Problem Set 1: Probability and Regression review

Alejandro Sarria

2025-09-02

Set up

```
library(ggplot2)
library(tidyverse)
library(here)
set.seed(1)

df <- readRDS('./data/nyc_marathon.RDS')
```

1. Probability review

1.a. Marginal probability P(Kenya) and P(USA)

```
total <- 3+4+3+38000+9+18000

p_kenya <- (4+9)/total
p_usa <- (3+38000)/total

sprintf("P(Kenya): %f, P(USA): %f", p_kenya, p_usa)
```

```
[1] "P(Kenya): 0.000232, P(USA): 0.678395"
```

1.b. P(Kenya, Top10), P(Kenya, ~Top10)

```
p_kenya_top10 <- 4/total
p_kenya_not10 <- 9/total

sprintf("P(Kenya, Top10): %f, P(Kenya, ~Top10): %f", p_kenya_top10,
        p_kenya_not10)
```

```
[1] "P(Kenya, Top10): 0.000071, P(Kenya, ~Top10): 0.000161"
```

1.c. P(Top10|Kenya), P(Kenya|Top10)

```
p_top <- 10/total

p_top10_c_kenya <- p_kenya_top10 / p_kenya
p_kenya_c_top10 <- p_kenya_top10 / p_top

sprintf("P(Top10|Kenya): %f, P(Kenya|Top10): %f", p_top10_c_kenya,
        p_kenya_c_top10)
```

```
[1] "P(Top10|Kenya): 0.307692, P(Kenya|Top10): 0.400000"
```

1.d. Using conditional probabilities to compare US and Kenyan runners

What is interesting from Kenyan runners in the joint distribution table is that, although their numbers in the Top 10 are similar to US runners (4 vs. 3), their representatives in the Top 10 amount for a sizable chunk of their participants. Using conditional probabilities, this can be shown by comparing the P(~Top10|Kenya) with P(~Top10|USA).

```
p_not10_c_USA <- (38000/total) / p_usa
p_not10_c_kenya <- (9/total) / p_kenya

sprintf("P(~Top10|Kenyan): %f, P(~Top10|USA): %f", p_not10_c_kenya,
        p_not10_c_USA)
```

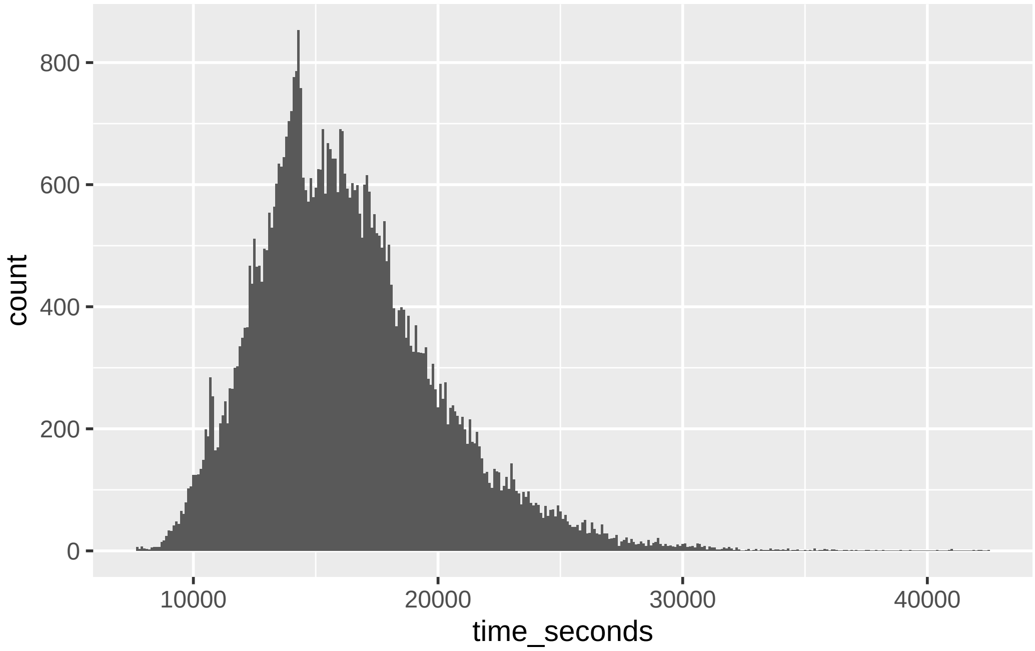
```
[1] "P(~Top10|Kenyan): 0.692308, P(~Top10|USA): 0.999921"
```

The probability of finishing outside the Top 10 while being american is much higher compared to the probability of doing so for Kenyan runners, showing the much better performance of Kenyans in the NYC Marathon.

