Homework2_5110_Bindulatha_Banisetti

Bindu Latha Banisetti

```
library("readr")
library("tidyr")
library("dplyr")
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

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

```
library("ggplot2")
```

```
library("rio")
library("tidyverse")
```

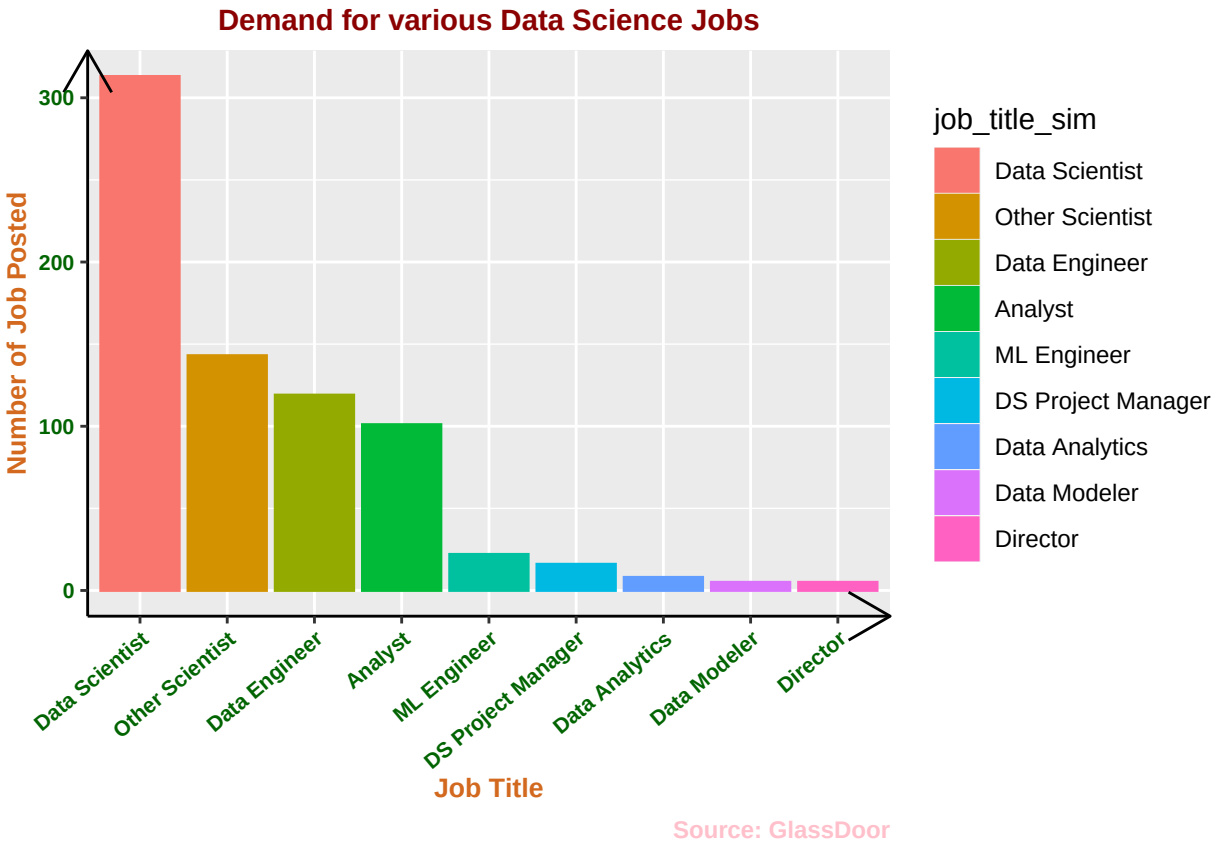
```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v tibble  3.1.6     v stringr 1.4.0
## v purrr   0.3.4     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library("ggplot2")
```

```
ds_salary<-read.csv("/Users/bindulathabanisetti/Desktop/data_cleaned_2022_salary.csv",header=TRUE)
# Removing the rows which have na values in the data
ds_salary<-filter(ds_salary,job_title_sim!="na")
```

```
#Counting the Number of jobs for each job kind
#we are doing this in order to explicitly specify the counts(Y-axis) for Bar-plot
ds_job_counts<-ds_salary %>% count(job_title_sim,sort=TRUE)

job_titles2<-c("data scientist"="Data Scientist","other scientist"="Other Scientist","data engineer"="Da
```

```r
ds_job_counts$job_title_sim <- as.vector(ds_job_counts$job_title_sim)
ds_job_counts$job_title_sim <- factor(ds_job_counts$job_title_sim,ds_job_counts$job_title_sim,labels=jo

