Data Task

Peichen Li

Data Task

Data Preparation

First we take a look at the dataset we are interested in.

```
setwd("C:/Users/90596/Desktop")
df = read_csv("test_data.txt")
glimpse(df)
```

```
#> Rows: 8,831
#> Columns: 6
#> $ ed_tc     <chr> "17may1982 14:47:00", "14jul1982 17:49:00", "30jun1982 13...
#> $ dcord_tc  <chr> "18may1982 10:49:00", "14jul1982 18:51:00", "30jun1982 14...
#> $ xb_lntdc  <dbl> 0.4086, 0.3384, 0.3097, 0.5928, 1.1748, 1.0945, 0.9509, 1...
#> $ shiftid   <chr> "17may1982 1 p.m. to 10 p.m.", "14jul1982 1 p.m. to 10 p....
#> $ phys_name <chr> "Andrew", "Andrew", "Andrew", "Andrew", "Andrew", "Andrew...
#> $ visit_num <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17...
```

Now we need to do some data cleaning

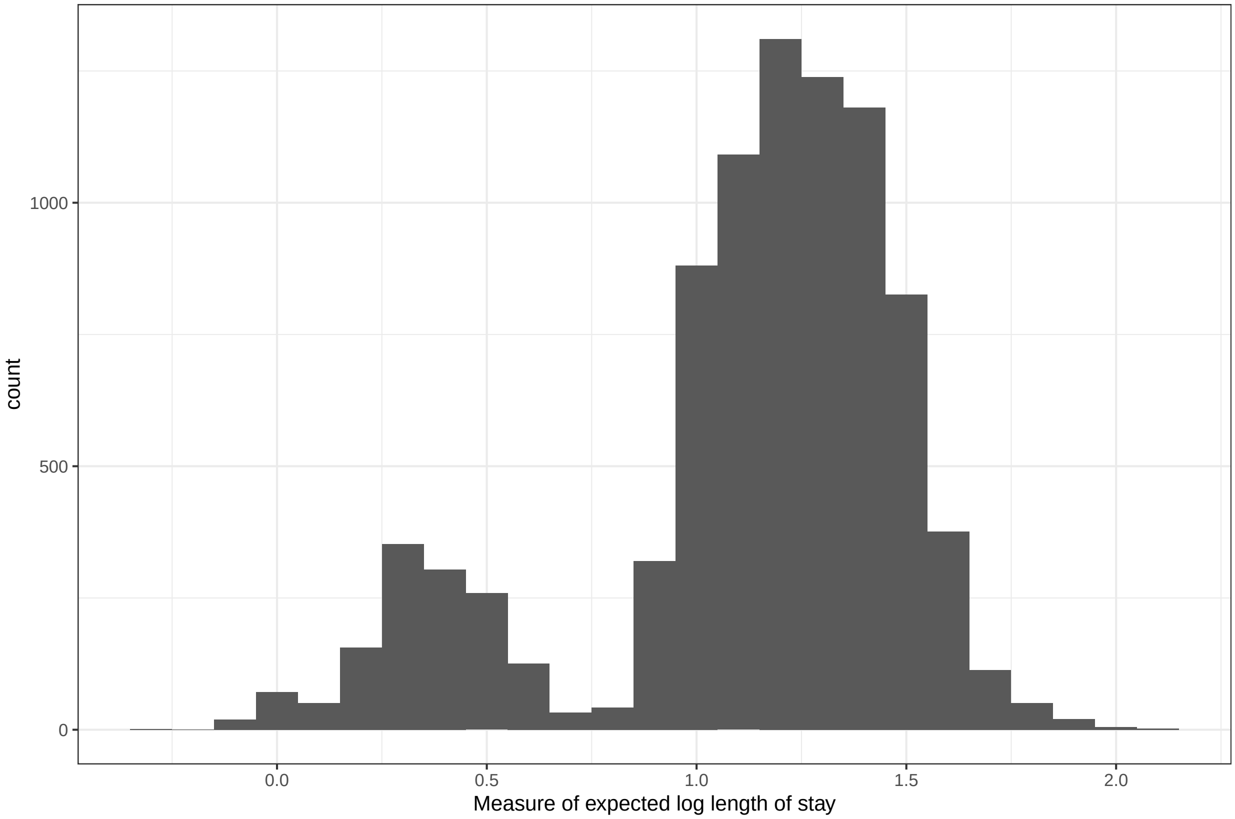
```
num_visits = nrow(df %>% distinct(visit_num))
num_visits
```

```
[1] 8831
```

```
phys_num <- length(unique(df$phys_name))
phys_num
```

```
[1] 43
```

```
# Data cleaning for patient arrival and discharge time
df = df %>%
  mutate( pat_arr = dmy_hms(ed_tc), pat_dis = dmy_hms(dcord_tc) ) %>%
  mutate( pat_duration = as.numeric(difftime(pat_dis, pat_arr,
                                             units = "hours")) )
# replace each occurrence of "noon" with an  string "12 p.m."
df$shiftid = gsub("noon", "12 p.m.", df$shiftid)
df$shiftid = gsub("p.m.", "pm", df$shiftid)
df$shiftid = gsub("a.m.", "am", df$shiftid)
# Create a histogram for expected log length of patient stay
df %>%
  ggplot(aes(x=xb_lntdc)) +
  geom_histogram(bins=25) +
  xlab("Measure of expected log length of stay") +
  theme_bw()
```



```
# Data cleaning for shiftid
df1 = df %>%
  mutate( shift_start = dmy_h( str_c(str_extract(shiftid,
                                  pattern = "[0-9]+[a-z]+[0-9]+ "),
                                  str_extract(df$shiftid,
                                  pattern = " [0-9]+ [ap]m ")) ))  %>%
  mutate( shift_end0 = dmy_h( str_c(str_extract(shiftid,
                                  pattern = "[0-9]+[a-z]+[0-9]+ "),
                                  str_sub(df$shiftid, - 7, - 1) ) )) %>%
  mutate( shift_end = as_datetime(ifelse( shift_start < shift_end0,
                                  shift_end0, shift_end0 + days(1) ) )) %>%
  mutate( shift_duration = as.numeric(difftime(shift_end, shift_start,
                                  units = "hours") ) ) %>%
  select( visit_num, shiftid, phys_name, pat_arr, pat_dis,
          xb_lntdc, pat_duration, shift_start, shift_end, shift_duration)
# Summarize the data
stargazer(as.data.frame(df1 %>% select(pat_duration, shift_duration, xb_lntdc)),
          omit.summary.stat = c("p25", "p75"),
          header = FALSE,
          title = 'Summary Statistics for the data',
          digits = 2)
```

