communication

Overview

Describe the problem. What substantive question are you trying to address? This needn't be long, but it should be clear.

Data and model

What data did you use to address the question, and how did you do it? When describing your approach, be specific. For example:

Don't say, *"I ran a regression"* when you instead can say, *"I fit a linear regression model to predict price that included a house's size and neighborhood as predictors."*

Justify important features of your modeling approach. For example: *"Neighborhood was included as a categorical predictor in the model because Figure 2 indicated clear differences in price across the neighborhoods."*

Sometimes your Data and Model section will contain plots or tables, and sometimes it won't. If you feel that a plot helps the reader understand the problem or data set itself—as opposed to your results—then go ahead and include it. A great example here is Tables 1 and 2 in the main paper on the PREDIMED study. These tables help the reader understand some important properties of the data and approach, but not the results of the study itself.

Results

In your results section, include any figures and tables necessary to make your case. Label them (Figure 1, 2, etc), give them informative captions, and refer to them in the text by their numbered labels where you discuss them. Typical things to include here may include: pictures of the data; pictures and tables that show the fitted model; tables of model coefficients and summaries.

Conclusion

What did you learn from the analysis? What is the answer, if any, to the question you set out to address? i.e., does just rephrase the problem, present **solution(s)** to the problem?

```
data_adult <-read.csv("https://raw.githubusercontent.com/guru99-edu/R-Programming/master/adult.csv")

#This inspects the dataframe and tells us what sort of data each of the columns are
str(data_adult)
```

```
## 'data.frame':    48842 obs. of  10 variables:
##  $ x              : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ age            : int  25 38 28 44 18 34 29 63 24 55 ...
##  $ workclass      : chr  "Private" "Private" "Local-gov" "Private" ...
##  $ education      : chr  "11th" "HS-grad" "Assoc-acdm" "Some-college" ...
##  $ educational.num: int  7 9 12 10 10 6 9 15 10 4 ...
##  $ marital.status : chr  "Never-married" "Married-civ-spouse" "Married-civ-spouse" "Married-civ-spous
##  $ race           : chr  "Black" "White" "White" "Black" ...
```

```
##  $ gender         : chr  "Male" "Male" "Male" "Male" ...
##  $ hours.per.week : int  40 50 40 40 30 30 40 32 40 10 ...
##  $ income         : chr  "<=50K" "<=50K" ">50K" ">50K" ...
```

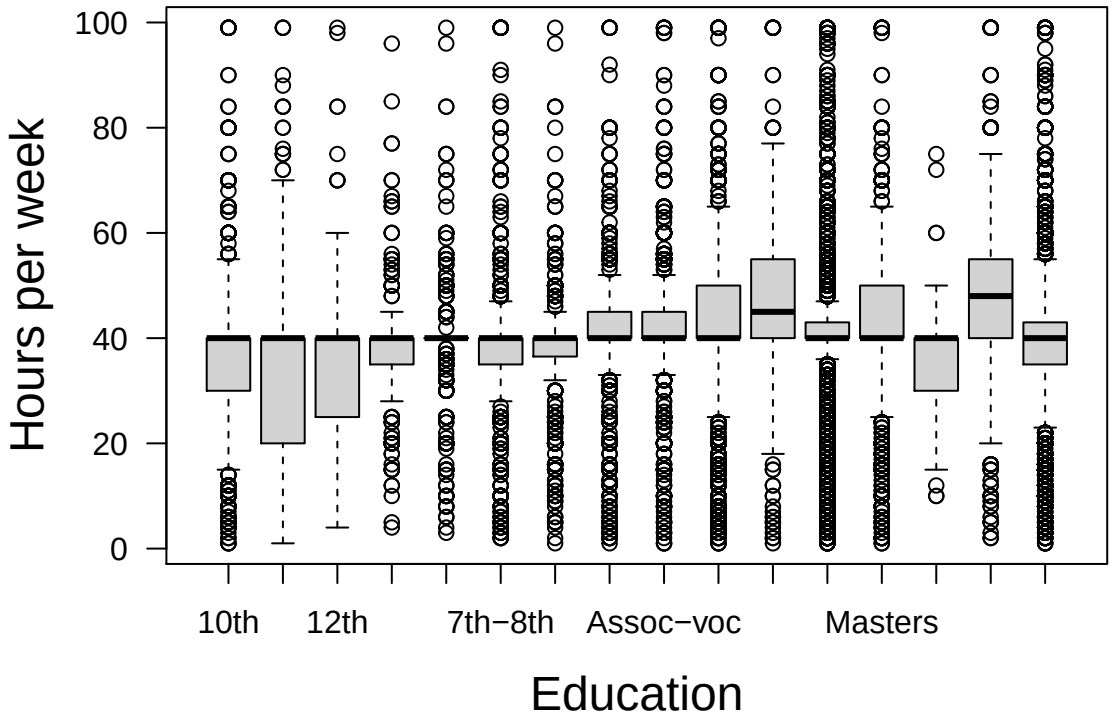
```
# 'data.frame': 48842 obs. of  10 variables:
#  $ x              : int  1 2 3 4 5 6 7 8 9 10 ...
#  $ age            : int  25 38 28 44 18 34 29 63 24 55 ...
#  $ workclass      : chr  "Private" "Private" "Local-gov" "Private" ...
#  $ education      : chr  "11th" "HS-grad" "Assoc-acdm" "Some-college" ...
#  $ educational.num: int  7 9 12 10 10 6 9 15 10 4 ...
#  $ marital.status : chr  "Never-married" "Married-civ-spouse" "Married-civ-spouse" "Married-civ-spous
#  $ race           : chr  "Black" "White" "White" "Black" ...
#  $ gender         : chr  "Male" "Male" "Male" "Male" ...
#  $ hours.per.week : int  40 50 40 40 30 30 40 32 40 10 ...
#  $ income         : chr  "<=50K" "<=50K" ">50K" ">50K" ...

dim(data_adult)
```

```
## [1] 48842    10
```

```
# size of the dataframe - 48842 rows, 10 columns

# DATA EXPLORATION
# Are there missing values?
colSums(is.na(data_adult))
```

