BA 810 Project Final Version

Team 6: Ji Qi, Yuxuan Mei, Yihan Jia, Yuhan Wang, Mochi Zhang

9/27/2021

Features

`enrollee_id` : Unique ID for candidate

`city`: City code

`city_development_index` : Developement index of the city (scaled)

`gender`: Gender of candidate

`relevent_experience`: Relevant experience of candidate

`enrolled_university`: Type of University course enrolled if any

`education_level`: Education level of candidate

`major_discipline` :Education major discipline of candidate

`experience`: Candidate total experience in years

`company_size`: No of employees in current employer's company

`company_type` : Type of current employer

`lastnewjob`: Difference in years between previous job and current job

`training_hours`: training hours completed

`target`: 0 – Not looking for job change, 1 – Looking for a job change

Load the dataset

```
library(data.table)
```

```
## Warning: package 'data.table' was built under R version 4.1.1
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.1.1
```

```
dd <- fread('/Users/moonqj/Desktop/Boston University/Semester/Fall 2021/BA 810/Project/data/aug_train.cs
str(dd)
```

```
## Classes 'data.table' and 'data.frame':   19158 obs. of  14 variables:
##  $ enrollee_id           : int  8949 29725 11561 33241 666 21651 28806 402 27107 699 ...
##  $ city                  : chr  "city_103" "city_40" "city_21" "city_115" ...
##  $ city_development_index: num  0.92 0.776 0.624 0.789 0.767 0.764 0.92 0.762 0.92 0.92 ...
##  $ gender                : chr  "Male" "Male" "" "" ...
##  $ relevent_experience   : chr  "Has relevent experience" "No relevent experience" "No relevent expe
##  $ enrolled_university   : chr  "no_enrollment" "no_enrollment" "Full time course" "" ...
##  $ education_level       : chr  "Graduate" "Graduate" "Graduate" "Graduate" ...
##  $ major_discipline      : chr  "STEM" "STEM" "STEM" "Business Degree" ...
##  $ experience            : chr  ">20" "15" "5" "<1" ...
##  $ company_size          : chr  "" "50-99" "" "" ...
##  $ company_type          : chr  "" "Pvt Ltd" "" "Pvt Ltd" ...
##  $ last_new_job          : chr  "1" ">4" "never" "never" ...
##  $ training_hours        : int  36 47 83 52 8 24 24 18 46 123 ...
##  $ target                : num  1 0 0 1 0 1 0 1 1 0 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

Information of the dataset

```
# how many rows and columns
dim(dd)
```

```
## [1] 19158    14
```

```
# basic stats
summary(dd)
```

```
##   enrollee_id        city           city_development_index    gender
##  Min.   :    1   Length:19158       Min.   :0.4480         Length:19158
##  1st Qu.: 8554   Class :character   1st Qu.:0.7400         Class :character
##  Median :16982   Mode  :character   Median :0.9030         Mode  :character
##  Mean   :16875                      Mean   :0.8288
##  3rd Qu.:25170                      3rd Qu.:0.9200
##  Max.   :33380                      Max.   :0.9490
##  relevent_experience enrolled_university education_level    major_discipline
##  Length:19158        Length:19158        Length:19158       Length:19158
##  Class :character    Class :character    Class :character   Class :character
##  Mode  :character    Mode  :character    Mode  :character   Mode  :character
##
##
##
##   experience        company_size       company_type       last_new_job
##  Length:19158       Length:19158       Length:19158       Length:19158
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##  training_hours       target
##  Min.   :  1.00   Min.   :0.0000
```

```
##  1st Qu.: 23.00   1st Qu.:0.0000
##  Median : 47.00   Median :0.0000
##  Mean   : 65.37   Mean   :0.2493
##  3rd Qu.: 88.00   3rd Qu.:0.0000
##  Max.   :336.00   Max.   :1.0000
```

Summary of the missing values

```
sum(dd == '')
```

```
## [1] 20733
```

```
check_missing <- function(x) {
  sum(is.null(x) | x == '')}
a <- data.frame(sapply(dd, check_missing))
setDT(a, keep.rownames = TRUE)[]
```

```
##                         rn sapply.dd..check_missing.
##  1:            enrollee_id                         0
##  2:                   city                         0
##  3: city_development_index                         0
##  4:                 gender                      4508
##  5:    relevent_experience                         0
##  6:    enrolled_university                       386
##  7:        education_level                       460
##  8:       major_discipline                      2813
##  9:             experience                        65
## 10:           company_size                      5938
## 11:           company_type                      6140
## 12:           last_new_job                       423
## 13:         training_hours                         0
## 14:                 target                         0
```

```
colnames(a) <- c ('variable_name', 'the_count_of_missing_values')

a[the_count_of_missing_values > 0][order(-the_count_of_missing_values)]
```

```
##          variable_name the_count_of_missing_values
## 1:        company_type                        6140
## 2:        company_size                        5938
## 3:              gender                        4508
## 4:    major_discipline                        2813
## 5:     education_level                         460
## 6:        last_new_job                         423
## 7: enrolled_university                         386
## 8:          experience                          65
```

Summary of notnull values

```
check_notnull <- function(x) {
  sum(x != '')}
b <- setDT(data.frame(sapply(dd, check_notnull)),  keep.rownames = TRUE)
colnames((b))
```

```
## [1] "rn"                          "sapply.dd..check_notnull."
```

```
b[,.(rn,(sapply.dd..check_notnull.))] [order(V2)]
```

```
##                          rn    V2
##  1:           company_type 13018
##  2:           company_size 13220
##  3:                 gender 14650
##  4:       major_discipline 16345
##  5:        education_level 18698
##  6:           last_new_job 18735
##  7:    enrolled_university 18772
##  8:             experience 19093
##  9:            enrollee_id 19158
## 10:                   city 19158
## 11: city_development_index 19158
## 12:    relevent_experience 19158
## 13:         training_hours 19158
## 14:                 target 19158
```

Specific info of each column

```
for (i in colnames(dd))
{
print(unique(dd[, i, with = FALSE]))
}
```

```
##        enrollee_id
##     1:        8949
##     2:       29725
##     3:       11561
##     4:       33241
##     5:         666
##    ---
## 19154:        7386
## 19155:       31398
## 19156:       24576
## 19157:        5756
## 19158:       23834
##          city
##   1: city_103
```

```
##   2:  city_40
##   3:  city_21
##   4: city_115
##   5: city_162
##  ---
## 119: city_121
## 120: city_129
## 121:   city_8
## 122:  city_31
## 123: city_171
##     city_development_index
##  1:                  0.920
##  2:                  0.776
##  3:                  0.624
##  4:                  0.789
##  5:                  0.767
##  6:                  0.764
##  7:                  0.762
##  8:                  0.913
##  9:                  0.926
## 10:                  0.827
## 11:                  0.843
## 12:                  0.804
## 13:                  0.855
## 14:                  0.887
## 15:                  0.910
## 16:                  0.884
## 17:                  0.924
## 18:                  0.666
## 19:                  0.558
## 20:                  0.923
## 21:                  0.794
## 22:                  0.754
## 23:                  0.939
## 24:                  0.550
## 25:                  0.865
## 26:                  0.698
## 27:                  0.893
## 28:                  0.796
## 29:                  0.866
## 30:                  0.682
## 31:                  0.802
## 32:                  0.579
## 33:                  0.878
## 34:                  0.897
## 35:                  0.949
## 36:                  0.925
## 37:                  0.896
## 38:                  0.836
## 39:                  0.693
## 40:                  0.769
## 41:                  0.775
## 42:                  0.903
## 43:                  0.555
```

```
## 44:                  0.727
## 45:                  0.640
## 46:                  0.516
## 47:                  0.743
## 48:                  0.899
## 49:                  0.915
## 50:                  0.689
## 51:                  0.895
## 52:                  0.890
## 53:                  0.847
## 54:                  0.527
## 55:                  0.766
## 56:                  0.738
## 57:                  0.647
## 58:                  0.795
## 59:                  0.740
## 60:                  0.701
## 61:                  0.493
## 62:                  0.840
## 63:                  0.691
## 64:                  0.735
## 65:                  0.742
## 66:                  0.479
## 67:                  0.722
## 68:                  0.921
## 69:                  0.848
## 70:                  0.856
## 71:                  0.898
## 72:                  0.830
## 73:                  0.730
## 74:                  0.680
## 75:                  0.725
## 76:                  0.556
## 77:                  0.448
## 78:                  0.763
## 79:                  0.745
## 80:                  0.645
## 81:                  0.788
## 82:                  0.780
## 83:                  0.512
## 84:                  0.739
## 85:                  0.563
## 86:                  0.518
## 87:                  0.824
## 88:                  0.487
## 89:                  0.649
## 90:                  0.781
## 91:                  0.625
## 92:                  0.807
## 93:                  0.664
##     city_development_index
##    gender
## 1:   Male
## 2:
```

```
## 3: Female
## 4:  Other
##        relevent_experience
## 1: Has relevent experience
## 2:  No relevent experience
##    enrolled_university
## 1:       no_enrollment
## 2:    Full time course
## 3:
## 4:    Part time course
##    education_level
## 1:        Graduate
## 2:         Masters
## 3:     High School
## 4:
## 5:             Phd
## 6:  Primary School
##    major_discipline
## 1:             STEM
## 2:  Business Degree
## 3:
## 4:             Arts
## 5:       Humanities
## 6:         No Major
## 7:            Other
##     experience
##  1:        >20
##  2:         15
##  3:          5
##  4:         <1
##  5:         11
##  6:         13
##  7:          7
##  8:         17
##  9:          2
## 10:         16
## 11:          1
## 12:          4
## 13:         10
## 14:         14
## 15:         18
## 16:         19
## 17:         12
## 18:          3
## 19:          6
## 20:          9
## 21:          8
## 22:         20
## 23:
##     experience
##    company_size
## 1:
## 2:        50-99
## 3:          <10
```

```
## 4:       10000+
## 5:    5000-9999
## 6:    1000-4999
## 7:        10/49
## 8:      100-500
## 9:      500-999
##           company_type
## 1:
## 2:             Pvt Ltd
## 3:      Funded Startup
## 4: Early Stage Startup
## 5:               Other
## 6:       Public Sector
## 7:                 NGO
##    last_new_job
## 1:            1
## 2:           >4
## 3:        never
## 4:            4
## 5:            3
## 6:            2
## 7:
##      training_hours
##   1:             36
##   2:             47
##   3:             83
##   4:             52
##   5:              8
##  ---
## 237:            244
## 238:            272
## 239:            294
## 240:            270
## 241:            286
##    target
## 1:      1
## 2:      0
```

```
for (i in colnames(dd))
{
print((dd[, .N, by = i ]))
}
```

```
##        enrollee_id N
##     1:        8949 1
##     2:       29725 1
##     3:       11561 1
##     4:       33241 1
##     5:         666 1
##    ---
## 19154:        7386 1
## 19155:       31398 1
## 19156:       24576 1
## 19157:        5756 1
```

```
## 19158:       23834 1
##          city    N
##   1: city_103 4355
##   2:  city_40   68
##   3:  city_21 2702
##   4: city_115   54
##   5: city_162  128
##  ---
## 119: city_121    3
## 120: city_129    3
## 121:   city_8    4
## 122:  city_31    4
## 123: city_171    1
##     city_development_index    N
##  1:                  0.920 5200
##  2:                  0.776   82
##  3:                  0.624 2702
##  4:                  0.789   54
##  5:                  0.767  128
##  6:                  0.764   24
##  7:                  0.762  128
##  8:                  0.913  197
##  9:                  0.926 1336
## 10:                  0.827  137
## 11:                  0.843   94
## 12:                  0.804  304
## 13:                  0.855  431
## 14:                  0.887  275
## 15:                  0.910 1533
## 16:                  0.884  266
## 17:                  0.924  301
## 18:                  0.666  114
## 19:                  0.558   75
## 20:                  0.923  143
## 21:                  0.794   93
## 22:                  0.754  280
## 23:                  0.939  497
## 24:                  0.550  247
## 25:                  0.865   26
## 26:                  0.698  683
## 27:                  0.893  160
## 28:                  0.796   29
## 29:                  0.866  103
## 30:                  0.682  119
## 31:                  0.802  175
## 32:                  0.579  135
## 33:                  0.878  151
## 34:                  0.897  586
## 35:                  0.949   79
## 36:                  0.925  171
## 37:                  0.896  140
## 38:                  0.836  120
## 39:                  0.693    4
## 40:                  0.769   22
```

```
## 41:                  0.775   10
## 42:                  0.903   82
## 43:                  0.555   63
## 44:                  0.727   53
## 45:                  0.640   13
## 46:                  0.516   12
## 47:                  0.743  146
## 48:                  0.899  182
## 49:                  0.915   94
## 50:                  0.689  102
## 51:                  0.895   86
## 52:                  0.890  113
## 53:                  0.847   41
## 54:                  0.527   92
## 55:                  0.766   49
## 56:                  0.738   79
## 57:                  0.647   27
## 58:                  0.795   20
## 59:                  0.740   67
## 60:                  0.701    9
## 61:                  0.493   13
## 62:                  0.840   29
## 63:                  0.691   45
## 64:                  0.735    8
## 65:                  0.742   10
## 66:                  0.479   28
## 67:                  0.722   27
## 68:                  0.921   10
## 69:                  0.848   47
## 70:                  0.856   32
## 71:                  0.898   11
## 72:                  0.830   32
## 73:                  0.730    7
## 74:                  0.680    9
## 75:                  0.725   18
## 76:                  0.556   14
## 77:                  0.448   17
## 78:                  0.763   27
## 79:                  0.745   10
## 80:                  0.645    5
## 81:                  0.788    7
## 82:                  0.780    6
## 83:                  0.512    5
## 84:                  0.739   14
## 85:                  0.563   13
## 86:                  0.518    6
## 87:                  0.824    4
## 88:                  0.487    5
## 89:                  0.649    4
## 90:                  0.781    3
## 91:                  0.625    3
## 92:                  0.807    4
## 93:                  0.664    1
##     city_development_index    N
```

