Salary prediction based on Stackoverflow 2022 survey

parshal Chitrakar

2023-05-07

## Problem Statement

The problem that we are trying to solve is to predict the salaries of employees based on numerous factors such as their experience, education level, job title, type of employment, and so on. This is important because it can help organizations to make informed decisions about compensation, recruitment, and retention strategies. By accurately predicting salaries, organizations can ensure that they are paying their employees fairly, attracting the right talent, and retaining their top performers.

## Software and Packages

1. Software: Rstudio
2. Packages:
   * tidyverse for data wrangling and visualization
   * caret for machine learning algorithms and model tuning
   * dplyr for simplifying the data
   * ggplot2 for visualization of data
   * tidymodels for modeling and statistical analysis of data
   * corrplot for the visual explanation of correlation
   * tidyr for cleaning the messy data

## Variables

1. Dependent variable: ConvertedCompYearly (Converted company salary in USD which comes with dataset)
2. Independent Variables
   * Development Type
   * Education level
   * Work experience
   * Country
   * Industry type
   * Company size
   * Age
   * Employment
   * Years Code
   * Years Code Pro

### Loading the Libraries

# Load required packages  
library(dplyr)  
library(tidyr)  
library(caret)  
library(ggplot2)  
library(corrplot)  
library(tidyverse)  
library(tidymodels)  
library(corrr)  
library(correlation)  
library(Metrics)  
library(stringr)  
library(stargazer)  
library(mltools)

### Setting up Directory and loading data

# First, turn off scientific notation for numbers  
options(scipen=999)  
getwd()

## [1] "C:/Users/parsh/OneDrive/Desktop"

setwd("C:/Users/parsh/OneDrive/Desktop/R\_prediction")  
initial\_data <- read.csv("survey\_results\_public.csv")  
df\_selected <- initial\_data %>% select(OrgSize,Employment,EdLevel,YearsCode, YearsCodePro,  
 DevType,Country, Currency,  
 Age, WorkExp,  
 ConvertedCompYearly)  
dim(df\_selected)

## [1] 73268 11

### Removing missing data

df\_selected <- df\_selected %>% drop\_na()  
#summarise the data  
summary(df\_selected)

## OrgSize Employment EdLevel YearsCode   
## Length:26903 Length:26903 Length:26903 Length:26903   
## Class :character Class :character Class :character Class :character   
## Mode :character Mode :character Mode :character Mode :character   
##   
##   
##   
## YearsCodePro DevType Country Currency   
## Length:26903 Length:26903 Length:26903 Length:26903   
## Class :character Class :character Class :character Class :character   
## Mode :character Mode :character Mode :character Mode :character   
##   
##   
##   
## Age WorkExp ConvertedCompYearly  
## Length:26903 Min. : 0.00 Min. : 1   
## Class :character 1st Qu.: 4.00 1st Qu.: 34126   
## Mode :character Median : 8.00 Median : 65591   
## Mean :10.43 Mean : 159300   
## 3rd Qu.:15.00 3rd Qu.: 117126   
## Max. :50.00 Max. :50000000

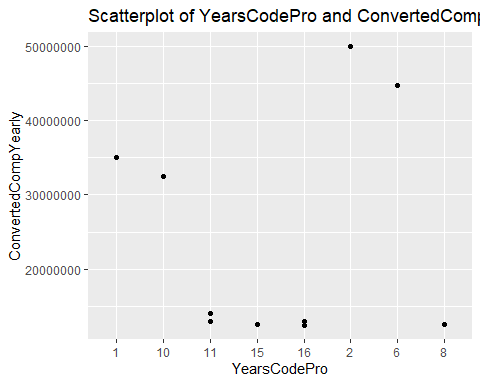
df\_selected %>% summarise\_all(list(~n\_distinct(.)))

