DATA ANALYSIS OF LOS ANGELES CITY PAYROLL

CIS-5250: Visual Analytics

Ramirez, Jose R
Parekh, Heta D

A. Introduction

The goal after graduating from graduate school is to obtain a good paying job. Aside from choosing what role one would like to obtain within an agency, one needs to determine whether to apply for public and/or private sector jobs. According to indeed (2021), government jobs offer various benefits which include: job security, consistent raises, and excellent health plans. These befits are echoed by Issid (n.d.). Issid is a writer for Monster and has a government job. He highlights job security as a benefit. In addition, he mentions his health benefits plan and pension plan as benefits.

One thing that is highlighted as a detriment when comparing private and public sector jobs is pay. Issid (n.d.) states that he could earn 25 percent more in the private sector in a similar capacity. This is supported by Yoder (2019) who reported that federal employees earn, on average, 27 percent less than private sector employees in similar roles.

Pension plans is one thing that is highlighted as benefit. According to Jamison (2016), the city of Los Angeles spent $1.04 billion in 2015 on retirement pensions and health care. This total amounts to 20 percent of the city’s operating budget. City employees benefit from generous pension plans. For example, a former Los Angeles detective received a pension of $109,232 in 2015. Similarly, the City of El Monte spends around 20 percent of its general fund on retirement costs. The high liability is due to pensions such as that of the prior El Monte city manager. The former city manager collects more than $216,000 a year (Dolan, 2016).

The focus of this paper is on pay as it relates to the public sector. This paper does not advocate for one specific sector; however, it aims to shed light on some aspects of Los Angeles payroll. The project is about employee payroll based in different departments of Los Angeles city. This project has been developed to find the pattern for the salary range and to understand which department offers the highest amount of salary with benefits. Employees have their own payroll management needs, and this project can help the HR department be aware of different department offers and improve the hiring process. The dataset contains various categories such as regular pay, benefit pay, and overtime pay. While the regular pay could consist of the salary employees earn, benefit pay would include various examples such as medical life, dental, disability, unemployment, and retirement benefits. The overtime pays as per California law states that employees paid overtime should be at the rate of one and half times the employee's regular rate of payment for all hours.

The authors of this paper explored the Los Angeles City payroll by answering the following questions:

- Which departments had the highest pay each year from 2013 to 2021?
- Which department generated the most overtime pay as per year?
- On average, which were the top 10 full-time positions that paid the most from 2013 to 2021?
- Which department generated the highest number of benefits pay each year?
- Which gender were given the highest amount of benefits each year?

B. Dataset URL’s

The dataset for this project has been taken from the URL link:

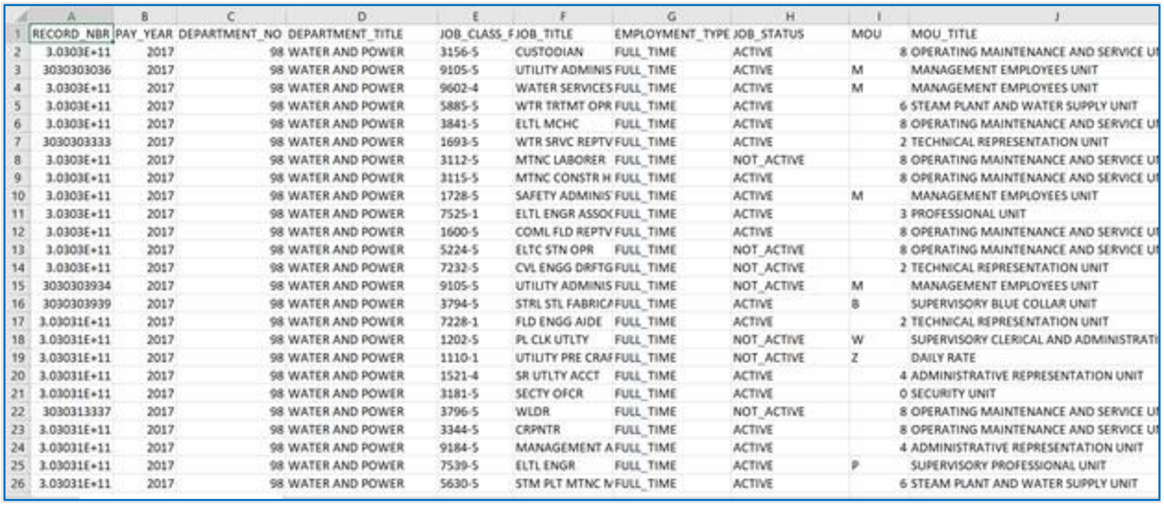
https://controllerdata.lacity.org/Payroll/City-Employee-Payroll-Current-/g9h8-fvhu/data

The dataset is frequently updated by the Los Angeles City. A copy of the dataset was downloaded on November 9, 2021.

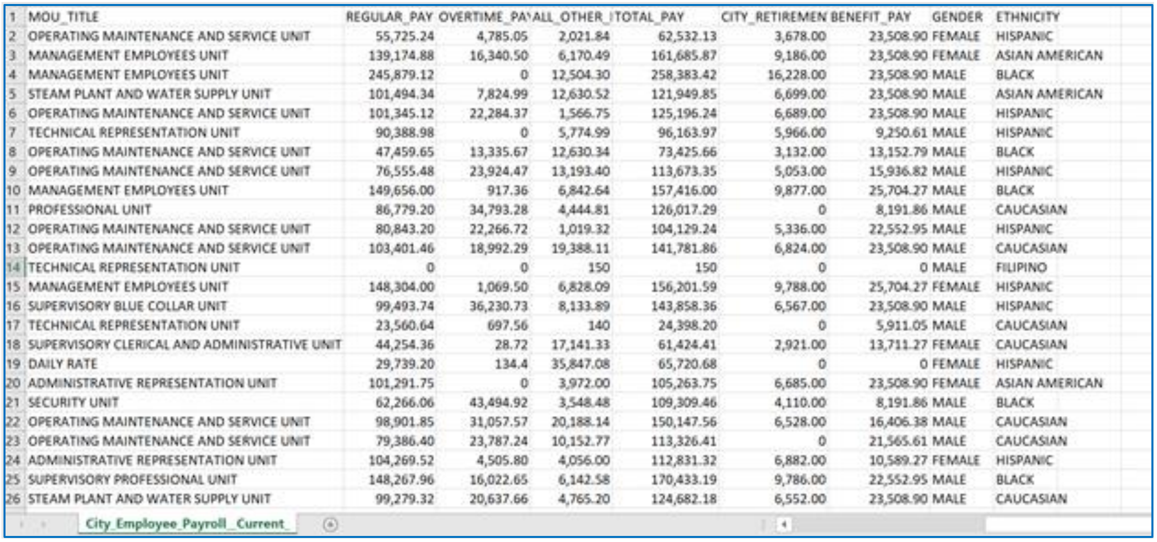
Format: CSV

Rows: 634,338

Columns: 9 usable fields



	A	B	C	D	E	F	G	H	I	J
1	RECORD_NBR	PAY_YEAR	DEPARTMENT_NO	DEPARTMENT_TITLE	JOB_CLASS_F	JOB_TITLE	EMPLOYMENT_TYPE	JOB_STATUS	MOU	MOU_TITLE
2	3.0303E+11	2017	98	WATER AND POWER	3156-5	CUSTODIAN	FULL_TIME	ACTIVE	8	OPERATING MAINTENANCE AND SERVICE UI
3	3030303036	2017	98	WATER AND POWER	9105-5	UTILITY ADMINIS	FULL_TIME	ACTIVE	M	MANAGEMENT EMPLOYEES UNIT
4	3.0303E+11	2017	98	WATER AND POWER	9602-4	WATER SERVICES	FULL_TIME	ACTIVE	M	MANAGEMENT EMPLOYEES UNIT
5	3.0303E+11	2017	98	WATER AND POWER	5885-5	WTR TRTMT OPR	FULL_TIME	ACTIVE	6	STEAM PLANT AND WATER SUPPLY UNIT
6	3.0303E+11	2017	98	WATER AND POWER	3841-5	ELTL MCHC	FULL_TIME	ACTIVE	8	OPERATING MAINTENANCE AND SERVICE UI
7	3030303333	2017	98	WATER AND POWER	1693-5	WTR SRVC REPTV	FULL_TIME	ACTIVE	2	TECHNICAL REPRESENTATION UNIT
8	3.0303E+11	2017	98	WATER AND POWER	3112-5	MTNC LABORER	FULL_TIME	NOT_ACTIVE	8	OPERATING MAINTENANCE AND SERVICE UI
9	3.0303E+11	2017	98	WATER AND POWER	3115-5	MTNC CONSTR H	FULL_TIME	ACTIVE	8	OPERATING MAINTENANCE AND SERVICE UI
10	3.0303E+11	2017	98	WATER AND POWER	1728-5	SAFETY ADMINIS	FULL_TIME	ACTIVE	M	MANAGEMENT EMPLOYEES UNIT
11	3.0303E+11	2017	98	WATER AND POWER	7525-1	ELTL ENGR ASSOC	FULL_TIME	ACTIVE	3	PROFESSIONAL UNIT
12	3.0303E+11	2017	98	WATER AND POWER	1600-5	COML FLD REPTV	FULL_TIME	ACTIVE	8	OPERATING MAINTENANCE AND SERVICE UI
13	3.0303E+11	2017	98	WATER AND POWER	5224-5	ELTC STN OPR	FULL_TIME	NOT_ACTIVE	8	OPERATING MAINTENANCE AND SERVICE UI
14	3.0303E+11	2017	98	WATER AND POWER	7232-5	CVL ENGG DRFTG	FULL_TIME	NOT_ACTIVE	2	TECHNICAL REPRESENTATION UNIT
15	3030303934	2017	98	WATER AND POWER	9105-5	UTILITY ADMINIS	FULL_TIME	NOT_ACTIVE	M	MANAGEMENT EMPLOYEES UNIT
16	3030303939	2017	98	WATER AND POWER	3794-5	STRL STL FABRICA	FULL_TIME	ACTIVE	B	SUPERVISORY BLUE COLLAR UNIT
17	3.03031E+11	2017	98	WATER AND POWER	7228-1	FLD ENGG AIDE	FULL_TIME	ACTIVE	2	TECHNICAL REPRESENTATION UNIT
18	3.03031E+11	2017	98	WATER AND POWER	1202-5	PL CLK UTLTY	FULL_TIME	NOT_ACTIVE	W	SUPERVISORY CLERICAL AND ADMINISTRATI
19	3.03031E+11	2017	98	WATER AND POWER	1110-1	UTILITY PRE CRAF	FULL_TIME	NOT_ACTIVE	Z	DAILY RATE
20	3.03031E+11	2017	98	WATER AND POWER	1521-4	SR UTLTY ACCT	FULL_TIME	ACTIVE	4	ADMINISTRATIVE REPRESENTATION UNIT
21	3.03031E+11	2017	98	WATER AND POWER	3181-5	SECTY OFCR	FULL_TIME	ACTIVE	0	SECURITY UNIT
22	3030313337	2017	98	WATER AND POWER	3796-5	WLDR	FULL_TIME	NOT_ACTIVE	8	OPERATING MAINTENANCE AND SERVICE UI
23	3.03031E+11	2017	98	WATER AND POWER	3344-5	CRPNTR	FULL_TIME	ACTIVE	8	OPERATING MAINTENANCE AND SERVICE UI
24	3.03031E+11	2017	98	WATER AND POWER	9184-5	MANAGEMENT A	FULL_TIME	ACTIVE	4	ADMINISTRATIVE REPRESENTATION UNIT
25	3.03031E+11	2017	98	WATER AND POWER	7539-5	ELTL ENGR	FULL_TIME	ACTIVE	P	SUPERVISORY PROFESSIONAL UNIT
26	3.03031E+11	2017	98	WATER AND POWER	5630-5	STM PLT MTNC M	FULL_TIME	ACTIVE	6	STEAM PLANT AND WATER SUPPLY UNIT



1	MOU_TITLE	REGULAR_PAY	OVERTIME_PA	ALL_OTHER_	TOTAL_PAY	CITY_RETIREMEN	BENEFIT_PAY	GENDER	ETHNICITY
2	OPERATING MAINTENANCE AND SERVICE UNIT	55,725.24	4,785.05	2,021.84	62,532.13	3,678.00	23,508.90	FEMALE	HISPANIC
3	MANAGEMENT EMPLOYEES UNIT	139,174.88	16,340.50	6,170.49	161,685.87	9,186.00	23,508.90	FEMALE	ASIAN AMERICAN
4	MANAGEMENT EMPLOYEES UNIT	245,879.12	0	12,504.30	258,383.42	16,228.00	23,508.90	MALE	BLACK
5	STEAM PLANT AND WATER SUPPLY UNIT	101,494.34	7,824.99	12,630.52	121,949.85	6,699.00	23,508.90	MALE	ASIAN AMERICAN
6	OPERATING MAINTENANCE AND SERVICE UNIT	101,345.12	22,284.37	1,566.75	125,196.24	6,689.00	23,508.90	MALE	HISPANIC
7	TECHNICAL REPRESENTATION UNIT	90,388.98	0	5,774.99	96,163.97	5,966.00	9,250.61	MALE	HISPANIC
8	OPERATING MAINTENANCE AND SERVICE UNIT	47,459.65	13,335.67	12,630.34	73,425.66	3,132.00	13,152.79	MALE	BLACK
9	OPERATING MAINTENANCE AND SERVICE UNIT	76,555.48	23,924.47	13,193.40	113,673.35	5,053.00	15,936.82	MALE	HISPANIC
10	MANAGEMENT EMPLOYEES UNIT	149,656.00	917.36	6,842.64	157,416.00	9,877.00	25,704.27	MALE	BLACK
11	PROFESSIONAL UNIT	86,779.20	34,793.28	4,444.81	126,017.29	0	8,191.86	MALE	CAUCASIAN
12	OPERATING MAINTENANCE AND SERVICE UNIT	80,843.20	22,266.72	1,019.32	104,129.24	5,336.00	22,552.95	MALE	HISPANIC
13	OPERATING MAINTENANCE AND SERVICE UNIT	103,401.46	18,992.29	19,388.11	141,781.86	6,824.00	23,508.90	MALE	CAUCASIAN
14	TECHNICAL REPRESENTATION UNIT	0	0	150	150	0	0	MALE	FILIPINO
15	MANAGEMENT EMPLOYEES UNIT	148,304.00	1,069.50	6,828.09	156,201.59	9,788.00	25,704.27	FEMALE	HISPANIC
16	SUPERVISORY BLUE COLLAR UNIT	99,493.74	36,230.73	8,133.89	143,858.36	6,567.00	23,508.90	MALE	HISPANIC
17	TECHNICAL REPRESENTATION UNIT	23,560.64	697.56	140	24,398.20	0	5,911.05	MALE	CAUCASIAN
18	SUPERVISORY CLERICAL AND ADMINISTRATIVE UNIT	44,254.36	28.72	17,141.33	61,424.41	2,921.00	13,711.27	FEMALE	CAUCASIAN
19	DAILY RATE	29,739.20	134.4	35,847.08	65,720.68	0	0	FEMALE	HISPANIC
20	ADMINISTRATIVE REPRESENTATION UNIT	101,291.75	0	3,972.00	105,263.75	6,685.00	23,508.90	FEMALE	ASIAN AMERICAN
21	SECURITY UNIT	62,266.06	43,494.92	3,548.48	109,309.46	4,110.00	8,191.86	MALE	BLACK
22	OPERATING MAINTENANCE AND SERVICE UNIT	98,901.85	31,057.57	20,188.14	150,147.56	6,528.00	16,406.38	MALE	CAUCASIAN
23	OPERATING MAINTENANCE AND SERVICE UNIT	79,386.40	23,787.24	10,152.77	113,326.41	0	21,565.61	MALE	CAUCASIAN
24	ADMINISTRATIVE REPRESENTATION UNIT	104,269.52	4,505.80	4,056.00	112,831.32	6,882.00	10,589.27	FEMALE	HISPANIC
25	SUPERVISORY PROFESSIONAL UNIT	148,267.96	16,022.65	6,142.58	170,433.19	9,786.00	22,552.95	MALE	BLACK
26	STEAM PLANT AND WATER SUPPLY UNIT	99,279.32	20,637.66	4,765.20	124,682.18	6,552.00	23,508.90	MALE	CAUCASIAN

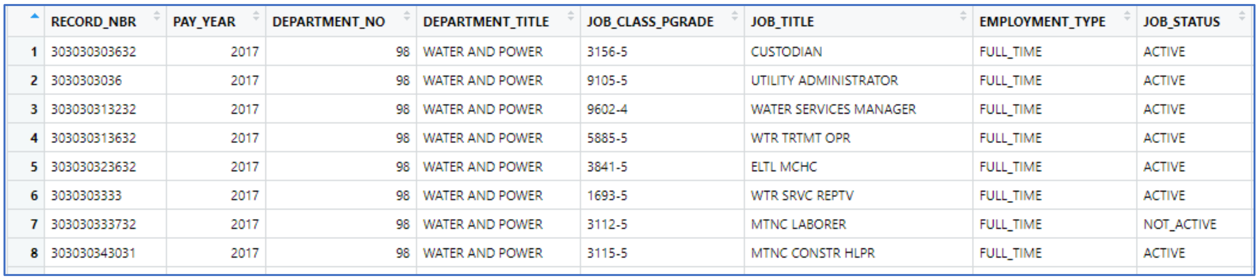
C. Dataset Description

Field Name	Description
PAY_YEAR	This field contains the year payroll was provided.
DEPARTMENT_TITLE	This field contains the department name of the employee.
MOU_TITLE	This field contains the job title of the employee.
EMPLOYMENT_TYPE	This field specifies if the employee is a full time, part time, or per event employee.
REGULAR_PAY	This field contains the pay provided to the employee. This field does not include benefit pay or overtime.
GENDER	This field contains the gender of the employee.
BENEFIT_PAY	This field includes the contribution the city provided to employee health benefits.
OVERTIME_PAY	This field refers to the compensation an employee receives for working beyond normal working hours.
ALL_OTHER_PAY	This field consists of all the non-monetary perks.