```
##    gender     N
## 1:   Male 13221
## 2:         4508
## 3: Female  1238
## 4:  Other   191
##        relevent_experience     N
## 1: Has relevent experience 13792
## 2:  No relevent experience  5366
##    enrolled_university     N
## 1:       no_enrollment 13817
## 2:    Full time course  3757
## 3:                       386
## 4:    Part time course  1198
##    education_level     N
## 1:        Graduate 11598
## 2:         Masters  4361
## 3:     High School  2017
## 4:                   460
## 5:             Phd   414
## 6:  Primary School   308
##    major_discipline     N
## 1:             STEM 14492
## 2:  Business Degree   327
## 3:                   2813
## 4:             Arts   253
## 5:       Humanities   669
## 6:         No Major   223
## 7:            Other   381
##     experience    N
##  1:        >20 3286
##  2:         15  686
##  3:          5 1430
##  4:         <1  522
##  5:         11  664
##  6:         13  399
##  7:          7 1028
##  8:         17  342
##  9:          2 1127
## 10:         16  508
## 11:          1  549
## 12:          4 1403
## 13:         10  985
## 14:         14  586
## 15:         18  280
## 16:         19  304
## 17:         12  494
## 18:          3 1354
## 19:          6 1216
## 20:          9  980
## 21:          8  802
## 22:         20  148
## 23:             65
##     experience    N
##    company_size    N
```

```
## 1:              5938
## 2:        50-99 3083
## 3:          <10 1308
## 4:       10000+ 2019
## 5:    5000-9999  563
## 6:    1000-4999 1328
## 7:        10/49 1471
## 8:      100-500 2571
## 9:      500-999  877
##           company_type    N
## 1:                     6140
## 2:             Pvt Ltd 9817
## 3:      Funded Startup 1001
## 4: Early Stage Startup  603
## 5:               Other  121
## 6:       Public Sector  955
## 7:                 NGO  521
##    last_new_job    N
## 1:            1 8040
## 2:           >4 3290
## 3:        never 2452
## 4:            4 1029
## 5:            3 1024
## 6:            2 2900
## 7:              423
##      training_hours   N
##   1:             36 211
##   2:             47 157
##   3:             83  86
##   4:             52 196
##   5:              8 227
##  ---
## 237:            244   8
## 238:            272   5
## 239:            294   6
## 240:            270   7
## 241:            286   5
##    target     N
## 1:      1  4777
## 2:      0 14381
```

Fill the missing data

company_type 6140
company_size 5938
gender 4508
major_discipline 2813
education_level 460
last_new_job 423
enrolled_university 386 experience 65

```r
# company_type  6140, fill with the mode value
company_type_mode <- dd[, max(.N), by = company_type][V1 == max(V1),company_type]
dd_cleaned <- dd[(company_type == ''), company_type := company_type_mode]
print((dd_cleaned[, .N, by = company_type]))
```

```
##           company_type     N
## 1:             Pvt Ltd 15957
## 2:      Funded Startup  1001
## 3: Early Stage Startup   603
## 4:               Other   121
## 5:       Public Sector   955
## 6:                 NGO   521
```

```r
# company_size 5938, fill with the mode value

dd_cleaned <- dd[(company_size == '10/49'), company_size := '10-49' ]
company_size_mode <-dd[company_size != '', max(.N), by = company_size][V1 == max(V1),company_size]
dd_cleaned <- dd[(company_size == ''), company_size := company_size_mode]
print((dd_cleaned[, .N, by = company_size]))
```

```
##    company_size    N
## 1:        50-99 9021
## 2:          <10 1308
## 3:       10000+ 2019
## 4:    5000-9999  563
## 5:    1000-4999 1328
## 6:        10-49 1471
## 7:      100-500 2571
## 8:      500-999  877
```

```r
# gender    4508, classified these unknown gender as other

dd_cleaned <- dd[gender == '', gender := 'Other' ]
print((dd_cleaned[, .N, by = gender]))
```

```
##    gender     N
## 1:   Male 13221
## 2:  Other  4699
## 3: Female  1238
```

```r
# major_discipline  2813, fill with the mode value
major_discipline__mode <-dd[major_discipline != '', max(.N), by = major_discipline][V1 == max(V1),major
dd_cleaned <- dd[(major_discipline == ''), major_discipline := major_discipline__mode]
print((dd_cleaned[, .N, by = major_discipline]))
```

```
##    major_discipline     N
## 1:             STEM 17305
## 2:  Business Degree   327
## 3:             Arts   253
## 4:       Humanities   669
## 5:         No Major   223
## 6:            Other   381
```

```
# education_level   460, fill with the "Primary School"
dd_cleaned <- dd[(education_level == ''), education_level :=  'Primary School']
print((dd_cleaned[, .N, by = education_level]))
```

```
##    education_level     N
## 1:        Graduate 11598
## 2:         Masters  4361
## 3:     High School  2017
## 4:  Primary School   768
## 5:             Phd   414
```

```
# last_new_job  423 , fill with the mode value
last_new_job_mode <-dd[last_new_job != '', max(.N), by = last_new_job][V1 == max(V1),last_new_job]
dd_cleaned <- dd[(last_new_job == ''), last_new_job := last_new_job_mode]
print((dd_cleaned[, .N, by = last_new_job]))
```

```
##    last_new_job    N
## 1:            1 8463
## 2:           >4 3290
## 3:        never 2452
## 4:            4 1029
## 5:            3 1024
## 6:            2 2900
```

```
# enrolled_university   386
enrolled_university_mode <-dd[enrolled_university!= '', max(.N), by = enrolled_university][V1 == max(V1)
dd_cleaned <- dd[(enrolled_university == ''),enrolled_university := enrolled_university_mode]
print((dd_cleaned[, .N, by = enrolled_university]))
```

```
##    enrolled_university     N
## 1:       no_enrollment 14203
## 2:    Full time course  3757
## 3:    Part time course  1198
```

```
# experience    65, classified NA as '<1', fill with the mode value
dd_cleaned[experience == '', experience := NA]

experience__mode <-dd[experience != '', max(.N), by = experience][V1 == max(V1),experience]

dd_cleaned[is.na(experience) , experience := experience__mode]

dd_cleaned[(experience == '>20'), experience := 21]

dd_cleaned[(experience == '<1'), experience := 0]

# change the datatype of experience into numeric
dd_cleaned[ , experience := as.numeric(experience)]

print((dd_cleaned[, .N, by = experience]))
```

```
##     experience    N
```

```
##  1:         21 3351
##  2:         15  686
##  3:          5 1430
##  4:          0  522
##  5:         11  664
##  6:         13  399
##  7:          7 1028
##  8:         17  342
##  9:          2 1127
## 10:         16  508
## 11:          1  549
## 12:          4 1403
## 13:         10  985
## 14:         14  586
## 15:         18  280
## 16:         19  304
## 17:         12  494
## 18:          3 1354
## 19:          6 1216
## 20:          9  980
## 21:          8  802
## 22:         20  148
##     experience    N
```

```
# Drop 'enrollee_id', 'city' columns
dd_cleaned[ , c('enrollee_id', 'city') := NULL]
```

```
# Change the categorical variables into dummy variables

install.packages('fastDummies', repos= 'https://github.com/jacobkap/fastDummies.git')
```

```
## Warning: unable to access index for repository https://github.com/jacobkap/fastDummies.git/src/contri
##   cannot open URL 'https://github.com/jacobkap/fastDummies.git/src/contrib/PACKAGES'
```

```
## Warning: package 'fastDummies' is not available for this version of R
##
## A version of this package for your version of R might be available elsewhere,
## see the ideas at
## https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#Installing-packages
```

```
## Warning: unable to access index for repository https://github.com/jacobkap/fastDummies.git/bin/macosx
##   cannot open URL 'https://github.com/jacobkap/fastDummies.git/bin/macosx/big-sur-arm64/contrib/4.1/P
```

```
library(fastDummies)
results <- fastDummies::dummy_cols(dd_cleaned, remove_first_dummy = TRUE)

library(data.table)

setnames(results, "relevent_experience_No relevent experience", "relevent_experience_No_relevent_experie
setnames(results, "enrolled_university_Part time course", "enrolled_university_Part_time_course")
setnames(results, "education_level_High School", "education_level_High_School")
setnames(results, "education_level_Primary School", "education_level_Primary_School")
```

```
setnames(results, "company_size_10-49", "company_size_10_49")
setnames(results, "company_size_50-99", "company_size_50_99")
setnames(results, "company_type_Funded Startup", "company_type_Funded_Startup")

setnames(results, "company_size_100-500", "company_size_100_500")
setnames(results, "company_size_500-999", "company_size_500_999")
setnames(results, "company_size_1000-4999", "company_size_1000_4999")
setnames(results, "company_size_5000-9999", "company_size_5000_9999")
setnames(results, "company_type_Pvt Ltd", "company_type_Pvt_Ltd")
setnames(results, 'company_type_Public Sector', "company_type_Public_Sector")
setnames(results, 'major_discipline_No Major', "major_discipline_No_Major")
setnames(results, 'major_discipline_Business Degree', "major_discipline_Business_Degree")
setnames(results, 'company_size_10000+', "company_size_10000")
```

```
write.csv(results, "~/cleaned_data_810_10_06.csv", row.names = FALSE)
```

Exploratory Data Analysis

```
library(ggplot2)
theme_Ji <- theme_bw()+
  theme(
    plot.title=element_text(hjust=0.5, vjust=0.5, face='bold.italic'),
    axis.text.x = element_text(face="bold", color="#993333",
                               size=10, angle=0),
    axis.text.y = element_text(face="bold", color="#993333",
                               size=10, angle=0),
    axis.title.x = element_text(color="black", size=14, face="bold"),
    axis.title.y = element_text(color="black", size=14, face="bold")
  )

theme_set(theme_Ji)
```

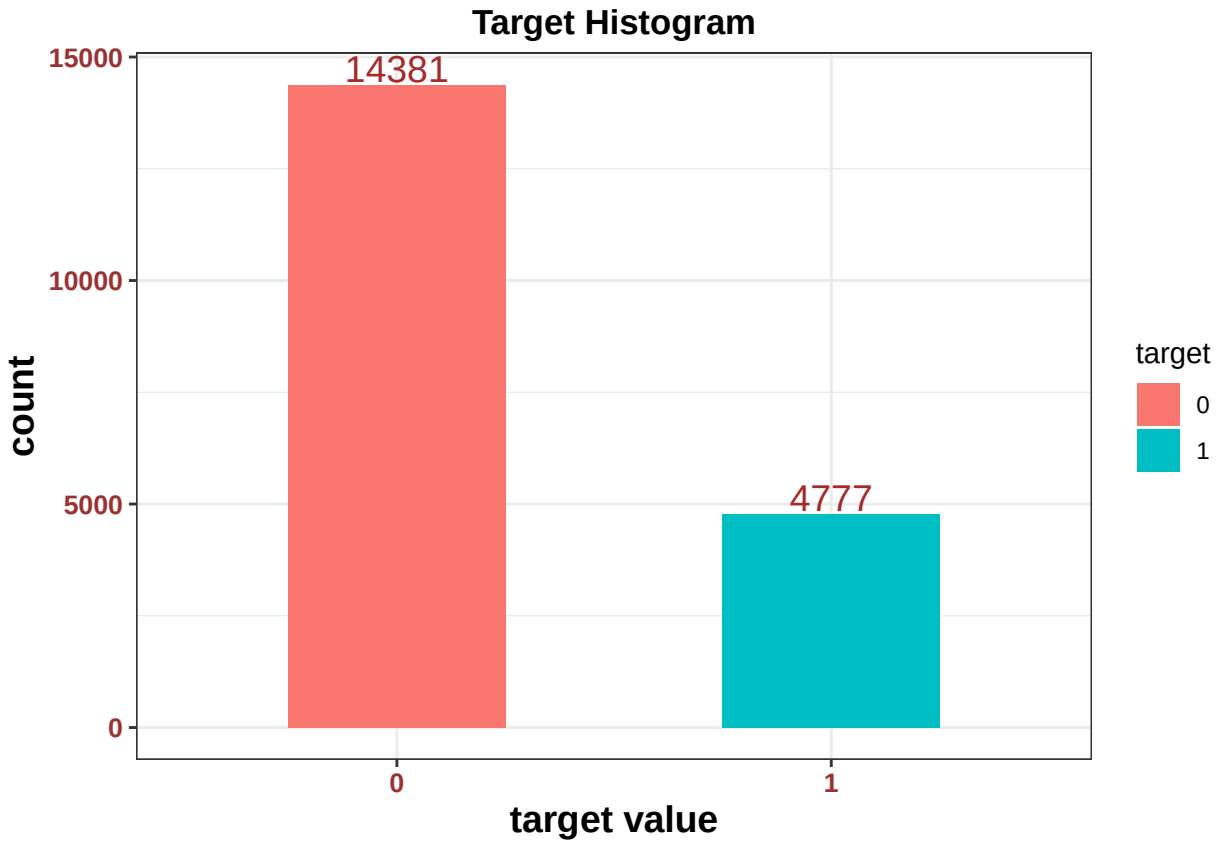
Target Column Histogram

0 - Not looking for job change 1 – Looking for a job change

This dataset is imbalanced and the ratio of '0 - Not looking for job change' to '1 – Looking for a job change' is equal to 3 : 1