Question 0

From the summary statistics, we can see that there are 43 different physicians and 8831 patient visits (observations), during three months (May, June, July) in 1982. From the distribution of expected log length

Table 1: Summary Statistics for the data

Statistic	N	Mean	St. Dev.	Min	Max
pat_duration	8,831	4.04	4.22	−1.82	36.85
shift_duration	8,831	9.10	0.36	2	10
xb_lntdc	8,831	1.12	0.38	−0.28	2.12

of stay, we can see that there are two main data centers. The center with more frequency and higher value in log length of stay might refer to patients with major illnesses, and the other with less frequency and lower value may refer to patients with minor illnesses. There is more probability mass in major illnesses. The mean log length of stay is 1.12 and the mean time of shift of physicians is 9.10 hours, and the mean patient duration is 4.04 hoours.

Also from the summary table, we can see that there are some negative values of patient duration, which should be logically incorrect.

```r
sum(df1$pat_duration < 0)
```

```
#> [1] 3
```

```r
df1 %>% arrange(pat_duration) %>% head(5)
```

```
#> # A tibble: 5 x 10
#>   visit_num shiftid phys_name pat_arr             pat_dis             xb_lntdc
#>       <dbl> <chr>   <chr>     <dttm>              <dttm>                 <dbl>
#> 1      8505 14jul1~ Oprah     1982-07-14 18:50:00 1982-07-14 17:01:00    1.03
#> 2      7911 23may1~ Ingrid    1982-05-23 14:59:00 1982-05-23 13:59:00    1.14
#> 3      3270 23jun1~ Diana     1982-06-23 15:43:00 1982-06-23 15:25:00    0.863
#> 4      2714 23jun1~ Barack    1982-06-23 15:44:00 1982-06-23 15:44:00    0.970
#> 5      2539 13jun1~ Barack    1982-06-13 07:40:00 1982-06-13 07:43:00    1.42
#> # ... with 4 more variables: pat_duration <dbl>, shift_start <dttm>,
#> #   shift_end <dttm>, shift_duration <dbl>
```

