UNIVERSITAT POLITÈCNICA DE CATALUNYA
FACULTAT D'INFORMÀTICA DE BARCELONA

DATA MINING COURSE
DELIVERY D4

Report delivery date:	27 / 3 / 2022
Group members:	Antón Calle, Nuria
	Ivaylov Patchaliev, Stefan
	Piotrowski, Maciej
	Rodulfo Cárdenas, Alejandro
	Rubio Serrano, Juan Diego
	Treviño Gutiérrez, Tomàs

Contents

Project introduction

Motivation

Initially, we have searched for some information about the topics we could use to do a data mining project and even though we have found a lot of interesting datasets to use, we have decided to use a dataset quite close to our field: “Data Science and STEM Salaries”.

We have chosen this one mainly because all team members are Computer Engineers and one day they could be working in this field. Also, we have compared a lot of the datasets we have initially seen and this has one of the most complete and useful information, and it is easier to do a study and to prevent bad predictions.

General description of the problem

Our objective in this project is to be able to predict with accuracy the salary of a worker from all data we have collected from the worker.

Data source Presentation

The data consists of information about Data Science and STEM workers. We want to focus our study on the wages that every worker gets, aiming to be able to predict the salary that a person can earn during their lifetime according to some basic information about them.

We first encountered that the dataset was bigger (nearly 62.000 examples) than what we needed (between 2.000 and 5.000 examples). That is why we deleted examples and to do so we decided that it was interesting to drop examples that are not from the United States of America and other filter. We also found there are columns in the original dataset which gave us useless information, so we dropped them.

Here you can find a direct link to the original dataset: Data Science and STEM Salaries dataset

Metadata

As we have said previously, we have erased some of the columns of the original dataset because it had some useless information for our project. Specifically *other details*, *timestamp*, *dmaid* and *number of row*. Furthermore, we decided to choose the 14 variables that seem more valuable for us. Our final dataset has the next variables:

Numerical

Variable Name	Meaning	Type	Measuring unit	Missing code	Missing %	Range	Role	Measuring procedure
totalyearlycompensation	Money spent by the company for the position in a year.	Discrete	years		0	[28000,1400000]	Explanatory	Person’s answer
yearsofexperience	How many years working at the titulation are needed	Discrete	years		0	[0,42]	Explanatory	Person’s answer
yearsatcompany	How many years spent working on it	Discrete	years		0	[0,25]	Explanatory	Person’s answer
basesalary	The money the worker receives directly per year	Discrete	$/year		0	[0,750000]	Explanatory	Person’s answer
stockgrantvalue	The average value per year of the stock granted	Discrete	avg/year		0	[0,510000]	Explanatory	Person’s answer
bonus	An average of the bonus per year the worker receives	Discrete	avg/year		0	[0,265000]	Explanatory	Person’s answer

Categorical

Variable name	Meaning	Modalities	Type	Missing code	Missing %	Role	Measuring procedure
company	Name of the company	*"Google" "Microsoft" "Amazon" "Facebook" "Others" "Apple"*	Nominal		0	Explanatory	Person’s answer
level	The level inside the company	*"L6" "L4" "L5" "L2" "L3" "L1"*	Ordinal		0	Explanatory	Person’s answer

title	Titulation of the work	*"Software Engineer" "Solution Architect" "Technical Program Manager" "Data Scientist"* *"Product Manager" "Recruiter" "Software Engineering Manager" "Sales"* *"Business Analyst" "Marketing" "Human Resources" "Product Designer"* *"Mechanical Engineer" "Management Consultant"*	Nominal		0	Explanatory	Person’s answer
location	The region where the work is.	*"REGION4" "REGION2" "REGION3" "REGION1"*	Nominal		0	Explanatory	Location of the job
tag	Extra information about the work	*"Distributed Systems (Back-End)" "DevOps" "Full Stack" "ML / AI"* *"Web Development (Front-End)" "Networking" "Security" "Data"* *"Product"*	Nominal		0	Explanatory	Person’s answer
gender	Gender of the person	*“Male” “Female” “Other”*	Nominal	NA	6,3735	Explanatory	Person’s answer
Masters_Degree	Tells if the worker has a master’s degree (1) or not (0)	*0* ; *1*	Binary		0	Explanatory	Person’s answer
Bachelors_Degree	Tells if the worker has a bachelor’s degree (1) or not (0)	*0* ; *1*	Binary		0	Explanatory	Person’s answer
Doctorate_Degree	Tells if the worker has a doctorate’s degree (1) or not (0)	*0* ; *1*	Binary		0	Explanatory	Person’s answer
Highschool	Tells if the worker went to highschool (1) or not (0)	*0* ; *1*	Binary		0	Explanatory	Person’s answer
Some_College	Tells if the worker went to a college (1) or not (0)	*0* ; *1*	Binary		0	Explanatory	Person’s answer
Race_Asian	Tells if the worker is asian (1) or not (0)	*0* ; *1*	Binary		0	Explanatory	Person’s answer
Race_White	Tells if the worker is white (1) or not (0)	*0* ; *1*	Binary		0	Explanatory	Person’s answer

Race_Two_Or_More	Tells if the worker belongs to two or more races (1) or not (0)	*0* ; *1*	Binary		0	Response	Counts if there is more the one 1 on the Race_* columns
Race_Black	Tells if the worker is black (1) or not (0)	*0* ; *1*	Binary		0	Explanatory	Person’s answer
Race_Hispanic	Tells if the worker is hispanic (1) or not (0)	*0* ; *1*	Binary		0	Explanatory	Person’s answer
Race	Tells the race the worker belongs to	*"Asian" "Two Or More" "White" "Hispanic" "Black"*	Nominal	NA	13,4984	Response	The factor of all the races’ columns.
Education	Tells the highest education the worker have	*"PhD" "Bachelor's Degree" "Master's Degree" "Some College" "Highschool"*	Nominal	NA	10,9379	Response	The factor of all the studies columns.

Data Mining process

To the data mining process we follow the structure that we learned in the class and the tasks of the different deliveries. So our final workflow would be the next one:

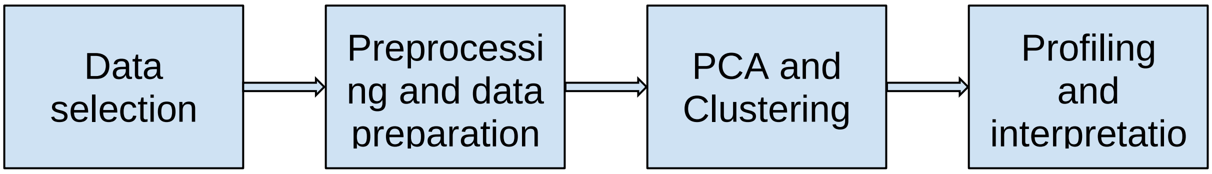

Data selection
Preprocessing and data preparation
PCA and Clustering
Profiling and interpretatio

Figure 1 - Workflow

The first thing we did was select the dataset. As we said in the motivations, we had several ideas for the project, but we chose this one because it was the option that met better the requirements and we thought it was interesting for our own future to learn what we should expect once we go out into the working world. So, we got the dataset of Kaggle and we started to choose the important variants and to do the data cleaning.

While we did the job, we found out problems with the variables, thus we had to come back to the data cleaning to fix all the problems that surged. But at the end, we select the final information that we specified in the data cleaning part.

In addition, in this part we explained too what methods we use for the missings treatment with more detail. But as an introduction, as we had just categorical missings in our dataset, we decided to apply the mice method rather than Knn, because the multiple imputation is considered one of the best strategies for treating the missings.

After these fixes, we could do the PCA, Clustering and the profiling ok. Each one would be more detailed in their respective sections.

Preprocessing and data preparation

First of all, our dataset had a big problem. There were some variables that had a lot of different modalities. So, the first step was to group them as much as possible and select just the most common ones. So we filtered our dataset by the most commons modalities of companies, tags, and level, and then we group the levels, because there was some of them with different name that represented the same level (remind that level is an ordinal variable but each companies has their own terminology).

Furthermore, as we had just a few rows of locations outside the United States, we decided to select only the rows of US to be more accurate. And also, this allowed us to group all the locations, that we converted to the acronym of each state, into their respective official regions. So we have 4 regions that are represented in figure 3.

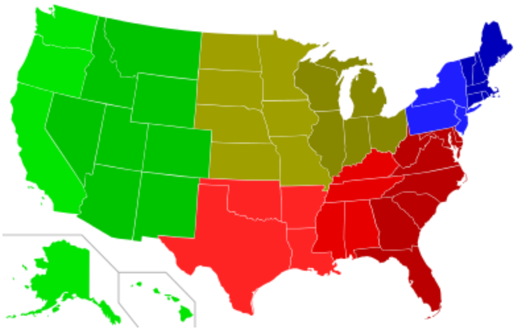


Figure 2 - Official USA regions

Missing data treatment

The first decision to make is if we want to remove the rows that contain missing values or if we want to impute them, which means fill the missing values with some method. Although, it is right that what is better for each case depends on the dataset and our model, the best option usually is to impute the data, because otherwise you are losing important data.

Inside the impute methods, the most common ones, and thus the ones we had more information about, were the Knn and the MICE methods. Knn calculates "n" number of neighbors and substitutes the missing values with the mean of the rows of his "neighborhood". This method usually is accurate but there is a problem when the model has a lot of outliers, because they affect the mean of each neighborhood. Due to this, the “multiple imputation” methods are more accurate than the single ones. So, our other option was Mice, that is also a multiple imputation method.

The Mice method has a first step which is to replace the missing values with the mean and these values are called "placeholders". The second step consists of comeback the missings of one variable and then try to predict these values supposing that it is dependent with the other ones. This step is repeated for each variable that contains missings, and the whole process is repeated "m" times. The disadvantage of this method is that it is so expensive and requires a lot of time if there are so many options or if the dataset is large.

So for our project we use Mice for the reasons mentioned above. In our case, we tried with m=10 but we didn’t notice so much improvement, so we chose 5, which was the default and was quick. Also for the Mice method we use "polyreg" for Race, Education and gender, due to in the Mice documentation is the method recommended for the nominals variables. And to decide which dataset to choose from the 5 imputed, we need to compare the coefficients of the old dataset with the new one. Of the five imputed datasets, the one that seemed most similar was the first, so that was the one we selected to go on.

Outliers treatment

Once we have treated the missing data with what we have decided is the best way to not manipulate the original dataset, we have to check the possible outliers our dataset has.

First of all, the outliers have to be numerical data, so we had to select which of our dataset row are numerical and we got the following ones:

- totalYearlyCompensation -> Row 4
- yearsOfExperience -> Row 6
- yearsAtCompany -> Row 7
- baseSalary -> Row 9
- stockGrantValue -> Row 10
- bonus -> Row 11

Then, we get a summary to check some information about the row and a boxplot from each to see easily if there are any outliers (as shown in figure 3). Once we have seen if there are any potential outliers, we use an auxiliary table to erase those rows with outliers and get yet another boxplot to compare visually (as shown in figure 4).

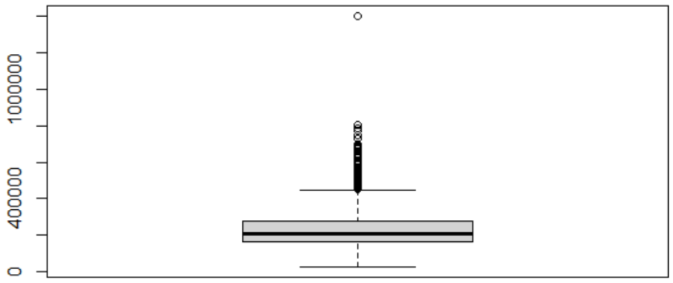


Figure 3 - Initial boxplot (totalYearlyCompensation)

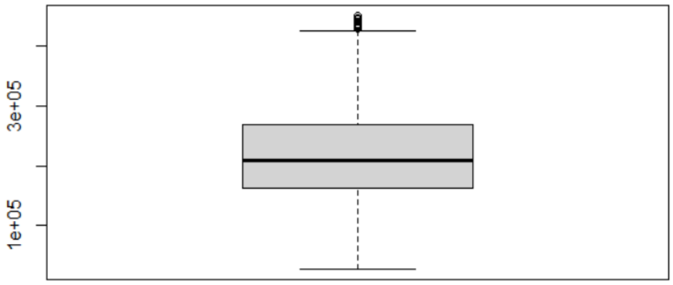


Figure 4 - Final boxplot (totalYearlyCompensation)

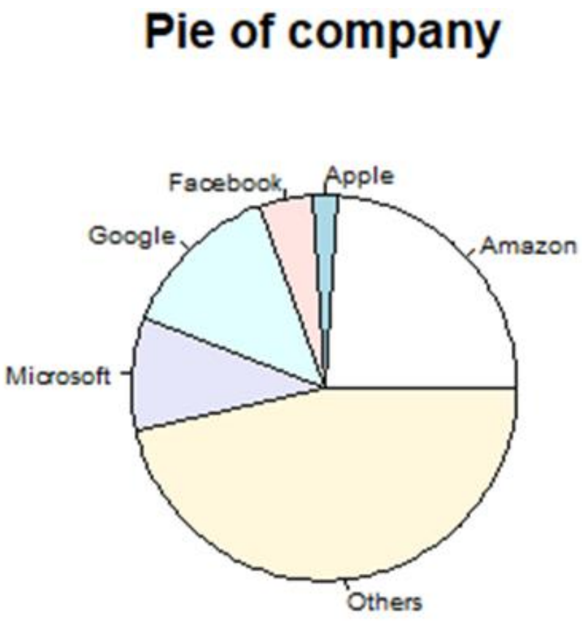
This is done, as previously mentioned, for each numerical column, taking out every row with possible outliers. To clear the dataset we have used the "*$out*" option, in order to get those values far beyond the whiskers in the boxplot (in other words, outliers). Once detected, we erase them from the final dataset.

At the end of the document, we have saved the R Script we have used to treat the outliers.

