STA2453 Lab 2

Yihan Duan

15/10/2021

Exercise 1

Scenario 1

No, I don't think this is a ideal problem for linear regression. Because we are more interested in "whether" (the probability) that a customer makes a purchase, this is more of a classification problem. Ideally the output of this model should be a probability score between 0 and 1 (indicating the probability that the customer will make a purchase), instead of an estimation for quantity. As all regression models predicts a quantity and the estimation does not necessarily lays between 0 and 1, this scenario is not ideal for linear regression.

However, if we rephrase the question to "predict a customer's total spending", this problem becomes more suitable for linear regression and can solved using the same data. We can perform a linear regression regarding times of visits and total length of stay as variables and the output as the total spending (0 if nothing is purchased).

Scenario 2

The dependent variable is 'child_inc30' and the potential independent variables are 'parents_inc50', 'child_gender' and 'child_edu'. We do not include the education level of the parents ('father_edu' and 'mother_edu') as they seems to have less affect on the child's salary. Assume the independent variables we chose are x_1, x_2, x_3 with their coefficients being $\beta_1, \beta_2, \beta_3$ respectively, we make the assumption that $y = \beta_0 + \sum_{i=1}^{3} \beta_i x_i + \epsilon$ where $\epsilon \sim Normal(0, \sigma)$. In other words, controlling for the child's gender and education level, his/her income should have a linear relation to the parents' income at 50.

Beyond the simple linear model (with only the parents' income as independent variable), we could include the other variables like suggested above. Moreover, we could include the interaction terms between 'parents_inc50' and 'child_gender', or between 'parents_inc50' and 'child_edu'. These would help understand the main relationship as including them can help understand and control the affects other variables have on the child's income.

Scenario 3

No, this is not a good candidate for a linear regression model.

There are too few variables collected other than salary. Many other factors (for example age, marital status, health, stress…) may have a significant influence on the employee's happiness rating, but those factors are not included in the data. Therefore, I suspect that we can draw a reliable conclusion from a simple linear model. Not controlling for the other factors may give us misleading results.

The happiness score presented in the data is highly biased as they are self-evaluated or self-measured scores. Without a reliable quantitative dependent variable, a linear model might not be suitable.

I also suspect that the happiness one receives from salary has diminishing returns, meaning the more money we get, the less we benefits from the same increment in salary. As the relationship is far from linear, I believe a linear model can not be used in this case.

Exercise 2

Take a look at the data.

```r
head(ern_full)
```

```
## # A tibble: 6 x 10
##   rank_this_week rank_last_week player_name      events money      ytd_victories
##   <chr>          <chr>          <chr>             <dbl> <chr>      <lgl>
## 1 1              T6             Jon Rahm              1 $1,710,000 NA
## 2 2              1              Dustin Johnson        1 $1,026,000 NA
## 3 T3             T29            Hideki Matsuyama      1 $551,000   NA
## 4 T3             <NA>           Joaquin Niemann       1 $551,000   NA
## 5 5              <NA>           Tony Finau            1 $384,750   NA
## 6 T6             T13            Jason Kokrak          1 $337,250   NA
## # ... with 4 more variables: tournamet_id <chr>, tournament_name <chr>,
## #   year <dbl>, type <chr>
```

```r
head(brd_full)
```

```
## # A tibble: 6 x 9
##   rank_this_week rank_last_week player_name       rounds total tournamet_id
##   <chr>          <chr>          <chr>              <dbl> <dbl> <chr>
## 1 T1             T6             Xander Schauffele      4    21 t060
## 2 T1             T12            Collin Morikawa        4    21 t060
## 3 3              T2             Dustin Johnson         4    20 t060
## 4 T4             T6             Justin Thomas          4    19 t060
## 5 T4             T35            Sungjae Im             4    19 t060
## 6 T6             T49            Harris English         4    18 t060
## # ... with 3 more variables: tournament_name <chr>, year <dbl>, type <chr>
```

```r
head(drdis_full)
```

```
## # A tibble: 6 x 11
##   rank_this_week rank_last_week player_name    rounds   avg total_distance
##   <chr>          <chr>          <chr>           <dbl> <dbl>          <dbl>
## 1 1              2              Cameron Champ       4  319.           2554
## 2 2              T4             Dustin Johnson      4  318.           2541
## 3 3              T4             Tony Finau          4  314.           2508
## 4 4              T53            Tyrrell Hatton      4  312.           2499
## 5 5              3              Rory McIlroy        4  312.           2494
## 6 6              T40            Lanto Griffin       4  310.           2476
## # ... with 5 more variables: total_drives <dbl>, tournamet_id <chr>,
## #   tournament_name <chr>, year <dbl>, type <chr>
```

Check if player name + tournament id is the unique identifier for all 3 datasets.

```r
length(unique(paste(ern_full$player_name, ern_full$tournamet_id))) == nrow(ern_full)
```

```
## [1] TRUE
```

```r
length(unique(paste(brd_full$player_name, brd_full$tournamet_id))) == nrow(brd_full)
```

```
## [1] TRUE
```

```r
length(unique(paste(drdis_full$player_name, drdis_full$tournamet_id))) == nrow(drdis_full)
```

```
## [1] TRUE
```