## OrgSize Employment EdLevel YearsCode YearsCodePro DevType Country Currency  
## 1 10 10 9 52 52 5067 156 120  
## Age WorkExp ConvertedCompYearly  
## 1 8 51 6588

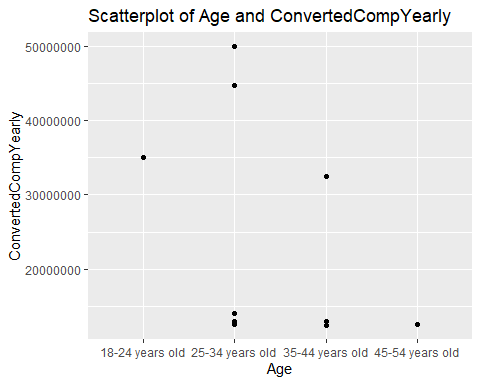
write.csv(df\_selected,"clean\_data.csv")

### Data Visualization based on highest top 10 convertedCompYearly

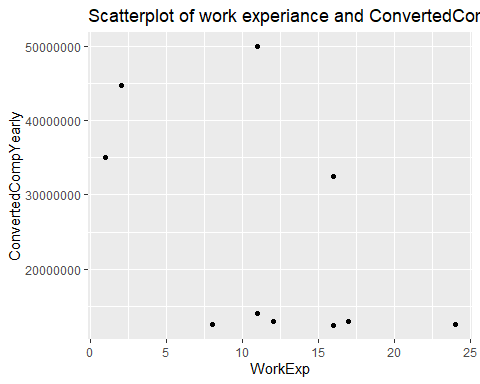
##### Scatter Plot between YearsCodePro VS ConvertedCompYearly



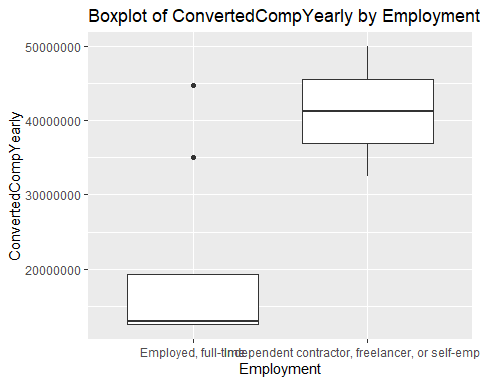
##### Scatter Plot of Age and ConvertedCompYearly

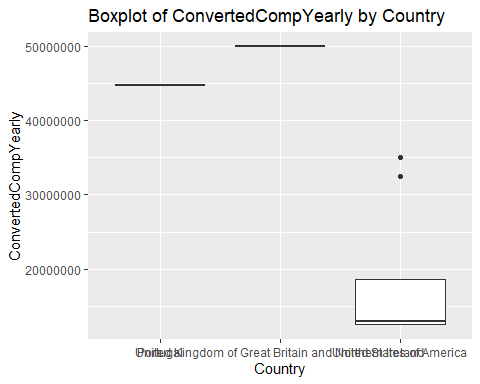


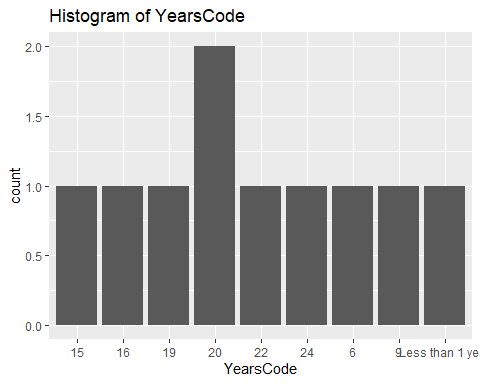
##### scatterplot of work experience and ConvertedCompYearly



##### Boxplot of employment and ConvertedCompYearly







## Data Processing

Generating the clean csv file which we extracted from the df\_selected data frame for further processing of data and predicting. Also, checking the summary of the data.

setwd("C:/Users/parsh/OneDrive/Desktop/R\_prediction")  
data <- read.csv("clean\_data.csv")  
  
df<- data %>%   
 select(OrgSize,Employment,EdLevel,YearsCode, YearsCodePro,DevType, Age, WorkExp, ConvertedCompYearly)  
  
  
df %>% summarise\_all(list(~n\_distinct(.)))