```
# target column
target <- results[, target]
target <- data.table(target)


ggplot(results, aes(x = as.factor(target), fill = as.factor(target)))+
  geom_bar(stat = 'count', width = 0.5, position = 'dodge')+
  labs(x='target value', y = 'count')+
  ggtitle("Target Histogram") +
  geom_text(stat='count', aes(label=..count..), position = position_dodge(width = .5), vjust=-.1, size =
  scale_fill_hue(name="target")+
```

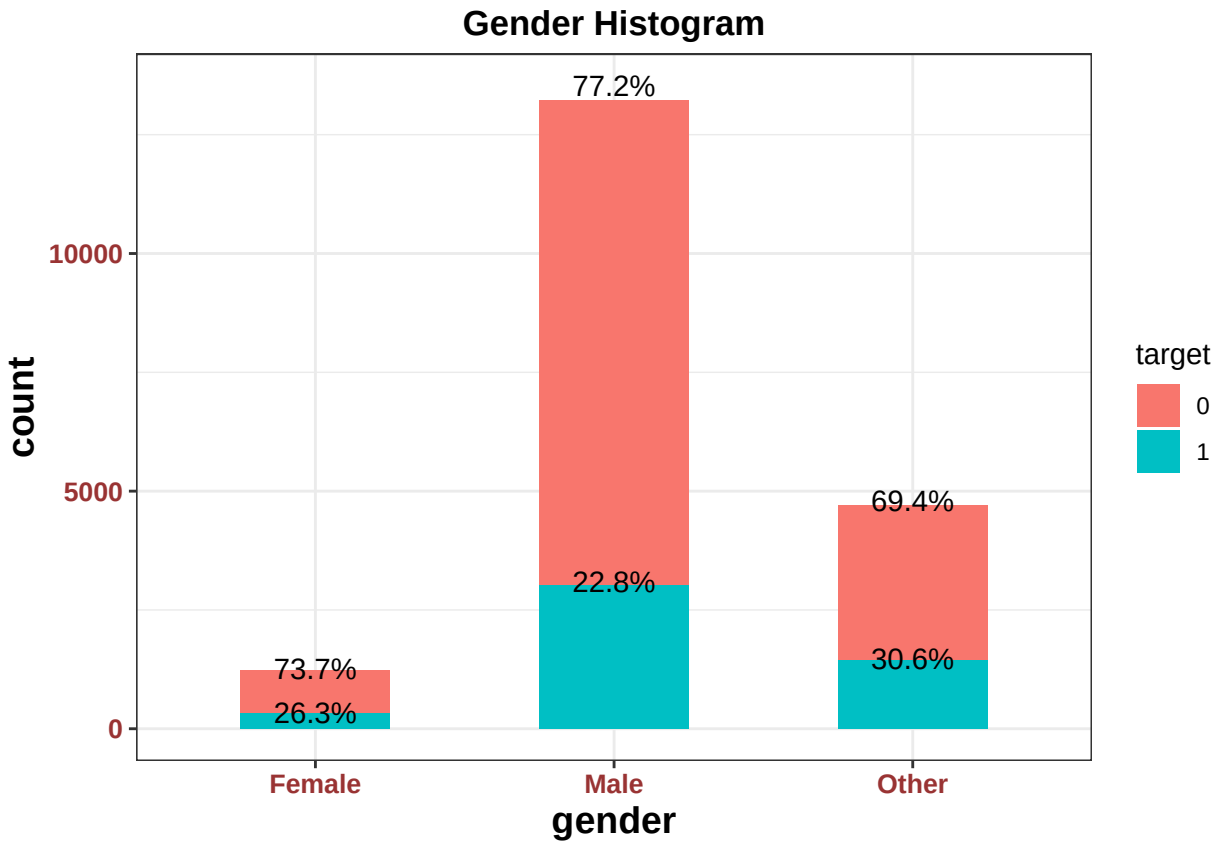
```
  theme(
    plot.title=element_text(hjust=0.5, vjust=0.5, face='bold')
  )
```



Gender Column Histogram

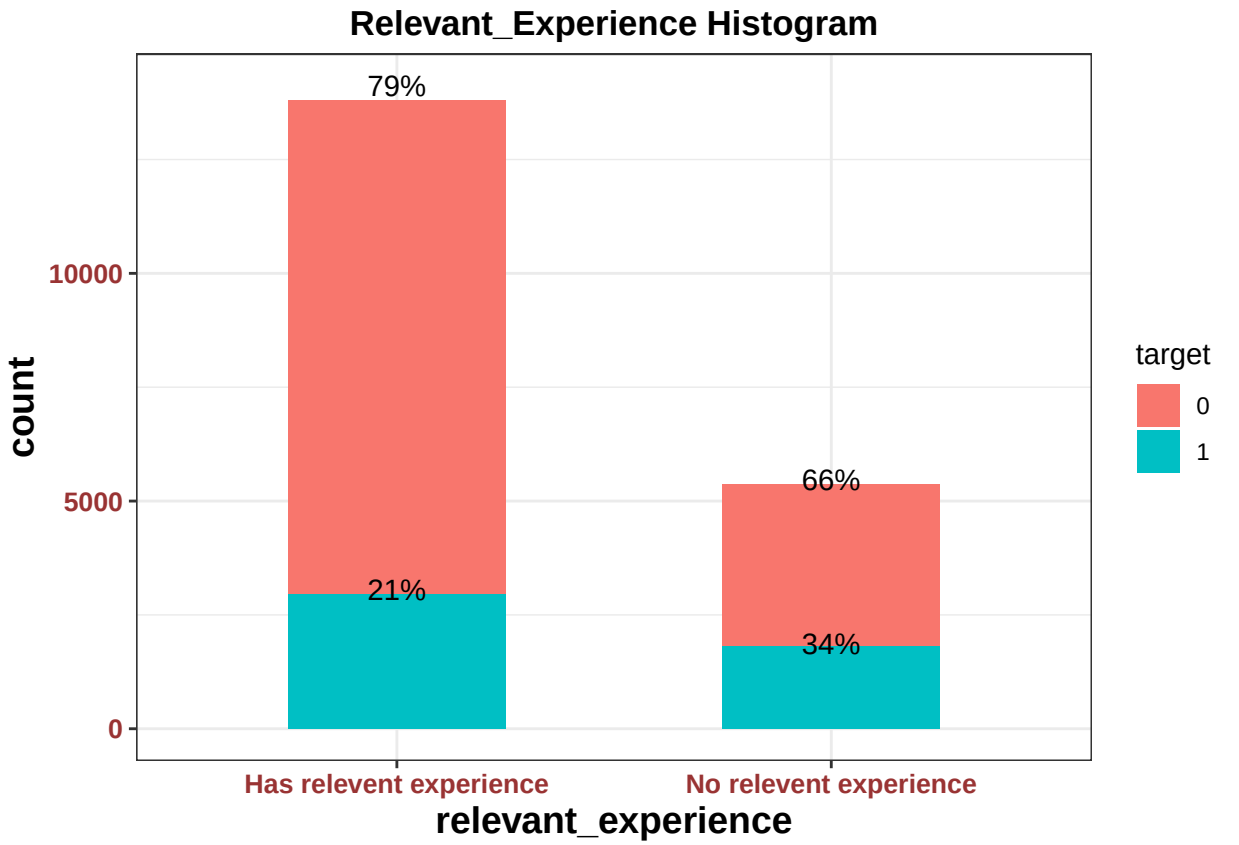
Female Data Scientists are more likely looking for a new job in comparison with other genders.

```
gender <- results[, gender]
gender <- data.table(gender)


ggplot(results, aes(x = as.factor(gender), fill = as.factor(target)))+
  geom_bar(stat = 'count', width = 0.5, position = 'stack')+
  labs(x='gender', y = 'count')+
  ggtitle("Gender Histogram") +
  geom_text(stat='count', aes(label=scales::percent(..count../tapply(..count.., ..x.. ,sum)[..x..])), p
  scale_fill_hue(name="target")+
  theme(
    plot.title=element_text(hjust=0.5, vjust=0.5, face='bold')
  )
```



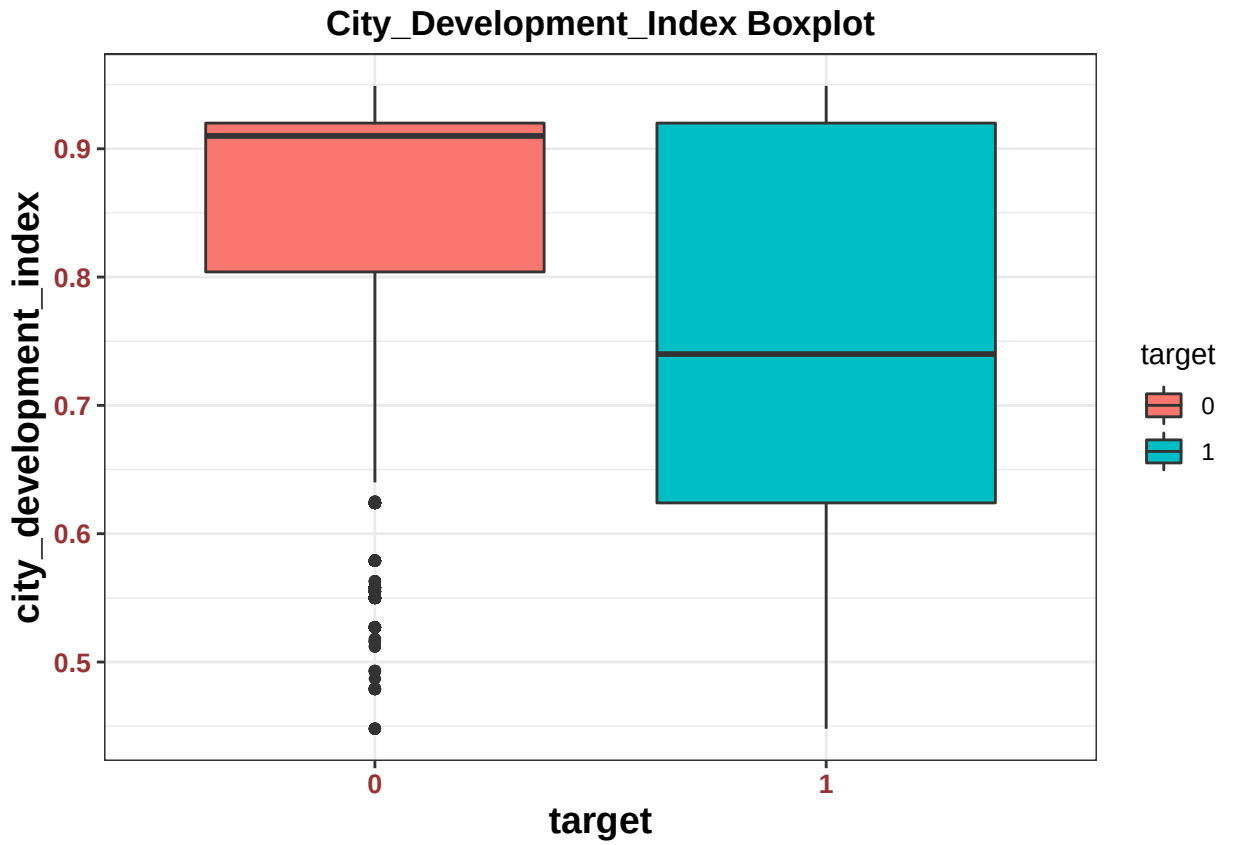
Relative Experience Column Histogram Data Scientists without relevant experience have higher chances of leaving a Job

```
relevent_experience <- results[, relevent_experience]
relevent_experience <- data.table(relevent_experience)


ggplot(results, aes(x = as.factor(relevent_experience), fill = as.factor(target)))+
  geom_bar(stat = 'count', width = 0.5, position = 'stack')+
  labs(x='relevant_experience', y = 'count')+
  ggtitle("Relevant_Experience Histogram") +
  geom_text(stat='count', aes(label=scales::percent(..count../tapply(..count.., ..x.. ,sum)[..x..])), p
  scale_fill_hue(name="target")+
  theme(
    plot.title=element_text(hjust=0.5, vjust=0.5, face='bold')
  )
```



City_Development_Index Boxplot Candidates are going to look for a new job, since the city where they live has a lower city_development_index.