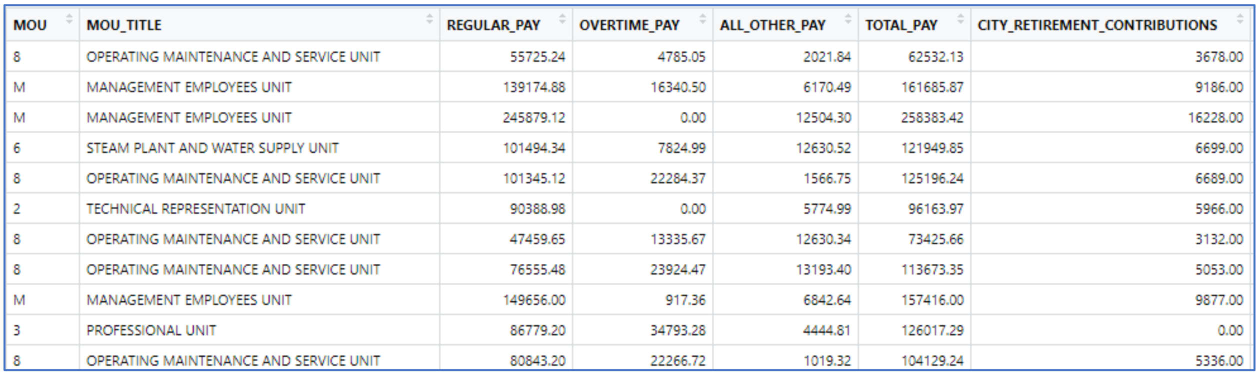
D. Data Cleaning

Removing Unnecessary Columns

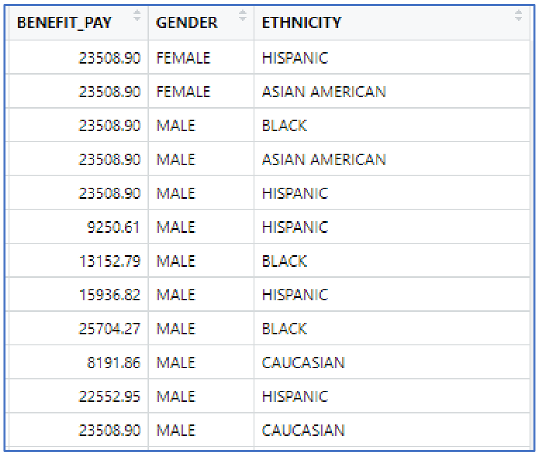
The payroll dataset contains a total of eighteen fields. For this project, nine of the eighteen columns were needed. As a result, one of the steps undertaken to clean the data was to remove unnecessary fields from the dataset. The dataset contained the following eighteen fields prior to removing the columns:



	RECORD_NBR	PAY_YEAR	DEPARTMENT_NO	DEPARTMENT_TITLE	JOB_CLASS_PGRADE	JOB_TITLE	EMPLOYMENT_TYPE	JOB_STATUS
1	303030303632	2017	98	WATER AND POWER	3156-5	CUSTODIAN	FULL_TIME	ACTIVE
2	3030303036	2017	98	WATER AND POWER	9105-5	UTILITY ADMINISTRATOR	FULL_TIME	ACTIVE
3	303030313232	2017	98	WATER AND POWER	9602-4	WATER SERVICES MANAGER	FULL_TIME	ACTIVE
4	303030313632	2017	98	WATER AND POWER	5885-5	WTR TRTMT OPR	FULL_TIME	ACTIVE
5	303030323632	2017	98	WATER AND POWER	3841-5	ELTL MCHC	FULL_TIME	ACTIVE
6	3030303333	2017	98	WATER AND POWER	1693-5	WTR SRVC REPTV	FULL_TIME	ACTIVE
7	303030333732	2017	98	WATER AND POWER	3112-5	MTNC LABORER	FULL_TIME	NOT_ACTIVE
8	303030343031	2017	98	WATER AND POWER	3115-5	MTNC CONSTR HLPR	FULL_TIME	ACTIVE

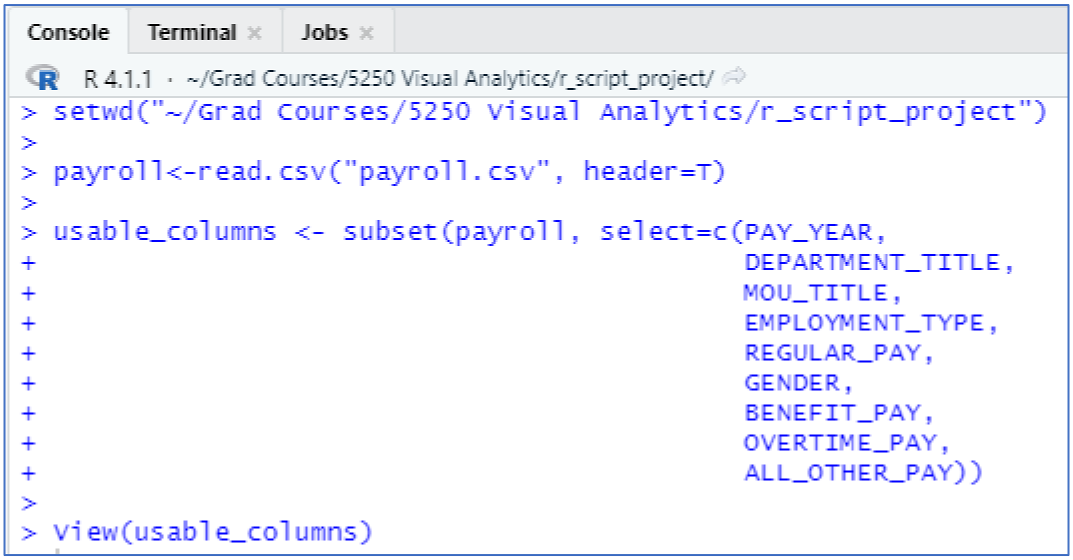


MOU	MOU_TITLE	REGULAR_PAY	OVERTIME_PAY	ALL_OTHER_PAY	TOTAL_PAY	CITY_RETIREMENT_CONTRIBUTIONS
8	OPERATING MAINTENANCE AND SERVICE UNIT	55725.24	4785.05	2021.84	62532.13	3678.00
M	MANAGEMENT EMPLOYEES UNIT	139174.88	16340.50	6170.49	161685.87	9186.00
M	MANAGEMENT EMPLOYEES UNIT	245879.12	0.00	12504.30	258383.42	16228.00
6	STEAM PLANT AND WATER SUPPLY UNIT	101494.34	7824.99	12630.52	121949.85	6699.00
8	OPERATING MAINTENANCE AND SERVICE UNIT	101345.12	22284.37	1566.75	125196.24	6689.00
2	TECHNICAL REPRESENTATION UNIT	90388.98	0.00	5774.99	96163.97	5966.00
8	OPERATING MAINTENANCE AND SERVICE UNIT	47459.65	13335.67	12630.34	73425.66	3132.00
8	OPERATING MAINTENANCE AND SERVICE UNIT	76555.48	23924.47	13193.40	113673.35	5053.00
M	MANAGEMENT EMPLOYEES UNIT	149656.00	917.36	6842.64	157416.00	9877.00
3	PROFESSIONAL UNIT	86779.20	34793.28	4444.81	126017.29	0.00
8	OPERATING MAINTENANCE AND SERVICE UNIT	80843.20	22266.72	1019.32	104129.24	5336.00



BENEFIT_PAY	GENDER	ETHNICITY
23508.90	FEMALE	HISPANIC
23508.90	FEMALE	ASIAN AMERICAN
23508.90	MALE	BLACK
23508.90	MALE	ASIAN AMERICAN
23508.90	MALE	HISPANIC
9250.61	MALE	HISPANIC
13152.79	MALE	BLACK
15936.82	MALE	HISPANIC
25704.27	MALE	BLACK
8191.86	MALE	CAUCASIAN
22552.95	MALE	HISPANIC
23508.90	MALE	CAUCASIAN

The R code used to remove unnecessary columns is as follows:

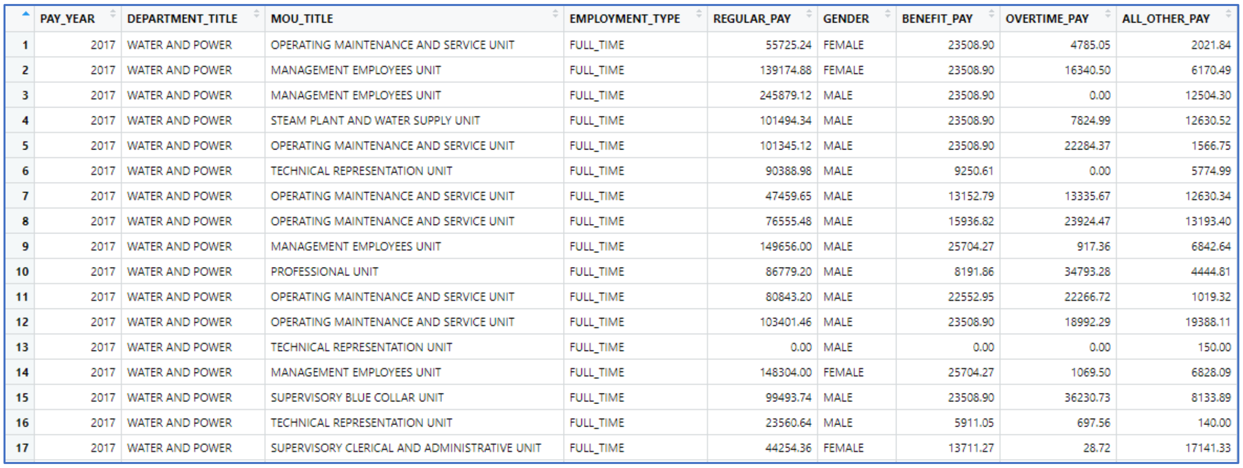

Console
Terminal
Jobs
R 4.1.1 · ~/Grad Courses/5250 Visual Analytics/r_script_project/

```
> setwd("~/Grad Courses/5250 Visual Analytics/r_script_project")
>
> payroll<-read.csv("payroll.csv", header=T)
>
> usable_columns <- subset(payroll, select=c(PAY_YEAR,
+                                            DEPARTMENT_TITLE,
+                                            MOU_TITLE,
+                                            EMPLOYMENT_TYPE,
+                                            REGULAR_PAY,
+                                            GENDER,
+                                            BENEFIT_PAY,
+                                            OVERTIME_PAY,
+                                            ALL_OTHER_PAY))
>
> View(usable_columns)
```

The code included in the screenshot of the R-Studio console is as follows:

```
> setwd("~/Grad Courses/5250 Visual Analytics/r_script_project")
>
> payroll<-read.csv("payroll.csv", header=T)
>
> usable_columns <- subset(payroll, select=c(PAY_YEAR,
+                                   DEPARTMENT_TITLE,
+                                   MOU_TITLE,
+                                   EMPLOYMENT_TYPE,
+                                   REGULAR_PAY,
+                                   GENDER,
+                                   BENEFIT_PAY,
+                                       OVERTIME_PAY,
+                                       ALL_OTHER_PAY))
>
> View(usable_columns)
```

Once the code is executed, the output generated by R Studio is as follows:



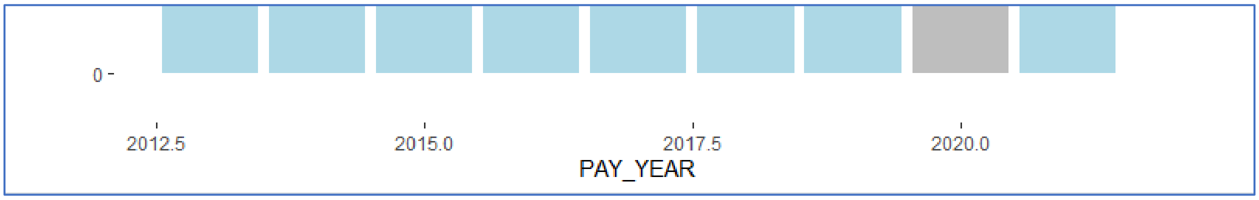
	PAY_YEAR	DEPARTMENT_TITLE	MOU_TITLE	EMPLOYMENT_TYPE	REGULAR_PAY	GENDER	BENEFIT_PAY	OVERTIME_PAY	ALL_OTHER_PAY
1	2017	WATER AND POWER	OPERATING MAINTENANCE AND SERVICE UNIT	FULL_TIME	55725.24	FEMALE	23508.90	4785.05	2021.84
2	2017	WATER AND POWER	MANAGEMENT EMPLOYEES UNIT	FULL_TIME	139174.88	FEMALE	23508.90	16340.50	6170.49
3	2017	WATER AND POWER	MANAGEMENT EMPLOYEES UNIT	FULL_TIME	245879.12	MALE	23508.90	0.00	12504.30
4	2017	WATER AND POWER	STEAM PLANT AND WATER SUPPLY UNIT	FULL_TIME	101494.34	MALE	23508.90	7824.99	12630.52
5	2017	WATER AND POWER	OPERATING MAINTENANCE AND SERVICE UNIT	FULL_TIME	101345.12	MALE	23508.90	22284.37	1566.75
6	2017	WATER AND POWER	TECHNICAL REPRESENTATION UNIT	FULL_TIME	90388.98	MALE	9250.61	0.00	5774.99
7	2017	WATER AND POWER	OPERATING MAINTENANCE AND SERVICE UNIT	FULL_TIME	47459.65	MALE	13152.79	13335.67	12630.34
8	2017	WATER AND POWER	OPERATING MAINTENANCE AND SERVICE UNIT	FULL_TIME	76555.48	MALE	15936.82	23924.47	13193.40
9	2017	WATER AND POWER	MANAGEMENT EMPLOYEES UNIT	FULL_TIME	149656.00	MALE	25704.27	917.36	6842.64
10	2017	WATER AND POWER	PROFESSIONAL UNIT	FULL_TIME	86779.20	MALE	8191.86	34793.28	4444.81
11	2017	WATER AND POWER	OPERATING MAINTENANCE AND SERVICE UNIT	FULL_TIME	80843.20	MALE	22552.95	22266.72	1019.32
12	2017	WATER AND POWER	OPERATING MAINTENANCE AND SERVICE UNIT	FULL_TIME	103401.46	MALE	23508.90	18992.29	19388.11
13	2017	WATER AND POWER	TECHNICAL REPRESENTATION UNIT	FULL_TIME	0.00	MALE	0.00	0.00	150.00
14	2017	WATER AND POWER	MANAGEMENT EMPLOYEES UNIT	FULL_TIME	148304.00	FEMALE	25704.27	1069.50	6828.09
15	2017	WATER AND POWER	SUPERVISORY BLUE COLLAR UNIT	FULL_TIME	99493.74	MALE	23508.90	36230.73	8133.89
16	2017	WATER AND POWER	TECHNICAL REPRESENTATION UNIT	FULL_TIME	23560.64	MALE	5911.05	697.56	140.00
17	2017	WATER AND POWER	SUPERVISORY CLERICAL AND ADMINISTRATIVE UNIT	FULL_TIME	44254.36	FEMALE	13711.27	28.72	17141.33

The code performs the following steps. First, the setwd function is executed to set the working directory. The working directory is where the R script and csv file containing the data is stored. The read.csv function is utilized to load the data contained in the csv file into a data frame titled “payroll.” The read.csv function accepts two parameters, the first is used to provide the name of the file whereas the second indicates that the csv file contains a header row. Once the data is loaded into the data frame, the subset function coupled with the select parameter is used to retrieve the necessary columns and store them in a new data frame title “usable_columns.” The select parameter is utilized to specify which columns to retrieve.

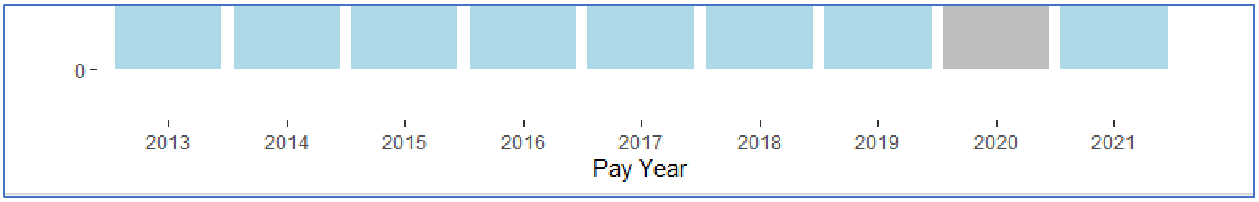
Convert Column Data Type

The columns used for this project were converted to an appropriate data type. Specifically, the pay_year field had to be converted from integer to character. This conversion was undertaken to ensure the pay_year was reflected correctly on the x-axis. In the process of converting this field, all other fields were also converted to ensure the appropriate data type was set.