Merge 3 useful columns of the 3 data sets.

```r
ern = ern_full[c('player_name', 'tournamet_id', 'events', 'money')]
brd = brd_full[c('player_name', 'tournamet_id', 'rounds', 'total')]
drdis = drdis_full[c('player_name', 'tournamet_id', 'total_distance', 'total_drives')]

# merge
merged_df = ern %>%
  merge(brd, by = c('player_name', 'tournamet_id'), all=TRUE) %>%
  merge(drdis, by = c('player_name', 'tournamet_id'), all=TRUE)

tail(merged_df)
```

```
##         player_name tournamet_id events   money rounds total total_distance
## 2495    Zack Sucher         t483      1 $20,119      4    19           2290
## 2496    Zack Sucher         t490      1 $59,732      4    19           2674
## 2497    Zack Sucher         t524      1 $15,600      4    13           2472
## 2498 Zander Lombard         t473     NA    <NA>      4    21           2688
## 2499 Zander Lombard         t489     NA    <NA>      4    14             NA
## 2500    Zecheng Dou         t489     NA    <NA>      4    12             NA
##      total_drives
## 2495            8
## 2496            8
## 2497            8
## 2498            8
## 2499           NA
## 2500           NA
```

There are obviously some NA's in the merged dataset. As we will be using the average, the inclusion of NA's might lead to inconsistent values, so we remove all the rows that contains NA's. An example of such is if a player is missing total driving distance field for multiple events, then it would be wrong to use the sum of distance as a variable for estimating total earnings.

```r
merged_df <- na.omit(merged_df)
```

Now transform player-week table to player-year table.

```r
merged_df <- merged_df %>%
  mutate(money = as.numeric(gsub('[$,]', '', money)))

df <- aggregate(cbind(events, money, rounds, total, total_distance, total_drives) ~ player_name, merged_
```

Now add average statistics the new player-year data frame

```r
# change column names
colnames(df) <- c("player_name", "num_weeks", "total_earnings",
                  "num_rounds", "total_birdies", "total_driving_distance",
                  "num_drives")
# Add new columns
df$avg_birdies = df$total_birdies / df$num_weeks
df$avg_driving_distance = df$total_driving_distance / df$num_drives
df$avg_earnings = df$total_earnings / df$num_weeks
```

Checkout the new df.

```r
head(df)
```

```
##      player_name num_weeks total_earnings num_rounds total_birdies
## 1 Aaron Baddeley         7         286503         28           121
```

```
## 2     Aaron Wise         5         246597         20            81
## 3  Abraham Ancer        13        2480071         52           201
## 4    Adam Hadwin        13        1617074         52           195
## 5      Adam Long        15        2063092         60           244
## 6    Adam Schenk        12         495476         48           200
##   total_driving_distance num_drives avg_birdies avg_driving_distance
## 1                  16007         56    17.28571             285.8393
## 2                  12051         40    16.20000             301.2750
## 3                  30912        104    15.46154             297.2308
## 4                  30572        104    15.00000             293.9615
## 5                  35388        120    16.26667             294.9000
## 6                  29083         96    16.66667             302.9479
##   avg_earnings
## 1     40929.00
## 2     49319.40
## 3    190774.69
## 4    124390.31
## 5    137539.47
## 6     41289.67
```