```
ggplot(results, aes(x=as.factor(target), y=city_development_index, fill = as.factor(target))) +
  geom_boxplot()+
  labs(x='target', y = 'city_development_index')+
  scale_fill_hue(name="target")+
  ggtitle("City_Development_Index Boxplot") +
theme(legend.position="right", plot.title=element_text(hjust=0.5, vjust=0.5, face='bold'))
```



Training Hours Violinplot

The data points of training hours are mainly located between 0 and 100 hours. No relationship between training hours and willingness to change their jobs

```
ggplot(results, aes(x=as.factor(target), y=training_hours, fill = as.factor(target))) +
  geom_violin(trim=FALSE) +
  labs(x='target', y = 'training_hours')+
  stat_summary(fun.y=mean, geom="point", shape=23, size=2)+
  geom_boxplot(width=0.1)+
  scale_fill_hue(name="target")+
  ggtitle("Training Hours Violinplot")+
  theme(plot.title=element_text(hjust=0.5, vjust=0.5, face='bold'))
```

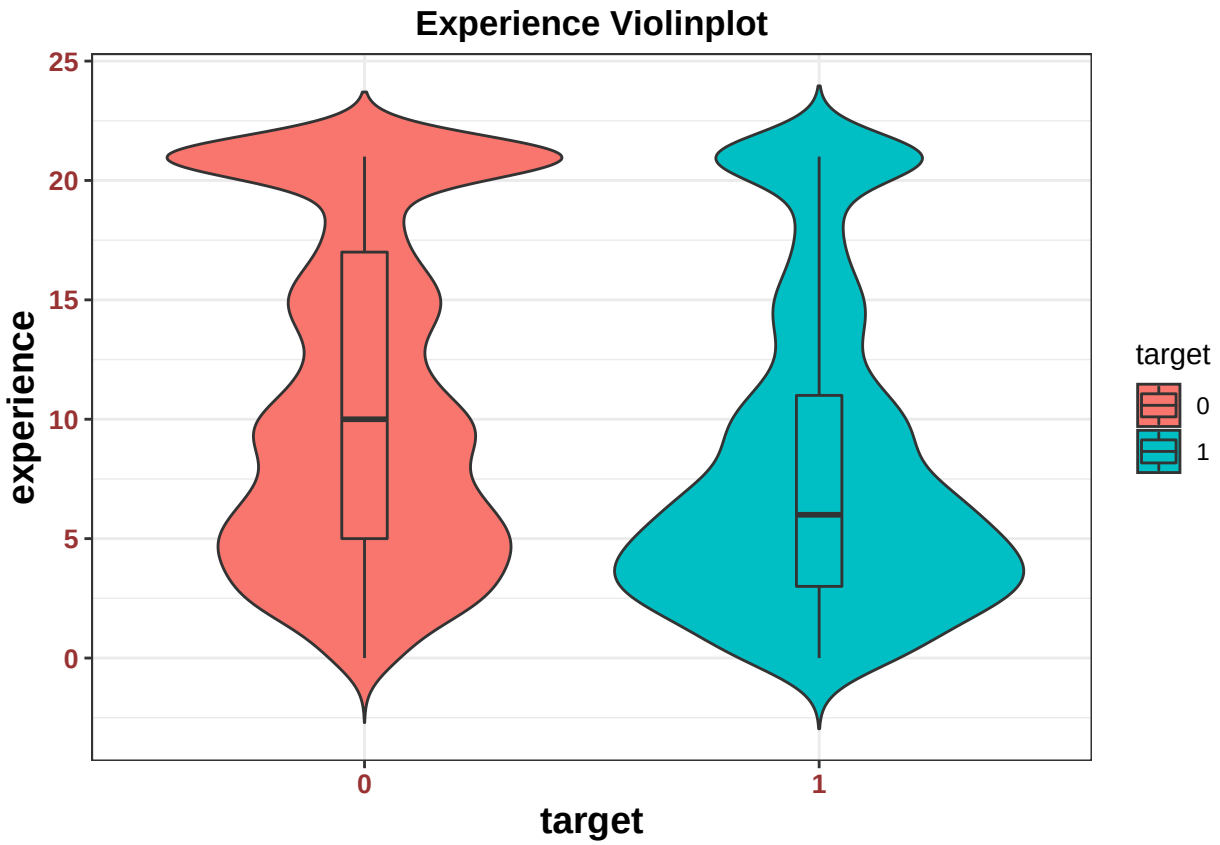
```
## Warning: `fun.y` is deprecated. Use `fun` instead.
```

Experience Violinplot

Most Data Scientists with less than 5 years' experience are likely to resign their jobs Candidates with more than 10 years' experience prefer to continue to work in the same company.

```
ggplot(results, aes(x=as.factor(target), y=experience, fill = as.factor(target))) +
  geom_violin(trim=FALSE) +
  labs(x='target', y = 'experience')+
  stat_summary(fun.y=mean, geom="point", shape=23, size=2)+
  geom_boxplot(width=0.1)+
  scale_fill_hue(name="target")+
  ggtitle("Experience Violinplot")+
  theme(plot.title=element_text(hjust=0.5, vjust=0.5, face='bold'))
```

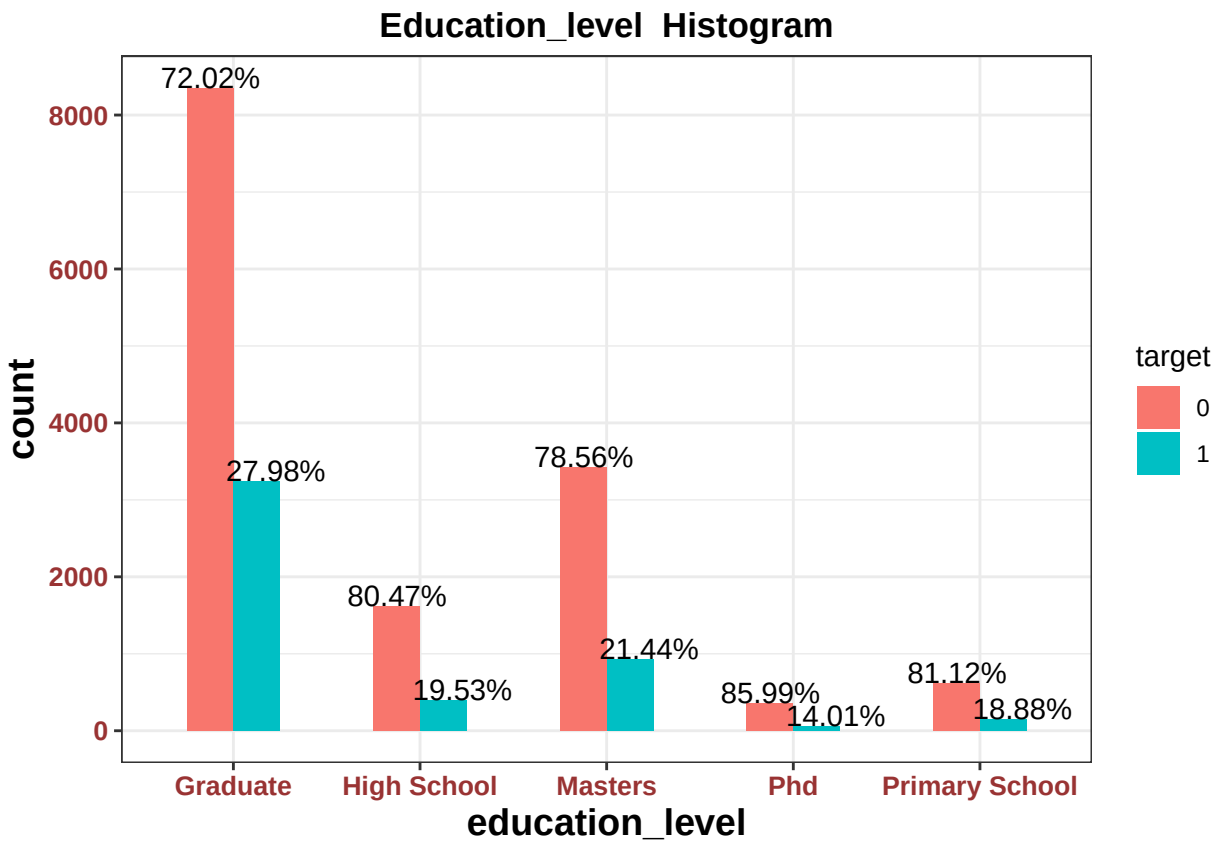
```
## Warning: `fun.y` is deprecated. Use `fun` instead.
```



Education_level Histogram

28 % of People with bachelor's degrees are more likely to stay in the company. This percentage is higher than that in other education level groups.

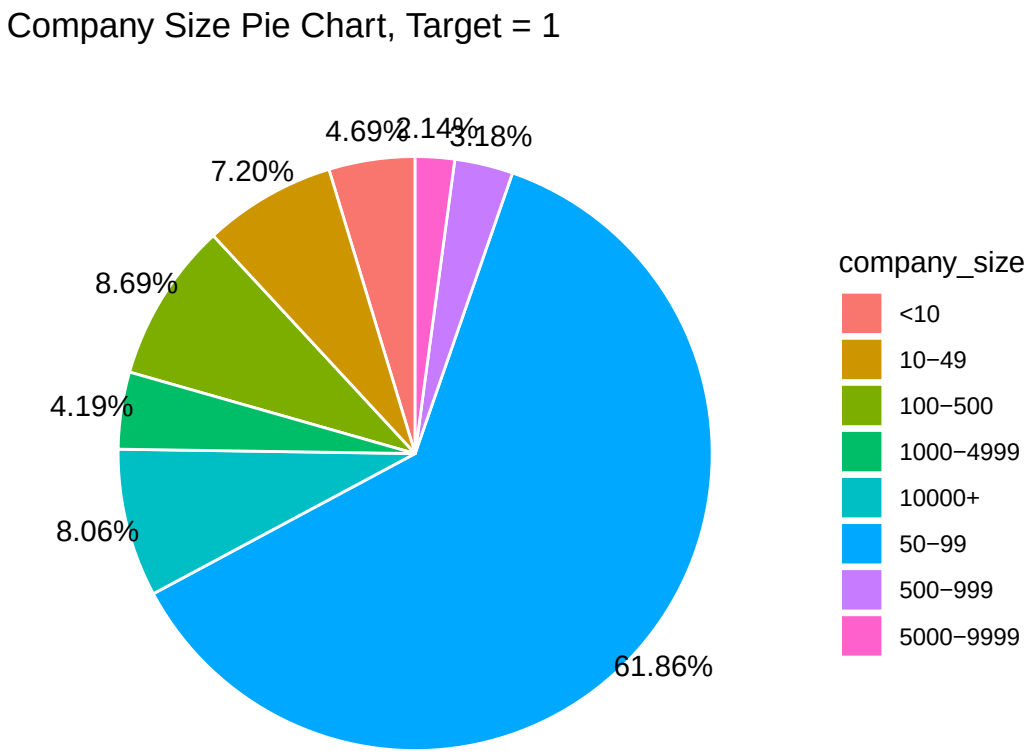
```
ggplot(results, aes(x = as.factor(education_level ), fill = as.factor(target)))+
  geom_bar(stat = 'count', width = 0.5, position = 'dodge')+
  labs(x='education_level', y = 'count')+
  ggtitle("Education_level  Histogram") +
  geom_text(stat='count', aes(label=scales::percent(..count../tapply(..count.., ..x.. ,sum)[..x..])), p
  scale_fill_hue(name="target")+
  theme(
    plot.title=element_text(hjust=0.5, vjust=0.5, face='bold')
  )
```



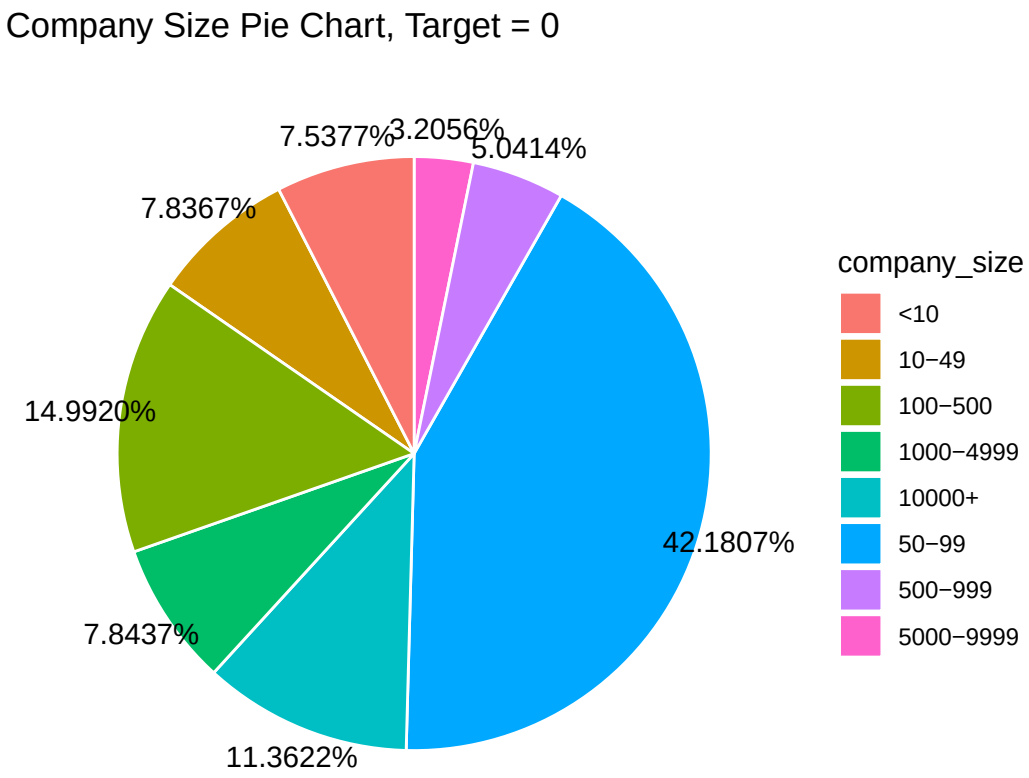
Company Size Pie Chart

For the company size about 50 - 99, people are willing to leave their jobs.

