STAT 847: Analysis Assignment 1

Andrew Girgis
21108082

Importing Libraries

```
library(readr)
library(formatR)
library(plyr)
library(dplyr)
library(ggplot2)
library(tidyverse)
```

Importing data

```
asba_data = read.csv('HRN assiniboia scraped data.csv')
hast_data = read.csv('HRN hastings scraped data.csv')
wdbn_data = read.csv('HRN woodbine scraped data.csv')
```

```
colnames(wdbn_data)
```

```
##  [1] "meet_location"      "meet_wday"          "meet_mday"
##  [4] "meet_year"          "racecount"          "race_number"
##  [7] "horse_number"       "horse_name"         "horse_sire"
## [10] "horse_trainer"      "horse_jockey"       "horse_odds"
## [13] "horse_odds_decimal" "horse_place"        "purse"
## [16] "time_frac1"         "time_frac2"         "time_frac3"
## [19] "time_frac4"         "time_frac5"         "time_final"
## [22] "track_length"       "track_type"         "race_class"
## [25] "dist_frac1"         "dist_frac2"         "dist_frac3"
## [28] "dist_frac4"         "dist_frac5"
```

The columns that pertain to the horse are: horse_number, horse_name,horse_sire, horse_trainer, horse_jockey, horse_odds, horse_odds_decimal,and horse_place.

Therefore the columns that dont pertain to the horse are:meet_location, meet_wday, meet_mday, meet_year, racecount, race_number, purse, time_frac1, time_frac2, time_frac3, time_frac4, time_frac5, time_final, track_length, track_type, race_class, dist_frac1, dist_frac2, dist_frac3, dist_frac4, dist_frac5

Question 1

```
#Transposing our data so that each row represents a race grouping by the race count
race_stats = ddply(wdbn_data, .(racecount), summarize,
                   meet_location = meet_location[1],
                   meet_wday = meet_wday[1],
                   meet_mday = meet_mday[1],
                   meet_year = meet_year[1],
                   race_number = race_number[1],
                   purse = purse[1],
                   time_frac1 = time_frac1[1],
                   time_frac2 = time_frac2[1],
                   time_frac3 = time_frac3[1],
                   time_frac4 = time_frac4[1] ,
                   time_frac5 = time_frac5[1],
                   time_final = time_final[1],
                   track_length = track_length[1] ,
                   track_type = track_type[1],
                   race_class = race_class[1],
                   dist_frac1 = dist_frac1[1],
                   dist_frac2 = dist_frac2[1],
                   dist_frac3 = dist_frac3[1],
                   dist_frac4 = dist_frac4[1],
                   dist_frac5 = dist_frac5[1]
                   )

#prints the first 3 rows of the dataframe
print(head(race_stats, 3))
```

```
##   racecount meet_location meet_wday meet_mday meet_year race_number  purse
## 1         1      Woodbine    Sunday August 21      2022           1  64300
## 2         2      Woodbine    Sunday August 21      2022           2 123200
## 3         3      Woodbine    Sunday August 21      2022           3 125000
##   time_frac1 time_frac2 time_frac3 time_frac4 time_frac5 time_final
## 1      23.40      47.04         NA         NA         NA      59.48
## 2      23.34      46.99      71.82      96.58         NA     103.13
## 3      23.05      46.25      69.88         NA         NA      76.14
##   track_length        track_type                        race_class dist_frac1
## 1           5F        Inner turf $40,000 Maiden Optional Claiming        1/4
## 2      1 1/16M              Turf            Maiden Special Weight        1/4
## 3       6 1/2F All Weather Track               Sweet Briar Too S.        1/4
##   dist_frac2 dist_frac3 dist_frac4 dist_frac5
## 1        1/2
## 2        1/2        3/4       MILE
## 3        1/2        3/4
```

