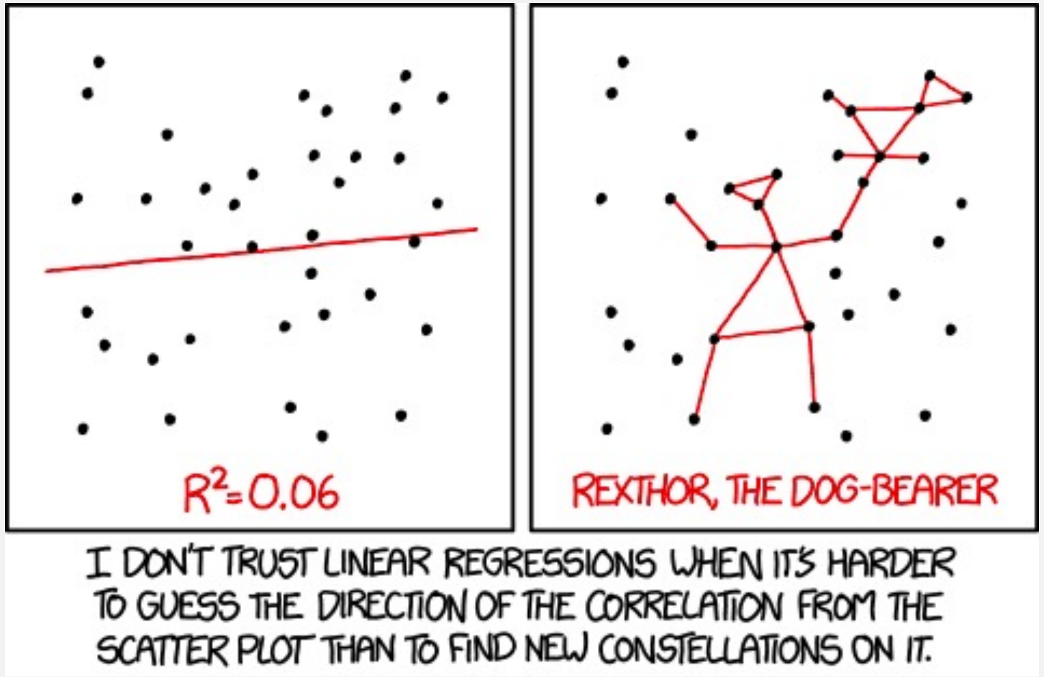
Plotting and Models

Biology 683

Heath Blackmon


R^2=0.06
REXTHOR, THE DOG-BEARER
I DON'T TRUST LINEAR REGRESSIONS WHEN IT'S HARDER TO GUESS THE DIRECTION OF THE CORRELATION FROM THE SCATTER PLOT THAN TO FIND NEW CONSTELLATIONS ON IT.

"Remember that all models are wrong; the practical question is how wrong do they have to be to not be useful."

George Box

Plotting in R

R has always had some plotting capabilities. However, the number of packages that are designed to produce data visualizations has grown dramatically over the last 15 years. Today the plotting landscape is dominated by two largely incompatible ecosystems one in base R and one integrated with the package ggplot2. I use both in my own work.

Base R	**ggplot2**
Shallow learning curve	Steep learning curve
More freedom to do anything you want to do	Many good decisions are default behavior

ggplot2 (data)

wide data

time 1	time 2
1.202	1.45
1.301	1.271
0.987	0.654
2.013	2.458
1.750	1.989

long data

Rate	Time
1.202	1
1.301	1
0.987	1
2.013	1
1.750	1
1.45	2
1.271	2
0.654	2
2.458	2
1.989	2

ggplot2 (grammar)

Heath made the cool plot.