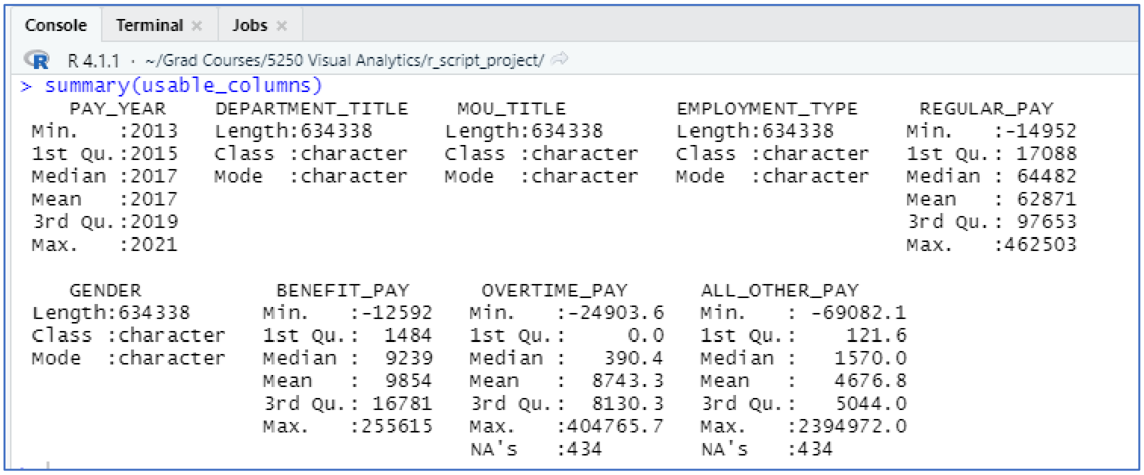
On one of the charts, the pay_year would be interpreted as an integer; as a result, it would be reflected as a scale with some values missing:



However, once the data type was changed to character, the label would be displayed correctly as follows:

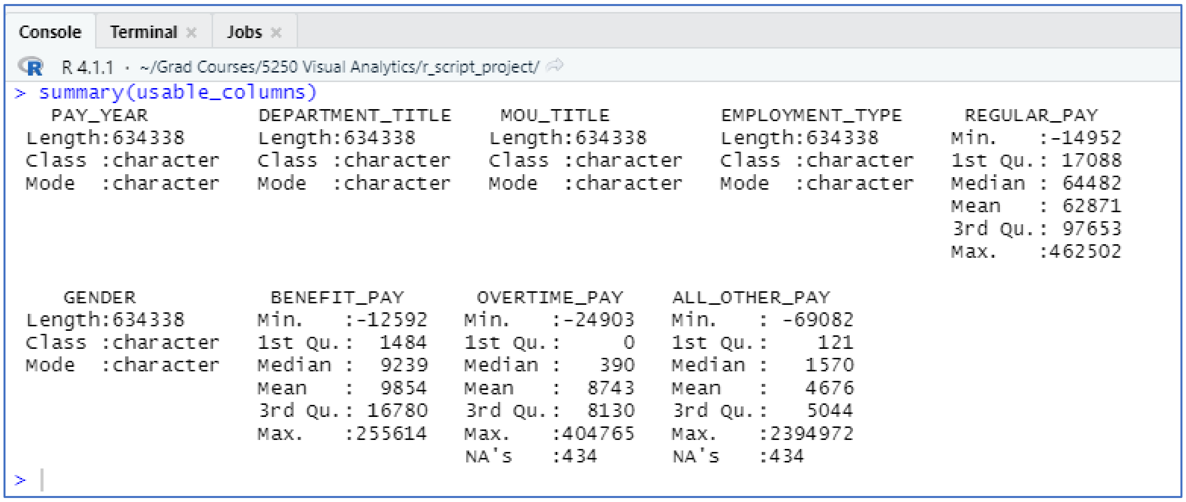


Prior to making the conversion, one can see that the pay_year field is defined as an integer by R as seen by the metrics provided by the summary function:


Console
Terminal
Jobs
R 4.1.1 · ~/Grad Courses/5250 Visual Analytics/r_script_project/

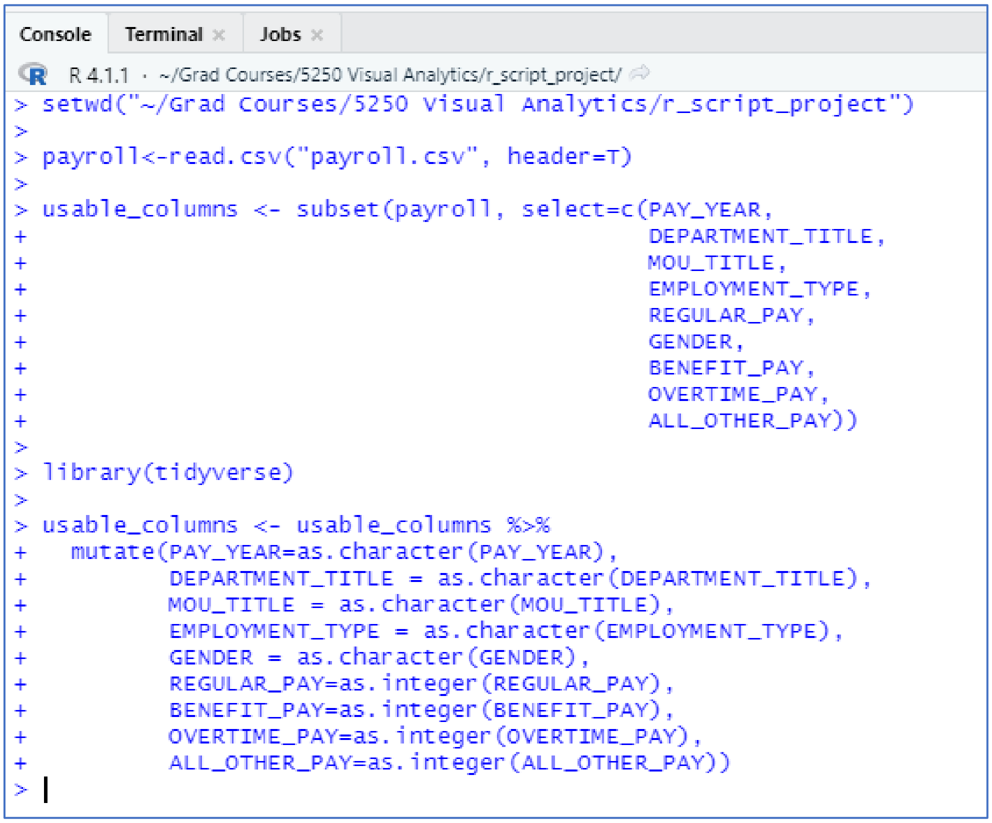
```
> summary(usable_columns)
    PAY_YEAR    DEPARTMENT_TITLE     MOU_TITLE         EMPLOYMENT_TYPE      REGULAR_PAY
 Min.   :2013   Length:634338      Length:634338      Length:634338      Min.   :-14952
 1st Qu.:2015   Class :character   Class :character   Class :character   1st Qu.: 17088
 Median :2017   Mode  :character   Mode  :character   Mode  :character   Median : 64482
 Mean   :2017                                                            Mean   : 62871
 3rd Qu.:2019                                                            3rd Qu.: 97653
 Max.   :2021                                                            Max.   :462503

    GENDER           BENEFIT_PAY     OVERTIME_PAY       ALL_OTHER_PAY
 Length:634338      Min.   :-12592   Min.   :-24903.6   Min.   : -69082.1
 Class :character   1st Qu.:  1484   1st Qu.:     0.0   1st Qu.:    121.6
 Mode  :character   Median :  9239   Median :   390.4   Median :   1570.0
                    Mean   :  9854   Mean   :  8743.3   Mean   :   4676.8
                    3rd Qu.: 16781   3rd Qu.:  8130.3   3rd Qu.:   5044.0
                    Max.   :255615   Max.   :404765.7   Max.   :2394972.0
                                     NA's   :434        NA's   :434
```

After running the code to convert the data type to character, the summary function identifies the pay_year field as character:


Console
Terminal
Jobs
R 4.1.1 · ~/Grad Courses/5250 Visual Analytics/r_script_project/

```
> summary(usable_columns)
   PAY_YEAR          DEPARTMENT_TITLE    MOU_TITLE          EMPLOYMENT_TYPE     REGULAR_PAY
 Length:634338      Length:634338      Length:634338      Length:634338      Min.   :-14952
 Class :character   Class :character   Class :character   Class :character   1st Qu.: 17088
 Mode  :character   Mode  :character   Mode  :character   Mode  :character   Median : 64482
                                                                             Mean   : 62871
                                                                             3rd Qu.: 97653
                                                                             Max.   :462502

    GENDER           BENEFIT_PAY      OVERTIME_PAY     ALL_OTHER_PAY
 Length:634338      Min.   :-12592   Min.   :-24903   Min.   : -69082
 Class :character   1st Qu.:  1484   1st Qu.:     0   1st Qu.:    121
 Mode  :character   Median :  9239   Median :   390   Median :   1570
                    Mean   :  9854   Mean   :  8743   Mean   :   4676
                    3rd Qu.: 16780   3rd Qu.:  8130   3rd Qu.:   5044
                    Max.   :255614   Max.   :404765   Max.   :2394972
                                     NA's   :434      NA's   :434
> 
```

The tidyverse library was employed to perform the data type conversion. The following code was executed:


Console
Terminal
Jobs
R 4.1.1 · ~/Grad Courses/5250 Visual Analytics/r_script_project/

```
> setwd("~/Grad Courses/5250 Visual Analytics/r_script_project")
>
> payroll<-read.csv("payroll.csv", header=T)
>
> usable_columns <- subset(payroll, select=c(PAY_YEAR,
+                                            DEPARTMENT_TITLE,
+                                            MOU_TITLE,
+                                            EMPLOYMENT_TYPE,
+                                            REGULAR_PAY,
+                                            GENDER,
+                                            BENEFIT_PAY,
+                                            OVERTIME_PAY,
+                                            ALL_OTHER_PAY))
>
> library(tidyverse)
>
> usable_columns <- usable_columns %>%
+   mutate(PAY_YEAR=as.character(PAY_YEAR),
+          DEPARTMENT_TITLE = as.character(DEPARTMENT_TITLE),
+          MOU_TITLE = as.character(MOU_TITLE),
+          EMPLOYMENT_TYPE = as.character(EMPLOYMENT_TYPE),
+          GENDER = as.character(GENDER),
+          REGULAR_PAY=as.integer(REGULAR_PAY),
+          BENEFIT_PAY=as.integer(BENEFIT_PAY),
+          OVERTIME_PAY=as.integer(OVERTIME_PAY),
+          ALL_OTHER_PAY=as.integer(ALL_OTHER_PAY))
> 
```

The following is the code displayed in the screenshot provided above:

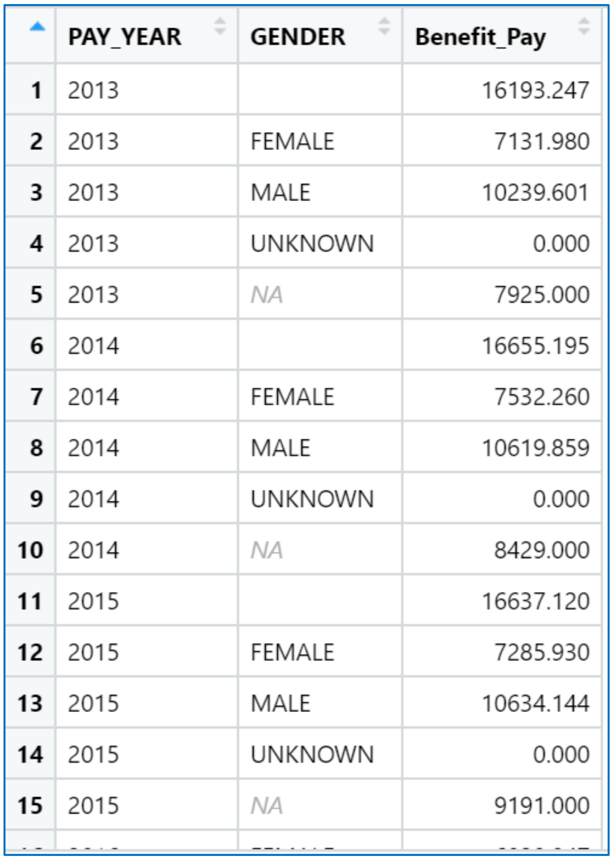
```
> setwd("~/Grad Courses/5250 Visual Analytics/r_script_project")
>
> payroll<-read.csv("payroll.csv", header=T)
>
> usable_columns <- subset(payroll, select=c(PAY_YEAR,
+                                     DEPARTMENT_TITLE,
+                                     MOU_TITLE,
+                                     EMPLOYMENT_TYPE,
+                                     REGULAR_PAY,
+                                     GENDER,
+                                     BENEFIT_PAY,
+                                     OVERTIME_PAY,
+                                     ALL_OTHER_PAY))
>
> library(tidyverse)
>
> usable_columns <- usable_columns %>%
+   mutate(PAY_YEAR=as.character(PAY_YEAR),
+          DEPARTMENT_TITLE = as.character(DEPARTMENT_TITLE),
+          MOU_TITLE = as.character(MOU_TITLE),
+          EMPLOYMENT_TYPE = as.character(EMPLOYMENT_TYPE),
+          GENDER = as.character(GENDER),
+          REGULAR_PAY=as.integer(REGULAR_PAY),
+          BENEFIT_PAY=as.integer(BENEFIT_PAY),
+          OVERTIME_PAY=as.integer(OVERTIME_PAY),
+          ALL_OTHER_PAY=as.integer(ALL_OTHER_PAY))
>
```

Most of the code was discussed in the previous section titled “removing unnecessary columns.” For the conversion, the library function was used to load the tidyverse library. The library first needs to be imported using the install.packages function as follows: install.packages(“tidyverse”). This function is not displayed in the provided code as it has already been installed. The library contains the pipe operator (i.e., %>%) and the mutate function which was used to convert the data types. The usable_columns vector is used to reassign the output of the mutate function. The output of the usable_columns is passed using the pipe operator to the mutate function which then converts the data type of each listed field to the specified data type.

Data Cleaning:

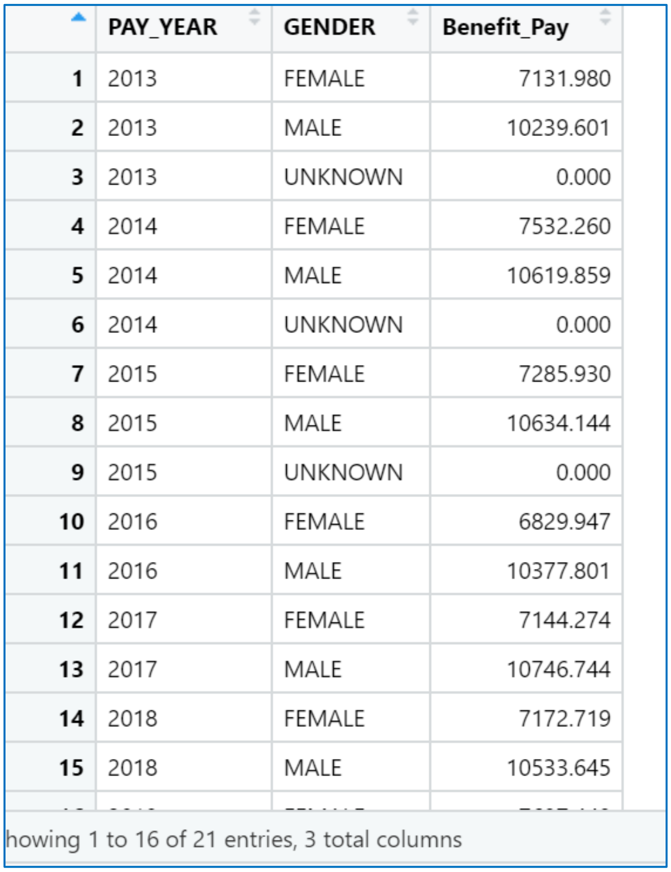
The gender column had string inconsistencies such as unknown and NA values. As a result, for this data cleaning technique, we removed these unwanted values using na.omit() and subset() function.

Before data cleaning:



	PAY_YEAR	GENDER	Benefit_Pay
1	2013		16193.247
2	2013	FEMALE	7131.980
3	2013	MALE	10239.601
4	2013	UNKNOWN	0.000
5	2013	NA	7925.000
6	2014		16655.195
7	2014	FEMALE	7532.260
8	2014	MALE	10619.859
9	2014	UNKNOWN	0.000
10	2014	NA	8429.000
11	2015		16637.120
12	2015	FEMALE	7285.930
13	2015	MALE	10634.144
14	2015	UNKNOWN	0.000
15	2015	NA	9191.000

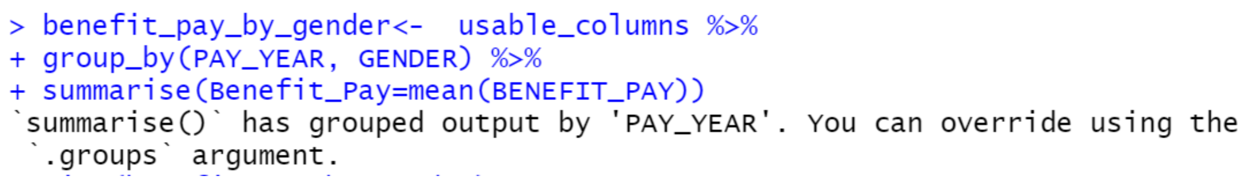
After data cleaning:



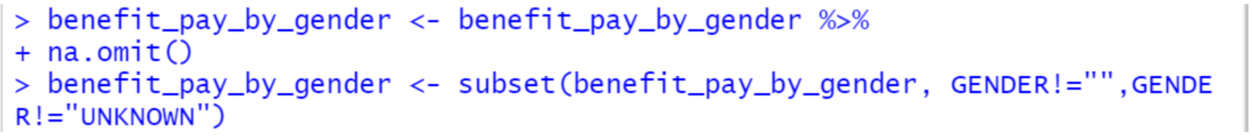
	PAY_YEAR	GENDER	Benefit_Pay
1	2013	FEMALE	7131.980
2	2013	MALE	10239.601
3	2013	UNKNOWN	0.000
4	2014	FEMALE	7532.260
5	2014	MALE	10619.859
6	2014	UNKNOWN	0.000
7	2015	FEMALE	7285.930
8	2015	MALE	10634.144
9	2015	UNKNOWN	0.000
10	2016	FEMALE	6829.947
11	2016	MALE	10377.801
12	2017	FEMALE	7144.274
13	2017	MALE	10746.744
14	2018	FEMALE	7172.719
15	2018	MALE	10533.645

howing 1 to 16 of 21 entries, 3 total columns

The following are screenshots of the code after it is executed in the R-Studio console:



```
> benefit_pay_by_gender<-  usable_columns %>%
+ group_by(PAY_YEAR, GENDER) %>%
+ summarise(Benefit_Pay=mean(BENEFIT_PAY))
`summarise()` has grouped output by 'PAY_YEAR'. You can override using the
  `.groups` argument.
```



```
> benefit_pay_by_gender <- benefit_pay_by_gender %>%
+ na.omit()
> benefit_pay_by_gender <- subset(benefit_pay_by_gender, GENDER!="",GENDE
R!="UNKNOWN")
```

