Formas Funcionales (parte 1)

```
knitr::opts_chunk$set(warning = FALSE, message = FALSE)
library(tidyverse)
library(tidymodels)
```

4. Ejercicio

Hay veces en las que ajustar una regresión lineal con datos crudos no es adecuado, por lo que se aplican transformaciones lineales y logaritmicas para poder interpretar el modelo. **Las transformaciones lineales no afectan el ajuste de un modelo de regresión y no afectan las predicciones**. Por otra parte, cambios en los inputs y coeficientes, **pueden mejorar la interpretabilidad** de los coeficientes y hacer el modelo más facil de interpretar.

Los coeficientes de regresión β_j representan la **diferencia promedio** de y cuando el predictor x_i cambia en una unidad. Es por esto, que al hablar de escalas originales, nos podemos dar cuenta que el coeficiente está relacionado con la escala del regresor. Analicemos el siguiente ejercicio:

Ejercicio 4.1: Se utilizarán datos de una encuensta de salarios en Estados Unidos que predice el salario basado en la altura de la persona (en pulgadas), para esto primero cargaremos los datos de la url:

```
wages <-
  read_csv("https://raw.githubusercontent.com/Clark-Rhodes/INFO523/99c046debb9230fbfedaf08a67577a9a0c37

wages %>% head
```

```
## # A tibble: 6 x 6
##      earn height sex    race     ed   age
##     <dbl>  <dbl> <chr>  <chr> <dbl> <dbl>
## 1 79571.   73.9 male   white    16    49
## 2 96397.   66.2 female white    16    62
## 3 48711.   63.8 female white    16    33
## 4 80478.   63.2 female other    16    95
## 5 82089.   63.1 female white    17    43
## 6 15313.   64.5 female white    15    30
```

De estos datos podemos observar que la altura está en pulgadas, por lo que primero la pasaremos a centímetros y a metros:

Ejercicio 4.2: Convierta la variable de height para tenerla en centímetros y en metros

```
wages <-
  wages %>%
  mutate(height_cm = height*2.54,
         height_m = height*2.54/100) %>%
  select(earn, height, height_cm, height_m)
wages %>% head
```

```
## # A tibble: 6 x 4
```

```
##     earn height height_cm height_m
##    <dbl>  <dbl>     <dbl>    <dbl>
## 1 79571.   73.9      188.     1.88
## 2 96397.   66.2      168.     1.68
## 3 48711.   63.8      162.     1.62
## 4 80478.   63.2      161.     1.61
## 5 82089.   63.1      160.     1.60
## 6 15313.   64.5      164.     1.64
```

Ejercicio 4.3: Realice la regresión lineal de earn~height_cm y earn~height_m, Interprete. ¿Qué observa de los coeficientes?, ¿Por qué sucede esto? ¿Hace sentido tener un intercept cuando height_cm = 0 o height_m = 0?. ¿Cuál es la R^2 del modelo?

```
model_fit_cm <-
  linear_reg()%>%
  fit(earn~height_cm, data = wages)

model_fit_m <-
  linear_reg()%>%
  fit(earn~height_m, data = wages)

tidy(model_fit_cm)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept) -126523.   14076.      -8.99 8.05e-19
## 2 height_cm       940.      83.1     11.3  1.96e-28
```

```
tidy(model_fit_m)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept) -126523.    14076.     -8.99 8.05e-19
## 2 height_m      93984.     8308.     11.3  1.96e-28
```

```
glance(model_fit_cm)
```