## Demand for various Data Science Jobs
ggplot(data=ds_job_counts,
       mapping=aes(x=job_title_sim,
                   y=n,
                   color=job_title_sim,
                   fill=job_title_sim))+
  geom_bar(stat="identity")+
  theme(
    axis.line = element_line(arrow=arrow()),
    axis.title.x = element_text(
      size=10,
      color="chocolate",
      face="bold",
      vjust=1.5,
      hjust=0.5
    ),
    axis.title.y = element_text(
      size=10,
      color="chocolate",
      face="bold"
    ),
    axis.text.x = element_text(
      size=8,
      color="darkgreen",
      angle=40,
      face="bold",
      hjust = 1
    ),
    axis.text.y = element_text(
      size=8,
      color="darkgreen",
      face="bold"
    ),
    plot.title = element_text(
      size=11,
      color="darkred",
      face="bold",
      hjust=0.5
    ),
    plot.caption = element_text(
      color="pink",
      face="bold"
    )
  )+
  labs(
    x="Job Title",
    y="Number of Job Posted",
    title="Demand for various Data Science Jobs",
    caption = "Source: GlassDoor"
  )
```



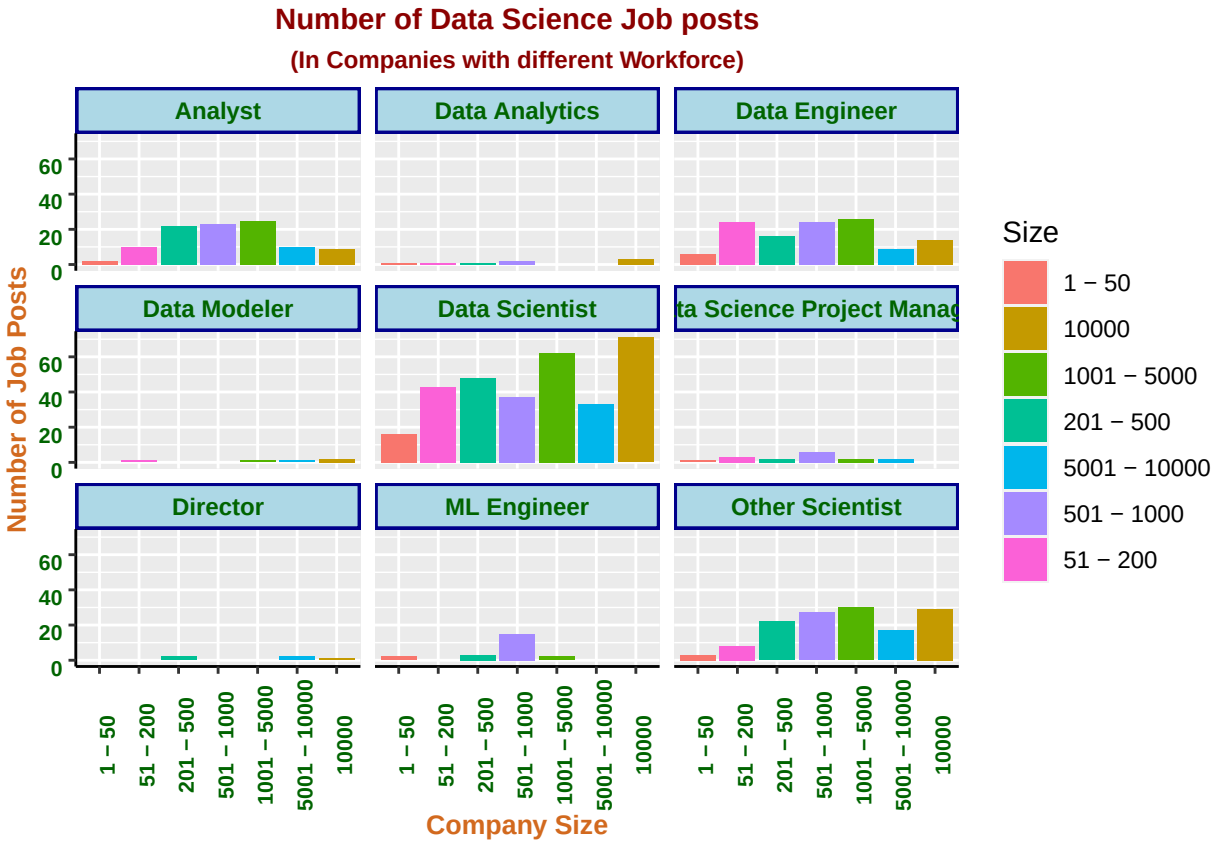
Observations

- Data Science jobs are sky rocketing nowadays.
- There are different kinds of jobs in this Domain. Most specifically Data Scientist, Machine Learning Engineer, Data Analytics, Analyst, Data Modeler, Data Engineer, Data Science Project Manager and so on.
- When we explore the number of jobs in each of these kinds, Data Scientist vacancies are way higher than any other job roles, where as the Data Engineer, other scientists and Analyst jobs are in mid-range constituting of just about 100 vacancies each.
- Lastly the demand for ML Engineer and remaining job roles is very less.

```
##Number of Data Science Job posts in companies having different workforce

job_titles3<-c("analyst"="Analyst","data analitics"="Data Analytics","data engineer"="Data Engineer","da

ggplot(data=ds_salary,
       mapping=aes(x=Size,
                   fill=Size))+
  geom_bar()+
  facet_wrap(~job_title_sim,labeller = labeller(job_title_sim=job_titles3))+
  scale_x_discrete(limits=c( "1 - 50","51 - 200","201 - 500","501 - 1000","1001 - 5000","5001 - 10000",
  theme(
    axis.line = element_line(linetype = "solid"),
    axis.title.x = element_text(
      size=10,
```

```
    color="chocolate",
    face="bold",
    vjust=1.5,
    hjust=0.5
  ),
  axis.title.y = element_text(
    size=10,
    color="chocolate",
    face="bold"
  ),
  axis.text.x = element_text(
    size=8,
    color="darkgreen",
    angle=90,
    face="bold",
    vjust=1
  ),
  axis.text.y = element_text(
    size=8,
    color="darkgreen",
    face="bold",
    vjust=1
  ),
  plot.title = element_text(
    size=11,
    color="darkred",
    face="bold",
    hjust=0.5
  ),
  plot.subtitle = element_text(
    size=9,
    color="darkred",
    face="bold",
    hjust=0.5
  ),
  strip.background = element_rect(
    color="darkblue",
    fill="lightblue",
    size=1,
    linetype = "solid"
  ),
  strip.text.x = element_text(
    face="bold",
    color = "darkgreen"
  )
)+
labs(
  x="Company Size",
  y="Number of Job Posts",
  title="Number of Data Science Job posts",
  subtitle = "(In Companies with different Workforce)"
)
```