The following code is displayed in the console screenshots provided above:

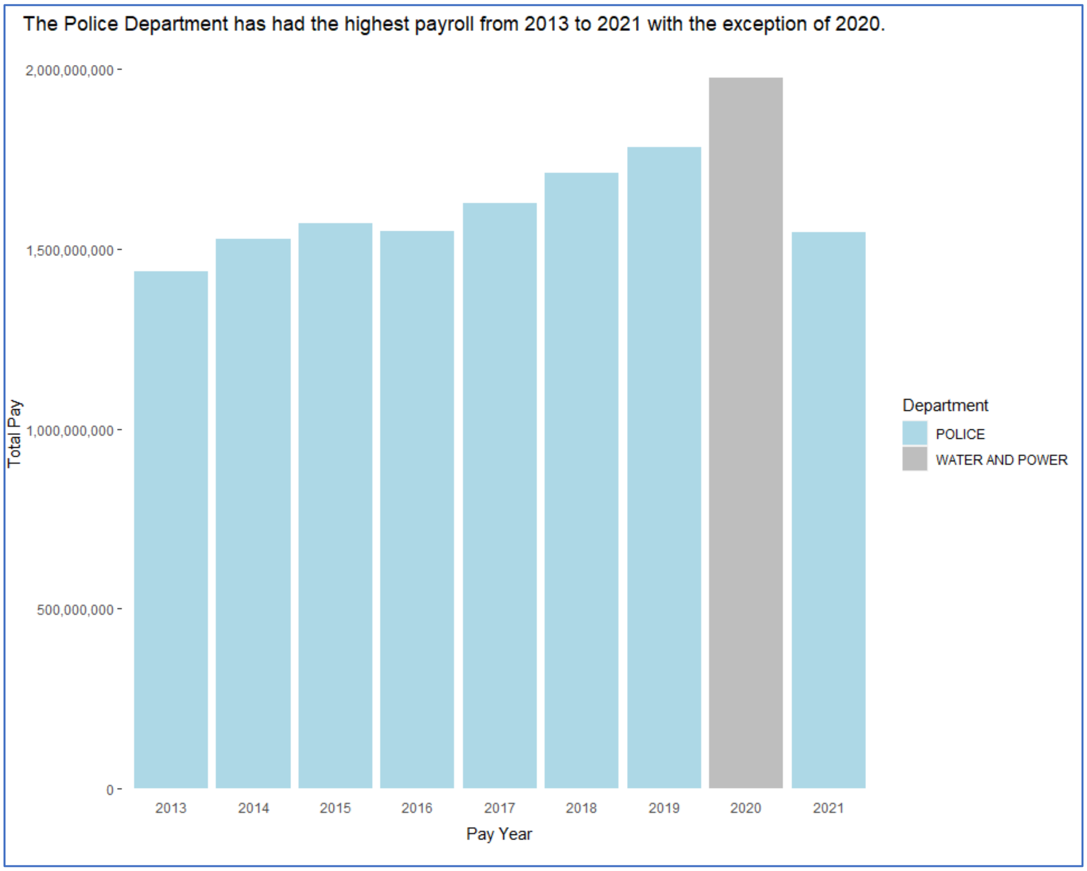
```
> benefit_pay_by_gender<-  usable_columns %>%
+ group_by(PAY_YEAR, GENDER) %>%
+ summarise(Benefit_Pay=mean(BENEFIT_PAY))
> benefit_pay_by_gender <- benefit_pay_by_gender %>%
+ na.omit
> benefit_pay_by_gender <- subset(benefit_pay_by_gender,
GENDER!="",GENDER!="UNKNOWN")
```

The Gender column consisted of NA, Unknown, and blank values. As we are building a visualization using Gender column, it was important to clean the unwanted values and hence in this code, one can see the function to clear them.

The benefit_pay_by_gender is used to group by the gender column and pay year. Na.omit() is one of the functions used to omit all the unnecessary cases from a data frame, matrix, or vector. It is the fastest ways to remove the NA vales from a column. As a result, it returns the object with a list wise deletion of the missing values; while the unknown and blank values were removed using the subset () function. Subset function can be used to select a data from the dataset using certain conditions.

E. Analysis & Visualizations

Which departments had the highest pay each year from 2013 to 2021?



Visualizations Used: bar chart

Functions Used: group_by, summarise, sum, slice, which.max, ggplot, geom_col, theme, scale_y_continuous, scale_x_discrete, ggtitle, and scale_fill_manual. Methods of these functions were also employed which will be discussed in this section.

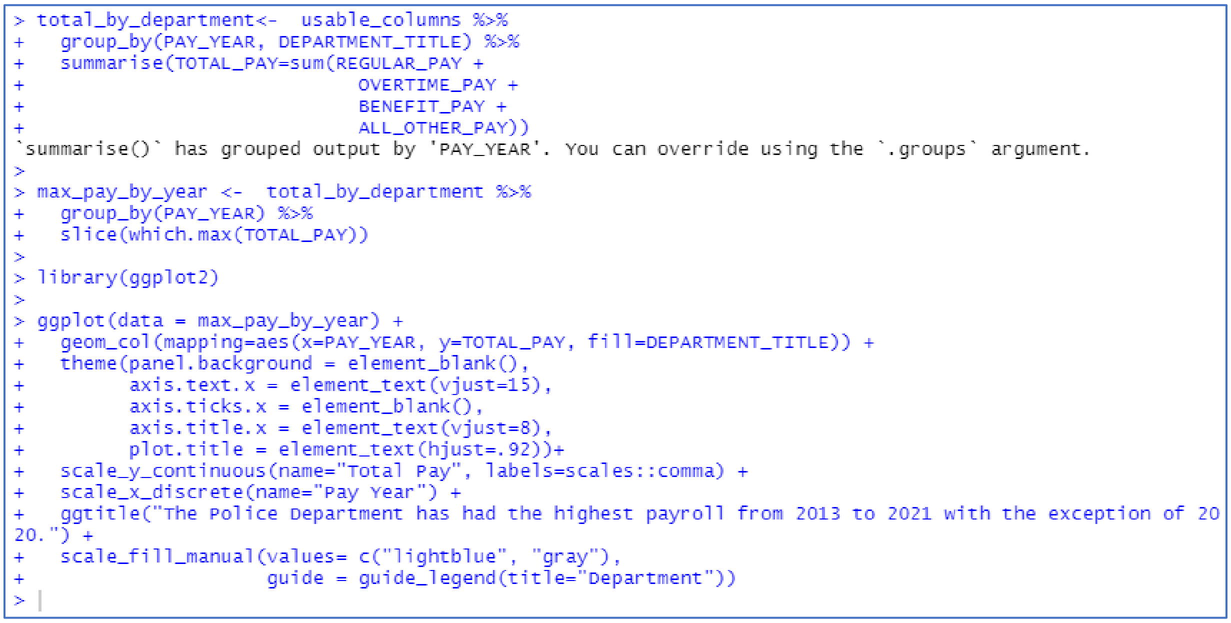
Packages/Library: ggplot2, tidyverse

Analysis/Description:

The Los Angeles City payroll dataset contains pay information from various departments in the city across various years. The dataset separates the type of pay into various fields which are as follows: regular_pay, overtime_pay, benefit_pay, and all_other_pay. For this chart, the values in these fields had to be aggregated into a single field called total_pay. The data, as it stood, did not lend itself to answer the research question in this section as it contains observations at the employee level. As a result, the data were grouped by pay_year and department_title to produce a total_pay field by pay_year and department_title. Once the data were grouped, the department with the most pay for each year had to be identified. The slice function coupled with the max method was employed to extract the max value for each pay_year. The output was a table containing the department with the highest total_pay for each pay_year.

Two out of fifty-three departments had the highest pay from 2013 to 2021. The department with the highest pay for most of those years was the police department. In 2020, the department of water and power reported the highest pay. The pay distributed by the department of water and power was the highest reported from 2013 to 2021. The department of water and power paid a little over 1.9 billion dollars. The police department's payroll was increasing each year from 2013 to 2020. In 2020, the police department paid 55 million less than the department of water and power before experiencing a decrease in 2021. Two months are left in 2021 as of this writing. As a result, the police department will have a higher amount for 2021; however, the amount may not be more than what was reported by the department of water and power in 2020. One possible reason for the department of water and power payroll increase may be due to people working from home due to the ongoing pandemic. As more people work and congregate at home, one can expect water and power usage to increase thus requiring a workforce to support the increase in usage.

Code Screenshot:



```
> total_by_department<-  usable_columns %>%
+   group_by(PAY_YEAR, DEPARTMENT_TITLE) %>%
+   summarise(TOTAL_PAY=sum(REGULAR_PAY +
+                            OVERTIME_PAY +
+                            BENEFIT_PAY +
+                            ALL_OTHER_PAY))
`summarise()` has grouped output by 'PAY_YEAR'. You can override using the `.groups` argument.
>
> max_pay_by_year <-  total_by_department %>%
+   group_by(PAY_YEAR) %>%
+   slice(which.max(TOTAL_PAY))
>
> library(ggplot2)
>
> ggplot(data = max_pay_by_year) +
+   geom_col(mapping=aes(x=PAY_YEAR, y=TOTAL_PAY, fill=DEPARTMENT_TITLE)) +
+   theme(panel.background = element_blank(),
+         axis.text.x = element_text(vjust=15),
+         axis.ticks.x = element_blank(),
+         axis.title.x = element_text(vjust=8),
+         plot.title = element_text(hjust=.92))+
+   scale_y_continuous(name="Total Pay", labels=scales::comma) +
+   scale_x_discrete(name="Pay Year") +
+   ggtitle("The Police Department has had the highest payroll from 2013 to 2021 with the exception of 20
20.") +
+   scale_fill_manual(values= c("lightblue", "gray"),
+                     guide = guide_legend(title="Department"))
>
```

Code Text:

```
> total_by_department<-  usable_columns %>%
+   group_by(PAY_YEAR, DEPARTMENT_TITLE) %>%
+   summarise(TOTAL_PAY=sum(REGULAR_PAY +
+                           OVERTIME_PAY +
+                           BENEFIT_PAY +
+                           ALL_OTHER_PAY))
`summarise()` has grouped output by 'PAY_YEAR'. You can override using the
`.groups` argument.
>
> max_pay_by_year <-  total_by_department %>%
+   group_by(PAY_YEAR) %>%
+   slice(which.max(TOTAL_PAY))
>
> library(ggplot2)
>
> ggplot(data = max_pay_by_year) +
+   geom_col(mapping=aes(x=PAY_YEAR, y=TOTAL_PAY,
fill=DEPARTMENT_TITLE)) +
+   theme(panel.background = element_blank(),
+        axis.text.x = element_text(vjust=15),
+        axis.ticks.x = element_blank(),
+        axis.title.x = element_text(vjust=8),
+        plot.title = element_text(hjust=.92))+
+   scale_y_continuous(name="Total Pay", labels=scales::comma) +
+   scale_x_discrete(name="Pay Year") +
+   ggtitle("The Police Department has had the highest payroll from 2013 to 2021 with
the exception of 2020.") +
+   scale_fill_manual(values= c("lightblue", "gray"),
+                   guide = guide_legend(title="Department"))
```

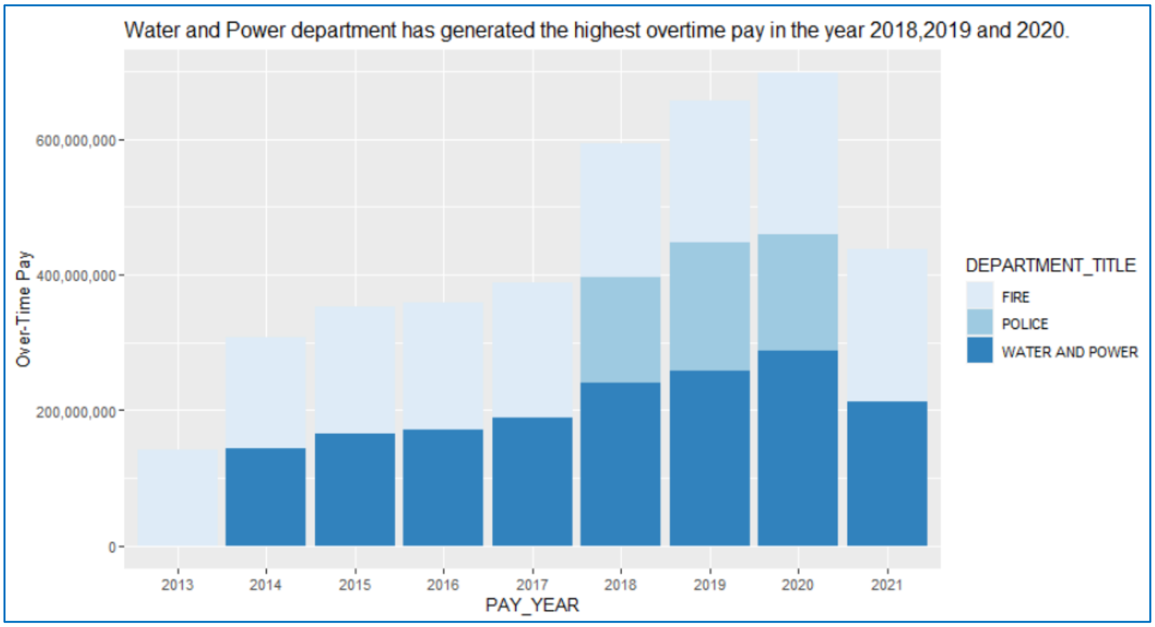
Code Description:

The usable_columns data frame contains the payroll data described in the data cleaning section. In the first line of the code, the usable_columns data frame is passed to the group_by function using the pipe operator. The group_by function groups the data using the pay_year and department_title fields. The grouped data is then passed to the summarise function. The summarise function creates a new row for each grouping created by the group_by function. In this case, the summarise function will aggregate the values contained in the regular_pay, overtime_pay, benefit_pay, and all_other_pay fields. The new field output by the summarise function is titled total_pay. The grouped data is then stored in a data frame titled total_by_department.

Next, the total_by_department data frame is grouped by pay_year, and the output is passed to the slice function. The slice function is employed to retrieve a value at a set position. The position of the max value is determined by the which.max function. Once the position of the max value is determined, it is passed to the slice function which returns it. The output of the slice function is stored in the max_pay_by_year data frame. At this point, the max_pay_by_year data frame contains the highest payroll by pay_year and department_title.

To create the bar graph, the ggplot2 library was utilized. The library function is used to load the package for usage. If the package has not been installed, the install.packages function will need to be executed to download the ggplot2 library. The geom_col function was used instead of the geom_bar function because geom_col allows both the x and y-axis to be specified. The other functions and attributes specified along with the ggplot function are used to adjust elements of the chart such as title, labels, legend, position, et cetera.

Which department generated the most overtime pay as per year?



Type of Visualization: Stacked Bar chart

Functions used: User-defined function, Aggregate, order, head, ggplot, ggtitle, geom_bar, scale_y_continuous, scale_fill_brewer()

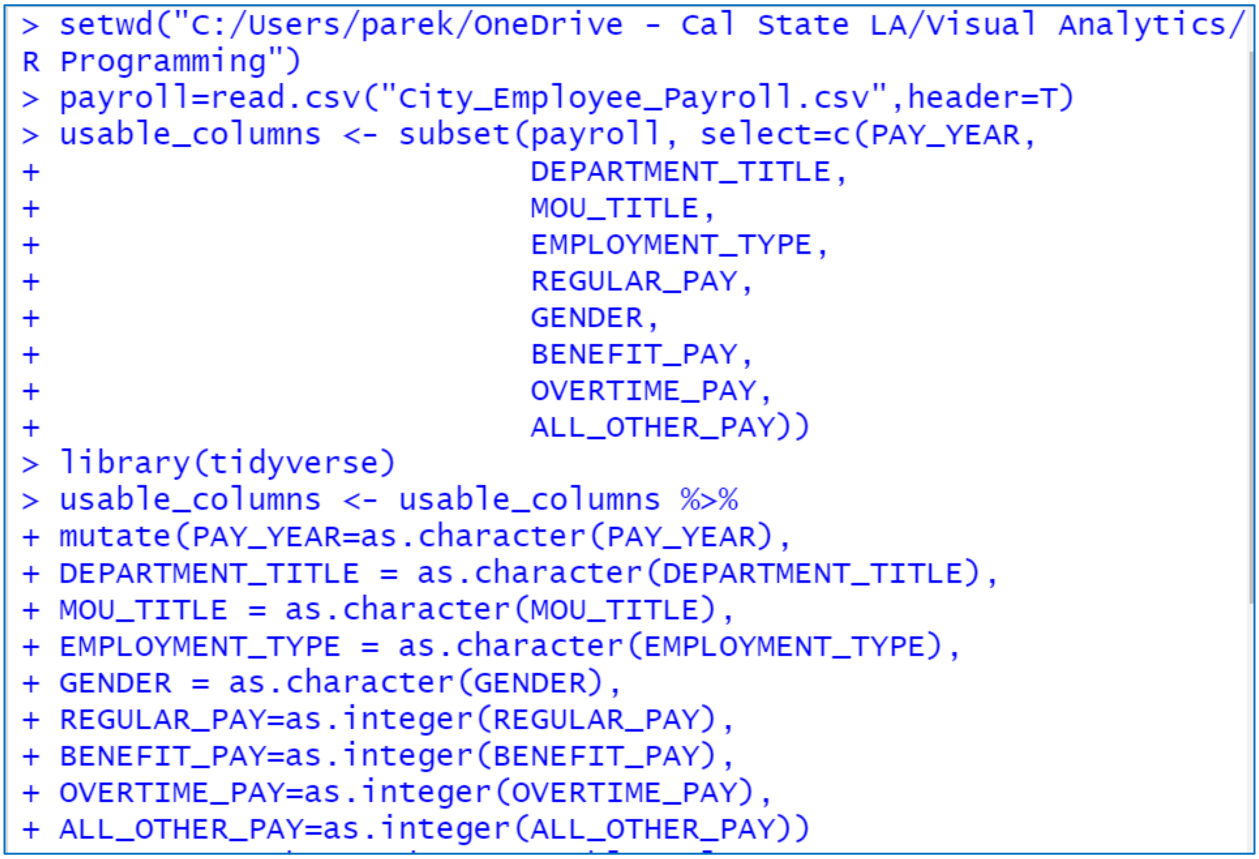
Library: Tidyverse, ggplot2

Los Angeles city consists of various departments ranging from the employees who work full time or part time in airports, water and power departments, zoo services, and public works and so on. The above dataset answers the question of which department generated the most overtime pay as per year. There are more than 30 departments in Los Angeles city giving employment since the year 2013. To find the top 3 departments, one can see the second code provided in this section which consists of the use of order and head. Order is useful in sorting a particular data value as per the descending order to find the departments that generated the maximum overtime pay each year. And we can see that the water and power department has contributed to the higher amount of overtime pay for the year 2020.