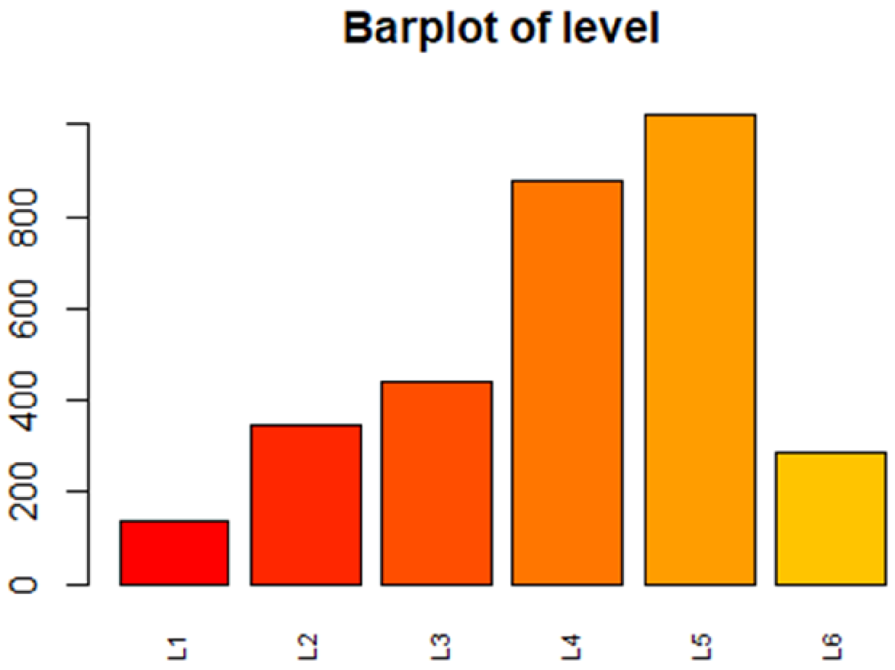
Basic statistical descriptive analysis

Some plots titles are followed by “(before)”. Those are the plots showing the descriptive analysis of the data before the preprocessing. Those plots are included only when we found it relevant :

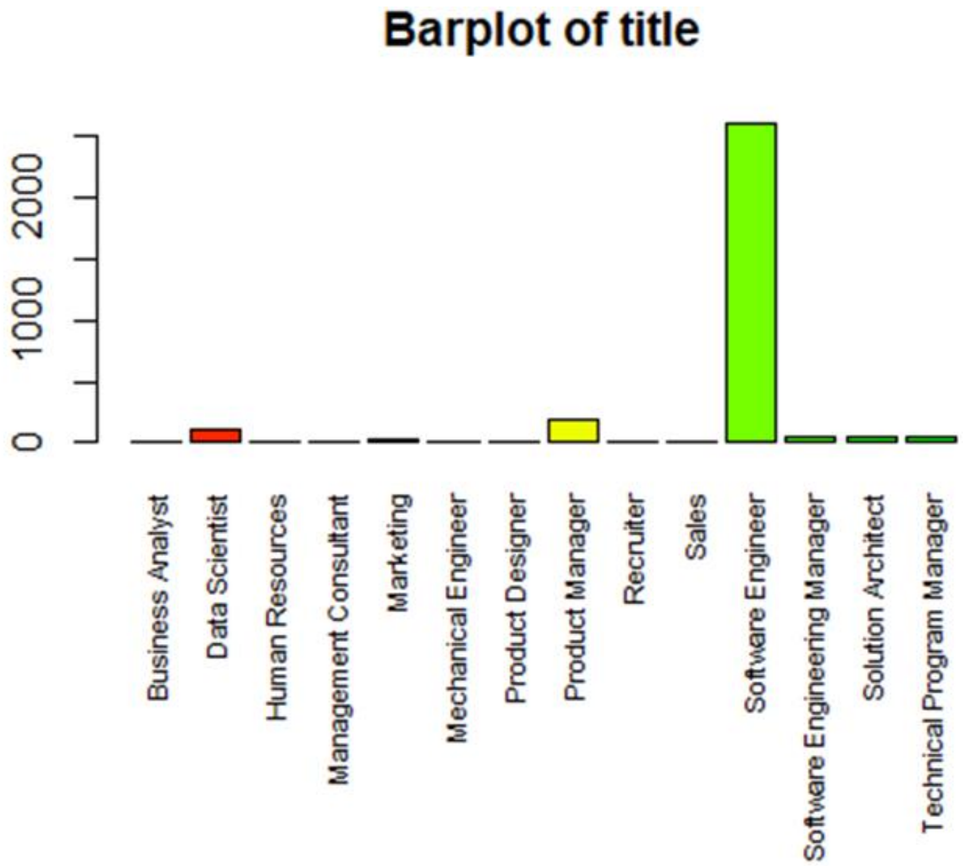
- To show the effect of treating the NA’s
- To show the effect of removing the outliers when needed
- To show the improvement of the clarity of the plot after preprocessing.



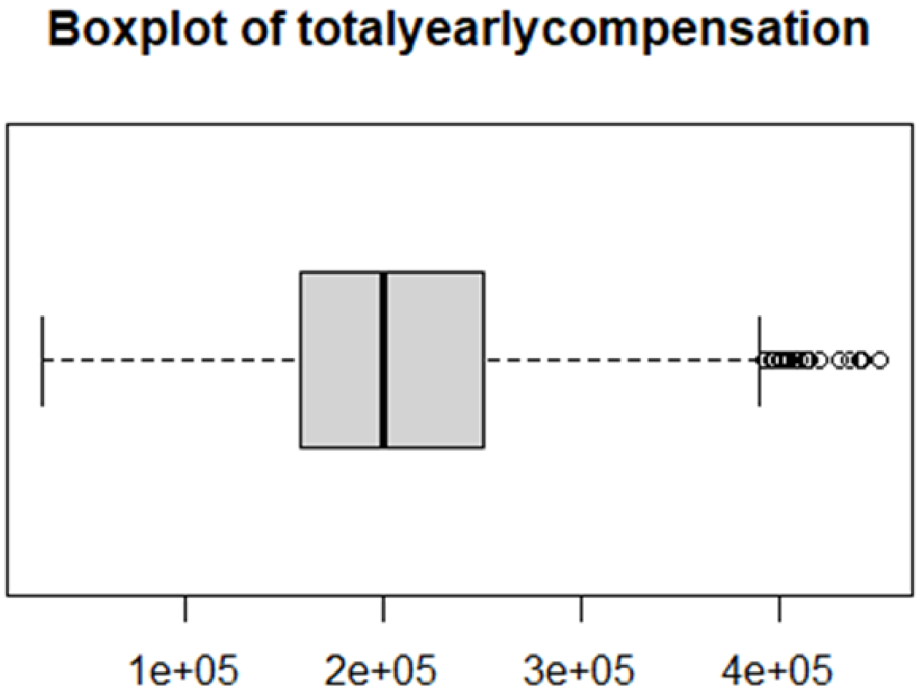
```
## [1] "Number of modalities:  6"

## [1] "Frequency table"

##      Amazon     Apple  Facebook  Google Microsoft       Others

##      740     66      141      426      290      1446

## [1] "Relative frequency table (proportions)"

##      Amazon     Apple  Facebook  Google Microsoft       Others

##  0.238019  0.021229  0.045352  0.137022  0.093278  0.465101

## [1] "Frequency table sorted"

##      Others    Amazon       Google Microsoft  Facebook     Apple

##      1446      740             426      290      141             66

## [1] "Relative frequency table (proportions) sorted"

##      Others    Amazon       Google Microsoft  Facebook     Apple

##  0.465101  0.238019  0.137022  0.093278  0.045352  0.021229
```



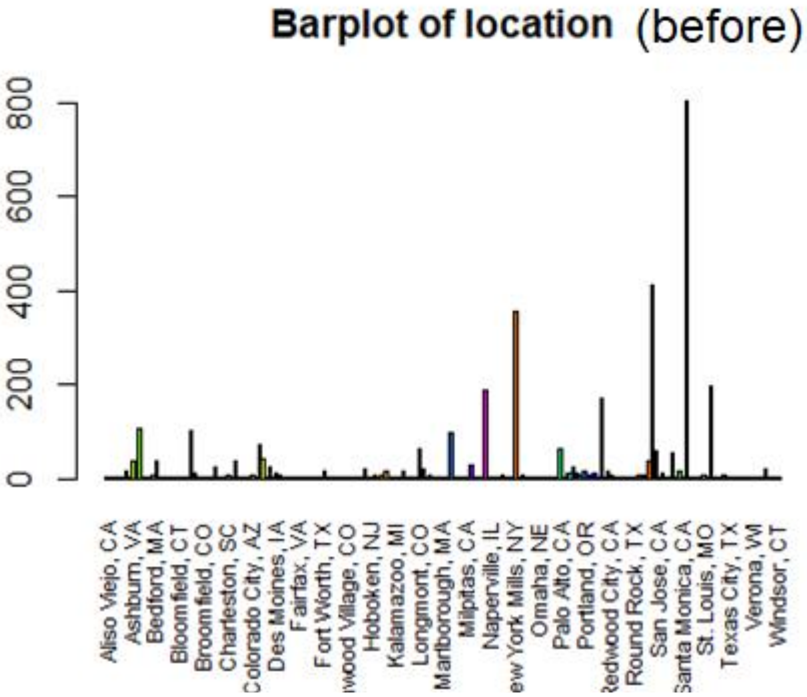
```
## [1] "Number of modalities:  6"

## [1] "Frequency table"

##   L1   L2   L3   L4   L5   L6

##  139  347  440  877 1021  285

## [1] "Relative frequency table (proportions)"

##       L1       L2       L3       L4       L5       L6

## 0.044709 0.111611 0.141525 0.282084 0.328401 0.091669

## [1] "Frequency table sorted"

##   L5   L4   L3   L2   L6   L1

## 1021  877  440  347  285  139

## [1] "Relative frequency table (proportions) sorted"

##       L5       L4       L3       L2       L6       L1

## 0.328401 0.282084 0.141525 0.111611 0.091669 0.044709
```

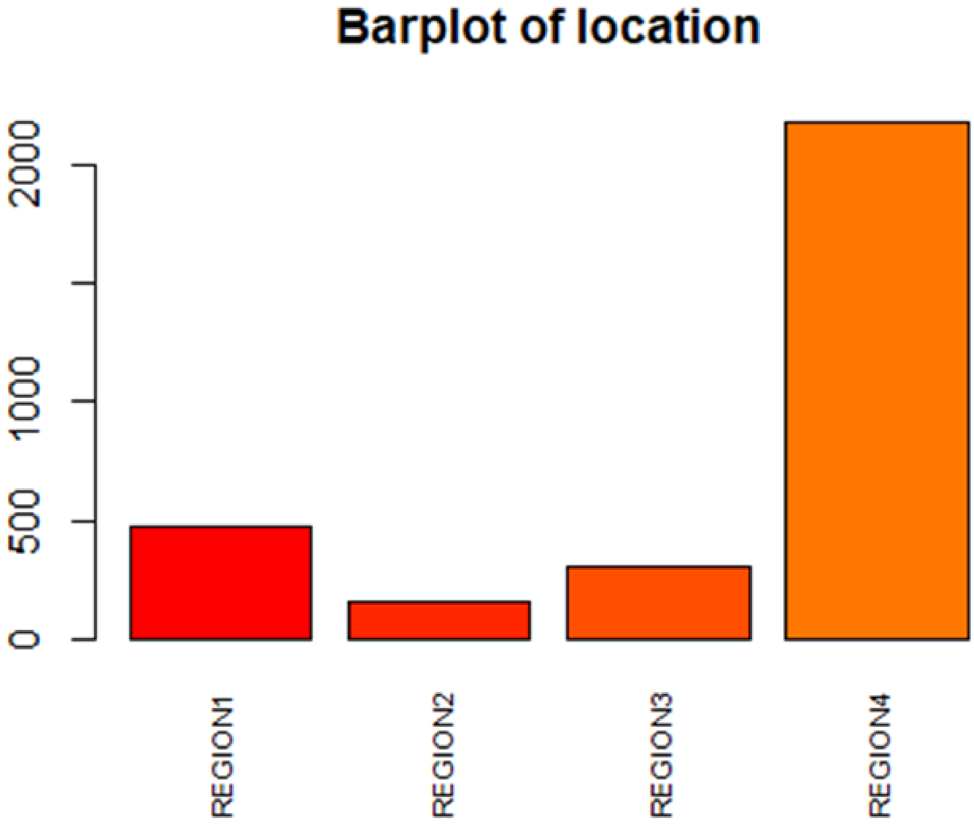


```
## [1] "Number of modalities:  14"

## [1] "Frequency table"

##              Business Analyst            Data Scientist

##                     13                   108

##              Human Resources     Management Consultant

##                     2                    1

##              Marketing     Mechanical Engineer

##                 25                 7

##              Product Designer            Product Manager

##                     6                    189

##              Recruiter                Sales

##                     8                   3
##             Software Engineer Software Engineering Manager
##                          2589                       59
##       Solution Architect      Technical Program Manager
##                          52                     47
## [1] "Relative frequency table (proportions)"
##             Business Analyst              Data Scientist
##             0.004181                0.034738
##             Human Resources     Management Consultant
##             0.000643                0.000322
##             Marketing       Mechanical Engineer
##             0.008041                0.002252
##             Product Designer              Product Manager
##               0.001930              0.060791
##             Recruiter               Sales
##             0.002573                0.000965
##             Software Engineer Software Engineering Manager
##            0.832744                 0.018977
##       Solution Architect      Technical Program Manager
##             0.016726                0.015117
## [1] "Frequency table sorted"
##             Software Engineer             Product Manager
##                          2589                       189
##             Data Scientist Software Engineering Manager
##                     108                   59
##         Solution Architect  Technical Program Manager
##                     52                    47
##             Marketing              Business Analyst
##                 Recruiter          Mechanical Engineer
##                     25                    13
##                     8                     7
##             Product Designer                     Sales
##                     6                     3
##             Human Resources    Management Consultant
##                     2                     1
## [1] "Relative frequency table (proportions) sorted"
##             Software Engineer              Product Manager
##             0.832744                 0.060791
##             Data Scientist Software Engineering Manager
##             0.034738              0.018977
##      Solution Architect     Technical Program Manager
##             0.016726            0.015117
##             Marketing              Business Analyst
##             0.008041              0.004181
##             Recruiter      Mechanical Engineer
##             0.002573              0.002252
##             Product Designer                     Sales
##              0.001930              0.000965
##              Human Resources       Management Consultant
##              0.000643              0.000322
```

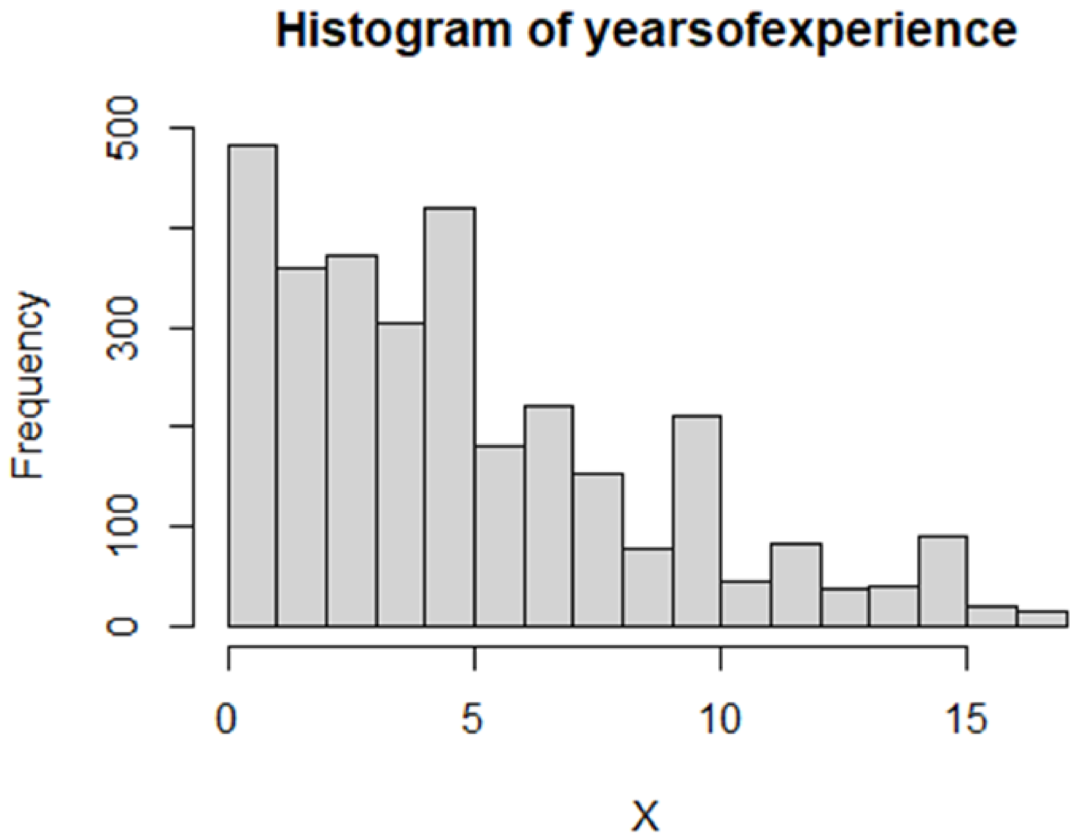


```
## [1] "Extended Summary Statistics"
##      Min. 1st Qu.  Median    Mean 3rd Qu.     Max.
##   28000  158000  200000  208438  251000  450000
## [1] "sd:  71742.4282761407"
## [1] "vc:  0.344190066139208"
```

