HR Data Analytics: Predicting Employee Churn

Kevin Ayala

3/26/2022

Segment 1

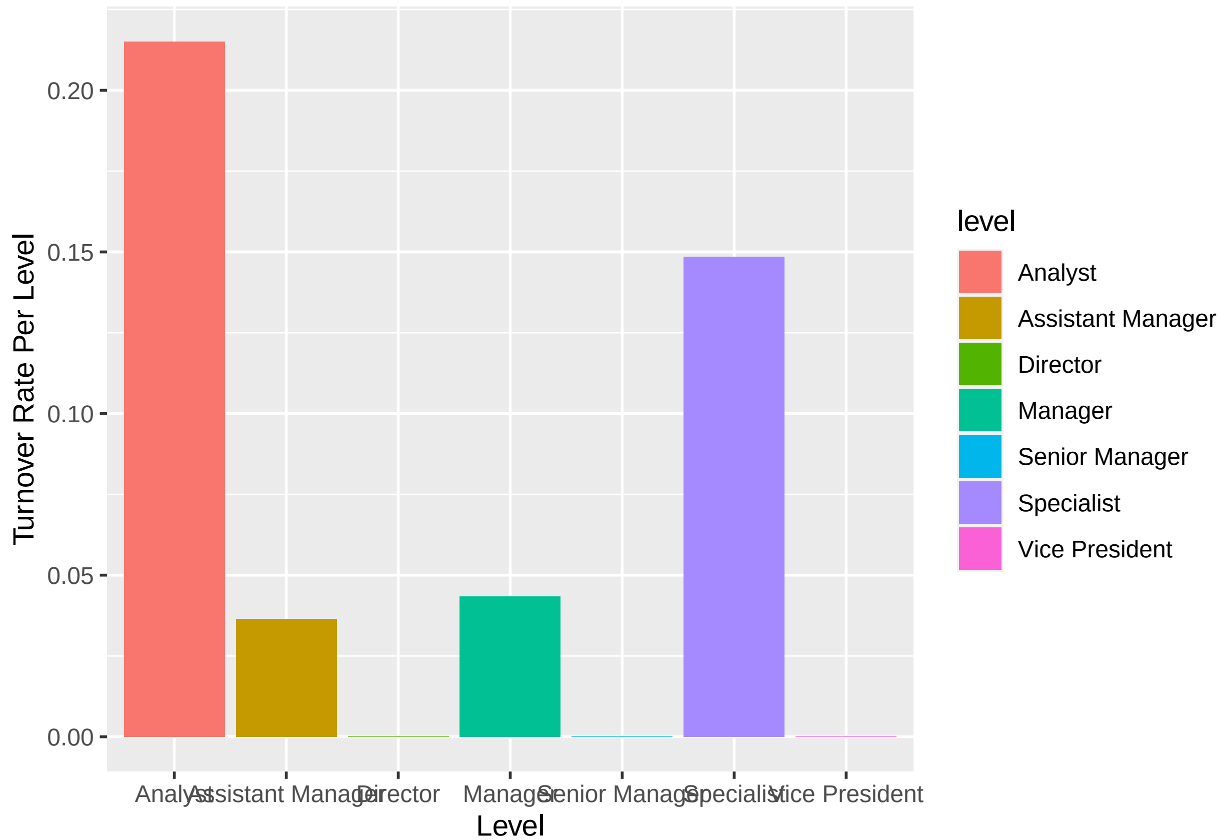
After loading the data, the proportion of employees who have left is 1881 active employees with 410 employees who have left the organization. Either voluntary or involuntary is unknown at this point. General turnover is 17.9%, meaning that employees across the organization have a 17.9% chance of leaving the company/organisation.

Level that have high Turnover

```r
# Level wise turnover rate per group
df_level <- org %>%
  group_by(level) %>%
  summarise(turnover_level = mean(turnover))

#results
df_level
```

```
## # A tibble: 7 x 2
##   level             turnover_level
##   <chr>                      <dbl>
## 1 Analyst                   0.215
## 2 Assistant Manager         0.0365
## 3 Director                  0
## 4 Manager                   0.0435
## 5 Senior Manager            0
## 6 Specialist                0.149
## 7 Vice President            0
```

```r
# Visualizing the results using ggplot2
ggplot(df_level, aes(x = level, y = turnover_level, fill = level)) +
  ylab("Turnover Rate Per Level") +
  xlab("Level")+
  geom_col()
```



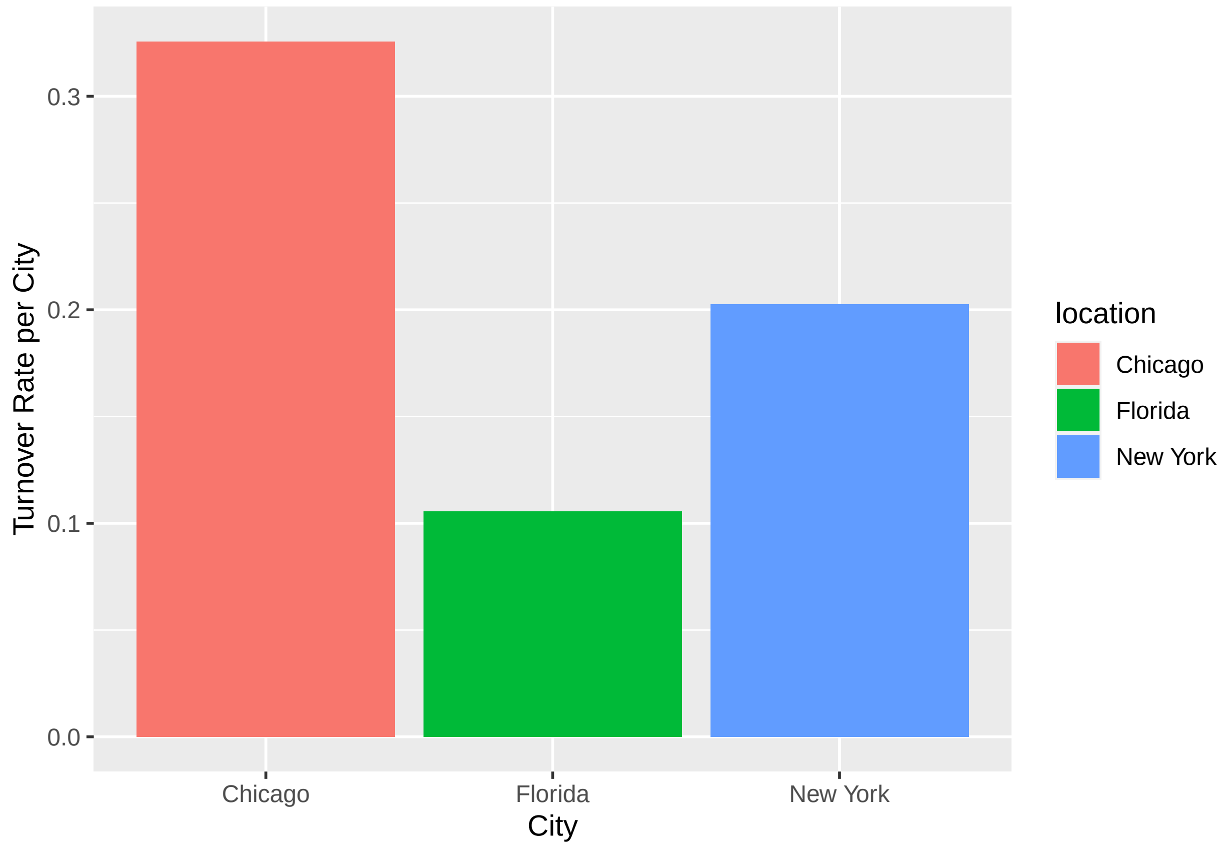
After doing a quick group by, we can now see the turnover rate based on the employees role within the company varies per role/specialization. Analyst has the highest turnover rate, followed by specialist.

Turnover rate and Locations

```
# Calculating location wise turnover rate
df_location <- org %>%
  group_by(location) %>%
  summarize(turnover_location = mean(turnover))

# results
df_location
```

```
## # A tibble: 3 x 2
##   location turnover_location
##   <chr>                <dbl>
## 1 Chicago              0.326
## 2 Florida              0.106
## 3 New York             0.203
```

```
# Visualizing the results with ggplot
ggplot(df_location, aes(x = location, y = turnover_location, fill = location)) +
  ylab("Turnover Rate per City") +
  xlab("City") +
  geom_col()
```



Chicago has the highest turnover rate, could it be people leave due to the bad winter? Could play a role.

Filtering the dataset

```
# Counting the number of employees across levels
org %>%
  count(level)
```

```
## # A tibble: 7 x 2
##   level                  n
##   <chr>              <int>
## 1 Analyst             1604
## 2 Assistant Manager    192
## 3 Director               1
## 4 Manager              138
## 5 Senior Manager         5
## 6 Specialist           350
## 7 Vice President         1
```

```
# filtering the employees at Analyst and Specialist level
org2 <- org %>%
  filter(level %in% c('Analyst','Specialist'))
# Validating the results
org2 %>%
  count(level)
```

```
## # A tibble: 2 x 2
##   level          n
##   <chr>      <int>
## 1 Analyst     1604
```

```
## 2 Specialist   350
```

High level counts between orginization and getting counts per employee level.

Combining HR datasets, Part 1

```r
#read in data set
rating <- read_csv("/Users/kevinlorenzoayala/Downloads/employee_data/rating.csv")
```

```
## Rows: 1954 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): emp_id, rating
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
# Viewing the structure of rating dataset
glimpse(rating)
```

```
## Rows: 1,954
## Columns: 2
## $ emp_id <chr> "E8", "E9", "E12", "E15", "E34", "E37", "E47", "E50", "E53", "E~
## $ rating <chr> "Acceptable", "Acceptable", "Acceptable", "Acceptable", "Accept~
```

```r
# merging datasets
org3 <- left_join(org2, rating, by = "emp_id")

# Calculatingnrating wise turnover rate
df_rating <- org3 %>%
  group_by(rating) %>%
  summarise(turnover_rating = mean(turnover))

# result
df_rating
```

```
## # A tibble: 5 x 2
##   rating        turnover_rating
##   <chr>                   <dbl>
## 1 Above Average          0.131
## 2 Acceptable             0.221
## 3 Below Average          0.385
## 4 Excellent              0.0305
## 5 Unacceptable           0.633
```

Once recieving employee ratings, we are able to calculate employee turnover per rating given to them during performance reviews. As expected, the employees with "Unacceptable" performance have the highest turnover rating, and in contrast the employees with an excellent rating have the least turnover.

