Analysis 1: UNC Salaries

JAKE JAMES

February 4, 2021

Introduction

Universities are typically opaque, bureaucratic institutions. To be transparent to tax payers, many public schools, such as the University of North Carolina, openly report **salary information**. In this assignment, we will analyze this information to answer pivotal questions that have endured over the course of time. The most recent salary data for UNC-Chapel Hill faculty and staff has already been downloaded in CSV format and titled *"UNC_System_Salaries Search and Report.csv"*. If you scan the spreadsheet, you will notice that Dr. Mario is not listed. People get depressed when they see that many digits after the decimal.

To answer all the questions, you will need the R package `tidyverse` to make figures and utilize `dplyr` functions.

Data Information

Make sure the CSV data file is contained in the folder of your RMarkdown file. First, we start by using the `read_csv` function from the `readr` package found within the tidyverse. The code below executes this process by creating a tibble in your R environment named "salary".

```
salary=read_csv("UNC_System_Salaries Search and Report.csv")
```

Now, we will explore the information that is contained in this dataset. The code below provides the names of the variables contained in the dataset.

```
names(salary)
```

```
##  [1] "Name"                  "campus2"               "dept"
##  [4] "position"              "PRIMARY_WORKING_TITLE" "hiredate"
##  [7] "exempt"                "fte"                   "employed"
## [10] "statesal"              "nonstsal"              "totalsal"
## [13] "stservyr"
```

Next, we will examine the type of data contains in these different variables.

```
str(salary,give.attr=F)
```

```
## tibble [12,646 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Name                  : chr [1:12646] "AACHOUI, YOUSSEF" "AARNIO, REA T" "ABAJAS, YASMINA L" "ABARI
##  $ campus2               : chr [1:12646] "UNC-CHAPEL HILL" "UNC-CHAPEL HILL" "UNC-CHAPEL HILL" "UNC-CI
```

```
##  $ dept                : chr [1:12646] "Microbiology and Immunology" "SW-Research Projects" "Peds-H
##  $ position            : chr [1:12646] "Research Professional, Medical" "Functional Paraprofessional
##  $ PRIMARY_WORKING_TITLE: chr [1:12646] "Research Associate" "Graphic Designer" "NODESCR" "Associate
##  $ hiredate            : chr [1:12646] "10/10/2011" "1/14/2013" "7/1/2015" "1/1/1999" ...
##  $ exempt              : chr [1:12646] "Exempt from Personnel Act" "Subject to State Personnel Act"
##  $ fte                 : num [1:12646] 1 0.8 1 1 1 1 1 1 1 1 ...
##  $ employed            : num [1:12646] 12 12 12 9 12 12 12 9 12 9 ...
##  $ statesal            : logi [1:12646] NA NA NA NA NA NA ...
##  $ nonstsal            : logi [1:12646] NA NA NA NA NA NA ...
##  $ totalsal            : num [1:12646] 49128 33257 139405 181000 41098 ...
##  $ stservyr            : num [1:12646] 1 5 2 20 6 8 6 1 19 1 ...
```

You will notice that the variable "hiredate" is recorded as a character. The following code will first modify the original dataset to change this to a date variable with the format *mm/dd/yyyy*. Then, we will remove the hyphens to create a numeric variable as *yyyymmdd*. Finally, in the spirit of tidyverse, we will convert this data frame to a tibble.

```
salary$hiredate=as.Date(salary$hiredate, format="%m/%d/%Y")
salary$hiredate=as.numeric(gsub("-","",salary$hiredate))
salary=as_tibble(salary)
```

Now, we will use `head()` to view of first five rows and the modifications made to the original data. The rest of the assignment will extend off this modified dataset named `salary` which by now should be in your global environment.

```
head(salary,5)
```

```
## # A tibble: 5 x 13
##   Name  campus2 dept  position PRIMARY_WORKING~ hiredate exempt   fte employed
##   <chr> <chr>   <chr> <chr>    <chr>               <dbl> <chr>  <dbl>    <dbl>
## 1 AACH~ UNC-CH~ Micr~ Researc~ Research Associ~ 20111010 Exemp~   1         12
## 2 AARN~ UNC-CH~ SW-R~ Functio~ Graphic Designer 20130114 Subje~   0.8       12
## 3 ABAJ~ UNC-CH~ Peds~ Assista~ NODESCR          20150701 Exemp~   1         12
## 4 ABAR~ UNC-CH~ Kena~ Associa~ Associate Profe~ 19990101 Exemp~   1          9
## 5 ABAR~ UNC-CH~ Inst~ Researc~ Research Techni~ 20110912 Subje~   1         12
## # ... with 4 more variables: statesal <lgl>, nonstsal <lgl>, totalsal <dbl>,
## #   stservyr <dbl>
```