## OrgSize Employment EdLevel YearsCode YearsCodePro DevType Age WorkExp  
## 1 10 10 9 52 52 5067 8 51  
## ConvertedCompYearly  
## 1 6588

# Data Types  
df\_dtypes=sapply(df, class)  
df\_dtypes <- data.frame(df\_dtypes)  
df\_dtypes

## df\_dtypes  
## OrgSize character  
## Employment character  
## EdLevel character  
## YearsCode character  
## YearsCodePro character  
## DevType character  
## Age character  
## WorkExp integer  
## ConvertedCompYearly integer

# Cleaning CompTotal   
df\_cleaned<-df  
df\_cleaned <- df[!is.na(df$ConvertedCompYearly), ]  
df\_cleaned$CompTotal <- replace(df\_cleaned$ConvertedCompYearly,df\_cleaned$ConvertedCompYearly>7000000,4500000)  
  
# Mean of the dependent Variable  
mean(df\_cleaned$ConvertedCompYearly)

## [1] 159300.5

## Checking the levels and the factors of each variable

# Education level#  
levels(factor(df\_cleaned$EdLevel))

## [1] "Associate degree (A.A., A.S., etc.)"   
## [2] "Bachelor's degree (B.A., B.S., B.Eng., etc.)"   
## [3] "Master's degree (M.A., M.S., M.Eng., MBA, etc.)"   
## [4] "Other doctoral degree (Ph.D., Ed.D., etc.)"   
## [5] "Primary/elementary school"   
## [6] "Professional degree (JD, MD, etc.)"   
## [7] "Secondary school (e.g. American high school, German Realschule or Gymnasium, etc.)"  
## [8] "Some college/university study without earning a degree"   
## [9] "Something else"

#Employment#  
levels(factor(df\_cleaned$Employment))

## [1] "Employed, full-time"   
## [2] "Employed, full-time;Employed, part-time"   
## [3] "Employed, full-time;Independent contractor, freelancer, or self-employed"   
## [4] "Employed, full-time;Independent contractor, freelancer, or self-employed;Employed, part-time"  
## [5] "Employed, full-time;Independent contractor, freelancer, or self-employed;Retired"   
## [6] "Employed, full-time;Retired"   
## [7] "Employed, part-time"   
## [8] "Employed, part-time;Retired"   
## [9] "Independent contractor, freelancer, or self-employed"   
## [10] "Independent contractor, freelancer, or self-employed;Employed, part-time"

#Age#  
levels(factor(df\_cleaned$Age))

## [1] "18-24 years old" "25-34 years old" "35-44 years old"   
## [4] "45-54 years old" "55-64 years old" "65 years or older"   
## [7] "Prefer not to say" "Under 18 years old"

#OrgSize#  
levels(factor(df\_cleaned$OrgSize))

## [1] "1,000 to 4,999 employees"   
## [2] "10 to 19 employees"   
## [3] "10,000 or more employees"   
## [4] "100 to 499 employees"   
## [5] "2 to 9 employees"   
## [6] "20 to 99 employees"   
## [7] "5,000 to 9,999 employees"   
## [8] "500 to 999 employees"   
## [9] "I don't know"   
## [10] "Just me - I am a freelancer, sole proprietor, etc."

## Hot encoding function for encoding the catagorical varibles

hot\_encoding <- function(df,column){  
 df\_updated <- one\_hot(df[column],dropCols=TRUE,sparsifyNAs=TRUE)  
 return(df\_updated)  
}

## Encoding the catagrical variables

#### Below function was borrowed from this [website Link!!!](https://www.kaggle.com/code/klmsathishkumar/stack-overflow-survey-eda-salary-prediction) and necessay modification was done based on the requirements.

clean\_employment<- function(df){  
   
 df$Employment[is.na(df\_cleaned$Employment)] <- "Employed full-time"  
 df$Employment[df$Employment == "Independent contractor, freelancer, or self-employed"] <- "Freelancer"  
   
 df\_employement <- hot\_encoding(df,"Employment")  
 df\_removed = subset(df, select = -c(Employment))  
 df\_total <- cbind(df\_removed,df\_employement)  
   
 return(df\_total)  
}  
  
# Function to Clean EDLevel  
clean\_edlevel<- function(df){  
 df$EdLevel[is.na(df$EdLevel)] <- "Bachelor’s degree (B.A., B.S., B.Eng., etc.)"  
   
