R Portfolio 2

Caitlin Jagla

3/25/2022

1. Preliminary Data Wrangling

```
# load data
df <- read_csv("nurses.csv") %>% as_tibble()

# preview dataset
glimpse(df)
```

```
## Rows: 1,242
## Columns: 22
## $ State                                        <chr> "Alabama", "Alaska",~
## $ Year                                         <dbl> 2020, 2020, 2020, 20~
## $ `Total Employed RN`                          <dbl> 48850, 6240, 55520, ~
## $ `Employed Standard Error (%)`                <dbl> 2.9, 13.0, 3.7, 4.2,~
## $ `Hourly Wage Avg`                            <dbl> 28.96, 45.81, 38.64,~
## $ `Hourly Wage Median`                         <dbl> 28.19, 45.23, 37.98,~
## $ `Annual Salary Avg`                          <dbl> 60230, 95270, 80380,~
## $ `Annual Salary Median`                       <dbl> 58630, 94070, 79010,~
## $ `Wage/Salary standard error (%)`             <dbl> 0.8, 1.4, 0.9, 1.4, ~
## $ `Hourly 10th Percentile`                     <dbl> 20.75, 31.50, 27.66,~
## $ `Hourly 25th Percentile`                     <dbl> 23.73, 36.94, 32.58,~
## $ `Hourly 75th Percentile`                     <dbl> 33.15, 53.31, 44.67,~
## $ `Hourly 90th Percentile`                     <dbl> 38.67, 60.70, 50.14,~
## $ `Annual 10th Percentile`                     <dbl> 43150, 65530, 57530,~
## $ `Annual 25th Percentile`                     <dbl> 49360, 76830, 67760,~
## $ `Annual 75th Percentile`                     <dbl> 68960, 110890, 92920~
## $ `Annual 90th Percentile`                     <dbl> 80420, 126260, 10429~
## $ `Location Quotient`                          <dbl> 1.20, 0.98, 0.91, 1.~
## $ `Total Employed (National)_Aggregate`        <dbl> 140019790, 140019790~
## $ `Total Employed (Healthcare, National)_Aggregate` <dbl> 8632190, 8632190, 86~
## $ `Total Employed (Healthcare, State)_Aggregate`    <dbl> 128600, 17730, 17101~
## $ `Yearly Total Employed (State)_Aggregate`    <dbl> 1903210, 296300, 283~
```

There are no duplicated rows in this dataset: 1242 rows out of 1242 are distinct.

The column/variable names are untidy so I built a description key table and replaced with names that are more R compatible.

```r
# build variable description key table to generate tidy column names
desc <- tibble(desc = colnames(df)) %>%
          mutate(new_name = desc %>%
                            tolower() %>%
                            str_replace_all(c(" percentile" = "",
                                              "hourly" = "hrly",
                                              "annual" = "ann",
                                              "standard error" = "se",
                                              "aggregate" = "agg",
                                              "national" = "natl",
                                              "employed" = "empl",
                                              " (%)" = "",
                                              "[,(%)]"= "",
                                              "[/]" = "_" ))  %>%
                            str_trim() %>%
                            str_replace_all(., "[ ]", "_"))

# rename columns with tidy names
df <- df %>% set_names(desc$new_name)

# view description key table
desc
```

```
## # A tibble: 22 x 2
##    desc                              new_name
##    <chr>                             <chr>
##  1 State                             state
##  2 Year                              year
##  3 Total Employed RN                 total_empl_rn
##  4 Employed Standard Error (%)       empl_se
##  5 Hourly Wage Avg                   hrly_wage_avg
##  6 Hourly Wage Median                hrly_wage_median
##  7 Annual Salary Avg                 ann_salary_avg
##  8 Annual Salary Median              ann_salary_median
##  9 Wage/Salary standard error (%)    wage_salary_se
## 10 Hourly 10th Percentile            hrly_10th
## # i 12 more rows
```