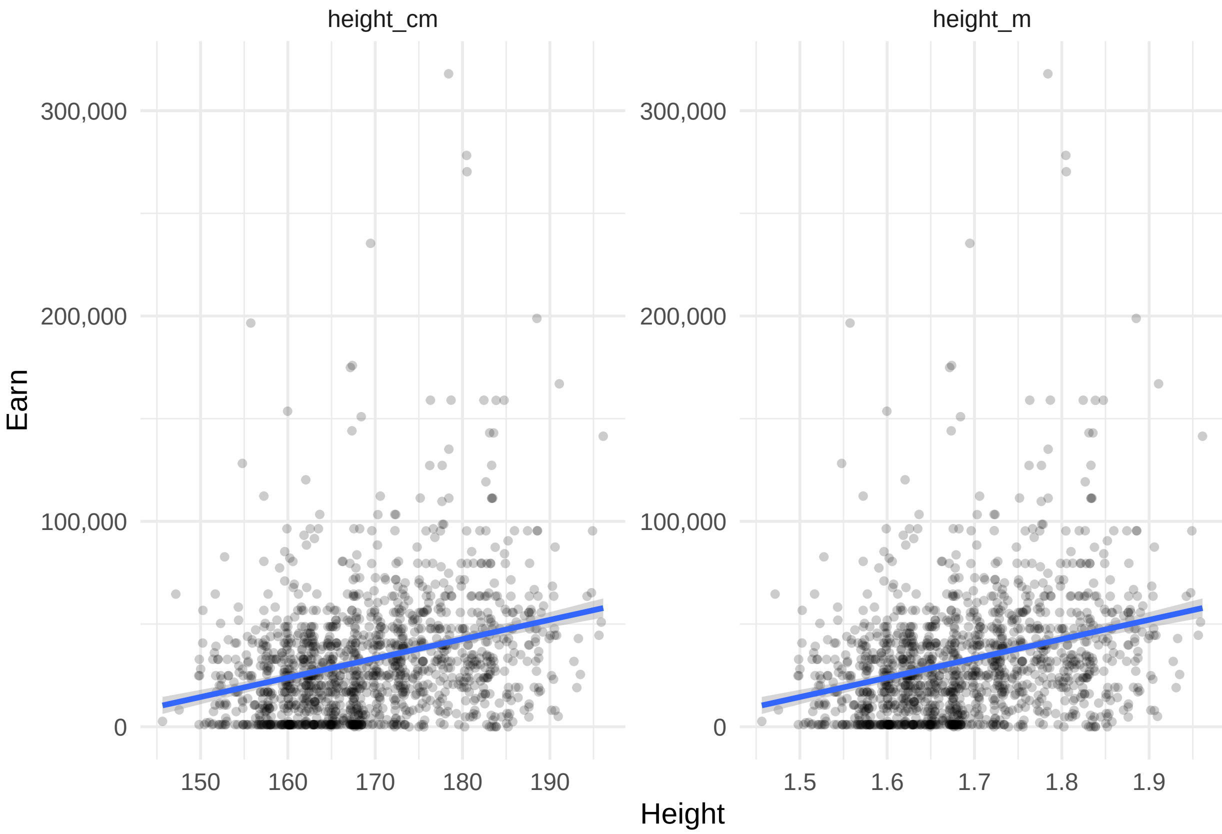
```
## # A tibble: 1 x 12
##   r.squared adj.r.squared  sigma statistic  p.value    df  logLik    AIC    BIC
##       <dbl>         <dbl>  <dbl>     <dbl>    <dbl> <dbl>   <dbl>  <dbl>  <dbl>
## 1    0.0850        0.0844 29910.      128. 1.96e-28     1 -16168. 32341. 32357.
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

```
glance(model_fit_m)
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared  sigma statistic  p.value    df  logLik    AIC    BIC
##       <dbl>         <dbl>  <dbl>     <dbl>    <dbl> <dbl>   <dbl>  <dbl>  <dbl>
## 1    0.0850        0.0844 29910.      128. 1.96e-28     1 -16168. 32341. 32357.
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

Podemos observar que β_0 en ambos casos es -125,523, pero el dato de β_1 difiere según el caso, para la regresión en metros, $\beta_1 = 93,984$, mientras que para la regresión en centímetros $\beta_1 = 939.84$. Al interpretar β_0 vemos que no tiene mucha interpretación, un salario negativo no hace sentido, ya que tampoco hace sentido que una persona mida 0 centímetros. Con este ejercicio podemos observar que los coeficientes de β_1 varian dependiendo la escala. La R^2 en ambos casos es de 0.085 (no afecta la calidad del modelo)

```
wages %>%
  gather(escala, valor, -c(earn, height)) %>%
  ggplot(aes(x = valor, y = earn))+
  geom_point(size = 1, alpha = 0.2)+
  geom_smooth(method = "lm")+
  facet_wrap(~escala, scales = "free")+
  theme_minimal()+
  scale_y_continuous(labels = scales::comma_format())+
  labs(x = "Height", y = "Earn")
```



En estas gráficas vemos que realmente no hace sentido tener un intercept cuando Height = 0, ya que nunca tenemos valores en ese rango, la interpretación en este caso es que en promedio una persona que mide 0 cm, gana -125,523. Por este motivo sería una buena idea centrar la variable. El intercepto en este caso no lo podemos interpretar.