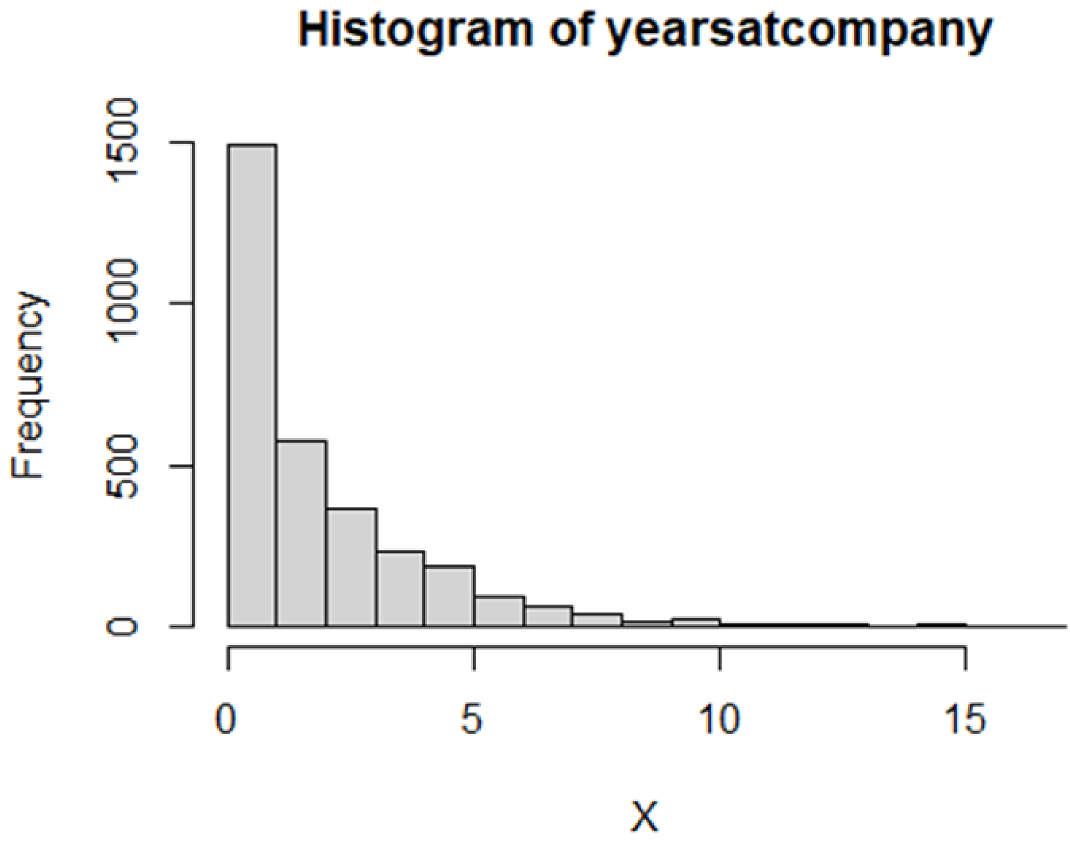




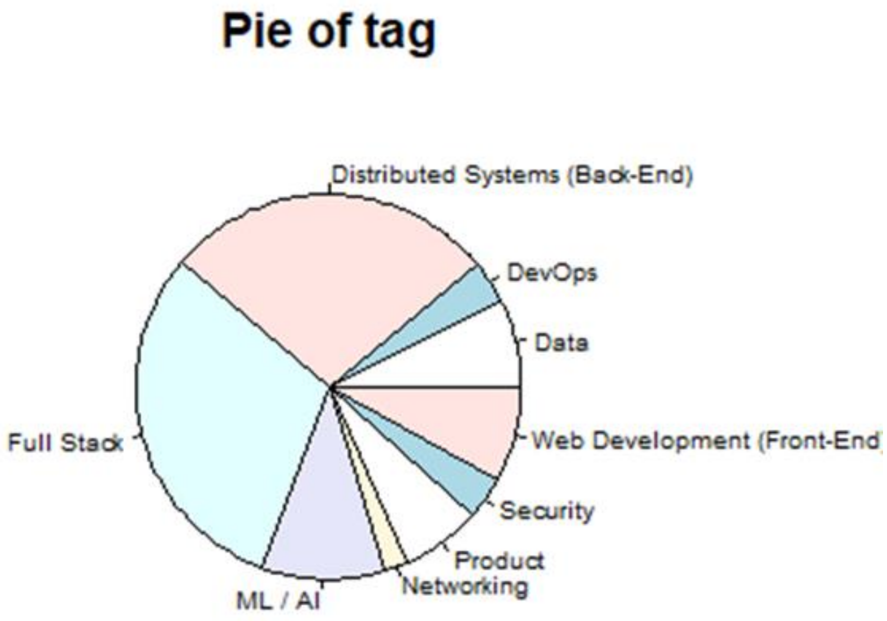
```
## [1] "Number of modalities:  4"
## [1] "Frequency table"

## REGION1 REGION2 REGION3 REGION4

##     471     156     299    2183

## [1] "Relative frequency table (proportions)"

##  REGION1  REGION2  REGION3  REGION4

## 0.151496 0.050177 0.096172 0.702155

## [1] "Frequency table sorted"

## REGION4 REGION1 REGION3 REGION2

##    2183     471     299     156

## [1] "Relative frequency table (proportions) sorted"

##  REGION4  REGION1  REGION3  REGION2

## 0.702155 0.151496 0.096172 0.050177
```



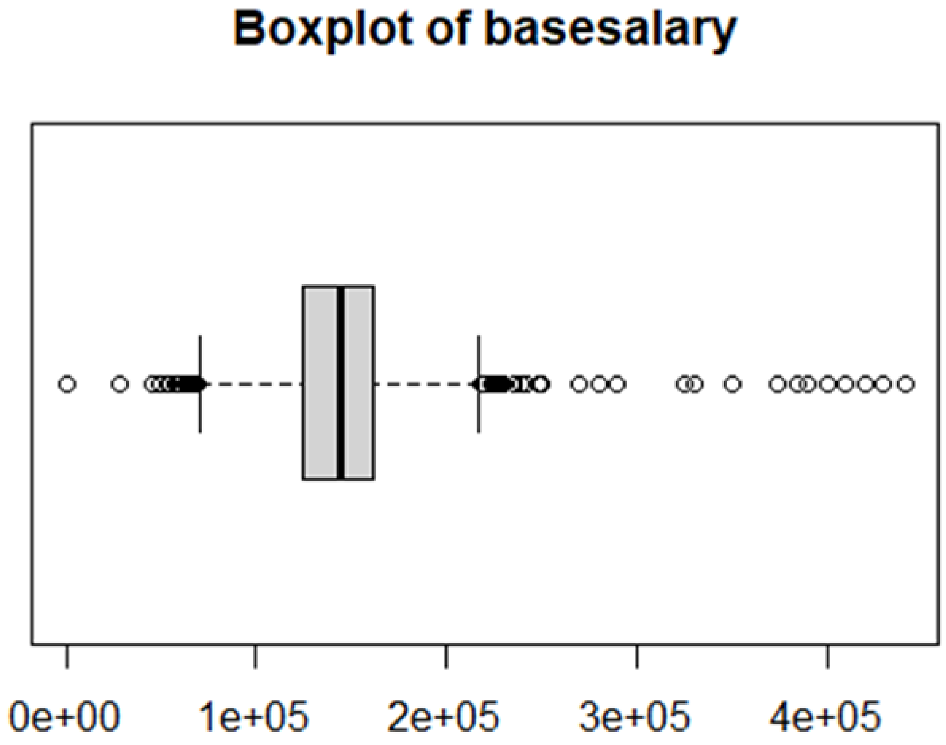
```
## [1] "Extended Summary Statistics"

##       Min. 1st Qu.  Median   Mean 3rd Qu.   Max.

##   0.000   2.000   5.000   5.296   7.000  17.000

## [1] "sd:  3.92994083827739"

## [1] "vc:  0.74209274901785"
```



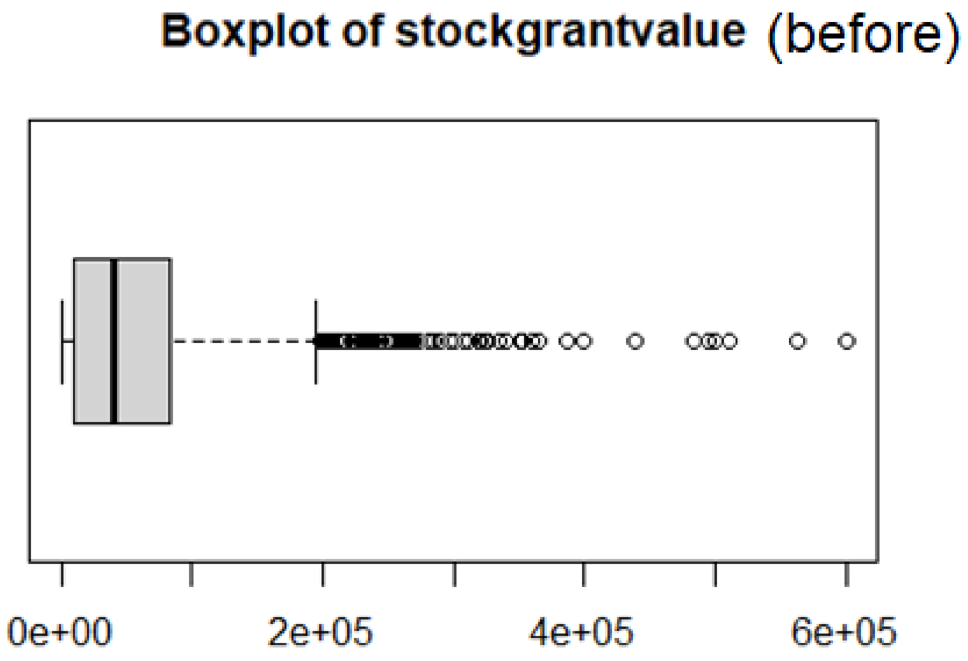
```
## [1] "Extended Summary Statistics"
##       Min. 1st Qu.  Median    Mean 3rd Qu.           Max.
##   0.000   0.000   2.000   2.143   3.000  17.000
## [1] "sd:  2.36868215043969"
## [1] "vc:  1.10508524309411"
```

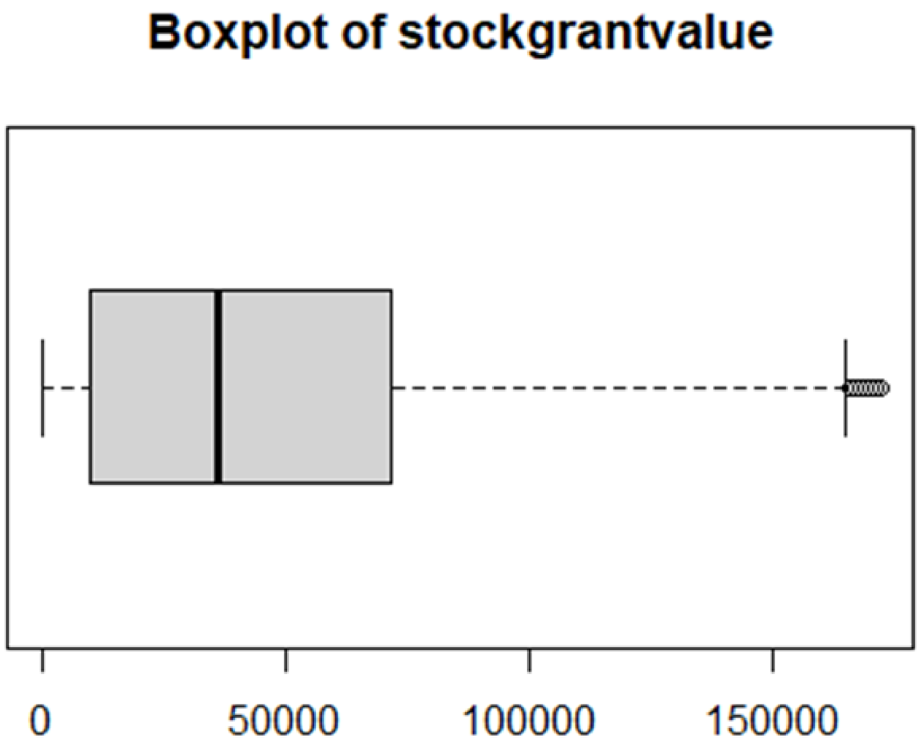


```
## [1] "Number of modalities:  9"
## [1] "Frequency table"
##                    Data                 DevOps
##                    229                   112
## Distributed Systems (Back-End)                 Full Stack
##                    863                  951
##                    ML / AI              Networking
##                    321                  66
##                    Product              Security
##                    204                  110
##       Web Development (Front-End)
##                    253
## [1] "Relative frequency table (proportions)"
##                    Data                 DevOps
##             0.073657             0.036024
## Distributed Systems (Back-End)                 Full Stack
##             0.277581             0.305886
##                    ML / AI              Networking
##             0.103249             0.021229
##                    Product              Security
##             0.065616             0.035381
##       Web Development (Front-End)
##             0.081377
## [1] "Frequency table sorted"
##             Full Stack Distributed Systems (Back-End)
##                  951                    863
##                    ML / AIWeb Development (Front-End)
##                    321                  253
##                    Data                 Product
##                    229                  204
##                     DevOps                 Security
##                     112                    110
##             Networking
##                 66
## [1] "Relative frequency table (proportions) sorted"
##             Full Stack Distributed Systems (Back-End)
##             0.305886               0.277581
##                     ML / AIWeb Development (Front-End)
##             0.103249               0.081377
##                     Data                   Product
##             0.073657               0.065616
##                     DevOps               Security
##             0.036024               0.035381
##             Networking
##             0.021229
```

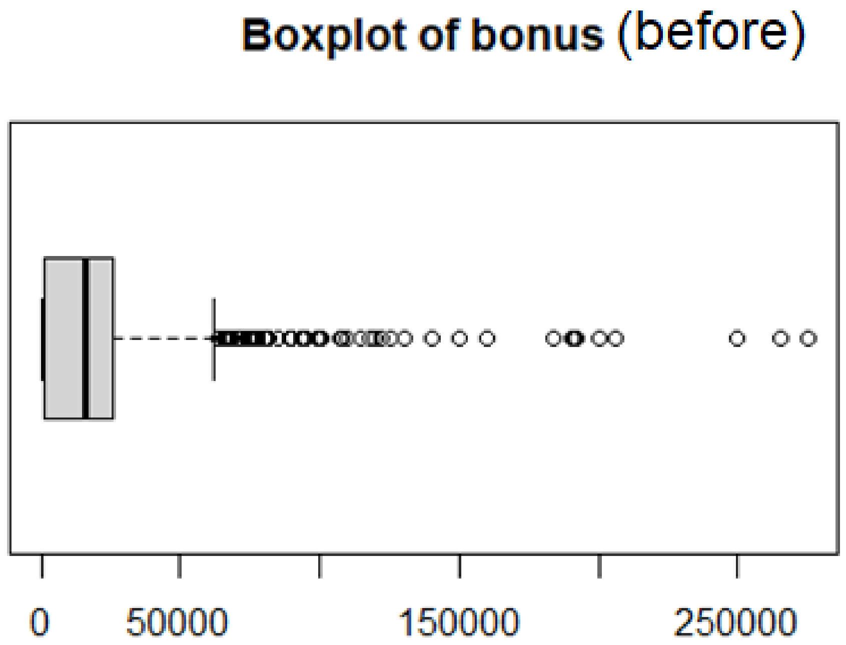


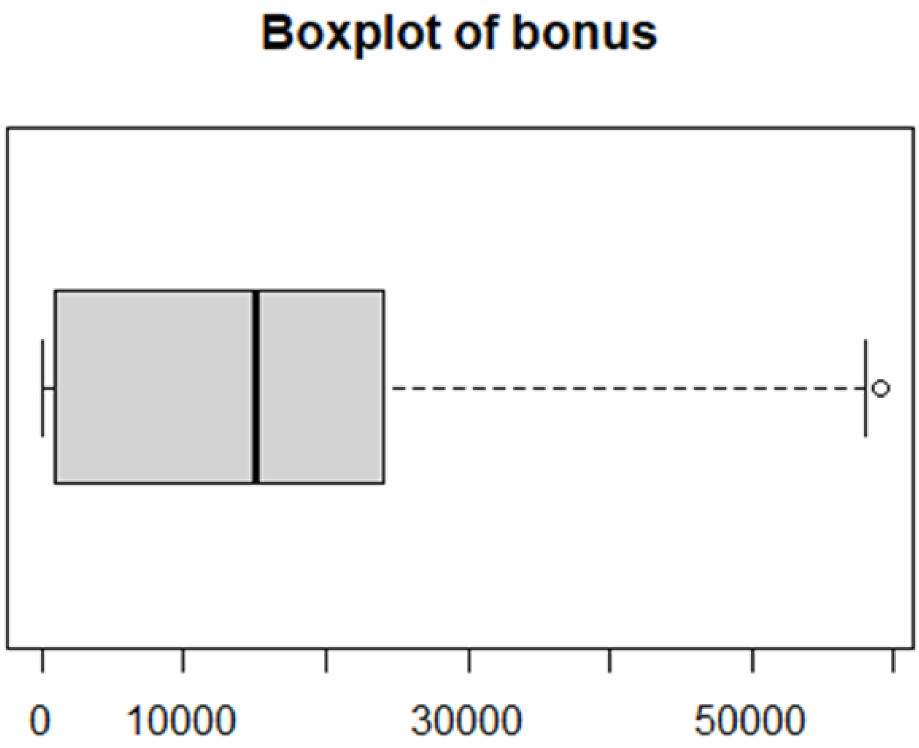
```
## [1] "Extended Summary Statistics"
##       Min. 1st Qu.  Median    Mean 3rd Qu.        Max.
##       0  125000  145000  144254  162000  441000
## [1] "sd:  38142.6772132952"
## [1] "vc:  0.264413711620533"
```