Question 2

```
# using the by() function to calculate the average time it takes for the
# winning horse to complete a race of each available length
time_by_length = by(race_stats$time_final, race_stats$track_length, mean)

# creating a variable track_length with all available track lengths
track_length <- c("1 1/16M", "1 1/2M", "1 1/4M", "1 1/8M", "1 3/4M", "1 3/8M", "1M",
    "1M 70Y", "4 1/2F", "5 1/2F", "5F", "6 1/2F", "6F", "7 1/2F", "7F")

# creating a variable average_time that contains all the averages tiems per
# track length
average_time <- c(86.20903, 91.73857, 87.32706, 84.53562, 92.31, 88.09571, 88.01048,
    98.694, 65.1875, 77.8134, 77.02262, 82.01258, 82.86664, 84.23183, 84.0166)

# combining track_length and average_time into a dataframe
track_data <- data.frame(track_length, round(average_time, 2))
```

Event length	Average Time
4 1/2F (4.5 Furlongs)	65.19
5F (5 Furlongs)	77.02
5 1/2F (5.5 Furlongs)	77.81
6F (6 Furlongs)	82.87
6 1/2F (6.5 Furlongs)	82.01
7F (7 Furlongs)	84.02
7 1/2F (7.5 Furlongs)	84.23
1M (1 Mile, 8 Furlongs)	88.01
1 1/16M	86.21
1 1/8M (1.125 Miles)	84.54
1 1/4M (1.25 Miles)	87.33
1 3/8M	88.10
1 1/2M (1.5 Miles, 12 Furlongs)	91.74
1 3/4M (1.75 Miles)	92.31
1M 70Y	98.69

Table 1: Table depicting average time to complete a race to complete a race of each length

Question 3

```
wdbn_data$floor_odds = round(wdbn_data$horse_odds_decimal)

tab1 = table(wdbn_data$floor_odds, wdbn_data$horse_place)
tab1

tab2 = round(prop.table(tab1, 1), 3)
tab2
```

Rounded Odds	Probability of 2nd
0	0.500
1	0.250
2	0.216
3	0.196
4	0.147
5	0.120
6	0.111
8	0.107
10	0.094
12	0.099
15	0.065
20	0.051
30	0.000
50	0.000

Table 2: Table depicting the probability of a horse coming in second place as a function of the decimal odds.

Question 4

```r
wdbn_6f = subset(wdbn_data, wdbn_data$track_length == "6F")
turf_data <- subset(wdbn_6f, track_type %in% c("Inner turf", "Turf"))
all_weather_data <- subset(wdbn_6f, track_type == "All Weather Track")

# Perform the two-sample t-test
t_test_result <- t.test(turf_data$time_final, all_weather_data$time_final)

tstat <- t_test_result$statistic
pval <- t_test_result$p.value

alpha <- 0.05
if (t_test_result$p.value < alpha) {
    cat("Reject the null hypothesis. There is a significant difference in average finish times.\n")
} else {
    cat("Fail to reject the null hypothesis. There is no significant difference in average finish times
}
```

T-Statistic: -0.1883969

P-Value: 0.8506062

Fail to reject the null hypothesis. There is no significant difference in average finish times.

Question 5

```
asba_6f = subset(asba_data, asba_data$track_length == "6F")
hast_6f = subset(hast_data, hast_data$track_length == "6F")


wdbn_box <- ggplot(wdbn_6f, aes(x = track_length, y = time_final)) + geom_boxplot(outlier.colour = "red
    outlier.shape = 8, outlier.size = 2) + scale_y_continuous(name = "Finish times (s)",
    breaks = seq(0, 200, 25)) + labs(x = "Track length (Furlongs) ") + ggtitle("Boxplot of track length
    theme(plot.title = element_text(hjust = 0.5))

asba_box <- ggplot(asba_6f, aes(x = track_length, y = time_final)) + geom_boxplot(outlier.colour = "red
    outlier.shape = 8, outlier.size = 2) + labs(y = "Finish times (s)", x = "Track length (Furlongs)") +
    ggtitle("Boxplot of track length on finish times at Assianboa") + theme(plot.title = element_text(h

hast_box <- ggplot(hast_6f, aes(x = track_length, y = time_final)) + geom_boxplot(outlier.colour = "red
    outlier.shape = 8, outlier.size = 2) + labs(y = "Finish times (s)", x = "Track length (Furlongs)") +
    ggtitle("Boxplot of track length on finish times at Hastings Park") + theme(plot.title = element_tex
```