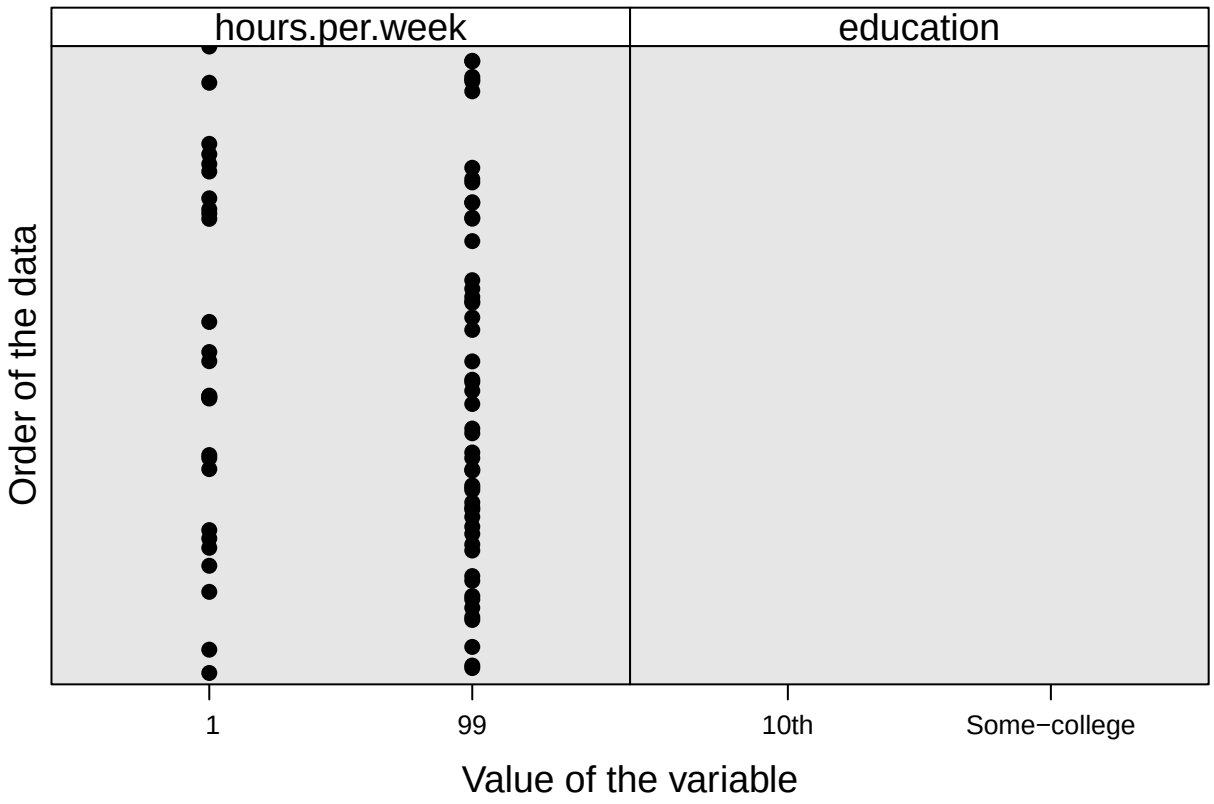
```
##               x             age       workclass       education educational.num
##               0               0               0               0               0
##  marital.status            race          gender  hours.per.week          income
##               0               0               0               0               0
```

```
#    x             age       workclass       education educational.num  marital.status
#    0               0               0               0               0               0
# race          gender  hours.per.week          income
#    0               0               0               0

# OUTLIERS

# Fig 2.1 life
par(mar = c(6,6,2,2), cex.lab = 1.5)
boxplot(hours.per.week ~ education,
        ylab = "Hours per week",
        xlab = "Education",
        data = data_adult,
        las=1)
```