Combining HR datasets

```r
survey <- read_csv("/Users/kevinlorenzoayala/Downloads/employee_data/survey.csv")
```

```
## Rows: 350 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): mgr_id
## dbl (4): mgr_effectiveness, career_satisfaction, perf_satisfaction, work_sat...
##
```

```
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

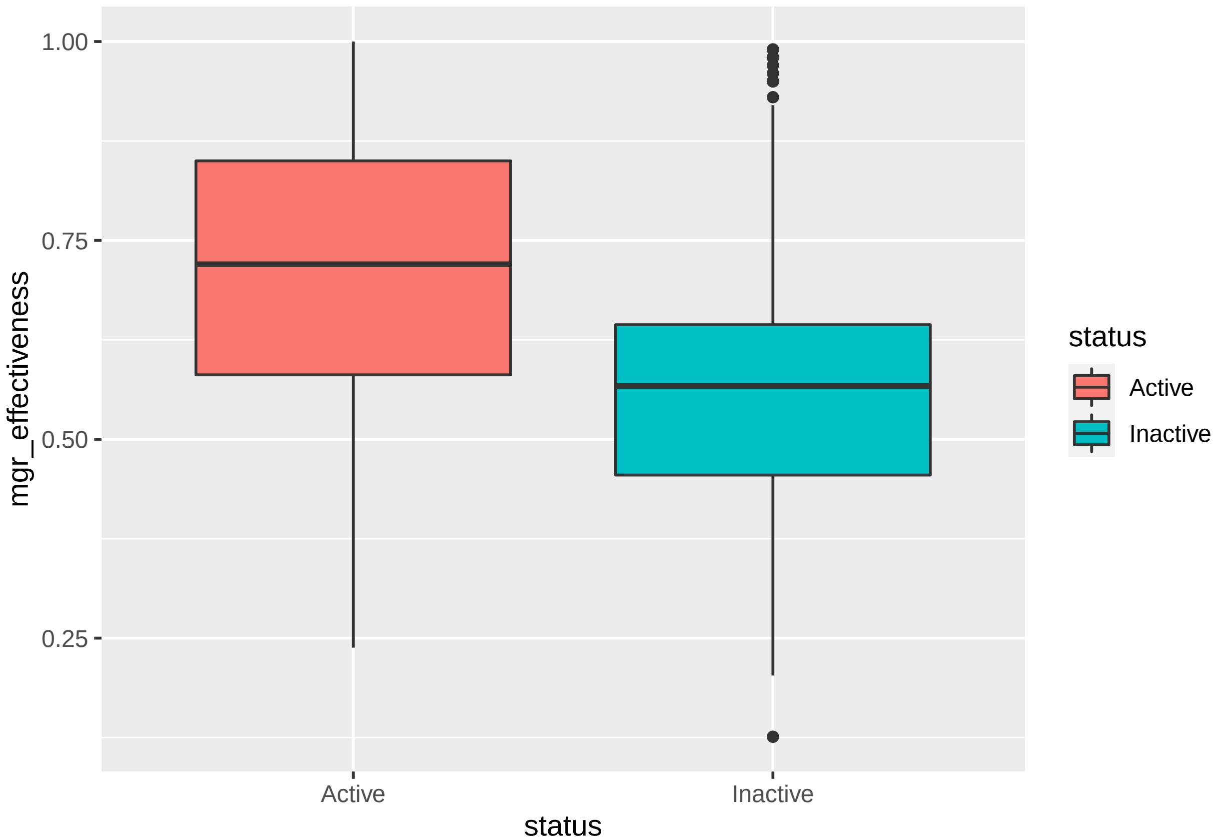
```
# Viewing the structure of survey dataset
glimpse(survey)
```

```
## Rows: 350
## Columns: 5
## $ mgr_id              <chr> "E1003", "E10072", "E10081", "E10234", "E1026", "E~
## $ mgr_effectiveness   <dbl> 0.760, 0.650, 0.800, 0.650, 0.700, 0.980, 0.520, 0~
## $ career_satisfaction <dbl> 0.76, 0.67, 0.82, 0.63, 1.00, 0.91, 0.56, 0.91, 0.~
## $ perf_satisfaction   <dbl> 0.71, 0.56, 0.73, 0.75, 1.00, 0.91, 0.50, 0.88, 0.~
## $ work_satisfaction   <dbl> 0.82, 0.84, 0.84, 0.70, 0.92, 0.77, 0.81, 0.84, 0.~
```

```
# merging datasets with a left join
org_final <- left_join(org3, survey, by = 'mgr_id')
org_final
```

```
## # A tibble: 1,954 x 19
##    emp_id status   turnover location level      date_of_joining date_of_birth
##    <chr>  <chr>       <dbl> <chr>    <chr>      <chr>           <chr>
##  1 E11061 Inactive        1 New York Analyst    22/03/2012      22/03/1992
##  2 E1031  Inactive        1 New York Analyst    09/03/2012      10/01/1992
##  3 E6213  Inactive        1 New York Analyst    06/01/2012      06/02/1992
##  4 E5900  Inactive        1 New York Analyst    22/03/2012      19/12/1991
##  5 E3044  Inactive        1 Florida  Analyst    29/03/2012      10/12/1991
##  6 E6636  Active          0 New York Specialist 17/02/2012      23/01/1992
##  7 E13796 Inactive        1 New York Analyst    30/03/2012      19/12/1990
##  8 E13549 Active          0 New York Analyst    09/03/2012      22/12/1991
##  9 E13430 Inactive        1 New York Analyst    09/03/2012      19/08/1991
## 10 E13349 Active          0 New York Analyst    09/03/2012      23/11/1991
## # ... with 1,944 more rows, and 12 more variables: last_working_date <chr>,
## #   gender <chr>, department <chr>, mgr_id <chr>, cutoff_date <chr>,
## #   generation <chr>, emp_age <dbl>, rating <chr>, mgr_effectiveness <dbl>,
## #   career_satisfaction <dbl>, perf_satisfaction <dbl>, work_satisfaction <dbl>
```

```
# Comparing manager effectiveness scores
ggplot(org_final, aes(x = status, y = mgr_effectiveness, fill = status)) +
  geom_boxplot()
```



After combining employee survey data our previuos data with the org, we see tht manager effectiveness is higher with the active employees who have stayed, indicating that a managers effectiveness may be tied in with employee retention. Where as with inactive manger effective scores are lower, meaning possible employees left due to lack of faith with their manager.

Master data overview

```
org_final <- read_csv("/Users/kevinlorenzoayala/Downloads/employee_data/org_final.csv")
```

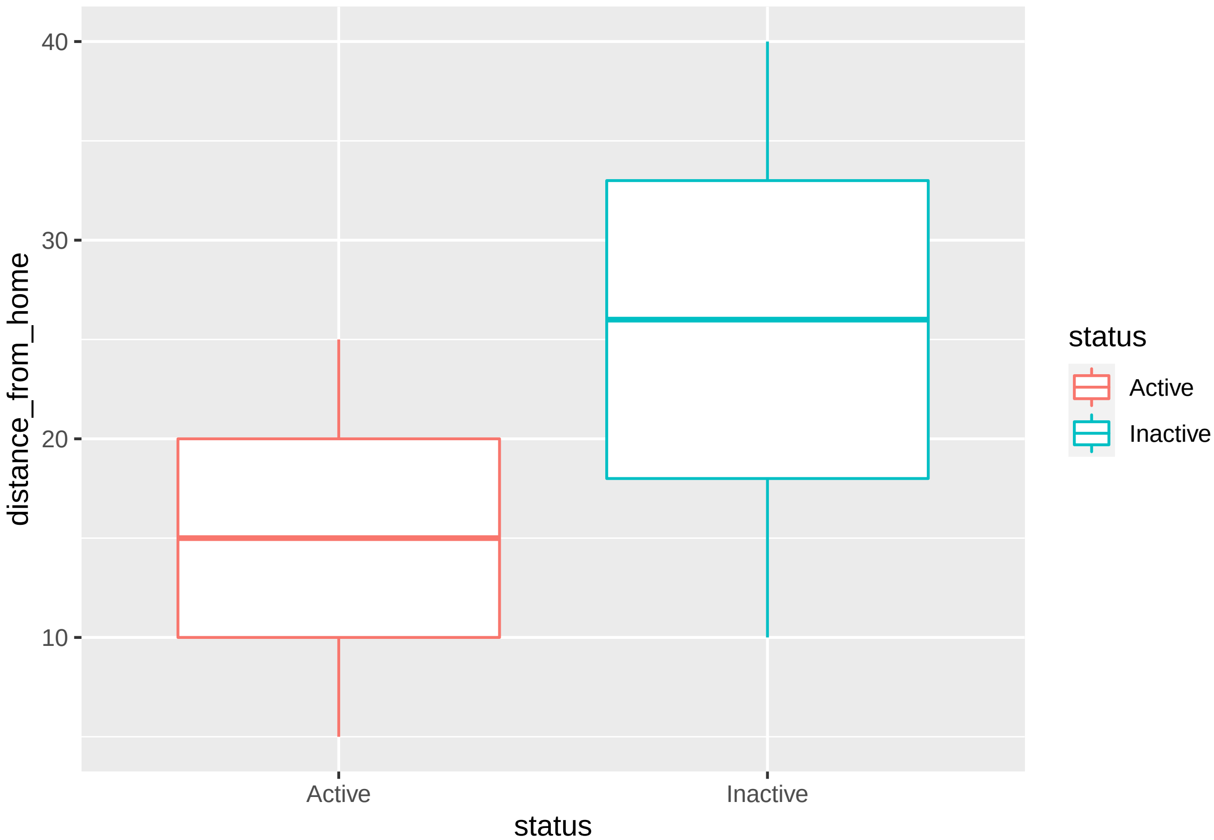
```
## Rows: 1954 Columns: 34
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (16): emp_id, status, location, level, gender, rating, mgr_rating, hirin...
## dbl (18): emp_age, mgr_reportees, mgr_age, mgr_tenure, compensation, percent...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Viewing the structure of the dataset
glimpse(org_final)
```

```
## Rows: 1,954
## Columns: 34
## $ emp_id                   <chr> "E10012", "E10025", "E10027", "E10048", "~
## $ status                   <chr> "Active", "Active", "Active", "Active", "~
## $ location                 <chr> "New York", "Chicago", "Orlando", "Chicag~
## $ level                    <chr> "Analyst", "Analyst", "Specialist", "Spec~
## $ gender                   <chr> "Female", "Female", "Female", "Male", "Ma~
## $ emp_age                  <dbl> 25.09, 25.98, 33.40, 24.55, 31.23, 31.98,~
```

