Project : Analysis of the Salary structure in Germany and prediction with regression models

Submitted by: *Shruti Sankhe, Arijit Paul, Arun Prakash Vadivelu*

Submitted on: *13-06-2017*

Introduction:

Wages are determined by number of factors like education, industry, skills, experience, etc, etc. This project tries to understand them by analyzing the wage data of around 50000 employees and tries to predict the salary based on few features. It uses various functions of R to clean, prepare and analyze the data and then tries to predict the salary and few other classes using linear regression and multi-nominal regression

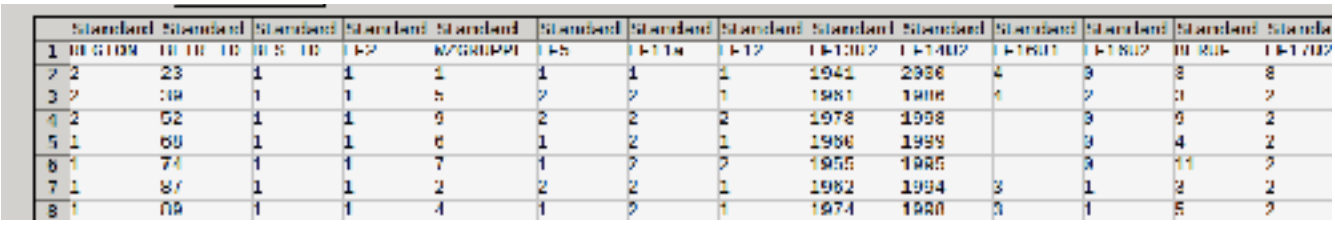
Data Source:

The data was provided by a government organization in Germany, that collects census data and allows students/ researchers to analyze and present their findings

source: https://www.destatis.de/DE/

Data format,Quality & Content:

The data was in a CSV format with its values represented by keys as shown.



1	[illegible]	[illegible]	[illegible]	[illegible]	WZGRUPPE	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
2	2	23	1	1	1	1	1	1	1941	2006	4	8	8	8
3	2	[illegible]	1	1	5	2	2	1	1961	1986	4	2	0	2
4	2	52	1	1	6	2	2	2	1978	1998		8	6	2
5	1	60	1	1	6	1	2	1	1956	1999		9	4	2
6	1	74	1	1	7	1	2	2	1955	1985		8	11	2
7	1	87	1	1	2	2	2	1	1962	1994	3	1	2	2
8	1	09	1	1	4	1	2	1	1974	1980	0	1	5	2

WZGRUPPE	Wirtschaftsgruppe, in der Haupttätigkeitsbereich des Betriebes liegt	1 = Ernährungsgewerbe und Tabakverarbeitung; Textil- und Bekleidungsgewerbe; Ledergewerbe 2 = Verarbeitendes Gewerbe ohne Ernährungsgewerbe, Tabakverarbeitung, Textil- und Bekleidungsgewerbe, Ledergewerbe 3 = Energie- und Wasserversorgung

Project plan and Execution:

Data Preparation :

- Translate columns & row values from German to English
- Understand the keys represented in the data (e.g nr. 7 in Edu. = Uni Degree)
- Remove the columns that are not relevant
- Import the CSV data in R to a data frame
- Merge & Remove few columns(e.g. female % = B_EF15+B_EF17)
- Use few arithmetic operations (e.g: age = 2001-year_of_birth,...)
- Normalize the numeric data
- Remove Outliers & NAs
- Convert Numeric to Factors

Data Analysis & Interpretation:

- find the salary distribution according to industry, skills, education
- Plot Histograms and Boxplots

Linear Regression Modeling:

- Find correlation between the features
- Split the training and test data
- prepare the model and find the fitted value
- Validate the model and rebuild the model
- Find Accuracy

Multi-Nominal and Logistic Model for Categorical Variable:

- Find correlation between the features
- Build the model
- Calculate confusion Matrix and accuracy

```
library(corrplot)
library(caret)
library(ggplot2)
library(readr)
library(nnet)
library(MASS)
```

Data Preparation

#Merge/Remove/Rename columns

Some The columns underwent merge, removal to reduce the data redundancy. Also, columns like ‘Position of the Employee’ were renamed to more precise column name like ‘Rank’.

```
german_sal_data                 <-      read.csv(filepath_w)
german_sal_data$birth_year      <-      2001-german_sal_data$birth_year

german_sal_data$job_start_year   <- 2002-german_sal_data$job_start_year
#merge:
german_sal_data$male_percent    <-
                                german_sal_data_org$B_EF14+german_sal_data_org$B_EF16
german_sal_data$female_percent  <-
                                german_sal_data_org$B_EF15+german_sal_data_org$B_EF17
#Rename:
colnames(german_sal_data)[which(names(german_sal_data) == "birth_year")] <- "Age"
colnames(german_sal_data)[which(names(german_sal_data) == "job_start_year")] <-   "yrs_of_Exp"
```

