Lab 0

Sawyer Balint

2024-10-25

```r
#create a custom graphing theme to reduce repetition
theme <- list(
  theme_classic(),
  scale_color_jco(),
  scale_fill_jco(),
  scale_shape_manual(values=c(21:25)),
  theme(legend.title=element_blank(),
        legend.position="inside",
        legend.position.inside=c(0.2,0.8))
)
```

First we model

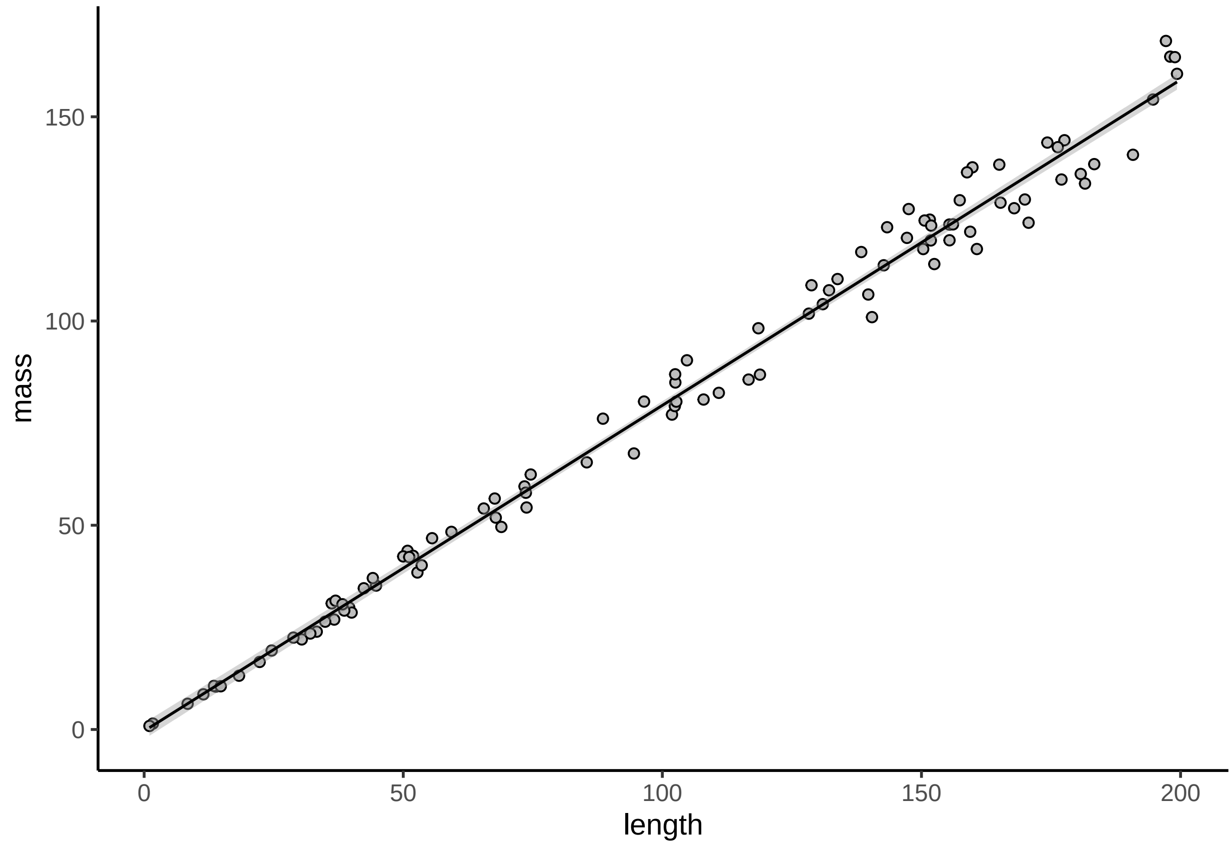
Population 1: constant cross-section

```r
#normal distribution of lengths
lengths <- runif(n=100, min=1, max=200)

#i'm also modeling density as a distribution because i want some noise
density <- runif(n=100, min=0.9, max=1.1)

#create dataframe of volume
cylinders.df <- data.frame(length=lengths,
                           density=density) %>%
  mutate(height=1,
         width=1,
         volume=pi*(height/2)*(width/2)*length,
         mass=volume*density)

#check data structure
str(cylinders.df)
```

```
## 'data.frame':    100 obs. of  6 variables:
##  $ length : num  155.4 18.3 159.4 131 36.7 ...
##  $ density: num  0.981 0.915 0.973 1.012 0.934 ...
##  $ height : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ width  : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ volume : num  122.1 14.4 125.2 102.9 28.8 ...
##  $ mass   : num  119.8 13.2 121.9 104.1 26.9 ...
```

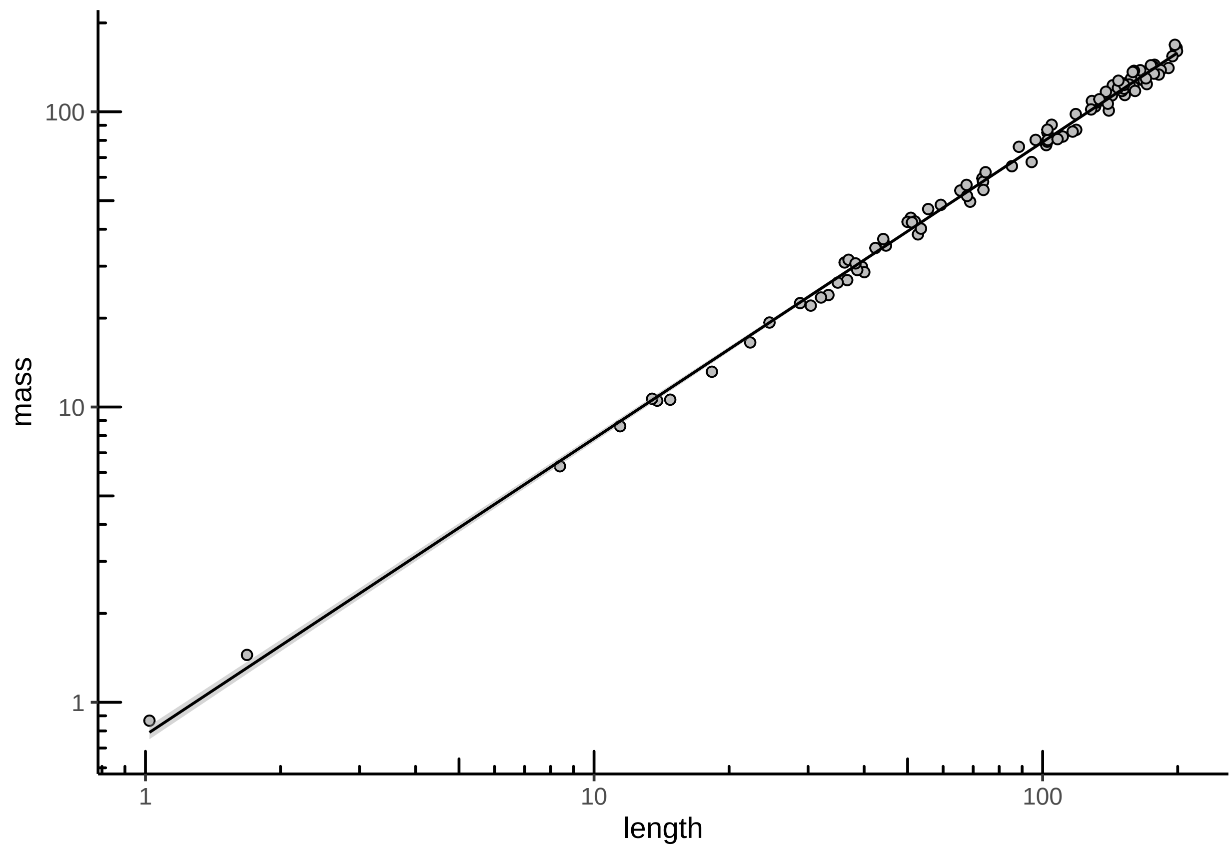
```r
#relationship between length and mass
plot <- ggplot(cylinders.df, aes(length, mass))+
  theme+
  geom_point(shape=21, fill="grey")+
  geom_smooth(method="lm", color="black", linewidth=0.5)

plot
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```r
#add log axes to plot
plot +
  scale_y_log10()+
  scale_x_log10()+
  annotation_logticks()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



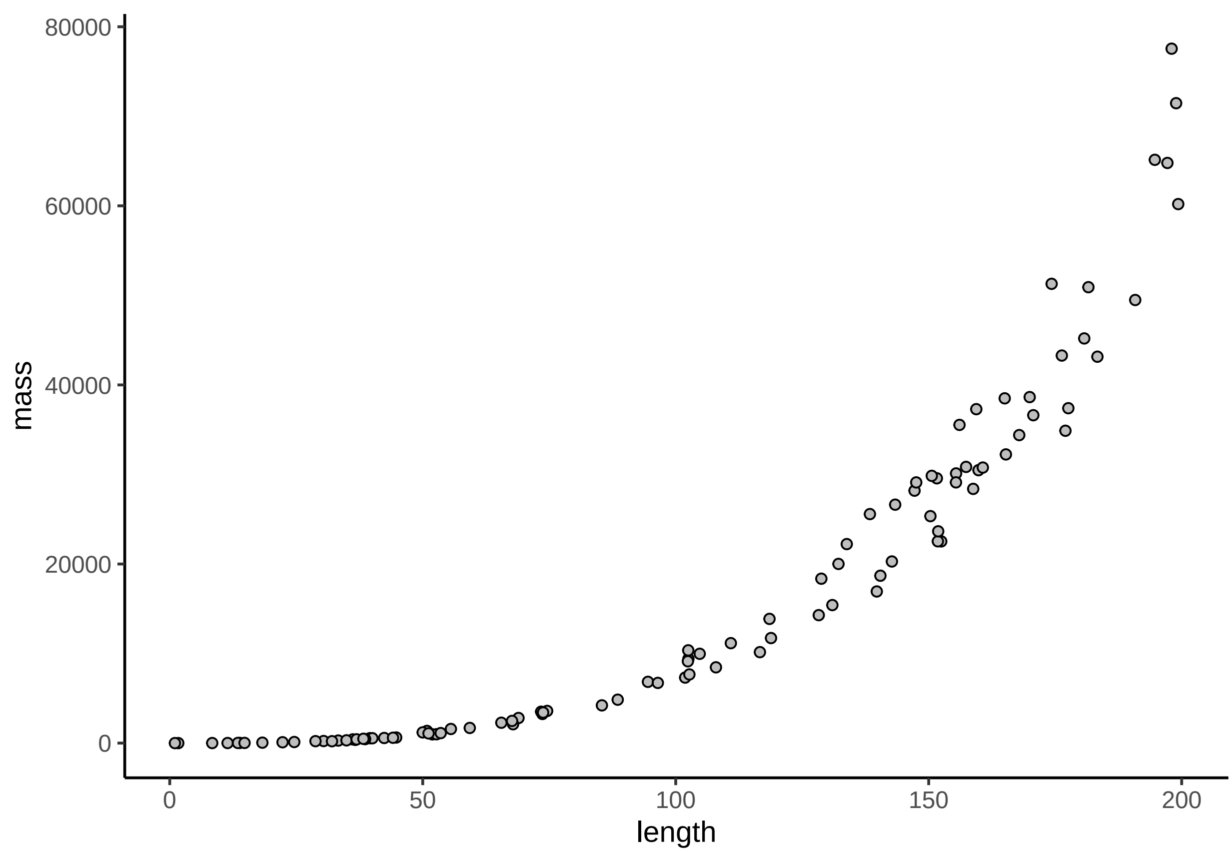