 df\_EdLevel <- hot\_encoding(df,"EdLevel")  
 df\_removed = subset(df, select = -c(EdLevel))  
   
 df\_total <- cbind(df\_removed,df\_EdLevel)  
 return(df\_total)  
}  
  
  
# Function to Clean YearsCode  
clean\_YearsCode <- function(df){  
 df$YearsCode[df$YearsCode == "Less than 1 year"] <- 0.5  
   
 df$YearsCode[df$YearsCode == "More than 50 years"] <- 55  
   
 df$YearsCode[is.na(df$YearsCode)] <- 4  
 df$YearsCode <- as.integer(as.numeric(as.character(df$YearsCode)))  
   
 return(df)  
}  
  
# Function to Clean YearsCode  
clean\_YearsCodepro <- function(df){  
 df$YearsCodePro[df$YearsCodePro == "Less than 1 year"] <- 0.5  
   
 df$YearsCodePro[df$YearsCodePro == "More than 50 years"] <- 55  
   
 df$YearsCodePro[is.na(df$YearsCodePro)] <- 4  
   
 df$YearsCodePro <- as.integer(as.numeric(as.character(df$YearsCodePro)))  
 return(df)  
}  
  
  
# Function to Clean Devtype  
clean\_devtype <- function(df){  
 df <- df %>%separate(DevType, c("DevType"),",")  
   
 df <- df %>%separate(DevType, c("DevType"),";")  
   
 df$DevType[is.na(df$DevType)] <- "Data scientist or machine learning specialist"  
   
 mean\_salary\_devtype = df %>% group\_by(DevType) %>% summarise(DevTypeEncoded = mean(CompTotal))  
   
 df\_final = left\_join(df, mean\_salary\_devtype)  
 df\_removed = subset(df\_final, select = -c(DevType))  
 return(df\_removed)  
}  
# Getting basic insights  
clean\_orgsize <- function(df){  
 df$OrgSize = factor(df$OrgSize,levels = c('1,000 to 4,999 employees',   
 '10 to 19 employees',   
 '10,000 or more employees',  
 '100 to 499 employees',  
 '2 to 9 employees',  
 '20 to 99 employees',  
 '5,000 to 9,999 employees',  
 '500 to 999 employees',  
 'I don’t know',  
 'Just me - I am a freelancer, sole proprietor, etc.'),  
 labels = c(1,2,3,4,5,6,7,8,9,10))  
   
 df$OrgSize[is.na(df$OrgSize)] <- 10  
   
 df$OrgSize <- as.integer(df$OrgSize)   
   
 return(df)  
}  
  
#function for clean age  
clean\_age <- function(df){  
 df$Age = factor(df$Age,  
 levels = c('Prefer not to say',  
 'Under 18 years old ',  
 '18-24 years old ',   
 '25-34 years old',  
 '35-44 years old',  
 '45-54 years old',  
 '55-64 years old',  
 '65 years or older'),  
 labels = c(0,1,2,3,4,5,6,7))  
 df$Age <- as.integer(df$Age)   
 df$Age[is.na(df$Age)] <- 0   
 return(df)  
}

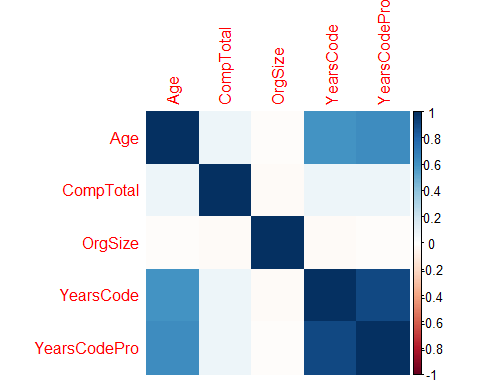
encoding\_function <- function(df){  
 df <- clean\_edlevel(df)  
 df <- clean\_YearsCode(df)  
 df <- clean\_devtype(df)  
 df <- clean\_YearsCodepro(df)  
 df <- clean\_employment(df)  
 df <- clean\_orgsize(df)  
 df <- clean\_age(df)  
 return(df)  
}  
  
df\_cleaned <- encoding\_function(df\_cleaned)  
  