 # Check Data dimensions, columns names and Structure

```
dim(german_sal_data)
## [1] 50728    21
colnames(german_sal_data)
##  [1] "REGION"                "employee_type"       "        Industry_type"
##  [4] "monthlyORhourly"       "skill_type"          "gender"
##  [7] "Age"                   "yrs_of_Exp"          "tax_grade"
## [10] "no_of_child"           "job"                 "rank"
## [13] "education"             "employment_type"     "salary_type"
## [16] "weekly_work_hour"      "yearly_income"       "paid_holiday_days"
## [19] "number_of_employee"    "male_percent"        "female_percent"
str(german_sal_data)
```

```
## 'data.frame':    43421 obs. of  21 variables:
##  $ REGION                : int  2 2 2 1 1 1 1 1 2 1 ...
##  $ employee_type         : int  1 1 1 1 1 1 2 1 1 1 ...
##  $ Industry_type         : int  1 5 9 6 7 2 7 2 5 2 ...
##  $ monthlyORhourly       : int  1 2 2 1 1 2 NA 1 2 1 ...
##  $ skill_type            : int  1 2 2 2 2 2 8 3 2 2 ...
##  $ gender                : int  1 1 2 1 2 1 1 1 1 1 ...
##  $ Age                   : int  60 40 23 41 46 39 22 34 30 40 ...
##  $ yrs_of_Exp            : int  2 16 4 3 7 8 2 1 9 16 ...
##  $ tax_grade             : int  4 4 NA NA NA 3 NA NA 3 3 ...
##  $ no_of_child           : num  0 2 0 0 0 1 0 0 2 0 ...
##  $ job                   : int  3 3 9 4 11 3 10 2 3 2 ...
##  $ rank                  : int  8 2 2 2 2 2 4 8 2 2 ...
##  $ education             : int  2 2 2 2 2 2 2 7 2 2 ...
##  $ employment_type       : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ salary_type           : int  1 1 NA NA NA 1 NA 1 5 3 ...
##  $ weekly_work_hour      : num  7 39 40 39 38.5 35 39 10 40 35 ...
##  $ yearly_income         : int  3150 26131 8434 22190 27515 31058 22219  32431 ...
##  $ paid_holiday_days     : int  NA 27 23 26 30 30 29 30 20 30 ...
##  $ number_of_employee    : int  1 4 56 3 4 4 2 2 2 4 ...
##  $ male_percent          : num  33.4 91.2 64.3 54.3 41.6 87.6 75.7 41.2 78.3 89.2 ...
##  $ female_percent        : num  66.6 8.8 35.7 45.7 58.4 12.4 24.3 58.8 21.7 10.8 ...
```

 # Normalize the Numeric Data :

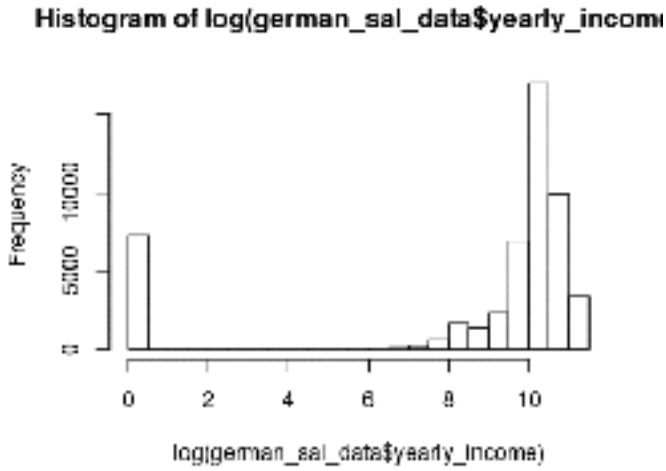
There are many ways to normalize the data.

- Min-Max Normalization
- Z-Square Normalization
- Log transform etc..

But since most of the variables were categorical there was no necessity to normalize the data.

Salary, Age, Experience years were few numeric data that needed some normalization.

```
hist(german_sal_data$yearly_income)
```



hist(**log**(german_sal_data$yearly_income))

the log transformation did not help and we retained the data as is.

Remove outliers and NAs

Since the salary below 16,000 Euros (10th percentile) are earned by the students, and doesn't contribute to the analysis and prediction, they are removed from the consideration.

Also since the columns like tax grade, Salary type, paid holidays and monthlyORhourly do not contribute to the Salary, we have removed that from the data frame

```
#removing the lower 10th percentile IQR :
german_sal_data <- subset(german_sal_data,yearly_income >
quantile(german_sal_data$yearly_income, c(.10)))

#removing the NAs:
colnames(german_sal_data)[ apply(german_sal_data, 2, anyNA) ]
## "monthlyORhourly"   "skill_type"    "tax_grade"    "salary_type"   “paid_holiday_days"
german_sal_data <- subset(german_sal_data, select = -
c(monthlyORhourly,tax_grade,salary_type,skill_type,paid_holiday_days))

dim(german_sal_data)
## [1] 43421    16
```