The dataset contains `NA` values (`anyNA(df)` =TRUE), so I checked to see how many are in each column. Since every column but `location_quotient` has less than 10 `NA`s, I filtered the dataset to exclude rows that contain `NA` in any column *except* `location_quotient`.

```r
# count number of NAs in each column
df %>%
  select_if(~any(is.na(.))) %>%
  summarise_all(~(sum(is.na(.)))) %>%
  t()
```

```
##                                    [,1]
## total_empl_rn                         5
## empl_se                               5
## hrly_wage_avg                         6
## hrly_wage_median                      6
```

```
## ann_salary_avg                     6
## ann_salary_median                  6
## wage_salary_se                     6
## hrly_10th                          6
## hrly_25th                          6
## hrly_75th                          6
## hrly_90th                          6
## ann_10th                           6
## ann_25th                           6
## ann_75th                           6
## ann_90th                           6
## location_quotient                649
## total_empl_natl_agg                4
## total_empl_healthcare_natl_agg     4
## total_empl_healthcare_state_agg    2
```

```r
# drop rows with NA in any column except location_quotient
df <- df %>% drop_na(!location_quotient)

# count rows of filtered dataset
nrow(df)
```

```
## [1] 1235
```

2) Analysis

Research Question: what is the relationship between change in total RN employment and change in RN salaries?

First I checked to see if all states/territories have data starting at the same year. Unfortunately, two territories (Guam and the Virgin Islands) didn't start reporting until later than the rest. Therefore, they were filtered out of the dataset. I also removed Puerto Rico as it was the only other territory in the dataset. This leaves the 50 states and Washington DC for analysis.

```r
# check to see if data starts with the same year for all states/territories
df %>%
  group_by(state) %>%
  summarize(first = min(year), # find first year reported for each state/territory
            last = max(year)) %>% # find last year reported for each state/territory
  distinct(first, last) # find all distinct combinations of first & last year reported
```

```
## # A tibble: 3 x 2
##   first  last
##   <dbl> <dbl>
## 1  1998  2020
## 2  1999  2020
## 3  2000  2020
```

```r
# check to see which states/territories did not start reporting in 1998
df %>%
  group_by(state) %>%
```

```r
  summarize(first = min(year), # find first year reported for each state
            last = max(year)) %>% # find last year reported for each state
  filter(first != 1998) # show only those that did not start with 1998
```

```
## # A tibble: 2 x 3
##   state          first  last
##   <chr>          <dbl> <dbl>
## 1 Guam            1999  2020
## 2 Virgin Islands  2000  2020
```

```r
# remove the territories from the dataset
df <- df %>% filter(state != "Guam" & state != "Virgin Islands" & state != "Puerto Rico")
```

Second, I calculated the percent difference in `total RNs employed` and `median annual salary` between 1998 and 2020 for each state:

$$100 * \frac{2020_{median} - 1998_{median}}{1998_{median}}$$

This should allow me to compare the *change in salary* between states with less confounding by the differences in salary magnitude between states.

```r
# calculate percent differences and store in a new tibble
diff <- df %>%
  group_by(state) %>%
  filter(year == 1998|year==2020) %>%
  pivot_wider(names_from = year, # rearrange
              values_from = c(ann_salary_median, total_empl_rn),
              id_cols = state) %>%
  mutate(
    # calculate percent difference in RNs employed
      empl_diff = 100 * (total_empl_rn_2020 - total_empl_rn_1998)/
                   total_empl_rn_1998,
    # calculate percent difference in salary
      salary_diff = 100 * (ann_salary_median_2020 - ann_salary_median_1998)/
                     ann_salary_median_1998) %>%
  as_tibble() %>%
  arrange(desc(empl_diff)) %>% # rank by percent difference in RN employment
  rowid_to_column(var = "empl_diff_rank") %>%
  arrange(desc(salary_diff)) %>% # rank by percent difference in salary
  rowid_to_column(var = "salary_diff_rank")


# display top & bottom ranked states for each calculated metric

#show 5 biggest increases in median annual salary
diff %>%
  select(state, salary_diff, salary_diff_rank, empl_diff, empl_diff_rank) %>%
  slice_max(order_by = salary_diff, n=5)
```

```
## # A tibble: 5 x 5
##   state         salary_diff salary_diff_rank empl_diff empl_diff_rank
```

```
##   <chr>                 <dbl>            <int>     <dbl>          <int>
## 1 California             138.                1      78.3              7
## 2 Oregon                 137.                2      75.5              8
## 3 Wyoming                105.                3      42.3             29
## 4 New Hampshire          104.                4      31.8             37
## 5 Montana                104.                5      45.7             23
```

```r
#show 5 smallest increases in median annual salary
diff %>%
  select(state, salary_diff, salary_diff_rank, empl_diff, empl_diff_rank) %>%
  slice_min(order_by = salary_diff, n=5)
```

```
## # A tibble: 5 x 5
##   state       salary_diff salary_diff_rank empl_diff empl_diff_rank
##   <chr>             <dbl>            <int>     <dbl>          <int>
## 1 Alabama            59.9               51      73.7             10
## 2 Mississippi        60.5               50      41.9             30
## 3 Louisiana          62.7               49      29.6             41
## 4 Utah               65.2               48      81.1              6
## 5 Tennessee          68.8               47      42.7             27
```

```r
#show 5 biggest increases in median annual salary
diff %>%
  select(state, salary_diff, salary_diff_rank, empl_diff, empl_diff_rank) %>%
  slice_max(order_by = empl_diff, n=5)
```

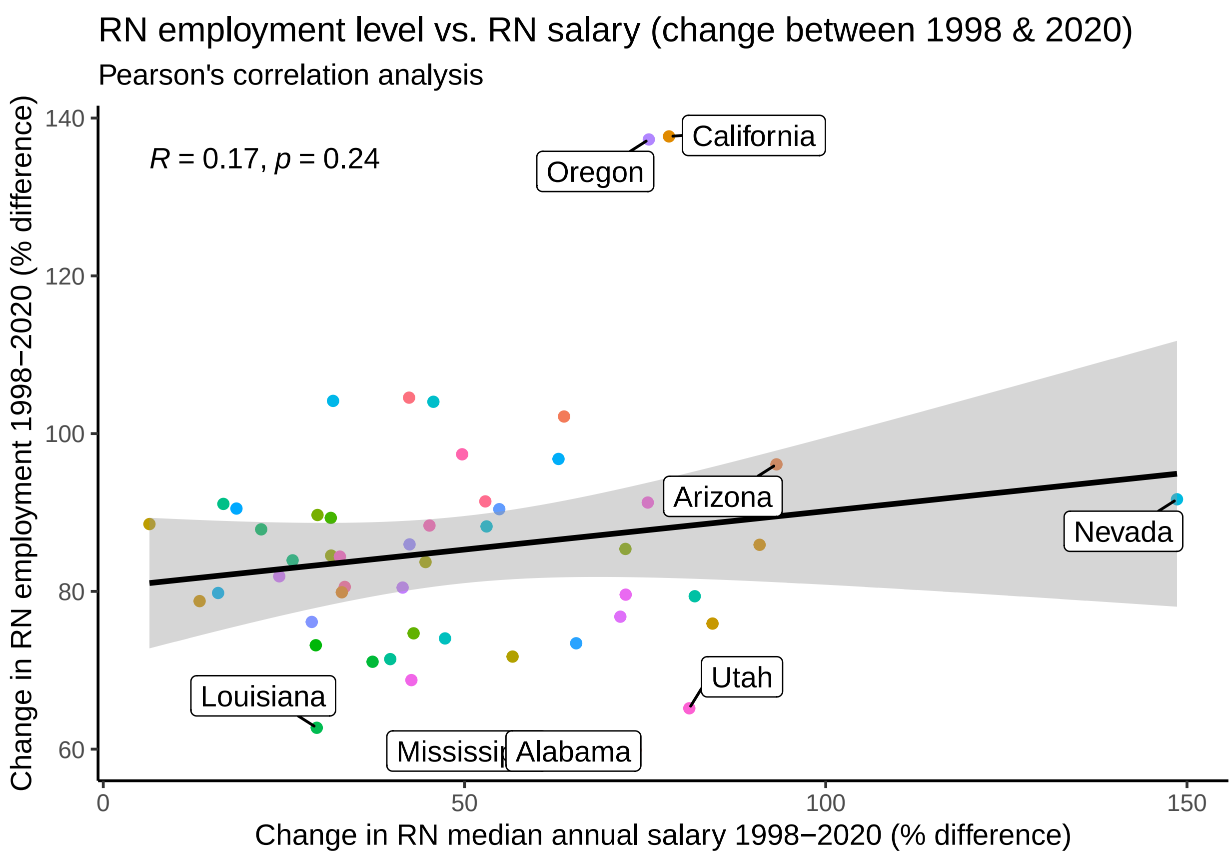
```
## # A tibble: 5 x 5
##   state     salary_diff salary_diff_rank empl_diff empl_diff_rank
##   <chr>           <dbl>            <int>     <dbl>          <int>
## 1 Nevada           91.7               10     149.               1
## 2 Arizona          96.1                9      93.2              2
## 3 Colorado         85.9               23      90.8              3
## 4 Delaware         75.9               39      84.3              4
## 5 Minnesota        79.4               35      81.9              5
```

```r
 #show 5 smallest increases in median annual salary
diff %>%
  select(state, salary_diff, salary_diff_rank, empl_diff, empl_diff_rank) %>%
  slice_min(order_by = empl_diff, n=5)
```

```
## # A tibble: 5 x 5
##   state                salary_diff salary_diff_rank empl_diff empl_diff_rank
##   <chr>                      <dbl>            <int>     <dbl>          <int>
## 1 District of Columbia        88.5               18      6.39             51
## 2 Connecticut                 78.8               36     13.3              50
## 3 New Jersey                  79.8               33     15.9              49
## 4 Massachusetts               91.1               13     16.6              48
## 5 New York                    90.5               14     18.4              47
```