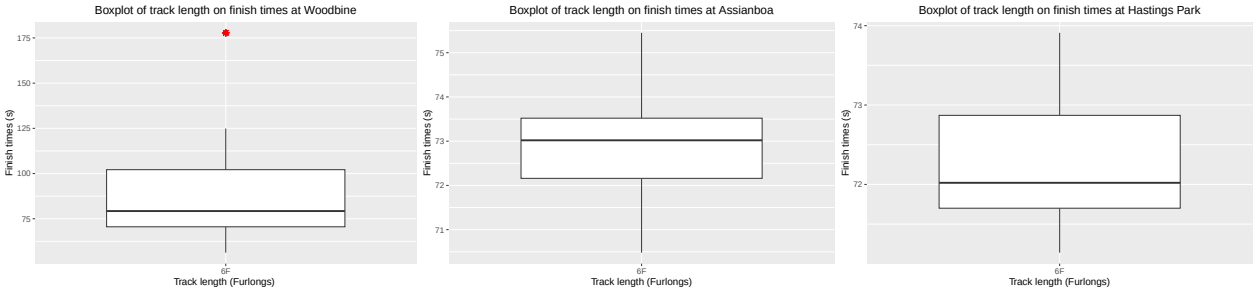

Figure 1: Boxplot of the finish times for 6F races between the three locations.

Note: The red star indictates outliers in the data.

Question 6

```
# use min and max year to ensure we are only looking at data between 2022 and
# 2023
min_yr = min(wdbn_data$meet_year)
max_yr = max(wdbn_data$meet_year)

place_tab = table(wdbn_data$horse_name, wdbn_data$horse_place)
place_tab

sorted_tab <- place_tab %>%
    as.data.frame() %>%
    arrange(desc(Freq))
sorted_tab

sorted_tab = subset(sorted_tab, sorted_tab$Var2 == 1)
```

Horse	Wins
Canadiansweetheart	8
Patches O'Houlihan	8
Hallie's Hero	6
Wentru	6
C C's Kingdom	5

Table 3: Table depicting the five horses that have won the most events at Woodbine

Question 7

```
# Find the min and max horse place
min_place = min(wdbn_data$horse_place, na.rm = TRUE)
min_place
max_place = max(wdbn_data$horse_place, na.rm = TRUE)
max_place
# Since we only have horse place from 1 to 5 we cant use that for the last 10%
# of the earnings however we can use it for the top 3 purse distributions.
```

```
sub_wdbn_data <- subset(wdbn_data, (!is.na(wdbn_data[, 14])))

num_horses = count(sub_wdbn_data, racecount)


sub_wdbn_data <- merge(sub_wdbn_data, num_horses, by = "racecount")

for (row in 1:nrow(sub_wdbn_data)) {
    sub_wdbn_data$payout[row] = 0
    if (isTRUE(sub_wdbn_data$horse_place[row] == 1) == TRUE) {
        sub_wdbn_data$payout[row] = sub_wdbn_data$purse[row] * 0.6
    } else if (isTRUE(sub_wdbn_data$horse_place[row] == 2) == TRUE) {
        sub_wdbn_data$payout[row] = sub_wdbn_data$purse[row] * 0.2
    } else if (isTRUE(sub_wdbn_data$horse_place[row] == 3) == TRUE) {
        sub_wdbn_data$payout[row] = sub_wdbn_data$purse[row] * 0.1
    } else {
        sub_wdbn_data$payout[row] = (wdbn_data$purse[row] * 0.1)/(sub_wdbn_data$n -
            3)
        # remember to divide by (number of horses - 3)
    }
}

total_pay = sub_wdbn_data %>%
    group_by(horse_name) %>%
    summarise(num = n(), totalpayout = sum(payout))

total_pay <- total_pay[order(total_pay$totalpayout, decreasing = TRUE), ]
```

