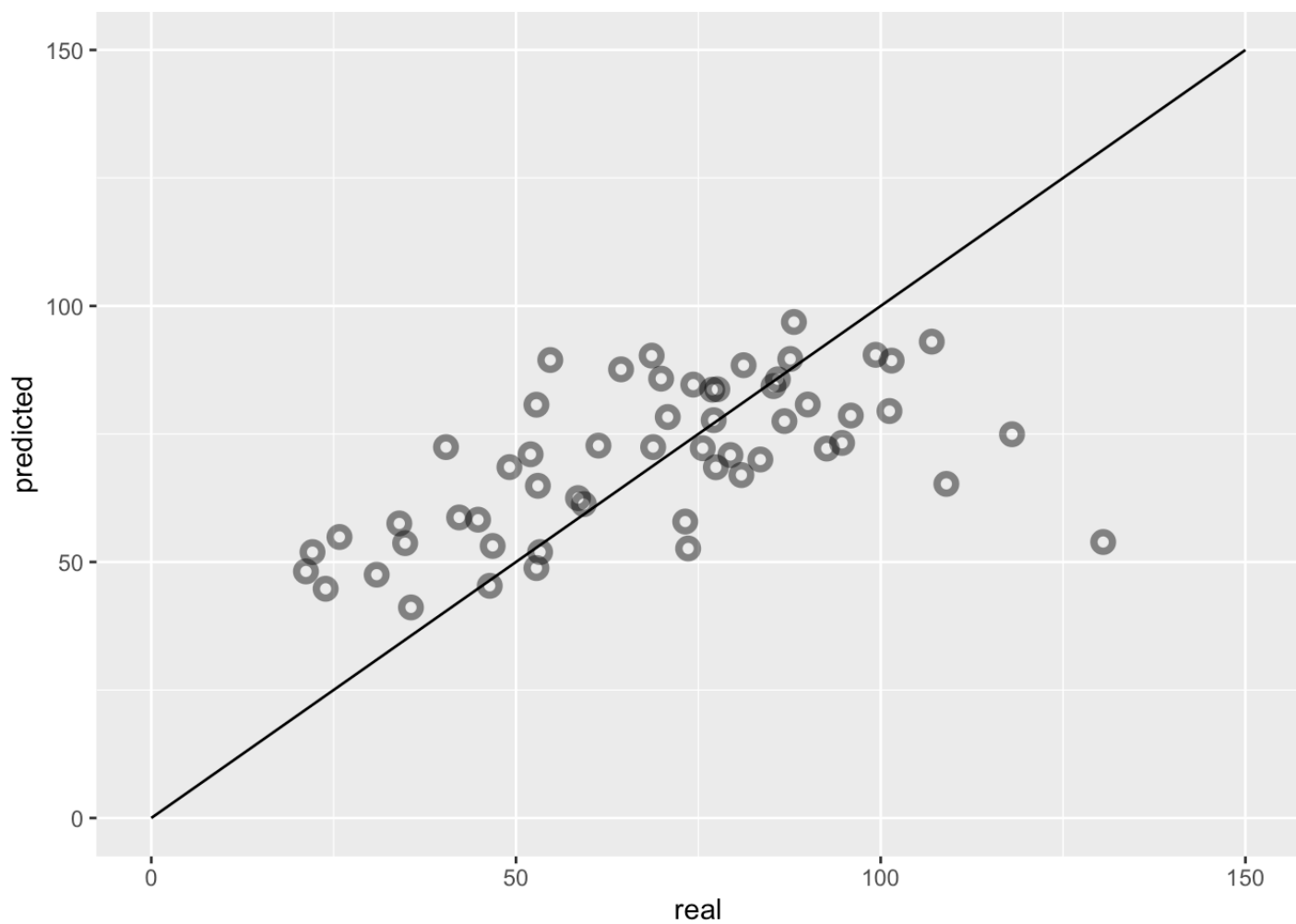
Observem un rmse òptim sobre tests estabilitzat amb 2500 arbres i $mtry=2$ variables i procedim a entrenar un model amb aquests paràmetres sense intervenir en la llargada dels arbres ni en el nombre mínim d'elements per node. Aquests paràmetres es podrien optimitzar seguint el mateix procediment que hem utilitzat per $mtry$ i $ntree$ òptim, però no esperem una millora substancial, però deixem la porta oberta.