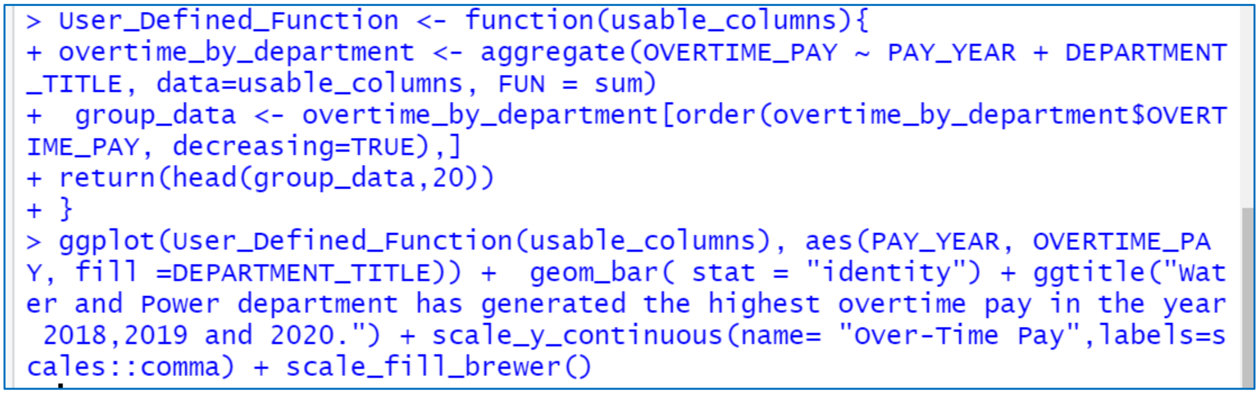
We can see that there has been a recent increase of overtime pay for the police department in the years 2018,2019 and 2020. On reading the news, it can be assumed that during those years LAPD had to cut down the police budget but however for the employees who worked over-time they had to receive their payment for the extra number of hours they mounted which resulted in the rise in the above visual.

The third highest department who was paid the highest is the fire department. The fire department consistently has employees working overtime from 2013 until 2021 with the same number of hours charged.

Code Screenshot:



```
> setwd("C:/Users/parek/OneDrive - Cal State LA/Visual Analytics/
R Programming")
> payroll=read.csv("City_Employee_Payroll.csv",header=T)
> usable_columns <- subset(payroll, select=c(PAY_YEAR,
+                             DEPARTMENT_TITLE,
+                             MOU_TITLE,
+                             EMPLOYMENT_TYPE,
+                             REGULAR_PAY,
+                             GENDER,
+                             BENEFIT_PAY,
+                             OVERTIME_PAY,
+                             ALL_OTHER_PAY))
> library(tidyverse)
> usable_columns <- usable_columns %>%
+ mutate(PAY_YEAR=as.character(PAY_YEAR),
+ DEPARTMENT_TITLE = as.character(DEPARTMENT_TITLE),
+ MOU_TITLE = as.character(MOU_TITLE),
+ EMPLOYMENT_TYPE = as.character(EMPLOYMENT_TYPE),
+ GENDER = as.character(GENDER),
+ REGULAR_PAY=as.integer(REGULAR_PAY),
+ BENEFIT_PAY=as.integer(BENEFIT_PAY),
+ OVERTIME_PAY=as.integer(OVERTIME_PAY),
+ ALL_OTHER_PAY=as.integer(ALL_OTHER_PAY))
```



```
> User_Defined_Function <- function(usable_columns){
+ overtime_by_department <- aggregate(OVERTIME_PAY ~ PAY_YEAR + DEPARTMENT
_TITLE, data=usable_columns, FUN = sum)
+  group_data <- overtime_by_department[order(overtime_by_department$OVERT
IME_PAY, decreasing=TRUE),]
+ return(head(group_data,20))
+ }
> ggplot(User_Defined_Function(usable_columns), aes(PAY_YEAR, OVERTIME_PA
Y, fill =DEPARTMENT_TITLE)) +  geom_bar( stat = "identity") + ggtitle("Wat
er and Power department has generated the highest overtime pay in the year
 2018,2019 and 2020.") + scale_y_continuous(name= "Over-Time Pay",labels=s
cales::comma) + scale_fill_brewer()
```

Code Text:

```
> setwd("C:/Users/parek/OneDrive - Cal State LA/Visual Analytics/R Programming")

> payroll=read.csv("City_Employee_Payroll.csv",header=T)

> usable_columns <- subset(payroll, select=c(PAY_YEAR,
+                                   DEPARTMENT_TITLE,
+                                   MOU_TITLE,
+                                   EMPLOYMENT_TYPE,
+                                    REGULAR_PAY,
+                                    GENDER,
+                                    BENEFIT_PAY,
+                                    OVERTIME_PAY,
+                                    ALL_OTHER_PAY))

>library(tidyverse)

> usable_columns <- usable_columns %>%

+ mutate(PAY_YEAR=as.character(PAY_YEAR),

+ DEPARTMENT_TITLE = as.character(DEPARTMENT_TITLE),

+ MOU_TITLE = as.character(MOU_TITLE),

+ EMPLOYMENT_TYPE = as.character(EMPLOYMENT_TYPE),

+ GENDER = as.character(GENDER),

+ REGULAR_PAY=as.integer(REGULAR_PAY),

+ BENEFIT_PAY=as.integer(BENEFIT_PAY),

+ OVERTIME_PAY=as.integer(OVERTIME_PAY),

+ ALL_OTHER_PAY=as.integer(ALL_OTHER_PAY))

> User_Defined_Function <- function(usable_columns){

+ overtime_by_department <- aggregate(OVERTIME_PAY ~ PAY_YEAR +

DEPARTMENT_TITLE, data=usable_columns, FUN = sum)

+  group_data <-

overtime_by_department[order(overtime_by_department$OVERTIME_PAY,

decreasing=TRUE),]

+ return(head(group_data,20))
+ }
> ggplot(User_Defined_Function(usable_columns), aes(PAY_YEAR, OVERTIME_PAY, fill
=DEPARTMENT_TITLE)) +  geom_bar( stat = "identity") + ggtitle("Water and Power
department has generated the highest overtime pay in the year 2018,2019 and 2020.") +
scale_y_continuous(name= "Over-Time Pay",labels=scales::comma) + scale_fill_brewer()
```

Code description:

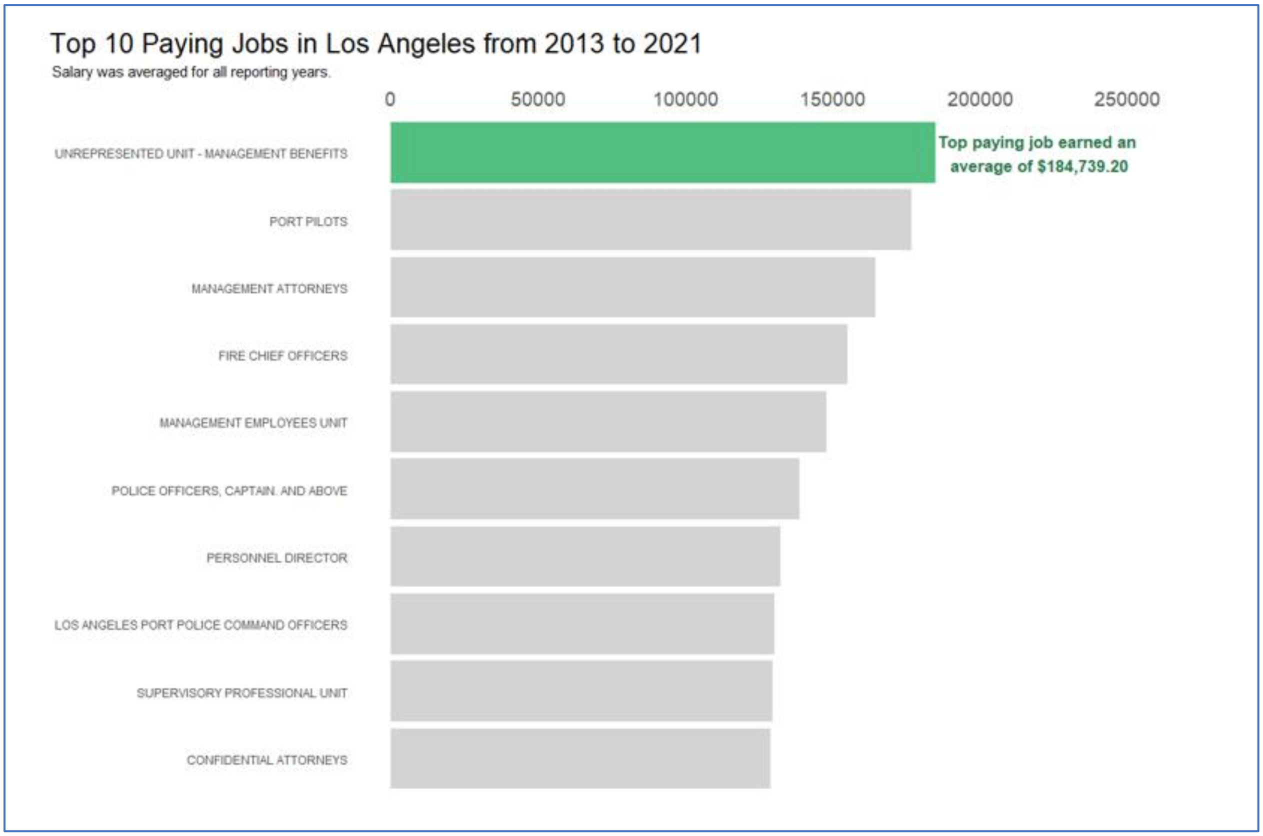
The above visual was built using the ggplot2 package. It is a package in R dedicated just for visualization. It can be used to improve the quality and aesthetics of a graph, making it easier for the target audience to read. The user-defined function is specific to what a user requires and once created it can be used like the built-in functions we used in the ggplot2 code. The function of aggregate is used to split the data into subsets and return the results in a group by form. The dataset consists of different departments working either part-time or full-time by each pay year.

The graph used in the visual is the basic stacked bar graph with the geom_bar. This plot displays the stacked sum for each group by we used for the department_title. To use geom_bar, we had to add the values for x and y axis to the aes and choose a variable which can be used in the fill field for the stack graph.

Stat=identity is used to create a stacked bar plot for multiple values. We wanted the same color hue for different stacks and hence a pre-defined color palette is used by adding scale_fill_brewer() to the code.

The ggtitle is included to add a sentence at the top of the visual. The sum of overtime pay values were more than thousands and hence to read in a numeric format with commas we used the function of scale_y_continuous where values of the y axis can be written in commas making it efficient to read for the users.

On average, which were the top 10 full-time positions that paid the most from 2013 to 2021?



Visualization Used: bar graph

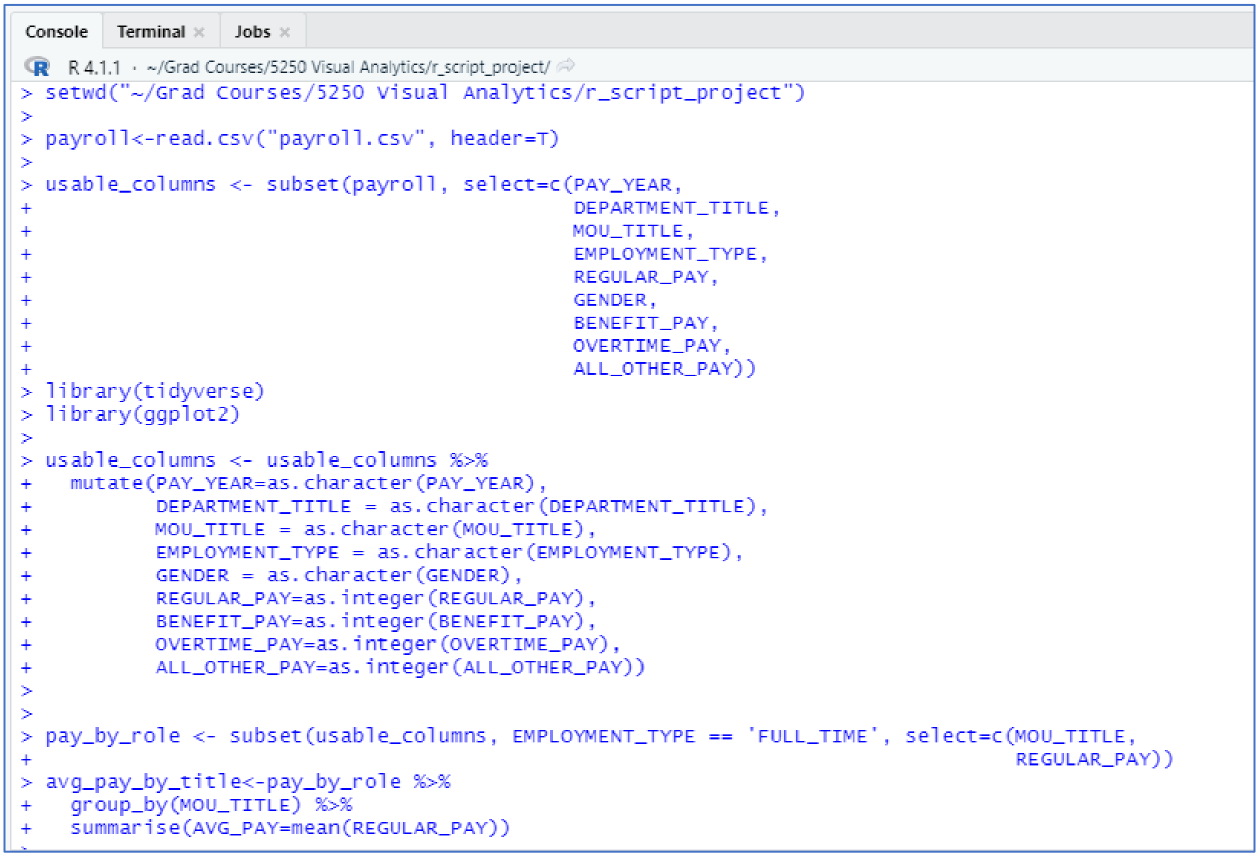
Functions used: setwd, read.csv, subset, mutate, group_by, summarise, head, arrange, ggplot, geom_col, reorder, coord_flip, scale_fill_manual, theme, scale_y_continous, ggtitle, annotate

Packages/Library: ggplot2, tinyverse

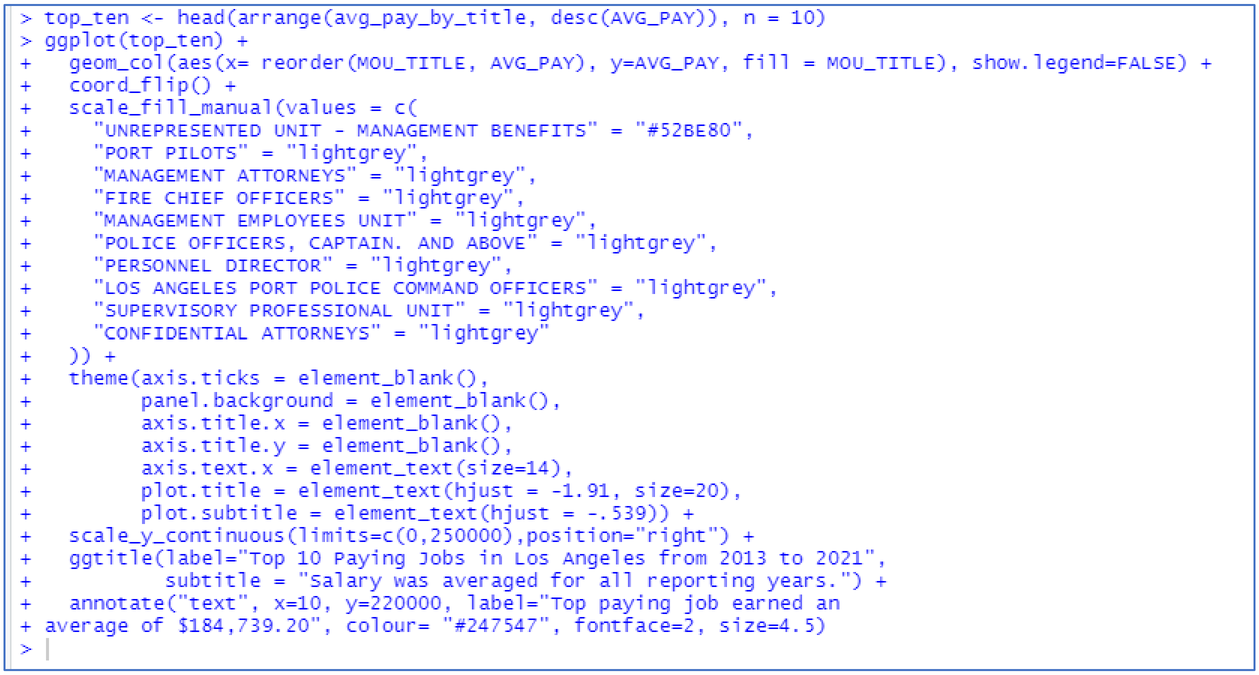
Analysis/Description:

The top ten paying positions earned an average of at least $120,0000. Most of the positions are in management. Collectively, the positions are either management, attorneys, or port personnel. The top paying position of the ten is the position titled “unrepresented unit – management benefits.” The position title does not provide much information regarding the type of role. This position earned an average of $184,739.20 from 2013 to 2021. The second highest position is the “port pilots” which earned an average of $176,773.70. The difference between the top first and second position is $7,965.50. The lowest paying position of the top ten is the “confidential attorneys role” which earned an average of $128,681.00.