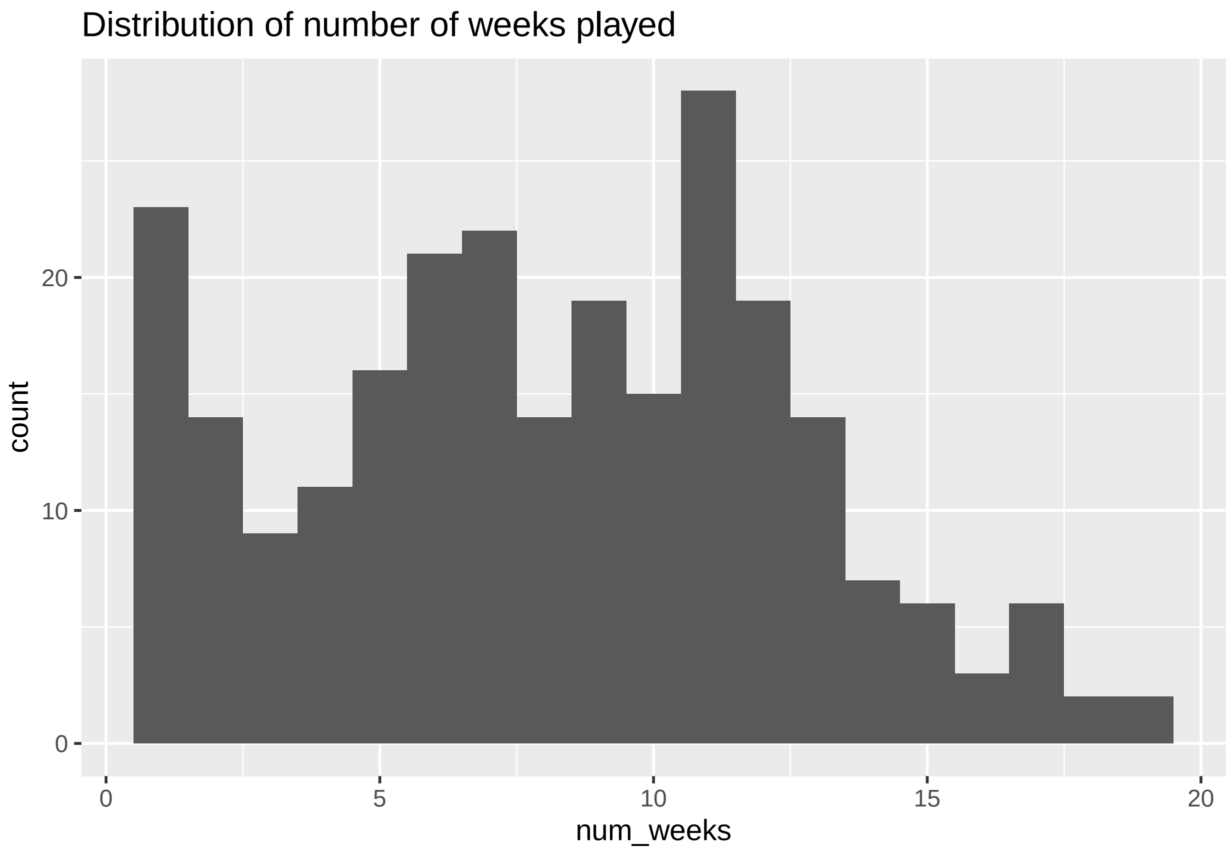
Exercise 3

How many players are in the data?

```r
length(unique(df$player_name))
```

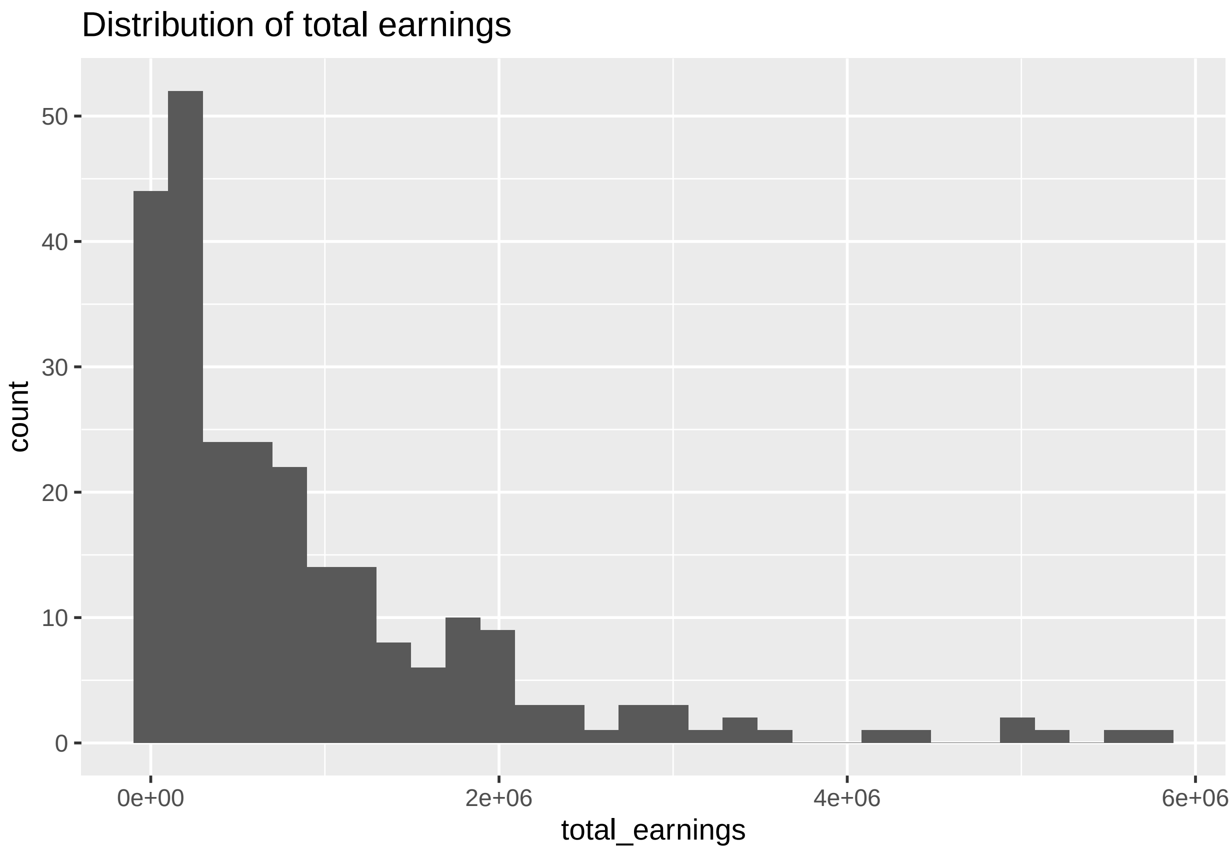
```
## [1] 251
```

```r
df %>%
  ggplot(aes(x=num_weeks)) +
  geom_histogram(binwidth = 1) +
  ggtitle("Distribution of number of weeks played")
```