```r
#make a model
cylinders.lm <- lm(log(cylinders.df$mass)~(log(cylinders.df$length)))
summary(cylinders.lm)
```

```
##
## Call:
## lm(formula = log(cylinders.df$mass) ~ (log(cylinders.df$length)))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.099493 -0.044218  0.000594  0.046583  0.101495
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -0.255429   0.026925  -9.487 1.58e-15 ***
## log(cylinders.df$length)  1.004270   0.006046 166.098  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05757 on 98 degrees of freedom
## Multiple R-squared:  0.9965, Adjusted R-squared:  0.9964
## F-statistic: 2.759e+04 on 1 and 98 DF,  p-value: < 2.2e-16
```

```r
slope1 <- coefficients(cylinders.lm)[2]
```

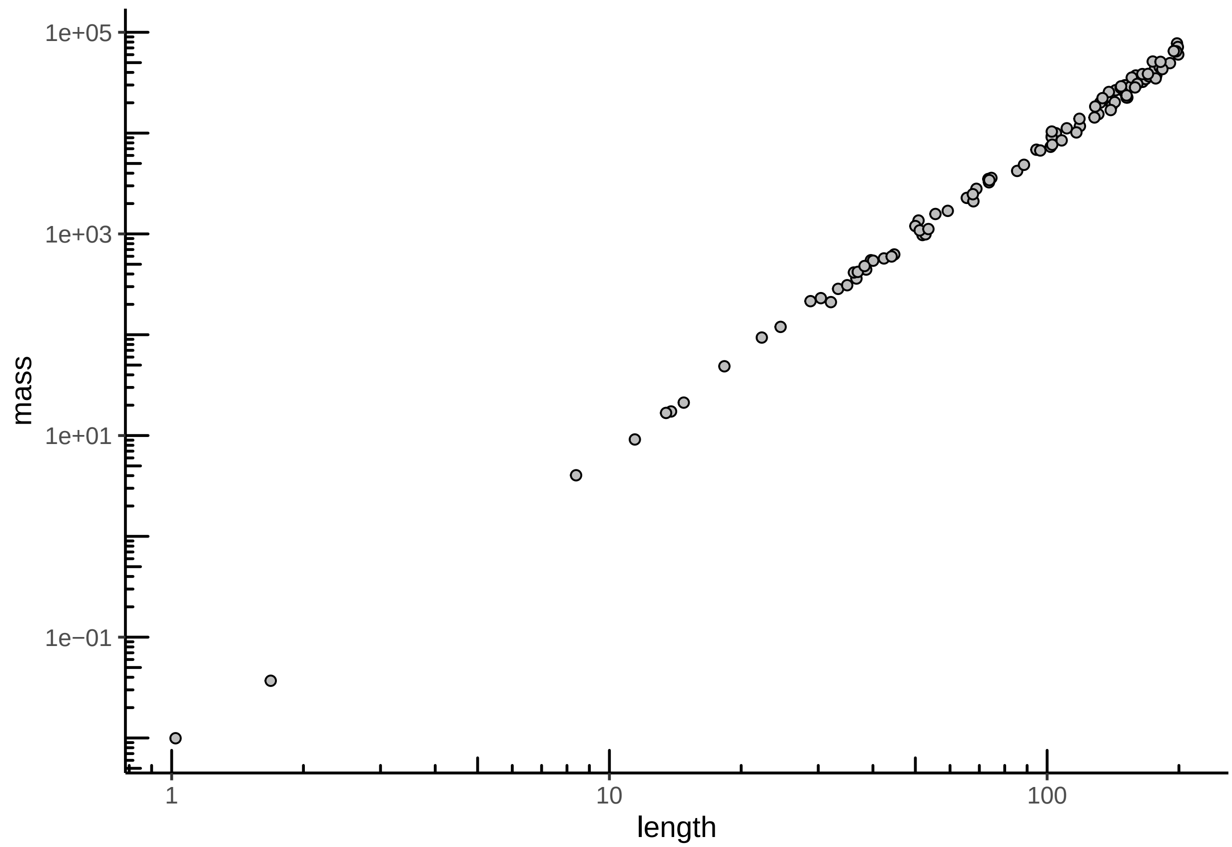
Population 2: cross-section changes in proportion to length

```
# code for cylindrical fish population 2

#relationship between width and height (nominally 0.1)
#I'm adding adding some noise for fun
width_factor <- runif(n=100, min=0.09, max=0.11)

cylinders.df <- cylinders.df %>%
  mutate(height=length*width_factor,
         width=height,
         volume=pi*(height/2)*(width/2)*length,
         mass=volume*density)

#make base plot
plot <- ggplot(cylinders.df, aes(length, mass))+
  theme+
  geom_point(shape=21, fill="grey")

