DATA 101 Exam 2

Shreehar Joshi

Due: Sunday, 11/29 at 11:59pm

Academic Honesty Statement (fill in your name)

I, Shreehar Joshi, hereby affirm that I have not communicated with or gained information in any way from my classmates or anyone other than the Professor during this exam, that I have not assisted anyone else with this exam, and that all work is my own.

Load packages and data

```
library(tidyverse)
library(tidymodels)
library(janitor)
library(lubridate)
library(NHANES)
```

```
# Load in the data here
nhanes <- NHANES %>%
  janitor::clean_names()
# DO NOT OVERWRITE THIS TABLE DURING THE EXAM
```

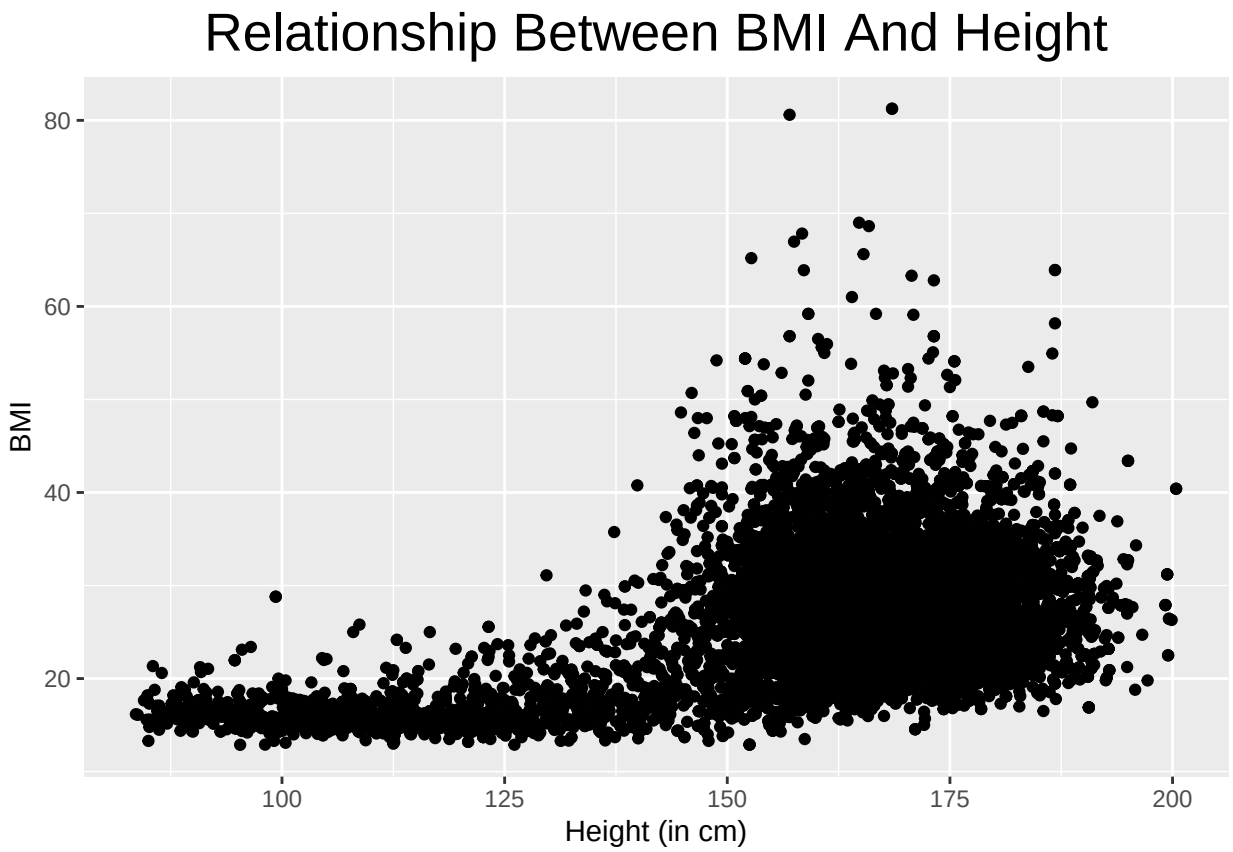
Questions

Question 1

At first, lets explore the relationship between BMI and Height.

```
nhanes %>%
  ggplot(aes(x = height, y = bmi)) +
  geom_point() +
  theme(plot.title = element_text(hjust = 0.5, size = 20)) +
  # Reference(https://www.xspdf.com/resolution/50423263.html)
  labs(
    title = "Relationship Between BMI And Height",
    x = "Height (in cm)",
    y = "BMI"
  )
```

```
## Warning: Removed 366 rows containing missing values (geom_point).
```

