STA130H1 – Winter 2018

Week 1 Practice Problems

Instructions

What should I bring to tutorial on January 12?

- R output (e.g., plots) for Question 2. You can either bring a hardcopy or bring your laptop with the output.
- Answer to Question 4(d), parts (i), (ii).

First steps to answering these questions.

- Open this R Notebook in RStudio.
- Type your answers below each question. Remember that R code chunks can be inserted directly into the notebook by choosing Insert R from the Insert menu (see Using R Markdown for Class Assignments). In addition this R Markdown cheatsheet, and reference are great resources as you get started with R Markdown.
- If you are NOT working on https://rstudio.chass.utoronto.ca/ then you will need to install the `tidyverse` and `mosaic` packages to complete the questions.

Practice Problems

Question 1

Exercise 3.1 in the textbook uses data that come with `R`. The dataset is in the `mosaic` package, which you must first load with the command `library(mosaic)`. The name of the dataframe is `Galton`.

a. Construct the plots that you are asked to construct in Exercise 3.1.
b. Name three additional plots that would be interesting to examine.

Question 2

Bring your output for this question to tutorial on Friday January 12 (either a hardcopy or on your laptop). For this question, we will use the data in Exercise 3.4 in the texbook. You can read more about the data and the variables here: https://rdrr.io/cran/mosaicData/man/Marriage.html.

a. Construct at least two plots that each show the distribution of one categorical variable.
b. Construct at least two plots that each show the distribution of one quantitative variable.
c. Construct a plot that shows the relationship between variables. What can you say about the relationship?
d. Can you consturct a plot using three variables? four variables? If you can, construct them!

Question 3

For this exercise, you will load data from an external source. You can read about the data here: http://sta220.utstat.utoronto.ca/data/the-skeleton-data/.

The data are in a plain text file with spaces between columns here: http://stats.onlinelearning.utoronto.ca/wp-content/uploaded/Data/SkeletonDatacomplete.txt. The following code will load the data into a tibble (the tidyverse version of a data frame).

a. Read the data into `R` using the following code.

```
library(tidyverse)
data_url <- "http://stats.onlinelearning.utoronto.ca/wp-content/uploaded/Data/SkeletonDatacomplete.txt"
skeleton_data <- read_table(data_url)
```

Inspect the data to make sure it is read in completely. You can compare by going directly to the `data_url`.

b. Construct at least four interesting graphs with the data, including: a graph of one categorical variable, a graph of one quantitative variable, a graph with at least two variables, a graph with at least three variables.

c. Describe what you learned about the data from your graphs.

Question 4

Recall from class that the histogram is a density estimator. Suppose that we have a sample of real observations (data) $X_1, X_2, \ldots, X_n$ and we wish to estimate the underlying density function.

(a) Given an *origin* x_0 and a *bin width* h, the *bins* of the histogram are left-closed, right-open intervals

$$[x_0 + mh, x_0 + (m+1)h)\,,$$

for some (positive or negative) integer m.

What is the length of a histogram bin?

(b) In this exercise you will create several histograms of math scores in `SAT_2010` data in the `mdsr` library (see page 39, 41 of MDSR) where you specify different lengths of histogram bins using `ggplot()`.

- Create a histogram without specifying the `binwidth` argument. What do you notice?
- Create histograms where `binwidth` has the values 10, 15, and 20.

Which histogram is the most accurate representation of the distribution of math scores?

(c) In this exercise you will recreate the histograms from (b), but will add several arguments to `geom_histogram()`: `aes(y=..density..)`; `alpha`; `fill`; and `colour` (a list of colours is here and see here for alpha, fill, and colour)) . The density argument changes the y-axis to relative frequency, and `aes(y=..count..)` specifies that frequency should be used on the y-axis. Here is starter code:

```
library(mdsr)
library(tidyverse)
SAT_2010 %>% ggplot(aes(x=math)) + geom_histogram(aes(y=..density..),binwidth = 10,fill="darkgrey",colou
```

Try different values of `alpha` and colours to create a histogram that's easy to interpret. Also, try the histogram with frequency and relative frequency on the y-axis. Which is easier to interpret?

Bring your solution for this question (4.(d) parts (i), (ii)) to tutorial on Friday January 12

(d) The naive estimator $\hat{f}$ of a density function f is given by choosing a small number $h > 0$ and setting

$$\hat{f}(x) = \frac{1}{2hn}\#\{X_i \in (x-h, x+h), i = 1, ..., n\}.$$

(i) Interpret $\hat{f}(x)$. Start by explaining what the numerator and denominator represent.

(ii) Prove that

$$\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{\epsilon}w\left(\frac{x-X_i}{h}\right),$$

where $w(x)$ is the rectangle weight function,

$$w(x) = \begin{cases} \frac{1}{2} & \text{if } |x| < 1 \\ 0 & \text{otherwise.} \end{cases}$$

(iii) The weight function $w(x)$ in part (ii) can be replaced with a *kernel function* $K(x) \geq 0$ which satisfies the condition:

$$\int_{-\infty}^{\infty} K(x)dx = 1.$$

The kernel estimator with kernal K is:

$$\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h}K\left(\frac{x-X_i}{h}\right).$$

h is often called the *smoothing parameter* or *bandwidth*. The kernal function $w(x)$ in (ii) is called the rectangular kernal function (why?).