After analysis, the results suggest that there are 3 data entry errors concerning with patient arrival and discharge time, which each is located at visit number 8505, 7911 or 3270, respectively.

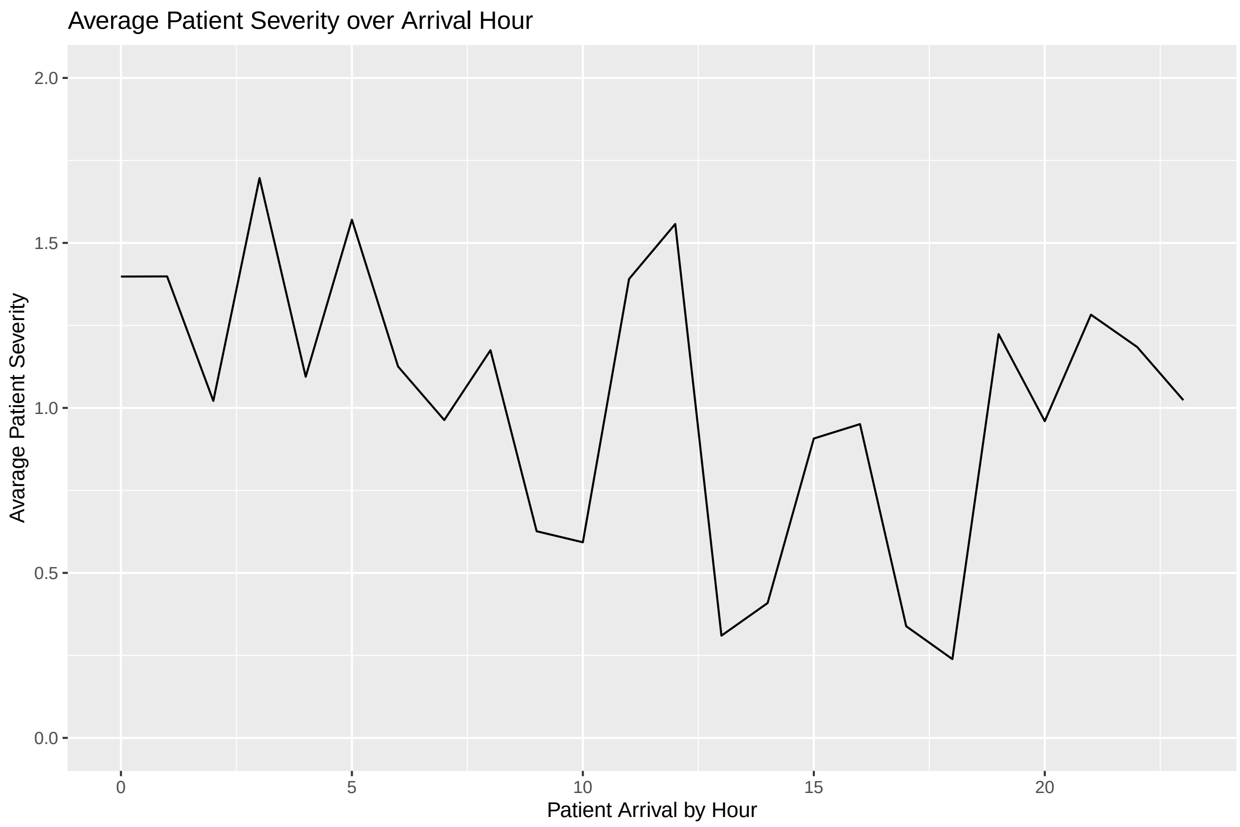
Question 1

```r
df = df1 %>%
  mutate( ArrEarDummy = ifelse(pat_arr < shift_start, 100, 0) ) %>%
  mutate( StayLateDummy = ifelse(pat_dis > shift_end, 100, 0) )
# Some patients may arrive before their physician's shift starts
# and therefore would have to wait
ArrEarPercentage = mean(df$ArrEarDummy)
# Other patients may be discharged after their physician's shift ends
StayLatePercentage = mean(df$StayLateDummy)
```