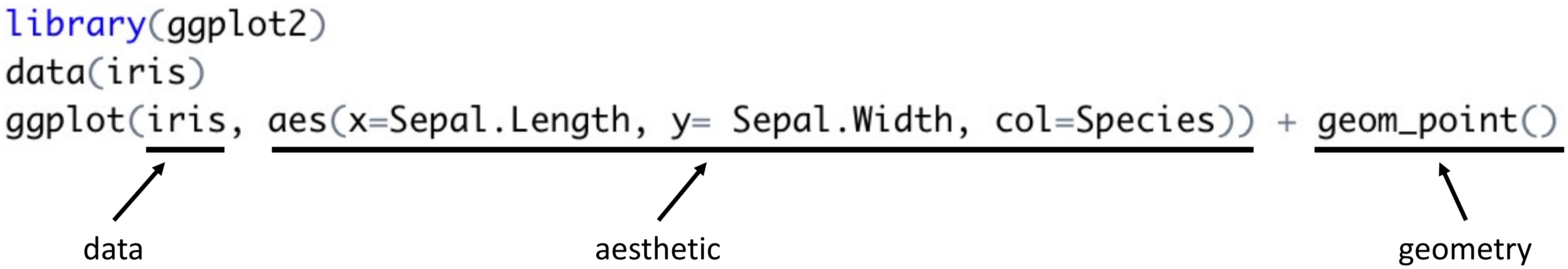
Noun	Heath	Heath	Heath
Verb	made	made	fixed
Article	the	the	the
Adjective	cool	horrible	horrible
Noun	plot	plot	plot

ggplot2 (grammar)

Grammatical elements in ggplot2

Element	Description
data	The data being plotted
aesthetics	The scales onto which we plot our data
geometries	The visual elements used for our data
facets	Splitting plots into multiples based on a variable
statistics	Ways of summarizing data
coordinates	The space on which data will be plotted
themes	Aspects unrelated to the data

ggplot2 (simple example)



```
library(ggplot2)
data(iris)
ggplot(iris, aes(x=Sepal.Length, y= Sepal.Width, col=Species)) + geom_point()
```