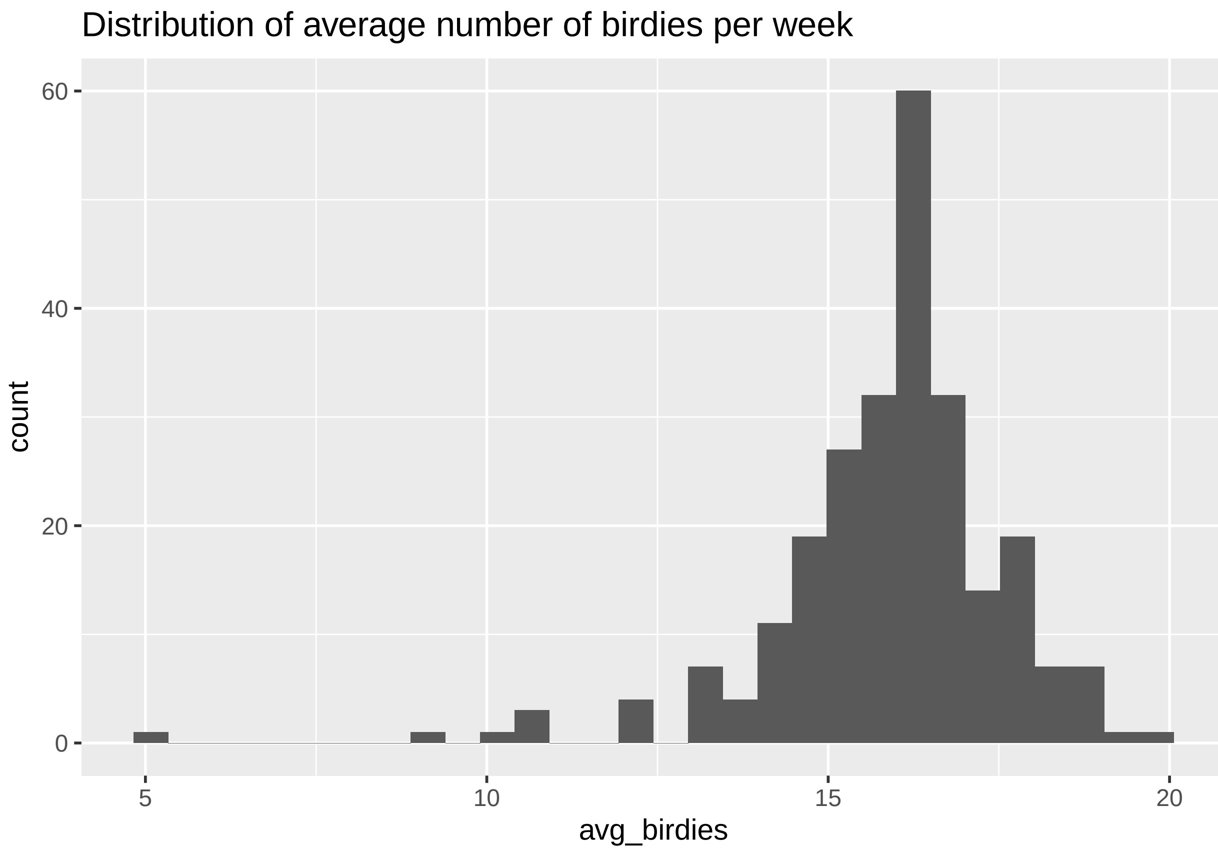

```r
df %>%
  ggplot(aes(x=total_earnings)) +
  geom_histogram() +
  ggtitle("Distribution of total earnings")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



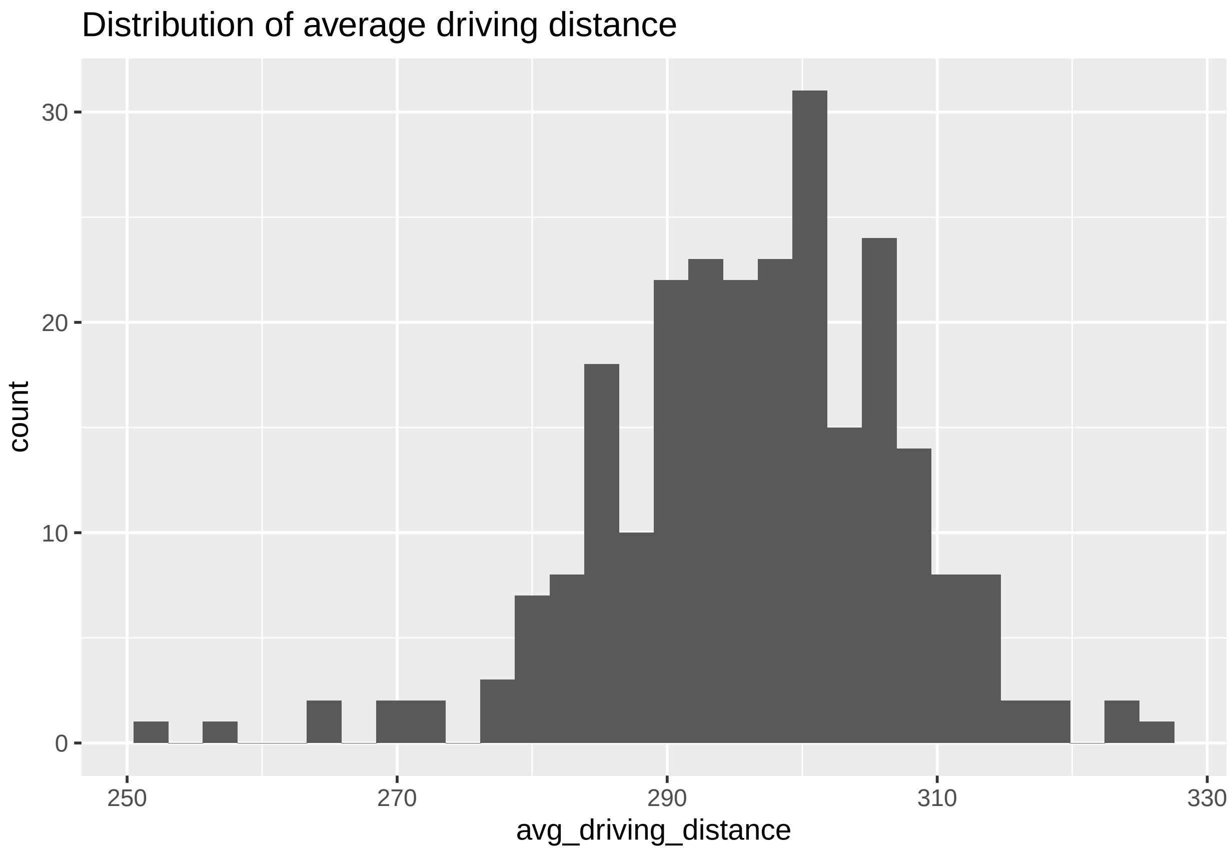