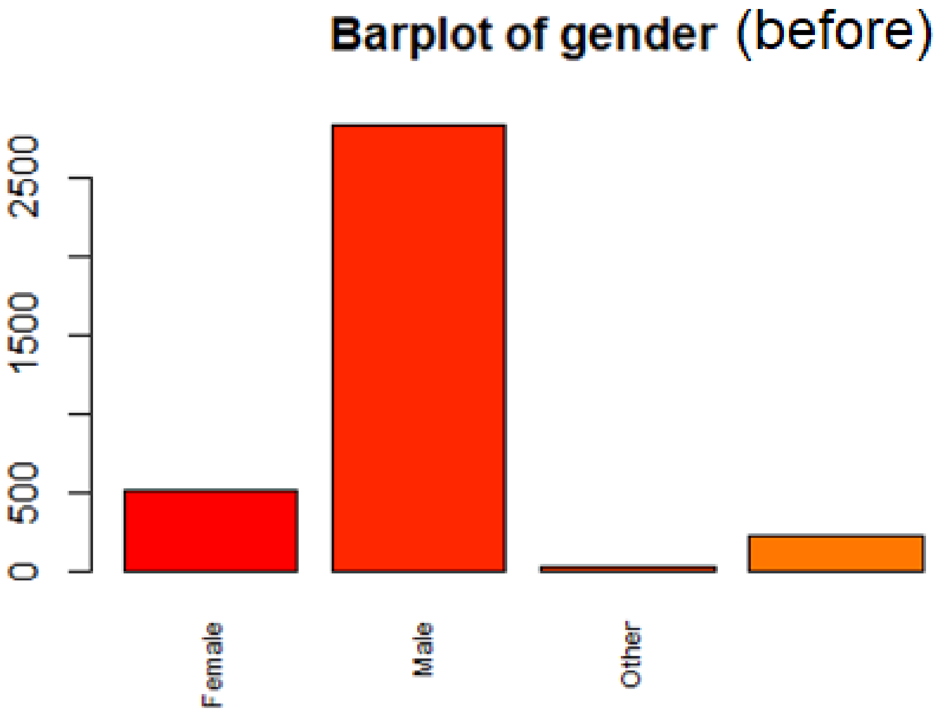


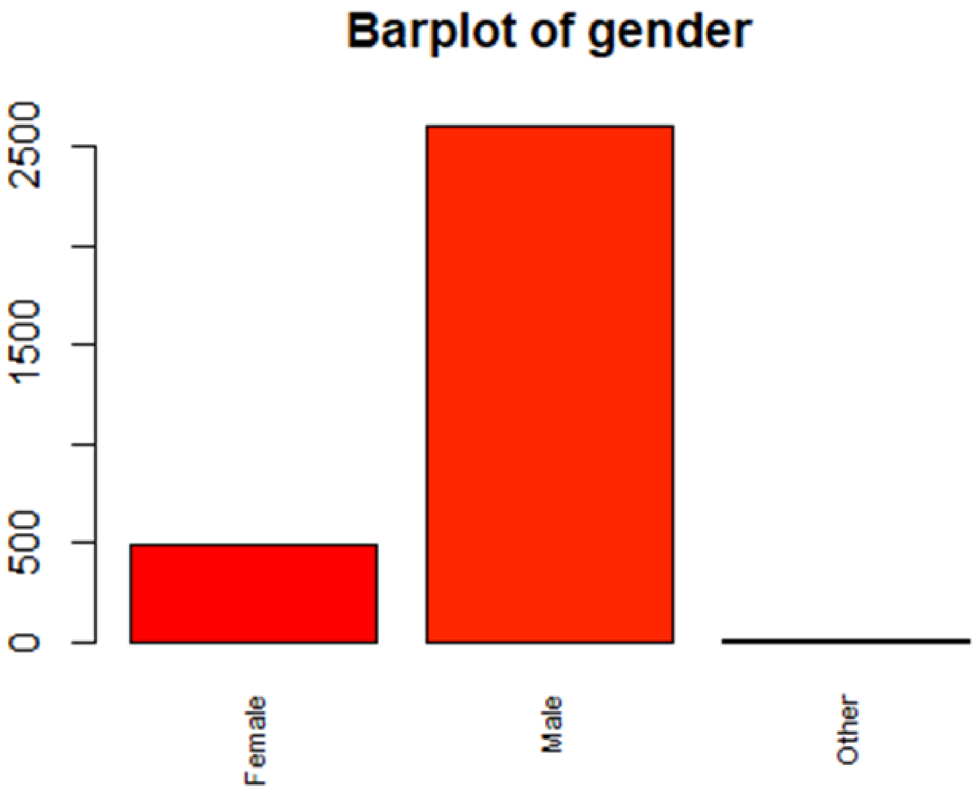
```
## [1] "Extended Summary Statistics"
##      Min. 1st Qu.  Median    Mean 3rd Qu.        Max.
##      0   10000   36000   46088   72000  172000
## [1] "sd:  43114.9700256553"
## [1] "vc:  0.935492920246094"
```



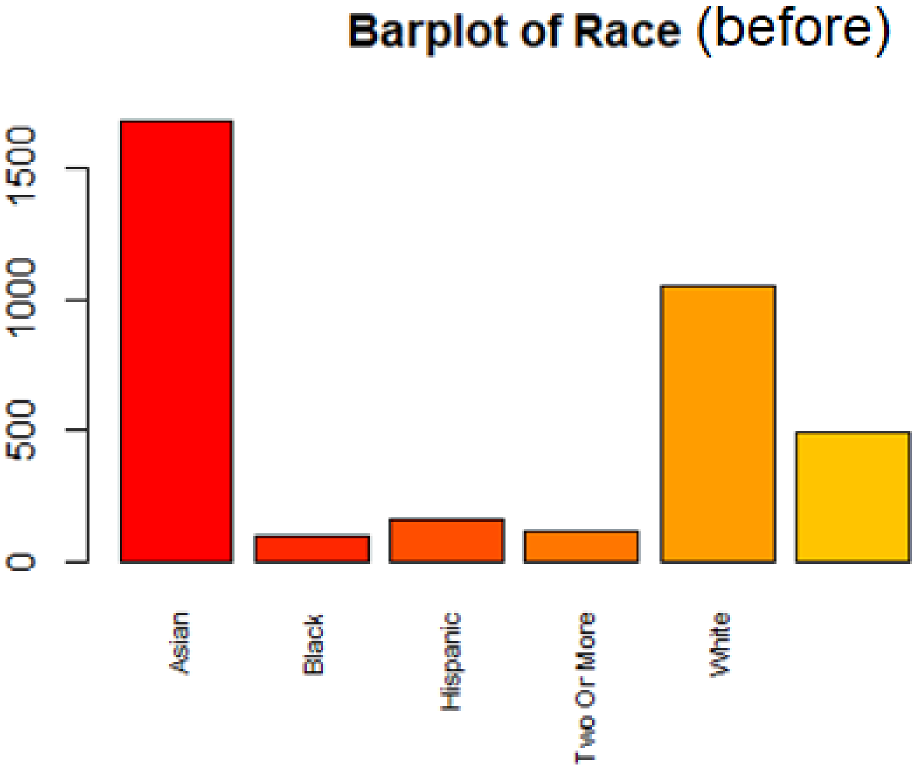


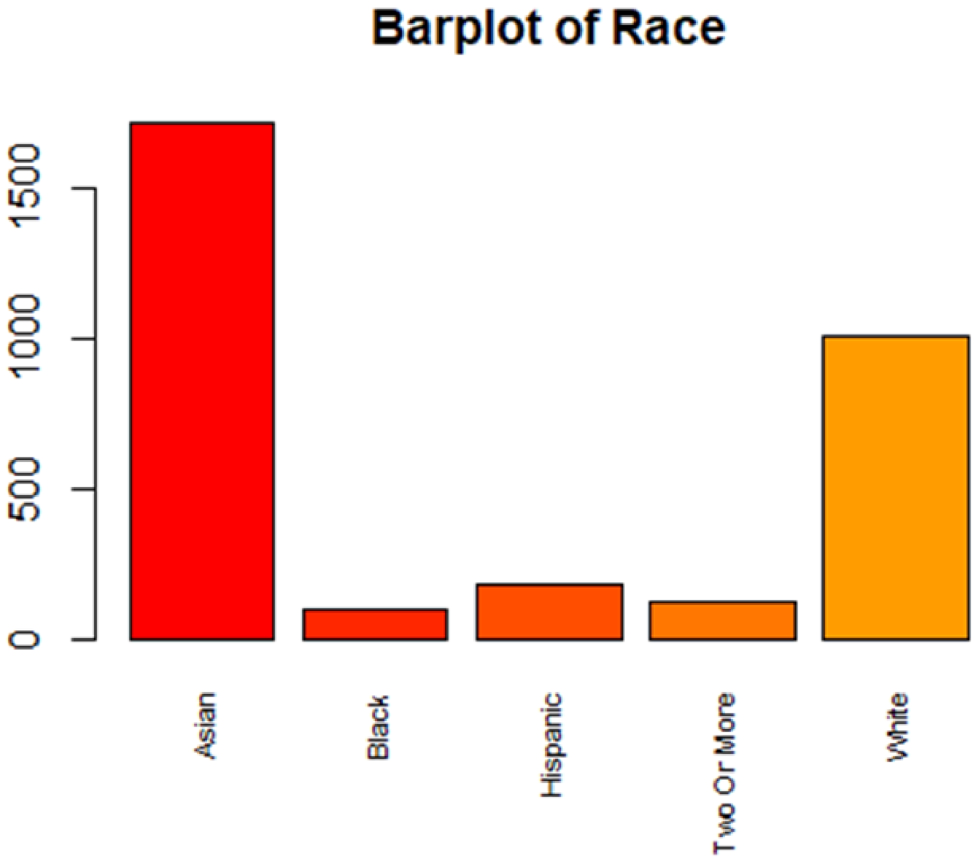
```
## [1] "Extended Summary Statistics"
##       Min. 1st Qu.  Median    Mean 3rd Qu.          Max.
##       0    1000   15000   15819   24000   59000
## [1] "sd:  13437.8354440555"
## [1] "vc:  0.84946180300859"
```


Barplot of gender (before)
2500
1500
500
0
Female
Male
Other

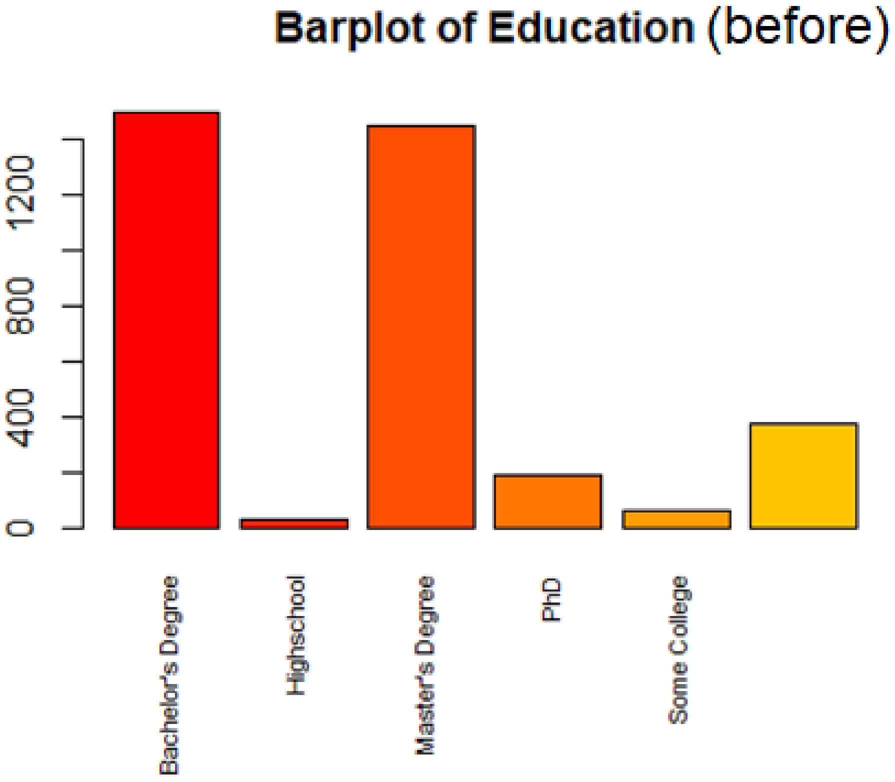

Barplot of gender
2500
1500
500
0
Female
Male
Other

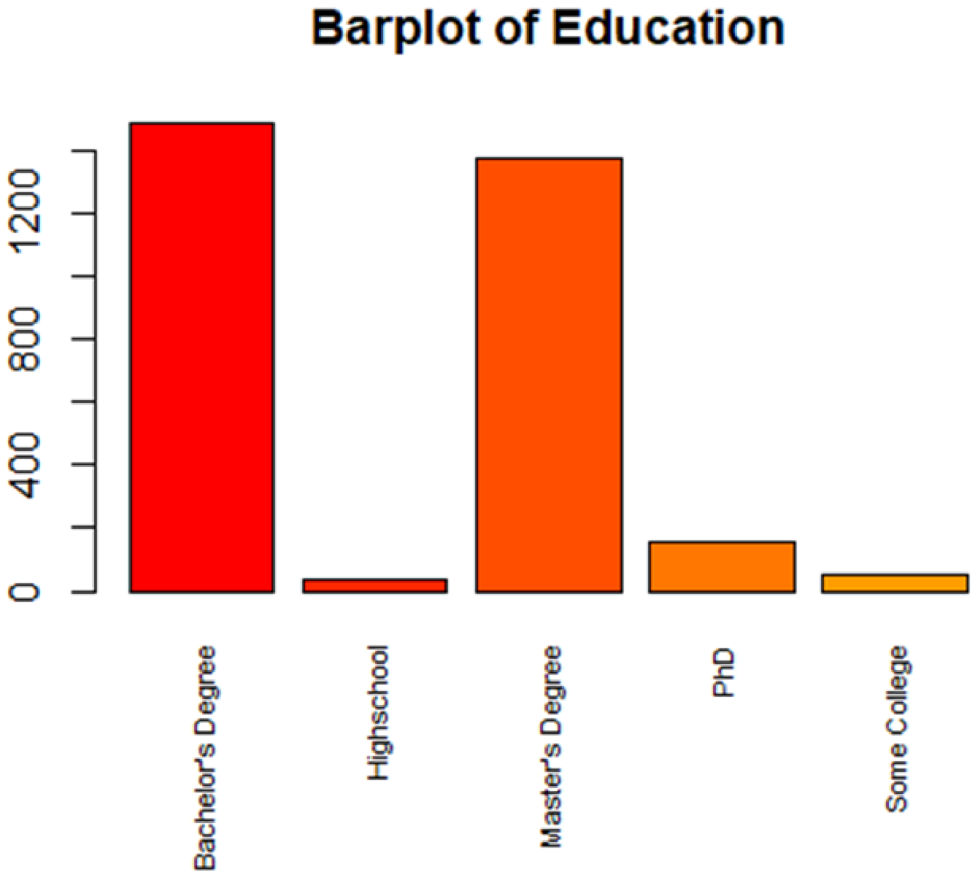
```
## [1] "Number of modalities:  3"
## [1] "Frequency table"
## Female   Male  Other
##    492   2604     13
## [1] "Relative frequency table (proportions)"
##   Female     Male    Other
## 0.158250 0.837568 0.004181
## [1] "Frequency table sorted"
##   Male Female  Other
##   2604    492     13
## [1] "Relative frequency table (proportions) sorted"
##     Male   Female    Other
## 0.837568 0.158250 0.004181
```