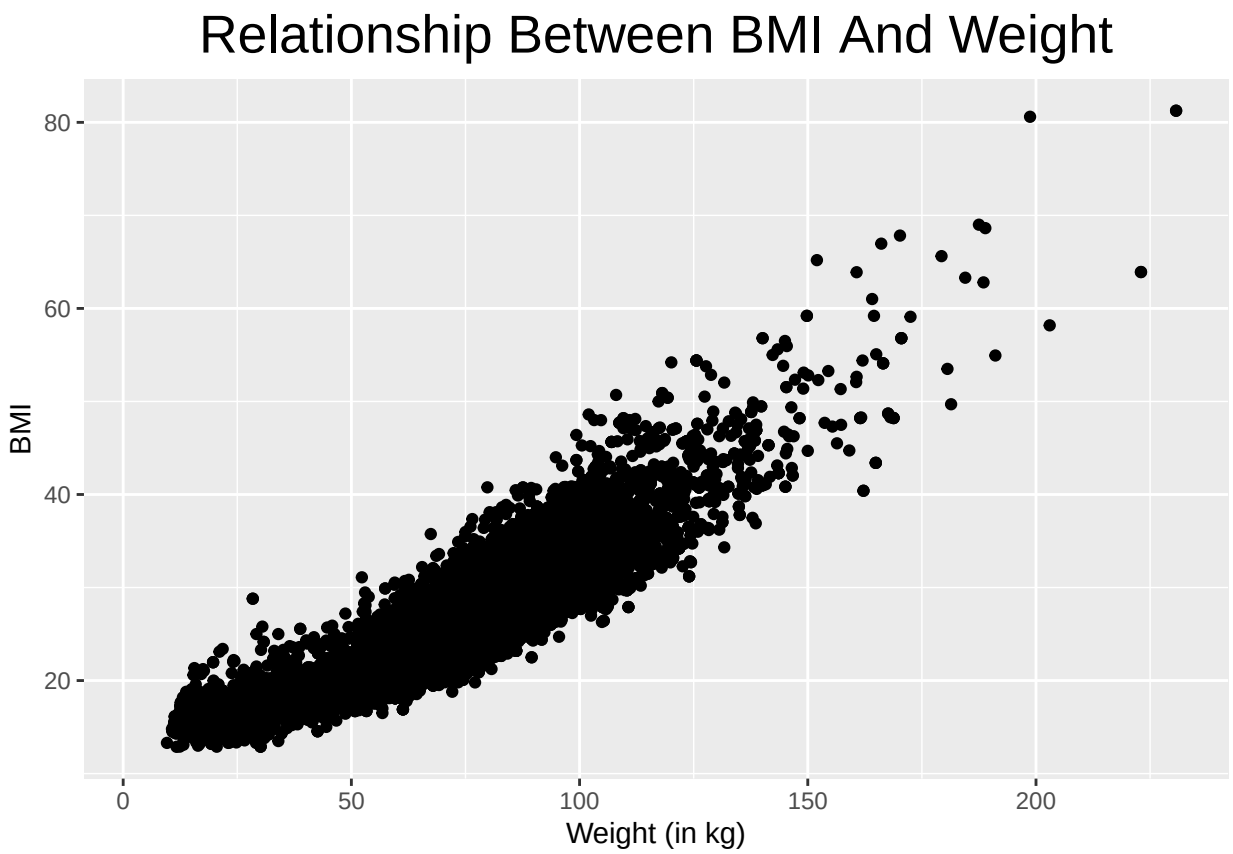


Overall, there seems to be a positive correlation between BMI and height. To be more specific, for people with height below 125 cm, the BMI appears to be roughly constant but as we go beyond, it becomes evident that BMI takes larger and varying values. While one would expect BMI to decrease with increase in height from its formula, this is not seen in the graph. The BMI might have been constant for initial height because as the height of a person increases, his/her weight also increases and this might have balanced out the BMI from decreasing (BMI is directly proportional to the weight). For people with height above 125 cm, their BMI is varying largely. This might be the case because even when people have the same height, their weight might vary widely. Some might be overweight and hence have higher BMI and some might be underweight and hence have lower BMI. This analysis can lead us to formulation of a hypothesis : "Short people have similar body weight for a given height that increases more or less proportionately with the height while as medium and tall people have largely varying body weight for any given height".