```
df %>%
  ggplot(aes(x=avg_birdies)) +
  geom_histogram() +
  ggtitle("Distribution of average number of birdies per week")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



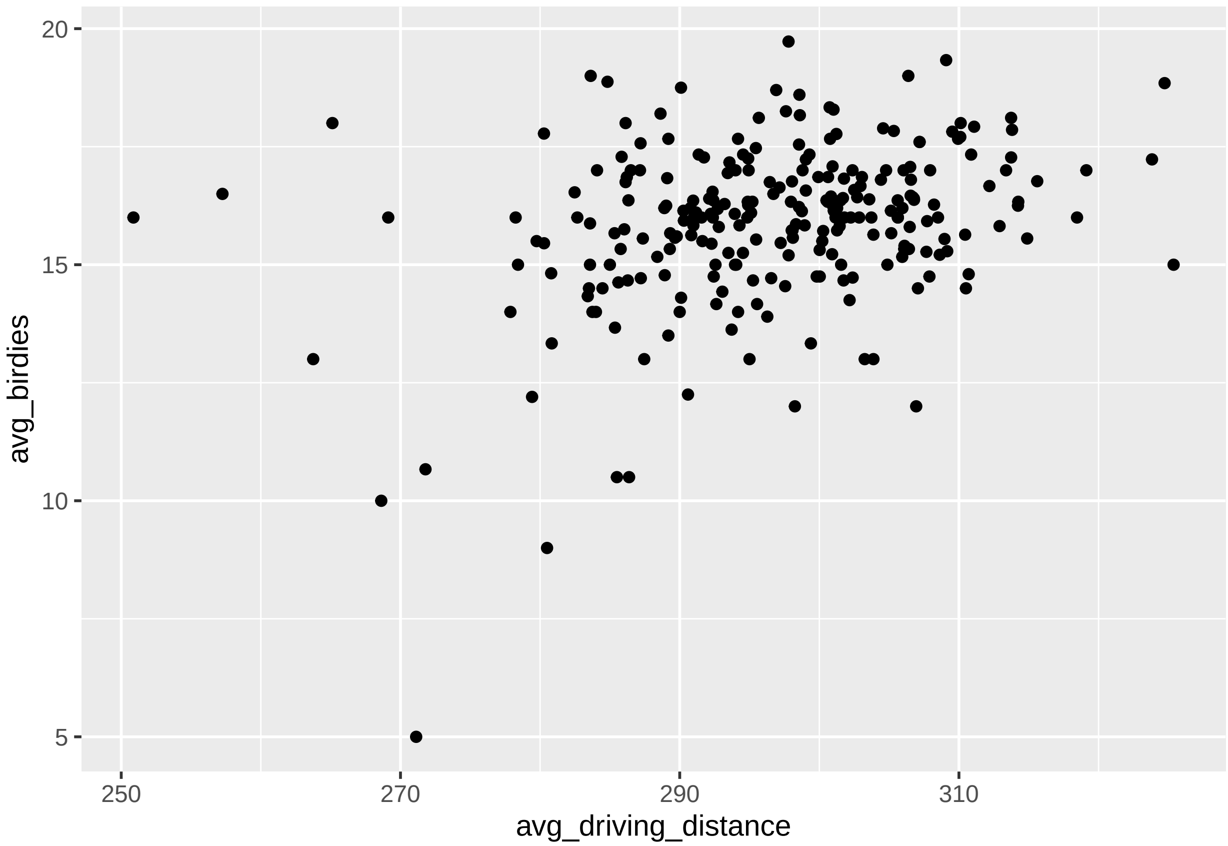

```
df %>%
  ggplot(aes(x=avg_driving_distance)) +
  geom_histogram() +
  ggtitle("Distribution of average driving distance")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



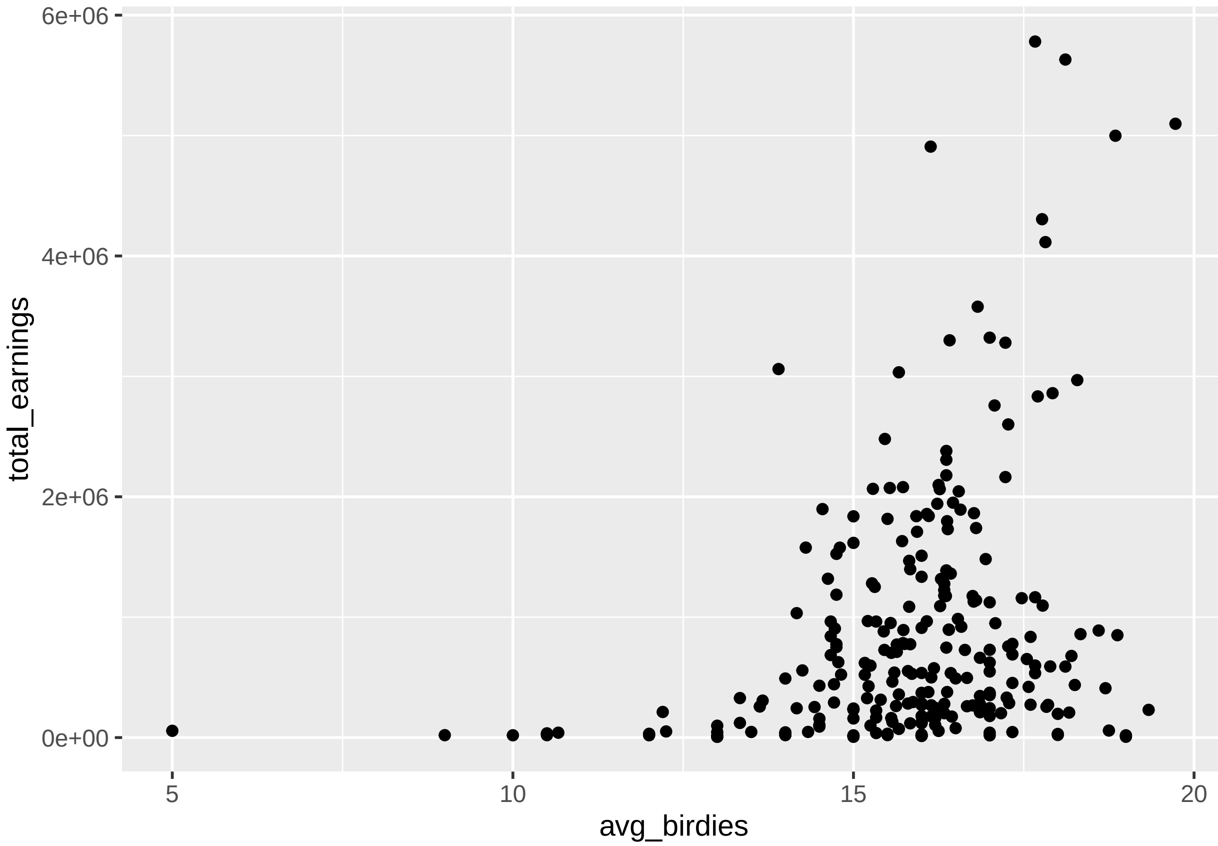
Let's plot some associations.