Code Screenshot:


Console
Terminal
Jobs
R 4.1.1 · ~/Grad Courses/5250 Visual Analytics/r_script_project/

```
> setwd("~/Grad Courses/5250 Visual Analytics/r_script_project")
> 
> payroll<-read.csv("payroll.csv", header=T)
> 
> usable_columns <- subset(payroll, select=c(PAY_YEAR,
+                                             DEPARTMENT_TITLE,
+                                             MOU_TITLE,
+                                             EMPLOYMENT_TYPE,
+                                             REGULAR_PAY,
+                                             GENDER,
+                                             BENEFIT_PAY,
+                                             OVERTIME_PAY,
+                                             ALL_OTHER_PAY))
> library(tidyverse)
> library(ggplot2)
> 
> usable_columns <- usable_columns %>%
+   mutate(PAY_YEAR=as.character(PAY_YEAR),
+          DEPARTMENT_TITLE = as.character(DEPARTMENT_TITLE),
+          MOU_TITLE = as.character(MOU_TITLE),
+          EMPLOYMENT_TYPE = as.character(EMPLOYMENT_TYPE),
+          GENDER = as.character(GENDER),
+          REGULAR_PAY=as.integer(REGULAR_PAY),
+          BENEFIT_PAY=as.integer(BENEFIT_PAY),
+          OVERTIME_PAY=as.integer(OVERTIME_PAY),
+          ALL_OTHER_PAY=as.integer(ALL_OTHER_PAY))
> 
> 
> pay_by_role <- subset(usable_columns, EMPLOYMENT_TYPE == 'FULL_TIME', select=c(MOU_TITLE,
+                                                                                REGULAR_PAY))
> avg_pay_by_title<-pay_by_role %>%
+   group_by(MOU_TITLE) %>%
+   summarise(AVG_PAY=mean(REGULAR_PAY))
```



```
> top_ten <- head(arrange(avg_pay_by_title, desc(AVG_PAY)), n = 10)
> ggplot(top_ten) +
+   geom_col(aes(x= reorder(MOU_TITLE, AVG_PAY), y=AVG_PAY, fill = MOU_TITLE), show.legend=FALSE) +
+   coord_flip() +
+   scale_fill_manual(values = c(
+     "UNREPRESENTED UNIT - MANAGEMENT BENEFITS" = "#52BE80",
+     "PORT PILOTS" = "lightgrey",
+     "MANAGEMENT ATTORNEYS" = "lightgrey",
+     "FIRE CHIEF OFFICERS" = "lightgrey",
+     "MANAGEMENT EMPLOYEES UNIT" = "lightgrey",
+     "POLICE OFFICERS, CAPTAIN. AND ABOVE" = "lightgrey",
+     "PERSONNEL DIRECTOR" = "lightgrey",
+     "LOS ANGELES PORT POLICE COMMAND OFFICERS" = "lightgrey",
+     "SUPERVISORY PROFESSIONAL UNIT" = "lightgrey",
+     "CONFIDENTIAL ATTORNEYS" = "lightgrey"
+   )) +
+   theme(axis.ticks = element_blank(),
+         panel.background = element_blank(),
+         axis.title.x = element_blank(),
+         axis.title.y = element_blank(),
+         axis.text.x = element_text(size=14),
+         plot.title = element_text(hjust = -1.91, size=20),
+         plot.subtitle = element_text(hjust = -.539)) +
+   scale_y_continuous(limits=c(0,250000),position="right") +
+   ggtitle(label="Top 10 Paying Jobs in Los Angeles from 2013 to 2021",
+           subtitle = "Salary was averaged for all reporting years.") +
+   annotate("text", x=10, y=220000, label="Top paying job earned an
+ average of $184,739.20", colour= "#247547", fontface=2, size=4.5)
> 
```

Code Text:

```
> setwd("~/Grad Courses/5250 Visual Analytics/r_script_project")
>
> payroll<-read.csv("payroll.csv", header=T)
>
> usable_columns <- subset(payroll, select=c(PAY_YEAR,
+                                  DEPARTMENT_TITLE,
+                                  MOU_TITLE,
+                                  EMPLOYMENT_TYPE,
+                                  REGULAR_PAY,
+                                  GENDER,
+                                  BENEFIT_PAY,
+                                  OVERTIME_PAY,
+                                  ALL_OTHER_PAY))
> library(tidyverse)
> library(ggplot2)
>
> usable_columns <- usable_columns %>%
+   mutate(PAY_YEAR=as.character(PAY_YEAR),
+          DEPARTMENT_TITLE = as.character(DEPARTMENT_TITLE),
+          MOU_TITLE = as.character(MOU_TITLE),
+          EMPLOYMENT_TYPE = as.character(EMPLOYMENT_TYPE),
+          GENDER = as.character(GENDER),
+          REGULAR_PAY=as.integer(REGULAR_PAY),
+          BENEFIT_PAY=as.integer(BENEFIT_PAY),
+         OVERTIME_PAY=as.integer(OVERTIME_PAY),
+         ALL_OTHER_PAY=as.integer(ALL_OTHER_PAY))
>
>
> pay_by_role <- subset(usable_columns, EMPLOYMENT_TYPE == 'FULL_TIME',
select=c(MOU_TITLE,
+                                                            REGULAR_PAY))
> avg_pay_by_title<-pay_by_role %>%
+   group_by(MOU_TITLE) %>%
+   summarise(AVG_PAY=mean(REGULAR_PAY))
>
> top_ten <- head(arrange(avg_pay_by_title, desc(AVG_PAY)), n = 10)
> ggplot(top_ten) +
+   geom_col(aes(x= reorder(MOU_TITLE, AVG_PAY), y=AVG_PAY, fill = MOU_TITLE),
show.legend=FALSE) +
+   coord_flip() +
+   scale_fill_manual(values = c(
+     "UNREPRESENTED UNIT - MANAGEMENT BENEFITS" = "#52BE80",
+     "PORT PILOTS" = "lightgrey",
+     "MANAGEMENT ATTORNEYS" = "lightgrey",
+     "FIRE CHIEF OFFICERS" = "lightgrey",
+     "MANAGEMENT EMPLOYEES UNIT" = "lightgrey",
+     "POLICE OFFICERS, CAPTAIN. AND ABOVE" = "lightgrey",
+     "PERSONNEL DIRECTOR" = "lightgrey",
+     "LOS ANGELES PORT POLICE COMMAND OFFICERS" = "lightgrey",
+     "SUPERVISORY PROFESSIONAL UNIT" = "lightgrey",
+     "CONFIDENTIAL ATTORNEYS" = "lightgrey"
+   )) +
+   theme(axis.ticks = element_blank(),
+         panel.background = element_blank(),
+         axis.title.x = element_blank(),
+         axis.title.y = element_blank(),
+         axis.text.x = element_text(size=14),
+         plot.title = element_text(hjust = -1.91, size=20),
+         plot.subtitle = element_text(hjust = -.539)) +
+   scale_y_continuous(limits=c(0,250000),position="right") +
+   ggtitle(label="Top 10 Paying Jobs in Los Angeles from 2013 to 2021",
+           subtitle = "Salary was averaged for all reporting years.") +
+   annotate("text", x=10, y=220000, label="Top paying job earned an
+ average of $184,739.20", colour= "#247547", fontface=2, size=4.5)
>
```

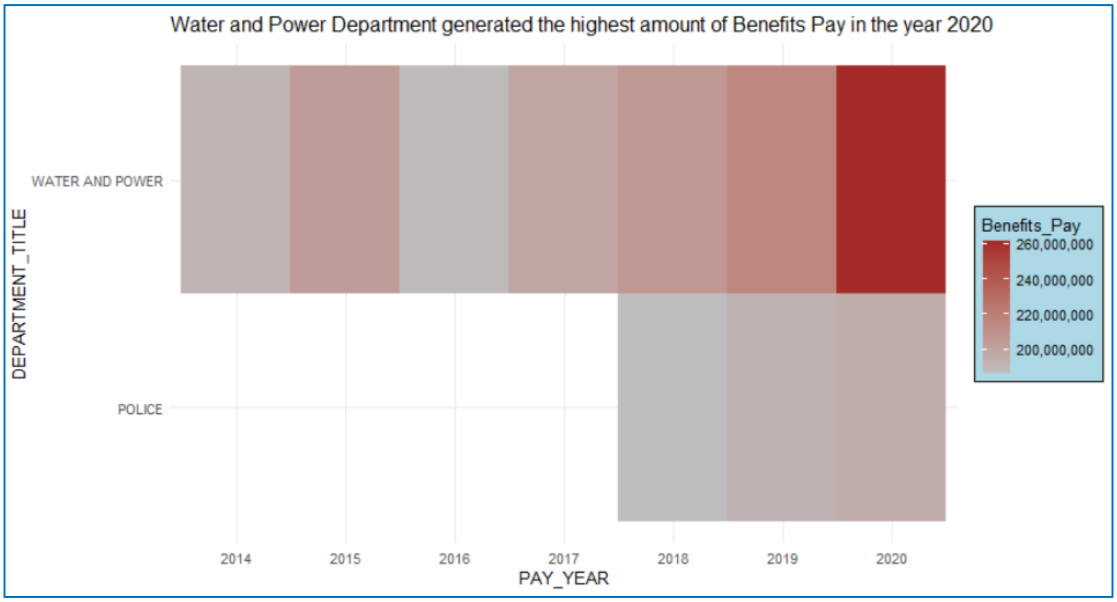
Code Description:

The setwd function is first executed to set the working directory for the current session. The directory contains the dataset used in this project. Then, the data is added to the payroll data frame by using the read.csv function. The subset function is then utilized to return the columns that are used for this report. The graph and data type conversion are possible using the tidyverse and ggplot2 libraries. The packages have already been installed; as a result, they are loaded using the library function. After the libraries are loaded, the fields used for this project are mutated to the appropriate data type using the mutate function. The mutated fields are reassigned to the usable_columns data frame. The subset field is used again to return the mou_title and regular_pay fields from the usable_columns data frame and store it in a new data frame titled "pay_by_role." The data frame returned by the subset function is filtered by the full_time employment type.

Once the data is cleaned, the data frame is grouped by the mou_title using the group_by function. For the grouped data, the regular_pay field is averaged and stored in a field titled "avg_pay." The grouped data is stored in a data frame called "avg_pay_by_title." The grouped data is then sorted using the avg_pay field in descending order. The sorting is performed by the arrange function. The head function is then used to return the top ten records. The ten records are stored in a data frame called "top_ten."

The top_ten data frame is used to create the bar graph. The ggplot function is used to specify the data frame that will be used to generate the graph. Then, the geom_col is used to generate the bar graph. The geom_col is used in lieu of the geom_bar function because the research question required that an x and y-axis be specified which is not possible with the geom_bar function. The geom_col is also used to hide the legend and reorder the bars from highest to lowest. Next, the coord_flip function is used to display the bars horizontally. The scale_fill_manual function is used to specify the color for each bar. All bars are filled light grey except for the top record. Then, the theme function is used to adjust various elements on the graph. The function is employed to remove the panel background, remove axis ticks, remove axis titles, adjust the size of the x-axis labels, and reposition the chart title and subtitle. The scale_y_continuous function is used to adjust the range of the x-axis scale and the position of the x-axis. The x-axis is repositioned to the top of the bar graph. This is followed by the ggtitle function which is used to add a title and subtitle to the bar graph. Lastly, the annotate function is declared to add an annotation.

Which department generated the highest number of benefits pay each year?



Type of Visualization: Heat map

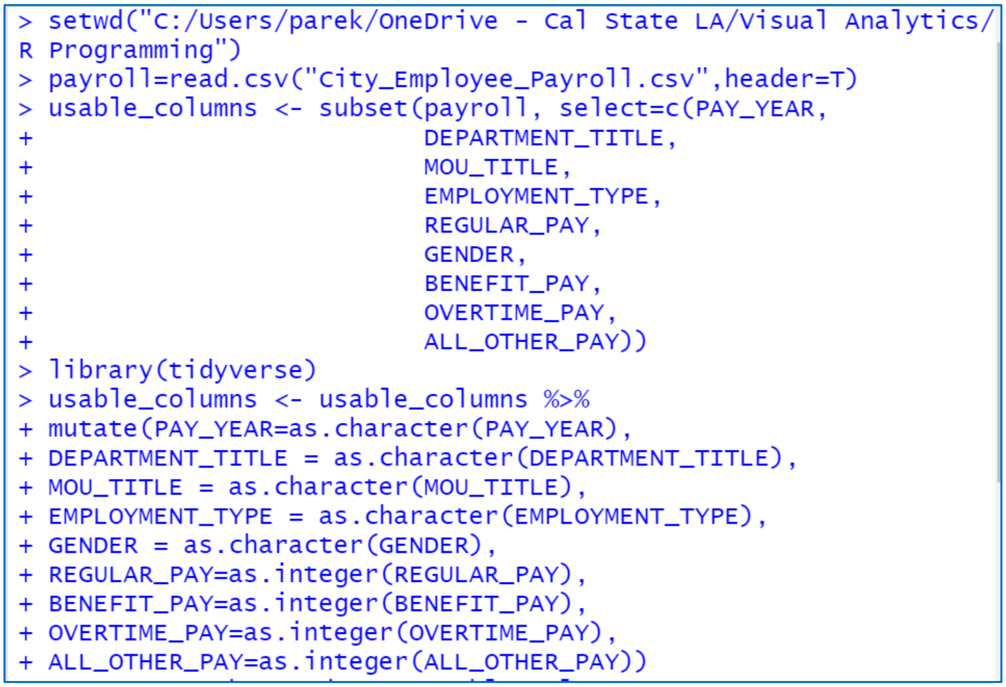
Functions used: Summarise, Filter, order, head, ggplot, ggtitle, geom_tile, scale_fill_gradient()

Library: tidyverse, ggplot2, scales

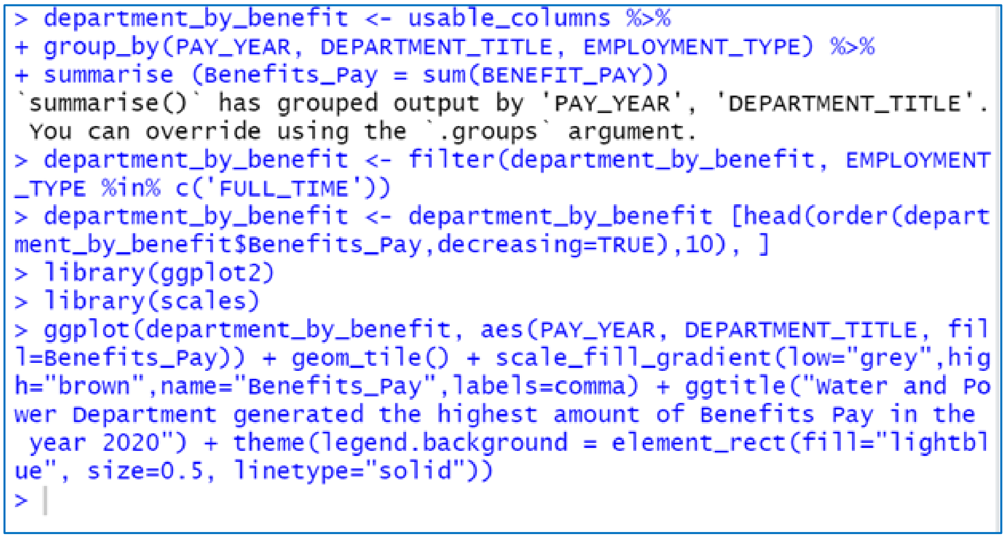
Analysis and Description:

Benefits pay are much more important to employees than the regular pay as it helps with employee satisfaction, and it gives a better experience for the employees working under the organization. Health, retirement plan, transportation pay and the other benefits not only work to motivate the employees to work harder for the organization, but it also encourages a form of economic security for the employees. The above visual tells us about the top departments under the full-time employment generating the benefits pay. And it can be inferred that water and power department has given the highest benefit pay on average to their full-time employees in the year 2020. LADWP department serves benefits such as vacation, sick leave, holidays, other paid leave, retirement, other government service (OGS) buyback, deferred compensation, flexible work schedules, tuition reimbursement and rideshare incentives. Other than LADWP, police department is the other department who provides great number of benefits to their employees. From the year 2018, LAPD departments' budget for their employees has increased and thus in the visual we can see the increasing number of benefits years after year.