Percentage of patients arrive before the shift starts: 7.3604
Percentage of patients discharge after the end of the shift: 18.9673

Question 2

```r
df = df %>% mutate( pat_arr_hour = hour(pat_arr) ) %>%
  group_by(pat_arr_hour) %>%
  mutate( PatSevHr = mean(pat_arr_hour) ) %>%
  slice(1)

g1 <- ggplot(data = df) +
  geom_line( aes(x = PatSevHr, y = xb_lntdc) ) +
  ylim(0,2) +
  labs( x ="Patient Arrival by Hour", y="Avarage Patient Severity", title=
        "Average Patient Severity over Arrival Hour")

g1
```



From the graph, we can see that there are some kinds of fluctuation of patient severity by hour. In particular, patients who arrive around 1pm and 6pm at ED seem to be the least severe. And patients who arrive around 3 am and 12 pm seem to be the most severe ones.

To test whether patient severity is or is not predicted by hour of the day, I would first create 23 variables for the first 23 hour intervals in each day. And then regress patient severity on these dummy variables. After obtaining the estimated coefficients, I would test the overall significance of the model to reach our conclusion (use F test).

Question 3

```
df <-df1 %>%
  mutate(diff_time = as.numeric(difftime(pat_dis,shift_end,units="hours"))) %>%
# compare patient discharge time with shift times
  filter(diff_time < 4) %>%
```