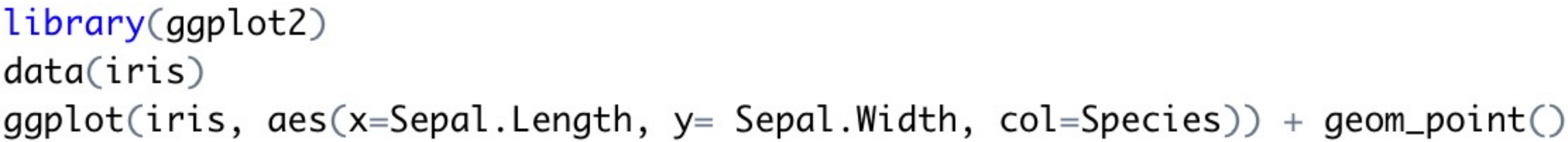
data

aesthetic

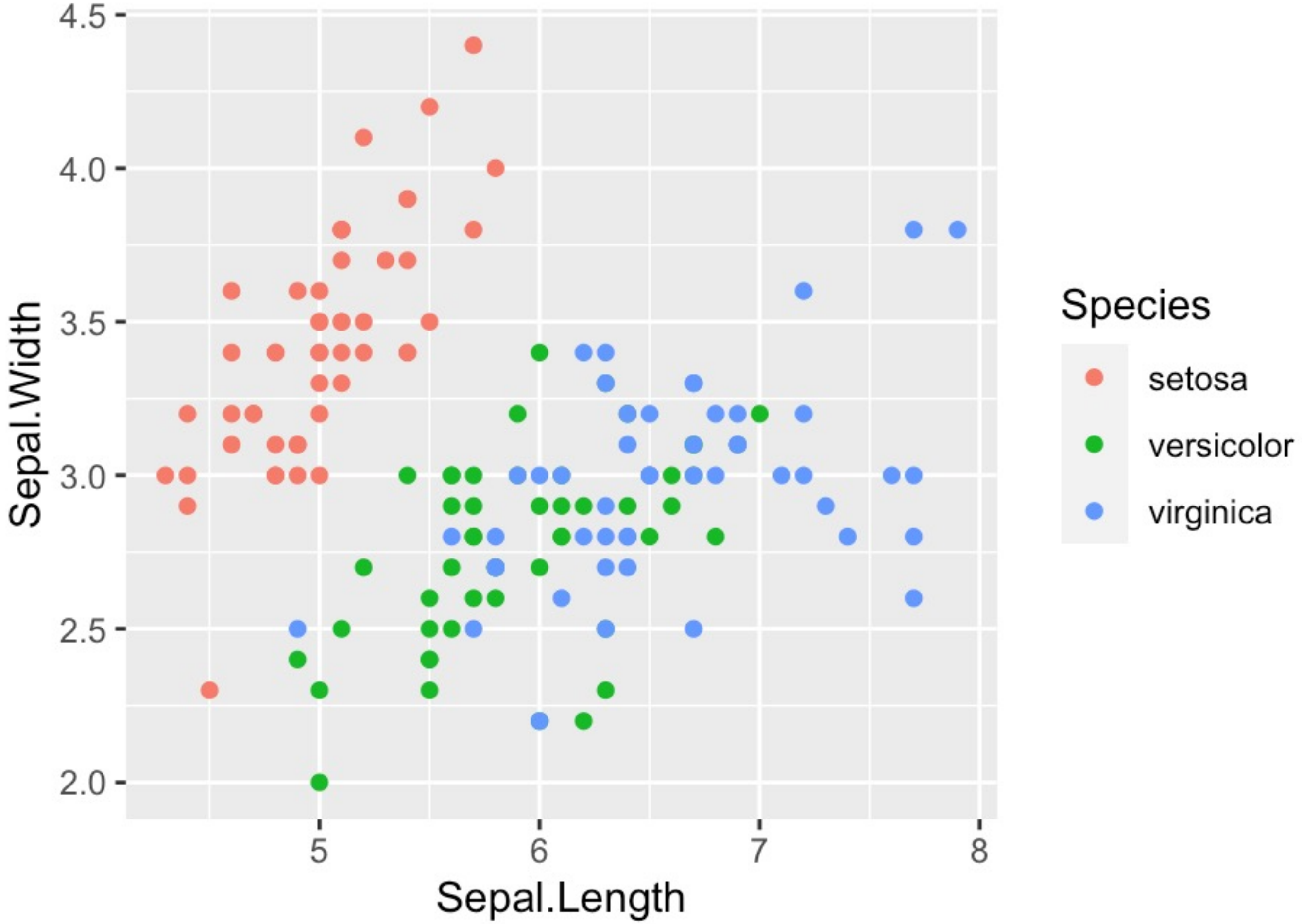
geometry

In this case I wanted an XY scatter plot so these aesthetics make sense. Depending on the geometry you will you will use other things may make more or less sense to include. Some common options include: x, y, fill, col, shape, size.

ggplot2 (simple example)



```
library(ggplot2)
data(iris)
ggplot(iris, aes(x=Sepal.Length, y= Sepal.Width, col=Species)) + geom_point()
```