#print the plot
plot
```




```
#make it log
plot +
  scale_x_log10()+
```

```
  scale_y_log10()+
  annotation_logticks()
```



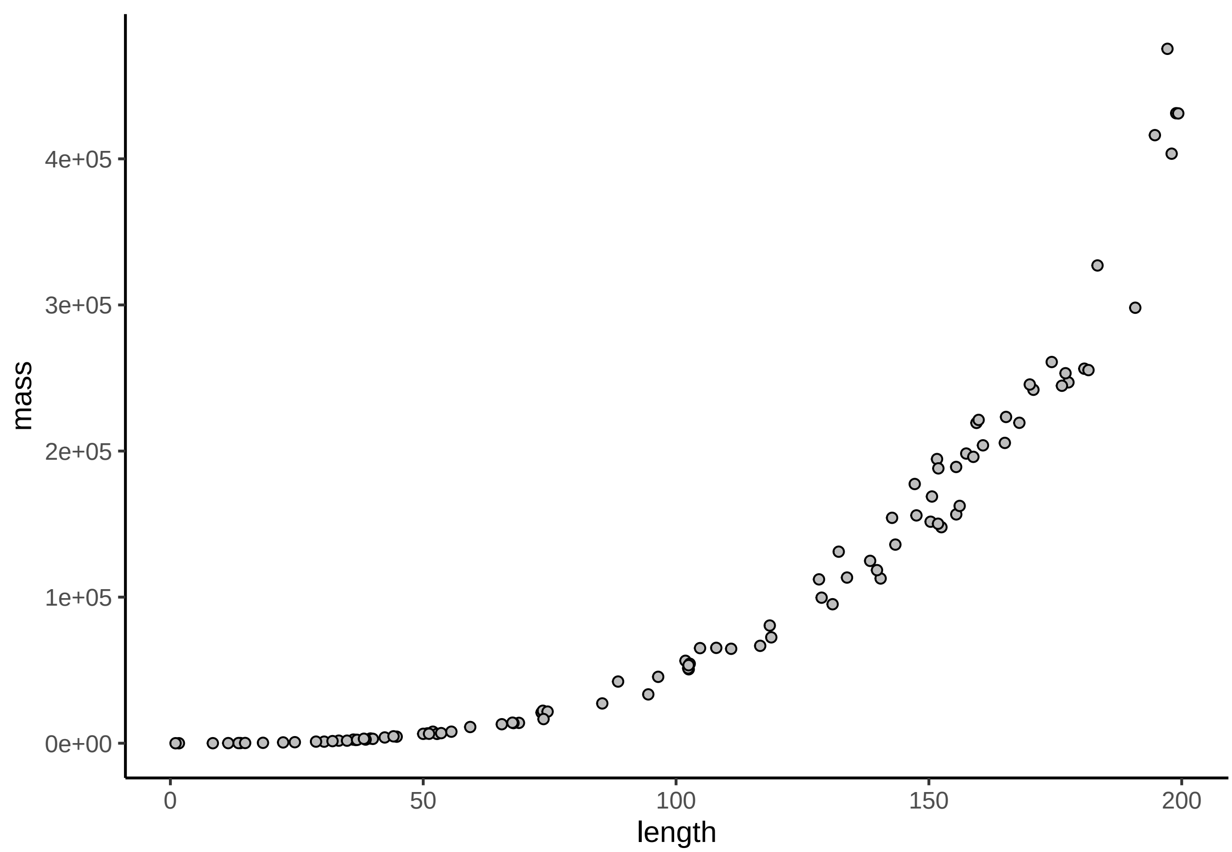

```
#make a model
cylinders.lm <- lm(log(cylinders.df$mass)~(log(cylinders.df$length)))
summary(cylinders.lm)
```

```
##
## Call:
## lm(formula = log(cylinders.df$mass) ~ (log(cylinders.df$length)))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.23465 -0.11374  0.01082  0.09696  0.28210
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -4.87904    0.05904  -82.64   <2e-16 ***
## log(cylinders.df$length)  3.00637    0.01326  226.77   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1262 on 98 degrees of freedom
## Multiple R-squared:  0.9981, Adjusted R-squared:  0.9981
## F-statistic: 5.142e+04 on 1 and 98 DF,  p-value: < 2.2e-16
```

```
slope2 <- coefficients(cylinders.lm)[2]
```

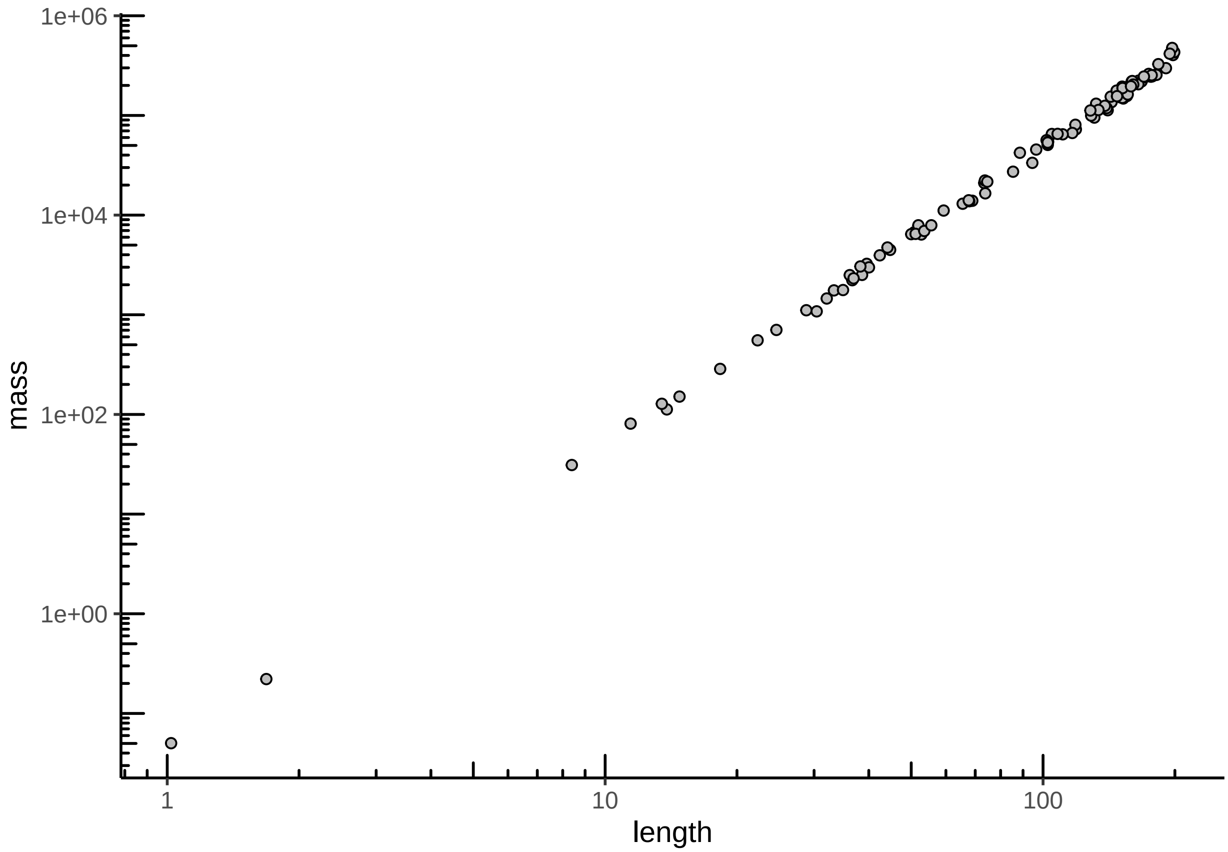
Challenge scenario 1

```
#chonky fish
width_factor <- runif(n=100, min=0.23, max=0.27)

cylinders.df <- cylinders.df %>%
  mutate(height=length*width_factor,
         width=height,
         volume=pi*(height/2)*(width/2)*length,
         mass=volume*density)

#make base plot
plot <- ggplot(cylinders.df, aes(length, mass))+
  theme+
  geom_point(shape=21, fill="grey")