Assignment

Part 1: Reducing the Data to a Smaller Set of Interest

Q1 *(2 Points)*

Create a new dataset named `salary2` that only contains the following variables:

- "Name"
- "dept"
- "position"

- "hiredate"
- "exempt"
- "totalsal"

Then, use the `names()` function to display the variable names of `salary2`.

```
salary2<-select(salary, "Name", "dept", "position", "hiredate", "exempt", "totalsal")
names(salary2)
```

```
## [1] "Name"     "dept"     "position" "hiredate" "exempt"   "totalsal"
```

Q2 ***(2 Points)***

Now, we modify `salary2`. Rename the variables "dept","position","exempt","totalsal" to "Department","Job","Exempt", and "Salary", respectively. Do this for a new dataset called `salary3` and use `names()` to display the variable names of `salary3`.

```
salary3=rename(salary2,Department=dept, Job=position, Exempt=exempt, Salary=totalsal)
names(salary3)
```

```
## [1] "Name"       "Department" "Job"        "hiredate"   "Exempt"
## [6] "Salary"
```

Q3 ***(2 Points)***

Now, we modify `salary3`. Create a new variable called "HireYear" that only contains the first four digits of the variable "hiredate" in a new dataset named `salary4`.

```
salary4=mutate(salary3,
          HireYear=hiredate%/%10000)
names(salary4)
```

```
## [1] "Name"       "Department" "Job"        "hiredate"   "Exempt"
## [6] "Salary"     "HireYear"
```

Q4 ***(2 points)***

Now, we modify `salary4`. Create a new variable called "YrsEmployed" which reports the number of full years the employee has worked at UNC. Assume that all employees are hired January 1. Create a new dataset named `salary5`.

```
salary5=mutate(salary4,
          YrsEmployed= 2020-HireYear)
names(salary5)
```

```
## [1] "Name"        "Department"  "Job"         "hiredate"    "Exempt"
## [6] "Salary"      "HireYear"    "YrsEmployed"
```

Q5 ***(4 points)***

Now, we modify `salary5` to create our final dataset named `salary.final`. Use the pipe `%>%` to make the following changes:

- Drop the variables "hiredate" and "HireYear".
- Sort the observations first by "Department" and then by "YrsEmployed".
- Rearrange the variables so that "YrsEmployed" and "Salary" are the first two variables in the dataset, in that order, without removing any of the other variables.

After you have used the `%>%` to make these changes, use the function `head()` to display the first 10 rows of `salary.final`.

```
salary.final =

  salary5 %>%

  select("YrsEmployed", "Salary", "Name", "Department", "Job", "Exempt") %>%

  arrange(Department, YrsEmployed)

head(salary.final,10)
```

```
## # A tibble: 10 x 6
##    YrsEmployed Salary Name         Department       Job             Exempt
##          <dbl>  <dbl> <chr>        <chr>            <chr>           <chr>
##  1           3  39646 DALEY, JOS~ A and S - Busin~ Fiscal Affair~ Subject to St~
##  2           3  48814 WEBSTER, C~ A and S - Busin~ HR Coordinator Subject to St~
##  3           3  48814 WOODSON, K~ A and S - Busin~ HR Coordinator Subject to St~
##  4           3  48814 WORTHEN, T~ A and S - Busin~ HR Coordinator Subject to St~
##  5           4  47164 CHESTER, A~ A and S - Busin~ HR Coordinator Subject to St~
##  6           4  47983 GIBSON, JE~ A and S - Busin~ Fiscal Affair~ Subject to St~
##  7           4  39646 RAUSCHER, ~ A and S - Busin~ Fiscal Affair~ Subject to St~
##  8           4  39646 STRINGFELL~ A and S - Busin~ Fiscal Affair~ Subject to St~
##  9           5  48814 WATSON, ST~ A and S - Busin~ HR Coordinator Subject to St~
## 10           5  47983 YOUSEF, HE~ A and S - Busin~ Fiscal Affair~ Subject to St~
```

Part 2: Answering Questions Based on All Data

Q6 ***(2 Points)***

What is the average salary of employees in the Law Department?

Code *(1 Point)*:

```
lawsalary =

  salary.final %>%

  filter(Department == "Law")

mean(lawsalary[["Salary"]])
```

```
## [1] 112567.1
```

Answer *(1 Point)*: The average salary of employees in the Law Department is $112,567.10.

Q7 *(4 Points)*

How many employees have worked in Family Medicine between 5 and 8 years (inclusive) and are exempt from personnel act?