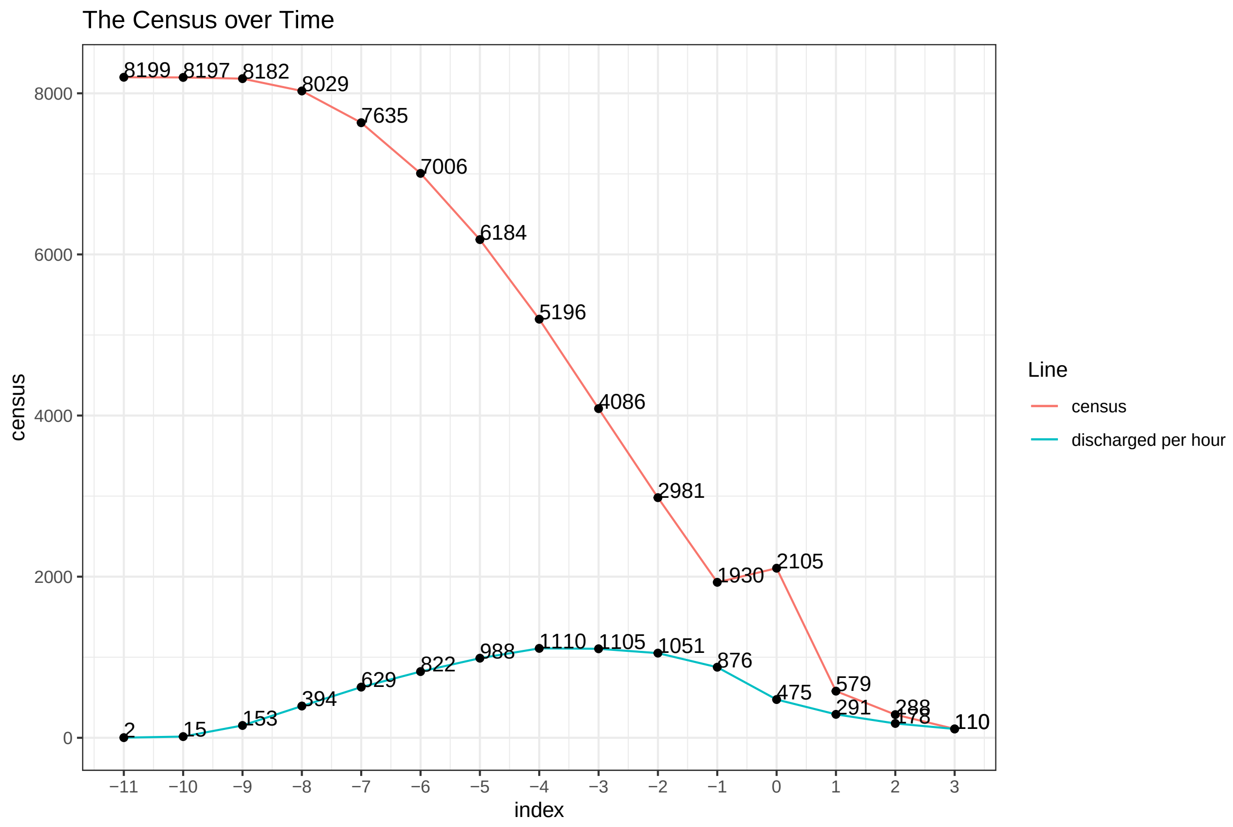
```r
# the extension of stay should not be greater than 4
  mutate(index = floor(diff_time)) %>%
  select(shiftid, phys_name, index)
# Output the "census" data set as census.txt
write_csv(df, file = "census.txt",col_names = TRUE)

df <- df %>%
  mutate(num = 1) %>%
  group_by(index) %>%
  mutate(discharged = sum(num)) %>%
  slice(1) %>%
  select(index, discharged) %>%
  ungroup()

df
```

```
#> # A tibble: 15 x 2
#>    index discharged
#>    <dbl>      <dbl>
#>  1   -11          2
#>  2   -10         15
#>  3    -9        153
#>  4    -8        394
#>  5    -7        629
#>  6    -6        822
#>  7    -5        988
#>  8    -4       1110
#>  9    -3       1105
#> 10    -2       1051
#> 11    -1        876
#> 12     0        475
#> 13     1        291
#> 14     2        178
#> 15     3        110
```

```r
df$census <- c(8199,8199-2, 8199-2-15, 8199-2-15-153, 8199-2-15
               -153-394, 8199-2-15-153-394-629, 8199-2-15-153-394-629
               -822, 8199-2-15-153-394-629-822-988, 8199-2-15-153-394
               -629-822-988-1110, 8199-2-15-153-394
               -629-822-988-1110-1105, 8199-2-15-153-394
               -629-822-988-1110-1105-1051,8199-2-15-153-394
               -629-822-988-1110-1105-876, 291+178+110, 110+178, 110)
df
```

```
#> # A tibble: 15 x 3
#>    index discharged census
#>    <dbl>      <dbl>  <dbl>
#>  1   -11          2   8199
#>  2   -10         15   8197
#>  3    -9        153   8182
#>  4    -8        394   8029
#>  5    -7        629   7635
#>  6    -6        822   7006
#>  7    -5        988   6184
#>  8    -4       1110   5196
#>  9    -3       1105   4086
#> 10    -2       1051   2981
#> 11    -1        876   1930
#> 12     0        475   2105
#> 13     1        291    579
#> 14     2        178    288
#> 15     3        110    110
```

```r
g2 <- ggplot(df, aes(x=index) ) +
  geom_line(aes(y=census, color = "black")) +
  geom_line(aes(y=discharged, color = "blue")) +
  geom_point(aes(y=census)) +
  geom_point(aes(y=discharged)) +
  geom_text(aes(y=census, label=census),hjust=0, vjust=0, na.rm=TRUE) +
  geom_text(aes(y=discharged, label=discharged),hjust=0, vjust=0, na.rm=TRUE) +
  scale_x_continuous(limits = c(-11, 3), breaks = seq(-11, 3, by = 1)) +
  theme_bw() +
  ggtitle("The Census over Time") +
  scale_color_discrete(name = "Line", labels = c("census", "discharged per hour"))

g2
```