Convert categorical variables to Factors

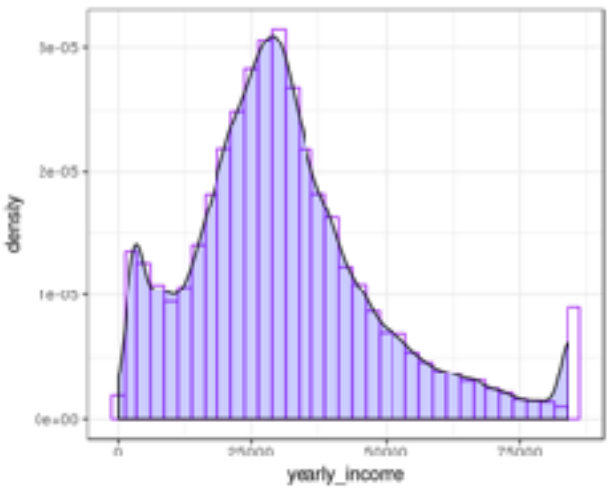
The integers values for the variables like Industry_type, Education, Region actually are keys that refer to the respective values in another table. So considering them as a continuous numeric/ integer doesn't make sense and needed to be converted to factors

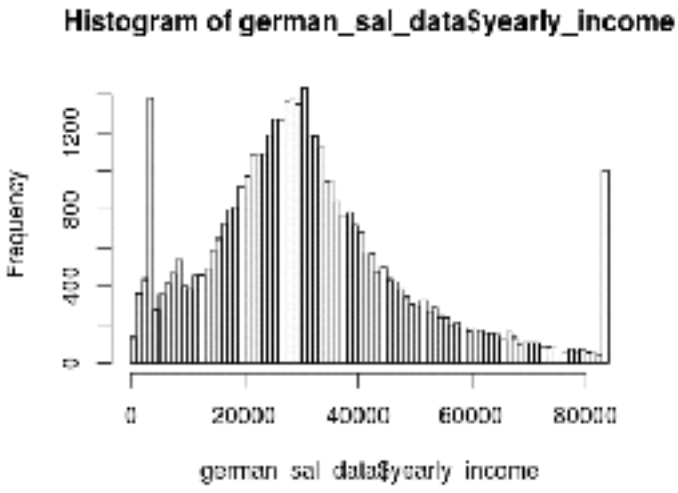
```
numericCol <- german_sal_data[ ,(colnames(german_sal_data) %in%
c("Age","yearly_income","male_percent","weekly_work_hour","female_percent","yrs_of_Exp"))]
factorcCol <-  german_sal_data[ ,!(colnames(german_sal_data) %in%
c("Age","yearly_income","male_percent","weekly_work_hour","female_percent","yrs_of_Exp"))] %>%
lapply(factor) %>% data.frame()
german_sal_data <-cbind(numericCol, factorcCol)

str(german_sal_data)
##  $ Age              : int  60 40 23 41 46 39 22 34 30 40 ...
##  $ yrs_of_Exp               : int  2 16 4 3 7 8 2 1 9 16 ...
##  $ weekly_work_hour  : num  7 39 40 39 38.5 35 39 10 40 35 ...
##  $ REGION                   : Factor w/ 2 levels "1","2": 2 2 2 1 1 1 1 1 2 1 ...
##  $ employee_type            : Factor w/ 2 levels "1","2": 1 1 1 1 1 1 2 1 1 1 ...
##  $ Industry_type            : Factor w/ 9 levels "1","2","3","4",..: 1 5 9 6 7 2 7 2 5 2 ...
##  $ gender           : Factor w/ 2 levels "1","2": 1 1 2 1 2 1 1 1 1 1 ...
##  $ no_of_child      : Factor w/ 9 levels "0","0.5","1",..: 1 5 1 1 1 3 1 1 5 1 …
…..
```

Data Analysis and Interpretation

Now that the data was clean we group that and sort to find the distribution of salary and their relationship with education, skill type, industry etc..

```
hist(german_sal_data$yearly_income, breaks = 100)
ggplot(german_sal_data, aes(x=yearly_income)) + geom_histogram(aes(y=..density..),binwidth =
2500,colour="purple", fill="white") + theme_bw()+geom_density(alpha=.2, fill="blue")
plot(sort(german_sal_data$yearly_income), main = "Sorted Base Pay")
```





the above distribution shows that the mean salary distribution is around 30000 Euros. Since the salaries above 80,000 are considered to be equal to 80,000 we see that there is a peak at the end.

Analyze how skills, education and Age are influencing the salary:

Top 5 salaries by industry:

```
finding1 <- german_sal_data %>% group_by(Industry_type) %>%
summarize(MEANSAL=round(mean(yearly_income),0),MEANEDU=round(mean(education) ,
0),MEANMALEPERCENT=round(mean(male_percent) ,
0),MEANFEMALEPERCENT=round(mean(female_percent) ,0)) %>% arrange(desc(MEANSAL)) %>%
head(n=5)
```

Top 5 salaries by skill type:

```
finding2 <- german_sal_data %>% group_by(skill_type) %>%
summarize(MEANSAL=round(mean(yearly_income),0),MEANAGE=round(mean(Age) ,
0),MEANEXPERIENCE=round(mean(yrs_of_Exp) ,
0),MEANWORKHR=round(mean(weekly_work_hour) ,0)) %>% arrange(desc(MEANSAL)) %>%
head(n=5)
```

Top 5 salaries by Education:

```
finding3 <- german_sal_data %>% group_by(education) %>%
summarize(MEANSAL=round(mean(yearly_income),0)) %>% arrange(desc(MEANSAL))
```

