Homework 1

Animesh Sengupta

9/18/2022

```
# Note that `#' is the comment character, and makes this line a comment instead of code

# Always make sure you are in the right working directory
# When you click on an .Rmd file to open it, RStudio should automatically open in the directory where t
# Note that when you `knit' a file, it will automatically treat the directory where it lives as the wor
# You can change the working directory through the menus, then compy the line of code here so you have
setwd("/Users/animeshsengupta/Work Directory/DACSS/STAT625/Homeworls")  ## replace this with your direc
# Next, we should load the packages we will use.  You only need to install them once:
# install.packages("alr4")
library(alr4)  # loads the installed package into the workspace so you can use it
```

```
## Loading required package: car
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
## The following object is masked from 'package:purrr':
##
##     some
```

```
## Loading required package: effects
```

```
## lattice theme set by effectsTheme()
## See ?effectsTheme for details.
```

```
# If you don't have LaTeX installed on your comupter, install tinytex by uncommenting and running the 2
# install.packages("tinytex")
# tinytex::install_tinytex()
```

1

I had chosen ggplot2 as a tutorial, this tutorial was part of DACSS 601 - Fundamentals of data science. This ggplot tutorial was made by DACSS course administrator under the advise of Dr Meredith rolfe. I am reusing the same tutorials for this section as the homework has things to do with graphs. Here is the code chunk which takes us through real world tidying techniques and then makes us to plot the graph.

```
#! label: Data loading
#| warning: false
US_household_data <- read_excel("../Homeworls/_data/USA Households by Total Money Income, Race, and His

head(US_household_data,5)
```

```
## # A tibble: 5 x 16
##   Year      Number Total pd_<1~1 pd_15~2 pd_25~3 pd_35~4 pd_50~5 pd_75~6 pd_10~7
##   <chr>     <chr>  <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
## 1 ALL RACES <NA>      NA    NA      NA      NA      NA      NA      NA      NA
## 2 2019      128451   100     9.1     8       8.3    11.7    16.5    12.3    15.5
## 3 2018      128579   100    10.1     8.8     8.7    12      17      12.5    15
## 4 2017 2    127669   100    10       9.1     9.2    12      16.4    12.4    14.7
## 5 2017      127586   100    10.1     9.1     9.2    11.9    16.3    12.6    14.8
## # ... with 6 more variables: ‘pd_150000-199999‘ <dbl>, ‘pd_>200000‘ <dbl>,
## #   median_income_estimate <dbl>, median_income_moe <dbl>,
## #   mean_income_estimate <chr>, mean_income_moe <chr>, and abbreviated variable
## #   names 1: ‘pd_<15000‘, 2: ‘pd_15000-24999‘, 3: ‘pd_25000-34999‘,
## #   4: ‘pd_35000-49999‘, 5: ‘pd_50000-74999‘, 6: ‘pd_75000-99999‘,
## #   7: ‘pd_100000-149999‘
## # i Use ‘colnames()‘ to see all variable names
```

```
#! label: Data processing
#| warning: false
US_processed_data <- US_household_data%>%
  rowwise()%>% #to ensure the following operation runs row wise
  mutate(Race=case_when(
    is.na(Number) ~ Year
  ))%>%
  ungroup()%>% # to stop rowwise operation
  fill(Race,.direction = "down")%>%
  subset(!is.na(Number))%>%
  rowwise()%>%
  mutate(
    Year=strsplit(Year,' ')[[1]][1],
    Race=ifelse(grepl("[0-9]", Race ,perl=TRUE)[1],strsplit(Race," \\s*(?=[^ ]+$)",perl=TRUE)[[1]][1],Ra
    mean_income_estimate=as.numeric(mean_income_estimate),
    Number=as.numeric(Number),
    Year=as.numeric(Year)
  )%>%
  pivot_longer(
    cols = starts_with("pd"),
    names_to = "income_range",
    values_to = "percent_distribution",
    names_prefix="pd_"
  )
```

```
## Warning in mask$eval_all_mutate(quo): NAs introduced by coercion

## Warning in mask$eval_all_mutate(quo): NAs introduced by coercion
```