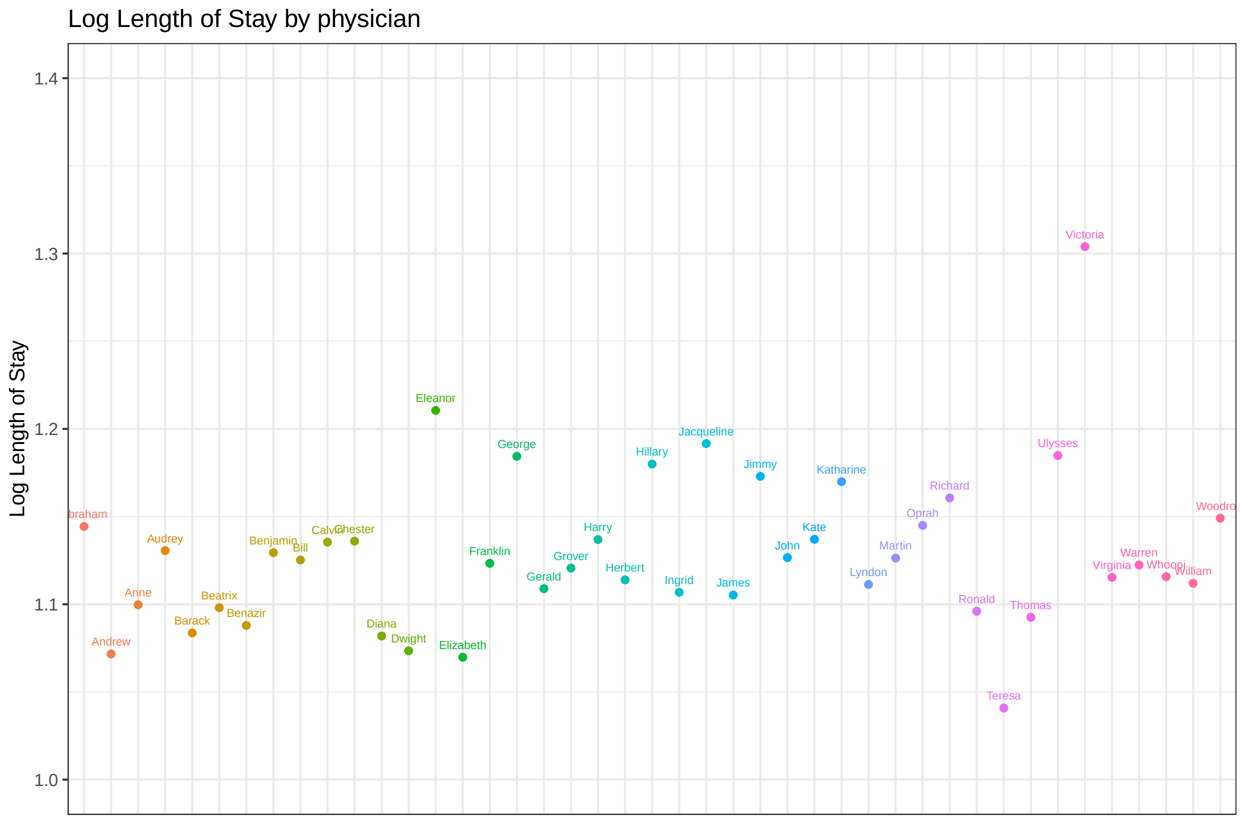
The graph shows that with time relative to end of shift, the number of patients generally tend to decrease and down to around zero eventually. However, note that from the hour index -1 to 0, the amount of patients under care seems to increase a bit. This abnormality suggests that the number of patients arriving at that hour is great than the number of patients discharged at the same time.

The idea of constructing the “census”: First I create a new column indicating the difference between patient discharge time and shift ending time. Then I filter out observations with difference time greater than four hours. Then we take the integer part of the difference time as the hour index. Grouping the index, we can get the number of discharged patients per hour. By calculating the remaining patient for each hour index, we can successfully get the census.

Since the hour index is a discrete variable, the conclusions we drew above are only consistent in an hourly pattern. For example, there might be results contradicting to the general trend during each hour, as it’s possible to have peaks between two hour indexes.

