Random Forest and Boosting

# Random Forest and Boosting

## 1. Upload and prepare data

### 1.1 Upload data

oneypd\_tree <- read.csv(file = 'Z:/Model Risk/Adam/IFRS9\_CECL\_MV/data/chap2oneypd.csv')  
dplyr::glimpse(oneypd\_tree)

## Observations: 25,906  
## Variables: 45  
## $ X <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1...  
## $ id <int> 6670001, 9131199, 4963167, 39185...  
## $ vintage\_year <int> 2005, 2006, 2004, 2005, 2006, 20...  
## $ monthly\_installment <dbl> 746.70, 887.40, 1008.50, 458.23,...  
## $ loan\_balance <dbl> 131304.44, 115486.51, 128381.73,...  
## $ bureau\_score <int> 541, 441, 282, 461, 466, 470, 51...  
## $ num\_bankrupt\_iva <int> 0, 0, 0, 0, 0, 0, 0, 0, NA, 0, 0...  
## $ time\_since\_bankrupt <int> 0, 0, 0, 0, 0, 0, 0, 0, NA, 0, 0...  
## $ num\_ccj <int> 0, 0, 1, 0, 0, 0, 0, 0, NA, 0, 0...  
## $ time\_since\_ccj <int> 0, 0, 36, 0, 0, 0, 0, 0, NA, 0, ...  
## $ ccj\_amount <int> 0, 0, 459, 0, 0, 0, 0, 0, NA, 0,...  
## $ num\_bankrupt <int> 0, 0, 0, 0, 0, 0, 0, 0, NA, 0, 0...  
## $ num\_iva <int> 0, 0, 0, 0, 0, 0, 0, 0, NA, 0, 0...  
## $ min\_months\_since\_bankrupt <int> 0, 0, 0, 0, 0, 0, 0, 0, NA, 0, 0...  
## $ pl\_flag <int> 1, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0,...  
## $ region <fct> r\_a, r\_b, r\_c, r\_d, r\_e, r\_c, r\_...  
## $ ltv <dbl> 0.7586, 0.6973, 0.6959, 0.1099, ...  
## $ arrears\_months <dbl> 0.0000000, 0.0000000, 2.1882300,...  
## $ origination\_date <fct> 9/14/2005, 1/20/2006, 12/21/2004...  
## $ maturity\_date <fct> 9/30/2040, 1/31/2031, 12/31/2029...  
## $ repayment\_type <fct> Non-IO, Non-IO, Non-IO, Non-IO, ...  
## $ arrears\_status <int> 1, 1, 2, 1, 1, 1, 1, 1, 2, 1, 2,...  
## $ arrears\_segment <int> 1, 1, 2, 1, 1, 1, 1, 1, 2, 1, 1,...  
## $ mob <int> 120, 116, 129, 123, 110, 120, 13...  
## $ remaining\_mat <int> 300, 184, 171, 93, 310, 0, 166, ...  
## $ loan\_term <int> 35, 25, 25, 18, 35, 10, 25, 25, ...  
## $ live\_status <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## $ repaid\_status <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...  
## $ month <int> 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9,...  
## $ arrears\_event <int> 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0,...  
## $ bankrupt\_event <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...  
## $ term\_expiry\_event <int> 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0,...  
## $ worst\_arrears\_status <int> 1, 1, 2, 1, 1, 1, 1, 1, 2, 1, 2,...  
## $ max\_arrears\_12m <dbl> 0.000000, 0.000000, 2.188230, 0....  
## $ recent\_arrears\_date <fct> NA, NA, 9/30/2015, NA, NA, NA, N...  
## $ months\_since\_2mia <int> NA, NA, 0, NA, NA, NA, NA, NA, N...  
## $ avg\_mia\_6m <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2,...  
## $ max\_arrears\_bal\_6m <int> -42, 0, 1198, -114, 0, 0, -114, ...  
## $ max\_mia\_6m <int> 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 2,...  
## $ avg\_bal\_6m <int> 132080, 116972, 128500, 36610, 7...  
## $ avg\_bureau\_score\_6m <int> 542, 494, 290, 460, 468, 484, 51...  
## $ cc\_util <dbl> 0.4578, 0.6299, 0.6331, 0.4990, ...  
## $ annual\_income <int> 76749, 78451, 31038, 56663, 7701...  
## $ emp\_length <int> 3, 10, 3, 8, 10, 3, 11, 5, 4, 1,...  
## $ months\_since\_recent\_cc\_delinq <int> 11, 7, 6, 6, 3, 3, 13, 5, 3, 14,...