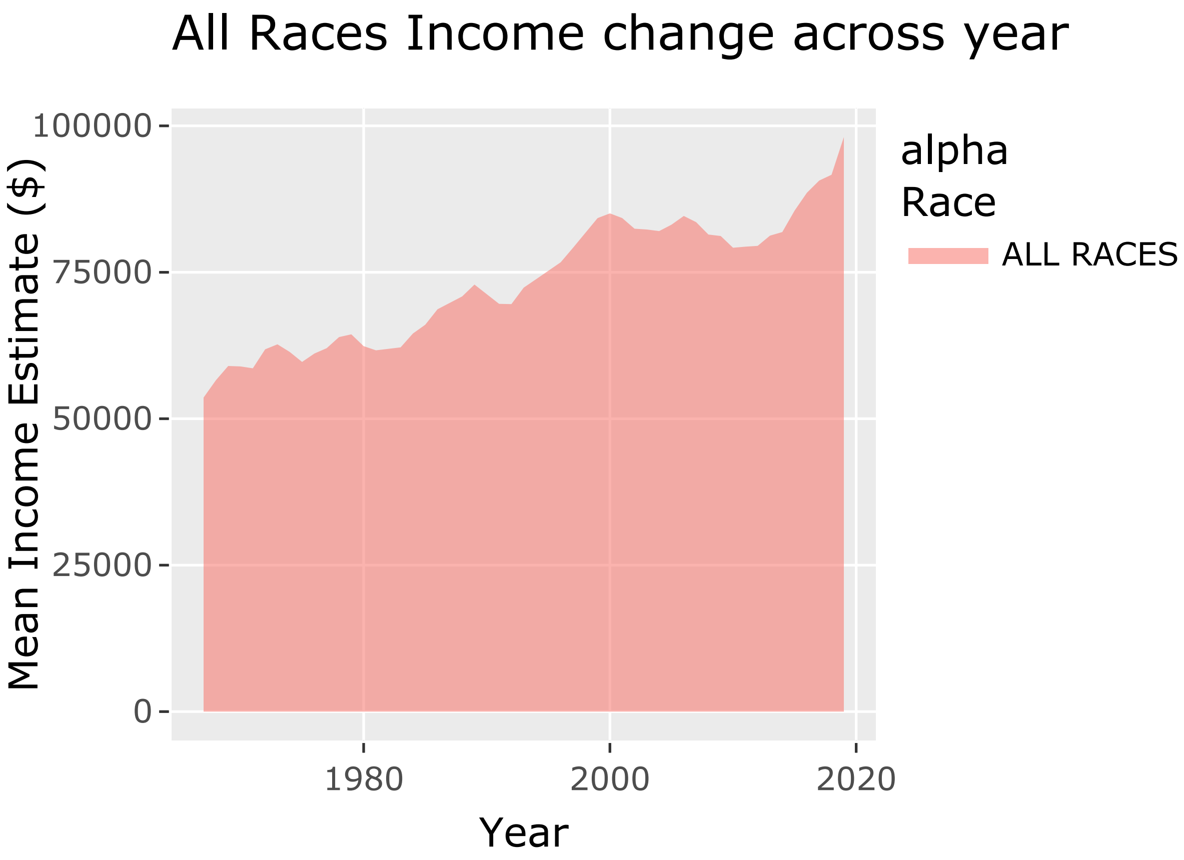
```
US_mean_income_data<-US_processed_data%>%
  select(Year,mean_income_estimate,Race)%>%
  group_by(Race,Year)%>%
  summarize(race_mean_income_estimate=mean(mean_income_estimate))
```

```
## `summarise()` has grouped output by 'Race'. You can override using the
## `.groups` argument.
```

```
grouped_race<- US_mean_income_data%>%
  mutate(CombinedRace=case_when(
    str_detect(Race,"ASIAN")~"ASIAN",
    str_detect(Race,"BLACK")~"BLACK",
    str_detect(Race,"WHITE")~"WHITE",
    TRUE ~ Race
  ))
head(grouped_race,5)
```

```
## # A tibble: 5 x 4
## # Groups:   Race [1]
##   Race       Year race_mean_income_estimate CombinedRace
##   <chr>     <dbl>                     <dbl> <chr>
## 1 ALL RACES  1967                     53616 ALL RACES
## 2 ALL RACES  1968                     56572 ALL RACES
## 3 ALL RACES  1969                     59004 ALL RACES
## 4 ALL RACES  1970                     58926 ALL RACES
## 5 ALL RACES  1971                     58609 ALL RACES
```

```
race_income_area <- ggplot(grouped_race%>%filter(CombinedRace=="ALL RACES"),aes(x=Year,y=race_mean_inco
  geom_area() +
  labs(title="All Races Income change across year",
   x="Year", y="Mean Income Estimate ($)")

ggplotly(race_income_area)
```