Now, lets explore the relationship between BMI and Weight.

```
nhanes %>%
  ggplot(aes(x = weight, y = bmi)) +
  geom_point() +
  theme(plot.title = element_text(hjust = 0.5, size = 20)) +
  # Reference(https://www.xspdf.com/resolution/50423263.html)
  labs(
    title = "Relationship Between BMI And Weight",
    x = "Weight (in kg)",
    y = "BMI"
  )
```

```
## Warning: Removed 366 rows containing missing values (geom_point).
```



There is a strong positive correlation between weight of a person and his/her BMI. BMI estimates the amount of body fat of a person and since a person who weighs heavier is more likely to have more amount of body fat, his/her BMI is also expected to be higher, which is exactly what is shown in the figure above. The relation above is also supported by the formula of BMI, where BMI = Weight / $Height^2$.

Question 2

Now, lets fit a linear model to predict the BMI from Weight.

```
bmi_fit <- lm(bmi ~ weight, data = nhanes)

tidy(bmi_fit)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)    9.09    0.0915       99.4       0
## 2 weight         0.241   0.00118     205.        0
```

The equation of the linear model for predicting BMI as a function of weight can be written as:
$\hat{BMI} = 9.09 + 0.241 * weight$

The coefficient of weight suggests that for every unit increase in the weight(increase of 1 kg), the BMI of a person is expected to increase by 0.241 on average if everything else is held constant.

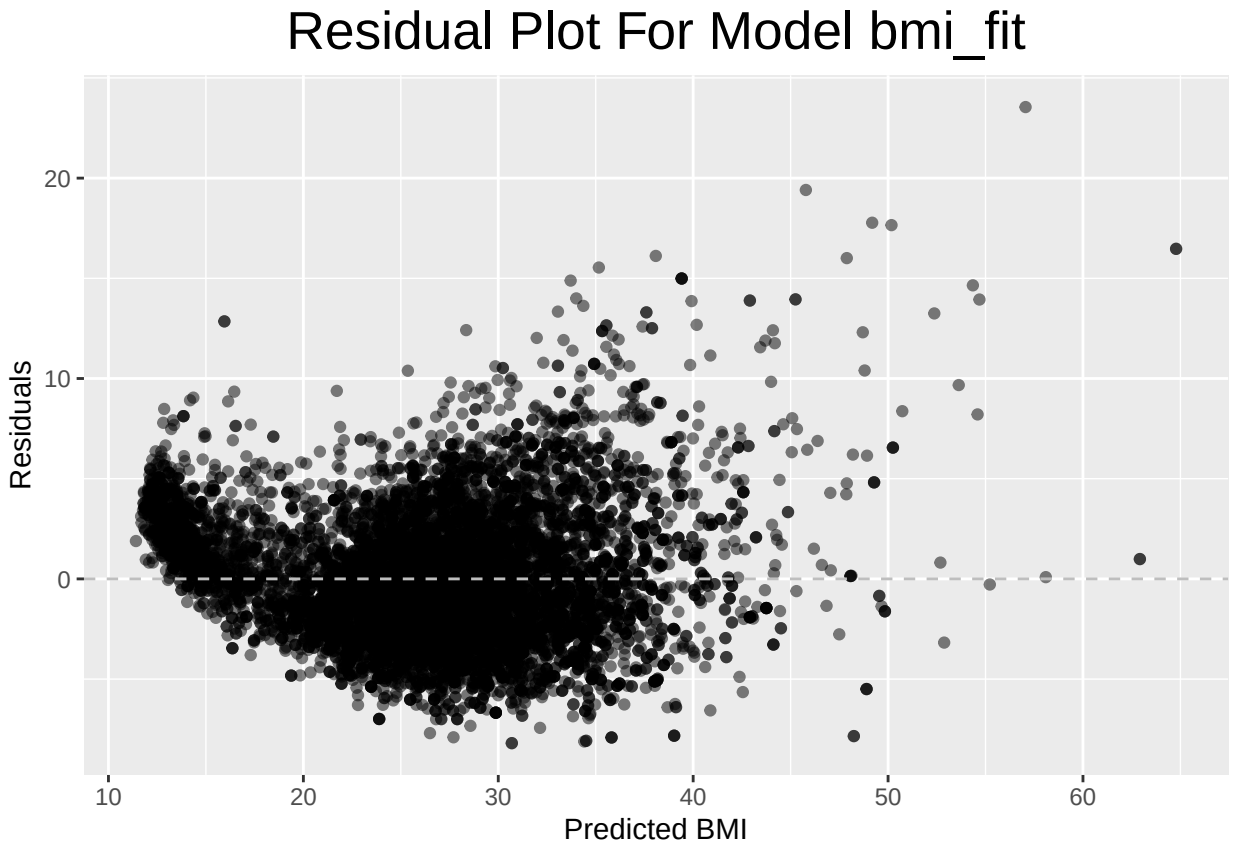
Question 3

The coefficient of determination, R^2, of the model is 0.8139379. It means that 81.39 percent of the variability in the value of BMI can be explained by the variability in the weight.