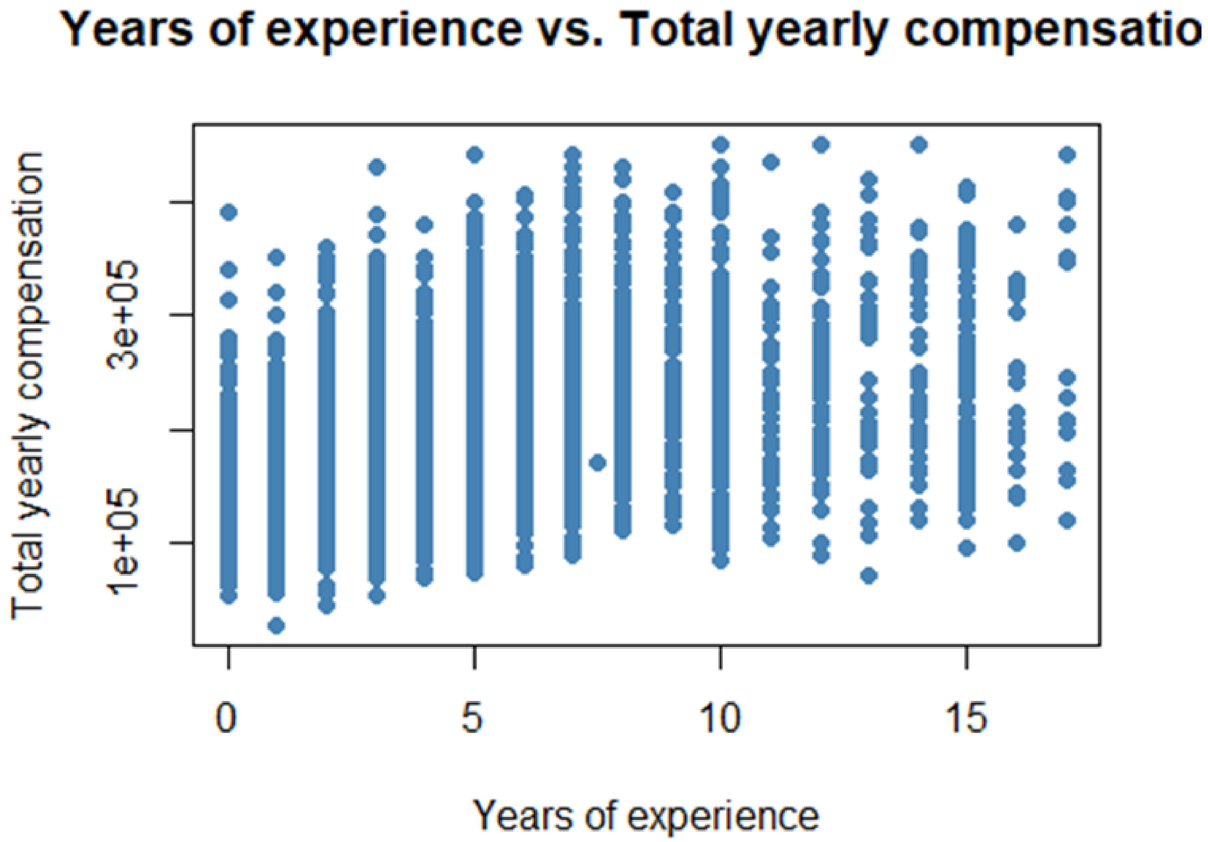
```
## [1] "Number of modalities:  5"
## [1] "Frequency table"
##      Asian          Black    Hispanic Two Or More       White
##      1712           95       177            119     1006
## [1] "Relative frequency table (proportions)"
##      Asian          Black    Hispanic Two Or More       White
##      0.550659   0.030556 0.056931       0.038276   0.323577
## [1] "Frequency table sorted"
##     Asian   White   Hispanic Two Or More       Black
##      1712           1006          177     119            95
## [1] "Relative frequency table (proportions) sorted"
##      Asian          White    Hispanic Two Or More       Black
##      0.550659   0.323577 0.056931       0.038276   0.030556
```

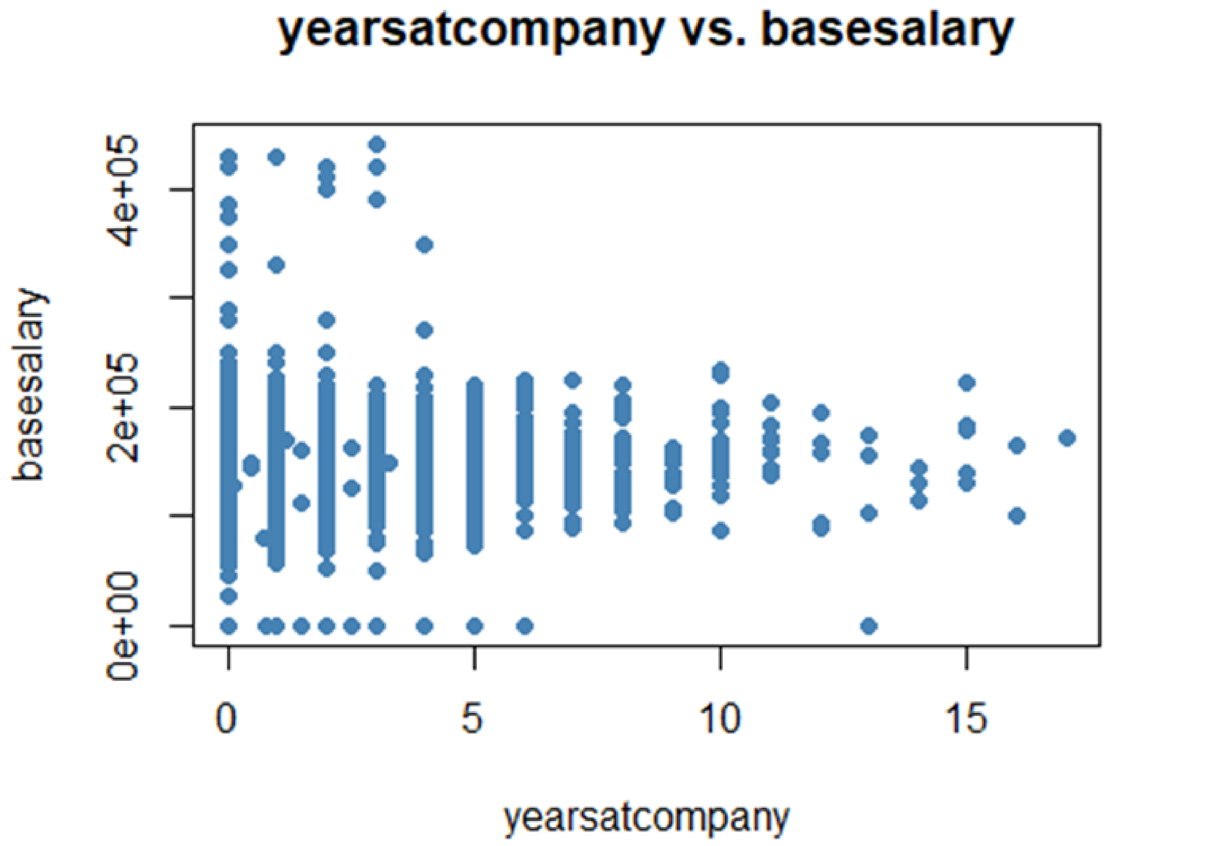

Barplot of Education (before)
0
400
800
1200
Bachelor's Degree
Highschool
Master's Degree
PhD
Some College

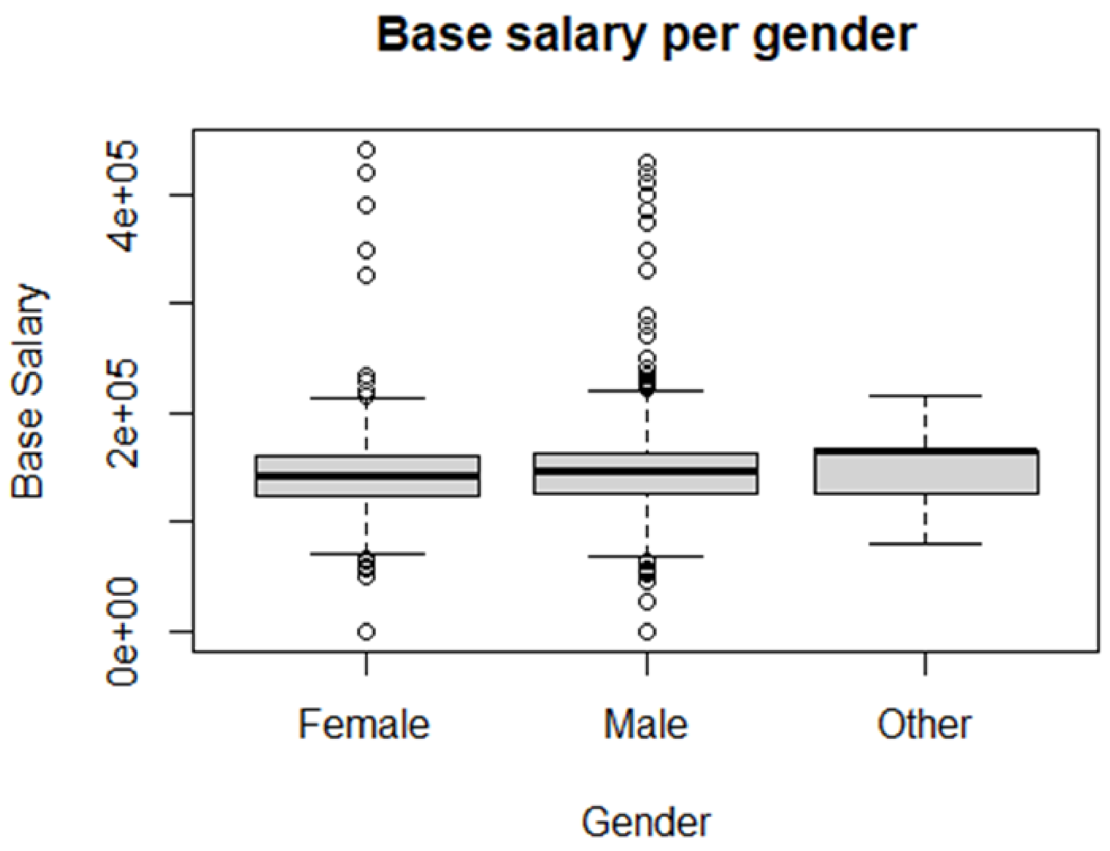

Barplot of Education
0
400
800
1200
Bachelor's Degree
Highschool
Master's Degree
PhD
Some College

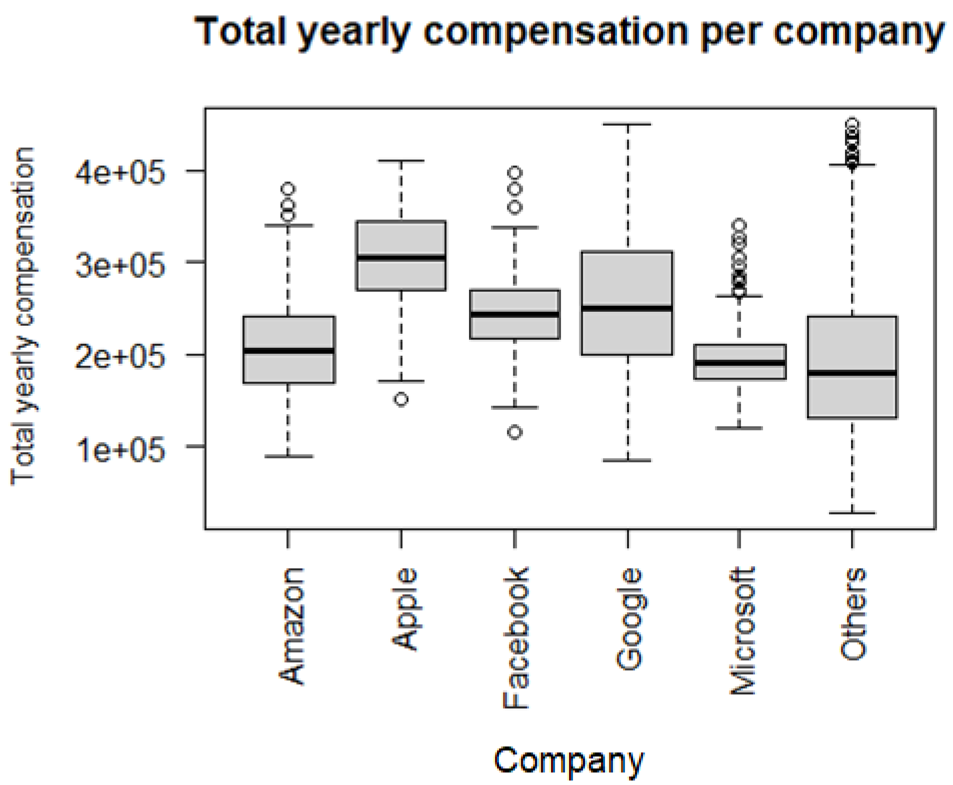
```
## [1] "Number of modalities:  5"
## [1] "Frequency table"
## Bachelor's Degree        Highschool   Master's Degree               PhD 
##              1489                33              1379               157 
##      Some College 
##                51 
## [1] "Relative frequency table (proportions)"
## Bachelor's Degree        Highschool   Master's Degree               PhD 
##          0.478932          0.010614          0.443551          0.050499 
##      Some College 
##          0.016404 
## [1] "Frequency table sorted"
## Bachelor's Degree   Master's Degree               PhD      Some College 
##              1489              1379               157                51 
##        Highschool 
##                33 
## [1] "Relative frequency table (proportions) sorted"
## Bachelor's Degree   Master's Degree               PhD      Some College 
##          0.478932          0.443551          0.050499          0.016404 
##        Highschool 
##          0.010614 
```