```
## $ rating                       <chr> "Above Average", "Acceptable", "Acceptabl~
## $ mgr_rating                   <chr> "Acceptable", "Excellent", "Above Average~
## $ mgr_reportees                <dbl> 9, 4, 6, 10, 11, 19, 21, 9, 12, 22, 17, 1~
## $ mgr_age                      <dbl> 44.07, 35.99, 35.78, 26.70, 34.28, 34.82,~
## $ mgr_tenure                   <dbl> 3.17, 7.92, 4.38, 2.87, 12.95, 10.88, 4.0~
## $ compensation                 <dbl> 64320, 48204, 85812, 49536, 75576, 56904,~
## $ percent_hike                 <dbl> 10, 8, 11, 8, 12, 8, 12, 9, 9, 6, 11, 7, ~
## $ hiring_score                 <dbl> 70, 70, 77, 71, 70, 75, 72, 70, 70, 70, 7~
## $ hiring_source                <chr> "Consultant", "Job Fairs", "Consultant", ~
## $ no_previous_companies_worked <dbl> 0, 9, 3, 5, 0, 8, 9, 6, 1, 3, 3, 6, 2, 6,~
## $ distance_from_home           <dbl> 14, 21, 15, 9, 25, 23, 17, 16, 22, 22, 18~
## $ total_dependents             <dbl> 2, 2, 5, 3, 4, 5, 2, 5, 2, 5, 5, 5, 4, 5,~
## $ marital_status               <chr> "Single", "Single", "Single", "Single", "~
## $ education                    <chr> "Bachelors", "Bachelors", "Bachelors", "B~
## $ promotion_last_2_years       <chr> "No", "No", "Yes", "Yes", "No", "No", "No~
## $ no_leaves_taken              <dbl> 2, 10, 18, 19, 25, 15, 10, 20, 22, 23, 24~
## $ total_experience             <dbl> 6.86, 4.88, 8.55, 4.76, 8.06, 13.72, 5.81~
## $ monthly_overtime_hrs         <dbl> 1, 5, 3, 8, 1, 7, 2, 10, 2, 10, 8, 3, 1, ~
## $ date_of_joining              <chr> "06/03/2011", "23/09/2009", "02/11/2005",~
## $ last_working_date            <chr> NA, NA, NA, NA, NA, "11/12/2014", NA, NA,~
## $ department                   <chr> "Customer Operations", "Customer Operatio~
## $ mgr_id                       <chr> "E9335", "E6655", "E13942", "E7063", "E56~
## $ cutoff_date                  <chr> "31/12/2014", "31/12/2014", "31/12/2014",~
## $ turnover                     <dbl> 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 1, 1,~
## $ mgr_effectiveness            <dbl> 0.730, 0.581, 0.770, 0.240, 0.710, 0.574,~
## $ career_satisfaction          <dbl> 0.73, 0.72, 0.85, 0.42, 0.78, 0.88, 0.68,~
## $ perf_satisfaction            <dbl> 0.73, 0.84, 0.80, 0.33, 0.67, 0.81, 0.57,~
## $ work_satisfaction            <dbl> 0.75, 0.85, 0.87, 0.85, 0.80, 0.86, 0.75,~
```

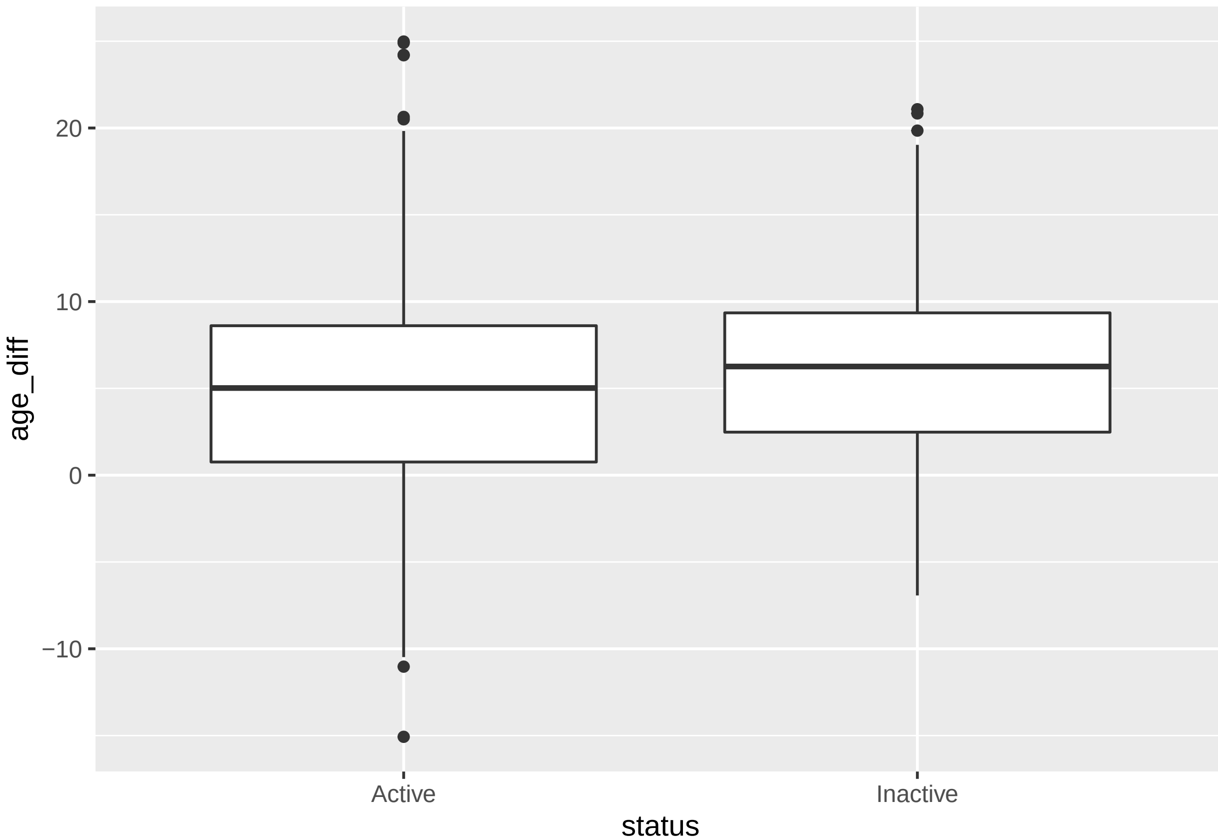
```
# Comparing the travel distance of Active and Inactive employees
ggplot(org_final, aes(x = status, y = distance_from_home, color = status)) +
  geom_boxplot()
```



Employees who live closer to where they work are less likely to turn to inactive employees. It would be interesting to see further data with the effect of remote work being implemented. There are a total of 34 variables.

Segment 2 Deriving Age Difference

```
# Adding in age_diff
emp_age_diff <- org_final %>%
  mutate(age_diff = mgr_age - emp_age)

# Ploting the distribution of age difference
ggplot(emp_age_diff, aes(x = status, y = age_diff)) +
  geom_boxplot()
```

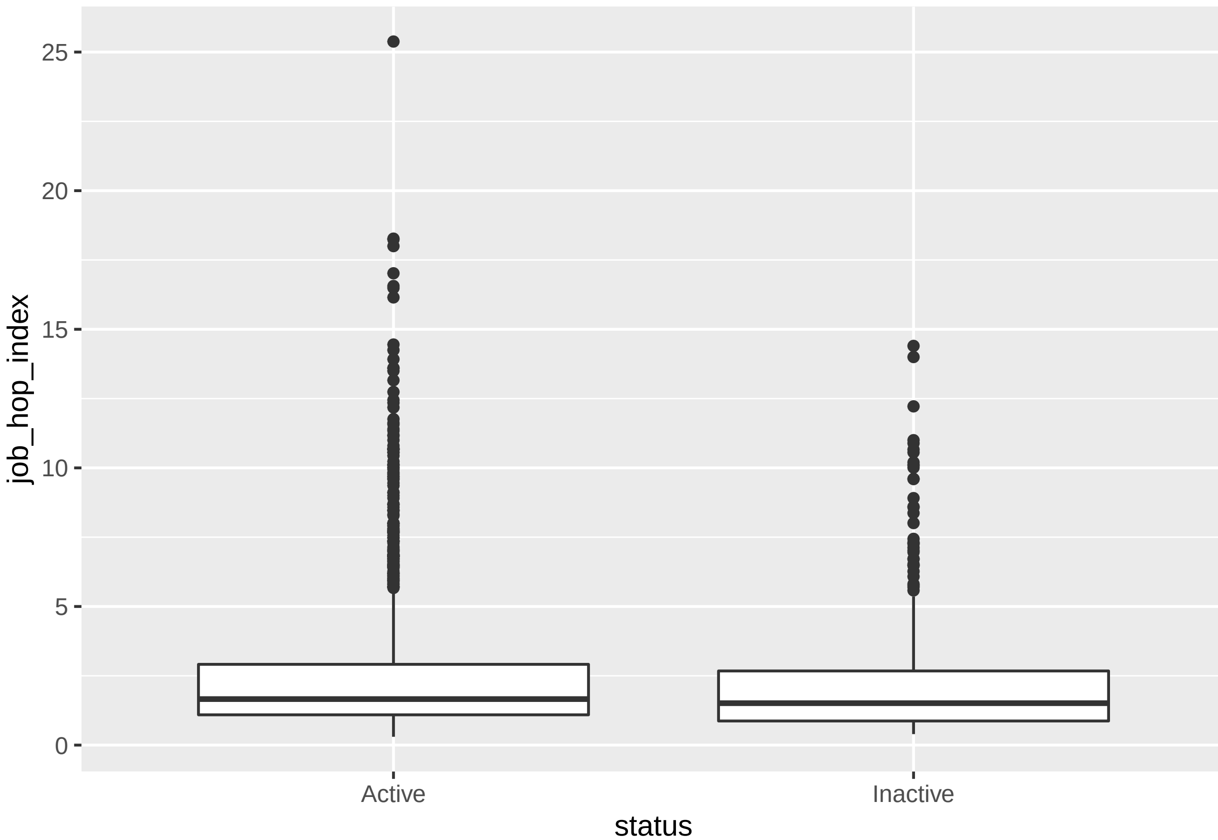


Employees who are closer to thier age with thier managers are likely to have more in common and thus have a happier time at work as opposed to workers who do not have anything in common with thier managers.

Deriving Job Hop Index

```
# Adding job_hop_index
emp_jhi <- emp_age_diff %>%
  mutate(job_hop_index = total_experience / no_previous_companies_worked)

# Comparing job hopping index of Active and Inactive employees
ggplot(emp_jhi, aes(x = status, y = job_hop_index)) +
  geom_boxplot()
```