```
com_size <- results[target == 1, .N, by = company_size]
com_size[, prop := .(scales :: percent(N/sum(N))),]

ggplot(com_size, aes(x = "", y = N, fill = company_size)) +
  geom_bar(width = 1, stat = "identity", color = "white") +
  coord_polar("y", start = 0)+
  geom_text(aes(x = 1.6, label = prop), color = "black", position = position_stack(vjust = .5))+
  ggtitle("Company Size Pie Chart, Target = 1") +
  theme(
    plot.title=element_text(hjust=-5, vjust=0.5, face='bold')
  )+
  theme_void()
```



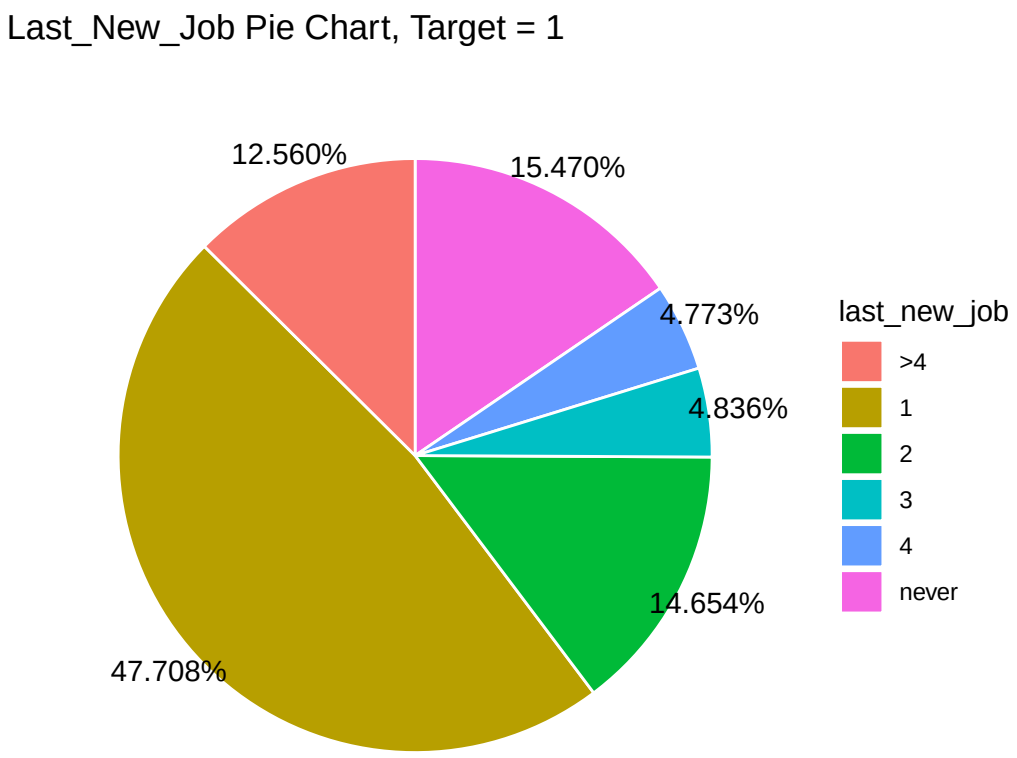

```
com_size <- results[target == 0, .N, by = company_size]
com_size[, prop := .(scales :: percent(N/sum(N)))]

ggplot(com_size, aes(x = "", y = N, fill = company_size)) +
  geom_bar(width = 1, stat = "identity", color = "white") +
  coord_polar("y")+
  geom_text(aes(x = 1.6, label = prop), color = "black", position = position_stack(vjust = 0.5))+
  ggtitle("Company Size Pie Chart, Target = 0") +
  theme(
    plot.title=element_text(hjust=0.5, vjust=0.5, face='bold')
  )+
  theme_void()
```



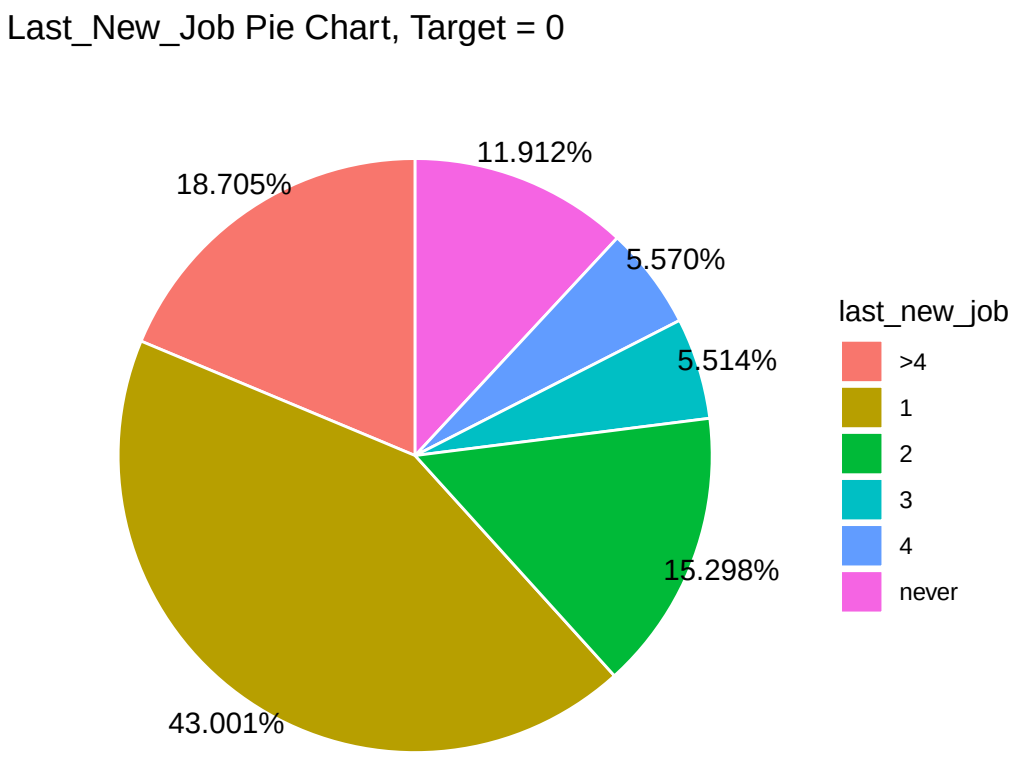
Last_New_Job Pie Chart

people whose last job was more than 4 years ago are willing to stay in the current company

```
com_size <- results[target == 1, .N, by = last_new_job]
com_size[, prop := .(scales :: percent(N/sum(N))),]

ggplot(com_size, aes(x = "", y = N, fill = last_new_job)) +
  geom_bar(width = 1, stat = "identity", color = "white") +
  coord_polar("y", start = 0)+
  geom_text(aes(x = 1.6, label = prop), color = "black", position = position_stack(vjust = .5))+
  ggtitle("Last_New_Job Pie Chart, Target = 1") +
  theme(
    plot.title=element_text(hjust=-5, vjust=0.5, face='bold')
  )+
  theme_void()
```




```
com_size <- results[target == 0, .N, by = last_new_job]
com_size[, prop := .(scales :: percent(N/sum(N))),]

ggplot(com_size, aes(x = "", y = N, fill = last_new_job)) +
  geom_bar(width = 1, stat = "identity", color = "white") +
  coord_polar("y", start = 0)+
  geom_text(aes(x = 1.6, label = prop), color = "black", position = position_stack(vjust = .5))+
  ggtitle("Last_New_Job Pie Chart, Target = 0") +
  theme(
    plot.title=element_text(hjust=-5, vjust=0.5, face='bold')
  )+
  theme_void()
```



Logistic regression (Generalized Linear Model)

Train and test datasets

```
logistic_data <- results[, c(1, 7, 11:43)]

# Total number of rows in the credit data frame
n <- nrow(results)

# Number of rows for the training set (70% of the dataset)
n_train <- round(0.7 * n)

# Create a vector of indices which is an 70% random sample
set.seed(123)
train_indices <- sample(1:n, n_train)

# Subset the credit data frame to training indices only
logistic_data_train <- logistic_data[train_indices, ]

# Exclude the training indices to create the test set
logistic_data_test <- logistic_data[-train_indices, ]
```

Model 1 summary

summary(model)$coef coef(model)

It can be seen that only 15 out of the 34 predictors are significantly associated to the outcome. These include: city index, experience, training hours and so on.

The coefficient estimate of the variable company_size_50_99 is b = 0.8950371, which is positive. The positive coefficient for this predictor suggests that all other variables being equal, the people from company size (50-99) is less likely to stay. However the coefficient for the variable city_development_index is b = -5.7581439, which is negative. This means that an increase in city_development_index will be associated with a decreased probability of leaving the company.

```
install.packages('caret', repos = 'https://github.com/topepo/caret/')
```

```
## Warning: unable to access index for repository https://github.com/topepo/caret/src/contrib:
##   cannot open URL 'https://github.com/topepo/caret/src/contrib/PACKAGES'
```

```
## Warning: package 'caret' is not available for this version of R
##
## A version of this package for your version of R might be available elsewhere,
## see the ideas at
## https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#Installing-packages
```

```
## Warning: unable to access index for repository https://github.com/topepo/caret/bin/macosx/big-sur-ar
##   cannot open URL 'https://github.com/topepo/caret/bin/macosx/big-sur-arm64/contrib/4.1/PACKAGES'
```

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.1.1
```

```
## Loading required package: lattice
```

```
library(data.table)
ctrl <- trainControl(method = "repeatedcv", number = 10, savePredictions = TRUE)

mod_fit <- train(as.factor(target) ~ ., data = logistic_data_train,method="glm", family="binomial",
                 trControl = ctrl, tuneLength = 5)
summary(mod_fit)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0488  -0.6885  -0.4853   0.4320   2.7520
##
## Coefficients:
##                                        Estimate Std. Error z value
## (Intercept)                           3.7102811  0.3069782  12.086
## city_development_index               -5.7581439  0.1803836 -31.922
```

```
## experience                                    -0.0221689  0.0043040  -5.151
## training_hours                                -0.0008811  0.0003716  -2.371
## gender_Male                                   -0.1076848  0.0909151  -1.184
## gender_Other                                  -0.0340240  0.0965996  -0.352
## relevent_experience_No_relevent_experience     0.4874858  0.0573494   8.500
## enrolled_university_no_enrollment             -0.2993169  0.0585413  -5.113
## enrolled_university_Part_time_course          -0.3400246  0.0991791  -3.428
## education_level_High_School                   -0.8903066  0.0831953 -10.701
## education_level_Masters                       -0.2165801  0.0564288  -3.838
## education_level_Phd                           -0.4724900  0.1852336  -2.551
## education_level_Primary_School                -0.8702386  0.1255322  -6.932
## major_discipline_Business_Degree              -0.0664563  0.2551690  -0.260
## major_discipline_Humanities                    0.0452474  0.2277620   0.199
## major_discipline_No_Major                     -0.0659505  0.2792759  -0.236
## major_discipline_Other                        -0.0551606  0.2472235  -0.223
## major_discipline_STEM                         -0.1506507  0.1969069  -0.765
## company_size_10_49                             0.4190967  0.1266953   3.308
## company_size_50_99                             0.8950371  0.1076932   8.311
## company_size_100_500                          -0.0178484  0.1237403  -0.144
## company_size_500_999                           0.0160644  0.1549252   0.104
## company_size_1000_4999                         0.0261925  0.1416417   0.185
## company_size_5000_9999                         0.1914405  0.1742758   1.098
## company_size_10000                             0.1979870  0.1264493   1.566
## company_type_Funded_Startup                   -0.4548826  0.1750622  -2.598
## company_type_NGO                               0.0027445  0.1996128   0.014
## company_type_Other                             0.4960505  0.2978521   1.665
## company_type_Public_Sector                     0.2229444  0.1718149   1.298
## company_type_Pvt_Ltd                           0.1325272  0.1370770   0.967
## last_new_job_1                                -0.0222766  0.0749623  -0.297
## last_new_job_2                                 0.0893078  0.0857626   1.041
## last_new_job_3                                -0.0457619  0.1166073  -0.392
## last_new_job_4                                 0.1318764  0.1143850   1.153
## last_new_job_never                            -0.4068810  0.0978097  -4.160
##                                               Pr(>|z|)
## (Intercept)                                    < 2e-16 ***
## city_development_index                         < 2e-16 ***
## experience                                    2.59e-07 ***
## training_hours                                0.017745 *
## gender_Male                                   0.236233
## gender_Other                                  0.724676
## relevent_experience_No_relevent_experience     < 2e-16 ***
## enrolled_university_no_enrollment             3.17e-07 ***
## enrolled_university_Part_time_course          0.000607 ***
## education_level_High_School                    < 2e-16 ***
## education_level_Masters                       0.000124 ***
## education_level_Phd                           0.010748 *
## education_level_Primary_School                4.14e-12 ***
## major_discipline_Business_Degree              0.794524
## major_discipline_Humanities                   0.842528
## major_discipline_No_Major                     0.813318
## major_discipline_Other                        0.823442
## major_discipline_STEM                         0.444220
## company_size_10_49                            0.000940 ***
## company_size_50_99                             < 2e-16 ***
```

```
## company_size_100_500                        0.885310
## company_size_500_999                        0.917414
## company_size_1000_4999                      0.853291
## company_size_5000_9999                      0.271990
## company_size_10000                          0.117409
## company_type_Funded_Startup                 0.009366 **
## company_type_NGO                            0.989030
## company_type_Other                          0.095828 .
## company_type_Public_Sector                  0.194430
## company_type_Pvt_Ltd                        0.333640
## last_new_job_1                              0.766336
## last_new_job_2                              0.297719
## last_new_job_3                              0.694729
## last_new_job_4                              0.248945
## last_new_job_never                          3.18e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 15086  on 13410  degrees of freedom
## Residual deviance: 12754  on 13376  degrees of freedom
## AIC: 12824
##
## Number of Fisher Scoring iterations: 4
```

calculate MSE

0.1587713

```
mod_fit_mse <- train(target ~ ., data = logistic_data_train,method="glm", family="binomial",
                 trControl = ctrl, tuneLength = 5)
```

```
## Warning in train.default(x, y, weights = w, ...): You are trying to do
## regression and your outcome only has two possible values Are you trying to do
## classification? If so, use a 2 level factor as your outcome column.
```

```
probabilities_mse_test = predict(mod_fit_mse, newdata=logistic_data_test)
head(probabilities_mse_test)
```

```
##         1         2         3         4         5         6
## 0.1967700 0.2082856 0.1141352 0.3500982 0.2934816 0.2610296
```

```
mse.logit.test = mean((logistic_data_test$target - probabilities_mse_test)^2)
print(mse.logit.test)
```

```
## [1] 0.1587713
```

```
probabilities_mse_train = predict(mod_fit_mse, newdata=logistic_data_train)
head(probabilities_mse_train)
```

```
##         1         2         3         4         5         6
## 0.1686561 0.4144857 0.1088706 0.0714473 0.1647828 0.5811898
```

```
mse.logit.train = mean((logistic_data_train$target - probabilities_mse_train)^2)
print(mse.logit.train)
```

```
## [1] 0.1545434
```