  
  
X\_cols\_to\_be\_scaled <- c("YearsCodePro","YearsCode","OrgSize","Age","CompTotal")  
# Slicing needed columns and standardizing and recombining with original DF  
to\_scale <- df\_cleaned %>% select(X\_cols\_to\_be\_scaled)  
scaled<- scale(to\_scale)  
stadardized\_df <-df\_cleaned  
stadardized\_df[X\_cols\_to\_be\_scaled] <- scaled  
  
  
df\_cleaned %>% correlate() %>% focus(CompTotal)

## # A tibble: 7 × 2  
## term CompTotal  
## <chr> <dbl>  
## 1 OrgSize -0.0222  
## 2 YearsCode 0.0729  
## 3 YearsCodePro 0.0770  
## 4 Age 0.0712  
## 5 WorkExp 0.0822  
## 6 ConvertedCompYearly 0.775   
## 7 DevTypeEncoded 0.0601

df\_cleaned %>% correlate() %>% focus(ConvertedCompYearly)

## # A tibble: 7 × 2  
## term ConvertedCompYearly  
## <chr> <dbl>  
## 1 OrgSize -0.0178  
## 2 YearsCode 0.0425  
## 3 YearsCodePro 0.0448  
## 4 Age 0.0437  
## 5 WorkExp 0.0494  
## 6 CompTotal 0.775   
## 7 DevTypeEncoded 0.0498

# Correlation Plot of columns that needs to be scaled  
corr = cor(stadardized\_df[X\_cols\_to\_be\_scaled])  
corrplot(corr, method = 'color', order = 'alphabet')



## Splitting of data into training and testing

set.seed(123)  
# Taking Random Numbers from a list and sepating train, test with repect to the values  
sample\_size <- floor(0.75 \* nrow(stadardized\_df))  
  
train\_ind <- sample(seq\_len(nrow(stadardized\_df)), size = sample\_size)  
  
# Train Split  
train <- stadardized\_df[train\_ind, ]  
# Test SPlit  
test <-stadardized\_df[-train\_ind, ]  
  
# Independent Varaibles Split  
X <- stadardized\_df[,!(names(stadardized\_df) %in% c("CompTotal"))]  
  
# dependent Varaibles Split  
Y <- stadardized\_df["CompTotal"]  
  
Xtrain <- X[train\_ind, ]  
Ytrain <- Y[train\_ind, ]  
Xtest <- X[-train\_ind, ]  
Ytest <- Y[-train\_ind, ]  
  