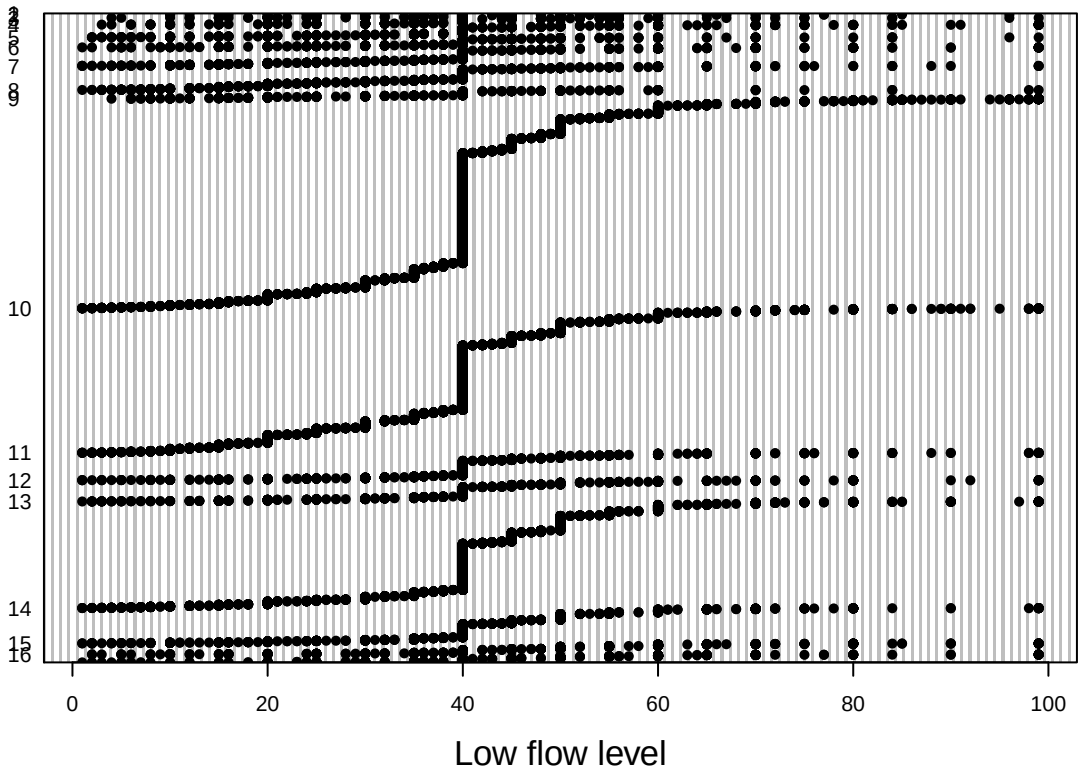



```
# Fig. 2.2 a multi-panel dotchart to see all variables simultaneously.

# First make a vector of variables of interest
Names <- c("hours.per.week", "education")

# Then plot
dotplot(as.matrix(as.matrix(data_adult[,Names])),
        groups=FALSE,
        strip = strip.custom(bg = 'white',
                             par.strip.text = list(cex = 1.2)),
        scales = list(x = list(relation = "free", draw = TRUE),
                      y = list(relation = "free", draw = FALSE)),
        col = 1, cex  = 1, pch = 16,
        xlab = list(label = "Value of the variable", cex = 1.2),
        ylab = list(label = "Order of the data",
                    cex = 1.2))
```

```
## Warning in (function (x, y, horizontal = TRUE, pch = if (is.null(groups))
## dot.symbol$pch else sup.symbol$pch, : NAs introduced by coercion
```




```
# Dotplots to examine each variable separately, grouped by season.
# It shows the order and value of each of the numerical variables.

#Fig 2.3
x <- data_adult[order(data_adult$hours.per.week),]
x$fEco <- factor(x$educational.num)
dotchart(x$hours.per.week, cex = 0.7, pch = 16,
         groups = x$fEco,
         xlab = "Low flow level")
```