Predict the probabilities of looking for a new job

```
mod_fit <- train(as.factor(target)~ ., data = logistic_data_train,method="glm", family="binomial",
                 trControl = ctrl, tuneLength = 5)
```

```
probabilities = predict(mod_fit, newdata=logistic_data_test)
head(probabilities)
```

```
## [1] 0 0 0 0 0 0
## Levels: 0 1
```

Confusion Matrix and Statistics

Low sensitivity and High Specificity many false negative results, and thus more cases of candidates who leaving a job are missed

```
# The diagonal elements of the confusion matrix indicate correct predictions, while the off-diagonals r
confusionMatrix(data=probabilities, as.factor(logistic_data_test$target), positive='1' )
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 4022 1034
##          1  302  389
##
##                Accuracy : 0.7675
##                  95% CI : (0.7564, 0.7784)
##     No Information Rate : 0.7524
##     P-Value [Acc > NIR] : 0.003913
##
##                   Kappa : 0.246
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.27337
##             Specificity : 0.93016
##          Pos Pred Value : 0.56295
##          Neg Pred Value : 0.79549
##              Prevalence : 0.24761
##          Detection Rate : 0.06769
```

```
##    Detection Prevalence : 0.12024
##       Balanced Accuracy : 0.60176
##
##        'Positive' Class : 1
##
```

Assessing model accuracy

76.75% of the observations have been correctly predicted.

```
mean(probabilities == logistic_data_test$target) # model accuracy
```

```
## [1] 0.7675309
```

```
mean(probabilities != logistic_data_test$target) #test set error rate
```

```
## [1] 0.2324691
```

Varible Importance

From the logistic regression results, it shows that some variables - gender_male and Major_discipline_No_Major - are not statistically significant. Keeping them in the model may lead to overfitting. Therefore, they should be eliminated.

We plan to use variable importance function to select the top 10 most important features and train the model again.

```
library(data.table)
var_imp <- varImp(mod_fit)
var_imp <- setDT(data.frame(var_imp[1]), rownames(TRUE))
var_imp[1:10][order(-Overall)]
```

```
##                                          rn    Overall
##  1:                   city_development_index 100.000000
##  2:               education_level_High_School  33.495307
##  3: relevent_experience_No_relevent_experience  26.596966
##  4:                                experience  16.099473
##  5:          enrolled_university_no_enrollment  15.980882
##  6:                    education_level_Masters  11.985638
##  7:       enrolled_university_Part_time_course  10.701551
##  8:                             training_hours   7.387339
##  9:                                gender_Male   3.669015
## 10:                              gender_Other   1.060763
```

Model 2 summary

```
ctrl_2 <- trainControl(method = "repeatedcv", number = 10, savePredictions = TRUE)

mod_fit_2 <- train(as.factor(target) ~ city_development_index + experience + training_hours + relevent_
                   trControl = ctrl_2, tuneLength = 5)
summary(mod_fit_2)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8977  -0.6882  -0.5207   0.5211   2.4591
##
## Coefficients:
##                                           Estimate Std. Error z value
## (Intercept)                                4.0454606  0.1656772  24.418
## city_development_index                    -5.5939664  0.1725220 -32.425
## experience                                -0.0192115  0.0038620  -4.975
## training_hours                            -0.0008790  0.0003651  -2.408
## relevent_experience_No_relevent_experience  0.5266753  0.0518159  10.164
## enrolled_university_no_enrollment         -0.4139358  0.0560937  -7.379
## enrolled_university_Part_time_course      -0.4046972  0.0963337  -4.201
## education_level_High_School               -0.7873517  0.0794483  -9.910
## education_level_Masters                   -0.1623158  0.0546833  -2.968
## gender_Male                               -0.1482754  0.0886883  -1.672
## gender_Other                              -0.0785887  0.0942888  -0.833
##                                           Pr(>|z|)
## (Intercept)                                < 2e-16 ***
## city_development_index                     < 2e-16 ***
## experience                                6.54e-07 ***
## training_hours                             0.01606 *
## relevent_experience_No_relevent_experience  < 2e-16 ***
## enrolled_university_no_enrollment         1.59e-13 ***
## enrolled_university_Part_time_course      2.66e-05 ***
## education_level_High_School                < 2e-16 ***
## education_level_Masters                    0.00299 **
## gender_Male                                0.09455 .
## gender_Other                               0.40457
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 15086  on 13410  degrees of freedom
## Residual deviance: 13162  on 13400  degrees of freedom
## AIC: 13184
##
## Number of Fisher Scoring iterations: 4
```

calculate MSE

0.1587713

```
mod_fit_mse_2 <- train(target ~ city_development_index + experience + training_hours + relevent_experien
                trControl = ctrl, tuneLength = 5)
```

```
## Warning in train.default(x, y, weights = w, ...): You are trying to do
## regression and your outcome only has two possible values Are you trying to do
```

```
## classification? If so, use a 2 level factor as your outcome column.
```

```
probabilities_mse_test_2 = predict(mod_fit_mse_2, newdata=logistic_data_test)
head(probabilities_mse_test_2)
```

```
##         1         2         3         4         5         6
## 0.2602010 0.1372747 0.1353822 0.4769175 0.2073072 0.2860184
```

```
mse.logit.test.varimp = mean((logistic_data_test$target - probabilities_mse_test_2)^2)
print(mse.logit.test.varimp)
```

```
## [1] 0.1629941
```

```
probabilities_mse_train_2 = predict(mod_fit_mse_2, newdata=logistic_data_train)
```

```
mse.logit.train.varimp = mean((logistic_data_train$target - probabilities_mse_train_2)^2)
print(mse.logit.train.varimp)
```

```
## [1] 0.1589124
```

Predict the probabilities_2 of looking for a new job

```
probabilities_2 <- predict(mod_fit_2, logistic_data_test)
head(probabilities_2)
```

```
## [1] 0 0 0 0 0 0
## Levels: 0 1
```

Confusion Matrix and Statistics

10 important features from variable important function.

10 fold Cross Validation.

Low sensitivity and High Specificity.

many false negative results, and thus more cases of candidates who leaving a job are missed.

Sensitivity is better than the last model without feature selection.

```
confusionMatrix(data=probabilities_2, as.factor(logistic_data_test$target),positive='1' )
```

```
## Confusion Matrix and Statistics
##
##            Reference
## Prediction    0    1
##          0 4046 1056
##          1  278  367
##
##                Accuracy : 0.7679
```

```
##                  95% CI : (0.7567, 0.7787)
##     No Information Rate : 0.7524
##     P-Value [Acc > NIR] : 0.003249
##
##                   Kappa : 0.2371
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.25791
##             Specificity : 0.93571
##          Pos Pred Value : 0.56899
##          Neg Pred Value : 0.79302
##              Prevalence : 0.24761
##          Detection Rate : 0.06386
##    Detection Prevalence : 0.11223
##       Balanced Accuracy : 0.59681
##
##        'Positive' Class : 1
##
```

Assessing model accuracy

The Accuracy of model is $0.7679 > 0.7675$.

76.75% of the observations have been correctly predicted.

```
mean(probabilities_2== logistic_data_test$target) # model accuracy
```

```
## [1] 0.7678789
```

```
mean(probabilities_2 != logistic_data_test$target) #test set error rate
```

```
## [1] 0.2321211
```

ROC for 2 logistic regression models

AUC (area under the ROC curve) which are typical performance measurements for a binary classifier. As a rule of thumb, a model with good predictive ability should have an AUC closer to 1 (1 is ideal) than to 0.5. Logistic regression model without feature selections has a slightly better performance.

MSE_test for both : 0.1587713

```
library(pROC)
```

```
## Warning: package 'pROC' was built under R version 4.1.1
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
par(pty = 's')
roc(logistic_data_test$target, as.numeric(probabilities), plot = TRUE, legacy.axes = TRUE, ylab = "True
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

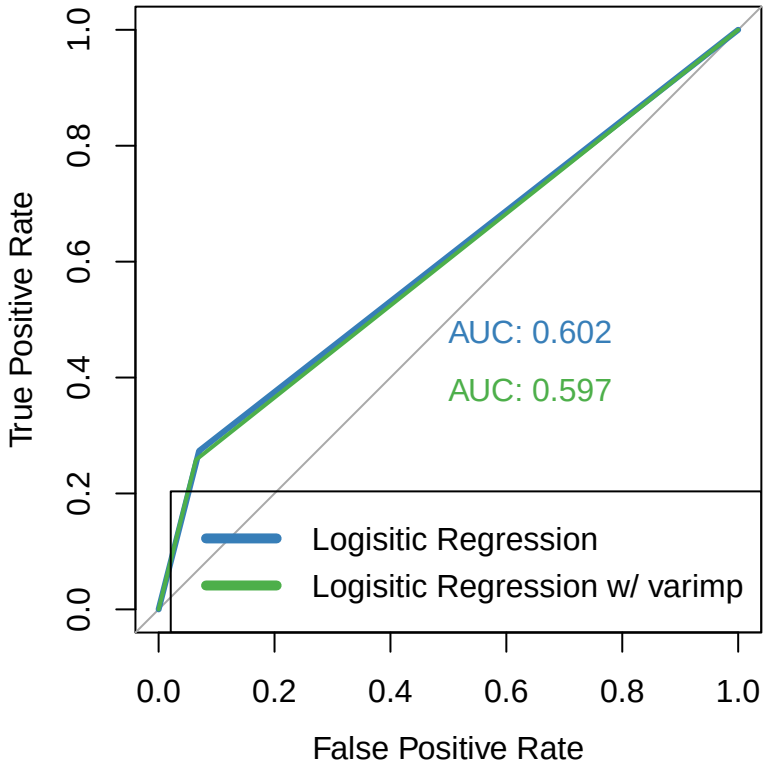
```
##
## Call:
## roc.default(response = logistic_data_test$target, predictor = as.numeric(probabilities),     plot = T
##
## Data: as.numeric(probabilities) in 4324 controls (logistic_data_test$target 0) < 1423 cases (logisti
## Area under the curve: 0.6018
```

```
roc(logistic_data_test$target, as.numeric(probabilities_2), plot = TRUE, legacy.axes = TRUE, ylab = "Tru
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
##
## Call:
## roc.default(response = logistic_data_test$target, predictor = as.numeric(probabilities_2),     plot =
##
## Data: as.numeric(probabilities_2) in 4324 controls (logistic_data_test$target 0) < 1423 cases (logist
## Area under the curve: 0.5968
```

```
legend("bottomright", legend=c("Logisitic Regression", "Logisitic Regression w/ varimp"), col=c("#377eb8
```



Lasso Linear Regression

10-fold Cross Validation

Tune a hyperparameter (lambda) : 76 times, lambda that minimizes training MSE is 0.0009059394

MSE_test = 0.1591651

It can be seen that only 9 out of the 34 predictors are significantly associated to the outcome. These include: city index, experience, training hours and company size_50_99.

Company_Size_50_99 (0.100476835) → the people from company size (50-99) is less likely to stay.

City_Development_Index (city_development_index) → a decreased probability of leaving the company.

Train and test datasets

```
lasso_data_x <- model.matrix( ~ -1 + city_development_index+experience+training_hours+gender_Male+gende

lasso_data_y <- results$target

# Total number of rows in the credit data frame
n <- nrow(results)

# Number of rows for the training set (70% of the dataset)
```

```
n_train <- round(0.7 * n)

# Create a vector of indices which is an 70% random sample
set.seed(123)
train_indices <- sample(1:n, n_train)

# Subset the credit data frame to training indices only
x_train <- lasso_data_x[train_indices, ]
y_train <- lasso_data_y[train_indices]

# Exclude the training indices to create the test set
x_test <- lasso_data_x[-train_indices, ]
y_test <- lasso_data_y[-train_indices]
```

Fits 100 different Lasso regressions for 100 decreasing values of

```
library(glmnet)
```

```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.1-2
```

```
fit.lasso <- cv.glmnet(x_train, y_train, alpha = 1, nfolds = 10)
fit.lasso$lambda
```