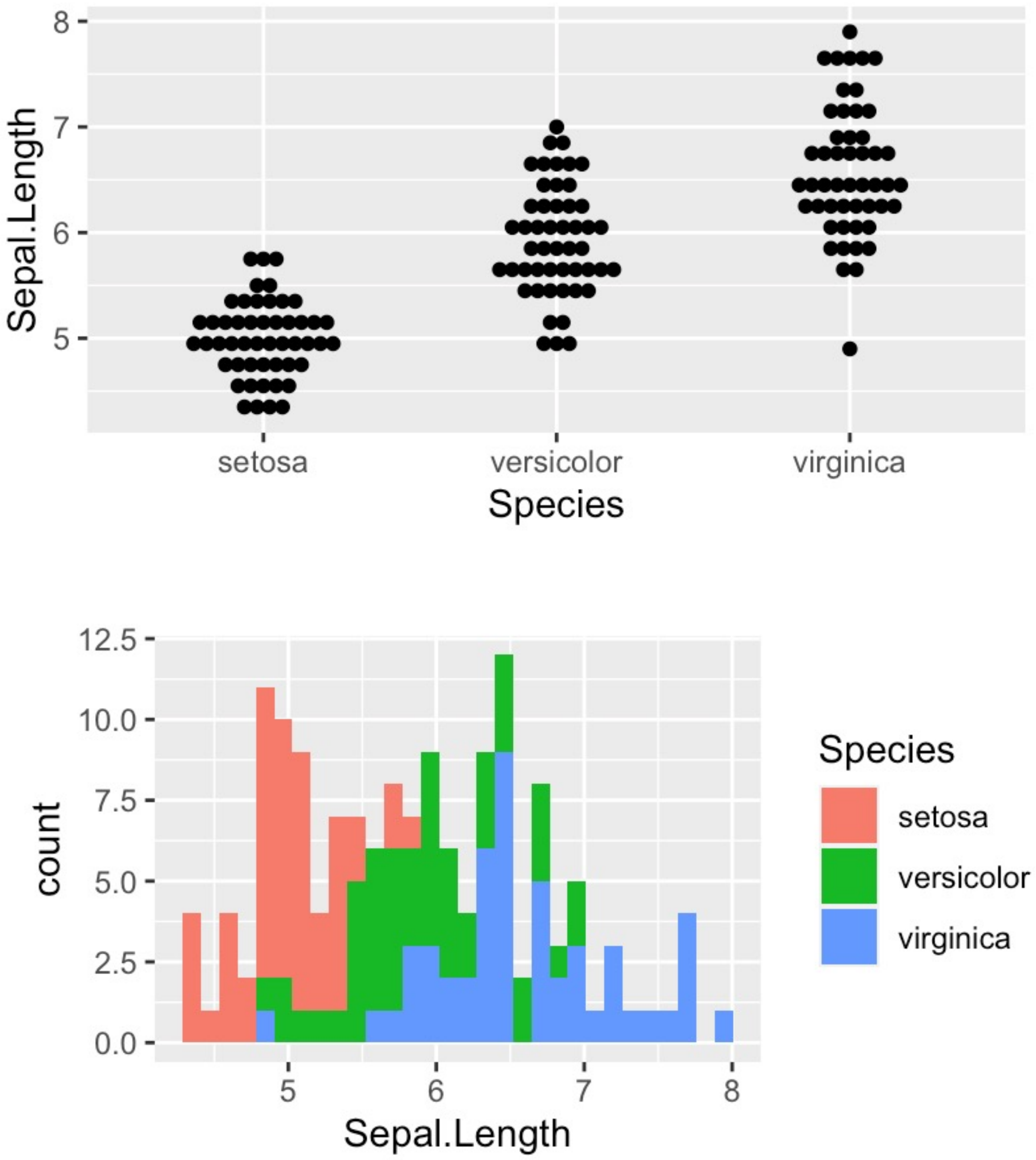


ggplot2 (simple example)

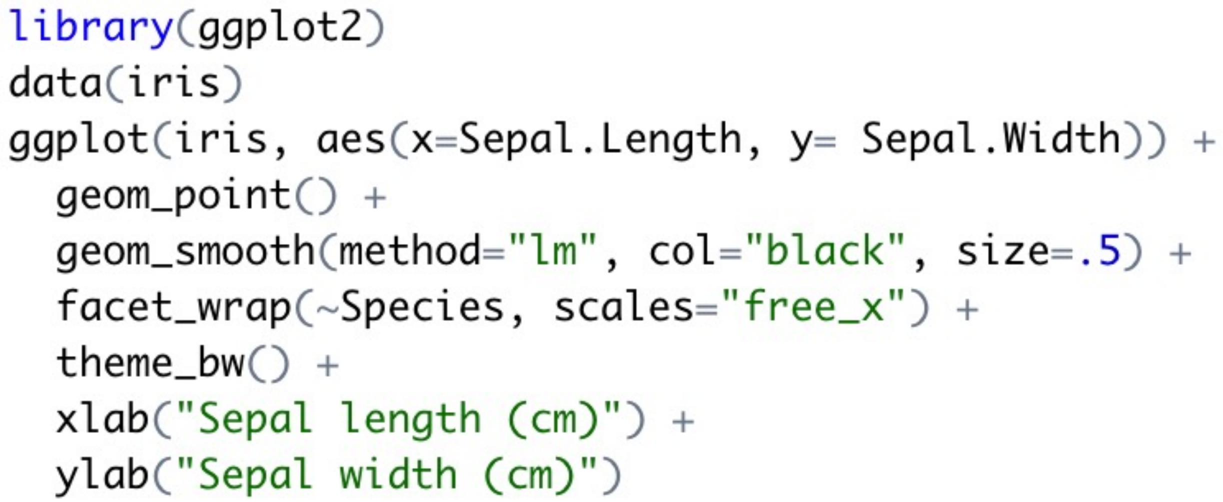
```r
a <- ggplot(iris, aes(x=Species, y=Sepal.Length)) +
      geom_dotplot(binaxis = "y", stackdir = "center")
```

```r
b <- ggplot(iris, aes(x=Sepal.Length, fill=Species)) +
      geom_histogram(position="dodge")
```