```
## Warning: Removed 10 rows containing non-finite values (stat_count).
```



Observations

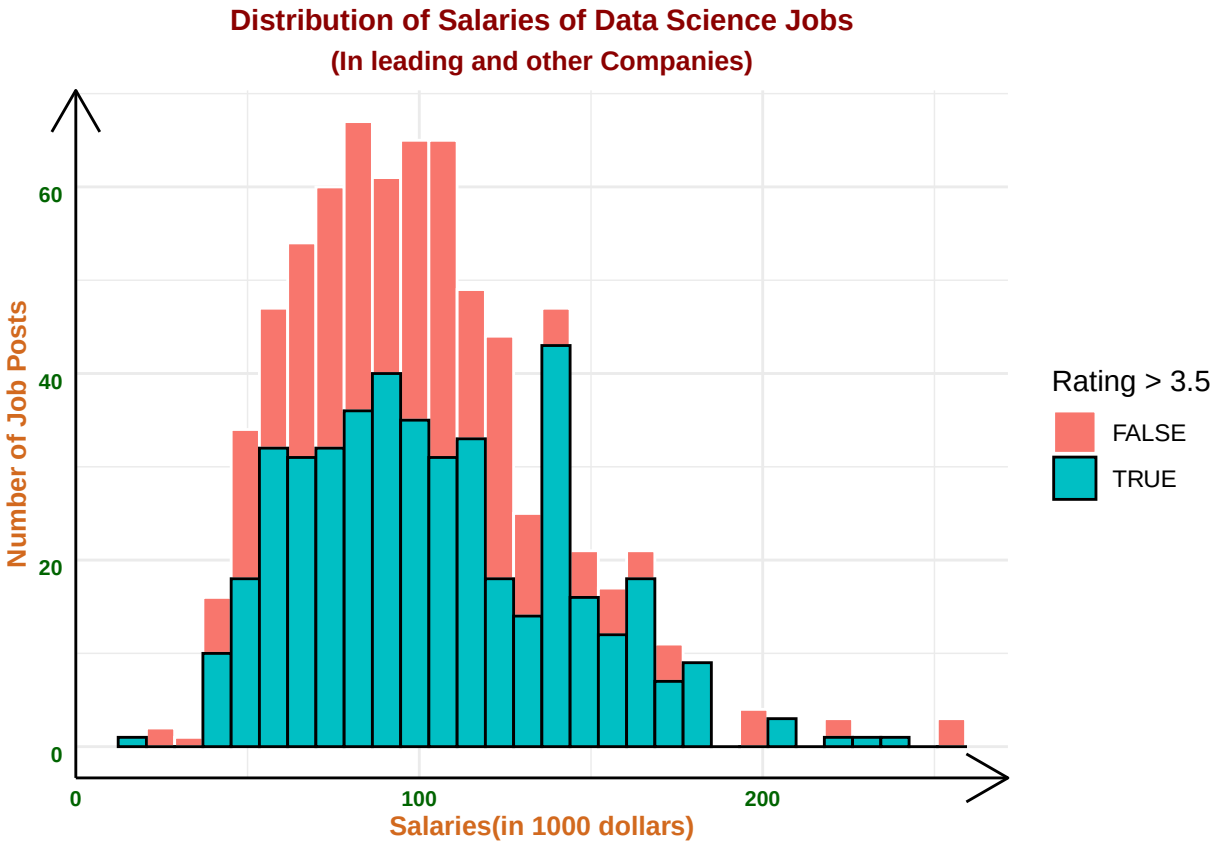
- Diving into the number of job positions in each kind of job role in the companies of different sizes, apparently Data Scientist jobs are very popular in all the small or medium and large scale companies.
- Specifically in large scale companies of size 10000 have the highest demand for Data Scientist jobs. But its demand is fluctuating while increasing the company's workforce.
- Interestingly the Data scientists jobs are significantly lower, which is just about 30 in the companies of size 5000 to 10000.
- Moreover, the ML Engineer posts are in demand only in the companies having 500 to 1000 employees

Distribution of Salaries of Data Science Jobs

```
ggplot(ds_salary,
       mapping=aes(x=Avg.Salary.K.,
                   fill=Rating>3.5,
                   color=Rating>3.5),
)+
  scale_color_manual(values=c("white","black"))+
  geom_histogram()+
  theme_minimal()+
  theme(
    axis.line = element_line(arrow=arrow()),
```