Horse	Prize Money (in $)
Last Call	918,802
Patches O'Houlihan	698,862
Malibu Secret	690,930
Bushido	686,750
Moira	675,502

Table 4: Table depicting the five horses that have won the most prize money at Woodbine

Question 8

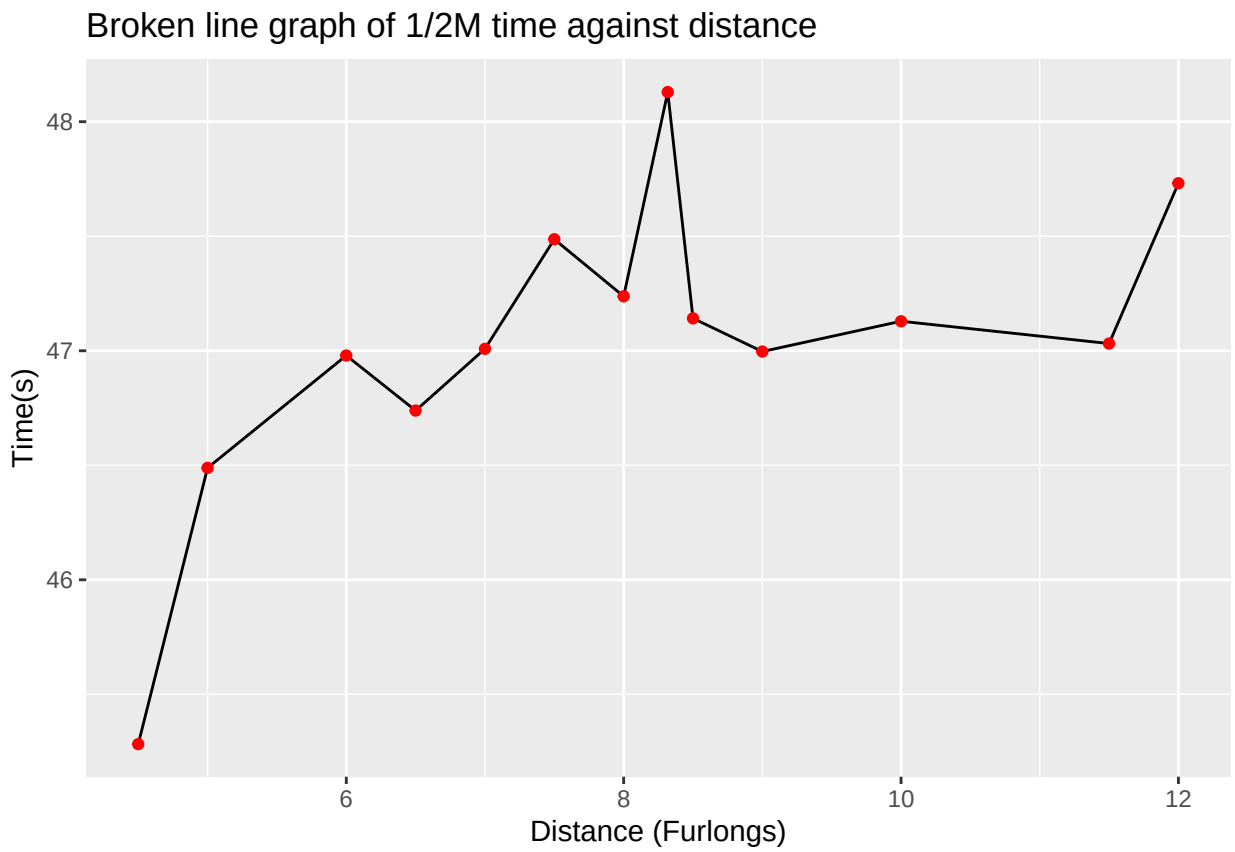

Figure 2: Broken line graph of average time the horse takes to complete 1/2 miles against distance in furlongs

```
unique(race_stats$track_length)

q8_df <- subset(race_stats, track_length != "1 3/4M")

unique(q8_df$track_length)

for (row in 1:nrow(q8_df)) {
    if (isTRUE(q8_df$track_length[row] == "4 1/2F") == TRUE) {
        q8_df$furlongs[row] = 4.5
    } else if (isTRUE(q8_df$track_length[row] == "5F") == TRUE) {
        q8_df$furlongs[row] = 5
    } else if (isTRUE(q8_df$track_length[row] == "6F") == TRUE) {
        q8_df$furlongs[row] = 6
    } else if (isTRUE(q8_df$track_length[row] == "6 1/2F") == TRUE) {
        q8_df$furlongs[row] = 6.5
    } else if (isTRUE(q8_df$track_length[row] == "7F") == TRUE) {
        q8_df$furlongs[row] = 7
    } else if (isTRUE(q8_df$track_length[row] == "7 1/2F") == TRUE) {
        q8_df$furlongs[row] = 7.5
    } else if (isTRUE(q8_df$track_length[row] == "1M") == TRUE) {
        q8_df$furlongs[row] = 8
    } else if (isTRUE(q8_df$track_length[row] == "1M 70Y") == TRUE) {
```

```
        q8_df$furlongs[row] = 8.318
    } else if (isTRUE(q8_df$track_length[row] == "1 1/16M") == TRUE) {
        q8_df$furlongs[row] = 8.5
    } else if (isTRUE(q8_df$track_length[row] == "1 1/8M") == TRUE) {
        q8_df$furlongs[row] = 9
    } else if (isTRUE(q8_df$track_length[row] == "1 1/4M") == TRUE) {
        q8_df$furlongs[row] = 10
    } else if (isTRUE(q8_df$track_length[row] == "1 3/8M") == TRUE) {
        q8_df$furlongs[row] = 11.5
    } else if (isTRUE(q8_df$track_length[row] == "1 1/2M") == TRUE) {