Code Screenshot:



```
> setwd("C:/Users/parek/OneDrive - Cal State LA/Visual Analytics/
R Programming")
> payroll=read.csv("City_Employee_Payroll.csv",header=T)
> usable_columns <- subset(payroll, select=c(PAY_YEAR,
+                            DEPARTMENT_TITLE,
+                            MOU_TITLE,
+                            EMPLOYMENT_TYPE,
+                            REGULAR_PAY,
+                            GENDER,
+                            BENEFIT_PAY,
+                            OVERTIME_PAY,
+                            ALL_OTHER_PAY))
> library(tidyverse)
> usable_columns <- usable_columns %>%
+ mutate(PAY_YEAR=as.character(PAY_YEAR),
+ DEPARTMENT_TITLE = as.character(DEPARTMENT_TITLE),
+ MOU_TITLE = as.character(MOU_TITLE),
+ EMPLOYMENT_TYPE = as.character(EMPLOYMENT_TYPE),
+ GENDER = as.character(GENDER),
+ REGULAR_PAY=as.integer(REGULAR_PAY),
+ BENEFIT_PAY=as.integer(BENEFIT_PAY),
+ OVERTIME_PAY=as.integer(OVERTIME_PAY),
+ ALL_OTHER_PAY=as.integer(ALL_OTHER_PAY))
```



```
> department_by_benefit <- usable_columns %>%
+ group_by(PAY_YEAR, DEPARTMENT_TITLE, EMPLOYMENT_TYPE) %>%
+ summarise (Benefits_Pay = sum(BENEFIT_PAY))
`summarise()` has grouped output by 'PAY_YEAR', 'DEPARTMENT_TITLE'.
 You can override using the `.groups` argument.
> department_by_benefit <- filter(department_by_benefit, EMPLOYMENT
_TYPE %in% c('FULL_TIME'))
> department_by_benefit <- department_by_benefit [head(order(depart
ment_by_benefit$Benefits_Pay,decreasing=TRUE),10), ]
> library(ggplot2)
> library(scales)
> ggplot(department_by_benefit, aes(PAY_YEAR, DEPARTMENT_TITLE, fil
l=Benefits_Pay)) + geom_tile() + scale_fill_gradient(low="grey",hig
h="brown",name="Benefits_Pay",labels=comma) + ggtitle("Water and Po
wer Department generated the highest amount of Benefits Pay in the
 year 2020") + theme(legend.background = element_rect(fill="lightbl
ue", size=0.5, linetype="solid"))
> |
```

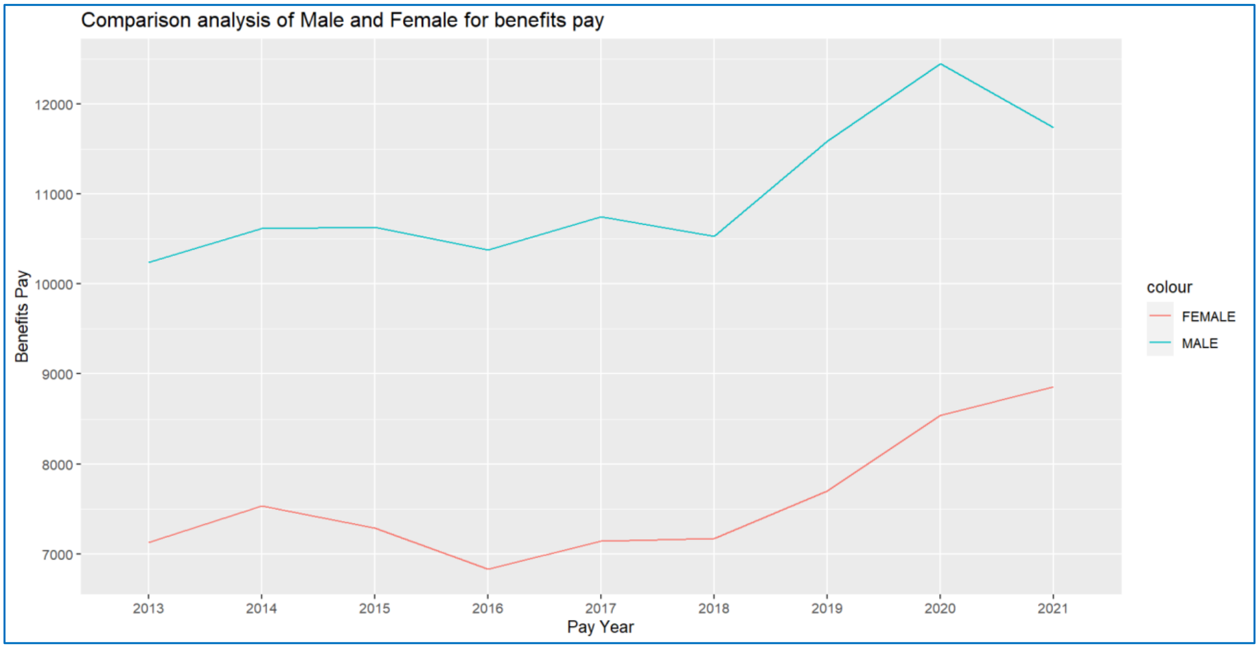
Code Text:

```
> setwd("C:/Users/parek/OneDrive - Cal State LA/Visual Analytics/R Programming")

> payroll=read.csv("City_Employee_Payroll.csv",header=T)

> usable_columns <- subset(payroll, select=c(PAY_YEAR,
+                            DEPARTMENT_TITLE,
+                                          MOU_TITLE,
+                                          EMPLOYMENT_TYPE,
+                                           REGULAR_PAY,
+                                           GENDER,
+                                           BENEFIT_PAY,
+                                           OVERTIME_PAY,
+                                           ALL_OTHER_PAY))

>library(tidyverse)

> usable_columns <- usable_columns %>%

+ mutate(PAY_YEAR=as.character(PAY_YEAR),

+ DEPARTMENT_TITLE = as.character(DEPARTMENT_TITLE),

+ MOU_TITLE = as.character(MOU_TITLE),

+ EMPLOYMENT_TYPE = as.character(EMPLOYMENT_TYPE),

+ GENDER = as.character(GENDER),

+ REGULAR_PAY=as.integer(REGULAR_PAY),

+ BENEFIT_PAY=as.integer(BENEFIT_PAY),

+ OVERTIME_PAY=as.integer(OVERTIME_PAY),

+ ALL_OTHER_PAY=as.integer(ALL_OTHER_PAY))

> department_by_benefit <- usable_columns %>%

+ group_by(PAY_YEAR, DEPARTMENT_TITLE, EMPLOYMENT_TYPE) %>%

+ summarise (Benefits_Pay = sum(BENEFIT_PAY))

> department_by_benefit <- filter(department_by_benefit, EMPLOYMENT_TYPE %in%

c('FULL_TIME'))

> department_by_benefit <- department_by_benefit

[head(order(department_by_benefit$Benefits_Pay,decreasing=TRUE),10), ]

library(ggplot2)
library(scales)

> ggplot(department_by_benefit, aes(PAY_YEAR, DEPARTMENT_TITLE,

fill=Benefits_Pay)) + geom_tile() +

scale_fill_gradient(low="grey",high="brown",name="Benefits_Pay",labels=comma) +

ggtitle("Water and Power Department generated the highest amount of Benefits Pay in the

year 2020") + theme(legend.background = element_rect(fill="lightblue", size=0.5,

linetype="solid"))
```

Code Description:

In the dataset, there are pay_years from 2013 until 2020 have different departments working in full-time or part-time. Group by is used here to have the same values under different rows in identical groups format. The summarize function adds up the benefit pay based on different groups. As for the visualization, we wanted to exclusively work on full-time employment status, filter is used to subset a data frame retaining all the rows that satisfy the condition. To get the top department as per the benefit pay, head and order is used to rank the top rows. An additional library of scales is used for converting the scientific E notation to comma-based values. To have a clear legend, theme background is used to set the legend according as per the position and requirement.

Which gender were given the highest amount of benefits each year?



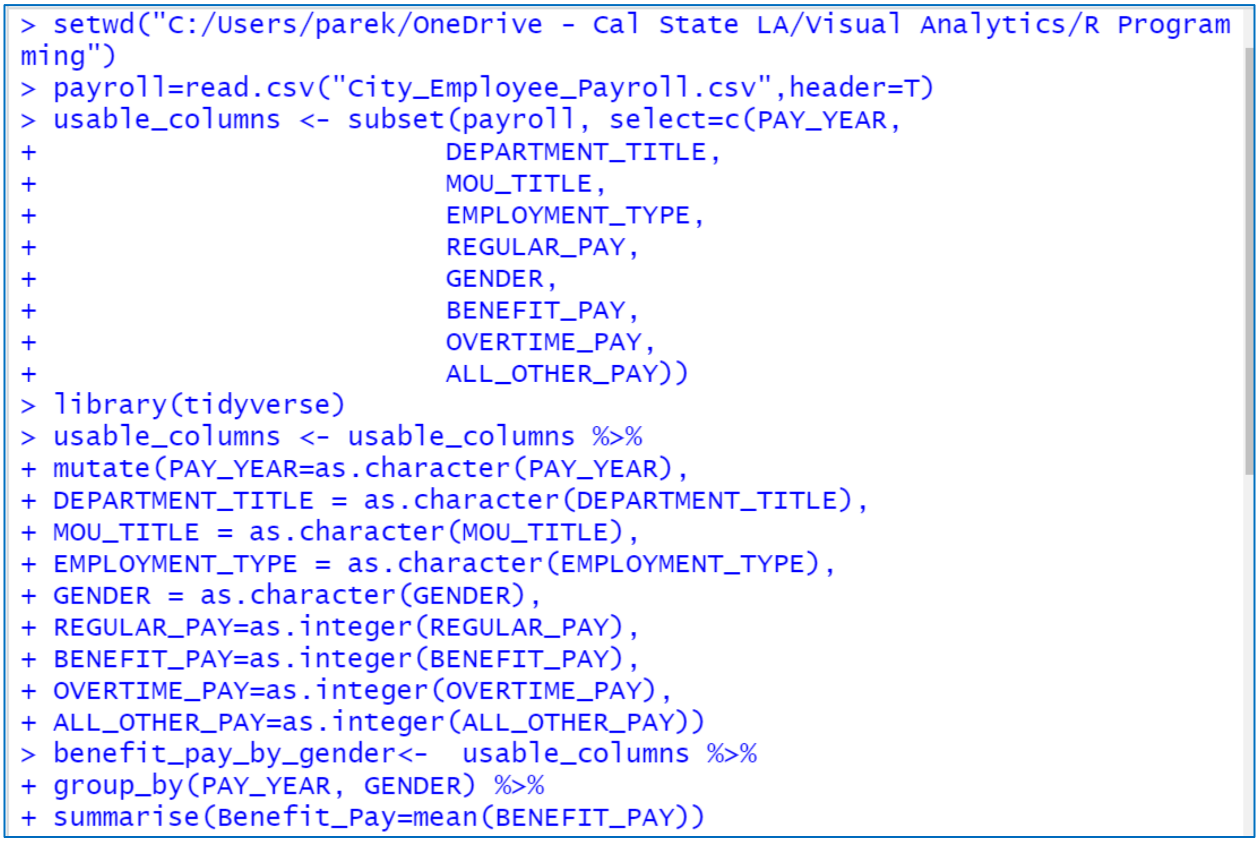
Type of Visualization: Line graph

Functions used: Summarise, subset, pivot_wider, ggplot, ggtitle, geom_line

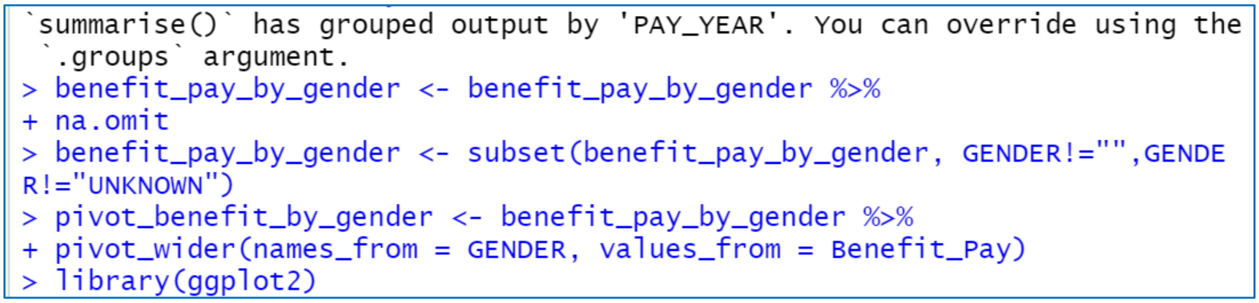
Library: Tidyverse, ggplot2

The graph shows here about which gender gets more benefits pay in their respective departments. We can see that the blue line is much higher than the red one telling us that males are given more benefits compared to females. I feel there would be many reasons responsible for this analysis. Like some departments such as water and power, Fire department require a lot of manpower and labor and hence the distribution of males might we more compared to other females. We can see there is a great increase in benefits pay for both males and females from the year 2019 to the year 2020 and it could be because of the pandemic, department would be offering more benefits in terms of medical and financial help to their employees. Whereas the lowest was in the month of 2016, as the budget distributed during that time was a bit low compared to other years. However, from 2020 to 2021 there is a drop of benefits pays in males compared to females. To conclude, benefits pay are an important factor in any department irrespective of gender, as not only it helps to retain and attract wide talent to your organization but shows your employees that you have not just invested in their health, but also in the future.

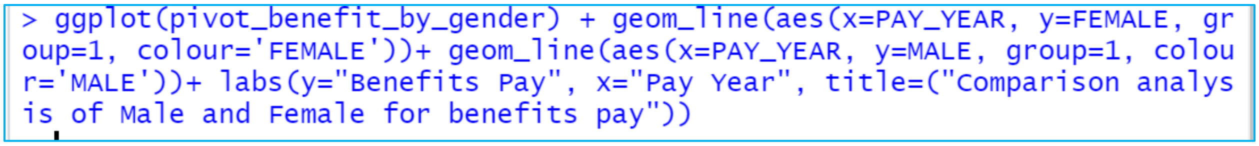
Code Screenshot:



```
> setwd("C:/Users/parek/OneDrive - Cal State LA/Visual Analytics/R Program
ming")
> payroll=read.csv("City_Employee_Payroll.csv",header=T)
> usable_columns <- subset(payroll, select=c(PAY_YEAR,
+                          DEPARTMENT_TITLE,
+                          MOU_TITLE,
+                          EMPLOYMENT_TYPE,
+                          REGULAR_PAY,
+                          GENDER,
+                          BENEFIT_PAY,
+                          OVERTIME_PAY,
+                          ALL_OTHER_PAY))
> library(tidyverse)
> usable_columns <- usable_columns %>%
+ mutate(PAY_YEAR=as.character(PAY_YEAR),
+ DEPARTMENT_TITLE = as.character(DEPARTMENT_TITLE),
+ MOU_TITLE = as.character(MOU_TITLE),
+ EMPLOYMENT_TYPE = as.character(EMPLOYMENT_TYPE),
+ GENDER = as.character(GENDER),
+ REGULAR_PAY=as.integer(REGULAR_PAY),
+ BENEFIT_PAY=as.integer(BENEFIT_PAY),
+ OVERTIME_PAY=as.integer(OVERTIME_PAY),
+ ALL_OTHER_PAY=as.integer(ALL_OTHER_PAY))
> benefit_pay_by_gender<-  usable_columns %>%
+ group_by(PAY_YEAR, GENDER) %>%
+ summarise(Benefit_Pay=mean(BENEFIT_PAY))
```



```
`summarise()` has grouped output by 'PAY_YEAR'. You can override using the
 `.groups` argument.
> benefit_pay_by_gender <- benefit_pay_by_gender %>%
+ na.omit
> benefit_pay_by_gender <- subset(benefit_pay_by_gender, GENDER!="",GENDE
R!="UNKNOWN")
> pivot_benefit_by_gender <- benefit_pay_by_gender %>%
+ pivot_wider(names_from = GENDER, values_from = Benefit_Pay)
> library(ggplot2)
```



```
> ggplot(pivot_benefit_by_gender) + geom_line(aes(x=PAY_YEAR, y=FEMALE, gr
oup=1, colour='FEMALE'))+ geom_line(aes(x=PAY_YEAR, y=MALE, group=1, colou
r='MALE'))+ labs(y="Benefits Pay", x="Pay Year", title=("Comparison analys
is of Male and Female for benefits pay"))
```

Code Text:

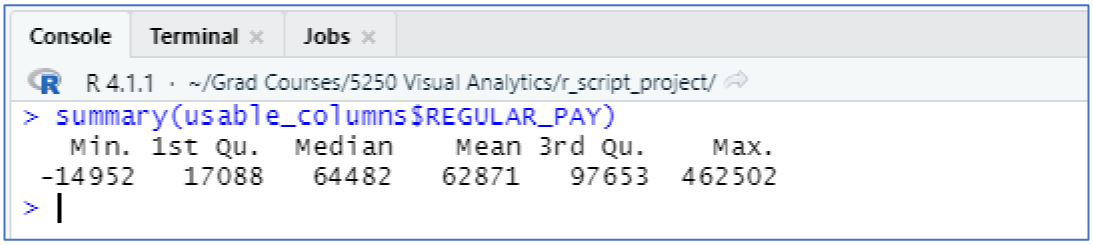
```
> setwd("C:/Users/parek/OneDrive - Cal State LA/Visual Analytics/R Programming")

> payroll=read.csv("City_Employee_Payroll.csv",header=T)

> usable_columns <- subset(payroll, select=c(PAY_YEAR,
+                                      DEPARTMENT_TITLE,
+                                      MOU_TITLE,
+                                      EMPLOYMENT_TYPE,
+                                       REGULAR_PAY,
+                                       GENDER,
+                                       BENEFIT_PAY,
+                                       OVERTIME_PAY,
+                                       ALL_OTHER_PAY))

>library(tidyverse)

> usable_columns <- usable_columns %>%

+ mutate(PAY_YEAR=as.character(PAY_YEAR),

+ DEPARTMENT_TITLE = as.character(DEPARTMENT_TITLE),

+ MOU_TITLE = as.character(MOU_TITLE),

+ EMPLOYMENT_TYPE = as.character(EMPLOYMENT_TYPE),

+ GENDER = as.character(GENDER),

+ REGULAR_PAY=as.integer(REGULAR_PAY),

+ BENEFIT_PAY=as.integer(BENEFIT_PAY),

+ OVERTIME_PAY=as.integer(OVERTIME_PAY),

+ ALL_OTHER_PAY=as.integer(ALL_OTHER_PAY))

> benefit_pay_by_gender<-  usable_columns %>%
+ group_by(PAY_YEAR, GENDER) %>%
+ summarise(Benefit_Pay=mean(BENEFIT_PAY))
> benefit_pay_by_gender <- benefit_pay_by_gender %>%
+ na.omit
> benefit_pay_by_gender <- subset(benefit_pay_by_gender,
GENDER!="",GENDER!="UNKNOWN")
> pivot_benefit_by_gender <- benefit_pay_by_gender %>%
+ pivot_wider(names_from = GENDER, values_from = Benefit_Pay)
> library(ggplot2)
ggplot(pivot_benefit_by_gender) + geom_line(aes(x=PAY_YEAR, y=FEMALE, group=1,
colour='FEMALE'))+ geom_line(aes(x=PAY_YEAR, y=MALE, group=1, colour='MALE'))+
labs(y="Benefits Pay", x="Pay Year", title=("Comparison analysis of Male and Female for
benefits pay"))
```

In this dataset, for doing the analysis initially the Gender column had NA values and unknown rows which had to be removed and so subset and na.omit() is used in the code. na.omit() majorly removes all the NA values from the data-frame specified while subset() acts as a where clause where one can put a specific condition and in our case it was to remove the “Unknown” and “blank values.” To build a multi-variate line graph, we need two separate columns for variables Male and Female and hence we used pivot_wider function for the same. Pivot_wider is used when one wants to increase the number of columns and decrease the number of rows. To make a line chart, we used the ggplot and tidyverse library. As there are two variables earlier mentioned, we used the function of geom_line twice in the code. labs() was used to name the respective X and Y axis and to name the title of the graph.