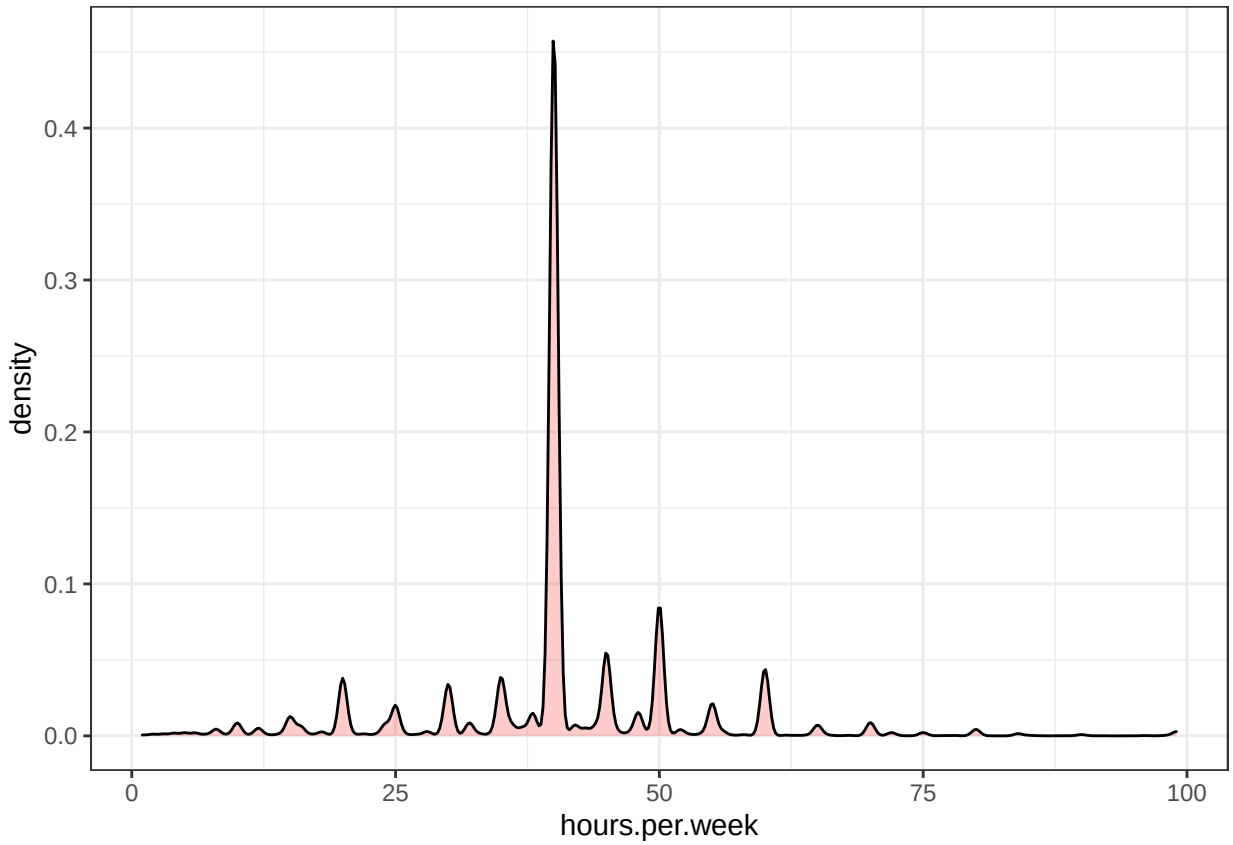


Check continuous variables

```
continuous <-select_if(data_adult, is.numeric)
summary(continuous)
```

```
##        x              age        educational.num hours.per.week
##  Min.   :    1   Min.   :17.00   Min.   : 1.00   Min.   : 1.00
##  1st Qu.:12211   1st Qu.:28.00   1st Qu.: 9.00   1st Qu.:40.00
##  Median :24422   Median :37.00   Median :10.00   Median :40.00
##  Mean   :24422   Mean   :38.64   Mean   :10.08   Mean   :40.42
##  3rd Qu.:36632   3rd Qu.:48.00   3rd Qu.:12.00   3rd Qu.:45.00
##  Max.   :48842   Max.   :90.00   Max.   :16.00   Max.   :99.00
```

```
# Histogram with kernel density curve

ggplot(continuous, aes(x = hours.per.week)) +
    geom_density(alpha = .2, fill = "red") + theme_bw()
```




```
test <- function(){

  if(params$interactive == TRUE){

  print('This is triggered by the interactive param being set to TRUE')

} else {

  print('This is triggered by the interactive param being set to FALSE')
}

}

test()
```

```
## [1] "This is triggered by the interactive param being set to FALSE"
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.