```
##  [1] 0.1511197929 0.1376947270 0.1254623069 0.1143165814 0.1041610114
##  [6] 0.0949076342 0.0864763016 0.0787939853 0.0717941448 0.0654161509
## [11] 0.0596047603 0.0543096377 0.0494849192 0.0450888153 0.0410832493
## [16] 0.0374335266 0.0341080353 0.0310779714 0.0283170901 0.0258014779
## [21] 0.0235093457 0.0214208402 0.0195178718 0.0177839579 0.0162040801
## [26] 0.0147645543 0.0134529120 0.0122577923 0.0111688439 0.0101766346
## [31] 0.0092725704 0.0084488208 0.0076982508 0.0070143595 0.0063912231
## [36] 0.0058234444 0.0053061057 0.0048347259 0.0044052222 0.0040138744
## [41] 0.0036572928 0.0033323890 0.0030363488 0.0027666079 0.0025208301
## [46] 0.0022968865 0.0020928374 0.0019069155 0.0017375104 0.0015831548
## [51] 0.0014425117 0.0013143629 0.0011975985 0.0010912072 0.0009942673
## [56] 0.0009059394 0.0008254582 0.0007521268 0.0006853099 0.0006244289
## [61] 0.0005689563 0.0005184118 0.0004723575 0.0004303946 0.0003921595
## [66] 0.0003573212 0.0003255777 0.0002966543 0.0002703003 0.0002462876
## [71] 0.0002244081 0.0002044723 0.0001863075 0.0001697565 0.0001546758
## [76] 0.0001409348
```

Predict the results

```
yhat.train.lasso <- predict(fit.lasso, x_train, s = fit.lasso$lambda.min)  # Select lambda that minimiz
yhat.test.lasso <- predict(fit.lasso, x_test, s = fit.lasso$lambda.min)


yhat.train.lasso_all <- predict(fit.lasso, x_train, s = fit.lasso$lambda)
yhat.test.lasso_all <- predict(fit.lasso, x_test, s = fit.lasso$lambda)
```

Compute train and test MSEs

```
mse_train <- colMeans((yhat.train.lasso_all - y_train) ** 2)
mse_test <- colMeans((yhat.train.lasso_all - y_test) ** 2)
```

```
## Warning in yhat.train.lasso_all - y_test: longer object length is not a multiple
## of shorter object length
```

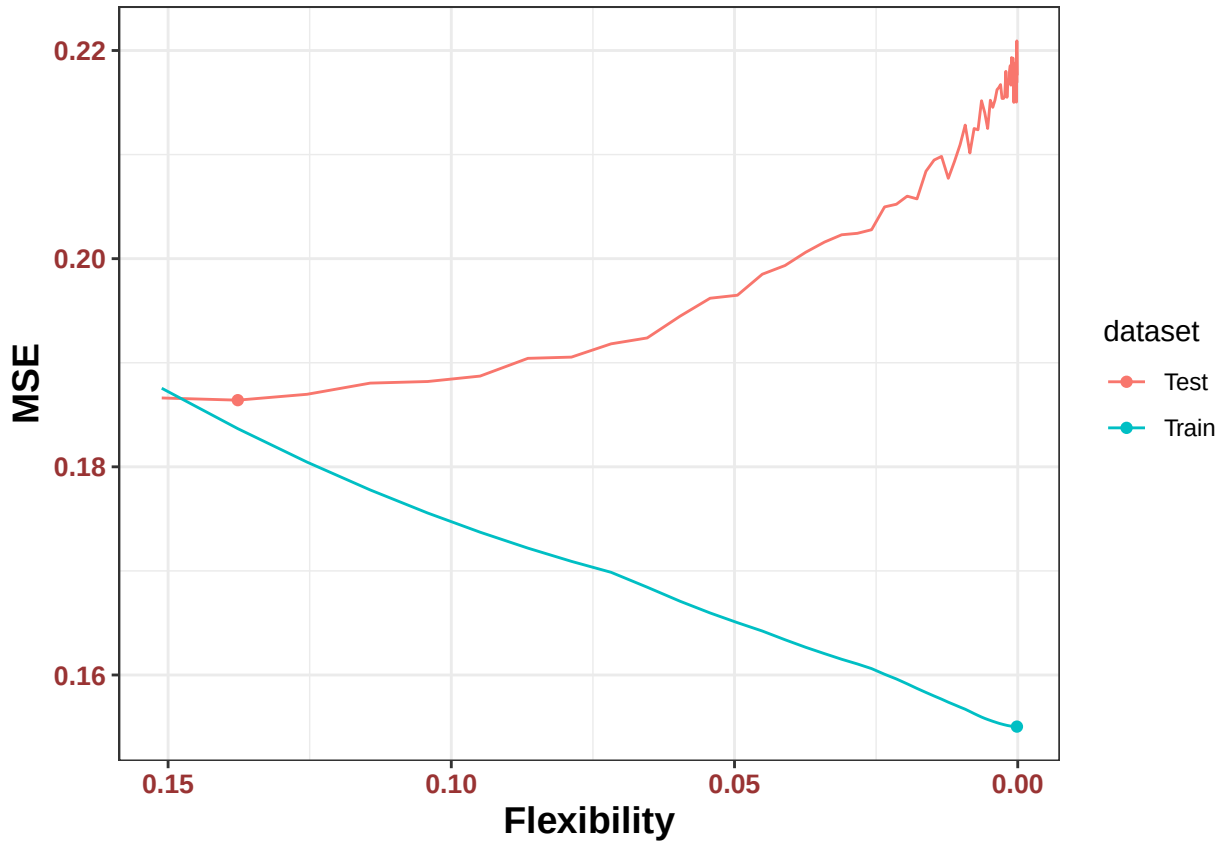
```
mse.lassolinear.train <- mean((y_train - yhat.train.lasso)^2)
mse.lassolinear.test <- mean((y_test - yhat.test.lasso)^2)
```

Aggregate all MSEs

```
dd_mse <- data.table(
  lambda = fit.lasso$lambda,
  mse = mse_train,
  dataset = "Train",
  is_min = mse_train == min(mse_train)
)
dd_mse <- rbind(dd_mse, data.table(
  lambda = fit.lasso$lambda,
  mse = mse_test,
  dataset = "Test",
  is_min = mse_test == min(mse_test)
))
```

Plot the MSE with lambda

```
ggplot(dd_mse, aes(lambda, mse, color=dataset)) +
  geom_line() +
  geom_point(data=dd_mse[is_min==TRUE]) +
  scale_y_continuous("MSE") +
  scale_x_reverse("Flexibility")
```



Compute test MSE:

```
print(mse.lassolinear.test)
```

```
## [1] 0.1591651
```

```
print(mse.lassolinear.train)
```

```
## [1] 0.1550688
```

Summary of the lasso linear regression

```
coef(fit.lasso)
```

```
## 35 x 1 sparse Matrix of class "dgCMatrix"
##                                                      s1
## (Intercept)                                1.101849227
## city_development_index                    -1.032391702
## experience                                -0.002407587
## training_hours                             .
```

```
## gender_Male                                .
## gender_Other                               .
## relevent_experience_No_relevent_experience  0.055454648
## enrolled_university_no_enrollment          -0.028594675
## enrolled_university_Part_time_course        .
## education_level_High_School                -0.078890639
## education_level_Masters                     .
## education_level_Phd                         .
## education_level_Primary_School             -0.068534902
## major_discipline_Business_Degree            .
## major_discipline_Humanities                 .
## major_discipline_No_Major                   .
## major_discipline_Other                      .
## major_discipline_STEM                       .
## company_size_10_49                          .
## company_size_50_99                          0.100476835
## company_size_100_500                        .
## company_size_500_999                        .
## company_size_1000_4999                      .
## company_size_5000_9999                      .
## company_size_10000                          .
## company_type_Funded_Startup                -0.024656158
## company_type_NGO                            .
## company_type_Other                          .
## company_type_Public_Sector                  .
## company_type_Pvt_Ltd                        .
## last_new_job_1                              .
## last_new_job_2                              .
## last_new_job_3                              .
## last_new_job_4                              .
## last_new_job_never                         -0.012386896
```

Randomforest

Preparation

```
data <- read.csv("/Users/moonqj/Desktop/Boston University/Semester/Fall 2021/BA 810/Project/data/cleaned
data$target <- factor(data$target)
data$gender <- factor(data$gender)
data$relevent_experience <- factor(data$relevent_experience)
data$enrolled_university <- factor(data$enrolled_university)
data$education_level <- factor(data$education_level)
data$major_discipline <- factor(data$major_discipline)
data$experience <- factor(data$experience)
data$company_size <- factor(data$company_size)
data$company_type <- factor(data$company_type)
data$last_new_job <- factor(data$last_new_job)
```

##set train and test

```r
set.seed(123)
test_size <- floor(0.3*nrow(data))
sam <- sample(nrow(data), test_size, replace = FALSE)
train <- data[-sam, 1:12]
test <- data[sam, 1:12]
```

##set the model

```r
library(randomForest)
```

```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```r
model <- randomForest(target~., data = train, importance = TRUE)
print(model)
```

```
##
## Call:
##  randomForest(formula = target ~ ., data = train, importance = TRUE)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 3
##
##         OOB estimate of  error rate: 22.41%
## Confusion matrix:
##      0    1 class.error
## 0 8960 1099   0.1092554
## 1 1906 1446   0.5686158
```

##predict and accuracy

```r
pred <- predict(model, test[, 1:11])
table(test=test[, 12], predict = pred)
```

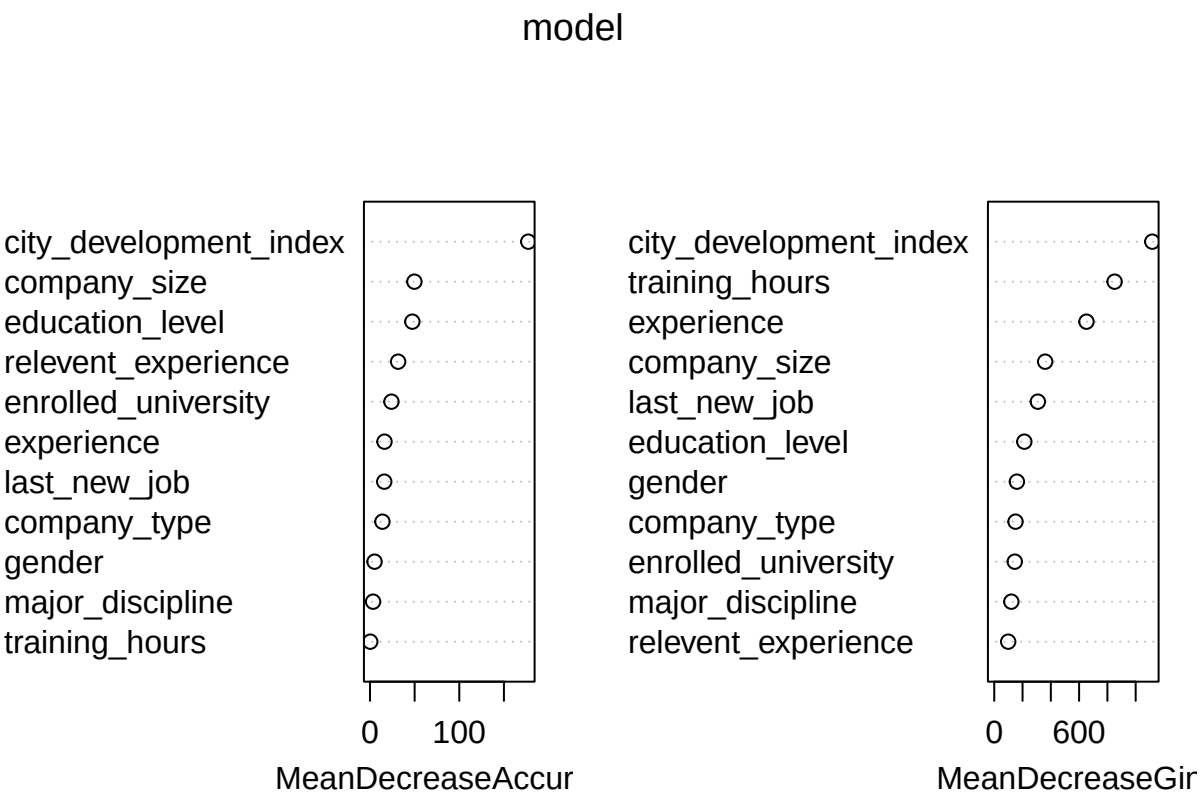
```
##     predict
## test    0    1
##    0 3870  452
##    1  811  614
```