```
## Warning: Removed 186 rows containing non-finite values (stat_boxplot).
```



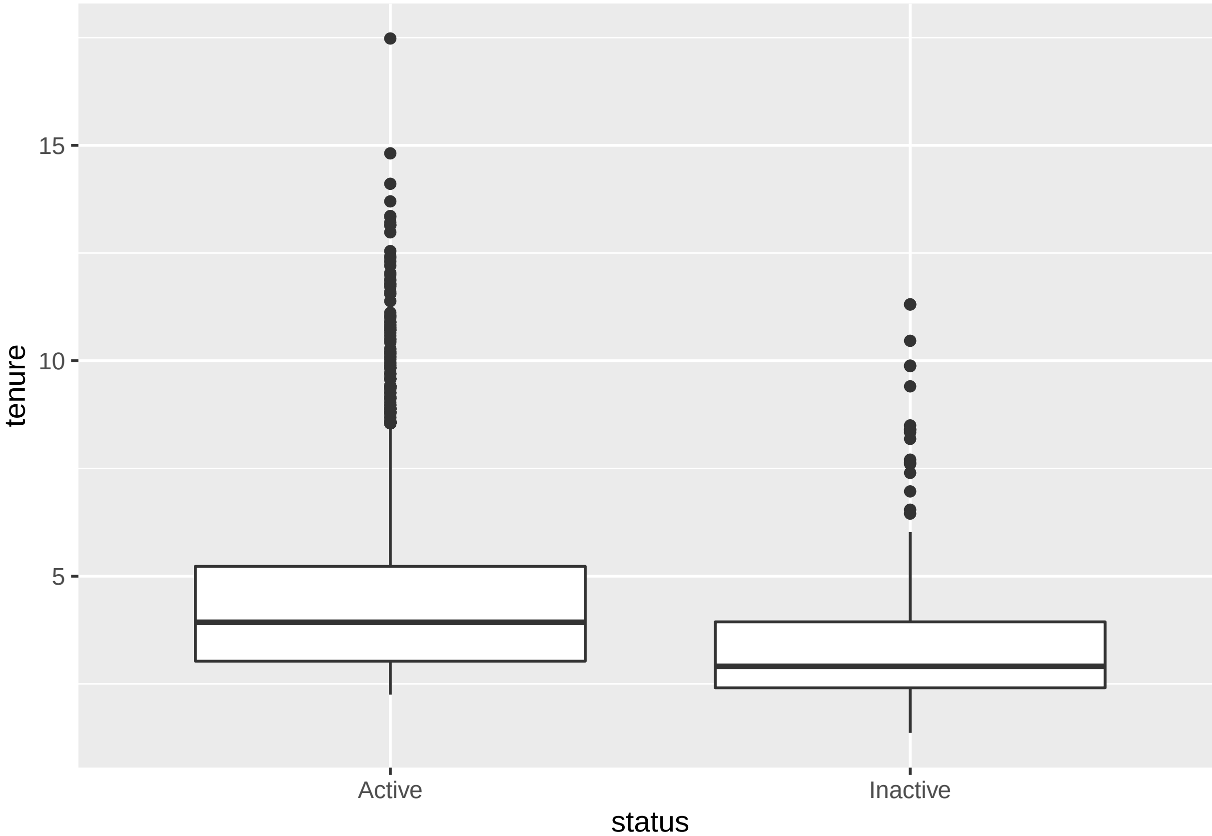
Median job hop index for active and inactive employee are similar.

Deriving Employee Tenure

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

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

```
#Converting data type from character to date with dmy format
emp_jhi <- org_final %>%
  mutate(date_of_joining= dmy(date_of_joining),
         cutoff_date = dmy(cutoff_date),
         last_working_date = dmy(last_working_date))

# Adding in tenure
emp_tenure <- emp_jhi %>%
  mutate(tenure = ifelse(status == "Active",
                         time_length(interval(date_of_joining, cutoff_date),
                                     "years"),
                         time_length(interval(date_of_joining, last_working_date),
                                     "years")))

# Comparing tenure of active and inactive employees
ggplot(emp_tenure, aes(x = status, y = tenure)) +
  geom_boxplot()
```

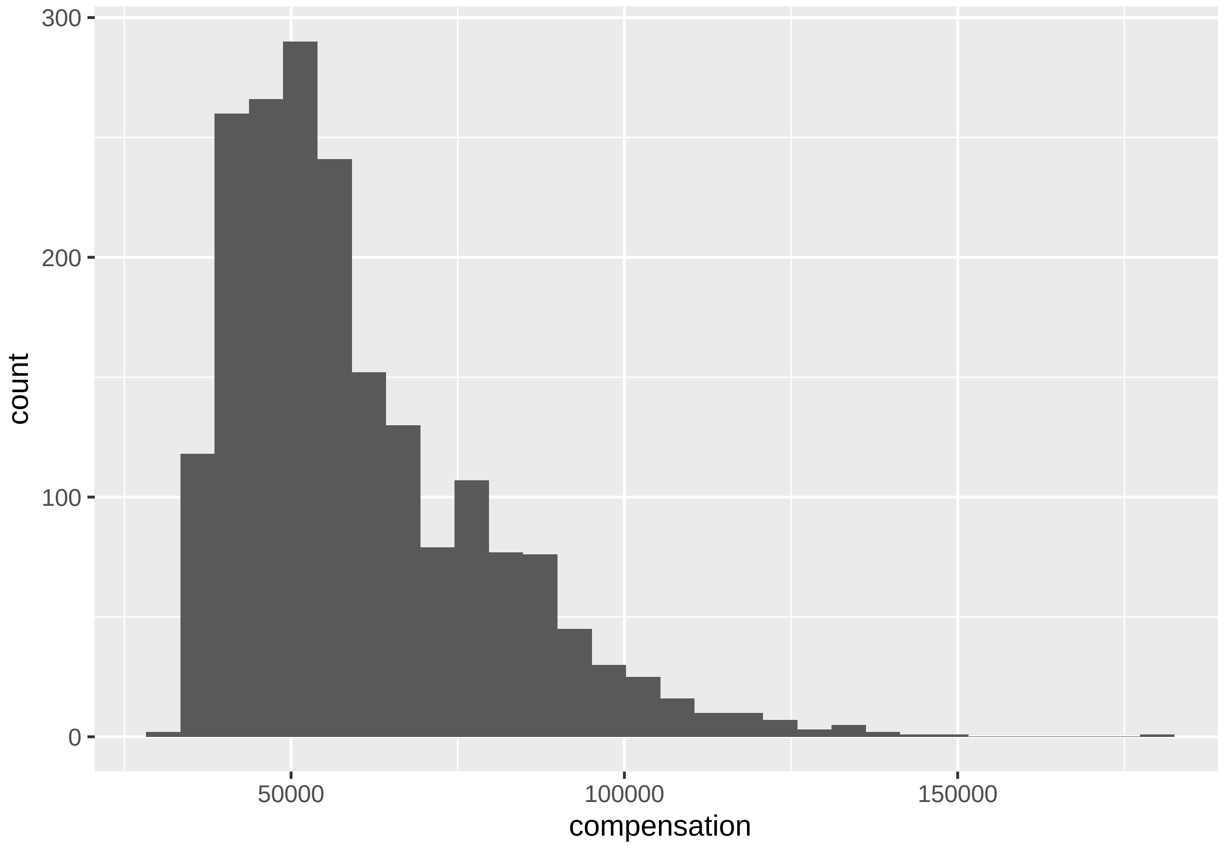


The median tenure of inactive employees is less than the tenure of active employees.

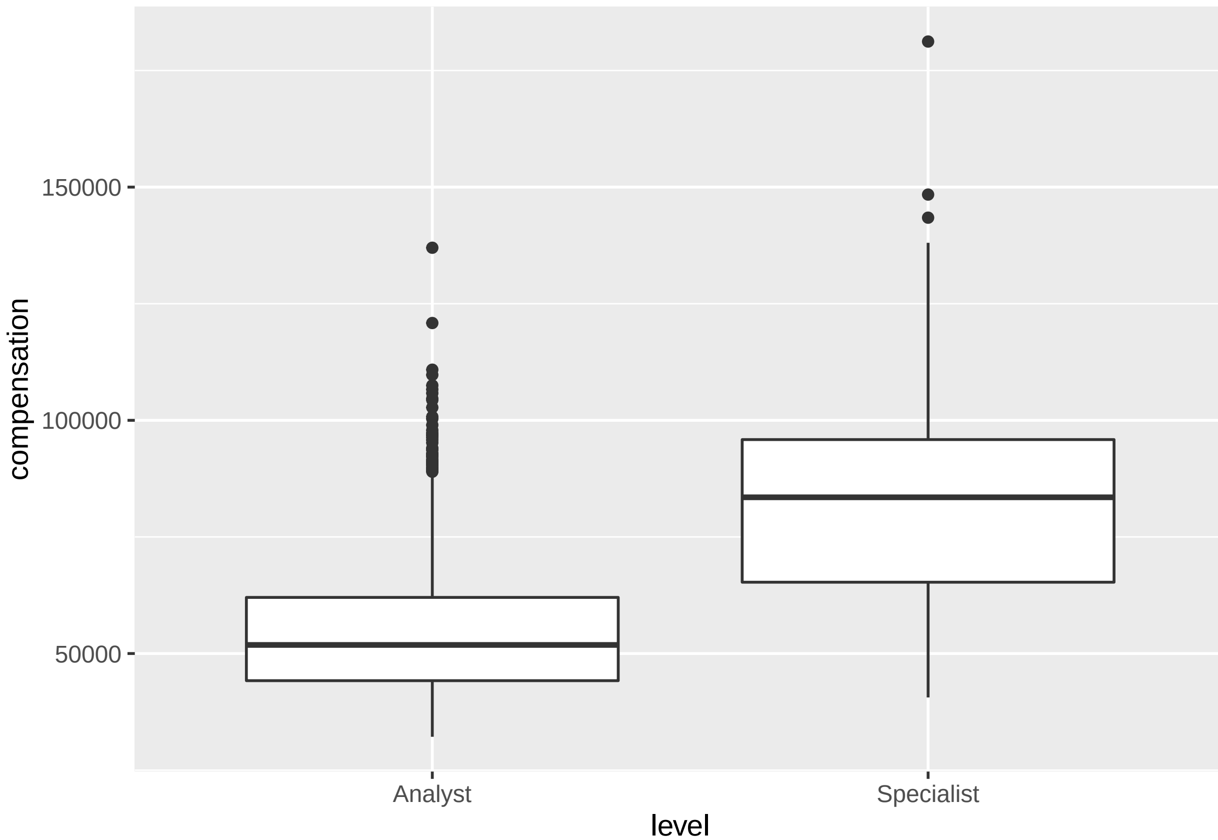
Exploring Compensation