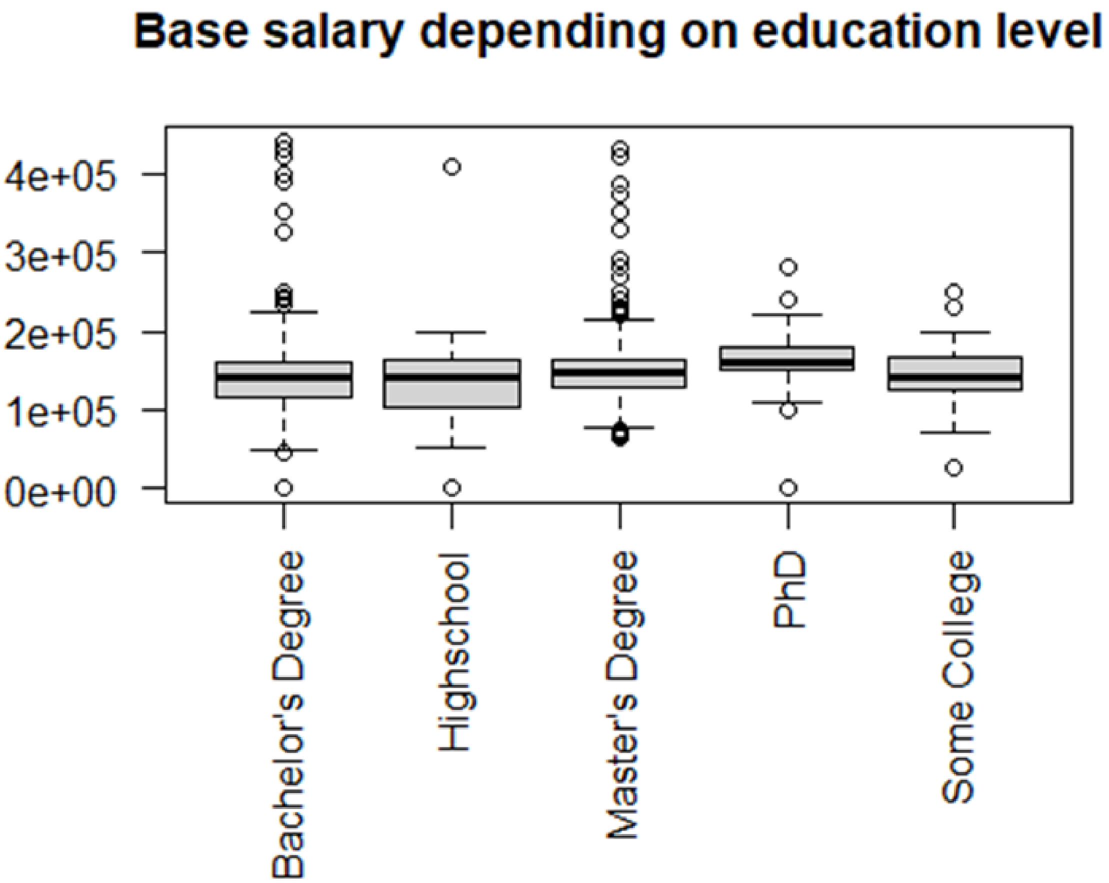
Bivariate Descriptive

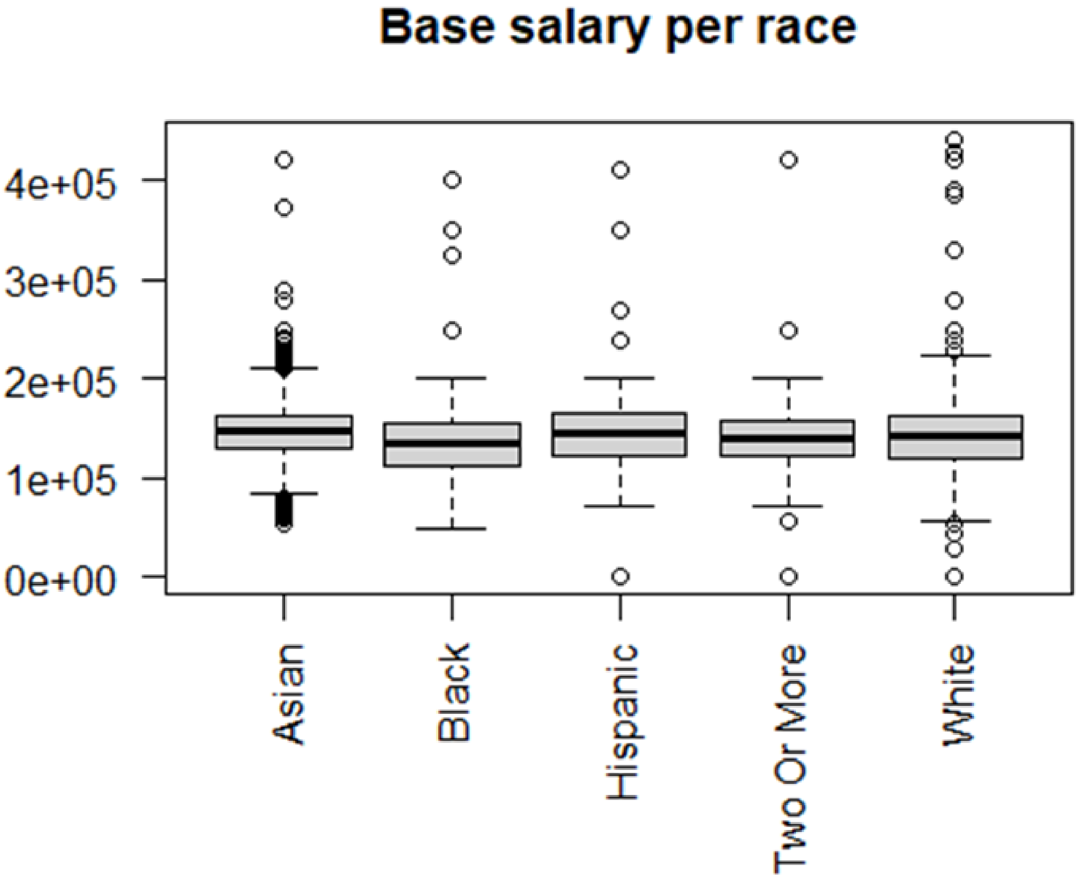

Years of experience vs. Total yearly compensatio


yearsatcompany vs. basesalary


Base salary per gender
Base Salary
4e+05
2e+05
0e+00
Female
Male
Other
Gender


Total yearly compensation per company
Total yearly compensation
4e+05
3e+05
2e+05
1e+05
Amazon
Apple
Facebook
Google
Microsoft
Others
Company


Base salary depending on education level
4e+05
3e+05
2e+05
1e+05
0e+00
Bachelor's Degree
Highschool
Master's Degree
PhD
Some College


Base salary per race
4e+05
3e+05
2e+05
1e+05
0e+00
Asian
Black
Hispanic
Two Or More
White

Conclusion

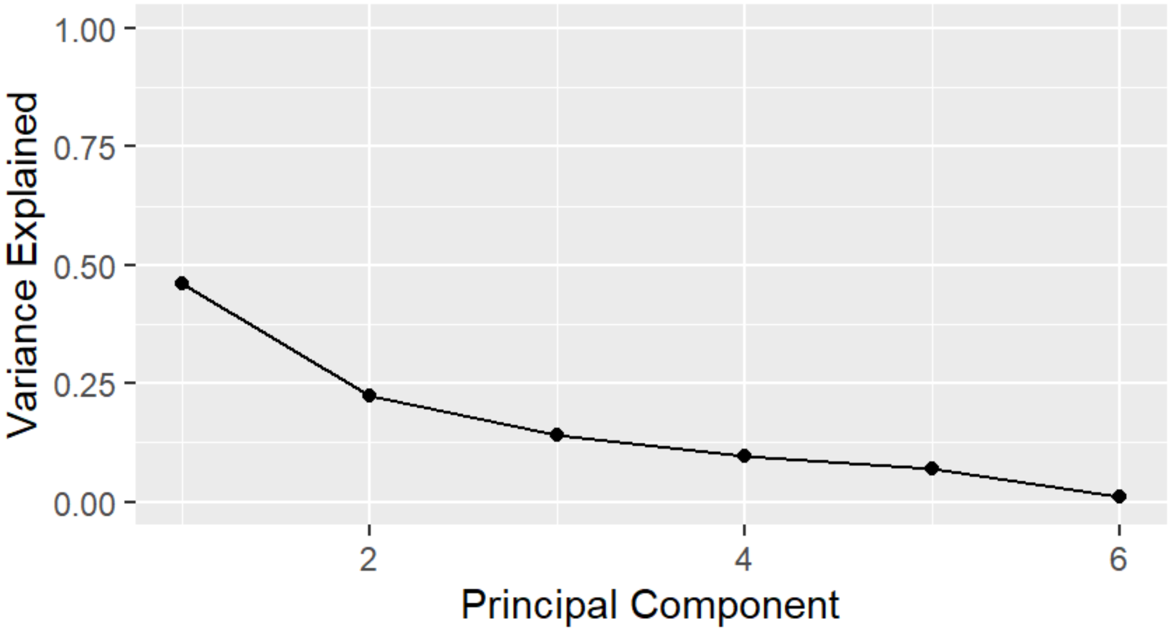
Thanks to the descriptive analysis of our data, we noticed some important facts:

- Our dataset has a minority of people working for Apple and Facebook
- The main levels represented in our data are the L5 and L6
- We can see that there's a BIG majority of software engineers
- The mean total yearly compensation is around 200k $
- The most represented region is the fourth one
- Senior workers are really rare
- People usually don't stay in the same company for a long period
- Full-stack and back-end devs are a majority
- The base salary is around 150k $ without any bonuses, etc…
- The females are a minority
- White and asian people are a majority in those fields
- Most of people have at least a Bachelor Degree
- We can observe that the total year compensation is more or less proportional to the number of years of experience, which is logic
- The more people stay in a company, the higher the base salary is
- The male base salary is slightly higher than the female one but it isn't really significant as the difference is really negligible
- The best paying companies are Google and Apple
- The base salary becomes higher only from the Master degree
- The base salaries are more or less the same for all the races, although it seems to be a bit higher for white, hispanic and asian people

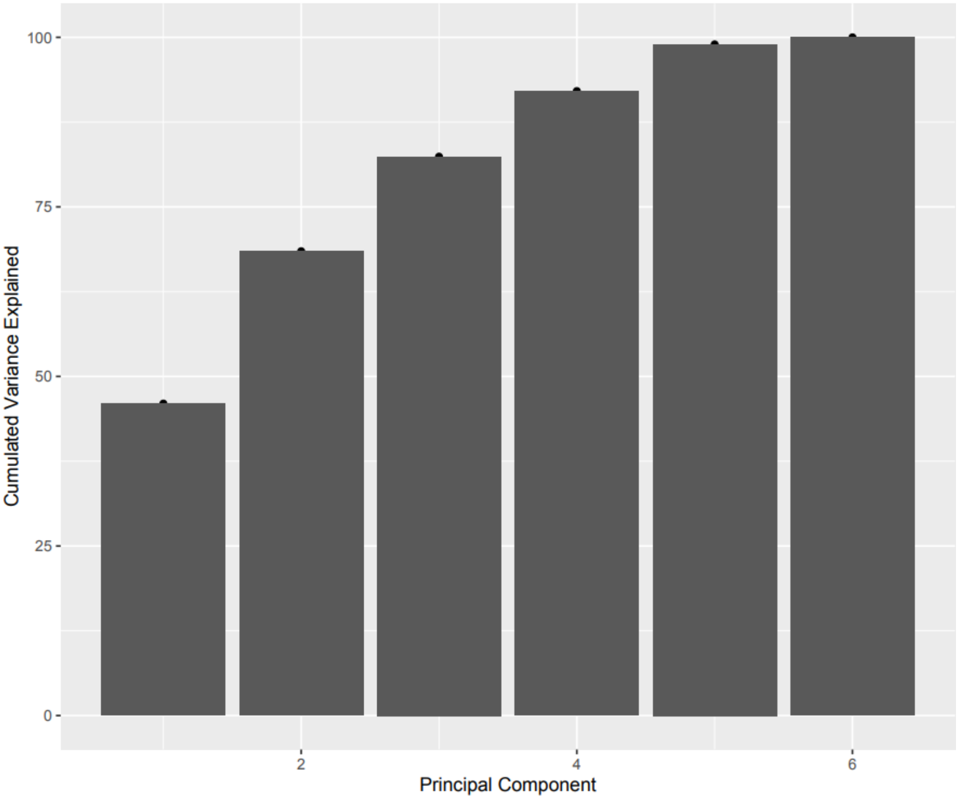
We still have to keep in mind that the data has been pre-processed to make the further experiments easier and that the descriptive analysis is done only on a certain sample of the original dataset (which has 62000 rows). That's why it can not be 100% representative of the real-world situation.

PCA analysis for numerical variables

Basic Scree Plot:

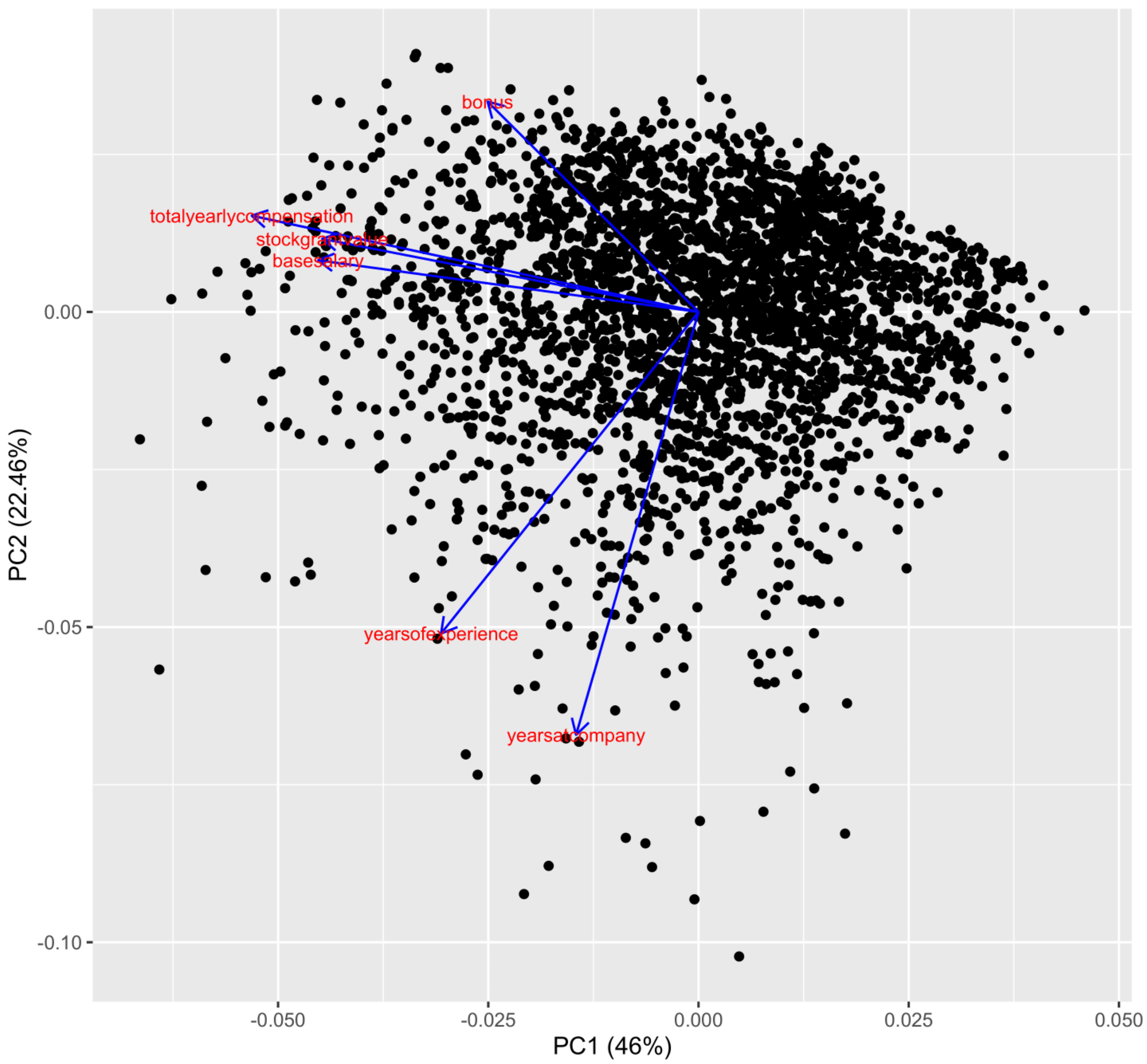


Cumulative Variance Plot:



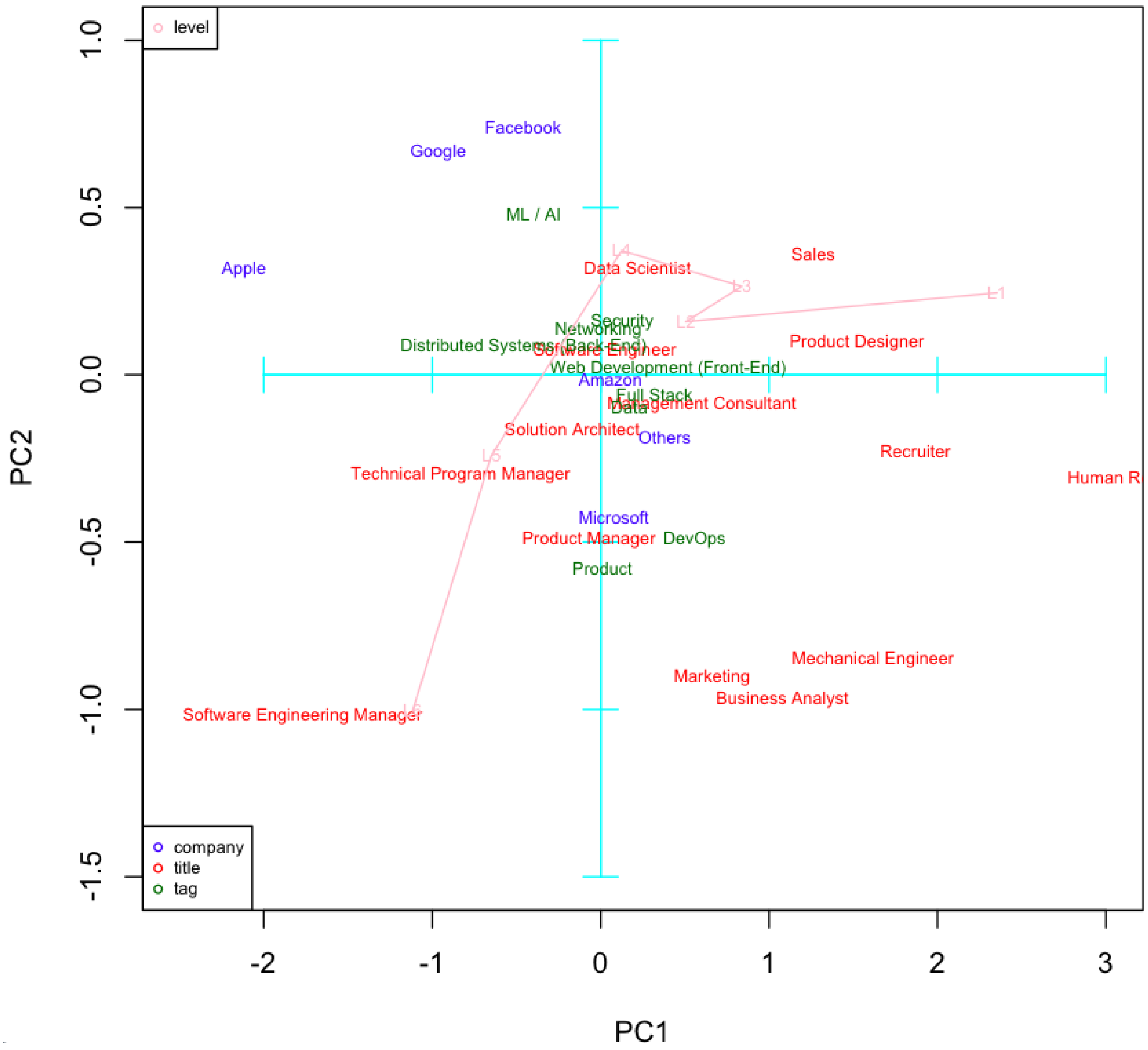
3 components are chosen as it is enough to reach the 80% of the variance explained.

Factorial map visualization:



In this projection of the variables we can see how most of the information is given by the first principal component, especially true for the variables “totalyearlycompensation”, “stockgrantvalue” and “basesalary”, which are closely related to each other.
The second principal component gives more information about the variables “yearsofexperience” and “yearsatcompany”, which are related, and on the other direction for “bonus”.

Projection of 4 of the qualitative variables

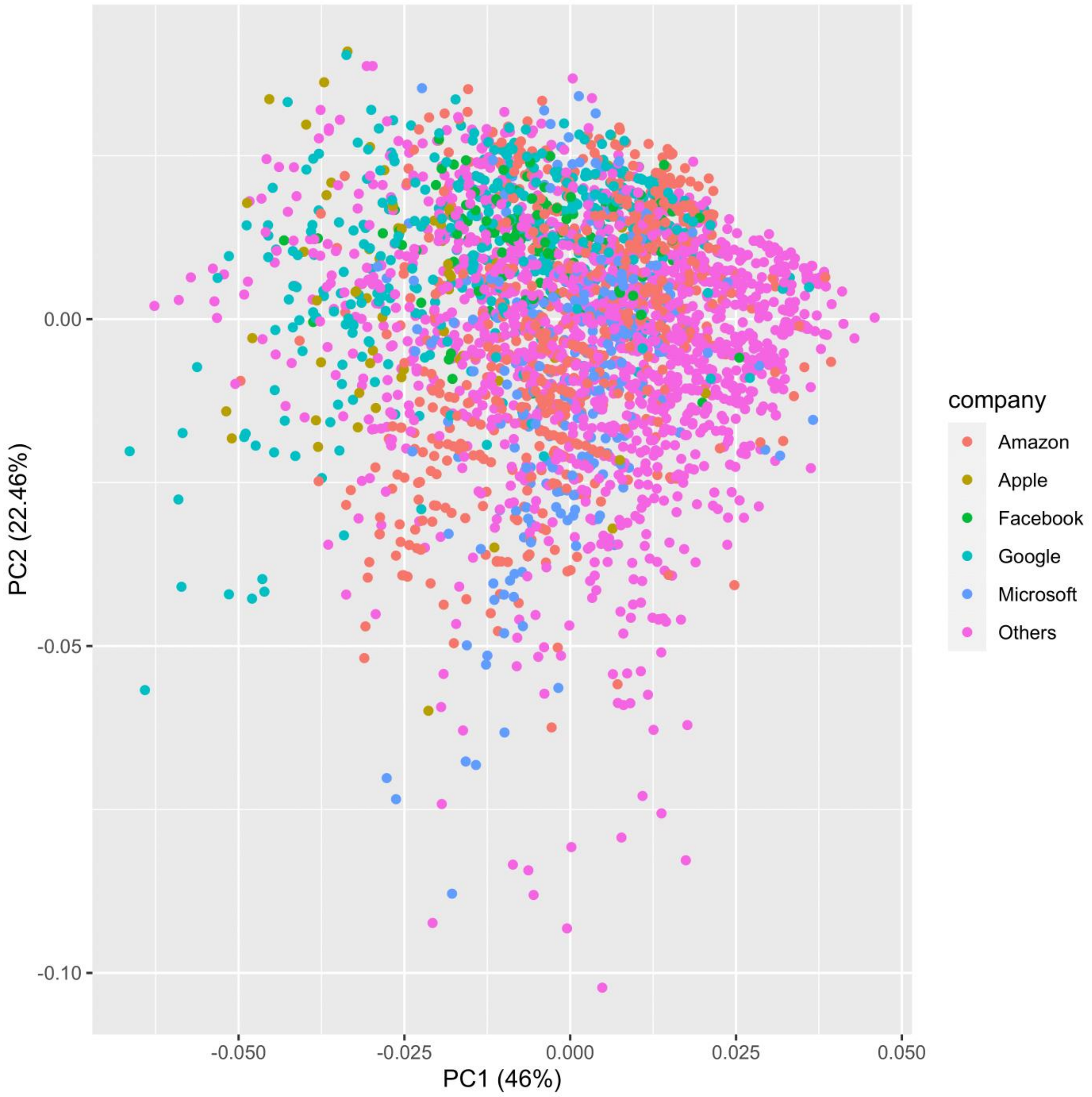


From this graph we can get a lot of information from the different qualitative variables. For starters we can see how Apple is the best paying company, followed by Google and Facebook. These last two also happen to employ the workers with the least amount of experience. Amazon, Microsoft and other companies don’t pay as much and their employees tend to stay there for more time and accumulate more years of experience, especially for Microsoft.

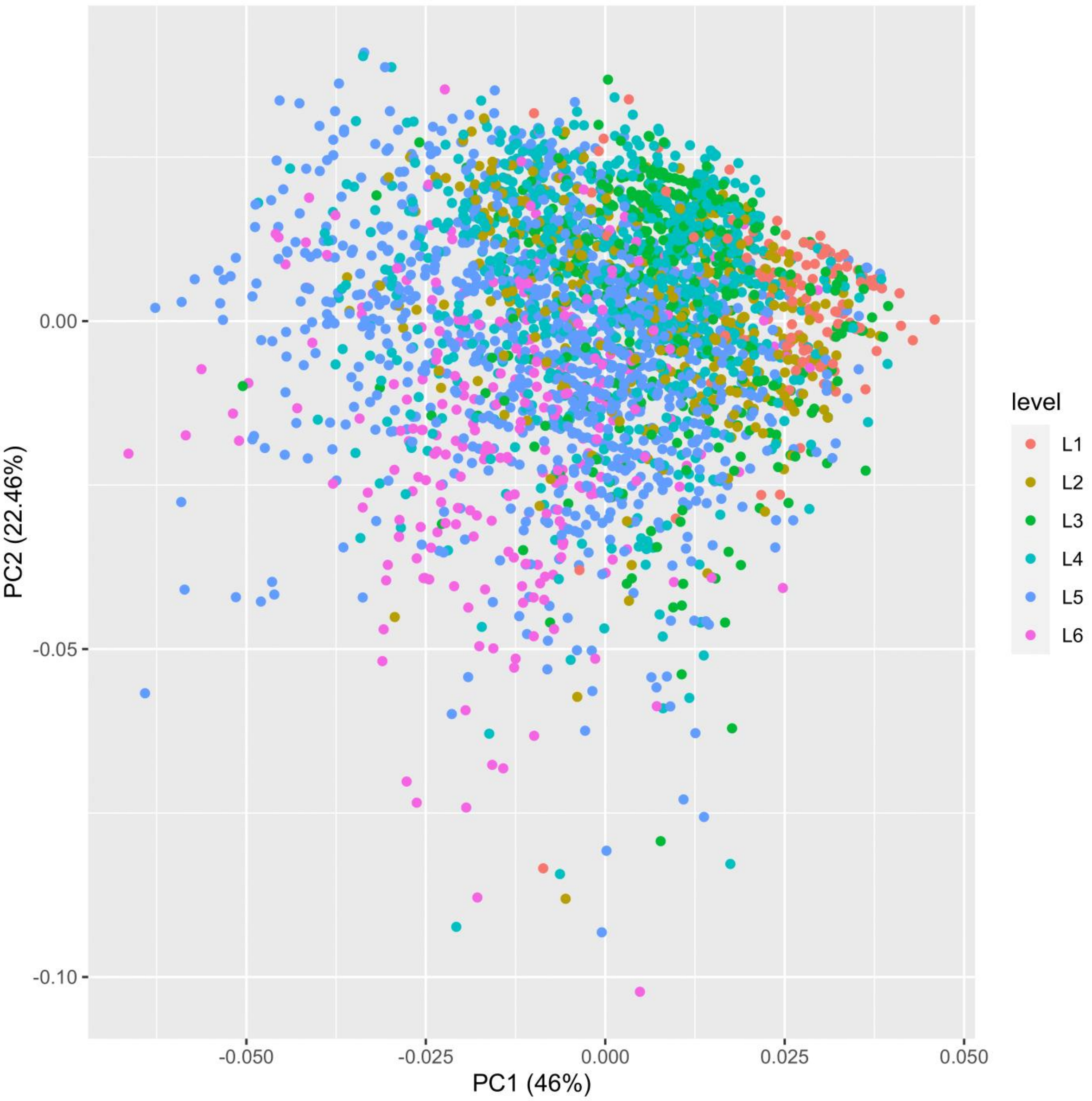
For the different kind of jobs, we can see how non-programming related jobs such as Human Resources, Recruiter… have much lower salaries and don’t involve long experience times, Software Engineers get paid more but is still within the area of average salary and low experience, while technical Managers such as Technical Program Manager and Software Engineering Manager involve more experience and come with a bigger salary. Marketing and Business Analyst workers get long years of experience but don’t come close to the level of salaries technical managers get. Software engineers get paid and are around the same level of experience regardless of the kind of work they do, having people in charge of Product or DevOps a bit more associated with longer experience times.

We can also see a clear progression in terms of the level the workers are at. The lowest levels are associated with jobs that don't involve programming, are taken by new people and people that don't last too long in the company, and as the level gets higher we get more technical jobs, that last longer, get more experience, and get paid more, going up to the job of Software Engineer Manager, which matches exactly the sixth level.

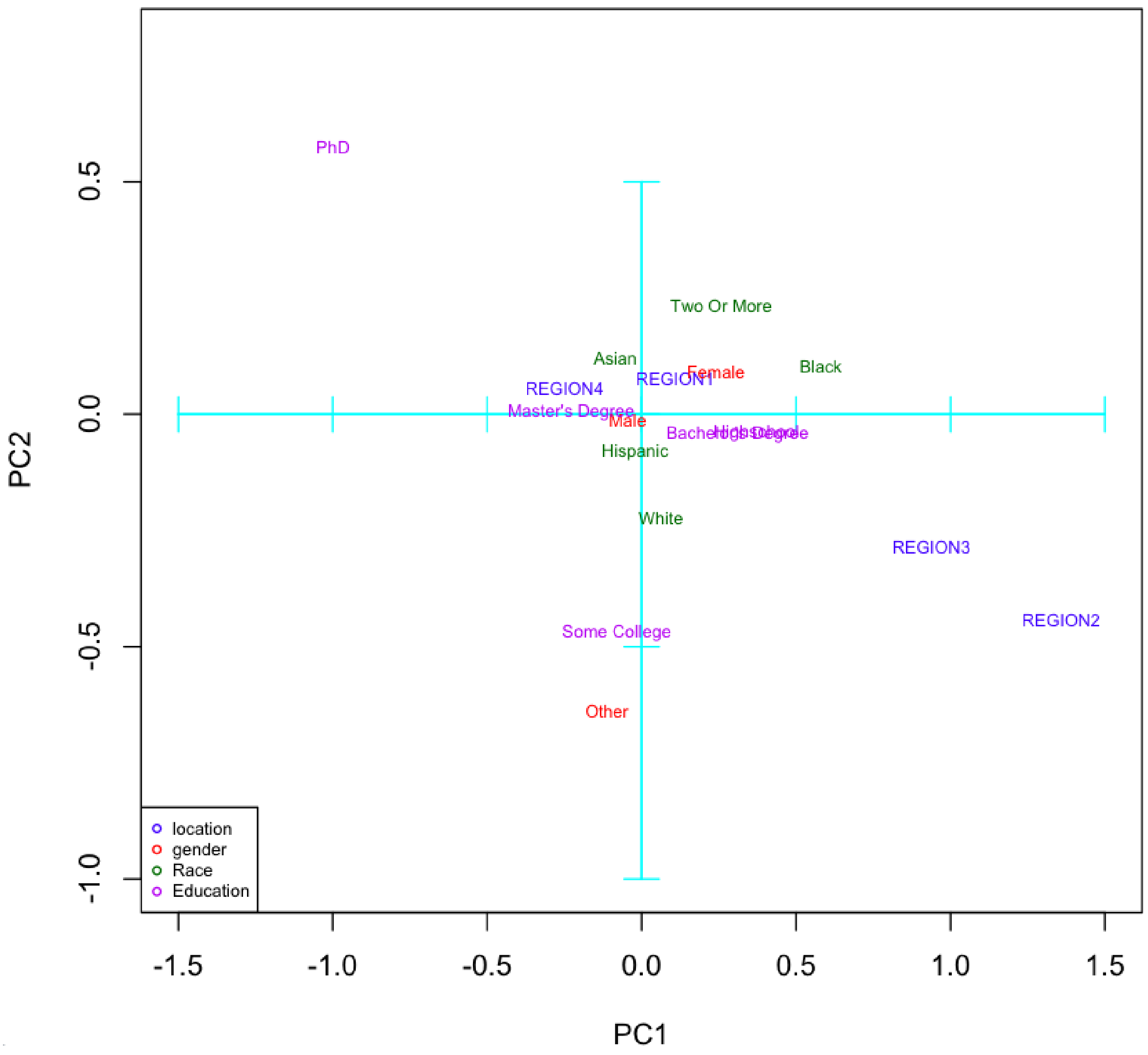
Representation of the individuals colored by "company"



Representation of the individuals colored by “level”



Projection of the remaining qualitative variables



In this second graph we don’t get as much information, but we can still make some very interesting observations.

