Linear Regression

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.4 ✔ readr 2.1.5  
## ✔ forcats 1.0.0 ✔ stringr 1.5.1  
## ✔ ggplot2 3.5.1 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.3 ✔ tidyr 1.3.1  
## ✔ purrr 1.0.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(modelr)  
library(broom)

##   
## Attaching package: 'broom'  
##   
## The following object is masked from 'package:modelr':  
##   
## bootstrap

library(readxl)  
  
# Import wages here  
wages <- read\_excel("wages-1.xlsx", na = "NA")  
  
# Fall back in case you cannot load wages  
# wages <- heights %>%  
# filter(income > 0) %>%  
# mutate(marital = as.character(marital),  
# sex = as.character(sex))

## Your Turn 1

* Save the wages.xlsx dataset to your computer.
* Change the working directory to the same location.
* Import wages.xlsx as wages and *copy the code to your setup chunk*.
* Be sure to set NA: to NA.

## Your Turn 2

Fit the model on the slide and then examine the output. What does it look like?

mod\_e <- lm(log(income)~ education, data = wages)  
mod\_e

##   
## Call:  
## lm(formula = log(income) ~ education, data = wages)  
##   
## Coefficients:  
## (Intercept) education   
## 8.5577 0.1418

class(mod\_e)

## [1] "lm"

## Your Turn 3

Use a pipe to model log(income) against height. Then use broom and dplyr functions to extract:

1. The **coefficient estimates** and their related statistics
2. The **adj.r.squared** and **p.value** for the overall model

mod\_h <- wages %>%   
 lm(log(income)~height, data =.)  
  
mod\_h %>% tidy()

## # A tibble: 2 × 5  
## term estimate std.error statistic p.value  
## <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 (Intercept) 6.98 0.237 29.4 4.13e-176  
## 2 height 0.0520 0.00352 14.8 2.44e- 48

mod\_h %>%   
 glance() %>%   
 select(adj.r.squared, p.value)

## # A tibble: 1 × 2  
## adj.r.squared p.value  
## <dbl> <dbl>  
## 1 0.0396 2.44e-48

## Your Turn 4

Model log(income) against education and height and sex. Interpret the coefficients in narrative here.

mod\_ehs <- wages %>%   
 lm(log(income)~education + height + sex, data = .)  
  
mod\_ehs %>% tidy()

## # A tibble: 4 × 5  
## term estimate std.error statistic p.value  
## <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 (Intercept) 7.79 0.307 25.3 9.41e-134  
## 2 education 0.148 0.00520 28.5 5.16e-166  
## 3 height 0.00673 0.00479 1.40 1.61e- 1  
## 4 sexmale 0.462 0.0389 11.9 5.02e- 32

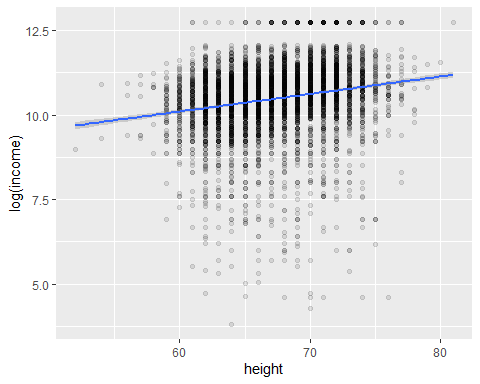
#The sexmale term is an indicator variable (0,1). As a result, if the person is a male they get 0.46 but  
#if they are a female, they get 0. We also see that taller people are expected to get paid   
#slightly more than shorter ones.

## Your Turn 5

Add + geom\_smooth(method = lm) to the code below. What happens?

wages %>%  
 ggplot(aes(x = height, y = log(income))) +  
 geom\_point(alpha = 0.1) +   
 geom\_smooth(method=lm)

## `geom\_smooth()` using formula = 'y ~ x'

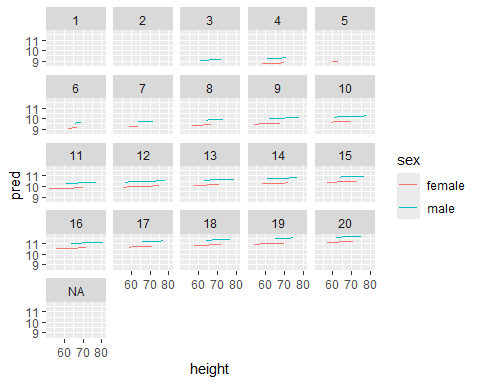


#geom\_smooth adds a line for predicting how height impacts income.

## Your Turn 6

Use add\_predictions to make the plot below.

# In case you haven't made the ehs model  
mod\_ehs <- wages %>% lm(log(income) ~ education + height + sex, data = .)  
  
# Make plot here  
wages %>%   
 add\_predictions(mod\_ehs) %>%   
 ggplot(mapping = aes(x = height, y = pred, color = sex)) +  
 geom\_line() +   
 facet\_wrap(~ education)



# Take Aways

* Use glance(), tidy(), and augment() from the **broom** package to return model values in a data frame.
* Use add\_predictions() or gather\_predictions() or spread\_predictions() from the **modelr** package to visualize predictions.
* Use add\_residuals() or gather\_residuals() or spread\_residuals() from the **modelr** package to visualize residuals.