```
# Ploting the distribution of compensation
ggplot(emp_tenure, aes(x = compensation)) +
  geom_histogram()
```

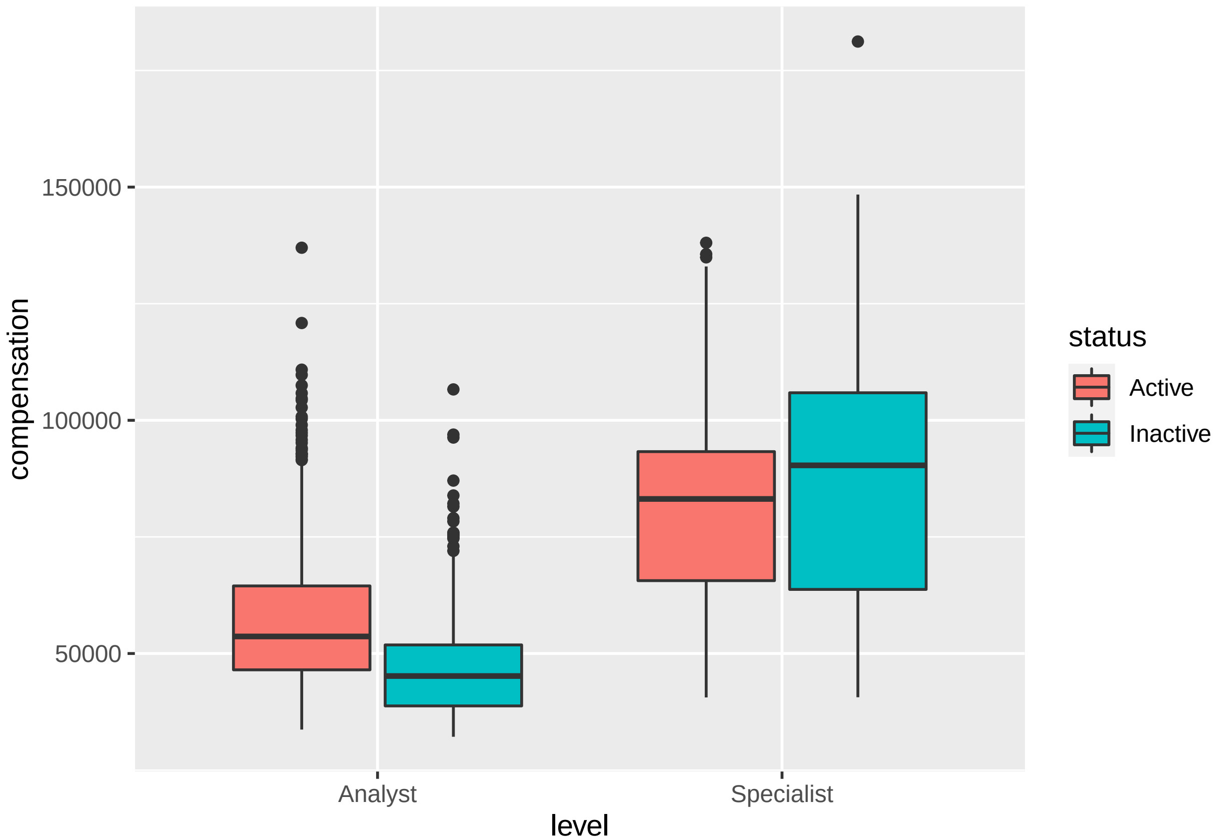
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
# Ploting the distribution of compensation across levels
ggplot(emp_tenure,
       aes(x = level, y = compensation)) +
  geom_boxplot()
```




```r
# Comparing compensation of Active and Inactive employees across levels
ggplot(emp_tenure,
       aes(x = level, y = compensation, fill = status)) +
  geom_boxplot()
```



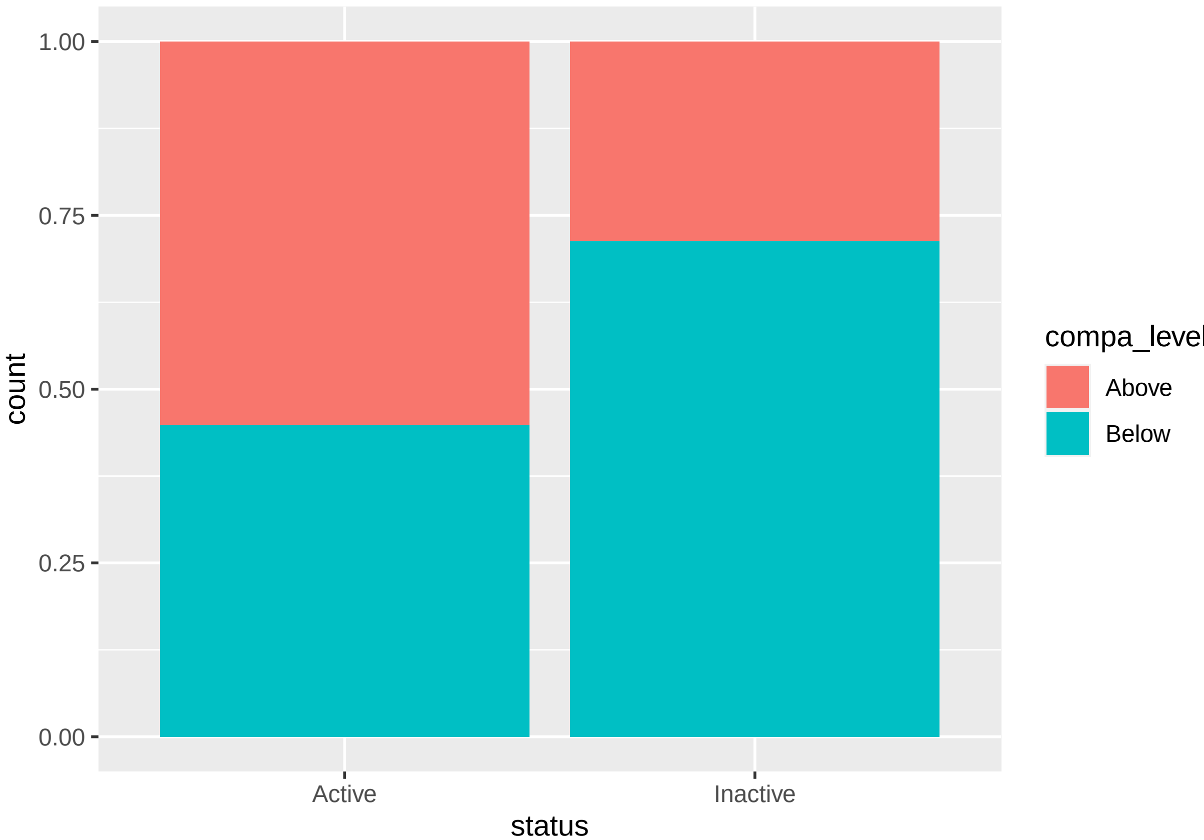
Variation exists within compensation for specialists and analysts.

Deriving Compa-ratio

```
# Adding median_compensation and compa_ratio
emp_compa_ratio <- emp_tenure %>%
  group_by(level) %>%
  mutate(median_compensation = median(compensation),
         compa_ratio = compensation / median_compensation)

# Looking at the median compensation for each level
emp_compa_ratio %>%
  distinct(level, median_compensation)
```

```
## # A tibble: 2 x 2
## # Groups:   level [2]
##   level      median_compensation
##   <chr>                    <dbl>
## 1 Analyst                  51840
## 2 Specialist               83496
```

```
# Adding compa_level
emp_final <- emp_compa_ratio  %>%
  mutate(compa_level = ifelse(compa_ratio > 1, "Above", "Below"))
# Comparing compa_level for Active and Inactive employees
ggplot(emp_final, aes(x = status, fill = compa_level)) +
  geom_bar(position = "fill")
```



Compa-ratio is a unique measure to calculate employee's pay competitiveness. A greater proportion of inactive employees were paid less than median compensation

Calculating Information Value

```
#Information package
library(Information)

# Computing Information Value
IV <- create_infotables(data = emp_final, y = "turnover")
```

```
## [1] "Variable emp_id was removed because it is a non-numeric variable with >1000 categories"
## [1] "Variable date_of_joining was removed because it is a Date variable"
## [1] "Variable last_working_date was removed because it is a Date variable"
## [1] "Variable department was removed because it has only 1 unique value"
## [1] "Variable cutoff_date was removed because it is a Date variable"
```

```
# Printing Information Value
IV$Summary
```

```
##                    Variable           IV
## 12              percent_hike 1.144784e+00
## 17          total_dependents 1.088645e+00
## 21           no_leaves_taken 9.404533e-01
## 29                    tenure 9.332570e-01
## 25         mgr_effectiveness 6.830020e-01
## 11              compensation 6.074885e-01
## 31               compa_ratio 4.768892e-01
## 6                     rating 3.869373e-01
## 23       monthly_overtime_hrs 3.786644e-01
```

```
## 8                 mgr_reportees 3.620543e-01
## 2                      location 2.963023e-01
## 32                  compa_level 2.940446e-01
## 24                       mgr_id 2.820235e-01
## 5                       emp_age 2.275477e-01
## 16           distance_from_home 1.470549e-01
## 28            work_satisfaction 1.378953e-01
## 22             total_experience 1.345781e-01
## 19                    education 1.253865e-01
## 20       promotion_last_2_years 9.979915e-02
## 9                       mgr_age 9.816205e-02
## 27            perf_satisfaction 7.099511e-02
## 13                 hiring_score 6.684727e-02
## 10                   mgr_tenure 5.918048e-02
## 26          career_satisfaction 3.539857e-02
## 3                         level 2.726491e-02
## 30          median_compensation 2.726491e-02
## 18               marital_status 2.588063e-02
## 7                    mgr_rating 2.172222e-02
## 15 no_previous_companies_worked 1.729893e-02
## 14                hiring_source 8.773529e-03
## 4                        gender 3.959968e-05
## 1                        status 0.000000e+00
```

```
# Loading caret
library('caret')

# Set seed of 567
set.seed(567)

# Storing row numbers for training dataset: index_train
index_train <- createDataPartition(emp_final$turnover, p = 0.7, list = FALSE)

# Creating training dataset: train_set
train_set <- emp_final[index_train, ]

# Creating testing dataset: test_set
test_set <- emp_final[-index_train, ]
```

Splitting data into test and training set.

```
# Calculating turnover proportion in train_set
train_set %>%
  count(status) %>%
  mutate(prop = n / sum(n))
```

```
## # A tibble: 4 x 4
## # Groups:   level [2]
##   level      status       n  prop
##   <chr>      <chr>    <int> <dbl>
## 1 Analyst    Active     882 0.792
## 2 Analyst    Inactive   232 0.208
## 3 Specialist Active     212 0.835
## 4 Specialist Inactive    42 0.165
```

```r
# Calculating turnover proportion in test_set
test_set %>%
  count(status) %>%
  mutate(prop = n / sum(n))
```

```
## # A tibble: 4 x 4
## # Groups:   level [2]
##   level      status       n  prop
##   <chr>      <chr>    <int> <dbl>
## 1 Analyst    Active     377 0.769
## 2 Analyst    Inactive   113 0.231
## 3 Specialist Active      86 0.896
## 4 Specialist Inactive    10 0.104
```