Question 4

Now, lets draw the residual plot for our model.

```
bmi_fit_aug <- augment(bmi_fit)
ggplot(bmi_fit, mapping = aes(x = .fitted, y = .resid)) +
  geom_point(alpha = 0.5) +
  theme(plot.title = element_text(hjust = 0.5, size = 20)) +
  # Reference(https://www.xspdf.com/resolution/50423263.html)
  geom_hline(yintercept = 0, color = "gray", lty = 2) +
  labs(
    title = "Residual Plot For Model bmi_fit",
    x = "Predicted BMI",
    y = "Residuals"
  )
```



The residuals are not randomly distributed around zero and show a specific pattern: they approach the zero from above(positive values) and then fan out as the predicted BMI increases. This implies that our linear model is not a good fit for our data neither at the lower nor at the higher end. Moreover, our model seems to be underestimating at the lower values(below 15) and it seems to be having more error while predicting the BMI as the values get higher.

Question 5

Now, lets fit a multiple linear regression model to predict systolic blood pressure as a function of weight, age and gender.

```
bp_fit <- lm(bp_sys_ave ~ weight + age + gender, data = nhanes)

tidy(bp_fit)
```

```
## # A tibble: 4 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)  92.3      0.580      159.   0.
## 2 weight        0.0937   0.00724     12.9  6.06e-38
## 3 age           0.414    0.00813     50.9  0.
## 4 gendermale    3.20     0.327        9.78 1.88e-22
```

The equation for the model above can be written as:
$\hat{bp\_sys\_ave} = 92.3 + 0.0937 * weight + 0.414 * age + 3.20 * gender_{male}$

The model equation for males can be written as:
$\hat{bp\_sys\_ave} = 92.3 + 0.0937 * weight + 0.414 * age + 3.20 * 1$
$\hat{bp\_sys\_ave} = 92.3 + 0.0937 * weight + 0.414 * age + 3.20$
$\hat{bp\_sys\_ave} = 95.5 + 0.0937 * weight + 0.414 * age$

The model equation for females can be written as:
$\hat{bp\_sys\_ave} = 92.3 + 0.0937 * weight + 0.414 * age + 3.20 * 0$
$\hat{bp\_sys\_ave} = 92.3 + 0.0937 * weight + 0.414 * age + 0$
$\hat{bp\_sys\_ave} = 92.3 + 0.0937 * weight + 0.414 * age$

Now, lets use the equation for male to predict the systolic blood pressure of a 60 year old man weighing 91 kg. His systolic blood pressure is predicted to be 128.87.

Question 6

Now, lets fit a logistic regression model to predict if a person has diabetes as a function of his/her age, gender and BMI.

```
diabetes_fit <- logistic_reg() %>%
  set_engine("glm") %>%
  fit(diabetes ~ age + gender + bmi, data = nhanes, family = "binomial")

tidy(diabetes_fit)
```

```
## # A tibble: 4 x 5
##   term        estimate std.error statistic   p.value
##   <chr>          <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)  -8.38     0.258      -32.5  5.16e-231
## 2 age           0.0582   0.00252     23.1  3.60e-118
## 3 gendermale    0.365    0.0833       4.38 1.21e-  5
## 4 bmi           0.0965   0.00560     17.2  1.25e- 66
```

The model equation can be written as:
$\log \frac{p}{1-p} = -8.38 + 0.0582 * age + 0.365 * gender_{male} + 0.0965 * bmi$

Question 7

Now, lets define our own inverse-logit function.

```
inv_logit <- function(x){
  round((1)/(1 + exp(-x)), 2)
}
```

The inverse logit function takes a value between $+\infty$ and $-\infty$ and maps it to a value between 0 and 1. This output can be compared to a threshold value to calculate the predicted outcome of an event.

Question 8

Now, lets predict the diabetes status of a 55-year old woman with a BMI of 24.

```
x <- -8.38 + 0.0582 * 55 + 0.365 * 0 + 0.0965 * 24
inv_logit(x)
```

```
## [1] 0.05
```

The probability that the woman is diabetic is 0.05 which is really low. In other words, she has just 5% chance of having a diabetes, according to our model.