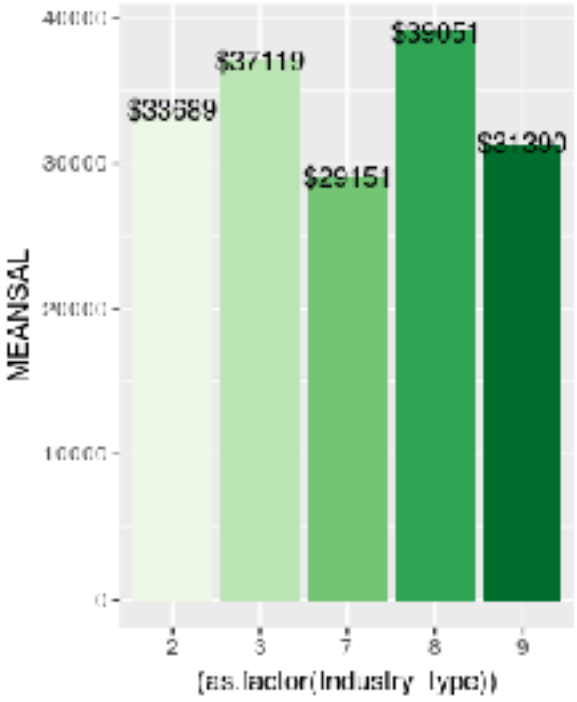
Salaries by Age:

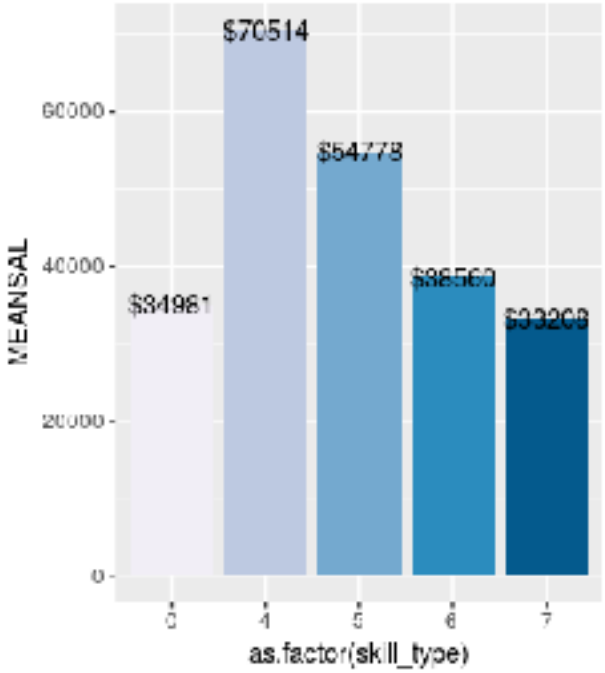
```
Age_bin <-  seq(15, 55, by=5)
Salary <- round(tapply(as.numeric(german_sal_data$yearly_income), cut(german_sal_data$Age,
seq(15, 60, by=5)), mean),0)
newdf <- data.frame(Age_bin,Salary )
```

Plot the above findings

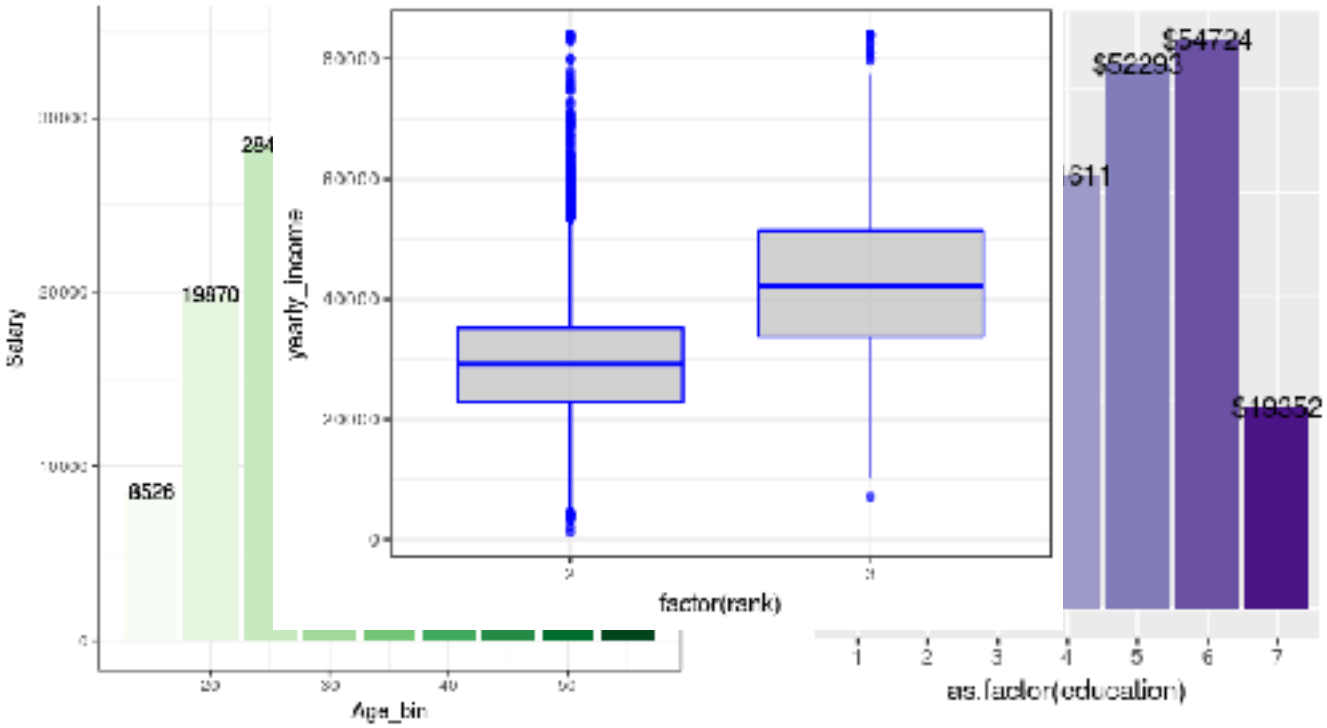
```
ggplot(data=finding1, aes(x=(as.factor(Industry_type)), y=MEANSAL, fill=factor(Industry_type))) +
geom_bar(stat="identity") +   geom_text(aes(x=as.factor(Industry_type), y=MEANSAL,
label=paste0("$",MEANSAL))) +  scale_fill_brewer(palette = "Green")

ggplot(data=finding2, aes(x=as.factor(skill_type), y=MEANSAL, fill=factor(skill_type))) +
```

```
geom_bar(stat="identity") +   geom_text(aes(x=as.factor(skill_type), y=MEANSAL,
label=paste0("$",MEANSAL))) + scale_fill_brewer(palette = "PuBu")

ggplot(data=finding3, aes(x=as.factor(education), y=MEANSAL, fill=factor(education))) +
geom_bar(stat="identity") +   geom_text(aes(x=as.factor(education), y=MEANSAL,
label=paste0("$",MEANSAL))) +   scale_fill_brewer(palette = "Purples")

ggplot(data=newdf, aes(x=Age_bin, y=Salary, fill=factor(Age_bin))) +geom_bar(stat="identity") +
  geom_text(aes(x=Age_bin, y=Salary, label=Salary)) +theme_bw() + geom_text(aes(label = Salary),
position = position_dodge(0.6)) + scale_fill_brewer(palette = "green")
```