Viewing turnover propertion in both train set and test set. Logistic regression model

```r
#Dropping variables that are irrelevant or offer no predictive power.
train_set_multi <- train_set %>%
  select(-c(emp_id, mgr_id,
            date_of_joining, last_working_date, cutoff_date,
            mgr_age, emp_age,
            median_compensation,
            department, status))

#simple logistic regression model
simple_log <- glm(turnover ~ percent_hike,
                  family = "binomial", data = train_set_multi)

# Print summary
summary(simple_log)
```

```
##
## Call:
## glm(formula = turnover ~ percent_hike, family = "binomial", data = train_set_multi)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7907  -0.6943  -0.4600  -0.2989   2.6141
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.37851    0.21950    6.28 3.38e-10 ***
## percent_hike -0.29762    0.02396  -12.42  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1370.2  on 1367  degrees of freedom
## Residual deviance: 1176.7  on 1366  degrees of freedom
## AIC: 1180.7
##
## Number of Fisher Scoring iterations: 5
```

Multiple logistic regression model

```r
# Building a multiple logistic regression model
multi_log <- glm(turnover ~., family = "binomial",
                 data = train_set_multi)

#  summary
summary(multi_log)
```

```
##
## Call:
## glm(formula = turnover ~ ., family = "binomial", data = train_set_multi)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.31388  -0.15658  -0.04295  -0.00114   3.07960
##
## Coefficients:
##                                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)                   -9.829e+00  3.839e+00  -2.561 0.010451 *
## locationNew York               8.853e-01  4.565e-01   1.939 0.052496 .
## locationOrlando               -8.350e-01  3.895e-01  -2.144 0.032046 *
## levelSpecialist                1.431e+01  6.637e+02   0.022 0.982804
## genderMale                     2.181e-01  3.273e-01   0.666 0.505125
## ratingAcceptable              -1.279e-01  3.905e-01  -0.328 0.743206
## ratingBelow Average           -2.664e+00  7.091e-01  -3.757 0.000172 ***
## ratingExcellent               -4.294e-01  8.803e-01  -0.488 0.625675
## ratingUnacceptable            -4.805e+00  1.229e+00  -3.909 9.26e-05 ***
## mgr_ratingAcceptable          -7.965e-02  3.612e-01  -0.221 0.825460
## mgr_ratingBelow Average       -9.747e-01  6.713e-01  -1.452 0.146470
## mgr_ratingExcellent           -6.490e-01  5.121e-01  -1.267 0.205047
## mgr_ratingUnacceptable         1.001e+00  1.216e+00   0.824 0.410077
## mgr_reportees                  8.774e-02  2.981e-02   2.943 0.003252 **
## mgr_tenure                    -1.789e-02  4.431e-02  -0.404 0.686418
## compensation                   5.139e-05  4.492e-05   1.144 0.252578
## percent_hike                  -5.887e-01  8.154e-02  -7.220 5.22e-13 ***
## hiring_score                   7.771e-02  4.459e-02   1.743 0.081371 .
## hiring_sourceConsultant       -6.458e-01  5.399e-01  -1.196 0.231591
## hiring_sourceEmployee Referral -5.639e-01  6.149e-01  -0.917 0.359090
## hiring_sourceJob Boards       -8.354e-01  6.025e-01  -1.387 0.165584
## hiring_sourceJob Fairs        -6.596e-01  5.683e-01  -1.161 0.245826
## hiring_sourceSocial Media     -3.028e-01  5.690e-01  -0.532 0.594654
## hiring_sourceWalk-In          -4.387e-01  5.855e-01  -0.749 0.453647
## no_previous_companies_worked  -8.123e-03  5.380e-02  -0.151 0.879974
## distance_from_home             2.038e-01  2.287e-02   8.912  < 2e-16 ***
## total_dependents               7.689e-01  1.156e-01   6.654 2.84e-11 ***
## marital_statusSingle           2.505e+00  5.552e-01   4.512 6.43e-06 ***
## educationMasters               2.088e+00  5.717e-01   3.653 0.000259 ***
## promotion_last_2_yearsYes     -1.528e+01  6.637e+02  -0.023 0.981629
## no_leaves_taken                1.033e-01  2.013e-02   5.132 2.86e-07 ***
## total_experience              -4.571e-02  6.207e-02  -0.736 0.461477
## monthly_overtime_hrs           2.428e-01  4.302e-02   5.643 1.67e-08 ***
## mgr_effectiveness             -9.807e+00  1.499e+00  -6.540 6.15e-11 ***
## career_satisfaction            3.821e+00  1.463e+00   2.612 0.009013 **
## perf_satisfaction              1.957e+00  1.287e+00   1.521 0.128275
## work_satisfaction              1.267e+00  1.471e+00   0.861 0.389173
```

```
## tenure                          -3.399e-01  1.000e-01  -3.398 0.000680 ***
## compa_ratio                     -4.744e+00  3.152e+00  -1.505 0.132278
## compa_levelBelow                -3.377e-01  5.283e-01  -0.639 0.522690
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1370.21  on 1367  degrees of freedom
## Residual deviance:  345.81  on 1328  degrees of freedom
## AIC: 425.81
##
## Number of Fisher Scoring iterations: 17
```

Several variables are insignificant based on thier z value when compared to a P score. In multiple regression models, this can happen due to multicollinearity.

mgr_effectivenss and mgr_reportees are statistically significant while total experience and no of previous companies worked are not significant. No leaves taken and distance from home are statistically significant based on the data.

Detecting multicollinearity

```r
#car package
library(car)
```

```
## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode
```

```r
# Mult Logistic Model
multi_log <- glm(turnover ~ ., family = "binomial", data = train_set_multi)
# Checking for multicollinearity
vif(multi_log)
```

```
##                                   GVIF Df GVIF^(1/(2*Df))
## location                  2.061534e+00  2        1.198250
## level                     3.086518e+06  1     1756.848790
## gender                    1.208165e+00  1        1.099166
## rating                    4.097918e+00  4        1.192808
## mgr_rating                2.113320e+00  4        1.098046
## mgr_reportees             1.333047e+00  1        1.154577
## mgr_tenure                1.261319e+00  1        1.123085
## compensation              3.925142e+01  1        6.265095
## percent_hike              3.090133e+00  1        1.757877
## hiring_score              1.223568e+00  1        1.106150
## hiring_source             1.787944e+00  6        1.049614
## no_previous_companies_worked 1.128653e+00  1        1.062381
## distance_from_home        1.251847e+00  1        1.118860
## total_dependents          1.902276e+00  1        1.379230
## marital_status            2.185445e+00  1        1.478325
## education                 1.320618e+00  1        1.149182
## promotion_last_2_years    3.086503e+06  1     1756.844601
```

```
## no_leaves_taken             1.154258e+00  1        1.074364
## total_experience            1.981283e+00  1        1.407581
## monthly_overtime_hrs        1.343117e+00  1        1.158929
## mgr_effectiveness           3.184936e+00  1        1.784639
## career_satisfaction         3.080901e+00  1        1.755250
## perf_satisfaction           2.717291e+00  1        1.648421
## work_satisfaction           1.845829e+00  1        1.358613
## tenure                      1.571282e+00  1        1.253508
## compa_ratio                 2.966706e+01  1        5.446748
## compa_level                 3.315243e+00  1        1.820781
```

Based on the data, the variable Level will need to be removed due to high multicolinearity within the model. Adds noise our prediction.

Dealing with multicollinearity

```r
# Removing level
model_1 <- glm(turnover ~ . - level, family = "binomial",
               data = train_set_multi)

# Checking for multi collinearity again
vif(model_1)
```

```
##                                  GVIF Df GVIF^(1/(2*Df))
## location                     2.052868  2        1.196989
## gender                       1.200194  1        1.095533
## rating                       3.971558  4        1.188147
## mgr_rating                   2.116631  4        1.098261
## mgr_reportees                1.336350  1        1.156006
## mgr_tenure                   1.259402  1        1.122231
## compensation                22.692191  1        4.763632
## percent_hike                 3.072166  1        1.752759
## hiring_score                 1.216653  1        1.103020
## hiring_source                1.778261  6        1.049139
## no_previous_companies_worked 1.132551  1        1.064214
## distance_from_home           1.256052  1        1.120737
## total_dependents             1.881978  1        1.371852
## marital_status               2.185658  1        1.478397
## education                    1.323201  1        1.150305
## promotion_last_2_years       9.208556  1        3.034560
## no_leaves_taken              1.155954  1        1.075153
## total_experience             1.993409  1        1.411881
## monthly_overtime_hrs         1.337486  1        1.156497
## mgr_effectiveness            3.183209  1        1.784155
## career_satisfaction          3.097896  1        1.760084
## perf_satisfaction            2.703996  1        1.644383
## work_satisfaction            1.841462  1        1.357005
## tenure                       1.516532  1        1.231475
## compa_ratio                 18.032058  1        4.246417
## compa_level                  3.284659  1        1.812363
```

```r
# Removing level & compensation in possible final model
model_2 <- glm(turnover ~ . - level - compensation, family = "binomial",
               data = train_set_multi)

# Checking multi colinearity again
```

```r
vif(model_2)
```

```
##                                 GVIF Df GVIF^(1/(2*Df))
## location                    2.047031  2        1.196138
## gender                      1.194295  1        1.092838
## rating                      3.961019  4        1.187752
## mgr_rating                  2.025522  4        1.092238
## mgr_reportees               1.328670  1        1.152679
## mgr_tenure                  1.251760  1        1.118821
## percent_hike                3.091315  1        1.758214
## hiring_score                1.207234  1        1.098742
## hiring_source               1.735915  6        1.047034
## no_previous_companies_worked 1.116551  1        1.056670
## distance_from_home          1.246196  1        1.116332
## total_dependents            1.941515  1        1.393382
## marital_status              2.164802  1        1.471326
## education                   1.320524  1        1.149141
## promotion_last_2_years      1.252638  1        1.119213
## no_leaves_taken             1.150075  1        1.072415
## total_experience            1.944477  1        1.394445
## monthly_overtime_hrs        1.336019  1        1.155863
## mgr_effectiveness           3.203894  1        1.789942
## career_satisfaction         3.108811  1        1.763182
## perf_satisfaction           2.700848  1        1.643426
## work_satisfaction           1.855076  1        1.362012
## tenure                      1.467938  1        1.211585
## compa_ratio                 3.212172  1        1.792253
## compa_level                 2.982313  1        1.726938
```

We again repeat the process to find if other varaibles are causing multicolinearity, we see compensation to be causing issues and thus remove the variable.

A second pass through confirms that all variables are appropriate due to thier coefficient score being between 1 and 5.