```r
accuracy <- mean(test[, 12] == pred)
print(accuracy)
```

```
## [1] 0.7802332
```

##variable importance

```
varImpPlot(model)
```

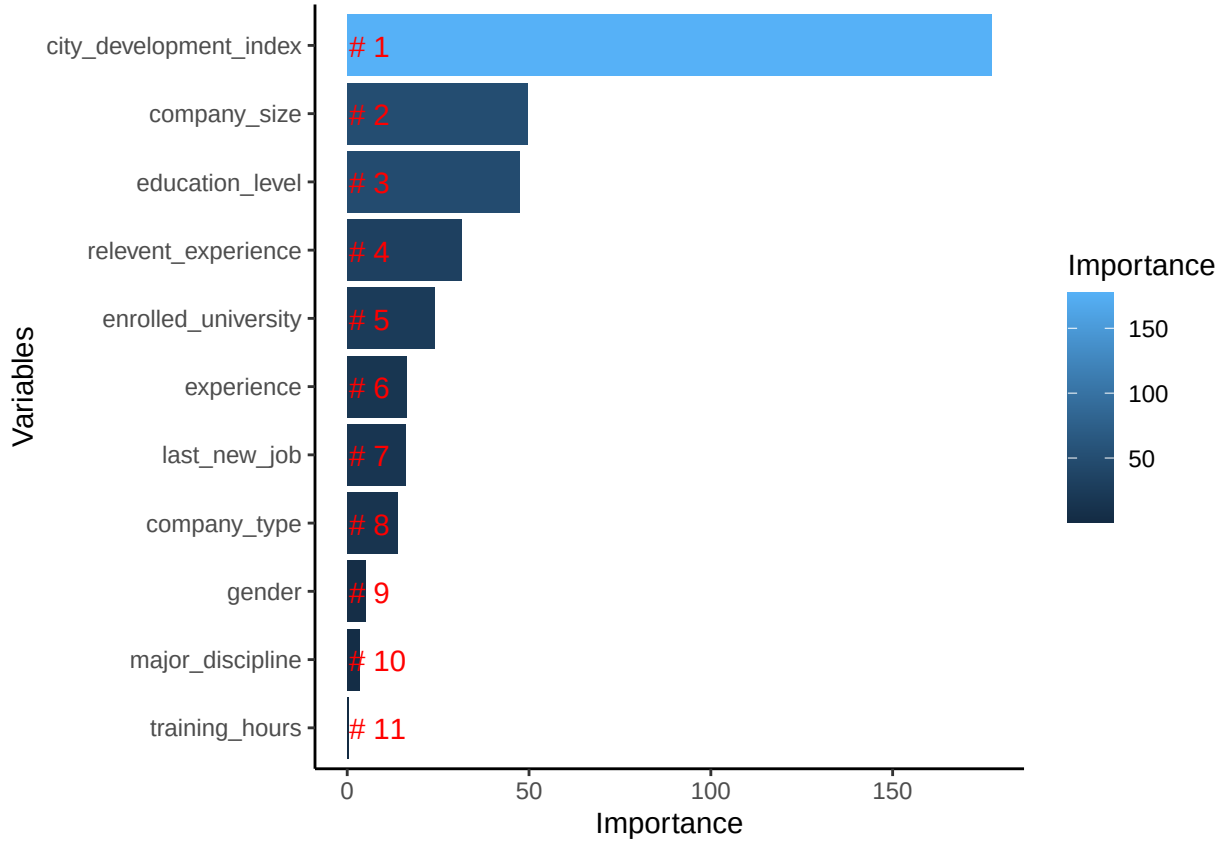



```
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following object is masked from 'package:randomForest':
##
##     combine

## The following objects are masked from 'package:data.table':
##
##     between, first, last

## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
importance <-  importance(model)
varImportance <-  data.frame(Variables = row.names(importance),
                             Importance =round(importance[, "MeanDecreaseAccuracy"],2))
rankImportance <- varImportance %>%
  mutate(Rank=paste('#',dense_rank(desc(Importance))))

ggplot(rankImportance,aes(x=reorder(Variables,Importance),
                          y=Importance,fill=Importance)) +
  geom_bar(stat='identity') +
  geom_text(aes(x = Variables, y = 0.5, label = Rank),
            hjust=0, vjust=0.55, size = 4, colour = 'red') +
  labs(x = 'Variables') +
  coord_flip() +
  theme_classic()
```



Decision Tree

```
library(data.table)
library(rpart)
```

```
library(rpart.plot)
dd <- fread("/Users/moonqj/Desktop/Boston University/Semester/Fall 2021/BA 810/Project/data/cleaned_data
```

create formula

```
f1 <- as.formula(target ~ city_development_index + gender +
                   relevent_experience + enrolled_university    +
                   education_level  + major_discipline  +
                   experience   + company_size  +
                   company_type + last_new_job  +training_hours)
```

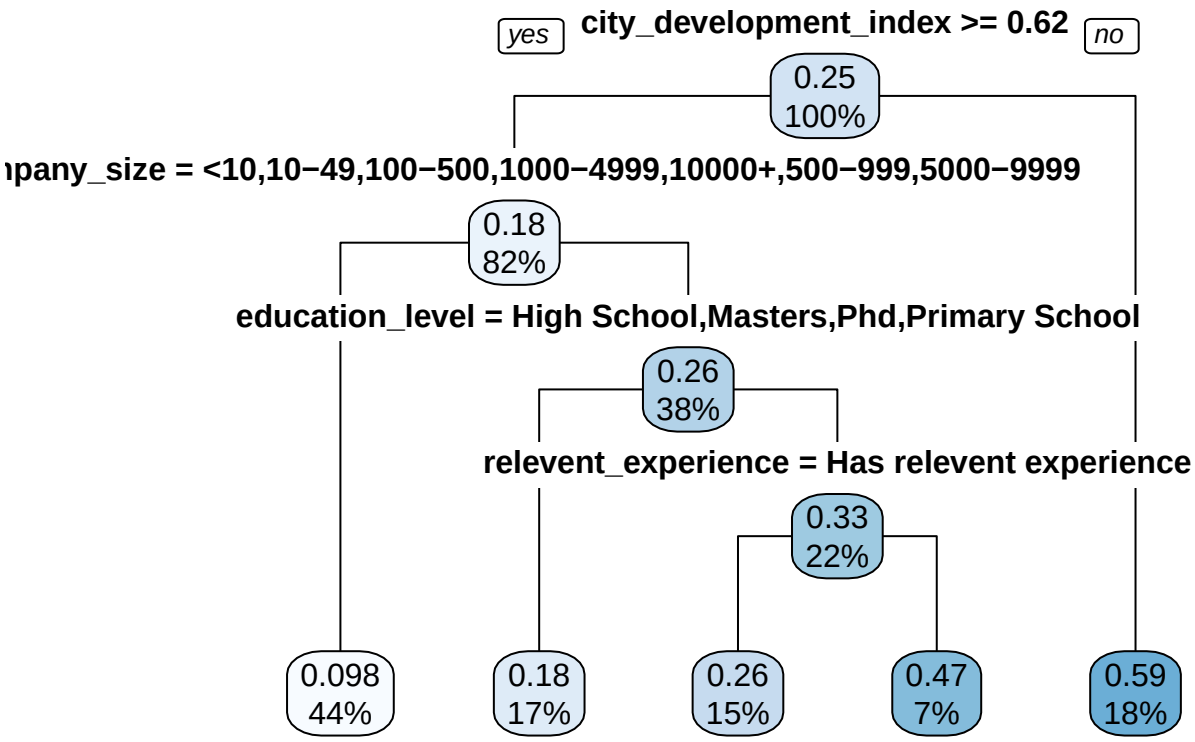
split train test data

```
set.seed(123)
test_size <- floor(0.3*nrow(data))
sam <- sample(nrow(data), test_size, replace = FALSE)
dd.train <- dd[-sam, c(1:12)]
dd.test <- dd[sam, c(1:12)]

x1.train <- model.matrix(f1, dd.train)[, -1]
y.train <- dd.train$target

x1.test <- model.matrix(f1, dd.test)[, -1]
y.test <- dd.test$target
```

fit the tree

```
fit.tree <- rpart(f1, dd.train, control = rpart.control(cp = 0.005))
```

```
rpart.plot(fit.tree, type = 1)
```



calculate mse train and mse test

```
ypred.train <- predict(fit.tree, dd.train)
mse.decisiontree.train <- mean((ypred.train - y.train) ^ 2)
print(mse.decisiontree.train)
```

```
## [1] 0.1527465
```

```
ypred.test <- predict(fit.tree, dd.test)
mse.decisiontree.test <- mean((ypred.test - y.test) ^ 2)
print(mse.decisiontree.test)
```

```
## [1] 0.1495115
```

Feature importance

```
df <- data.frame(Feature_Importance = fit.tree$variable.importance)
df
```

```
##                         Feature_Importance
```

```
## city_development_index        335.9171943
## company_size                   75.8576799
## relevent_experience            46.0503465
## education_level                36.0739912
## enrolled_university            19.1108386
## last_new_job                   18.5833027
## experience                     10.2630228
## company_type                    0.1955906
```

Boosting tree

```
install.packages(c("gbm"), repos= 'https://github.com/gbm-developers/gbm.git')
```

```
## Warning: unable to access index for repository https://github.com/gbm-developers/gbm.git/src/contrib
##   cannot open URL 'https://github.com/gbm-developers/gbm.git/src/contrib/PACKAGES'
```

```
## Warning: package 'gbm' is not available for this version of R
##
## A version of this package for your version of R might be available elsewhere,
## see the ideas at
## https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#Installing-packages
```

```
## Warning: unable to access index for repository https://github.com/gbm-developers/gbm.git/bin/macosx/
##   cannot open URL 'https://github.com/gbm-developers/gbm.git/bin/macosx/big-sur-arm64/contrib/4.1/PA
```

```
library(ggthemes)
library(scales)
library(gbm)
```

```
## Loaded gbm 2.1.8
```

Load and split data

```
dd_gbm <- fread("/Users/moonqj/Desktop/Boston University/Semester/Fall 2021/BA 810/Project/data/cleaned

set.seed(123)
test_size <- floor(0.3*nrow(data))
sam <- sample(nrow(data), test_size, replace = FALSE)

dd_gbm.train <- dd_gbm[-sam, c(1:12)]
dd_gbm.test <- dd_gbm[sam, c(1:12)]

x1gbm.train <- model.matrix(f1, dd_gbm.train)[, -1]
ygbm.train <- dd_gbm.train$target

x1gbm.test <- model.matrix(f1, dd_gbm.test)[, -1]
ygbm.test <- dd_gbm.test$target
```

Fit the tree

```
fit_gbm <- gbm(f1, data = dd_gbm.train,
                 distribution = "gaussian",
                 n.trees = 100,
                 interaction.depth = 2,
                 shrinkage = 0.005)
```

Get relative feature influence

```
relative.influence(fit_gbm)
```

```
## n.trees not given. Using 100 trees.
```

```
## city_development_index                 gender    relevent_experience
##            10644.01285                0.00000               40.07430
##    enrolled_university        education_level       major_discipline
##               29.24479                0.00000                0.00000
##             experience           company_size           company_type
##                0.00000             2324.76219                0.00000
##           last_new_job         training_hours
##                0.00000                0.00000
```

```
df2 <- data.frame(Relative_Influence = relative.influence(fit_gbm))
```

```
## n.trees not given. Using 100 trees.
```

```
df2
```

```
##                         Relative_Influence
## city_development_index        10644.01285
## gender                            0.00000
## relevent_experience              40.07430
## enrolled_university              29.24479
## education_level                   0.00000
## major_discipline                  0.00000
## experience                        0.00000
## company_size                   2324.76219
## company_type                      0.00000
## last_new_job                      0.00000
## training_hours                    0.00000
```

Calculate MSE train

```
yhat.gbm <- predict(fit_gbm, dd_gbm.train, n.trees = 100)
mse.gbm.train <- mean((yhat.gbm - ygbm.train) ^ 2)
print(mse.gbm.train)
```

```
## [1] 0.1681107
```

Calculate MSE test

```
yhat.gbm_test <- predict(fit_gbm, dd_gbm.test, n.trees = 100)
mse.gbm.test <- mean((yhat.gbm_test - ygbm.test) ^ 2)
print(mse.gbm.test)
```

```
## [1] 0.1668571
```

MSE Summary

```
MSE_Test_Value <- c(mse.lassolinear.test, mse.logit.test, mse.logit.test.varimp, mse.decisiontree.test,
MSE_Train_Value <- c(mse.lassolinear.train, mse.logit.train, mse.logit.train.varimp, mse.decisiontree.t

MSE_Test_Name <- c('mse.lassolinear.test', 'mse.logit.test', 'mse.logit.test.varimp', 'mse.decisiontree
MSE_Train_Name <- c('mse.lassolinear.train', 'mse.logit.train', 'mse.logit.train.varimp', 'mse.decision

MSE_Table <- data.table(MSE_Test_Name, MSE_Test_Value, MSE_Train_Name, MSE_Train_Value)
MSE_Table
```

```
##            MSE_Test_Name MSE_Test_Value         MSE_Train_Name MSE_Train_Value
## 1:  mse.lassolinear.test      0.1591651  mse.lassolinear.train       0.1550688
## 2:        mse.logit.test      0.1587713        mse.logit.train       0.1545434
## 3: mse.logit.test.varimp      0.1629941 mse.logit.train.varimp       0.1589124
## 4: mse.decisiontree.test      0.1495115 mse.decisiontree.train       0.1527465
## 5:          mse.gbm.test      0.1668571          mse.gbm.train       0.1681107
```

```
setorder(MSE_Table, cols = "MSE_Test_Value")
MSE_Table
```

```
##            MSE_Test_Name MSE_Test_Value         MSE_Train_Name MSE_Train_Value
## 1: mse.decisiontree.test      0.1495115 mse.decisiontree.train       0.1527465
## 2:        mse.logit.test      0.1587713        mse.logit.train       0.1545434
## 3:  mse.lassolinear.test      0.1591651  mse.lassolinear.train       0.1550688
## 4: mse.logit.test.varimp      0.1629941 mse.logit.train.varimp       0.1589124
## 5:          mse.gbm.test      0.1668571          mse.gbm.train       0.1681107
```

Conclusion

- Top factors for employees leaving:
  - Employees in less developed cities
  - Employees in size 50-99 companies
  - Employees with relevant experience
- Irrelevant factors:
  - Training hours

  - Major (Field of study)

- The Best model is decision tree with MSE_test 0.1495.
- If a 50-99 company in less developed cities and wants to retain their employees, it needs to consider provide them with some incentives or bonus. In addition, more team building is a good way to bond the current employees.