#print the plot
plot
```




```
#make it log
plot +
  scale_x_log10()+
```

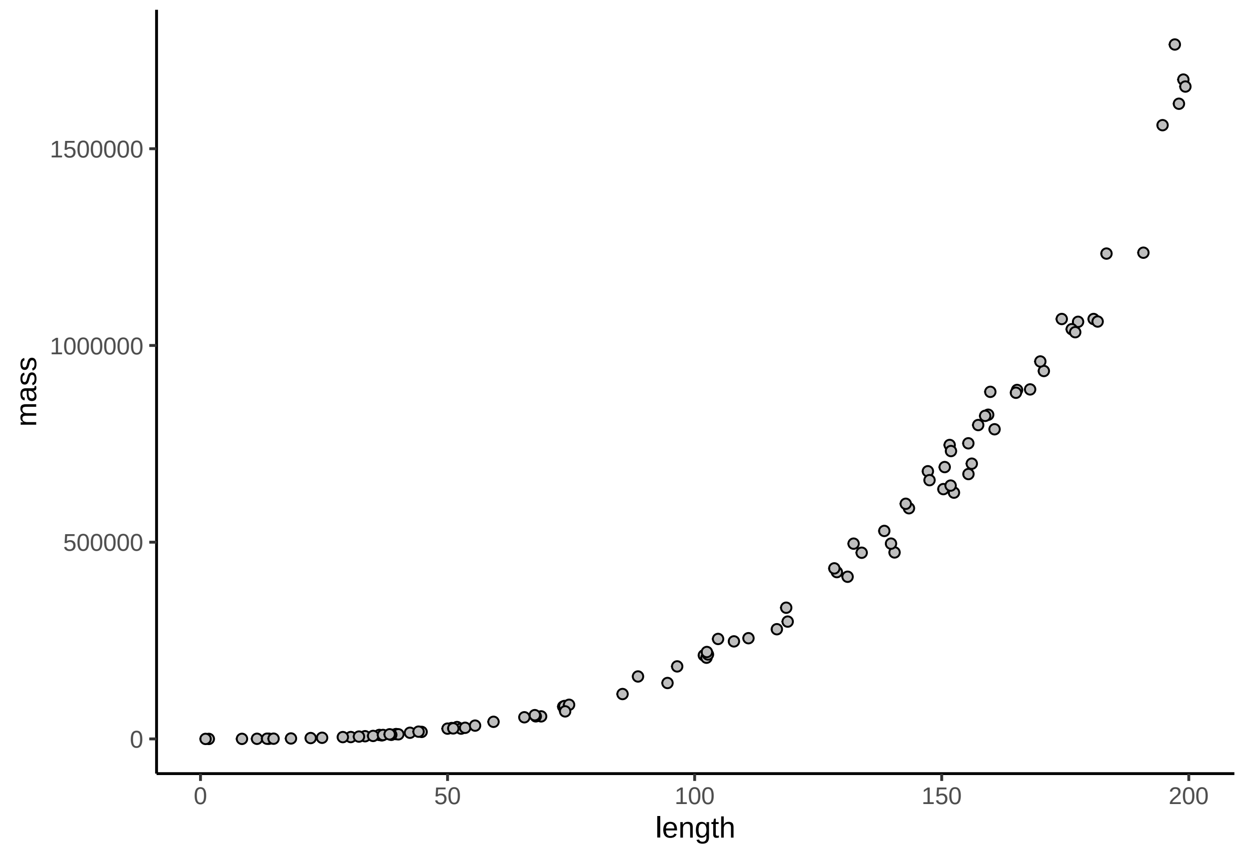
```r
  scale_y_log10()+
  annotation_logticks()
```



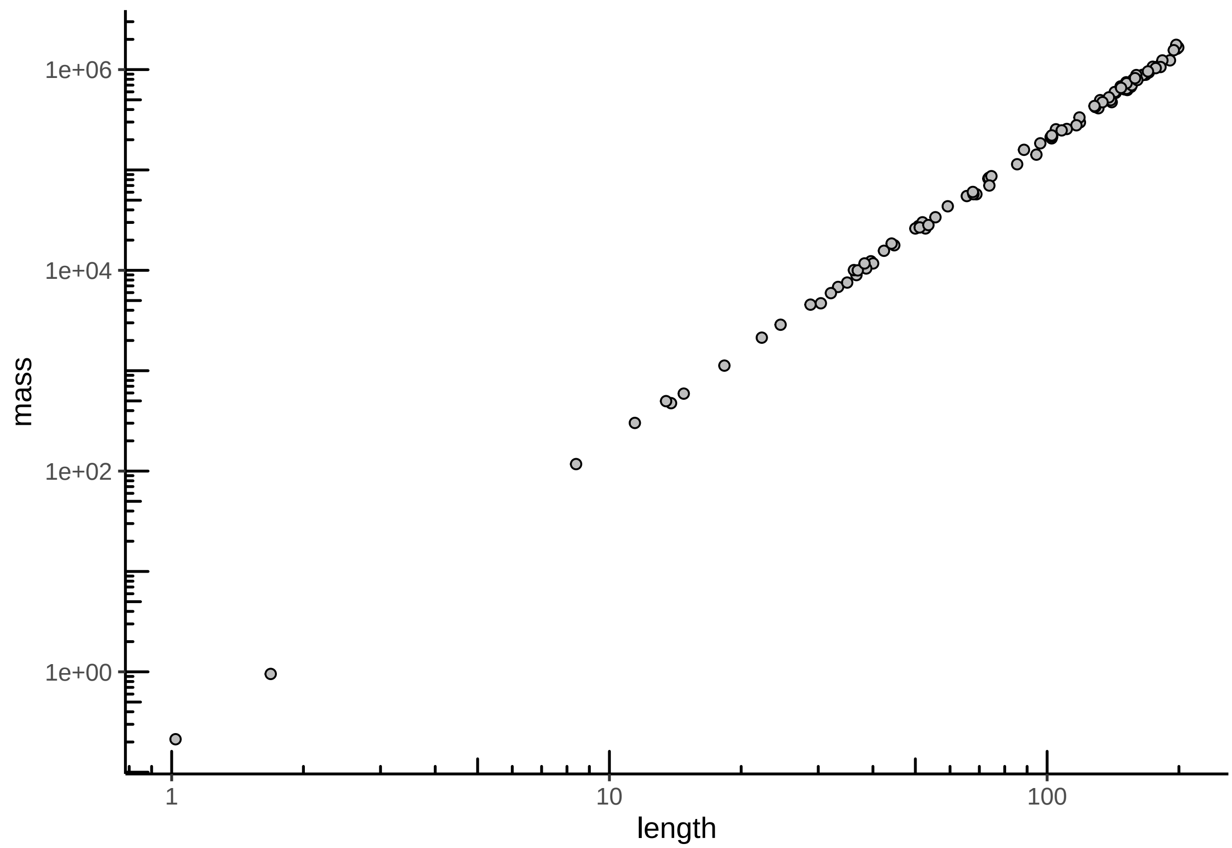

```r
#make a model
cylinders.lm <- lm(log(cylinders.df$mass)~(log(cylinders.df$length)))
summary(cylinders.lm)
```

```
##
## Call:
## lm(formula = log(cylinders.df$mass) ~ (log(cylinders.df$length)))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.222779 -0.083773 -0.000243  0.088552  0.243567
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -3.05556    0.04716  -64.79   <2e-16 ***
## log(cylinders.df$length)  3.00595    0.01059  283.82   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1009 on 98 degrees of freedom
## Multiple R-squared:  0.9988, Adjusted R-squared:  0.9988
## F-statistic: 8.055e+04 on 1 and 98 DF,  p-value: < 2.2e-16
```

```
#thin fish
cylinders.df <- cylinders.df %>%
  mutate(width=length*width_factor,
         height=length,
         volume=pi*(height/2)*(width/2)*length,
         mass=volume*density)

#make base plot
plot <- ggplot(cylinders.df, aes(length, mass))+
  theme+
  geom_point(shape=21, fill="grey")