Building final logistic regression model

```r
#Final Data Set with Level and Compensation removed
train_set_final <- train_set_multi %>% select(c(-level,-compensation))
```

```
## Adding missing grouping variables: `level`
```

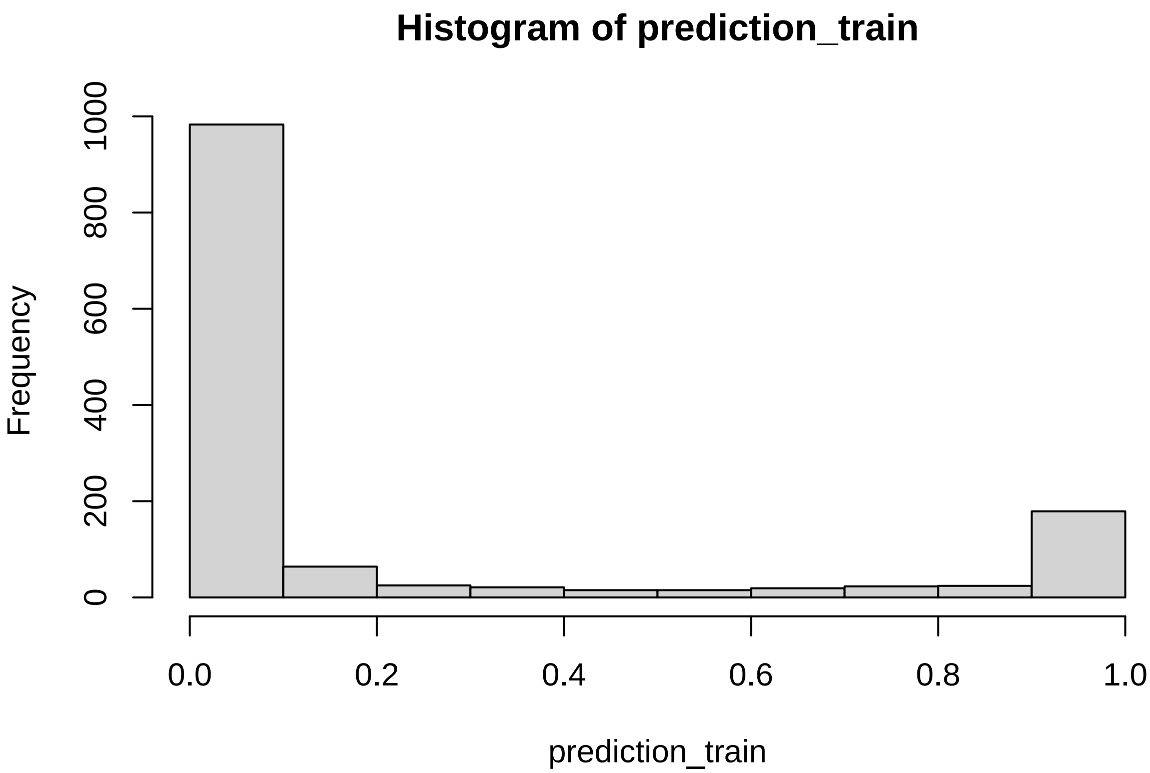
```r
# Building final logistic regression model
final_log <- glm(turnover ~ ., family = "binomial",
                 data = train_set_final)

# summary
summary(final_log)
```

```
##
## Call:
## glm(formula = turnover ~ ., family = "binomial", data = train_set_final)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.30953  -0.15853  -0.04369  -0.00120   3.08819
##
```

```
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)    
## (Intercept)                 -10.64835    3.73071  -2.854 0.004314 ** 
## levelSpecialist              15.96613  664.23321   0.024 0.980823    
## locationNew York              0.83179    0.45242   1.839 0.065984 .  
## locationOrlando              -0.87375    0.38647  -2.261 0.023768 *  
## genderMale                    0.20343    0.32663   0.623 0.533398    
## ratingAcceptable             -0.04198    0.38279  -0.110 0.912681    
## ratingBelow Average          -2.56467    0.70012  -3.663 0.000249 ***
## ratingExcellent              -0.45246    0.87301  -0.518 0.604270    
## ratingUnacceptable           -4.72055    1.22198  -3.863 0.000112 ***
## mgr_ratingAcceptable         -0.05626    0.36079  -0.156 0.876072    
## mgr_ratingBelow Average      -0.95242    0.66902  -1.424 0.154563    
## mgr_ratingExcellent          -0.59357    0.51107  -1.161 0.245468    
## mgr_ratingUnacceptable        0.97976    1.22902   0.797 0.425338    
## mgr_reportees                 0.08947    0.02982   3.001 0.002692 ** 
## mgr_tenure                   -0.01429    0.04399  -0.325 0.745301    
## percent_hike                 -0.58278    0.08092  -7.202 5.95e-13 ***
## hiring_score                  0.07349    0.04383   1.676 0.093643 .  
## hiring_sourceConsultant      -0.62220    0.53446  -1.164 0.244360    
## hiring_sourceEmployee Referral -0.54500  0.61268  -0.890 0.373713    
## hiring_sourceJob Boards      -0.85841    0.60037  -1.430 0.152777    
## hiring_sourceJob Fairs       -0.66069    0.56749  -1.164 0.244329    
## hiring_sourceSocial Media    -0.30504    0.56753  -0.537 0.590930    
## hiring_sourceWalk-In         -0.46192    0.58127  -0.795 0.426806    
## no_previous_companies_worked -0.01120    0.05358  -0.209 0.834440    
## distance_from_home            0.20548    0.02277   9.025  < 2e-16 ***
## total_dependents              0.77870    0.11594   6.716 1.86e-11 ***
## marital_statusSingle          2.51786    0.55251   4.557 5.19e-06 ***
## educationMasters              2.04174    0.56511   3.613 0.000303 ***
## promotion_last_2_yearsYes   -15.28726  664.23313  -0.023 0.981638    
## no_leaves_taken               0.10292    0.02000   5.146 2.65e-07 ***
## total_experience             -0.03841    0.06152  -0.624 0.532397    
## monthly_overtime_hrs          0.24731    0.04269   5.793 6.93e-09 ***
## mgr_effectiveness            -9.86578    1.49149  -6.615 3.72e-11 ***
## career_satisfaction           3.80528    1.45754   2.611 0.009034 ** 
## perf_satisfaction             2.09932    1.27772   1.643 0.100378    
## work_satisfaction             1.36072    1.47043   0.925 0.354765    
## tenure                       -0.36595    0.09883  -3.703 0.000213 ***
## compa_ratio                  -1.35328    0.99409  -1.361 0.173412    
## compa_levelBelow             -0.21257    0.50431  -0.422 0.673382    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1370.21  on 1367  degrees of freedom
## Residual deviance:  347.16  on 1329  degrees of freedom
## AIC: 425.16
##
## Number of Fisher Scoring iterations: 17
```

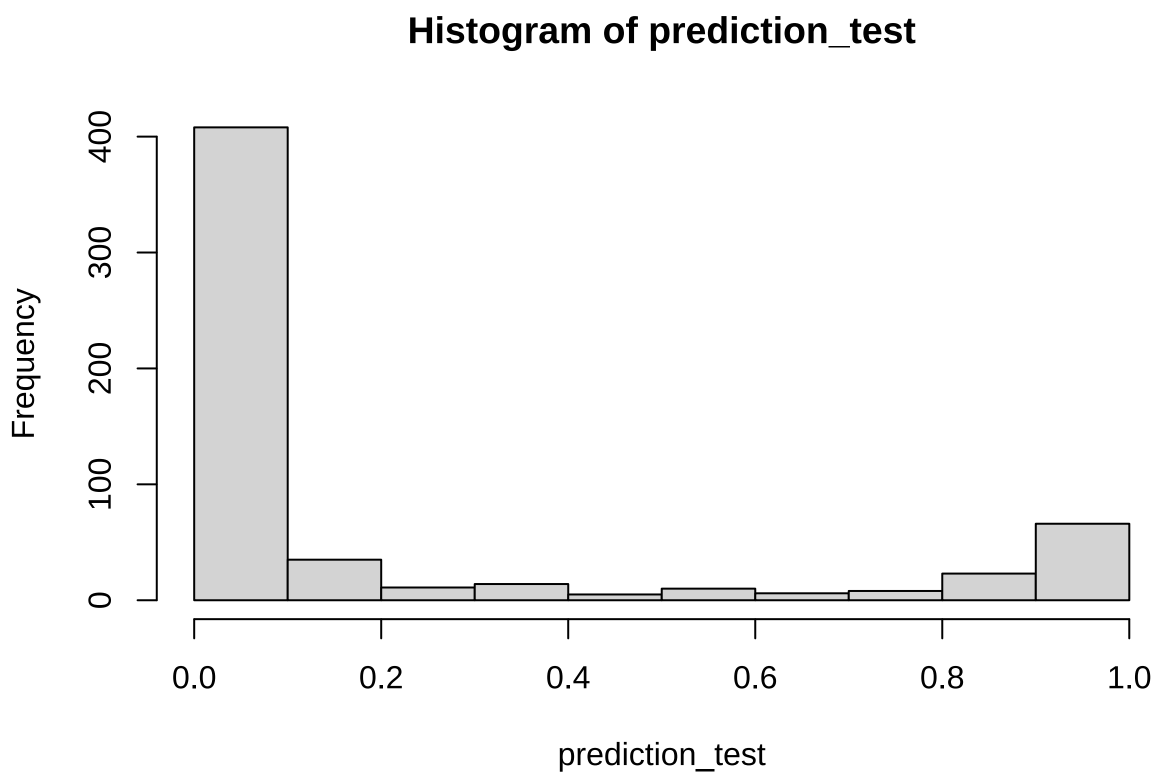
```
#Understanding the model predictions
# Make predictions for training dataset
```

```r
prediction_train <- predict(final_log, newdata = train_set,
                            type = "response")
#prediction range
hist(prediction_train)
```



Using final model to make predictions

```r
#predictions for testing dataset
prediction_test <- predict(final_log, newdata = test_set,
                        type = "response")

# Looking at the prediction range
hist(prediction_test)
```




```r
# Printing the probability of turnover
prediction_test[c(150, 200)]
```

```
##         150         200
## 0.007043613 0.258400055
```

probability range for training and test datasets are similar as confirmed visually by their histograms.

Creating a confusion matrix

```r
# Classifies predictions using a  standard cut-off of 0.5
prediction_categories <- ifelse(prediction_test > 0.5, 1, 0)

# Constructing a confusion matrix
conf_matrix <- table(prediction_categories, test_set$turnover)
conf_matrix
```

```
##
## prediction_categories   0   1
##                     0 447  26
##                     1  16  97
```

Constructing a confusion matrix for accuracy testing of the model.