```
df %>%
  ggplot(aes(x=avg_driving_distance, y=avg_birdies)) +
  geom_point()
```

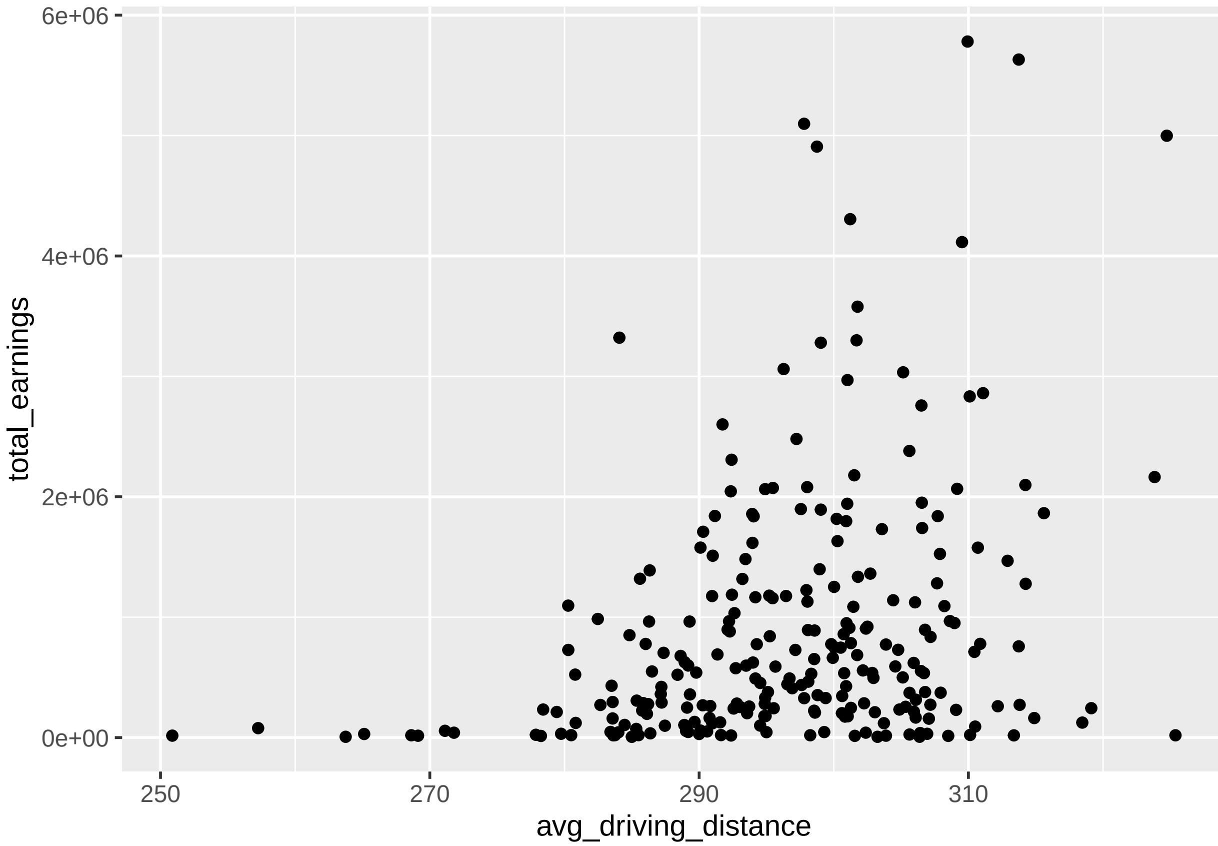


There seems to be a weak association between average number of birdies and average driving distance. Namely any player that is able to drive further seems to have more birdies per week on average.