Importància dels predictors

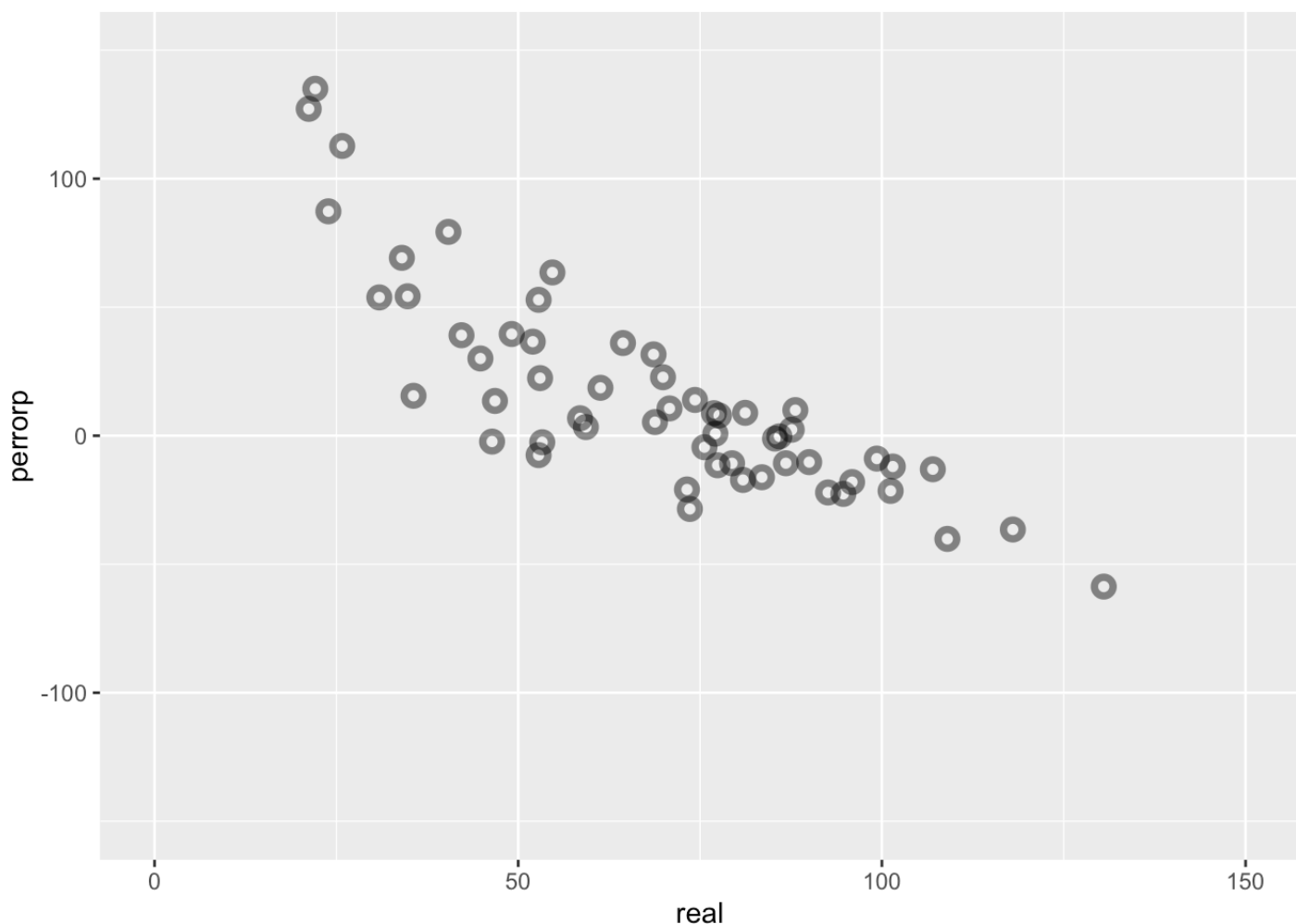
També podem veure quins predictors han agafat més importància en el model.