#print the plot
plot
```




```
#make it log
plot +
  scale_x_log10()+
  scale_y_log10()+
  annotation_logticks()
```

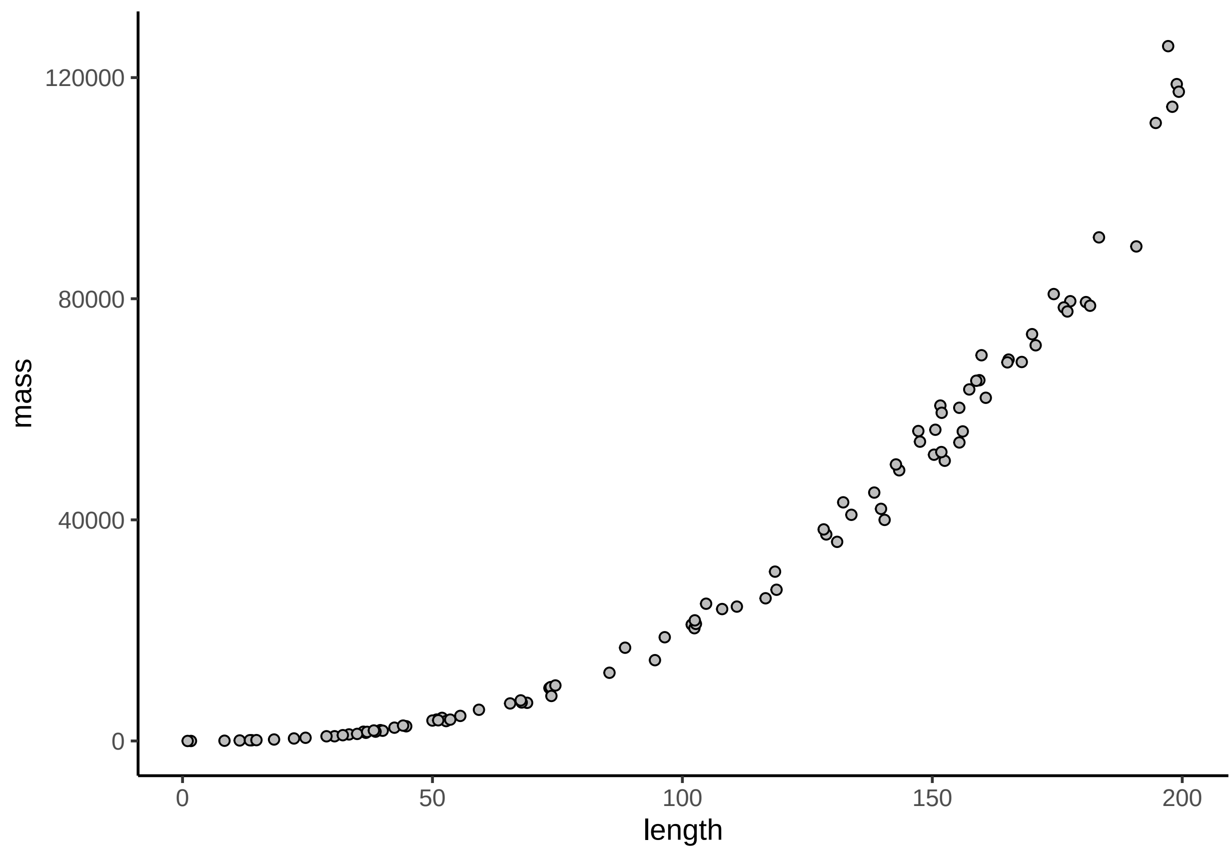



```
#make a model
cylinders.lm <- lm(log(cylinders.df$mass)~(log(cylinders.df$length)))
summary(cylinders.lm)
```

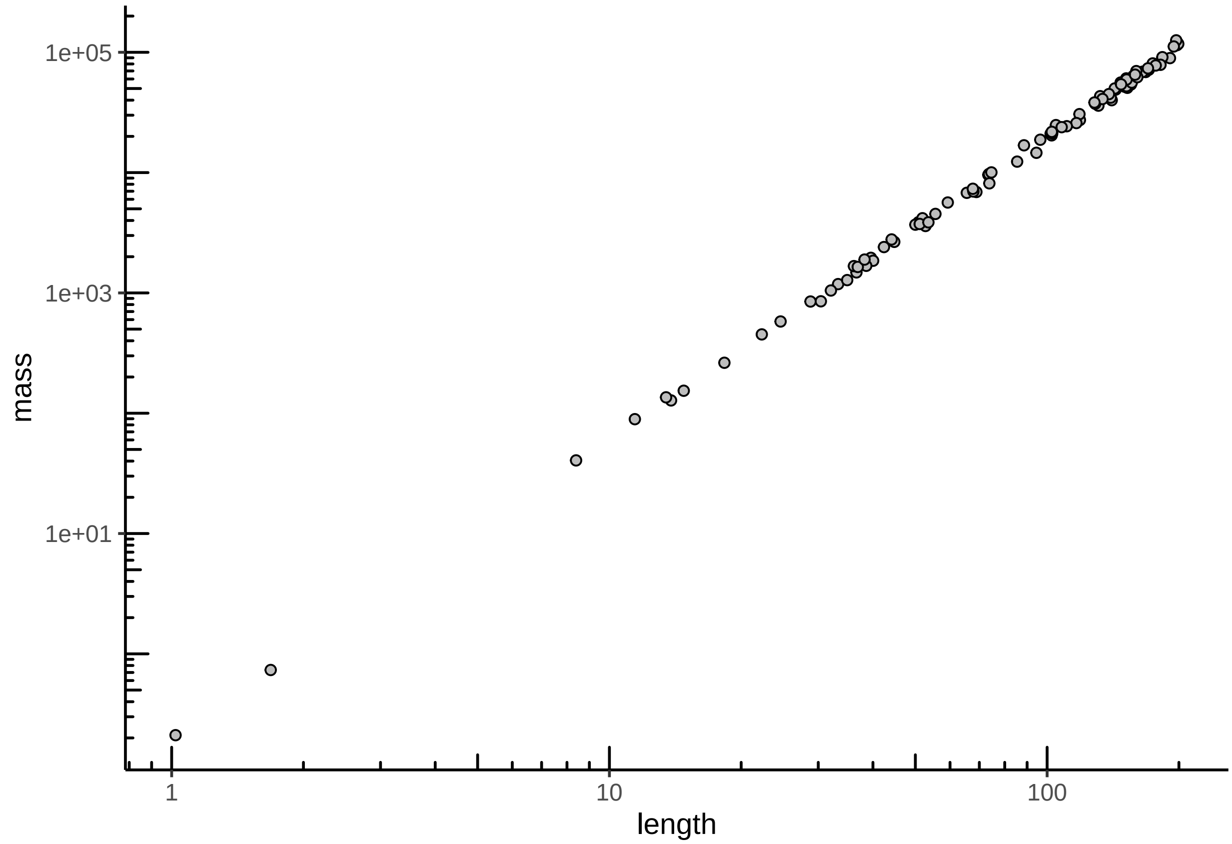
```
##
## Call:
## lm(formula = log(cylinders.df$mass) ~ (log(cylinders.df$length)))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.15170 -0.05839  0.01184  0.04832  0.15979
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -1.655497   0.032322  -51.22   <2e-16 ***
## log(cylinders.df$length)  3.005108   0.007258  414.03   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06911 on 98 degrees of freedom
## Multiple R-squared:  0.9994, Adjusted R-squared:  0.9994
## F-statistic: 1.714e+05 on 1 and 98 DF,  p-value: < 2.2e-16
```

Challenge scenario 2

```r
#nonlinear
cylinders.df <- cylinders.df %>%
  mutate(height=sqrt(length),
         volume=pi*(height/2)*(width/2)*length,
         mass=volume*density)

#make base plot
plot <- ggplot(cylinders.df, aes(length, mass))+
  theme+
  geom_point(shape=21, fill="grey")

