

Use the R code below to create a train dataset and a test database

```
library
(mlbench)
set.seed
(42)
sim_trn = mlbench.spirals(n = 2500, cycles = 1.5, sd = 0.125)
sim_trn = data.frame(sim_trn$x, class = as.factor(sim_trn$classes))
sim_tst = mlbench.spirals(n = 10000, cycles = 1.5, sd = 0.125)
sim_tst = data.frame(sim_tst$x, class = as.factor(sim_tst$classes))
library(caret)
cv_5 = trainControl(method = "cv", number = 5)
glm_cv_time = system.time({
sim_glm_cv = train(
class ~ .,
data = sim_trn,
trControl = cv_5,
method = "glm")
})
tree_cv_time = system.time({
sim_tree_cv = train(
class ~ .,
data = sim_trn,
trControl = cv_5,
method = "rpart")
})
glm_cv_time["elapsed"]
tree_cv_time["elapsed"]
```

you only have two predictor here, so for both, use the following grid

```
rf_grid = expand.grid(mtry = c(1, 2))
```

create a table summarizing the results of the following four models

a) Logistic model with CV, (2) a Tree model with CV, (3) an RF model with OOB, and (4) an RF model with CV.

Then, for each model, report the following items:

- a. Chosen value of tuning parameter (If applicable)
- b. Elapsed tuning time
- c. Resampled (CV or OOB) Accuracy
- d. Test Accuracy