Question 9

Now, lets find the probability of being diabetic for five different patients.

```
outputs = c(-2.20, 0.01, -0.35, 1.15, 0.83)
results <- map_dbl(outputs, inv_logit)
results
```

```
## [1] 0.10 0.50 0.41 0.76 0.70
```

The probability of the third patient being diabetic is 0.41. This means that he/she has 41 percent chance of having diabetes according to our model. He/she seems to be on the brink of having diabetes. The value of 0.41 is less than our threshold value of 0.5, hence we predict that the third patient doesn't have diabetes.

Question 10

An example of binary classification situation where one would want to choose a relatively high threshold is in the case of black hole detection. If we choose low threshold in this case, then we will have high false positive rate. This means that our model would predict large number of the non-black hole entities like stars in the galaxy as black holes, which is contrary to what an astronaut would want. To confirm that entity is a black hole, it requires days and months of analysis and observation and if our model starts wrongly predicting other objects as black holes, then the astronauts might have to waste large amount of time and resource just to have an unproductive outcome. Hence, an astronaut would strongly prefer his/her model to predict black holes only if there is a high probability that the entity is indeed a black hole. While the number of instances a black hole can go undetected might increase when choosing the high threshold, the cost of falsely detecting the black holes and wasting time and resource for further analysis outweigh the cost of not detecting the blackhole in this case.

Similarly, an example of binary classification situation where one would want to choose a relatively low threshold is breast cancer detection. If we choose high threshold in this case, then we will have high false negative rate. This means our model would predict even the cancer causing tissues as "normal" tissues, which is contrary to what medical personnels would want. The cancer, if detected early, can be cured and even the slightest possibility of having a cancer must be taken care of. If a person is predicted to have a breast

cancer, then he/she can undergo further analysis to confirm its presence and remove the cancer casusing tissues as soon as possible. While, our model might predict even the non-cancer causing tissues as one that causes cancer in low threshold, but the cost of letting a cancer go undetected far outweighs the cost of falsely detecting cancers.

Question 11

A site might not allow scraping of its data for several reasons:
1. The data gathered from web-scraping might disclose people's identity and potentially harm them.
2. The website might not have informed consent from its users to provide their data for other purpose.
3. To prevent replication of functionality of the site and hence loss of its revenue from the added unfair competitors.
4. To prevent denial of service(DOS) for the intended users of the site.

Question 12

The Amazon's experimental data hiring algorithm trained on a biased data which showed that male candidates were more preferable than their female counterparts because till 2018 amazon had hired more number of male candidates than the female counterparts for the role of software developer and other technical posts. As a result, it created a gender bias and led to them choosing unqualified candidates.

Similarly, the robots from the cartoon also must have trained on a biased data that had majority of battles that were won using spears and rocks as before the invention of gunpowder, all the battles had spears and other pre-modern tools used in them. This resulted in them being biased on choosing spears and rocks, which were incompetent weapons for their battle similar to choosing males some of whom were unqualified candidates in the case of amazon.

Question 13

Now lets read the "people.csv" dataset, manipulate it and display its data.

```
people <- read_csv("data/people.csv", na = c("", "N/A", "Unknown")) %>%
  janitor::clean_names() %>%
  mutate(
    date_of_birth = mdy(date_of_birth),
    height = factor(height, levels = c("short", "medium", "tall", "very tall"))
  )
```

```
##
## -- Column specification ------
## cols(
##   Name = col_character(),
##   `Date of Birth` = col_character(),
##   Height = col_character(),
##   `Number of Siblings` = col_double()
## )
```

```
people %>%
  arrange(height)
```

```
## # A tibble: 6 x 4
##   name    date_of_birth height    number_of_siblings
##   <chr>   <date>        <fct>                  <dbl>
## 1 Alice   1999-09-09    short                      3
```

```
## 2 Carlos  2002-10-01    medium                    1
## 3 Denise  1983-01-28    medium                   NA
## 4 Elaine  2006-08-13    tall                      2
## 5 Bob     1978-02-04    very tall                NA
## 6 Juanita 1965-11-10    <NA>                      0
```

Instead of defining our missing values or "na"s explicitly, we could have specified the column types for each of the columns by passing an additional argument "col_types" in the read_csv function to ensure that they had appropriate data types. Then we wouldn't have to change the data type by using mutate functions separately. While this method can coerce some of the missing or null values in the dataset into "NA", it has the downside of not being able to do it for all the required values.