  
# Applying Linear Model for all varaiables  
lm <- lm(CompTotal~.,train)  
summary(lm)

##   
## Call:  
## lm(formula = CompTotal ~ ., data = train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -41.627 -0.095 -0.062 -0.017 7.258   
##   
## Coefficients:  
## Estimate  
## (Intercept) -0.303006799943  
## OrgSize -0.004368830746  
## YearsCode -0.004031184279  
## YearsCodePro -0.005040235267  
## Age 0.013087298732  
## WorkExp 0.004877095437  
## ConvertedCompYearly 0.000001149565  
## EdLevelBachelor's degree (B.A., B.S., B.Eng., etc.) -0.015224391797  
## EdLevelMaster's degree (M.A., M.S., M.Eng., MBA, etc.) -0.030285326214  
## EdLevelOther doctoral degree (Ph.D., Ed.D., etc.) 0.001102900244  
## EdLevelPrimary/elementary school -0.011859504195  
## EdLevelProfessional degree (JD, MD, etc.) -0.093104186575  
## EdLevelSecondary school (e.g. American high school, German Realschule or Gymnasium, etc.) -0.033587617004  
## EdLevelSome college/university study without earning a degree -0.027657781653  
## EdLevelSomething else -0.043800508242  
## DevTypeEncoded 0.000000672219  
## EmploymentEmployed, full-time;Employed, part-time -0.022380054278  
## EmploymentEmployed, full-time;Independent contractor, freelancer, or self-employed -0.014258553219  
## EmploymentEmployed, full-time;Independent contractor, freelancer, or self-employed;Employed, part-time -0.026973674827  
## EmploymentEmployed, full-time;Retired -0.073877209645  
## EmploymentEmployed, part-time -0.036321330046  
## EmploymentEmployed, part-time;Retired -0.270102829690  
## EmploymentFreelancer -0.031449431396  
## EmploymentIndependent contractor, freelancer, or self-employed;Employed, part-time -0.079304000157  
## Std. Error  
## (Intercept) 0.033828014755  
## OrgSize 0.004016474281  
## YearsCode 0.009740134543  
## YearsCodePro 0.012295915675  
## Age 0.005433480779  
## WorkExp 0.001189233728  
## ConvertedCompYearly 0.000000005625  
## EdLevelBachelor's degree (B.A., B.S., B.Eng., etc.) 0.023274004954  
## EdLevelMaster's degree (M.A., M.S., M.Eng., MBA, etc.) 0.023974099053  
## EdLevelOther doctoral degree (Ph.D., Ed.D., etc.) 0.032260137870  
## EdLevelPrimary/elementary school 0.060324417686  
## EdLevelProfessional degree (JD, MD, etc.) 0.039479554153  
## EdLevelSecondary school (e.g. American high school, German Realschule or Gymnasium, etc.) 0.029289303729  
## EdLevelSome college/university study without earning a degree 0.025318368449  
## EdLevelSomething else 0.048818778465  
## DevTypeEncoded 0.000000152306  
## EmploymentEmployed, full-time;Employed, part-time 0.062106344951  
## EmploymentEmployed, full-time;Independent contractor, freelancer, or self-employed 0.015084654089  
## EmploymentEmployed, full-time;Independent contractor, freelancer, or self-employed;Employed, part-time 0.064474723399  
## EmploymentEmployed, full-time;Retired 0.567892962753  
## EmploymentEmployed, part-time 0.031407885548  
## EmploymentEmployed, part-time;Retired 0.568021206494  
## EmploymentFreelancer 0.020555118751  
## EmploymentIndependent contractor, freelancer, or self-employed;Employed, part-time 0.059081755124  
## t value  
## (Intercept) -8.957  
## OrgSize -1.088  
## YearsCode -0.414  
## YearsCodePro -0.410  
## Age 2.409  
## WorkExp 4.101  
## ConvertedCompYearly 204.355  
## EdLevelBachelor's degree (B.A., B.S., B.Eng., etc.) -0.654  
## EdLevelMaster's degree (M.A., M.S., M.Eng., MBA, etc.) -1.263  
## EdLevelOther doctoral degree (Ph.D., Ed.D., etc.) 0.034  
## EdLevelPrimary/elementary school -0.197  
## EdLevelProfessional degree (JD, MD, etc.) -2.358  
## EdLevelSecondary school (e.g. American high school, German Realschule or Gymnasium, etc.) -1.147  
## EdLevelSome college/university study without earning a degree -1.092  
## EdLevelSomething else -0.897  
## DevTypeEncoded 4.414  
## EmploymentEmployed, full-time;Employed, part-time -0.360  
## EmploymentEmployed, full-time;Independent contractor, freelancer, or self-employed -0.945  
## EmploymentEmployed, full-time;Independent contractor, freelancer, or self-employed;Employed, part-time -0.418  
## EmploymentEmployed, full-time;Retired -0.130  
## EmploymentEmployed, part-time -1.156  
## EmploymentEmployed, part-time;Retired -0.476  
## EmploymentFreelancer -1.530  
## EmploymentIndependent contractor, freelancer, or self-employed;Employed, part-time -1.342  
## Pr(>|t|)  
## (Intercept) < 0.0000000000000002  
## OrgSize 0.2767  
## YearsCode 0.6790  
## YearsCodePro 0.6819  
## Age 0.0160  
## WorkExp 0.0000413  
## ConvertedCompYearly < 0.0000000000000002  
## EdLevelBachelor's degree (B.A., B.S., B.Eng., etc.) 0.5130  
## EdLevelMaster's degree (M.A., M.S., M.Eng., MBA, etc.) 0.2065  
## EdLevelOther doctoral degree (Ph.D., Ed.D., etc.) 0.9727  
## EdLevelPrimary/elementary school 0.8441  
## EdLevelProfessional degree (JD, MD, etc.) 0.0184  
## EdLevelSecondary school (e.g. American high school, German Realschule or Gymnasium, etc.) 0.2515  
## EdLevelSome college/university study without earning a degree 0.2747  
## EdLevelSomething else 0.3696  
## DevTypeEncoded 0.0000102  
## EmploymentEmployed, full-time;Employed, part-time 0.7186  
## EmploymentEmployed, full-time;Independent contractor, freelancer, or self-employed 0.3445  
## EmploymentEmployed, full-time;Independent contractor, freelancer, or self-employed;Employed, part-time 0.6757  
## EmploymentEmployed, full-time;Retired 0.8965  
## EmploymentEmployed, part-time 0.2475  
## EmploymentEmployed, part-time;Retired 0.6344  
## EmploymentFreelancer 0.1260  
## EmploymentIndependent contractor, freelancer, or self-employed;Employed, part-time 0.1795  
##   
## (Intercept) \*\*\*  
## OrgSize   
## YearsCode   
## YearsCodePro   
## Age \*   
## WorkExp \*\*\*  
## ConvertedCompYearly \*\*\*  
## EdLevelBachelor's degree (B.A., B.S., B.Eng., etc.)   
## EdLevelMaster's degree (M.A., M.S., M.Eng., MBA, etc.)   
## EdLevelOther doctoral degree (Ph.D., Ed.D., etc.)   
## EdLevelPrimary/elementary school   
## EdLevelProfessional degree (JD, MD, etc.) \*   
## EdLevelSecondary school (e.g. American high school, German Realschule or Gymnasium, etc.)   
## EdLevelSome college/university study without earning a degree   
## EdLevelSomething else   
## DevTypeEncoded \*\*\*  
## EmploymentEmployed, full-time;Employed, part-time   
## EmploymentEmployed, full-time;Independent contractor, freelancer, or self-employed   
## EmploymentEmployed, full-time;Independent contractor, freelancer, or self-employed;Employed, part-time   
## EmploymentEmployed, full-time;Retired   
## EmploymentEmployed, part-time   
## EmploymentEmployed, part-time;Retired   
## EmploymentFreelancer   
## EmploymentIndependent contractor, freelancer, or self-employed;Employed, part-time   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.5677 on 20153 degrees of freedom  
## Multiple R-squared: 0.6788, Adjusted R-squared: 0.6785   
## F-statistic: 1852 on 23 and 20153 DF, p-value: < 0.00000000000000022

## Conclusion

Based on the multiple regression analysis that we performed for predicting the salary, we can see that the Age and the work experience have the p-value less than the significance level of 0.05. This results shows that these predictors have the statistically significant relationship with the salary.

Also, the Adj. R-squared value is found to be 67.84%. This suggests that the model is a good fit for the data and that a significant proportion of the variation in the dependent variable (salary) can be explained by the independent variables like Age, work experience, Organization size etc. included in the model. However, it is important to note that there may still be some variation in the dependent variable that is not accounted for by this model. Therefore, further analysis may be necessary to identify additional variables that could improve the model’s predictive power.

## Refrences

Fang, X., & Wu, M. (2019). Salary prediction with random forest. Journal of Computational Science, 34, 65-74.

Mukherjee, S., & Kumar, S. (2019). Impact of location on salary prediction: An empirical study. International Journal of Data Science and Analytics, 8(1), 25-38.

<https://www.kaggle.com/code/klmsathishkumar/stack-overflow-survey-eda-salary-prediction>

<https://baescott.medium.com/predicting-personal-compensation-with-survey-data-from-stack-overflow-3dfff4832a4b>