#print the plot
plot
```




```r
#make it log
plot +
  scale_x_log10()+
  scale_y_log10()+
  annotation_logticks()
```



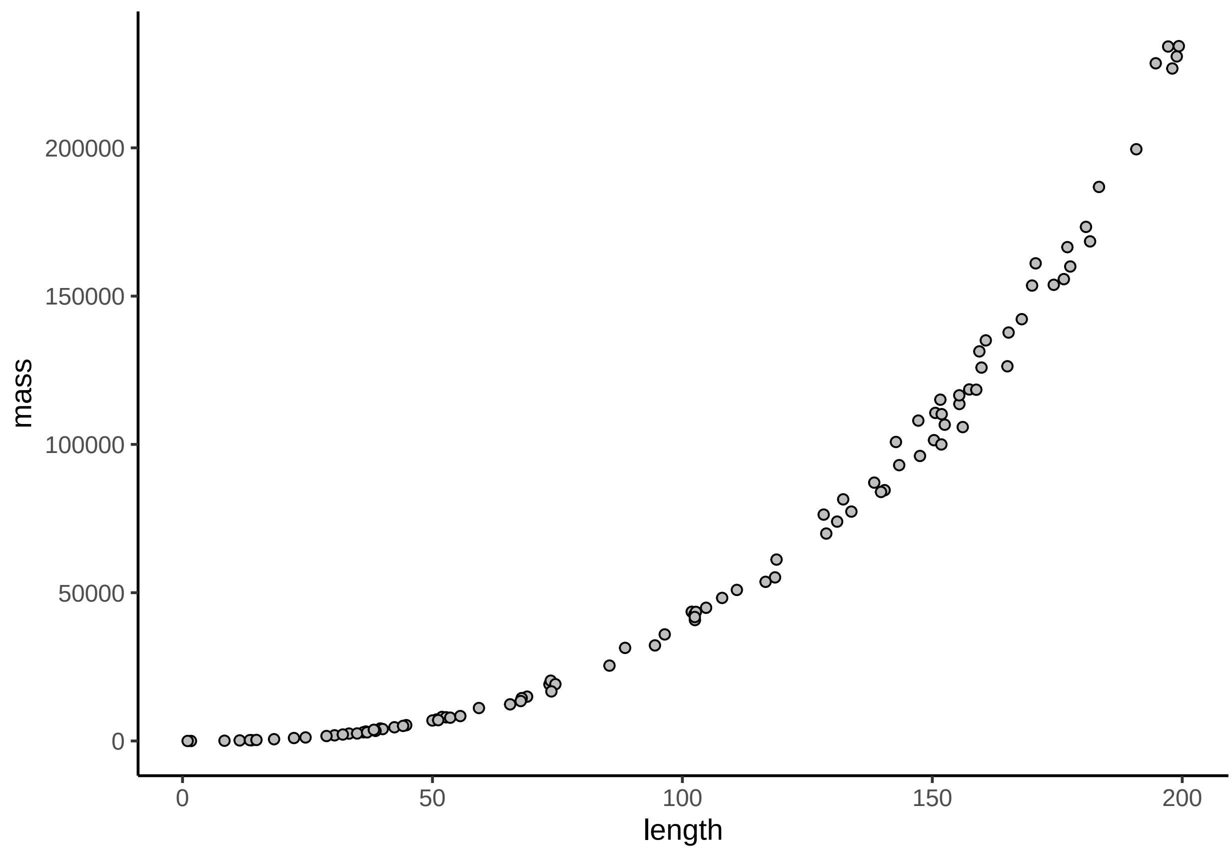

```r
#make a model
cylinders.lm <- lm(log(cylinders.df$mass)~(log(cylinders.df$length)))
summary(cylinders.lm)
```

```
##
## Call:
## lm(formula = log(cylinders.df$mass) ~ (log(cylinders.df$length)))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.15170 -0.05839  0.01184  0.04832  0.15979
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -1.655497   0.032322  -51.22   <2e-16 ***
## log(cylinders.df$length)  2.505108   0.007258  345.14   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06911 on 98 degrees of freedom
## Multiple R-squared:  0.9992, Adjusted R-squared:  0.9992
## F-statistic: 1.191e+05 on 1 and 98 DF,  p-value: < 2.2e-16
```

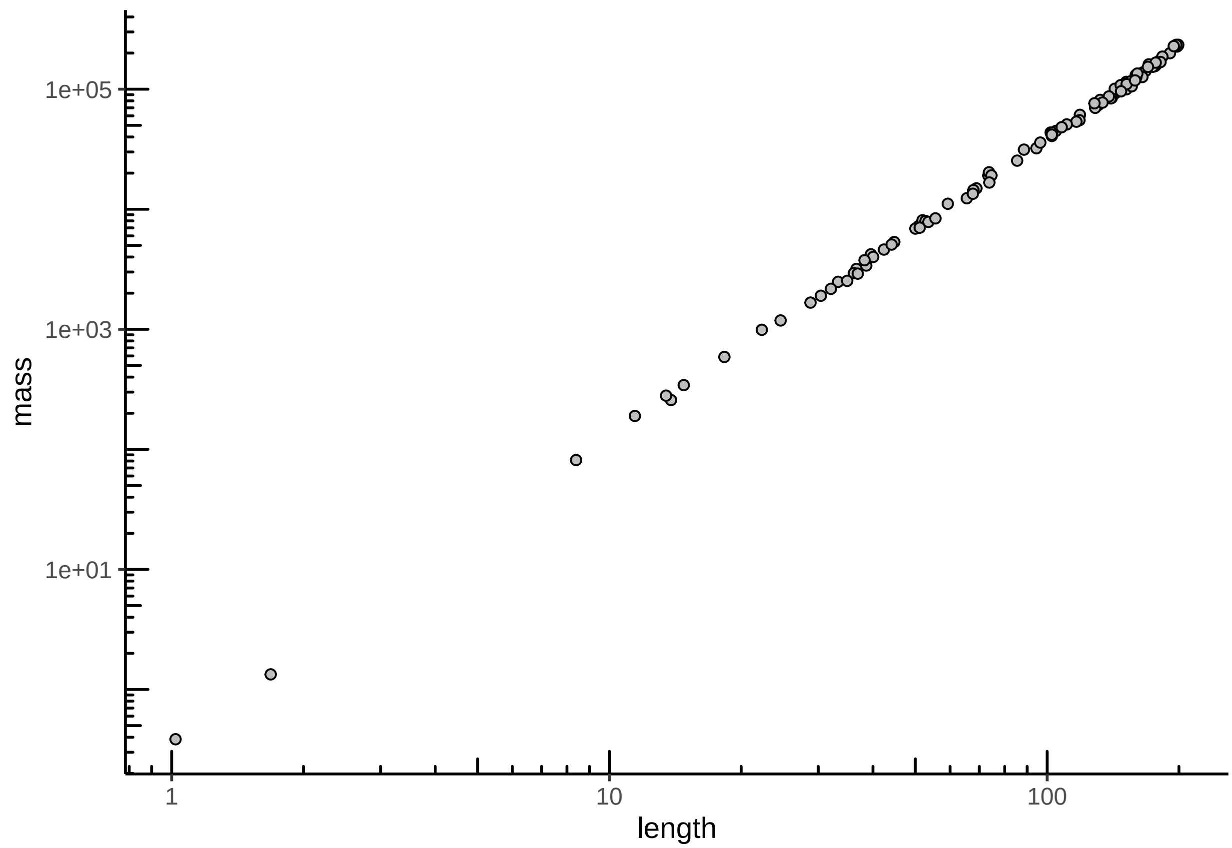
The slope of the model (log(A)) has changed, but the intercept remains the same.

Challenge scenario 3

```r
#change density from 1 to 2
cylinders.df$density <- runif(n=100, min=1.9, max=2.1)

cylinders.df <- cylinders.df %>%
  mutate(mass=volume*density)

#make base plot
plot <- ggplot(cylinders.df, aes(length, mass))+
  theme+
  geom_point(shape=21, fill="grey")

#print the plot
plot
```




```r
#make it log
plot +
  scale_x_log10()+
  scale_y_log10()+
  annotation_logticks()
```




```r
#make a model
cylinders.lm <- lm(log(cylinders.df$mass)~(log(cylinders.df$length)))
summary(cylinders.lm)
```