Code *(2 Points)*:

```
medsalary =

  salary.final %>%

  filter(Department == "Family Medicine" & YrsEmployed >=5 & YrsEmployed <=8 & Exempt == "Exempt from P

count(medsalary)
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1    16
```

Answer *(2 Points)*: There are 16 employees at UNC that have worked in Family Medicine between 5 and 8 years and are exempt from personnel act.

Q8 *(4 Points)*

What is the mean salary of employees from the Linguistics department who are professors, associate professors, or assistant professors?

Code *(2 Points)*:

```
lingsalary =

  salary.final %>%

  filter(Job == "Professor" | Job == "Associate Professor" | Job == "Assistant Professor")

mean(lingsalary[["Salary"]])
```

```
## [1] 155931.7
```

Answer *(2 Points)*: At UNC, the mean salary of employees from the Linguistics department who are professors, associate professors, or assistant professors is $155,931.70.

Part 3: Answering Questions Based on Summarized Data

Q9 *(4 Points)*

Based off the data in `salary.final`, create a grouped summary based off combinations of "Department" and "YrsEmployed". Call the new tibble `deptyear_summary`. Your summarized tibble, `deptyear_summary`, should report all of the following statistics with corresponding variable names in the following order.

- "n" = number of employees for each combination
- "mean" = average salary for each combination
- "sd" = standard deviation of salary for each combination.
- "min" = minimum salary for each combination.
- "max" = maximum salary for each combination

In the process, make sure you use `ungroup()` with the pipe `%>%` to release the grouping so future work is no longer group specific. Following the creation of `deptyear_summary`, prove that your code worked by using `head()` to view the first 10 rows.

```
deptyear_summary =

  salary.final %>%

  group_by(Department, YrsEmployed) %>%

  summarize(
    count=n(),
    mean=mean(Salary, na.rm=T),
    sd=sd(Salary, na.rm=T),
    min=min(Salary, na.rm=T),
    max=max(Salary, na.rm=T),
    .groups="keep"
  ) %>%


  ungroup()

  head(deptyear_summary, 10)
```

```
## # A tibble: 10 x 7
##    Department                YrsEmployed count   mean     sd   min   max
##    <chr>                           <dbl> <int>  <dbl>  <dbl> <dbl> <dbl>
##  1 A and S - Business Center           3     4 46522   4584  39646 48814
##  2 A and S - Business Center           4     4 43610.  4589. 39646 47983
##  3 A and S - Business Center           5     2 48398.   588. 47983 48814
##  4 A and S - Business Center           6     2 52190.  2703. 50278 54101
##  5 A and S - Business Center           7     2 54488   9199. 47983 60993
##  6 Acad Initiatives-UBC                4     1 23250     NA  23250 23250
##  7 Acad Initiatives-UBC                6     1 48782     NA  48782 48782
##  8 Acad Initiatives-UBC                9     1 60341     NA  60341 60341
##  9 Acad Initiatives-UBC               10     1 54851     NA  54851 54851
## 10 Acad Initiatives-UBC               17     2 64916  12875. 55812 74020
```

Q10 ***(4 Points)***

Using the summarized data in `deptyear_summary`, use the `dplyr` functions to identify the 3 departments that award the lowest average salary for employees who have been employed for 3 years. The output should only show the 3 departments along with the corresponding years employed, which should all be 3, and the four summarizing statistics created.

Furthermore, explain why the standard deviation for the 3 departments in your list has a salary standard deviation of "NaN". What does this mean and how did it occur?

Code *(2 Points)*:

```
lowsal_summary =

deptyear_summary %>%

  filter(YrsEmployed == 3) %>%

  arrange(mean)

head(lowsal_summary, 3)
```

```
## # A tibble: 3 x 7
##   Department                      YrsEmployed count  mean    sd   min   max
##   <chr>                                 <dbl> <int> <dbl> <dbl> <dbl> <dbl>
## 1 Religious Studies                         3     1 16852    NA 16852 16852
## 2 Ath Olympic Sport Administratn            3     1 19276    NA 19276 19276
## 3 Jewish Studies                            3     1 19750    NA 19750 19750
```

Answer *(2 Points)*: At UNC, the 3 departments that award the lowest average salary for employees who have been employed for 3 years are Religious Studies, Ath Olympic Sport Administration, and Jewish Studies. The salary standard deviation of "NaN" is caused by there only being one person in the department that has been employed at Carolina for exactly 3 years.