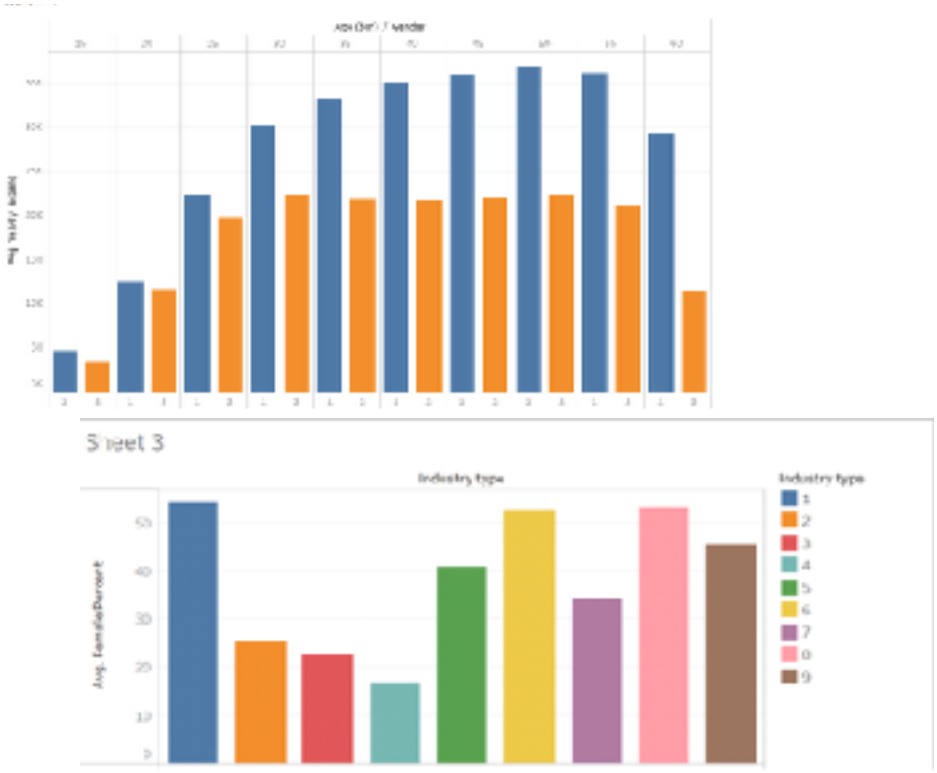





```
### Compare Salary range of a Technician Vs
Supervisorggplot(german_sal_data_tmp,aes(x=factor(rank),y=yearly_income)) +geom_boxplot(fill =
"grey", colour = "blue", alpha = 0.7)+ theme_bw()
```



Salary distribution by Gender & % of female by industries



The above graph shows the distribution of salary by age and distinguish

There is a significant difference in the earnings between male and .male and female earnings The below chart represents favorable industry as chosen by average female workers .female yellow ,The blue .where the industry types is plotted against the average female percentage banking and hospitality industries respectively ,and pink bars represent retail

From the all the histograms and analysis, following contributes to the top earnings:

Education	Industries	Skill Type
Technical Universtiy Degree	Banking & Insurance	Management
Universities of Applied sciences	Engineering	Technical & Highly Skilled

Linear Regression Modeling:

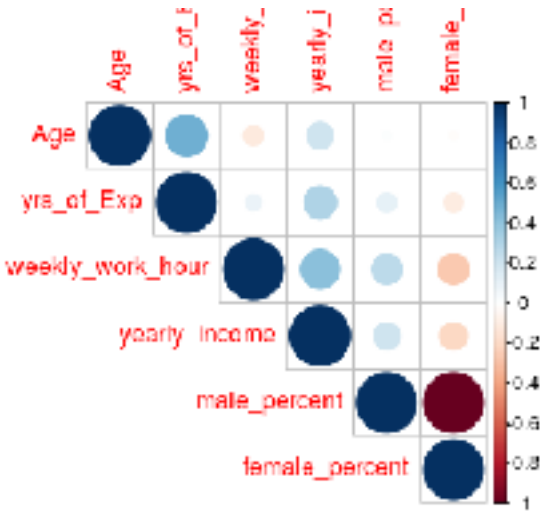
Finding Correlation for the Numeric Columns

for the numeric variables, a correlation between them was evaluated and found that male percent was dependent on the female percent (male%=100-female%)

cor(numericCol)

```
##                    Age                 yrs_of_Exp     weekly_work_hr yearly_income
## Age                1.00000000  0.48821923  -0.11816163   0.2019092
## yrs_of_Exp          0.48821923  1.00000000   0.08148372   0.3029238
## weekly_work_hour   -0.11816163  0.08148372   1.00000000   0.4155431
## yearly_income      0.20190920  0.30292381   0.41554305   1.0000000
## male_percent       0.01758176  0.10454913   0.26666819   0.2085824
## female_percent     -0.01757995 -0.10455128  -0.26667377  -0.2085882
##                    male_percent female_percent
## Age                0.01758176  -0.01757995
## yrs_of_Exp         0.10454913  -0.10455128
## weekly_work_hour   0.26666819  -0.26667377
## yearly_income      0.20858242  -0.20858817
## male_percent       1.00000000  -0.99999998
## female_percent     -0.99999998   1.00000000
```

```
corrplot(cor(numericCol), type="upper")
```



Anova test ***for categorical variables:***

to find the influence of the categorical variables on the Salary an Anova test is performed. The Anova test showed that all variables are significant

```
AnovaTest <- aov( yearly_income ~ Age + education +gender + job + rank + employment_type +
male_percent + Industry_type + yrs_of_Exp + REGION, data = german_sal_data )

summary(AnovaTest)
##                    Df    Sum Sq       Mean Sq F value Pr(>F)
## Age                 1      5.637e+11   5.637e+11  5079.3 <2e-16 ***
## education           6       3.380e+12   5.634e+11  5076.7 <2e-16 ***
## gender              1       8.898e+11   8.898e+11  8017.5 <2e-16 ***
## job                 1       9 1.20e+12   6.340e+10   571.2 <2e-16 ***
## rank                6       2.181e+12   3.635e+11  3275.2 <2e-16 ***
## employment_type     3       6.270e+10   2.090e+10   188.3 <2e-16 ***
## male_percent        1       1.887e+10   1.887e+10   170.0 <2e-16 ***
## Industry_type       8       9.064e+10   1.133e+10   102.1 <2e-16 ***
## yrs_of_Exp          1       1.820e+11   1.820e+11  1639.7 <2e-16 ***
## REGION              1       4.401e+11   4.401e+11  3965.5 <2e-16 ***
## Residuals       43373 4.813e+12   1.110e+08
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Splitting Training & Test Data

For the linear model the data is split for the training and testing. 70% of the data is considered for the training and the rest for the testing.

```
set.seed(12345)
TrainSet <- createDataPartition(y = german_sal_data$yearly_income, p = 0.70, list = FALSE)
training <- german_sal_data[TrainSet,]
testing <- german_sal_data[-TrainSet,]
set.seed(12345)
```

Linear Model

we assume that the salary is linearly dependent on the factors like education, skill, experience, etc. and we built a linear model as shown below

```
Model1 <- lm(yearly_income~.,data = training)
## Regression model
```

```
summary(Model1)
## Call:
## lm(formula = yearly_income ~ ., data = training)
## Residuals:
##    Min     1Q Median     3Q    Max
## -51146  -5958   -839   4707  68108
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   8.404e+05  1.227e+06   0.685 0.493361
## Age                           1.381e+02  6.881e+00  20.068  < 2e-16 ***
## yrs_of_Exp                    2.108e+02  7.946e+00  26.531  < 2e-16 ***
## weekly_work_hour              2.678e+02  2.150e+01  12.453  < 2e-16 ***
## male_percent                 -8.375e+03  1.227e+04  -0.683 0.494867
## female_percent                -8.393e+03  1.227e+04  -0.684 0.493933
## REGION2                      -8.433e+03  1.943e+02 -43.405  < 2e-16 ***
## employee_type2                2.756e+03  3.309e+02   8.328  < 2e-16 ***
```