`geom_density` adds the density estimate of the data to the plot. The kernel of the density can be specified using the kernel option in `geom_density`, and the `adjust` option (see page 41 of mdsr) can be used to set the value of the bandwidth.

In this exercise you will investigate the effect of the `adjust` parameter in `geom_density`, and the choice of kernal. The starter code below adds a kernal density estimate to the histogram.

```
library(tidyverse)
library(mdsr)

SAT_2010 %>% ggplot(aes(math))  + geom_histogram(aes(y=..density..),binwidth = 10,fill="gold1",colour="
```

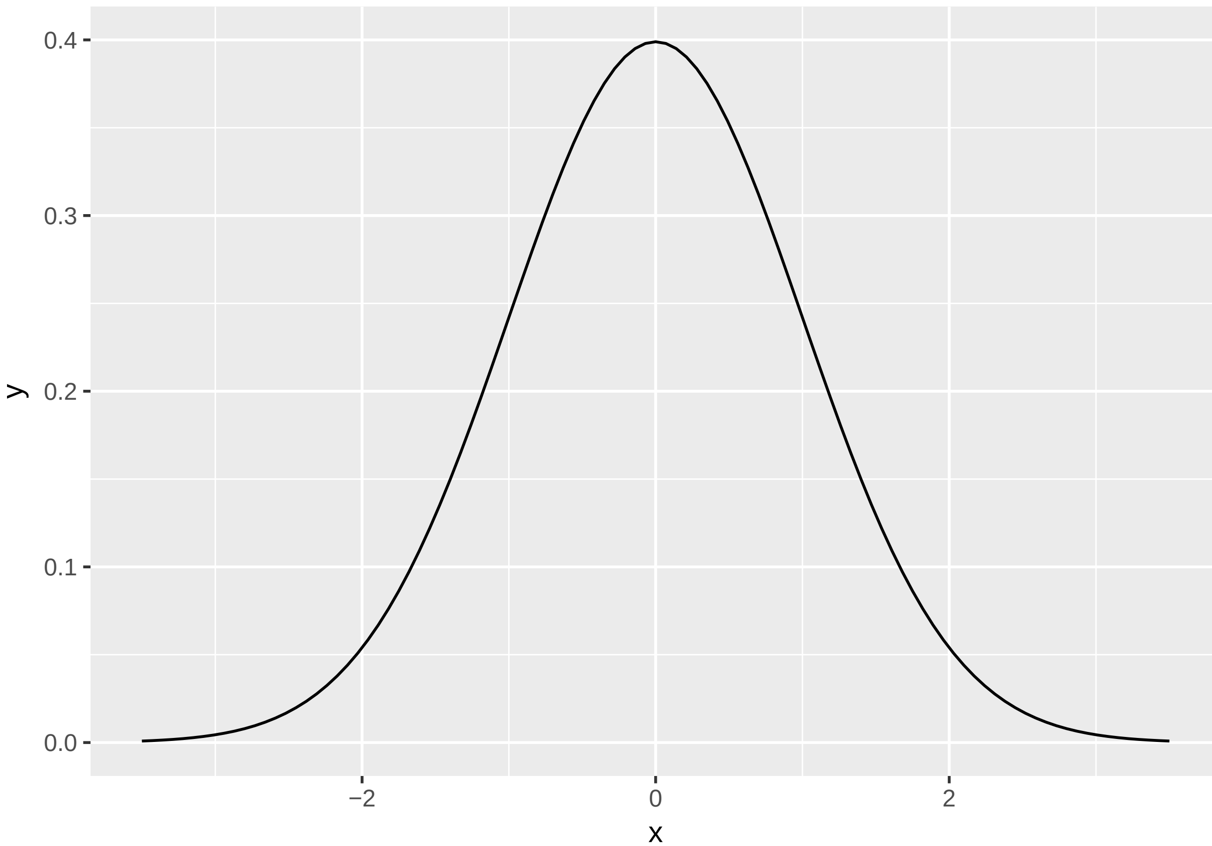
Change the value of `adjust` to 0.3, 0.5, and 0.8. What do you observe? Now, repeat what you just did, but this time change the value of `kernel`="gaussian". Which value of bandwidth and kernal gives the most accurate representation of the distribution of math scores?

NB: The Gaussian kernal is the famous bell curve (normal desnity curve) with mean 0 and standard deviation 1:

$$K(x) = \frac{1}{\sqrt{2\pi}}\exp\left(x^2/2\right), -\infty < x < \infty.$$

We can plot this function using `ggplot` using the built in density function `dnorm()` (we will come back to this function later in the course).

```
library(tidyverse)
dat <- data_frame(x = seq(-3.5, 3.5, by = 0.1))
dat %>% ggplot(aes(x)) + stat_function(fun = dnorm)
```



Extra (just for fun): Plot the rectangular kernal using `ggplot` (for `gplot` syntax see).

(iv) If you were required to choose **only one** of the histograms, with or without a kernel density estimate, to convey the distribution to people without a statistics background which plot would you choose? Which plot would you choose if the intended audience had a background in statistics? Explain your choice(s).