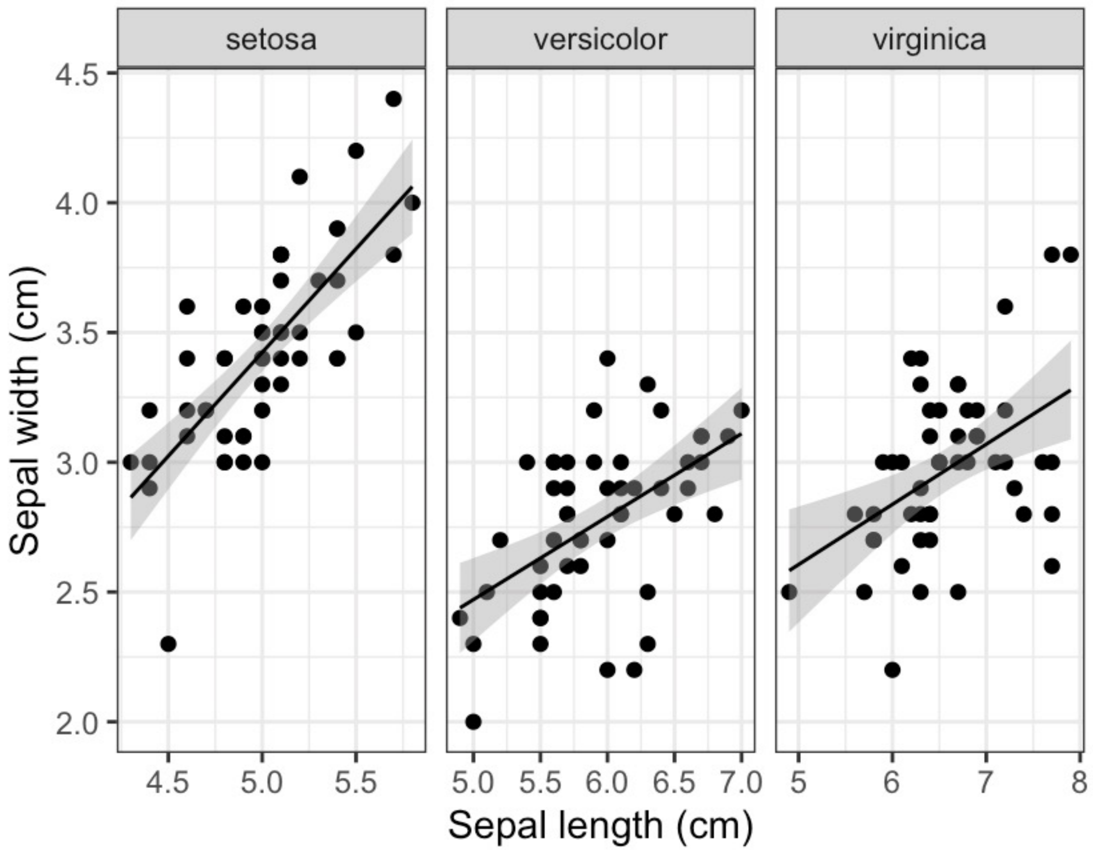
```r
library(gridExtra)
grid.arrange(a,b)
```