```
df %>%
  ggplot(aes(x=avg_birdies, y=total_earnings)) +
  geom_point()
```




```
df %>%
  ggplot(aes(x=avg_driving_distance, y=total_earnings)) +
  geom_point()
```



There is a weak positive association between average birdies/average driving distance and total earnings too.

Model 1 total earnings ~ average birdies

```r
summary(lm(total_earnings ~ avg_birdies, df))
```

```
##
## Call:
## lm(formula = total_earnings ~ avg_birdies, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1438206  -679092  -305939   335397  4575199
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1965956     583592  -3.369 0.000875 ***
## avg_birdies   179523      36528   4.915 1.61e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1001000 on 249 degrees of freedom
## Multiple R-squared:  0.08843,    Adjusted R-squared:  0.08477
## F-statistic: 24.15 on 1 and 249 DF,  p-value: 1.613e-06
```

The intercept is the (hypothetical) expected total earnings for a player with 0 average birdies per week. In other words, a player is expected to earn -1965956 dollars, which is not possible.

The coefficient avg_birdies is the increase in total earnings when the average number of birdies per week is increased by 1. In this case, we expect a player's yearly earnings to increase by 179523 dollars yearly if he can make 1 more birdie per week.

Model 2 total earnings ~ average driving distance

```r
summary(lm(total_earnings ~ avg_driving_distance, df))
```

```
##
## Call:
## lm(formula = total_earnings ~ avg_driving_distance, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1700900  -589990  -267232   308359  4507860
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -7700918    1704000  -4.519 9.59e-06 ***
## avg_driving_distance    28954       5742   5.042 8.86e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 998700 on 249 degrees of freedom
## Multiple R-squared:  0.09265,    Adjusted R-squared:  0.08901
## F-statistic: 25.43 on 1 and 249 DF,  p-value: 8.86e-07
```

The intercept is the (hypothetical) expected total earnings for a player with average driving distance of 0. In other words, a player is expected to earn -7700918 dollars if his average driving distance is 0, which is not possible.

The coefficient avg_driving_distance is the increase in total earnings when the average driving distance is increased by 1. In this case, we expect a player's yearly earnings to increase by 28954 dollars yearly if his average driving distance increases by 1 yard.

Model 3 total earnings ~ average birdies + average driving distance

```r
summary(lm(total_earnings ~ avg_birdies + avg_driving_distance, df))
```

```
##
## Call:
## lm(formula = total_earnings ~ avg_birdies + avg_driving_distance,
##     data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1515209  -624082  -264638   337221  4359084
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -7820336    1665414  -4.696  4.4e-06 ***
## avg_birdies            134395      37599   3.574 0.000422 ***
## avg_driving_distance    22158       5924   3.740 0.000228 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 975900 on 248 degrees of freedom
## Multiple R-squared:  0.1371, Adjusted R-squared:  0.1301
## F-statistic:  19.7 on 2 and 248 DF,  p-value: 1.145e-08
```

The intercept is the (hypothetical) expected total earnings for a player with average weekly birdies number of 0 and average driving distance of 0. In other words, a player is expected to earn -7820336 dollars if both of his/hers average number of birdies and driving distance are 0.

The coefficient avg_birdies is, controlling for driving distance, the increase in total earnings when the average number of weekly birdies is increased by 1. In this case, we expect a player's yearly earnings to increase by 134395 dollars if his average number of birdies increases by 1, all else stays the same.

The coefficient avg_driving_distance is, controlling for average number of birdies, the increase in total earnings when the average driving distance is increased by 1 yard. In this case, we expect a player's yearly earnings to increase by 22158 dollars yearly if his average driving distance increases by 1 yard, all else stays the same.

Model 4 total earnings ~ average birdies + average driving distance

```r
summary(lm(total_earnings ~ log2(avg_birdies) + log2(avg_driving_distance), df))
```

```
##
## Call:
## lm(formula = total_earnings ~ log2(avg_birdies) + log2(avg_driving_distance),
##     data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1412843  -625570  -277357   344482  4418090
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -41444298    9590167  -4.322 2.24e-05 ***
## log2(avg_birdies)           1082637     350112   3.092 0.002214 **
## log2(avg_driving_distance)  4630556    1212498   3.819 0.000169 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 982200 on 248 degrees of freedom
## Multiple R-squared:  0.1259, Adjusted R-squared:  0.1189
## F-statistic: 17.87 on 2 and 248 DF,  p-value: 5.64e-08
```

The intercept is the (hypothetical) expected total earnings for a player with a 1 weekly birdies and 1 yard of average driving distance. In other words, a player is expected to earn -41444298 dollars if both of his/hers average number of birdies and driving distance are 1. Note that the intercept is not the expected value at 0 because $log(x) = 0$ when $x = 1$.

The coefficient avg_birdies is, controlling for driving distance, the increase in total earnings when the average number of weekly birdies doubles. (Notice that we are using log function with base 2). In this case, we expect a player's yearly earnings to increase by 1082637 dollars if his average number of birdies doubles, all else stays the same.

The coefficient avg_driving_distance is, controlling for average number of birdies, the increase in total earnings when the average driving distance doubles. In this case, we expect a player's yearly earnings to increase by 4630556 dollars yearly if his average driving distance doubles, all else stays the same.

Model 5 total earnings ~ average birdies + average driving distance + num_weeks