Question 4

```
df <- df1 %>%
    group_by(phys_name) %>%
    mutate(LogLenStay = mean(xb_lntdc)) %>%
    slice(1) %>%
    select(LogLenStay, phys_name) %>%
    arrange(desc(LogLenStay))


g3 <- ggplot(df, aes(x=phys_name, y=LogLenStay,color=phys_name,
                     label = phys_name)) +
    geom_point(na.rm=TRUE) +
    geom_text(aes(label = phys_name), hjust = 0.5,  vjust = -1, size = 2) +
    theme_bw() +
    theme(legend.position="none") +
    ylab("Log Length of Stay") +
    ylim(1,1.4) +
    ggtitle("Log Length of Stay by physician") +
    theme(legend.title=element_blank()) +
    theme(axis.title.x=element_blank(),
          axis.text.x=element_blank(),
          axis.ticks.x=element_blank())
g3
```



```
model_1 <- df1 %>% lm(xb_lntdc~phys_name, data = .)

stargazer(model_1,
          title = 'Regression Results',
          single.row = TRUE,
          header = FALSE)
```

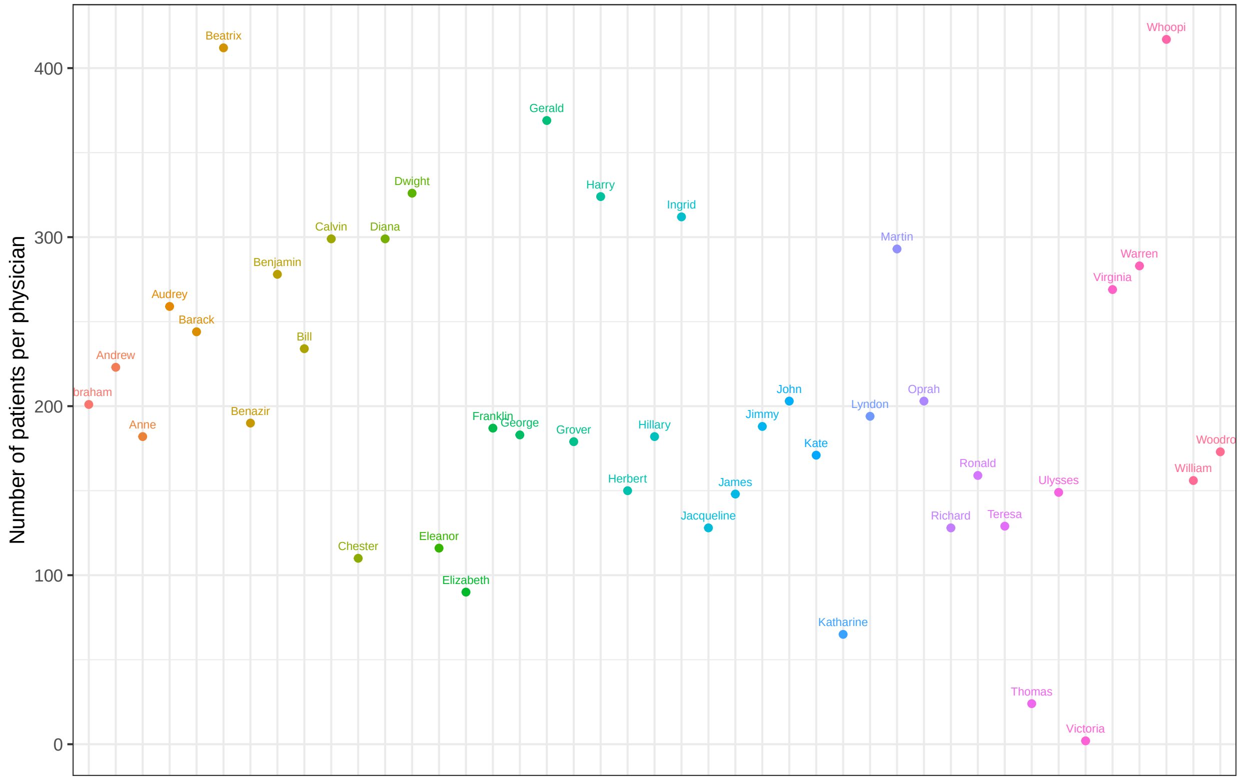
```
df <- df1 %>%
  group_by(phys_name) %>%
  mutate(num = 1) %>%
  mutate(pat_num_per_phys = sum(num)) %>%
  slice(1) %>%
  select(pat_num_per_phys,phys_name) %>%
  arrange(desc(pat_num_per_phys))

df %>%
  ggplot(aes(x=phys_name, y=pat_num_per_phys,color=phys_name,
             label = phys_name)) +
  geom_point(na.rm=TRUE) +
  geom_text(aes(label = phys_name), hjust = 0.5,  vjust = -1, size = 2) +
  theme_bw() +
  theme(legend.position="none") +
  ylab("Number of patients per physician") +
  ggtitle("Number of patients by physician") +
  theme(legend.title=element_blank()) +
  theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
```