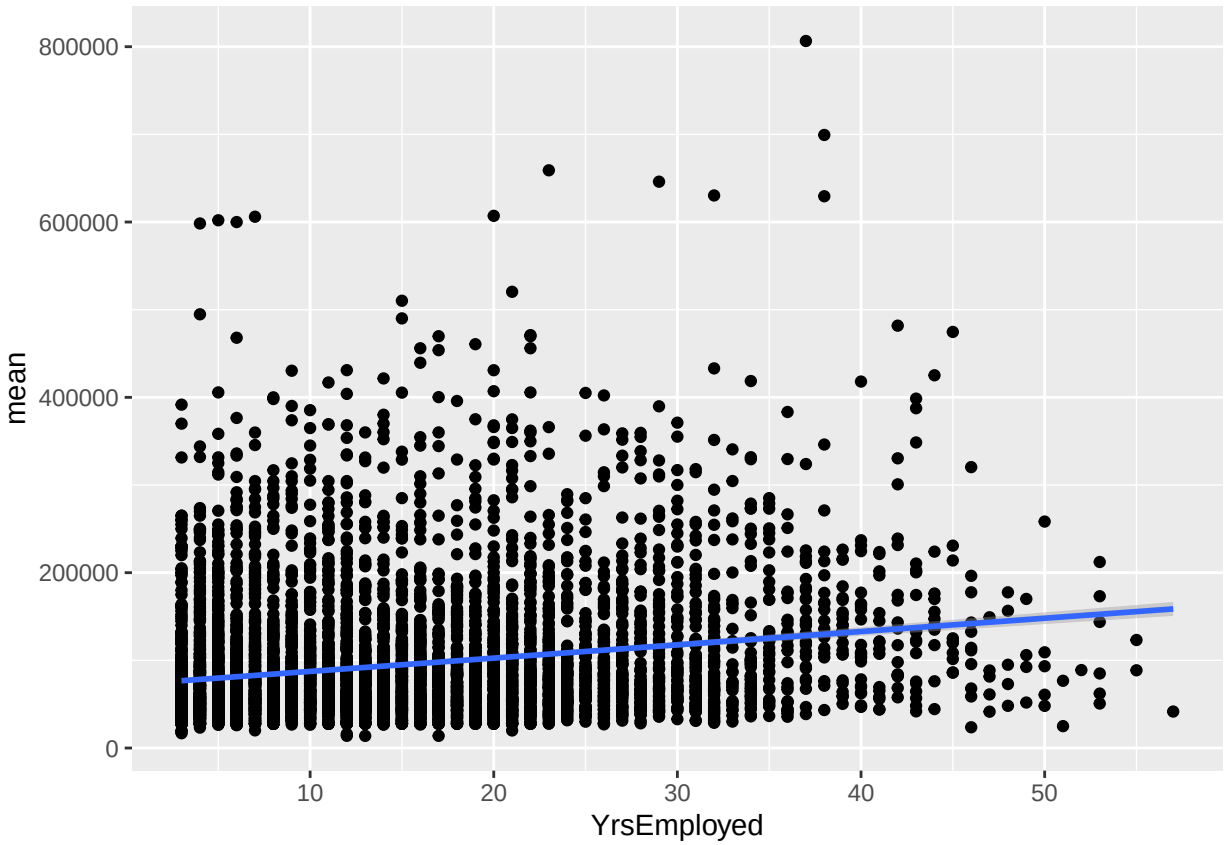
Q11 ***(4 points)***

Create a scatter plot using `geom_point()` along with fitted lines using `geom_smooth` with the argument `method="lm"` showing the linear relationship between average salary and the years employed. For this plot, use the summarized data in `deptyear_summary`. Following the plot, please explain what this plot suggests about the relationship between the salary a UNC employee makes and how many years that employee has served.