Ejercicio 4.4: Centre las variables height_cm, height_m, ajuste nuevamente y observe los resultados. ¿El valor de β_0 cambió? ¿Cómo interpreta este nuevo valor? ¿El valor de β_1 cambió? ¿Cómo lo interpreta? Concluya

```
wages_cent <-
  wages %>%
  mutate(height_cm_center = scale(height_cm, center = T, scale = F),
         height_m_center = scale(height_m, center = T, scale = F))

model_fit_cm_center <-
```

```
  linear_reg() %>%
  fit(earn ~ height_cm_center, data = wages_cent)

tidy(model_fit_cm_center)
```

```
## # A tibble: 2 x 5
##   term             estimate std.error statistic   p.value
##   <chr>               <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)        32446.     805.       40.3 4.36e-235
## 2 height_cm_center     940.      83.1      11.3 1.96e- 28
```

```
glance(model_fit_cm_center)
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared  sigma statistic  p.value    df  logLik    AIC    BIC
##       <dbl>         <dbl>  <dbl>     <dbl>    <dbl> <dbl>   <dbl>  <dbl>  <dbl>
## 1    0.0850        0.0844 29910.      128. 1.96e-28     1 -16168. 32341. 32357.
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

```
mean(wages_cent$height_cm)
```

```
## [1] 169.1453
```

Se tiene que centramos los datos con una media de 169.1453, por lo que ahora el "nuevo" cero representa este dato. Los coeficientes en este caso cambian, el intercepto ahora explica el salario promedio para una estatura en centímetros promedio, es decir cuando una persona mide 169.1453, su salario es de $32,446, por otra parte, por cada centímetro adicional, su salario incrementa en promedio $939 dólares.

Algo importante es que la R^2 sigue siendo la misma 0.085, es decir no perdimos calidad de la regresión al centrar la variable, pero ganamos interpretabilidad

Este modelo todavía tiene un problema, si cambiamos los datos a metros el valor de β_1 sigue moviendose, es decir se sigue afectando por la escala de la variable (de cm a metros).

Ejercicio 4.5: Centre y escale las variables height__cm, height__m, ajuste nuevamente y observe los resultados. ¿El valor de β_0 cambió? ¿Cómo interpreta este nuevo valor? ¿El valor de β_1 cambió? ¿Cómo lo interpreta? Concluya

```
sd(wages$height_cm)
```

```
## [1] 9.697993
```

```
sd(wages$height_m)
```

```
## [1] 0.09697993
```

```
wages <- wages %>%
  mutate(height_cm_est = scale(height_cm, center = T, scale = T),
         height_m_est = scale(height_m, center = T, scale = T))

wages %>% head
```

```
## # A tibble: 6 x 6
##     earn height height_cm height_m height_cm_est[,1] height_m_est[,1]
##    <dbl>  <dbl>     <dbl>    <dbl>             <dbl>            <dbl>
## 1 79571.   73.9      188.     1.88             1.91             1.91
## 2 96397.   66.2      168.     1.68            -0.0950          -0.0950
## 3 48711.   63.8      162.     1.62            -0.739           -0.739
```

```
## 4 80478.   63.2      161.     1.61           -0.883           -0.883
## 5 82089.   63.1      160.     1.60           -0.920           -0.920
## 6 15313.   64.5      164.     1.64           -0.540           -0.540
```

```
model_fit_cm_scaled <-
  linear_reg() %>%
  fit(earn ~ height_cm_est, data = wages)

tidy(model_fit_cm_scaled)
```

```
## # A tibble: 2 x 5
##   term          estimate std.error statistic   p.value
##   <chr>            <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)     32446.      805.      40.3 4.36e-235
## 2 height_cm_est    9115.      806.      11.3 1.96e- 28
```

```
glance(model_fit_cm_scaled)
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared  sigma statistic  p.value    df  logLik    AIC    BIC
##       <dbl>         <dbl>  <dbl>     <dbl>    <dbl> <dbl>   <dbl>  <dbl>  <dbl>
## 1    0.0850        0.0844 29910.      128. 1.96e-28     1 -16168. 32341. 32357.
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

¿Que podemos observar de estos datos? en la versión estandarizada, ambas variables (en centimetros y en metros) tienen los mismos valores, esto quiere decir que al estandarizar la variable eliminamos el efecto de la escala. Adicional, podemos observar que la desviación estándar de la altura en centímetros es de 9.6979 y la desviación estándar de la altura en metros es de 0.9697, ahora el cambio en una unidad representará el cambio en una desviación estándar.

La R^2 sigue siendo la misma

Ejercicio 4.6: Realize una tabla comparativa de los modelos y su interpretación de los coeficientes.