```
    axis.title.x = element_text(
      size=10,
      color="chocolate",
      face="bold",
      vjust=1.5,
      hjust=0.5
    ),
    axis.title.y = element_text(
      size=10,
      color="chocolate",
      face="bold"
    ),
    axis.text.x = element_text(
      size=8,
      color="darkgreen",
      face="bold",
      vjust=1
    ),
    axis.text.y = element_text(
      size=8,
      color="darkgreen",
      face="bold",
      vjust=1
    ),
    plot.title = element_text(
      size=11,
      color="darkred",
      face="bold",
      hjust=0.5
    ),
    plot.subtitle = element_text(
      size=10,
      color="darkred",
      face="bold",
      hjust=0.5
    )
  )+
  labs(
    x="Salaries(in 1000 dollars)",
    y="Number of Job Posts",
    title="Distribution of Salaries of Data Science Jobs",
    subtitle="(In leading and other Companies)"
  )
```

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

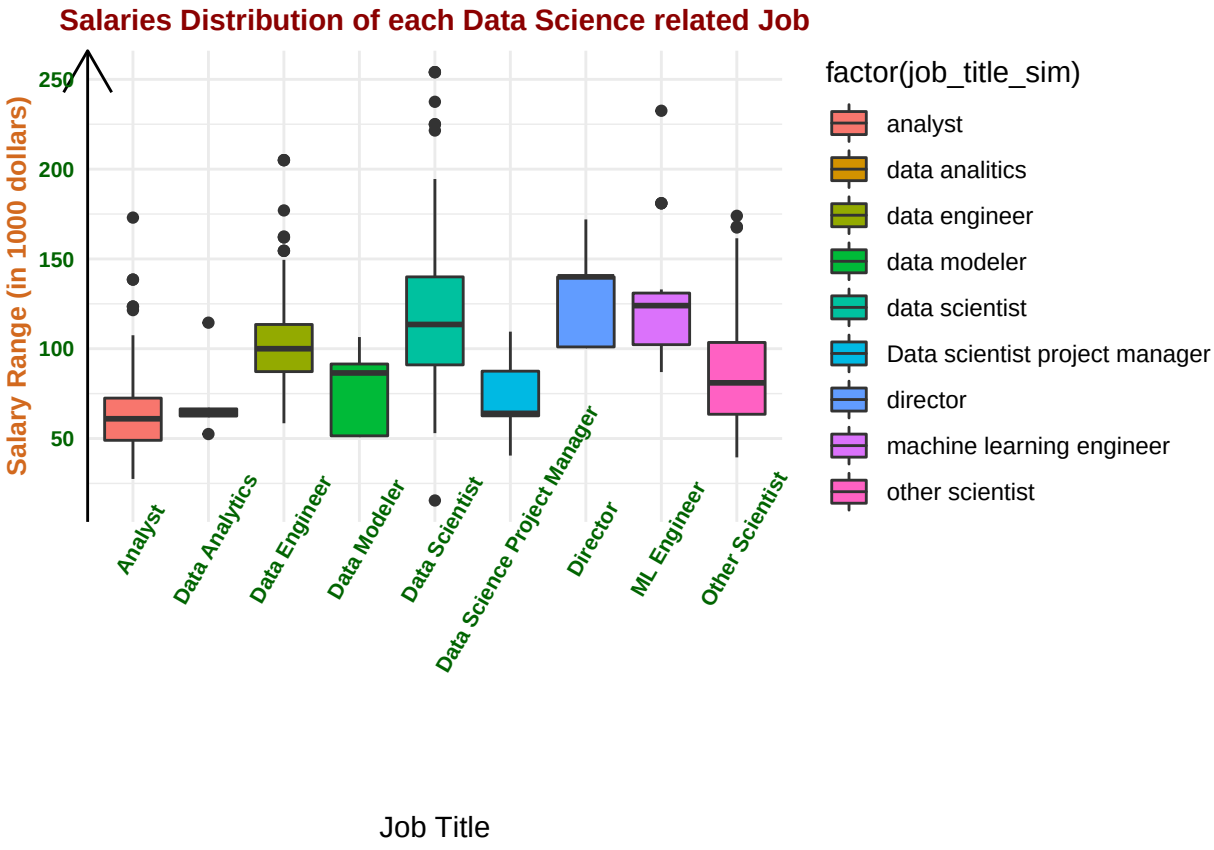


Observations

- The distribution of Salaries for Data Science specific jobs shows that highest number which is more than 60 job positions are offered with around $100K
- The jobs with highest packages(about $200K) can just be counted on our finger tips, making our graph right-skewed.
- Interestingly, the low rated companies are having higher number of job positions of all the salary ranges, compared to that of the leading companies.

```
# Creating labels for Job_title_sim variable
job_titles1<-c("analyst"="Analyst","data analitics"="Data Analytics","data engineer"="Data Engineer","da

#we are providing labels for displaying Job Titles in formatted way

# Salaries Distribution for each Data Science related Job
ggplot(data=ds_salary,
       mapping=aes(x=factor(job_title_sim,labels = job_titles1),
                   y=Avg.Salary.K.,
                   fill=factor(job_title_sim)),
       color="black")+
  geom_boxplot()+
  theme_minimal()+
  theme(
    axis.line.y = element_line(arrow=arrow()),
    axis.title.y = element_text(
```

