PSTAT 131 Final Project: Model comparison for predicting the salary of a data science job

Nicholas Axl Andrian

2023-11-23

Dataset used: https://www.kaggle.com/datasets/arnabchaki/data-science-salaries-2023

In this project, we will be fitting several different machine learning algorithms to find out which method of prediction is the most accurate in getting the predicted salary(in usd).

About the dataset's variables (excerpt from the kaggle site)

- work_year: The year the salary was paid.
- experience_level: The experience level in the job during the year
- employment_type: The type of employment for the role
- job_title: The role worked in during the year.
- salary: The total gross salary amount paid.
- salary_currency: The currency of the salary paid as an ISO 4217 currency code.
- salaryinusd: The salary in USD
- employee_residence: Employee's primary country of residence in during the work + year as an ISO 3166 country code.
- remote_ratio: The overall amount of work done remotely
- company_location: The country of the employer's main office or contracting branch
- company_size: The median number of people that worked for the company during the year

```
library(dplyr)
library(randomForest)
library(gbm)
library(ISLR)
library(tree)
library(tidyverse)

options("max.print" = 10) # to prevent page number bloat
```

Part 1: Exploratory Data Analysis

```
salaries <- read.csv("ds_salaries.csv")
```

Loading the dataset

```
head(salaries)
```

Checking the structure of the dataset

```
##      work_year experience_level employment_type job_title salary
##      salary_currency salary_in_usd employee_residence remote_ratio
##      company_location company_size
##  [ reached 'max' / getOption("max.print") -- omitted 6 rows ]
```

```
str(salaries)
```

```
## 'data.frame':    3755 obs. of  11 variables:
##  $ work_year         : int  2023 2023 2023 2023 2023 2023 2023 2023 2023 2023 ...
##  $ experience_level  : chr  "SE" "MI" "MI" "SE" ...
##  $ employment_type   : chr  "FT" "CT" "CT" "FT" ...
##  $ job_title         : chr  "Principal Data Scientist" "ML Engineer" "ML Engineer" "Data Scientist"
##  $ salary            : int  80000 30000 25500 175000 120000 222200 136000 219000 141000 147100 ...
##  $ salary_currency   : chr  "EUR" "USD" "USD" "USD" ...
##  $ salary_in_usd     : int  85847 30000 25500 175000 120000 222200 136000 219000 141000 147100 ...
##  $ employee_residence: chr  "ES" "US" "US" "CA" ...
##  $ remote_ratio      : int  100 100 100 100 100 0 0 0 0 0 ...
##  $ company_location  : chr  "ES" "US" "US" "CA" ...
##  $ company_size      : chr  "L" "S" "S" "M" ...
```

Already we can see an issue that needs to be worked on. Several variables seem to supposedly be read in as factors. We will finish conducting checks on the dataset before converting said columns.

```
summary(salaries)
```

Checking the summary of the dataset

```
##    work_year    experience_level   employment_type     job_title
##      salary         salary_currency    salary_in_usd    employee_residence
##   remote_ratio    company_location   company_size
##  [ reached getOption("max.print") -- omitted 6 rows ]
```

```
colSums(is.na(salaries))
```

Checking for null values

```
##          work_year   experience_level    employment_type          job_title
##                  0                  0                  0                  0
##             salary    salary_currency      salary_in_usd employee_residence
##                  0                  0                  0                  0
##       remote_ratio   company_location       company_size
##                  0                  0                  0
```

Fortunately, we have no null values so imputing is not required

```r
factor_cols <- salaries[, c(2, 3, 4, 6, 8, 10, 11)]

# finding unique values, referenced code from https://www.kaggle.com/code/abdulfaheem11/data-science-sa

# output ommitted to prevent too much space being taken up
sapply(factor_cols, function(col) unique(col))
```

Checking potential factor columns for their unique values

```
## $experience_level
## [1] "SE" "MI" "EN" "EX"
##
## $employment_type
## [1] "FT" "CT" "FL" "PT"
##
## $job_title
##  [1] "Principal Data Scientist"       "ML Engineer"
##  [3] "Data Scientist"                 "Applied Scientist"
##  [5] "Data Analyst"                   "Data Modeler"
##  [7] "Research Engineer"              "Analytics Engineer"
##  [9] "Business Intelligence Engineer" "Machine Learning Engineer"
##  [ reached getOption("max.print") -- omitted 83 entries ]
##
## $salary_currency
##  [1] "EUR" "USD" "INR" "HKD" "CHF" "GBP" "AUD" "SGD" "CAD" "ILS"
##  [ reached getOption("max.print") -- omitted 10 entries ]
##
## $employee_residence
##  [1] "ES" "US" "CA" "DE" "GB" "NG" "IN" "HK" "PT" "NL"
##  [ reached getOption("max.print") -- omitted 68 entries ]
##
## $company_location
##  [1] "ES" "US" "CA" "DE" "GB" "NG" "IN" "HK" "NL" "CH"
##  [ reached getOption("max.print") -- omitted 62 entries ]
##
## $company_size
## [1] "L" "S" "M"
```

Changing said variables to become factors

```r
salaries[, c(2, 3, 4, 6, 8, 10, 11)] <- lapply(factor_cols, factor)
str(salaries)
```

```
## 'data.frame':    3755 obs. of  11 variables:
##  $ work_year         : int  2023 2023 2023 2023 2023 2023 2023 2023 2023 2023 ...
##  $ experience_level  : Factor w/ 4 levels "EN","EX","MI",..: 4 3 3 4 4 4 4 4 4 4 ...
##  $ employment_type   : Factor w/ 4 levels "CT","FL","FT",..: 3 1 1 3 3 3 3 3 3 3 ...
##  $ job_title         : Factor w/ 93 levels "3D Computer Vision Researcher",..: 85 78 78 48 48 9 9 48
##  $ salary            : int  80000 30000 25500 175000 120000 222200 136000 219000 141000 147100 ...
##  $ salary_currency   : Factor w/ 20 levels "AUD","BRL","CAD",..: 8 20 20 20 20 20 20 20 20 20 ...
##  $ salary_in_usd     : int  85847 30000 25500 175000 120000 222200 136000 219000 141000 147100 ...
```

```
##  $ employee_residence: Factor w/ 78 levels "AE","AM","AR",..: 27 76 76 12 12 76 76 12 12 76 ...
##  $ remote_ratio      : int  100 100 100 100 100 0 0 0 0 0 ...
##  $ company_location  : Factor w/ 72 levels "AE","AL","AM",..: 26 71 71 13 13 71 71 13 13 71 ...
##  $ company_size      : Factor w/ 3 levels "L","M","S": 1 3 3 2 2 1 1 2 2 2 ...
```

We can also drop the salary and salary_currency as we will just be using the salary_in_usd to simplify our steps.

```
salaries <- salaries[, !(names(salaries) %in% c('salary_currency', 'salary'))]
```

Visualization to search for patterns with regards to the salary_in_usd