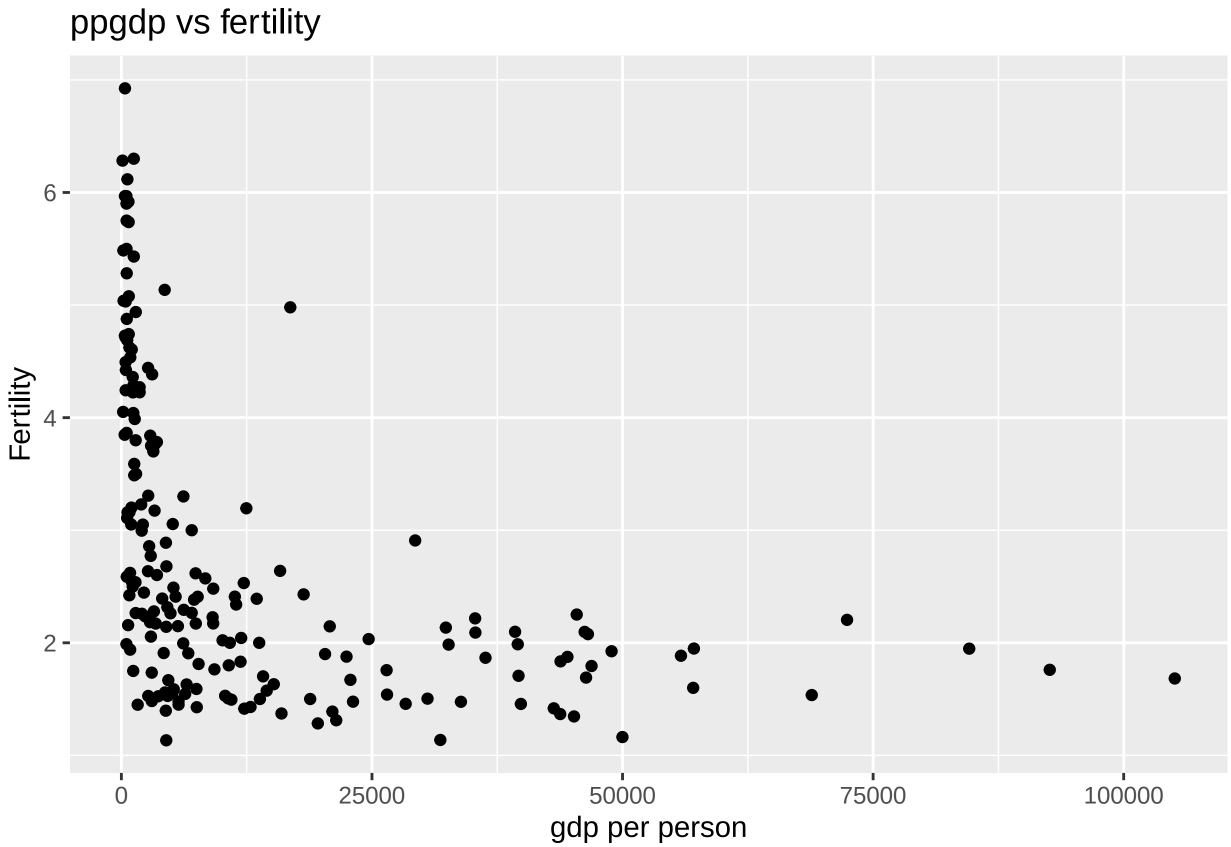
2

2a

predictor - ppgdp response - fertility

2b

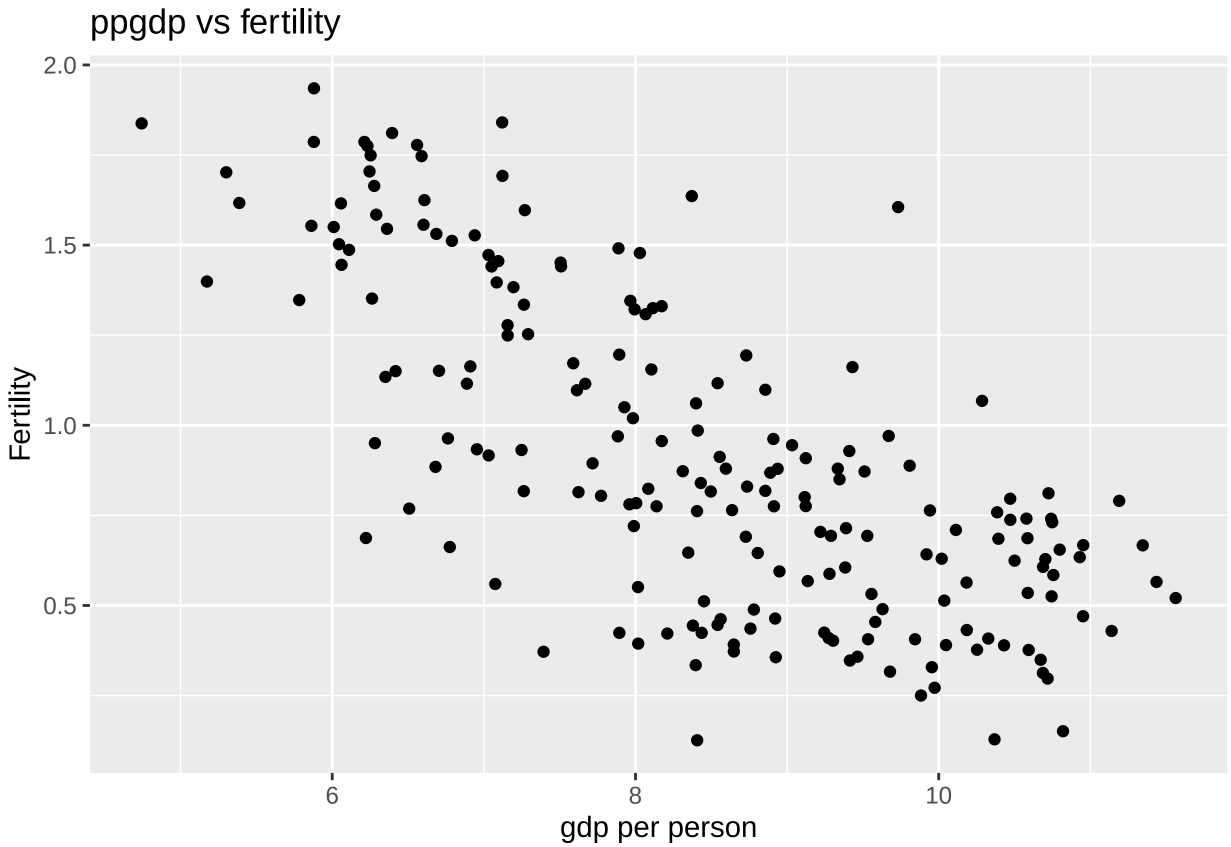
```
# modify the below code to use the appropriate dataset and make a scatterplot of the approporiate value

# Note that the example code in this document is mostly from the textbook supplement:
# http://users.stat.umn.edu/~sandy/alr4ed/links/alrprimer.pdf
# you can find similarly helpful code there for future homeworks

b2<- ggplot(UN11,aes(x=ppgdp, y=fertility)) +
    geom_point()+
    xlab("gdp per person")+
    ylab("Fertility")+
    ggtitle("ppgdp vs fertility")
b2
```



As seen in the graph , a straight line wouldnt be able to propely model/fit the summary of scatter plot.