F. Statistical Summary and Functions

Summary Function

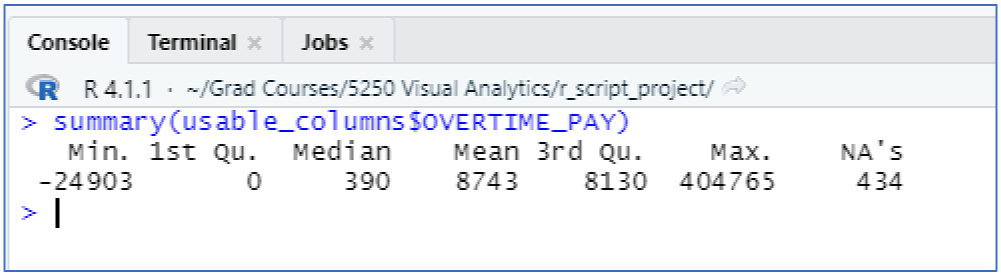
The summary function was applied to the regular_pay and overtime_pay fields. The following screenshot displays the output provided by the summary function for the regular_pay field:


Console
Terminal
Jobs
R 4.1.1 · ~/Grad Courses/5250 Visual Analytics/r_script_project/

```
> summary(usable_columns$REGULAR_PAY)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
 -14952   17088   64482   62871   97653  462502 
> 
```

The max salary paid by the City of Los Angeles from 2013 to 2021 was $462,502 dollars which is over seven times more than the average. The median and mean do not differ by much; as a result, one may assume that the outliers, if any, are not skewing the mean. The minimum value is a negative value which is a possible value. Based on the 1st Quartile, twenty five percent of values fall below $17,088. The dataset contains part time employees and payroll for one-time events. As a result, the low value is logical. Lastly, twenty five percent of values fall above $97,653. The dataset contains data for eight years. As a result, the summary statistics do not reflect if the twenty five percent are a small group of individuals that have been with the city throughout those years (McLeod, 2019).

The following screenshot displays the output provided for the overtime_pay field:

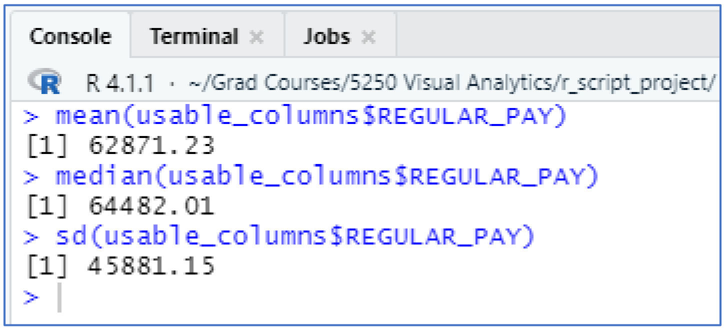

Console
Terminal
Jobs
R 4.1.1 · ~/Grad Courses/5250 Visual Analytics/r_script_project/

```
> summary(usable_columns$OVERTIME_PAY)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
 -24903       0     390    8743    8130  404765     434 
> 
```

The mean and median differ by over $8,000, indicating that the dataset is skewed as it relates to overtime pay. Several records fall under zero dollars which is valid as some employee roles may not require overtime such as non-critical roles. A total of 434 payroll records do not have a value for the overtime_pay field. This may indicate that overtime was not earned. Twenty five percent of records (158,482/634,338) have an overtime value that falls over $8,130 which indicates that most employees were compensated less than eight thousand in overtime. The max value, at first glance, appears to be an error; however, a total of 304 entries has an overtime value over $200,000. These payroll records correspond to the fire department and water and power.

Statistical Functions

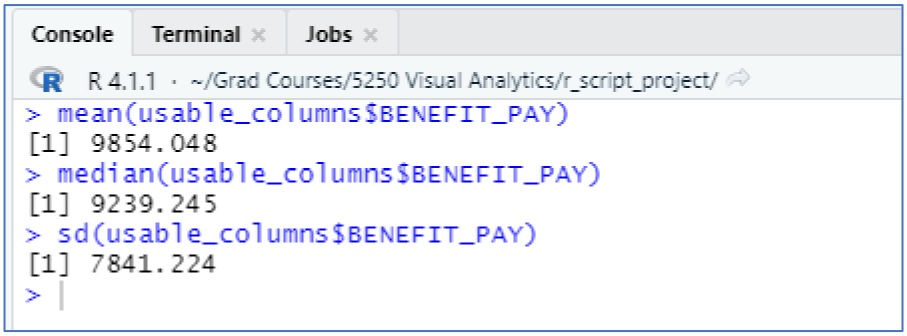
The regular_pay and benefit_pay fields were analyzed using the mean, median, and standard deviation functions. The code that was executed for the regular_pay field is as follows:


Console
Terminal
Jobs
R 4.1.1 · ~/Grad Courses/5250 Visual Analytics/r_script_project/

```
> mean(usable_columns$REGULAR_PAY)
[1] 62871.23
> median(usable_columns$REGULAR_PAY)
[1] 64482.01
> sd(usable_columns$REGULAR_PAY)
[1] 45881.15
> 
```

As mentioned in the prior section, the median and mean do not vary by much which possibly indicates that the mean is not being skewed. On average, employees are paid an annual salary of $64,481.01. However, based on the standard deviation, this salary varies on average by $45,881.15. This may indicate that salaries are spread out and are not closely clustered around the mean.

The code that was execute for the benefit_pay field is as follows:


Console
Terminal
Jobs
R 4.1.1 · ~/Grad Courses/5250 Visual Analytics/r_script_project/

```
> mean(usable_columns$BENEFIT_PAY)
[1] 9854.048
> median(usable_columns$BENEFIT_PAY)
[1] 9239.245
> sd(usable_columns$BENEFIT_PAY)
[1] 7841.224
> 
```

Like the regular pay field, the mean and median do not vary much. The mean has $614.80 more than the median which is the halfway point. As a result, the mean is reliable. Therefore, one can state that the average benefit pay provided by the City of Los Angeles is $9854.05. The standard deviation is close to the median and mean in value. This indicates that values are widely dispersed around the mean.

Script One

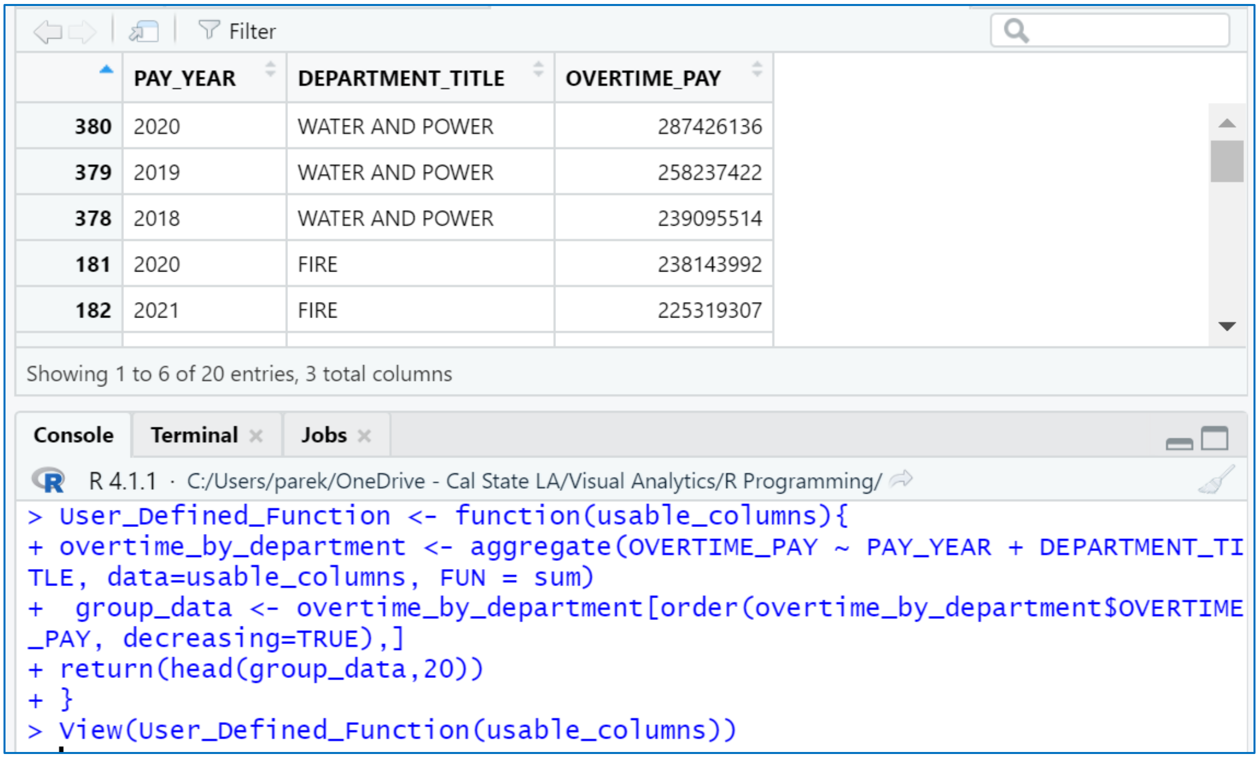
```
setwd("~/Grad Courses/5250 Visual Analytics/r_script_project")

payroll<-read.csv("payroll.csv", header=T)

usable_columns <- subset(payroll, select=c(PAY_YEAR,
                                 DEPARTMENT_TITLE,
                                 MOU_TITLE,
                                 EMPLOYMENT_TYPE,
                                 REGULAR_PAY,
                                 GENDER,
                                 BENEFIT_PAY,
                                        OVERTIME_PAY,
                                        ALL_OTHER_PAY))

library(tidyverse)
library(ggplot2)

usable_columns <- usable_columns %>%
  mutate(PAY_YEAR=as.character(PAY_YEAR),
         DEPARTMENT_TITLE = as.character(DEPARTMENT_TITLE),
         MOU_TITLE = as.character(MOU_TITLE),
         EMPLOYMENT_TYPE = as.character(EMPLOYMENT_TYPE),
         GENDER = as.character(GENDER),
         REGULAR_PAY=as.integer(REGULAR_PAY),
         BENEFIT_PAY=as.integer(BENEFIT_PAY),
         OVERTIME_PAY=as.integer(OVERTIME_PAY),
         ALL_OTHER_PAY=as.integer(ALL_OTHER_PAY))


pay_by_role <- subset(usable_columns, EMPLOYMENT_TYPE == 'FULL_TIME',
select=c(MOU_TITLE,
                                                                          REGULAR_PAY))

avg_pay_by_title<-pay_by_role %>%
  group_by(MOU_TITLE) %>%
  summarise(AVG_PAY=mean(REGULAR_PAY))

top_ten <- head(arrange(avg_pay_by_title, desc(AVG_PAY)), n = 10)

ggplot(top_ten) +
  geom_col(aes(x= reorder(MOU_TITLE, AVG_PAY), y=AVG_PAY, fill = MOU_TITLE),
show.legend=FALSE) +
  coord_flip() +
  scale_fill_manual(values = c(
    "UNREPRESENTED UNIT - MANAGEMENT BENEFITS" = "#52BE80",
    "PORT PILOTS" = "lightgrey",
    "MANAGEMENT ATTORNEYS" = "lightgrey",
    "FIRE CHIEF OFFICERS" = "lightgrey",
    "MANAGEMENT EMPLOYEES UNIT" = "lightgrey",
    "POLICE OFFICERS, CAPTAIN. AND ABOVE" = "lightgrey",
    "PERSONNEL DIRECTOR" = "lightgrey",
    "LOS ANGELES PORT POLICE COMMAND OFFICERS" = "lightgrey",
    "SUPERVISORY PROFESSIONAL UNIT" = "lightgrey",
    "CONFIDENTIAL ATTORNEYS" = "lightgrey"
  )) +
  theme(axis.ticks = element_blank(),
        panel.background = element_blank(),
        axis.title.x = element_blank(),
        axis.title.y = element_blank(),
        axis.text.x = element_text(size=14),
        #mayneed to adjust hjust values when executing
        plot.title = element_text(hjust = -1.91, size=20),
        plot.subtitle = element_text(hjust = -.539)) +
  scale_y_continuous(limits=c(0,250000),position="right") +
  ggtitle(label="Top 10 Paying Jobs in Los Angeles from 2013 to 2021",
          subtitle = "Salary was averaged for all reporting years.") +
  annotate("text", x=10, y=220000, label="Top paying job earned an
average of $184,739.20", colour= "#247547", fontface=2, size=4.5)
```

Script Two

```
#group by department_title and pay_year and adding the sum of overtime_pay
overtime_by_department <- aggregate(OVERTIME_PAY ~ PAY_YEAR +
DEPARTMENT_TITLE, data=usable_columns, FUN = sum)
#return the max values by overtime_pay
> overtime_by_department <-
overtime_by_department[head(order(overtime_by_department$OVERTIME_PAY,
decreasing=TRUE),20), ]
#using library ggplot2
library(ggplot2)
#using overtime_by_department for ggplot2
ggplot(overtime_by_department, aes(DEPARTMENT_TITLE, OVERTIME_PAY, fill
=PAY_YEAR)) +  geom_bar( stat = "identity")
#adding the title
+ ggtitle("Water and Power department has generated the highest overtime pay in the year
2018,2019 and 2020.")
# changing the y-axis label, and removing e-notation from y-axis values and using comma
separated
+ scale_y_continuous(name= "Over-Time Pay",labels=scales::comma)
#Changing the color to pre-defined color palette
+ scale_fill_brewer()
```

User-Defined Function

Code Screenshot:


Filter
Console
Terminal
Jobs
R 4.1.1 · C:/Users/parek/OneDrive - Cal State LA/Visual Analytics/R Programming/

	PAY_YEAR	DEPARTMENT_TITLE	OVERTIME_PAY
380	2020	WATER AND POWER	287426136
379	2019	WATER AND POWER	258237422
378	2018	WATER AND POWER	239095514
181	2020	FIRE	238143992
182	2021	FIRE	225319307

Showing 1 to 6 of 20 entries, 3 total columns

```
> User_Defined_Function <- function(usable_columns){
+ overtime_by_department <- aggregate(OVERTIME_PAY ~ PAY_YEAR + DEPARTMENT_TI
TLE, data=usable_columns, FUN = sum)
+  group_data <- overtime_by_department[order(overtime_by_department$OVERTIME
_PAY, decreasing=TRUE),]
+ return(head(group_data,20))
+ }
> View(User_Defined_Function(usable_columns))
```

Code Text:

```
> User_Defined_Function <- function(usable_columns){

+ overtime_by_department <- aggregate(OVERTIME_PAY ~ PAY_YEAR +

DEPARTMENT_TITLE, data=usable_columns, FUN = sum)

+  group_data <-

overtime_by_department[order(overtime_by_department$OVERTIME_PAY,

decreasing=TRUE),]

+ return(head(group_data,10))

+ }
```

The user-defined function is specific to what a user requires, and, once created, it can be used like the built-in functions we used with the ggplot2 code. The function of aggregate is used to split the data into subsets and return the results in a group by form. As the dataset consists of each department having multiple entries for several types of employee types and pay year, we used the aggregate function. Head and order are used for ranking the top departments as per the over time pay.

REFERENCES

Dolan, J. (2016, December 30). *When city retirement pays better than the job.* https://www.latimes.com/projects/la-me-el-monte-pensions/

Indeed (2021, March 25). *Government Jobs vs. Private Jobs: What's the Difference?* https://www.indeed.com/career-advice/finding-a-job/government-jobs-vs-private-jobs

Issid, J. (n.d.). *The Myths and Realities of Working in the Public Sector*. https://www.monster.ca/career-advice/article/myths-and-realities-of-working-in-public-sector

Jamison, P. (2016, November 18). *Paying for public retirees has never cost L.A. taxpayers more. And that's after pension reform.* https://www.latimes.com/projects/la-me-pension-squeeze/

McLeod, S. (2019). *What does a box plot tell you?* SimplyPsychology. https://www.simplypsychology.org/boxplots.html

Yoder, E. (2019, November 6). *Federal employees salaries lag private sector by 27 percent on average, report says*. https://www.washingtonpost.com/politics/2019/11/06/federal-employee-salaries-lag-private-sector-by-percent-average-report-says/

LADWP Benefits, https://insidedwp.ladwp.com/webcenter/portal/lr/home/LADWP_Benefits?_adf.ctrl-state=om27ih4_4.