```
    size=10,
    color="chocolate",
    face="bold"
  ),
  axis.text.x = element_text(
    size=8,
    color="darkgreen",
    angle=60,
    face="bold",
    vjust=1
  ),
  axis.text.y = element_text(
    size=8,
    color="darkgreen",
    face="bold"
  ),
  plot.title = element_text(
    size=11,
    color="darkred",
    face="bold",
    hjust=0.5
  )
)+
labs(
  x="Job Title",
  y="Salary Range (in 1000 dollars)",
  title="Salaries Distribution of each Data Science related Job"
)
```



Observations

- The range of salaries is larger in Data Scientist jobs, where as the range of salaries is very less in Data Analytics.
- The highest salary is given for Data Scientist and Director roles
- On the other hand, least salaries is given for Analysts.
- And, the average salary for Data Scientists, ML Engineers, Data Engineers and Directors is above $100K
- While for Data Modellers and other Scientists the average salary is less than $100K, and the mean salary for the remaining job roles are less than $75K.

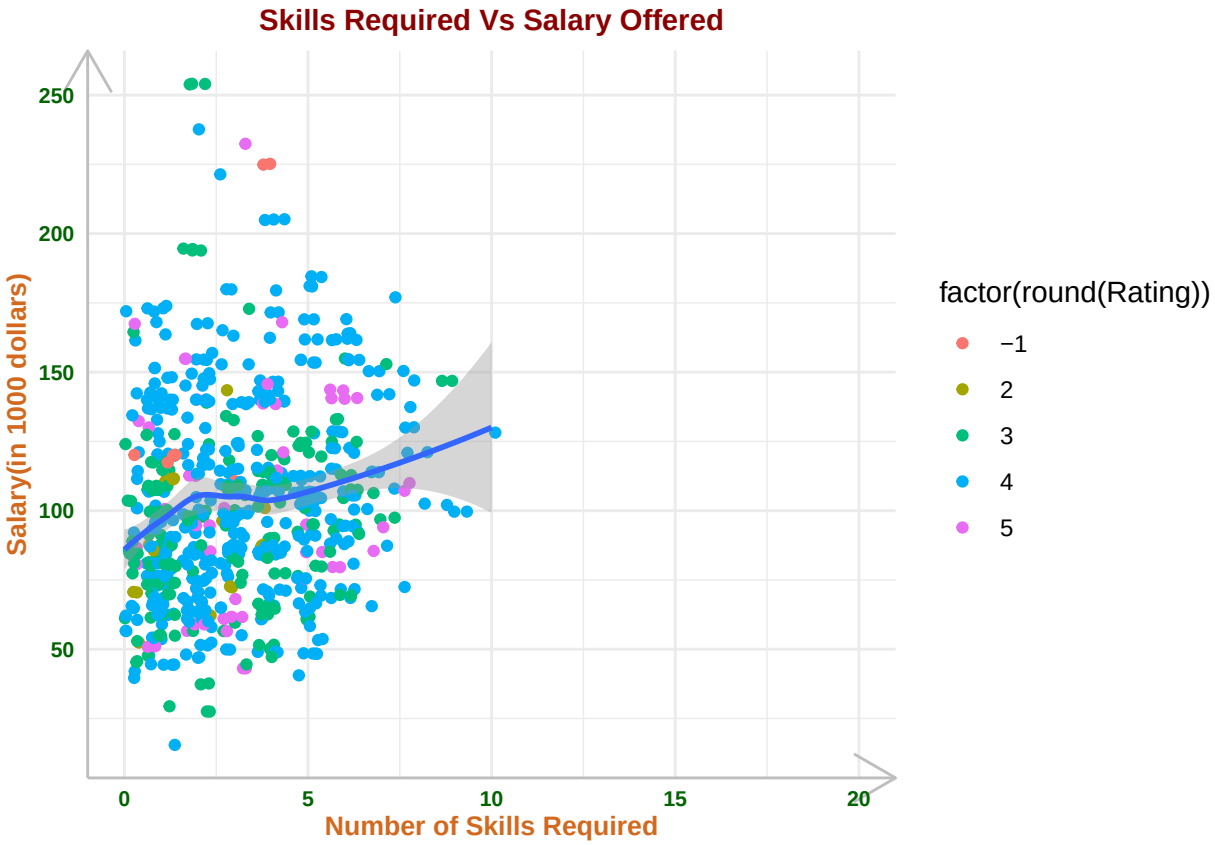
##Skills Required Vs Salaries Offered

```
ds_salary_skill<-ds_salary

#Adding Skill_num variable to capture the total no.of skills required for that particular job post
ds_salary_skill$Skills_num<-rowSums(cbind(ds_salary_skill[24:39]),na.rm=T)

ggplot(data=ds_salary_skill,
       mapping=aes(x=Skills_num,
                   y=Avg.Salary.K.))+
  geom_jitter(mapping=aes(color=factor(round(Rating))))+
  xlim(0,20)+
  theme_minimal()+
  geom_smooth()+
```

```
  theme(
    axis.line = element_line(arrow=arrow(),
                             color="grey"),
    axis.title.x = element_text(
      size=10,
      color="chocolate",
      face="bold",
      vjust=1.5,
      hjust=0.5
    ),
    axis.title.y = element_text(
      size=10,
      color="chocolate",
      face="bold"
    ),
    axis.text.x = element_text(
      size=8,
      color="darkgreen",
      face="bold",
      vjust=1
    ),
    axis.text.y = element_text(
      size=8,
      color="darkgreen",
      face="bold"
    ),
    plot.title = element_text(
      size=11,
      color="darkred",
      face="bold",
      hjust=0.5
    )
  )+
  labs(
    x="Number of Skills Required",
    y="Salary(in 1000 dollars)",
    title="Skills Required Vs Salary Offered"
  )
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

```
## Warning: Removed 53 rows containing missing values (geom_point).
```