```
##
## Call:
## lm(formula = log(cylinders.df$mass) ~ (log(cylinders.df$length)))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.116338 -0.039578  0.006326  0.044150  0.114403
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -0.96312    0.02583  -37.29   <2e-16 ***
## log(cylinders.df$length)  2.50458    0.00580  431.84   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05523 on 98 degrees of freedom
## Multiple R-squared:  0.9995, Adjusted R-squared:  0.9995
## F-statistic: 1.865e+05 on 1 and 98 DF,  p-value: < 2.2e-16
```

I increased the density from 1 to 2. The intercept of the model has changed, but the slope of log(A) has remained the same. This suggests that the proportionality of the relationship between length and weight has not changed, but fishes have become systematically heavier.

QUESTION: Pay attention to the slopes of the models fitted with lm() – i.e., the slope of the line in the log(length)-log(weight) plots. What do these slopes tell you about how length and weight are related?

The slope of the first population is 1, compared to 3.01 for the second population. In the first population, weight exhibits a 1:1 relationship with length (i.e., a fish that is twice the length is also twice the weight) while in the second population, weight is 3*log(length) (i.e., a fish that is twice as long will have more than twice the weight, although the change in weight in non-linear.)

QUESTION: Based on this geometric example, make a prediction about what you'd expect to see (in the plots and/or coefficients) if we do this analysis on a population of real fish that grow in length faster than they grow in cross-section. What about a population that grows proportionally?

A population of fish that grows faster in length will look more similar to cylinder population 1 than cylinder population 2. The coefficient of log(1A) will be closer to 1.

Population 2 meets the definition of a population of fish that grows proportionally. Thus, the plots will look similar to population 2 and the coefficient of log(1A) will be approximately 3. Indeed, in the below example we find that the coefficient of log(1A) for most fish is around 3.

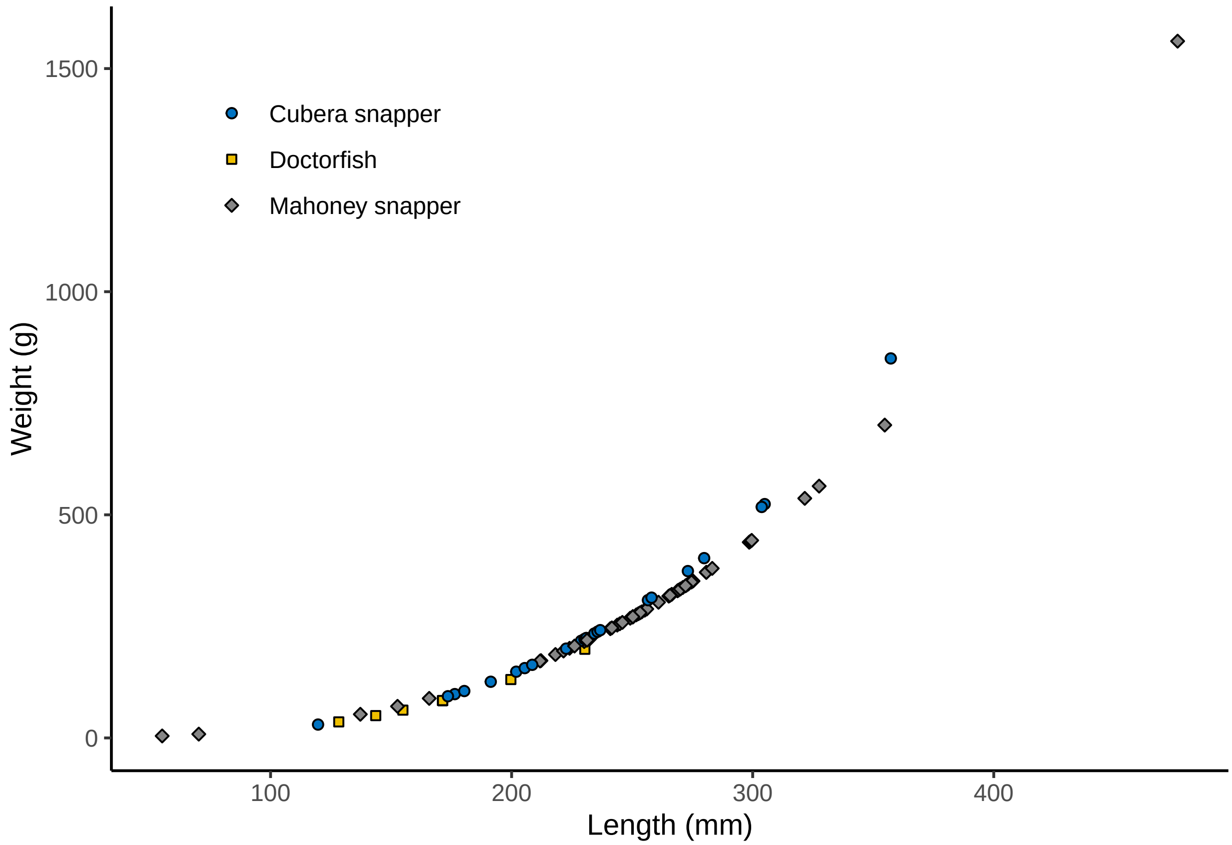
Where do real fish fit in?

```
#use the here package as an alternative to setwd()
df <- read.table(here("raw/labs/Lab 0 - pone.0156641.s002.csv"),
                 sep=",", header=TRUE)

str(df)
```

```
## 'data.frame':    9379 obs. of  9 variables:
##  $ site     : chr  "BCW" "BCW" "BCW" "BCW" ...
##  $ zone     : int  5 5 5 5 5 5 5 5 5 5 ...
##  $ transect : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ name     : chr  "Carangidae_Caranx_ruber" "Labridae_Thalassoma_bifasciatum" "Labridae_Thalassoma_
##  $ family   : chr  "Carangidae" "Labridae" "Labridae" "Labridae" ...
##  $ genus    : chr  "Caranx" "Thalassoma" "Thalassoma" "Thalassoma" ...
##  $ species  : chr  "ruber" "bifasciatum" "bifasciatum" "bifasciatum" ...
##  $ length.mm: num  170.7 23.1 29.3 29.4 33 ...
##  $ weight.g : num  87.037 0.141 0.292 0.294 0.418 ...
```

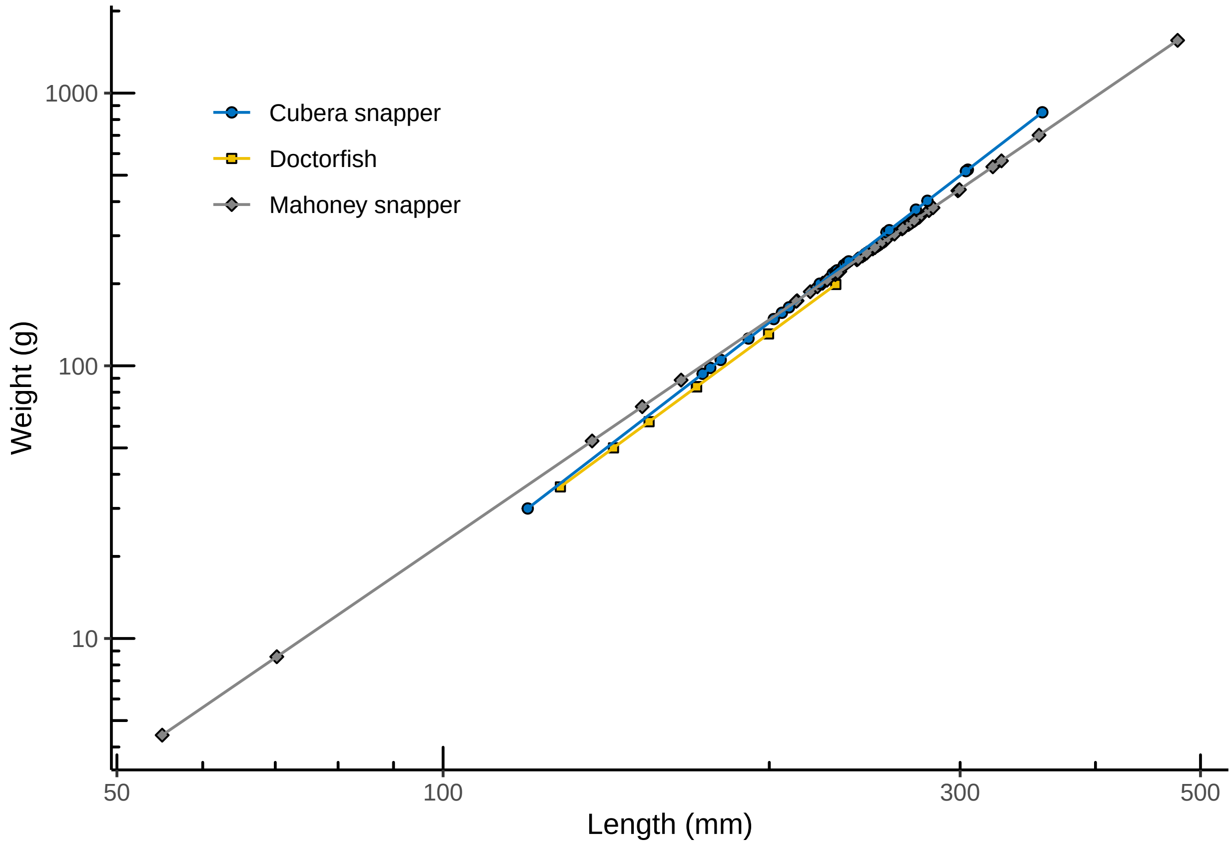
```
# subset the data to look at one species
# do the same length vs weight analysis as you did for the simulated cylinder data. You should show a p

# analyze Cubera snapper
subset.df <- df %>%
  filter(name %in% c("Lutjanidae_Lutjanus_cyanopterus",
                     "Acanthuridae_Acanthurus_chirurgus",
                     "Lutjanidae_Lutjanus_mahogoni")) %>%
  mutate(common_name = factor(name,
                              levels=c("Lutjanidae_Lutjanus_cyanopterus",
                                       "Acanthuridae_Acanthurus_chirurgus",
                                       "Lutjanidae_Lutjanus_mahogoni"),
                              labels=c("Cubera snapper",
```