2. Regression review

a. Distribution of times

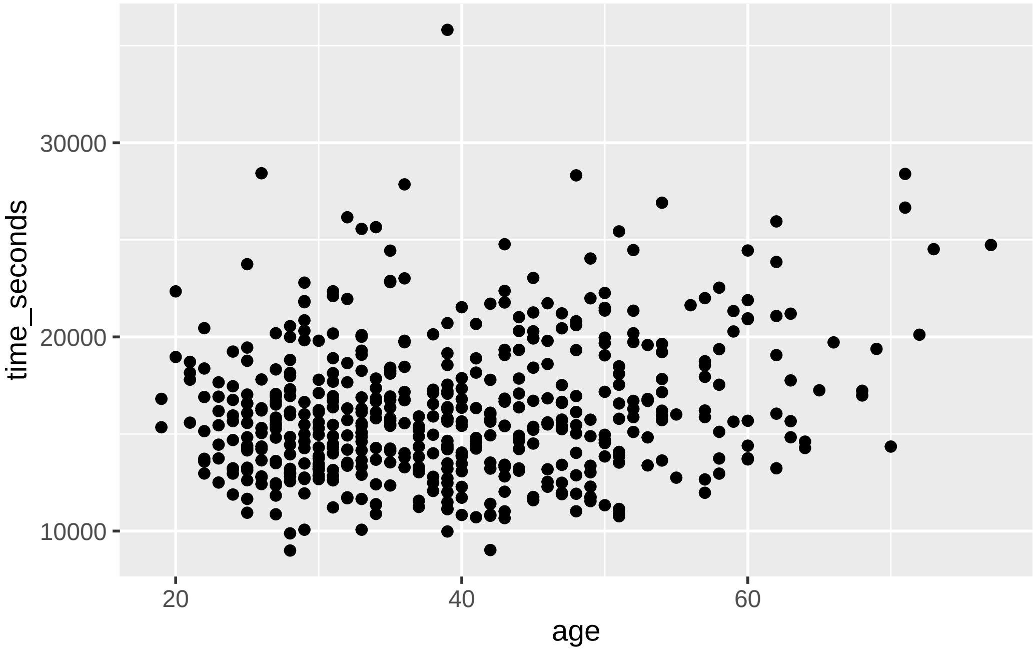
```r
p <- df |>
  ggplot(aes(x=time_seconds)) +
    geom_histogram(binwidth=100)

p
```



The finishing times for the NYC marathon seems to follow a bi modal distribution, with two concentration of times: one right below 4 hours and the other right above. The causes behind this distribution is surely complex. The two modes may be a function of gender, with the 4 hour mark being the usual cut-off time to qualify to other top maratons.

b. Sample and time vs. age

```
df_sample <- df |>
  sample_n(size = 500)

p_sample <- df_sample |>
  ggplot(aes(x=age, y=time_seconds)) +
    geom_point()
p_sample
```



c. Linear regression

```
model_sample <- lm(formula = time_seconds ~ age,
             data = df_sample)
summary(model_sample)
```

```
Call:
lm(formula = time_seconds ~ age, data = df_sample)
```

```
Residuals:
    Min      1Q  Median      3Q     Max 
-7370.2 -2390.1  -472.8  1916.0 19621.1 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 13603.41     564.14  24.113  < 2e-16 ***
age            66.45      13.81   4.813 1.97e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3550 on 498 degrees of freedom
Multiple R-squared:  0.04445,	Adjusted R-squared:  0.04253 
F-statistic: 23.17 on 1 and 498 DF,  p-value: 1.973e-06
```

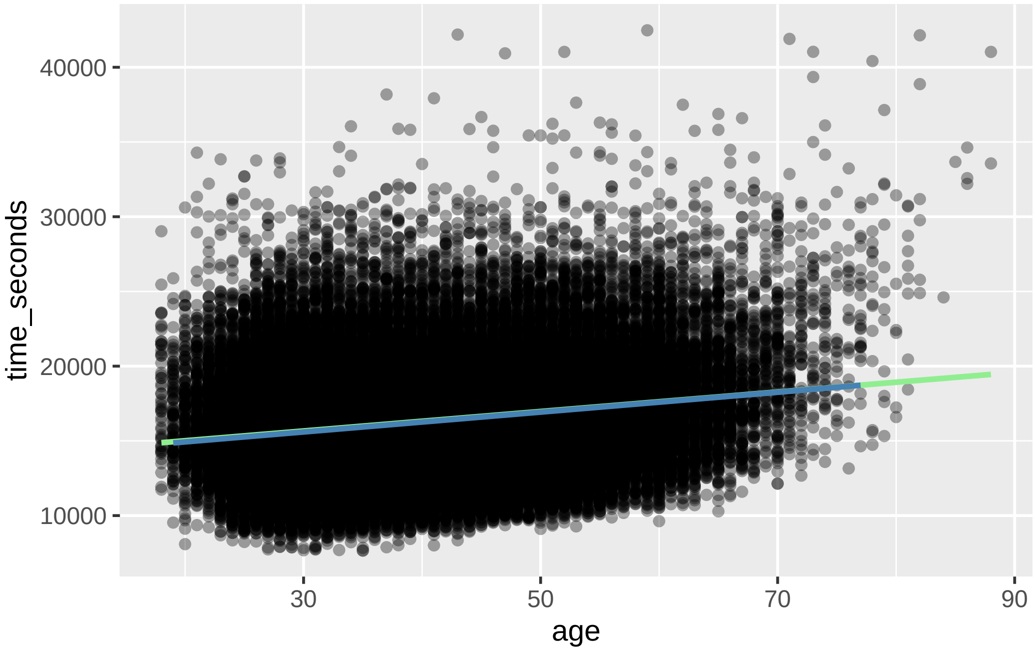
Coefficient of age means that a unit increase in age (that is, every extra year) predicts and extra 66.45 seconds in finishing time.The intercept - 13603.41 seconds - represents an approximation of the estimated time if the age of a participant was 0 , showing how the model cannot understand the true meaning of the age variable in this case.