Observations

- To identify the number of skills required, among the Python, Spark, AWS, Excel, SQL,SAS, Keras, PyTorch, SciKit, Tensor, Hadoop, Tableau, Bi, Flink, Mongo, Google Analytics to secure a specific range of salary plotted the scatter plot on Number of skills required and Salaries offered for that many skills.
- The association between the number of skills required and Salaries offered is kind of a linear regression with positive correlation.
- Interestingly, the top rated companies are offering lesser salaries than that of the lower rated companies for the employees with the same number of skills.
- Overall, with atmost 5 of the listed specific skills - one could secure a job with more than $100K.

```
#Reading .tsv file keeping NAs to the missing value
apr_df<-read_tsv("/Users/bindulathabanisetti/Desktop/APR_Data.tsv", na=c("","NA"))
```

```
## Rows: 6511 Columns: 76
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr  (4): SCL_NAME, SPORT_NAME, CONFNAME_14, D1_FB_CONF_14
## dbl (69): SCL_UNITID, SPORT_CODE, ACADEMIC_YEAR, SCL_DIV_14, SCL_SUB_14, SCL...
## lgl  (3): DATA_TAB_GENERALINFO, DATA_TAB_MULTIYRRATE, DATA_TAB_ANNUALRATE
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

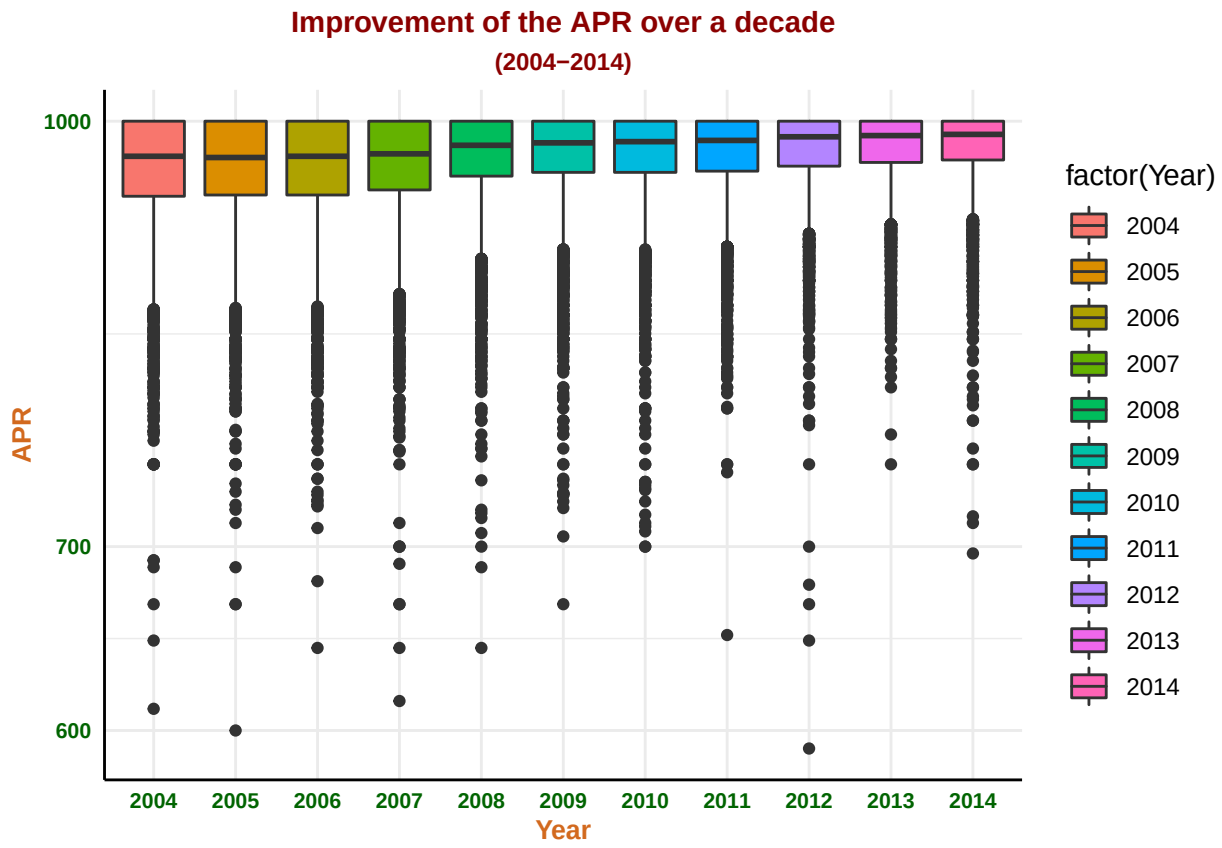
```
#Converting multiple columns into two columns making the table longer
apr_long<-pivot_longer(apr_df,starts_with("APR"),
                       names_to = "Year",
                       names_prefix = "APR_RATE_",
                       values_to ="APR" )

#Formatting Year column
apr_long<-mutate(apr_long,Year=as.integer(substring(apr_long$Year,1,4)))

#Selecting only required columns
apr_tidy<-apr_long %>%
  transmute(School_ID=SCL_UNITID,
            School_Name=SCL_NAME,
            Sport_Code=SPORT_CODE,
            Sport_Name=SPORT_NAME,
            Year=Year,
            APR=APR)

ggplot(data=apr_tidy,
       mapping=aes(x=factor(Year),
                   y=APR,
                   fill=factor(Year)))+
 geom_boxplot()+
  scale_y_log10()+
  theme_minimal()+
  theme(
    axis.line = element_line(linetype = "solid"),
    axis.title = element_text(
      size=10,
      color="chocolate",
      face="bold"
    ),
    axis.text.x = element_text(
      size=8,
      color="darkgreen",
      face="bold",
      vjust=1
    ),
    axis.text.y = element_text(
      size=8,
      color="darkgreen",
      face="bold"
    ),
    plot.title = element_text(
      size=11,
      color="darkred",
      face="bold",
      hjust=0.5
    ),
    plot.subtitle = element_text(
      size=9,
      color="darkred",
      face="bold",
```