Table 2: Regression Results

	Dependent variable:
	xb_lntdc
phys_nameAndrew	-0.073^{**} (0.037)
phys_nameAnne	-0.045 (0.039)
phys_nameAudrey	-0.014 (0.036)
phys_nameBarack	-0.061^{*} (0.036)
phys_nameBeatrix	-0.046 (0.033)
phys_nameBenazir	-0.056 (0.039)
phys_nameBenjamin	-0.015 (0.035)
phys_nameBill	-0.019 (0.037)
phys_nameCalvin	-0.009 (0.035)
phys_nameChester	-0.008 (0.045)
phys_nameDiana	-0.062^{*} (0.035)
phys_nameDwight	-0.071^{**} (0.034)
phys_nameEleanor	0.066 (0.044)
phys_nameElizabeth	-0.075 (0.048)
phys_nameFranklin	-0.021 (0.039)
phys_nameGeorge	0.040 (0.039)
phys_nameGerald	-0.035 (0.033)
phys_nameGrover	-0.024 (0.039)
phys_nameHarry	-0.007 (0.034)
phys_nameHerbert	-0.030 (0.041)
phys_nameHillary	0.036 (0.039)
phys_nameIngrid	-0.038 (0.034)
phys_nameJacqueline	0.047 (0.043)
phys_nameJames	-0.039 (0.041)
phys_nameJimmy	0.029 (0.039)
phys_nameJohn	-0.018 (0.038)
phys_nameKate	-0.007 (0.040)
phys_nameKatharine	0.026 (0.054)
phys_nameLyndon	-0.033 (0.038)
phys_nameMartin	-0.018 (0.035)
phys_nameOprah	0.001 (0.038)
phys_nameRichard	0.016 (0.043)
phys_nameRonald	-0.048 (0.040)
phys_nameTeresa	-0.104^{**} (0.043)
phys_nameThomas	-0.052 (0.082)
phys_nameUlysses	0.040 (0.041)
phys_nameVictoria	0.160 (0.271)
phys_nameVirginia	-0.029 (0.036)
phys_nameWarren	-0.022 (0.035)
phys_nameWhoopi	-0.029 (0.033)
phys_nameWilliam	-0.032 (0.041)
phys_nameWoodrow	0.005 (0.040)
Constant	1.144^{***} (0.027)
Observations	8,831
R^2	0.007
Adjusted R^2	0.002
Residual Std. Error	0.381 (df = 8788)
F Statistic	1.489^{**} (df = 42; 8788)
Note:	*p<0.1; **p<0.05; ***p<0.01

```
        axis.ticks.x=element_blank())
```

Number of patients by physician



From the graph and the regression table, we see that physician Teresa is the fastest at discharging patients. For this regression model, I only control for *phys_name* variable and fail to include *index* variable. By Question_3, it may be possible that patient discharging speed may be correlated with the hour index of patient arrival.

Potential threats: 1) the *phys_name* variable may be too weak to explain the discharging speed since the R squared is very close to zero. 2) Omitted variable bias: as discussed above, we fail to include the *index* variable, which leaves our estimated coefficients biased. 3) Invalid F-test and t-test since we did not test for heteroskedasticity. This point may also suggest that our estimates may not be robust. Regarding the number of patients per physician, we see that physician Victoria only has few observations. We may take observations with Victoria as outliers.

References

- RA Data Task from DChan Lab (2020), SIEPR.