        q8_df$furlongs[row] = 12
    } else if (isTRUE(q8_df$track_length[row] == "1 3/4M") == TRUE) {
        q8_df$furlongs[row] = 13.75
    }
}
summary(q8_df$furlongs)

tapply(q8_df$time_frac2, q8_df$furlongs, mean)
# I am getting NA values using this method so i must check the data for NA in
# timefrac2

sum(is.na(q8_df$time_frac2))
# We do have NAs!

which(is.na(q8_df$time_frac2))
# now that we have located them and we only have 3/1710(1.75%) NA in the data I
# find it more useful to drop the rows rather than fill since they wont benfit
# the outcome of the mean

q8_df[c(1037, 1512, 1557), ]
# just to check the rows and ensure we are dropping the right ones

q8_df <- q8_df[-c(1037, 1512, 1557), ]

testin = tapply(q8_df$time_frac2, q8_df$furlongs, mean)

table_q8 = data.frame(sort(unique(q8_df$furlongs)), testin)

colnames(table_q8) <- c("furlongs", "mean_time2")
```

```
ggplot(table_q8, aes(x = furlongs, y = mean_time2)) +
  geom_line() +
  geom_point(col='red') +
  labs(x = "Distance (Furlongs)", y = "Time(s)") +
  ggtitle("Broken line graph of 1/2M time against distance")
```

Question 9

```
model = lm(q8_df$time_frac2 ~ I(q8_df$furlongs^2) + q8_df$furlongs)
summary(model)
```

```
##
## Call:
## lm(formula = q8_df$time_frac2 ~ I(q8_df$furlongs^2) + q8_df$furlongs)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.318 -1.233 -0.450  0.872 39.160
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         44.14491    1.20023  36.781   <2e-16 ***
## I(q8_df$furlongs^2) -0.03263    0.02338  -1.396   0.1629
## q8_df$furlongs       0.63771    0.33851   1.884   0.0598 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.385 on 1704 degrees of freedom
## Multiple R-squared:  0.01002,    Adjusted R-squared:  0.008854
## F-statistic:  8.62 on 2 and 1704 DF,  p-value: 0.0001884
```