Obtenim un rmse de 20.45 sobre el Test set i un rrmse de 0.30



Predicted vs Real

Observem l'adequació de les prediccions respecte a les dades reals



% error predit sobre real

I finalment observem que els percentatges d'error són molt elevats per valors baixos de CAT reals, pel que en la predicció de CAT amb aquest model sobre la resta del territori, els valors baixos mereixerien una atenció suplementària a determinar.

Valor total del CAT sobre el Test Set : 3842 valor total predit sobre les mateixes parcel·les del test set : 3936 desviació de 1.02%.

4. Aplicació del model

Disposem de dades LIDAR per mallatge quadrats de 20mx20m als quals s'ha afegit els estrats de vegetació. Per tot el territori Andorrà. Identifiquem els noms necessaris del nostre model random forest escollit de entre tots els disponibles i efectuem un treball de càlcul i associació (eliminem les dades Estrat =BBR del que no disposàvem i modifiquem els retorns 4 buits o erronmis per valor 0) I afegim també la superfície d'aquestes quadrats.

```
datal <- read.csv("metriques1.csv", header = TRUE, sep = ";", dec = ".")
nomsnecessaris <- rownames(importance)
nomsdisp <- names(datal)

head(nomsnecessaris)
```

```
## [1] "TR"          "R4C"          "superficie"  "R2C"          "R3C"
## [6] "Habitat"
```



```
head(nomsdisp)
```

```
## [1] "row"          "center.X"      "center.Y"      "TR"            "Elev.minim"
## [6] "Elev.maxim"
```

```
# Les substituim per "0" i transformem en numeric
data1$Return.4.c[data1$Return.4.c == ""] <- "0"
data1$Return.4.c[data1$Return.4.c == "false"] <- "0"
data1$Return.4.c <- as.numeric(data1$Return.4.c)
```

```
# suprimim els "BBR" de habitats
data1 <- data1[data1$Habitat != "BBR", ]
```

```
# creem la nova taula
df <- data.frame(
  center.X = data1$center.X,
  center.Y = data1$center.Y,
  Habitat = data1$Habitat,
  R1C = data1$Return.1.c,
  R2C = data1$Return.2.c,
  R3C = data1$Return.3.c,
  R4C = data1$Return.4.c,
  TR = data1$TR,
  P10 = data1$Elev.P10,
  P25 = data1$Elev.P25,
  P50 = data1$Elev.P50,
  P75 = data1$Elev.P75,
  P99 = data1$Elev.P99,
  ELEVMIN = data1$Elev.minim,
  ELEVMAX = data1$Elev.maxim,
  IQ = data1$IQ,
  KURTO = data1$Elev.kurto,
  superficie = rep(400, length(data1$Elev.kurto)),
  CV = data1$CV,
  ARM = data1$Percenta_2,
  CRR = data1$Percenta_4 / 100
)
```

```
# afegim els calculats LFCC no es logaritme FCC!
```

```
df1 <- data.frame(df,
  LFCC = 100 * df$R1C / (df$TR),
  RMH = df$P50 / df$ELEVMAX,
  P99P75 = df$P99 - df$P75,
  P75P50 = df$P99 - df$P75,
  P50P25 = df$P50 - df$P25
)
```

```
df1$Habitat <- as.factor(df1$Habitat)
```

Ja podem fer correr el nostre model sobre aquestes dades per predir CAT i els afegim a un fitxer de sortida csv que conté les coordenades dels centres dels quadrats de la malla associats als valor del CAT predir, que ja es pot tractar amb QGIS.

```
predicted <- predict(data.rf, df1[nomsnecessaris])
result <- data.frame(
  X = df1$center.X,
  Y = df1$center.Y,
  CAT = as.vector(predicted)
)
result2 <- result[!is.na(result$CAT), ]
head(result2)
```

```
##           X      Y      CAT
## 1 532840 27360 47.10708
## 2 532860 27360 53.93316
## 3 532880 27360 76.09020
## 4 532840 27340 53.97923
## 5 532860 27340 71.86452
## 6 532960 27340 63.86864
```

```
write.csv(result2, "CAT_Jordi_01_07_23.csv", row.names = TRUE)
```