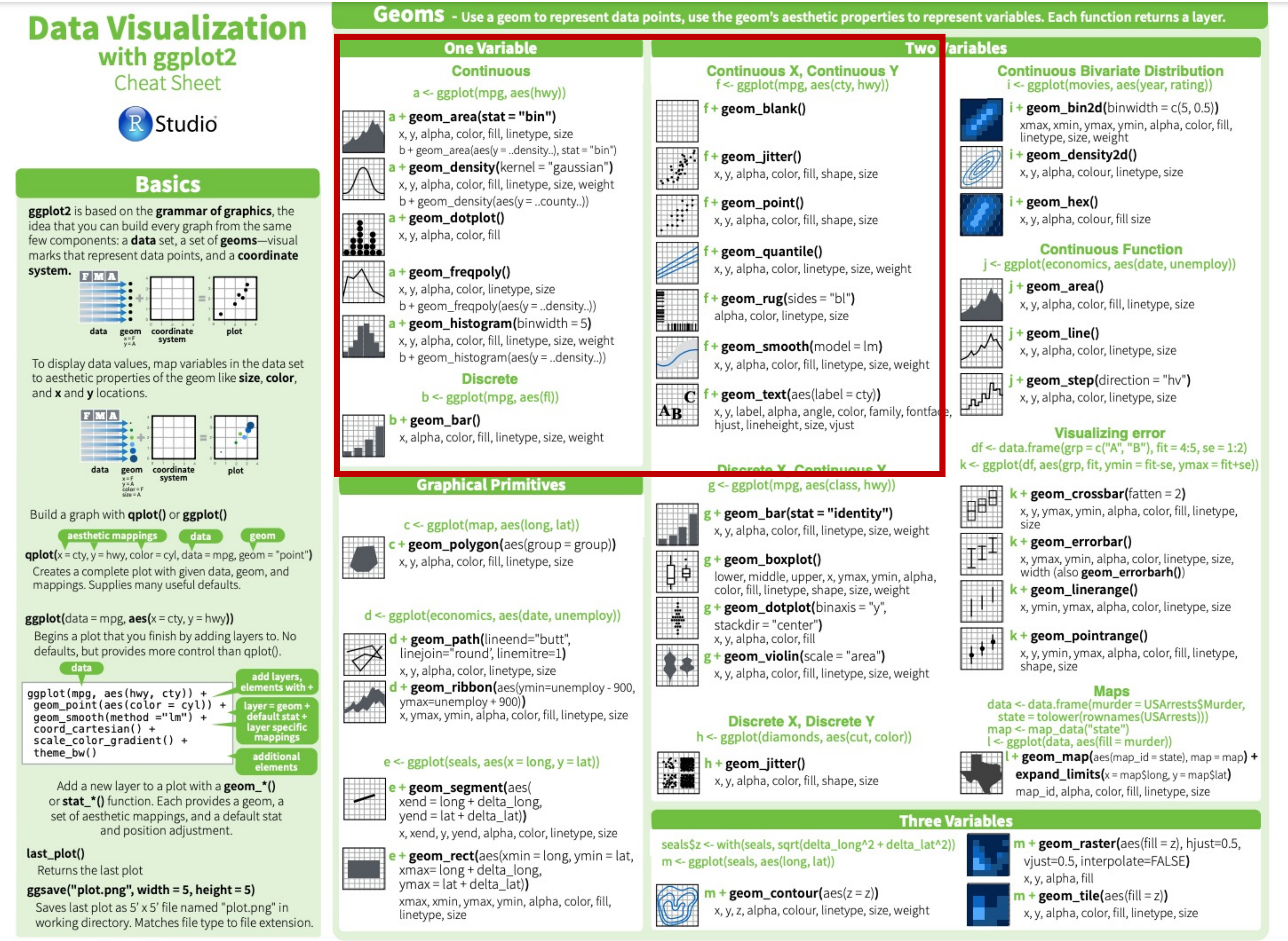
ggplot2 (nicer example)



```
library(ggplot2)
data(iris)
ggplot(iris, aes(x=Sepal.Length, y= Sepal.Width)) +
  geom_point() +
  geom_smooth(method="lm", col="black", size=.5) +
  facet_wrap(~Species, scales="free_x") +
  theme_bw() +
  xlab("Sepal length (cm)") +
  ylab("Sepal width (cm)")
```



ggplot2 (cheat sheet)



Data Visualization with ggplot2 Cheat Sheet

RStudio

Basics

ggplot2 is based on the **grammar of graphics**, the idea that you can build every graph from the same few components: a **data** set, a set of **geoms**—visual marks that represent data points, and a **coordinate system.**

F M A
data
geom
x = F
y = A
coordinate system
plot

To display data values, map variables in the data set to aesthetic properties of the geom like **size**, **color**, and **x** and **y** locations.