Code and Figure *(2 Points)*:

```
ggplot(data = deptyear_summary) +
  geom_point(aes(x = YrsEmployed, y = mean)) +
    geom_smooth(aes(x = YrsEmployed, y = mean),method = lm)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Answer *(2 Points)*: The Figure above shows that there is a weak positive correlation between the amount of years someone has been employed at UNC and their average salary. This shows that people will start to earn more money through their years of service, but as seen in the 11 groups that make around $600,000+, the spread of those groups goes from as little as 5 years to as many as 40.

Q12 ***(6 Points)***

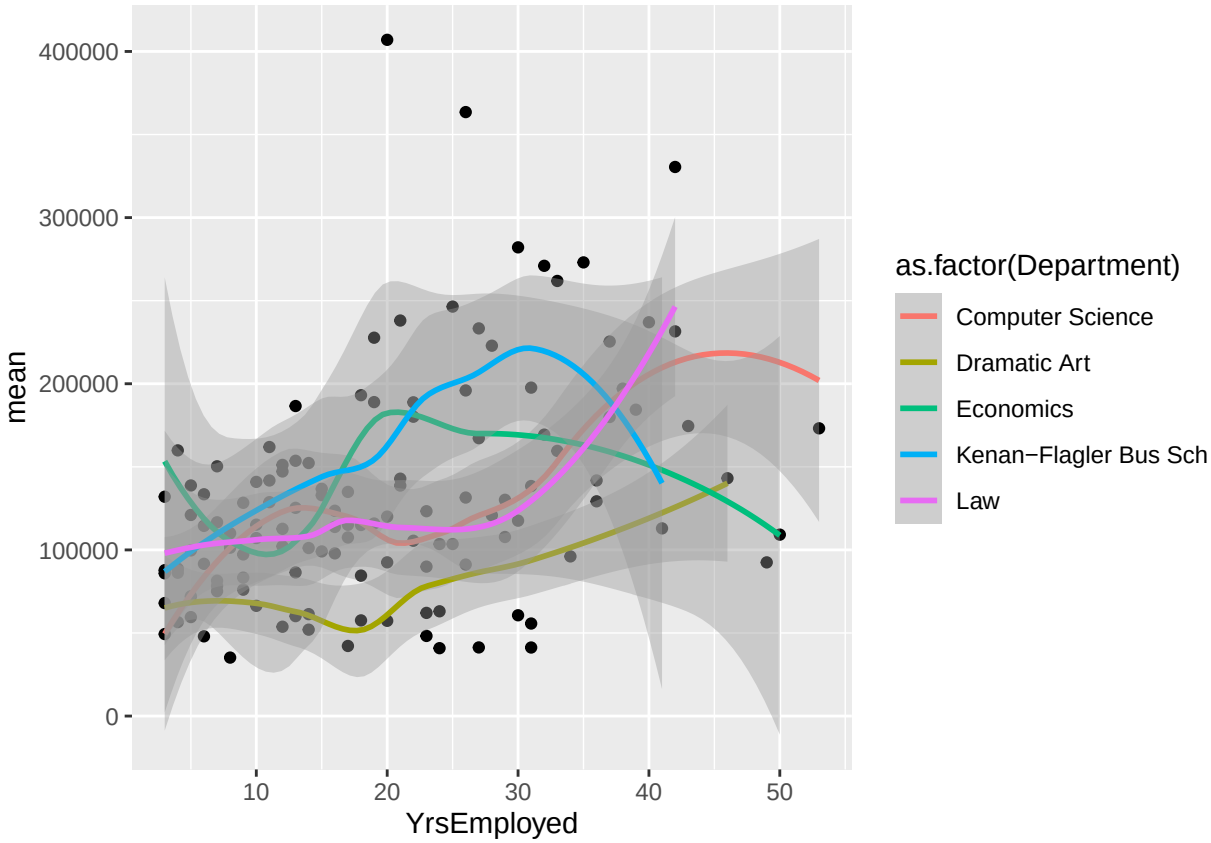
The purpose of summarizing the data was to analyze the previously discussed linear relationship by group. In `deptyear_summary`, there are 702 unique departments represented. You can verify this by using `length(unique(deptyear_summary$Department))`. In this part, I want you to select 5 academic departments, not previously discussed, and in one figure, display the scatter plots and fitted regression lines representing the relationship between average salary and years employed in 5 different colors. Then, in complete sentences, I want you to state what departments you chose and explain the differences and/or similarities between the groups regarding the previously mentioned relationship. Compare departments on the starting salary and the rate of increase in salary based on the fitted lines.

Code and Figure: *(3 Points)*:

```
deptyear_summary5 =

  deptyear_summary %>%

  filter(Department == "Law" | Department == "Economics" | Department == "Dramatic Art" | Department ==

dys5plot = ggplot(data = deptyear_summary5) +
  geom_point(aes(x = YrsEmployed, y = mean)) +
    geom_smooth(aes(x=YrsEmployed,y=mean,color=as.factor(Department)))
```

```
dys5plot
```

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



Answer *(3 Points)*: I chose Computer Science, Dramatic Art, Economics, Business, and Law as my 5 departments. Most of them as a whole follow the larger trend, however some of the lines plateau and even decrease into the higher years of experience. I predict this is because of the lack of data points in that range.

Computer Science and Law follow almost the exact same path, a similar one to the original, slightly stronger positively.

Business has the highest salaries overall, but falls off dramatically after 30 years of employment, which is unique to it.

Economics has the highest starting salary but slowly declines after 20 years of experience, leaving it the lowest salary at the end of the figure.

Dramatic Art has the lowest starting salary, but slowly increases as experience increases, this department follows the original trend the best.