```
## Industry_type2                2.508e+03  3.217e+02   7.795 6.64e-15 ***
## Industry_type3                2.464e+03  4.494e+02   5.483 4.22e-08 ***
….
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error: 10270 on 30331 degrees of freedom
## Multiple R-squared:  0.6706, Adjusted R-squared:  0.6699
## F-statistic:   965 on 64 and 30331 DF,  p-value: < 2.2e-16
```

from the above results we see that male_percent and female_percent have no influence on the salary earned. So both columns were removed and the model was rebuilt again

Iterated Model

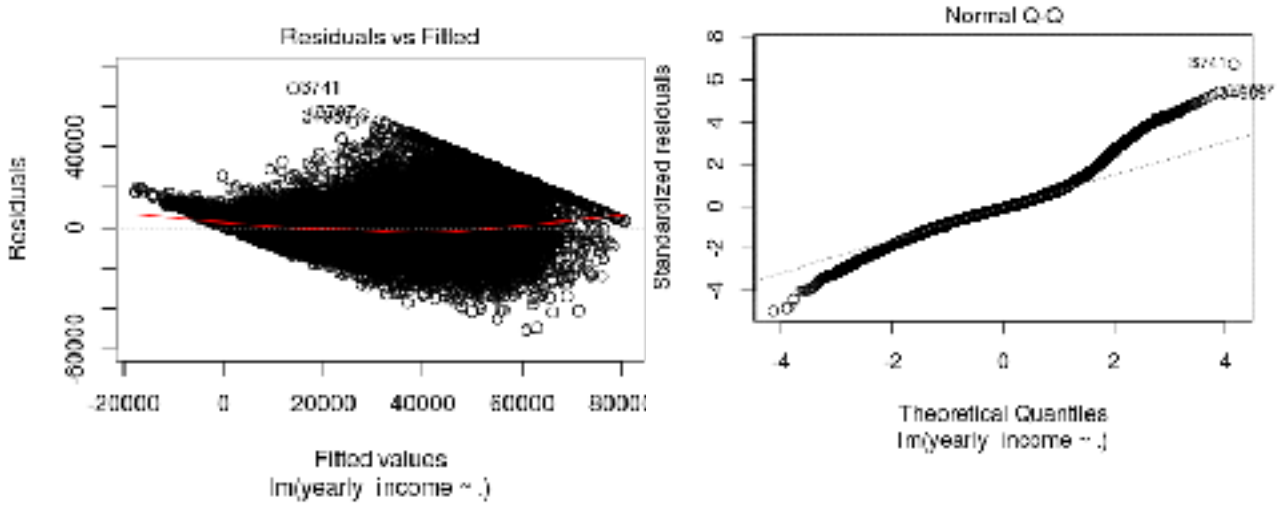
```
german_sal_data<-german_sal_data[ ,!(colnames(german_sal_data) %in%
c("male_percent","female_percent"))]
training<-training[ ,!(colnames(training) %in% c("male_percent","female_percent"))]
testing<-testing[ ,!(colnames(testing) %in% c("male_percent","female_percent"))]

Model2 <- lm(yearly_income~.,data = training)
summary(Model2)
## Call:
## lm(formula = yearly_income ~ ., data = training)
## Residuals:
##    Min     1Q Median     3Q    Max
## -50594  -5962   -827   4722  68656
## Coefficients:
##                              Estimate Std. Error             t value Pr(>|t|)
## (Intercept)           2033.934       2322.384       0.876 0.381148
## Age                    138.605      6.883      20.138  < 2e-16 ***
## yrs_of_Exp                 210.918      7.950              26.530  < 2e-16 ***
## weekly_work_hour      270.913     21.499         12.601  < 2e-16 ***
## REGION2                    -8521.222   193.689 -43.994  < 2e-16 ***
## employee_type2             2725.435    330.941            8.235  < 2e-16 ***
## Industry_type2             2819.230    316.803            8.899  < 2e-16 ***
## Industry_type3             2819.164    445.010            6.335 2.41e-10 ***
  ...
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error: 10270 on 30333 degrees of freedom
```

Multiple R-squared: 0.6703, Adjusted R-squared: 0.6696
F-statistic: 994.6 on 62 and 30333 DF, p-value: < 2.2e-16

Fitted Values on Test data and RMSE