2c

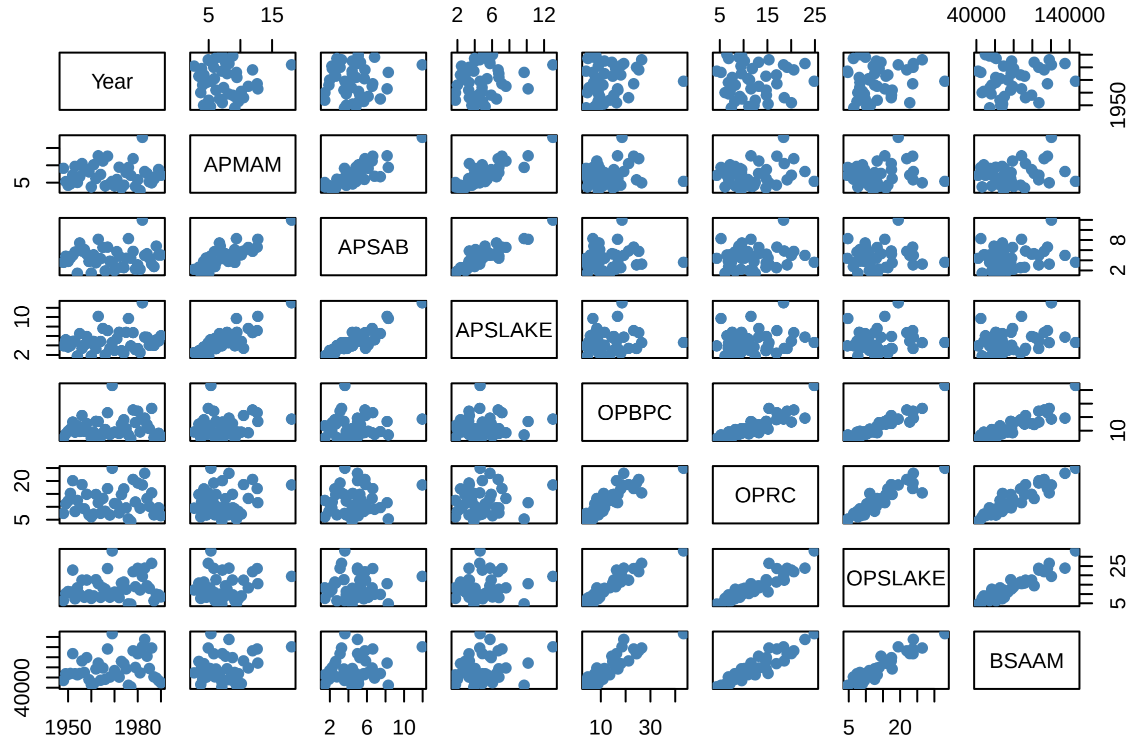
```
# modify the below code to use the appropriate dataset and make a scatterplot of the approporiate value

#plot(dheight ~ log(mheight), data=Heights, xlab="Log Mothers' Heights", ylab="Daughters' Heights")

b3<- ggplot(UN11,aes(x=log(ppgdp), y=log(fertility))) +
    geom_point()+
    xlab("gdp per person")+
    ylab("Fertility")+
    ggtitle("ppgdp vs fertility")
b3
```



A simple linear regression model seems plausible for this graph and may have a negative slope as well.

3

```
#modify this code
plot(water, pch=20, cex=1.5, col='steelblue')
```



Year doesnt seem to have a clear linear relation with most of the other variables. Most of them are pretty much stabilised against a certain value for each progressing year.

We can also conclude a clear linear relation between APMAM with APSAB and APSLAKE. With a positive slope , it may fit well with y=x model.

Similarly, OPBPC, OPRC, OPSLAKE and BSAAM have clear indicative linear relation between them and can be fitted better with linear regrssion of y=x model. This means that whenever one of them increrases, the other rating tends to increase too.

4

As evident from the graph the ratings for parameter quality, clarity and helpfulness shows a clear linear relation with very little variation within each other. We can easily deduce from the graph that the model will generate very little error variance between predicted and actual regressor. The model y=x can best fit , with a positive slope. We can deduce that as either of the ratings increases the other rating would follow suit.

Easiness seems to have very weak relation with all the other rating rubrics as the pairwise scatterplots shows a lot of deviation. RateInterest also seems to have weak relation between the other variables and it would be hard to draw conclusions based on simple observations. Some model may weakly fit its relationship with other variables.

5

5a

$$\hat{\beta}_0 = \bar{y} - \beta_1 \bar{x}$$

5b

$$\hat{\beta}_1 = \frac{\Sigma(x_i - \bar{x})(y_i - \bar{y})}{\Sigma{(x_i - \bar{x})}^2}$$

5c

$$SXX = \Sigma{(x_i - \bar{x})}^2$$

5d

$$SXY = \Sigma(x_i - \bar{x})(y_i - \bar{y})$$

$$SYY = \Sigma{(y_i - \bar{y})}^2$$

$$SXX = \sqrt{\frac{237}{\sum_i x_i}}$$

$$\hat{\sigma}^2 = \frac{\Sigma(y_i - \beta_0 - \beta_1 x_i)^2}{n - df}$$