```
      hjust=0.5
    )
  )+
  labs(
    x="Year",
    y="APR",
    title="Improvement of the APR over a decade",
    subtitle="(2004-2014)"
  )
```

```
## Warning in self$trans$transform(x): NaNs produced
```

```
## Warning: Transformation introduced infinite values in continuous y-axis
```

```
## Warning: Removed 4732 rows containing non-finite values (stat_boxplot).
```



Observations

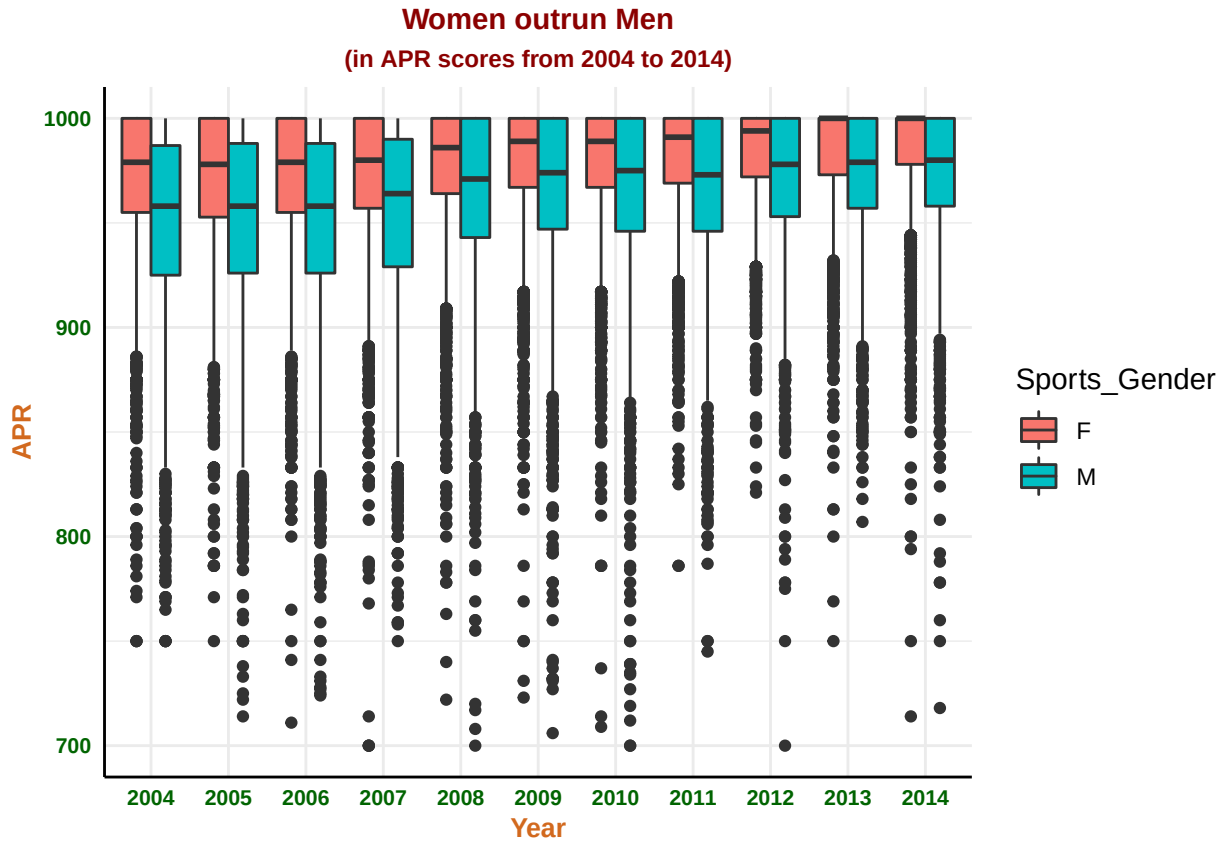
- The distributions of APRs from the year 2004 to 2014 is a kind of linear with a very small positive correlation.
- Range of APR distribution became thinner from 2004 to 2014 and more closer to 1000(highest possible score).
- APR value from 2004 to 2014 got improved on average, from the value of 950 to 1000 approximately

```r
#Removing Mixed Sports from the dataset
apr_tidy <- filter(apr_tidy,!grepl("Mixed",apr_tidy$Sport_Name))

#Creating Gender column to differentiate the sports between Male and Female
apr_tidy <- mutate(apr_tidy,
                   Sports_Gender=if_else((Sport_Code>=1 & Sport_Code<=18),
                                         "M","F"))

#Plotting boxplots to see the distribution of APR in men and women from 2004-2014
ggplot(data=apr_tidy,
       mapping=aes(x=factor(Year),
                   y=APR,
                   fill=Sports_Gender))+
  geom_boxplot(position=position_dodge())+
  scale_y_log10()+
  ylim(700,1000)+
  theme_minimal()+
  theme(
    axis.line = element_line(linetype = "solid"),
    axis.title = element_text(
      size=10,
      color="chocolate",
      face="bold"
    ),
    axis.text.x = element_text(
      size=8,
      color="darkgreen",
      face="bold",
      vjust=1
    ),
    axis.text.y = element_text(
      size=8,
      color="darkgreen",
      face="bold"
    ),
    plot.title = element_text(
      size=11,
      color="darkred",
      face="bold",
      hjust=0.5
    ),
    plot.subtitle = element_text(
      size=9,
      color="darkred",
      face="bold",
      hjust=0.5
    )
  )+
  labs(
    x="Year",
    y="APR",
    title="Women outrun Men",
    subtitle="(in APR scores from 2004 to 2014)"
```

