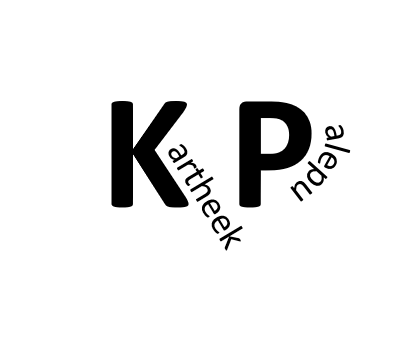


**R: Hyper Parameter Tuning**



**Note:** Use grid search with CV to search the space. If more values then use **random** grid search else use **cartesian** grid search.

**Logistic Regression**

alpha = (0, 0.25, 0.5, 0.5, 1)

**Random Forest**

mtries = 1 to 10

max\_depth = seq(1, 30, 5)

min\_rows = seq(10, 50, 5)

**GBM**

ntrees = 500 to 1000

learn\_rate = 0 to 1, tune it by selecting fixed number of trees

**(or)**

ntrees = 500 to 1000

learn\_rate = ((2-10)/ntrees)

**(or)**

# use this if in doubt

learn\_rate = 0.5

learn\_rate\_annealing = 0.99

sample\_rate = (0.5, 0.75, 1) # row sampling

col\_sample\_rate = (0.6, 0.8, 1)

max\_leaf\_weight = 3/(% of rare events)

max\_depth = (4, 6, 8, 10)

min\_split\_improvement = 0

**(or)**

max\_depth = (4, 6, 8, 10, 12, 14, 16, 18, 20)

min\_rows = (1, 5, 10, 20, 50, 100)

learn\_rate = (0.001, 0.01, 0.1, 1)

sample\_rate = seq(0.3, 1, 0.05)

col\_sample\_rate = seq(0.3, 1, 0.05)

**XGBoost**

subsample = (0.6, 0.7, 0.8, 0.9, 1) # number of samples used to grow trees

colsample\_bytree = seq(0.6, 1, 0.1)

ntrees = 100

eta = (0.01, 0.015, 0.02, 0.025, 0.05, 0.1) tune it using ntrees

gamma = if no clue use ‘5’

max\_depth = (3, 5, 7, 9, 12, 15, 17, 25)

min\_child\_weight = (1, 3, 5, 7)

lambda = (0.01, 0.1, 1)

alpha = (0, 0.01, 0.5, 1)

**Deep Learning**

input\_dropout\_ratio = (0, 0.15, 0.3)

hidden\_dropout\_ratio = (0, 0.15, 0.3)

hidden = (64)

epochs = 100

l1 = (0, 0.001, 0.00001)

l2 = (0, 0.001, 0.00001)

**(or)**

input\_dropout\_ratio = ((0, 0), (0.15, 0.15), (0.3, 0.3))

hidden\_dropout\_ratio = ((0, 0), (0.15, 0.15), (0.3, 0.3))

hidden = (32, 32)

epochs = 100

l1 = (0, 0.001, 0.00001)

l2 = (0, 0.001, 0.00001)