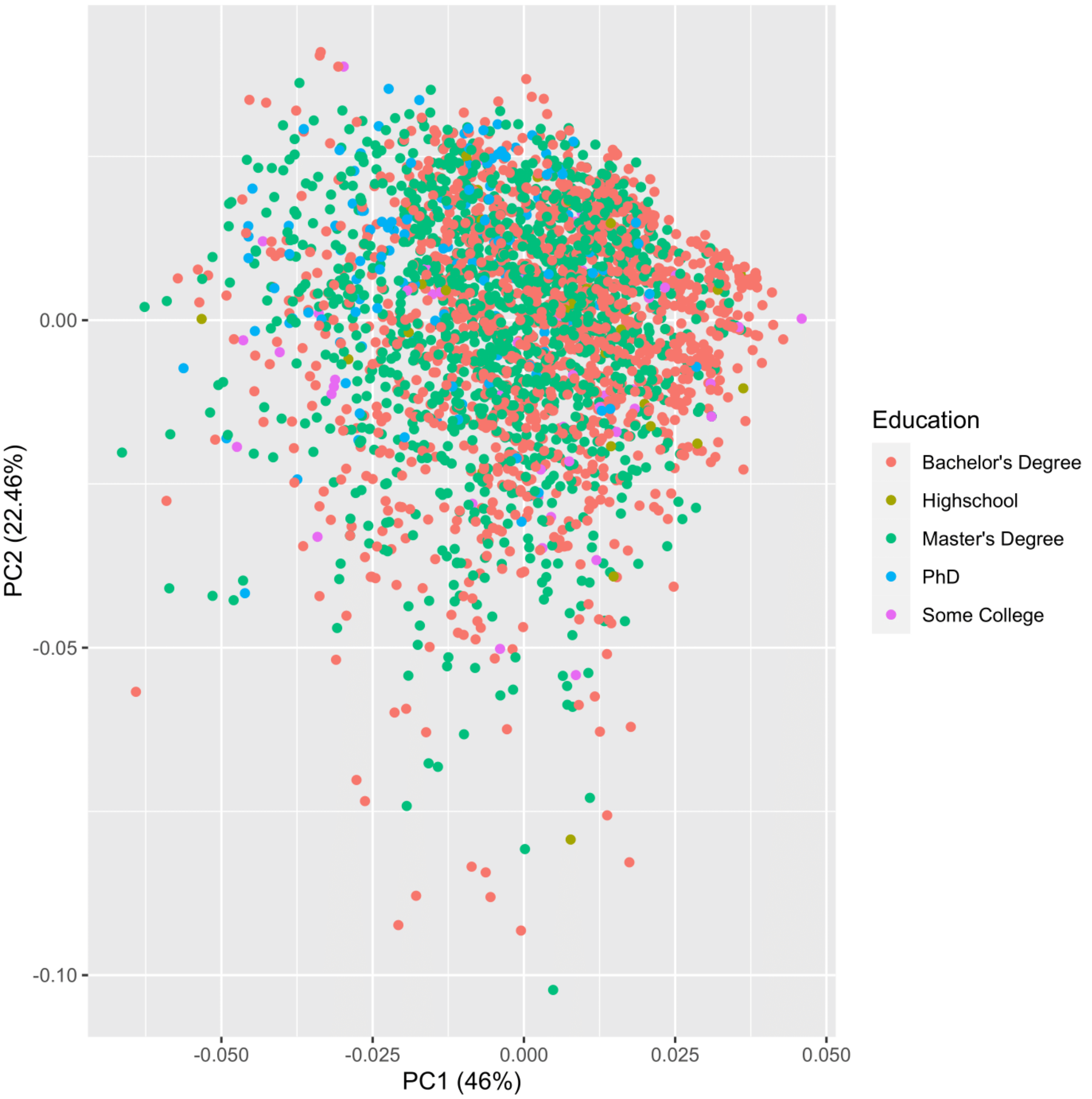
We appear to see something that wasn’t as obvious in the descriptive analysis, and it’s the fact that Female workers do tend to have worse salaries and don’t get as many years of experience as men. It’s possible that with time, if more women join the field and the ones who are working right now stay, more women will be experienced and the salaries will grow, so the gap will get smaller.

We can also see from here that there appears to be some inequality between races. While Asian, Hispanic and White people have reasonably close salaries, people with two or more races and especially black people are paid less than the rest. White people have also been in the field for more time since they have more years of experience.

In terms of education, there’s a massive difference between PhDs and the rest of education levels, since they clearly are paid the most and seem to stay for very little in companies. We can also note that older, more experienced workers don’t appear to have a clear category of studies and go for the label “Some college”.

As for the locations, there’s also a clear difference between regions. While regions 1 and 4 are within the average, regions 2 and 3 are significantly paid less, and workers there tend to have been working for longer in the field.

Representation of the individuals colored by “Education”



Hierarchical Clustering on original data

Given that our dataset is a mixed set of numerical and categorical variables, it is not possible for us to execute an algorithm like k_means, since this kind of algorithm requires all the variables to be numerical. Hence, we decided to use the PAM algorithm, which works pretty similarly to K-means or hierarchical clustering save that it uses the gower distance.
Basically, the algorithm selects a sequence of medoids (elements of the dataset which are more or less centric in terms of attributes for a specific set of rows) and then tries to reduce the dissimilarities within a single cluster by swapping elements from a group to another.
In order to normalize the numerical values and forcing them to have a similar ingerence, we have used the scale() function in r over these numerical variables.
For computing the gower distance matrix we have used the daisy function: It is worth mentioning that daisy() doesn't accept the data as we have it, but it needs the variables to be expressed as factors. Thus, we have used the as.factor() function over all columns:
Aiming to get the optimal number of clusters, the silhouette function has been used.
We found that the optimal number of clusters was 2, and so we executed the PAM algorithm with 2 clusters, discovering that our medoids are row 2733 and row 3068. That means that these rows symbolize the mean of each variable in each of the clusters:

Profiling of clusters

Our clusters are basically composed of two kinds of people, that we will define with the aid of the medoids that we have found by doing the clustering.

cluster 1:

```
2733  Others    L5 Software Engineer                  350000  REGION4 5            0 Distributed
Systems (Back-End)          0 0     0   Male White Bachelor's Degree
```

As we can see, we are talking about a white man who works as a software engineer in one of the secondary companies that we have grouped in our dataset as others. He has a bachelor’s degree and 5 years of experience. It is evident that this sort of person earns a good salary (350000 total yearly compensation), so this means that white men with bachelor degrees and some years of experience are well paid.

cluster 2:

```
initial_dataset[3068,]
3068  Others    L4 Software Engineer                  147000  REGION4 2            0 Full Stack
137000               0 10000   Male Asian Master's Degree
```

In this case we see an asian man, who works also as a worker engineer, but this time as a full stack. This man has a higher education level and despite that, his salary is less than a half of the previous example. Probably, this is due to a mix of a lesser experience and its race.

Conclusions

As this is the first year that we are brought to have a class about Data Mining, none of the members of the group had any experience with it before. This project was very interesting because of multiple reasons:

- It provided a better understanding of the different subjects that we saw during the lectures
- It allowed us to deepen our working skills with the 'R' language
- It allowed us to have a first grip on the different tools available nowadays to practice a little bit the different techniques that we saw during the lectures
- It showed us how important Data Mining is nowadays, especially with the current overload of information and why the pre-processing part is one of the most important

The whole project pushed us to do a lot of web search and that, in addition to the course material, allowed us to observe all the particularity of the different techniques presented during the lectures. We can now really understand the importance of the pre-processing part just by comparing our initial dataset with the final one. The difference is huge. And not just because the dataset is trimmed but because of the data transformation as well. A simple glance on the descriptive analysis allows us to see how much more understandable is the dataset in this delivery document compared with the previous one. This project is very valuable and will surely help us for the exam preparation or some future projects related to Machine Learning.

The rendering of the PCA doesn't allow us to specify the precise number of clusters but it seems to be a unique cluster. It's different from what we obtained during clustering using the PAM algorithm which gave us 2 clusters. Because we were missing time, we couldn't apply the k-means algorithm on the data retrieved by the PCA. That's why we can't really analyze coincidences and divergences between ACP, AMC and Clustering. It's a shame because we missed the ability to notice otherwise unseen patterns in the data points.

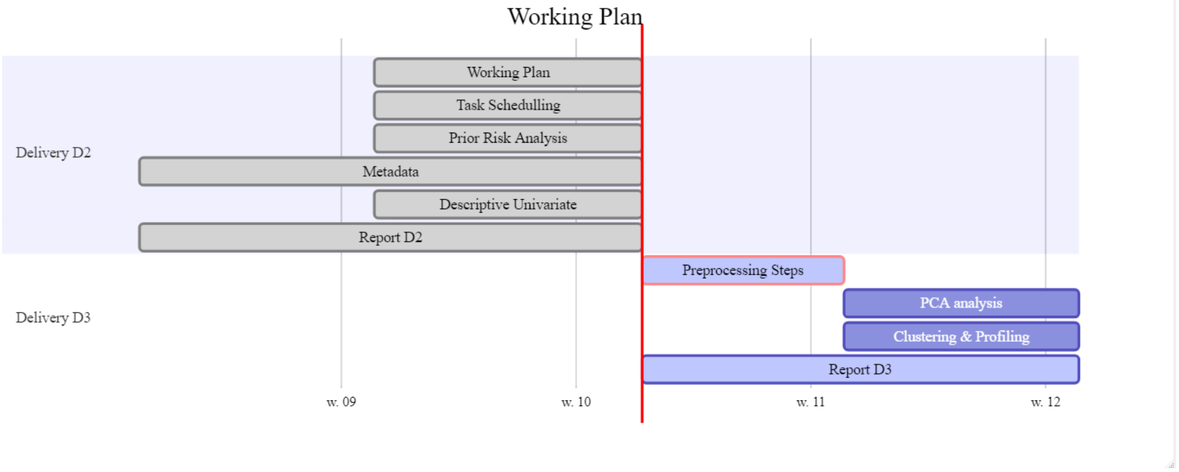
Working Plan

Gantt

First of all, we have a Gantt chart with all tasks done till this day (in gray), the tasks we will have to do next (in dark blue) and also those ones that we have already started (in light blue). Also, we have one with a red contour, being one of the most critical tasks because we cannot keep working until it's done.

The red line we see is the delivery's day and the chart finishes by the next delivery date.

For now we have the following chart, but it may be subjected to changes.



Task scheduling

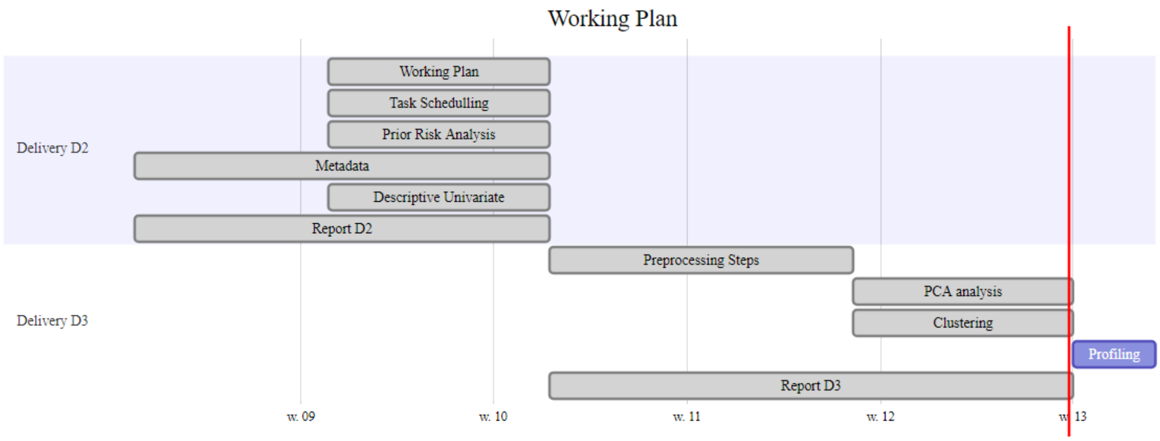
For those tasks we have done by now, we have assigned pur members like this:

- Working Plan: Everybody has taken part in organizing ourselves, but Stefan was the one to put it on the report.
- Task Scheduling: As the Working Plan, everybody has given an opinion and agreed to do concrete tasks, but Stefan has put it on the report.
- Prior Risk Analysis: Alejandro was the one that made it.
- Metadata: Being one of the most important tasks, there were two people on this task, Nuria and Juan Diego.
- Processing Steps: Nuria and Stefan will take this part.
- Descriptive Univariate: Maciej and Tomàs were the ones that took care of this task.
- PCA Analysis: Maciej and Tomi will take this part.
- Clustering & Profiling: Alejandro and Juan Diego will take this part.
- Report: Everyone has made their part and put it together.

Final scheduling

At the end we kept the same task assignment. However, we had several problems when we were doing the PCA and the Clustering that forced us to be a week of work behind because we had to redo a lot of the preprocessing and data cleaning work for a second time. So we needed the extra time that the teachers gave us to be able to finish the delivery before the deadline.

As previously mentioned, the final Gantt chart was subjected to changes, and in the following image you can see the final Gantt chart. We had to change some things, such as dividing the “Cluster & Profiling” task into two separate tasks and the “Profiling” one wasn't completely finished.



Final task assignment grid

	Work Plan	Task sched.	Risk Anal.	Meta-data	Preproc.	PCA	Clust.	Profil.	Rep.
Alejandro	✓	✓	✓				✓	✓	✓
Juan Diego	✓	✓		✓			✓	✓	✓
Nuria	✓	✓		✓	✓				✓
Stefan	✓	✓			✓				✓
Maciej	✓	✓				✓			✓
Tomas	✓	✓				✓			✓

Risk plan

Now we are going to explain which risks this project can have and how to deal with them:

1. *A member of the team is unable to perform their designed tasks due to abandonment of the course*
   - Prevention: Keep the team members motivated, work in pairs.
   - Dealing: When possible, convince the team member to comply with their responsibility. In case this doesn't work, split their tasks between the resting members of the group. If the missing members are so many that it becomes impossible to complete the tasks, turn to the professors asking for aid (people from other groups, postpone the deadline, etc)

2. *A member of the team is unable to perform their designed tasks due to illness, a disease*
   - Prevention: Work in pairs.
   - Dealing: If worked in pairs, there is a team member with the knowledge of what has been done. If not or both happen to be ill or with a disease then we redistribute the tasks.

3. *Losing access to the files*
   - Prevention: Work in the cloud as often as possible. Keep backups in external devices as USBs.
   - Dealing: Restore the missing files.

4. *The pandemic prevents us from meeting*
   - Prevention: Use the cloud or any system capable of sharing the files with all the team members without a face-to-face meeting.
   - Dealing: Use online meeting services as Discord, Google meet, Zoom...

5. *A team member doesn't comply with their part*
   - Prevention: Try to finish the tasks a couple of days before the deadline in order to have reaction time.
   - Dealing: Split the task between the other members of the group. Give a warning to the irresponsible team member and control them in the future

6. *A member of the group doesn't know how to do something*
   - Prevention: Assigning the tasks attending to each one's abilities
   - Dealing: The team members who have less work will help him find the solution. If no one knows how to deal with something and nothing is found browsing the web we will turn to the professors.