To visualize the relevant data to answer this research question, I plotted the percent differences in total RN employment and median annual salary against each other. A linear regression and Pearson's correlation test help to mathematically determine if the two metrics are linked. **Based on these results, it does not appear that changes in RN salaries are associated with changes in RN employment levels.**

```r
# plot percent differences in employment vs salary
diff %>%
  ggplot(aes(x = empl_diff, y = salary_diff)) +
  geom_point(aes(color = state), show.legend = FALSE) +
  geom_smooth(method = lm, # linear regression
              color = "black",
              show.legend = FALSE) +
  geom_label_repel(aes(label = state), # label some outlier points
            min.segment.length = 0,
            max.overlaps = 3) +
  stat_cor(method = "pearson") + # annotate with correlation analysis statistics
  labs(title = "RN employment level vs. RN salary (change between 1998 & 2020)",
       subtitle = "Pearson's correlation analysis",
       x = "Change in RN median annual salary 1998-2020 (% difference)",
       y = "Change in RN employment 1998-2020 (% difference)") +
  theme_classic()
```



These data can be plotted in a slightly different way, using the rank order of the changes in employment level and salary. In this case, I used Spearman's correlation analysis because the data in this plot are on an ordinal scale, not an interval scale. I think this is a good "sanity check" to help validate that the results of the original analysis make sense and that nothing went wrong on a technical level.

```r
# plot rank of percent differences in employment vs salary

diff %>%
  ggplot(aes(x = empl_diff_rank, y = salary_diff_rank)) +
```

```r
geom_point(aes(color = state), show.legend = FALSE) +
geom_smooth(method = lm, # plot linear regression line
            color = "black",
            show.legend = FALSE) +
stat_cor(method = "spearman") + # annotate with correlation analysis statistics
labs(title = "RN employment level vs. RN salary (rank of change)",
     subtitle = "Spearman's correlation analysis",
     x = "Rank of change in RN median annual salary 1998-2020 (% difference)",
     y = "Rank of change in RN employment 1998-2020 (% difference)") +
theme_classic()
```