Accuracy of model

```r
# Load caret
library(caret)

# Calls confusionMatrix
confusionMatrix(conf_matrix)
```

```
## Confusion Matrix and Statistics
```

```
##
##
## prediction_categories   0   1
##                     0 447  26
##                     1  16  97
##
##                Accuracy : 0.9283
##                  95% CI : (0.9044, 0.9479)
##     No Information Rate : 0.7901
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.7773
##
##  Mcnemar's Test P-Value : 0.1649
##
##             Sensitivity : 0.9654
##             Specificity : 0.7886
##          Pos Pred Value : 0.9450
##          Neg Pred Value : 0.8584
##              Prevalence : 0.7901
##          Detection Rate : 0.7628
##    Detection Prevalence : 0.8072
##       Balanced Accuracy : 0.8770
##
##        'Positive' Class : 0
##
```

After turning in the mdel into the accuracy, we see a satisfactory score well in the .9 or 90% accuracy which is good.

Segment 4 Calculating turnover risk probability

```r
# Loading tidypredict
library(tidypredict)

# Probability's of turnover
emp_risk <- emp_final %>%
  filter(status == "Active") %>%
  tidypredict_to_column(final_log)

# Running the code
emp_risk %>%
  select(emp_id, fit) %>%
  top_n(2)
```

```
## Adding missing grouping variables: `level`
## Selecting by fit
```

```
## # A tibble: 4 x 3
## # Groups:   level [2]
##   level      emp_id   fit
##   <chr>      <chr>  <dbl>
## 1 Analyst    E13342 0.931
## 2 Specialist E202   0.888
## 3 Analyst    E6037  0.941
## 4 Specialist E6475  0.890
```

Calculating employee turnover probability.

Creating turnover risk buckets

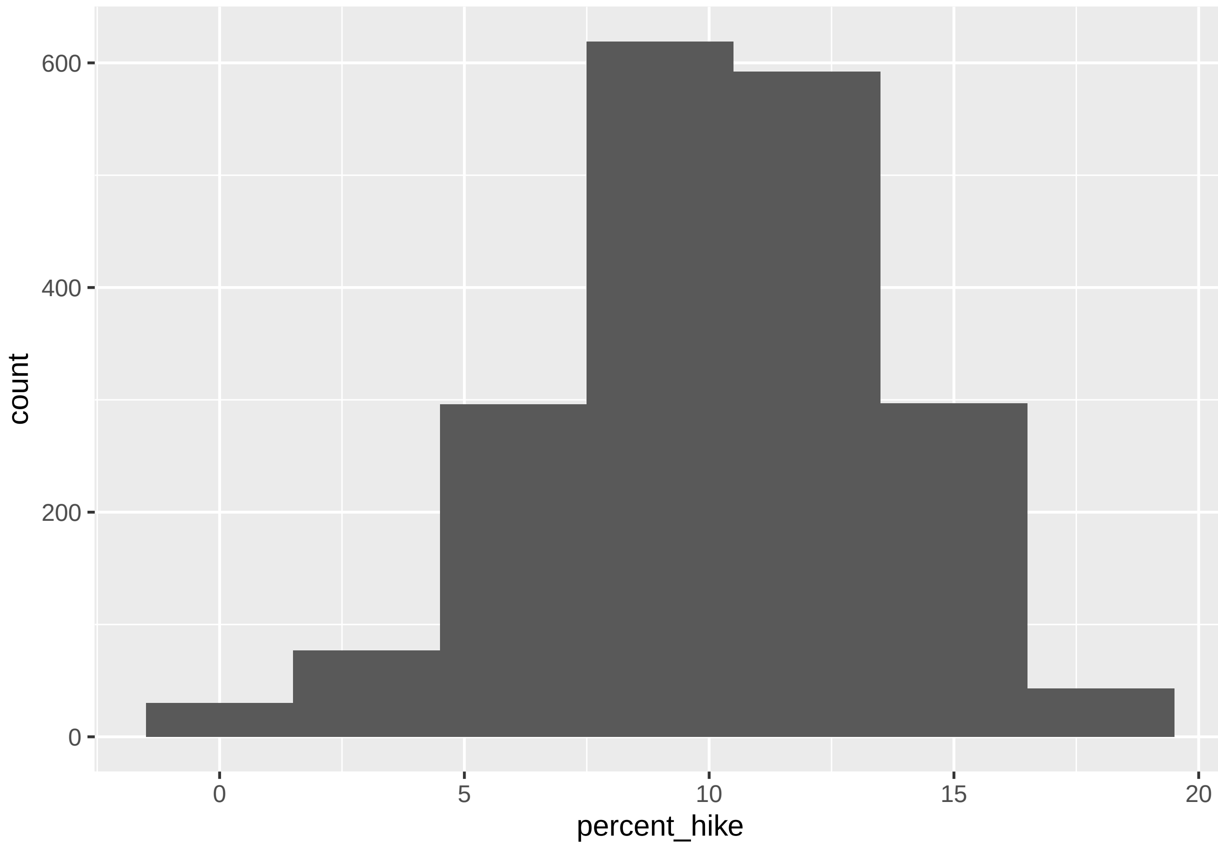
```
# Creating turnover risk buckets
emp_risk_bucket <- emp_risk %>%
  mutate(risk_bucket = cut(fit, breaks = c(0, 0.5, 0.6, 0.8, 1),
                           labels = c("no-risk", "low-risk",
                                      "medium-risk", "high-risk")))

# Counting employees in each risk bucket
emp_risk_bucket %>%
  count(risk_bucket)
```

```
## # A tibble: 7 x 3
## # Groups:   level [2]
##   level      risk_bucket     n
##   <chr>      <fct>       <int>
## 1 Analyst    no-risk      1225
## 2 Analyst    low-risk        9
## 3 Analyst    medium-risk    15
## 4 Analyst    high-risk      10
## 5 Specialist no-risk       293
## 6 Specialist medium-risk     2
## 7 Specialist high-risk       3
```

no-risk, if 0 <= fit <= 0.5 low-risk, if 0.5 < fit <= 0.6 medium-risk, if 0.6 < fit <= 0.8 high-risk, if 0.8 < fit <= 1

Percent hike effects

```
#histogram of percent hike
ggplot(emp_final, aes(x = percent_hike)) +
  geom_histogram(binwidth = 3)
```



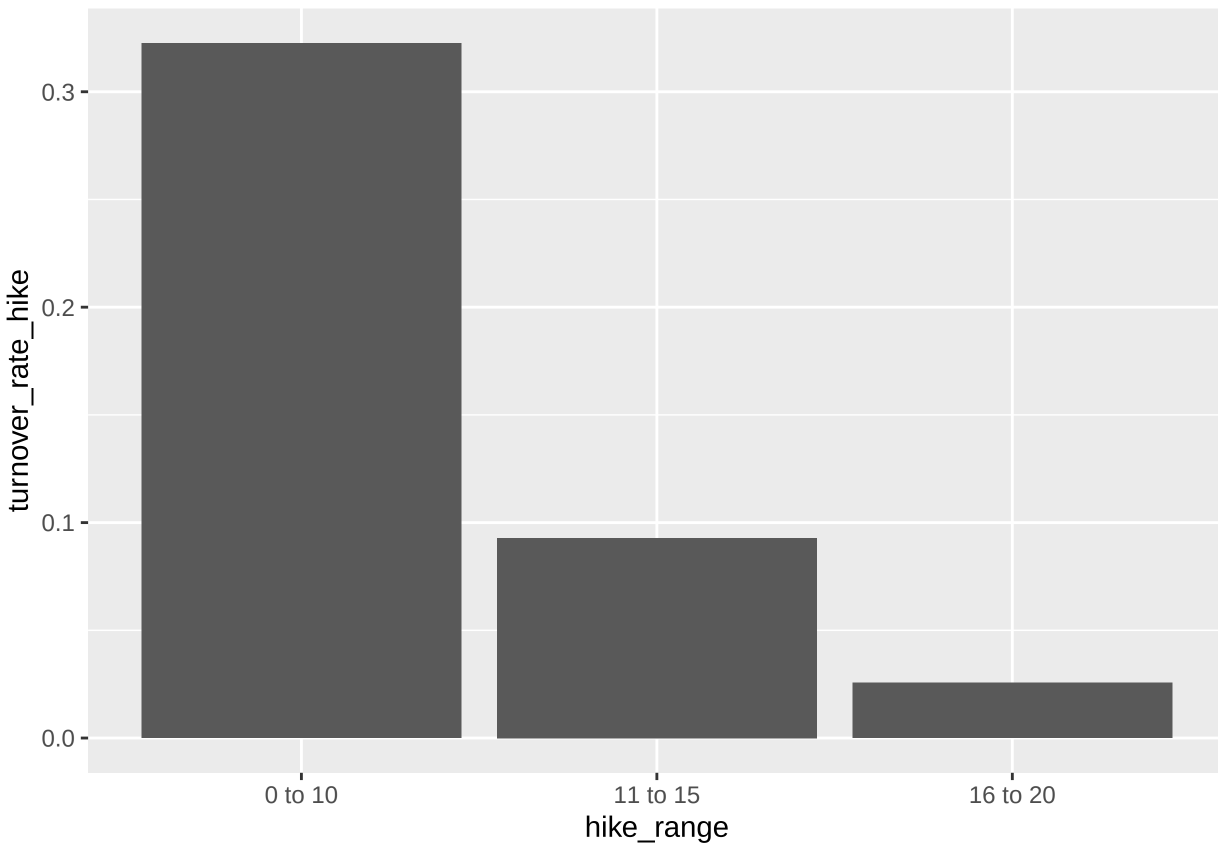

```
#salary hike_range of Analyst level employees
emp_hike_range <- emp_final %>%
  filter(level == "Analyst") %>%
  mutate(hike_range = cut(percent_hike, breaks = c(0, 10, 15, 20),
                          include.lowest = TRUE,
                          labels = c("0 to 10",
                                     "11 to 15", "16 to 20")))
```

0 to 10, if 0 <= percent_hike <= 10 11 to 15, if 11 <= percent_hike <= 15 16 to 20, if 16 <= percent_hike <= 20 Calculate turnover rate across salary hike range

```
# turnover rates for each salary hike range
df_hike <- emp_hike_range %>%
  group_by(hike_range) %>%
  summarize(turnover_rate_hike = mean(turnover))

# Checking the results
df_hike
```

```
## # A tibble: 3 x 2
##   hike_range turnover_rate_hike
##   <fct>                   <dbl>
## 1 0 to 10                0.323
## 2 11 to 15               0.0929
## 3 16 to 20               0.0256
```

```
# Visualizing the results with ggplot2
ggplot(df_hike, aes(x = hike_range, y = turnover_rate_hike)) +
  geom_col()
```



This graph helps us understand if there is a difference in the percentage of employees leaving the organization in different categories of salary hike