```
                                        "Doctorfish",
                                        "Mahoney snapper")))

#make base plot
plot <- ggplot(subset.df, aes(length.mm, weight.g,
                              fill=common_name,
                              shape=common_name))+
  theme+
  geom_point()+
  labs(x="Length (mm)",
       y="Weight (g)")

plot
```




```
plot +
  scale_x_log10()+
  scale_y_log10()+
  annotation_logticks()+
  geom_smooth(aes(color=common_name), method="lm", se=FALSE, linewidth=0.5)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
#make a model for species 1
lm1 <- lm(log(weight.g)~(log(length.mm)),
          data=subset(subset.df, common_name=="Cubera snapper"))
summary(lm1)
```

```
##
## Call:
## lm(formula = log(weight.g) ~ (log(length.mm)), data = subset(subset.df,
##     common_name == "Cubera snapper"))
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -3.623e-10 -3.562e-11  2.341e-11  6.689e-11  2.779e-10
##
## Coefficients:
##                  Estimate Std. Error    t value Pr(>|t|)
## (Intercept)    -1.124e+01  7.135e-10 -1.575e+10   <2e-16 ***
## log(length.mm)  3.059e+00  1.315e-10  2.327e+10   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.41e-10 on 20 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 5.415e+20 on 1 and 20 DF,  p-value: < 2.2e-16
```

```r
#make a model for species 2
lm2 <- lm(log(weight.g)~(log(length.mm)),
         data=subset(subset.df, common_name=="Doctorfish"))
summary(lm2)
```

```
##
## Call:
## lm(formula = log(weight.g) ~ (log(length.mm)), data = subset(subset.df,
##     common_name == "Doctorfish"))
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -4.501e-11 -4.501e-11 -2.428e-11  4.220e-11  1.107e-10
##
## Coefficients:
##                   Estimate Std. Error    t value Pr(>|t|)
## (Intercept)    -1.059e+01  6.298e-10 -1.682e+10   <2e-16 ***
## log(length.mm)  2.920e+00  1.227e-10  2.381e+10   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.942e-11 on 8 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 5.667e+20 on 1 and 8 DF,  p-value: < 2.2e-16
```

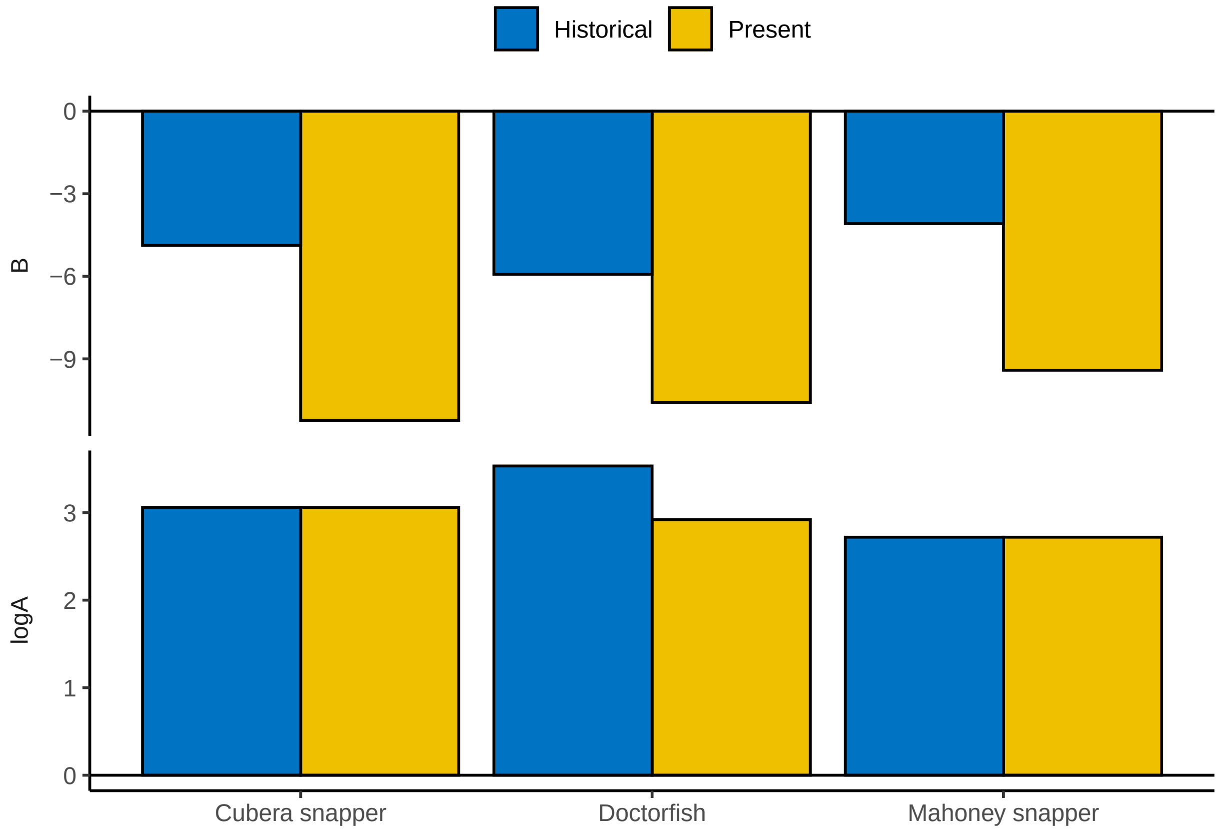
```r
#make a model for species 2
lm3 <- lm(log(weight.g)~(log(length.mm)),
         data=subset(subset.df, common_name=="Mahoney snapper"))
summary(lm3)
```

```
##
## Call:
## lm(formula = log(weight.g) ~ (log(length.mm)), data = subset(subset.df,
##     common_name == "Mahoney snapper"))
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -2.194e-10 -8.382e-11 -4.133e-11  9.926e-11  2.647e-10
##
## Coefficients:
##                   Estimate Std. Error    t value Pr(>|t|)
## (Intercept)    -9.412e+00  2.704e-10 -3.481e+10   <2e-16 ***
## log(length.mm)  2.719e+00  4.920e-11  5.526e+10   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.168e-10 on 58 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 3.054e+21 on 1 and 58 DF,  p-value: < 2.2e-16
```

QUESTION: How well do your estimates match? Ask around the class. Are the errors systematic? Are the "eating fish" getting chunkier?

As shown in the figure below, my estimates for log(A) are fairly accurate but my estimates for B are systematically more negative. This suggests that fish were historically more chonky than they are today.

```
common_name <- c("Cubera snapper", "Doctorfish", "Mahoney snapper")

logA <- c(coefficients(lm1)[2],
          coefficients(lm2)[2],
          coefficients(lm3)[2])

B <- c(coefficients(lm1)[1],
       coefficients(lm2)[1],
       coefficients(lm3)[1])

observed.df <- data.frame(common_name=common_name,
                          logA=logA,
                          B=B) %>%
  mutate(type="Present")

actual.df <- data.frame(common_name=common_name,
                        logA=c(3.0601, 3.5328, 2.7190),
                        B=c(-4.8799, -5.9255, -4.0870))%>%
  mutate(type="Historical")

df <- bind_rows(observed.df, actual.df) %>%
  pivot_longer(cols=c("logA", "B"))

ggplot(df, aes(common_name, value, fill=type))+
  theme+
  geom_col(color="black", position="dodge")+
  facet_wrap(.~name, ncol=1, scales="free_y", strip.position = "left")+
  geom_hline(yintercept=0)+
  labs(x=NULL, y=NULL)+
  theme(strip.placement="outside",
        strip.background=element_blank(),
        legend.position="top")
```



QUESTION: "All models are wrong, some models are useful." We used two different types of models today. 1) We used geometric models (cylinders) to approximate the body shape of fish. 2) We used a linear model (the lm() function in R) to describe the relationship between log(length) and log(weight). In what ways were each of these models wrong (i.e., what did they leave out)? Were they useful?

The geometric models simplified the actual shape of the fish: for a given length, width, and height the actual volume of the fish is not the same as the volume of a cylinder. However, fish are likely more similar to a cylinder than a cube or a sphere, and perhaps the mass of the fish calculated from the estimated volume is useful.

Similarly, the linear models predicted fish weight from length but did not account for variations in the "chunkiness" of individuals within a population or variations in density. Length is undoubtedly a useful predictor of fish weight - the R squared of all three models is approximately 1 - but any variations in fish weight that are not related to length are not captured by the model.