d. Predict based on model

```r
pred_time <- function(age){
  coef(model_sample)[1] + coef(model_sample)[2]*age
}

for (x in c(5, 20, 55, 110)) {
  sprintf("Predicted finishing time for a runner of age %f: %f",
          x, pred_time(x))
}
```

e. Full vs sample model

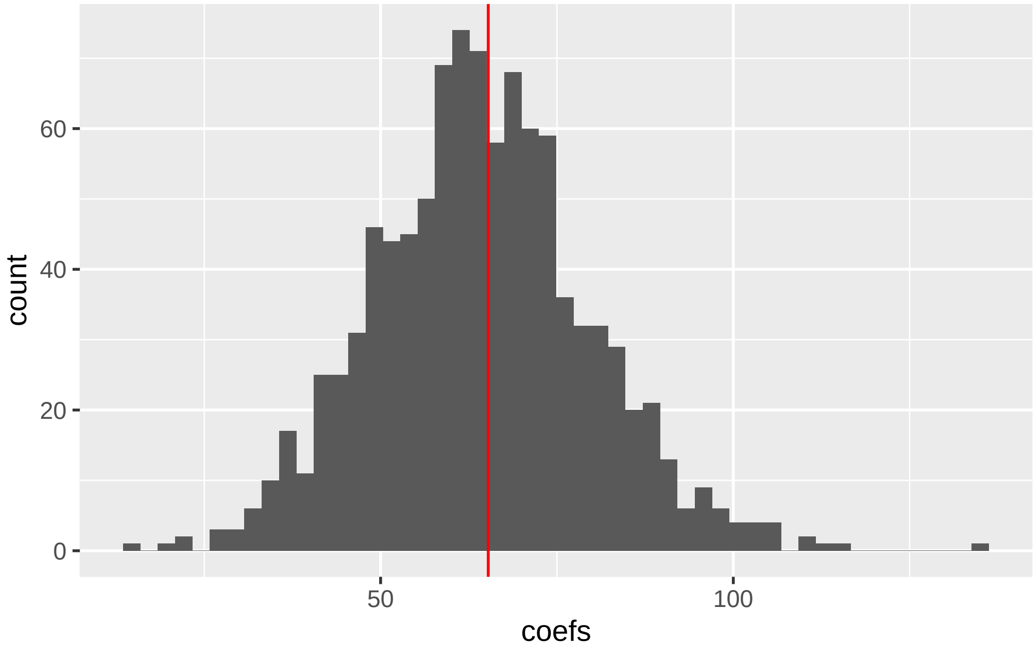
```r
p <- df |>
  ggplot(aes(x=age, y=time_seconds)) +
    geom_point(alpha=0.35) +
    geom_smooth(method="lm", se=FALSE, color="lightgreen") +
    geom_smooth(data=df_sample, aes(x=age, y=time_seconds),
                  method="lm", se=FALSE, color="steelblue")
p
```



The two lines are mostly similar. The figure shows that they are mostly overlapping, suggesting that their slopes-which corresponds to the coefficient for age- is close enough. There is a minuscule level of divergence at the lower values of the range, where there are more observations, showing that in the parts of the distribution where the amount of data in the sample difers form the one in the population, the the difference in coefficients is more notorious.

f. Distribution of coefficients

```
coefs <- c()

for (i in 1:1000){
  samp <- df |>
    sample_n(size = 500)

  temp_model <- lm(formula = time_seconds ~ age,
            data = samp)

  coefs <- c(coefs, coef(temp_model)[2])
}

full_model <- lm(formula=time_seconds ~ age,
                 data=df)

coef_df <- data.frame(coefs = coefs)

p <- coef_df |>
  ggplot(aes(x=coefs)) +
    geom_histogram(bins = 50) +
    geom_vline(xintercept = coef(full_model)[2], color='red')
p
```



The sample coefficients follow a normal distribution with a mean close to the coefficient of the population model. This shows that, in most cases, the sample model generates a coefficient similar to the true coefficient of the population.