```
predictedValues <- predict(Model2,testing,interval="confidence")
RMSE <- rmse(testing$yearly_income,predictedValues[,1]); RMSE
## [1] 10194.77
```

plot(Model2)

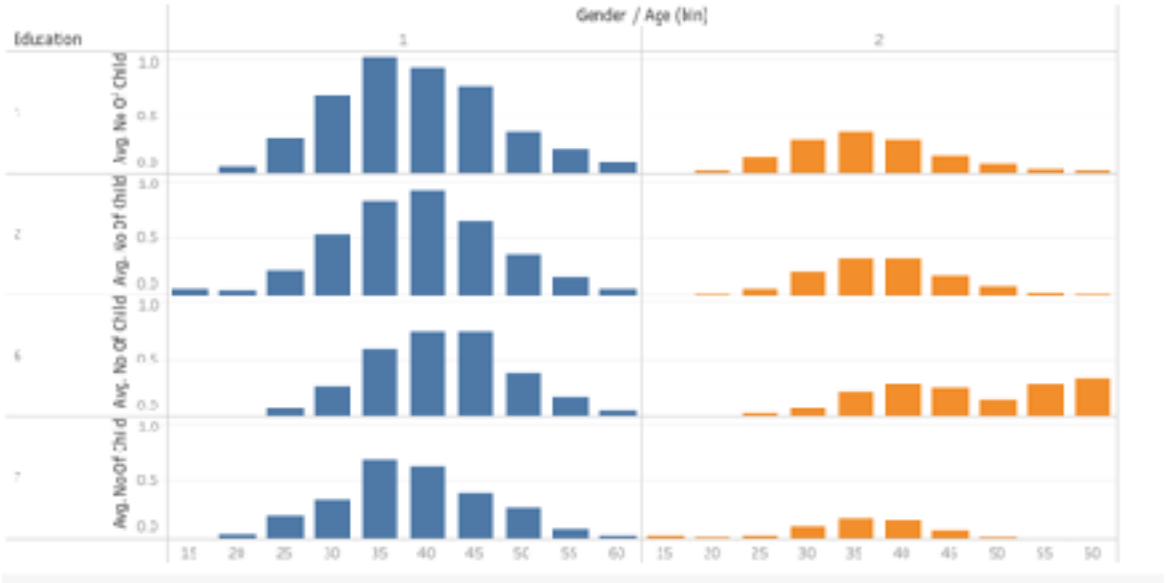


- from the above graph we detect **Heteroscedasticity.**
- **So we could infer that linear model does not suit this analysis. Either a non-linear model or some feature engineering prior to modeling is required**

Multi nominal modeling to predict no. of Child reported

Data Analysis:

One of the major challenge in developing economy is the shrinking population. It was interesting to find that the Nr. Of child reported by the employees depend on the age, education and industry they have been working. It appears that male reported more children that the female. There were certain industries where they reported more children that the other.



Chi-square-Test

Chi-square test is used to compare the relationship between the categorical variables. It is pairwise test and has to be run against all combinations of the categorical variables

```
chitest <- data.frame()
for(i in 1:(dim(factorcCol)[2]))
  {X <- mapply(function(x, y) chisq.test(x, y)$p.value, factorcCol[, -i], MoreArgs=list(factorcCol[,i]))
   chitest <- rbind(chitest,X)  }
```

the Chi-square test values were almost 0 for all the pairs and hence the features were significant and were not removed

Multi Nominal Model

```
mnModel2 <- multinom(no_of_child ~., training,maxit=1000)
## initial  value 66786.838253
## iter  10 value 34887.842268
….
## iter 260 value 23334.138623
## final  value 23334.103718
## converged
```

Predicted values & confusion matrix:

```
MnPredict2 <- predict(mnModel2,testing)
confusionMatrix(testing$no_of_child,MnPredict2)
## Confusion Matrix and Statistics
##                  Reference
## Prediction       0    0.5  1    1.5  2    2.5  3    3.5  4
##       0       9617    0    4    0   11    0    0    0    0
##       0.5      430    0    0    0    0    0    0    0    0
##       1       1412    0    1    0    7    0    0    0    0
##       1.5       85    0    0    0    0    0    0    0    0
##       2       1095    0    0    0    7    0    0    0    0
##       2.5       30    0    0    0    0    0    0    0    0
##       3        216    0    0    0    3    0    0    0    0
##       3.5        3    0    0    0    0    0    0    0    0
##       4        103    0    0    0    1    0    0    0    0

##
## Statistics by Class:
##
##                      Class: 0   Class: 0.5   Class: 1   Class: 1.5   Class: 2
## Sensitivity          0.7403      NA          2.000e-01      NA       0.2413793
## Specificity          0.5588      0.96699     8.910e-01   0.993474    0.9157433
## Pos Pred Value       0.9984      NA          7.042e-04      NA       0.0063521
## Neg Pred Value       0.0056      NA          9.997e-01      NA       0.9981548
## Prevalence           0.9974    0.00000       3.839e-04   0.000000    0.0022265
## Detection Rate       0.7383    0.00000       7.678e-05   0.000000    0.0005374
## Detection Prevalence  0.7395   0.03301       1.090e-01   0.006526    0.0846065
```

## Balanced Accuracy	**0.6496**	**NA**	**.54**	**NA**	**0.5785613**

##	Class: 2.5	Class: 3	Class: 3.5	Class: 4
## Sensitivity	NA	NA	NA	NA
## Specificity	0.997697	0.98319	0.9997697	0.992015
## Pos Pred Value	NA	NA	NA	NA
## Neg Pred Value	NA	NA	NA	NA
## Prevalence	0.000000	0.00000	0.0000000	0.000000
## Detection Rate	0.000000	0.00000	0.0000000	0.000000
## Detection Prevalence	0.002303	0.01681	0.0002303	0.007985
## Balanced Accuracy	**NA**	**NA**	**NA**	**NA**

The accuracy of the above model is marginal and gives scope for further improvement by appropriately sampling the training data and also with the feature selection

Conclusion:

Data Analysis:

By analyzing the salary data we were able to infer that following features contributed significantly for a good salary

- Education from a technical university
- Industries like Banking & Insurance
- management skills
- Gender

Modeling:

- Predicting the salary from the given features was not linear and resulted in Heteroscedasticity.
- Using the multi-nominal prediction too resulted in low accuracy and warrants a good training set and better feature engineering