```r
summary(lm(total_earnings ~ avg_birdies + avg_driving_distance + num_weeks, df))
```

```
##
## Call:
## lm(formula = total_earnings ~ avg_birdies + avg_driving_distance +
##     num_weeks, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1473906  -465180  -169345   226387  4290270
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -4633670    1386530  -3.342 0.000961 ***
## avg_birdies             64652      31263   2.068 0.039682 *
## avg_driving_distance    11448       4921   2.326 0.020810 *
## num_weeks              134668      11978  11.243  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 795300 on 247 degrees of freedom
## Multiple R-squared:  0.4292, Adjusted R-squared:  0.4223
## F-statistic: 61.91 on 3 and 247 DF,  p-value: < 2.2e-16
```

The intercept is the (hypothetical) expected total earnings for a player with average weekly birdies number of 0, average driving distance of 0 and played 0 weeks (events). In other words, a player is expected to earn -4633670 dollars if he is really bad at the game as also doesn't play, which is not possible.

The coefficient avg_birdies is, controlling for other variables, the increase in total earnings when the average number of weekly birdies is increased by 1. In this case, we expect a player's yearly earnings to increase by 64652 dollars if his average number of birdies increases by 1, all else stays the same.

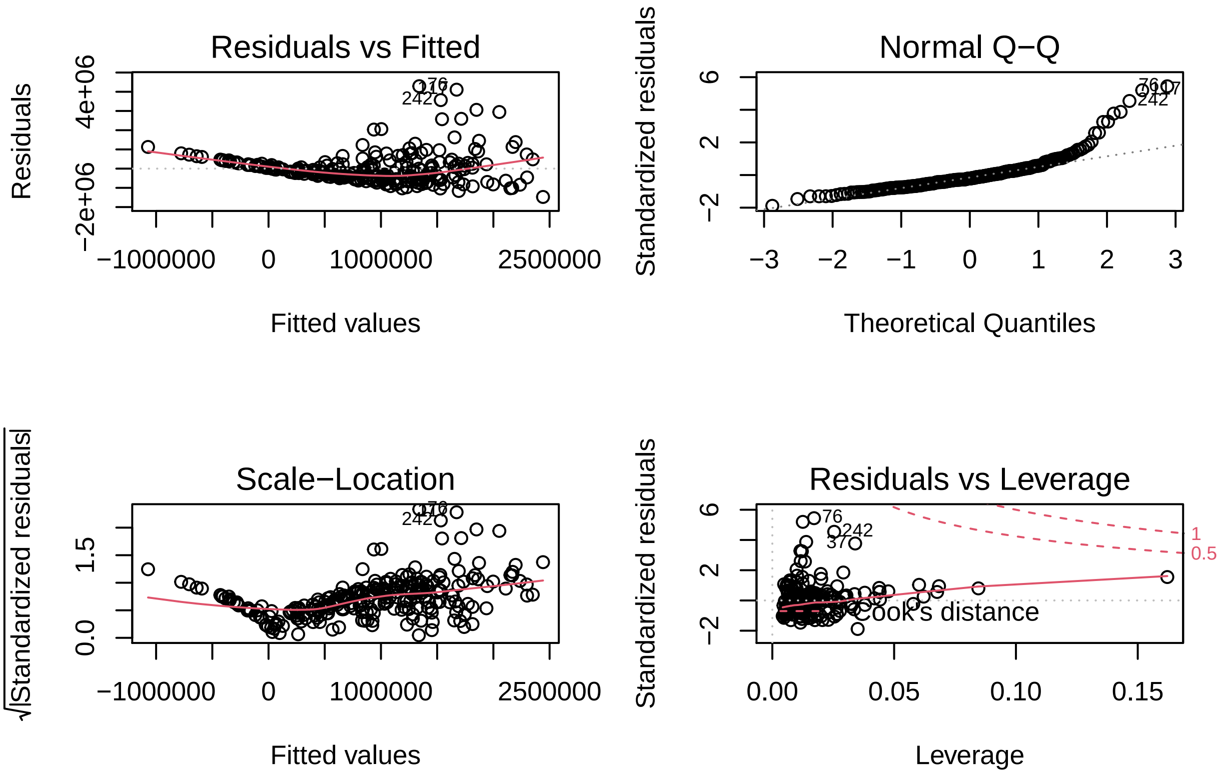
The coefficient avg_driving_distance is, controlling for all other variables, the increase in total earnings when the average driving distance is increased by 1 yard. In this case, we expect a player's yearly earnings to increase by 11448 dollars yearly if his average driving distance increases by 1 yard, all else stays the same.

The coefficient num_weeks is, controlling for other variables, the increase in total earnings when the player plays for 1 more week (event). In this case, we expect a player's yearly earnings to increase by 134668 dollars if he plays 1 more week a year, all else stays the same.

model fit We can see that the adjusted R-squared is 0.4223, meaning that the model is able to explain 42.33% of the variance, which is not a very good fit.

model assumptions Plot the fitted plots.

```r
par(mfrow = c(2, 2))
plot(lm(total_earnings ~ avg_birdies + avg_driving_distance + num_weeks, df))
```



1. Linearity

Yes, from the residuals vs fitted plot, we can see the estimated curve is close to horizontal line at $y = 0$.

2. Normality of residuals

No, from the Q-Q plot, we can tell the distribution of residuals is right-skewed.

3. Homogeneity of residuals variance

No, from the scale-location plot, the variablity increases with the fitted value.

4. Independence of residuals

No, clear pattern in the residuals vs fitted plot.