```
  )
```

```
## Scale for 'y' is already present. Adding another scale for 'y', which will
## replace the existing scale.
```

```
## Warning: Removed 4722 rows containing non-finite values (stat_boxplot).
```



Observations

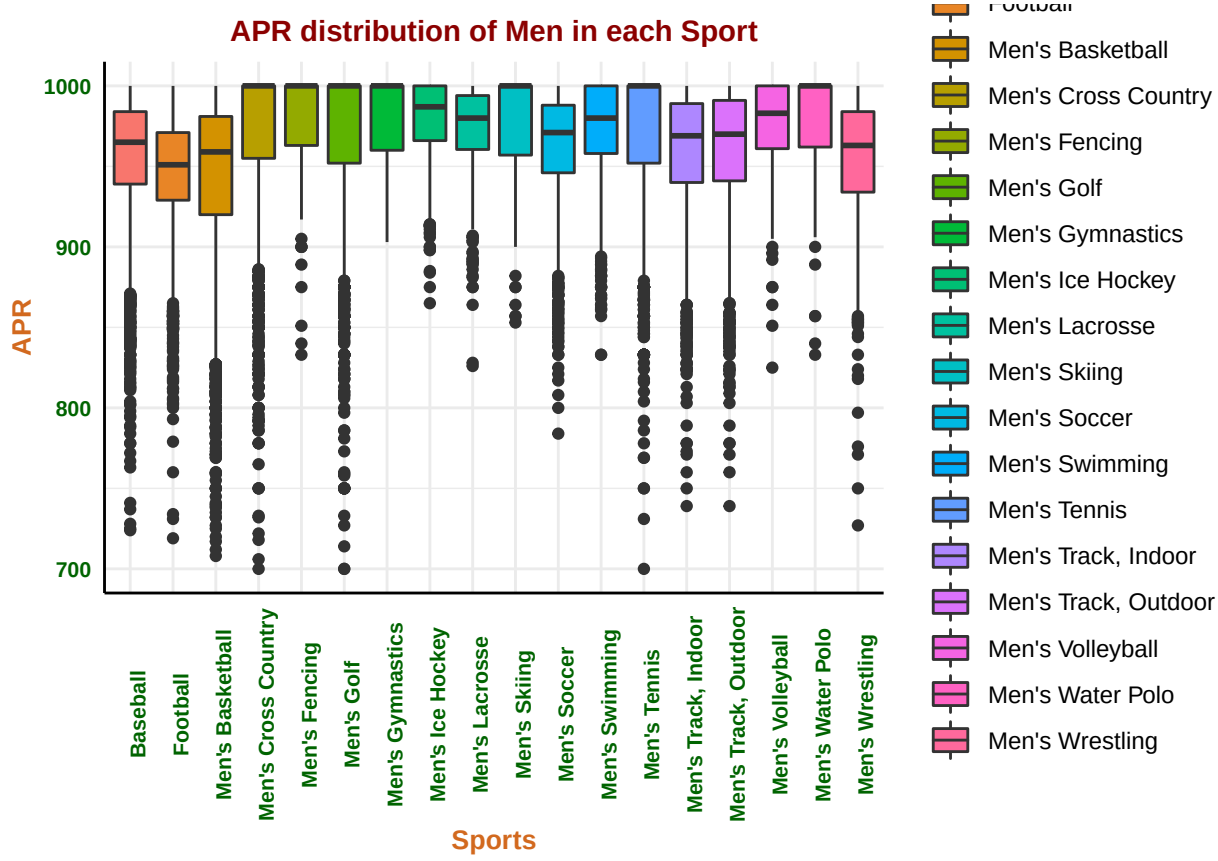
- Women are having better APR scores compared to men from the year 2004 to 2014.
- The median APR value of women is approximately 25 points higher than that of in men's APR values throughtput the whole decade 2004 to 2014
- The APR values of men and women differ in such a way that the range of APR values in women is more closer to 1000(highest possible points) than that of the men

```
#Filtering to have APR of only men
apr_men <- filter(apr_tidy, Sports_Gender=="M")

#Plotting boxplots to see the distribution of APR of Men in each sport category
ggplot(data=apr_men,
       mapping=aes(x=Sport_Name,
                   y=APR,
                   fill=Sport_Name))+
```

```
  scale_y_log10()+
  ylim(700,1000)+
  geom_boxplot()+
  theme_minimal()+
  theme(
    axis.line = element_line(linetype = "solid"),
    axis.title = element_text(
      size=10,
      color="chocolate",
      face="bold"
    ),
    axis.text.x = element_text(
      size=8,
      color="darkgreen",
      face="bold",
      angle=90,
      vjust=1
    ),
    axis.text.y = element_text(
      size=8,
      color="darkgreen",
      face="bold"
    ),
    plot.title = element_text(
      size=11,
      color="darkred",
      face="bold",
      hjust=0.5
    )
  )+
  labs(
    x="Sports",
    y="APR",
    title="APR distribution of Men in each Sport"
  )
```

```
## Scale for 'y' is already present. Adding another scale for 'y', which will
## replace the existing scale.
```

```
## Warning: Removed 2222 rows containing non-finite values (stat_boxplot).
```



Observations

- In men, the least APR value is scored by the Football team
- Apart from Football team, the teams of Baseball, Basketball and Wrestling also scored lower APR value on average
- On average, the highest APR value is scored by Cross-Country, Fencing, Golf, Gymnastics, Skiing, Tennis, Water Polo teams in men.