Build a graph with **qplot()** or **ggplot()**

qplot(x = cty, y = hwy, color = cyl, data = mpg, geom = "point"**)**
Creates a complete plot with given data, geom, and mappings. Supplies many useful defaults.

ggplot(data = mpg, **aes(**x = cty, y = hwy**))**
Begins a plot that you finish by adding layers to. No defaults, but provides more control than qplot().

```
ggplot(mpg, aes(hwy, cty)) +
 geom_point(aes(color = cyl)) +
 geom_smooth(method ="lm") +
 coord_cartesian() +
 scale_color_gradient() +
 theme_bw()
```

add layers, elements with +
layer = geom + default stat + layer specific mappings
additional elements

Add a new layer to a plot with a **geom_*()** or **stat_*()** function. Each provides a geom, a set of aesthetic mappings, and a default stat and position adjustment.

last_plot()
Returns the last plot

ggsave("plot.png", width = 5, height = 5)
Saves last plot as 5' x 5' file named "plot.png" in working directory. Matches file type to file extension.

Geoms - Use a geom to represent data points, use the geom's aesthetic properties to represent variables. Each function returns a layer.

One Variable

Continuous

a <- ggplot(mpg, aes(hwy))

a + **geom_area(stat = "bin")**
x, y, alpha, color, fill, linetype, size
b + geom_area(aes(y = ..density..), stat = "bin")

a + **geom_density(**kernel = "gaussian"**)**
x, y, alpha, color, fill, linetype, size, weight
b + geom_density(aes(y = ..county..))

a + **geom_dotplot()**
x, y, alpha, color, fill

a + **geom_freqpoly()**
x, y, alpha, color, linetype, size
b + geom_freqpoly(aes(y = ..density..))

a + **geom_histogram(**binwidth = 5**)**
x, y, alpha, color, fill, linetype, size, weight
b + geom_histogram(aes(y = ..density..))

Discrete

b <- ggplot(mpg, aes(fl))

b + **geom_bar()**
x, alpha, color, fill, linetype, size, weight

Graphical Primitives

c <- ggplot(map, aes(long, lat))

c + **geom_polygon(**aes(group = group)**)**
x, y, alpha, color, fill, linetype, size

d <- ggplot(economics, aes(date, unemploy))

d + **geom_path(**lineend="butt", linejoin="round', linemitre=1**)**
x, y, alpha, color, linetype, size

d + **geom_ribbon(**aes(ymin=unemploy - 900, ymax=unemploy + 900)**)**
x, ymax, ymin, alpha, color, fill, linetype, size

e <- ggplot(seals, aes(x = long, y = lat))

e + **geom_segment(**aes(xend = long + delta_long, yend = lat + delta_lat)**)**
x, xend, y, yend, alpha, color, linetype, size

e + **geom_rect(**aes(xmin = long, ymin = lat, xmax= long + delta_long, ymax = lat + delta_lat)**)**
xmax, xmin, ymax, ymin, alpha, color, fill, linetype, size

Two Variables

Continuous X, Continuous Y

f <- ggplot(mpg, aes(cty, hwy))

f + **geom_blank()**

f + **geom_jitter()**
x, y, alpha, color, fill, shape, size

f + **geom_point()**
x, y, alpha, color, fill, shape, size

f + **geom_quantile()**
x, y, alpha, color, linetype, size, weight

f + **geom_rug(**sides = "bl"**)**
alpha, color, linetype, size

f + **geom_smooth(**model = lm**)**
x, y, alpha, color, fill, linetype, size, weight

f + **geom_text(**aes(label = cty)**)**
x, y, label, alpha, angle, color, family, fontface, hjust, lineheight, size, vjust

Discrete X, Continuous Y

g <- ggplot(mpg, aes(class, hwy))

g + **geom_bar(stat = "identity")**
x, y, alpha, color, fill, linetype, size, weight

g + **geom_boxplot()**
lower, middle, upper, x, ymax, ymin, alpha, color, fill, linetype, shape, size, weight

g + **geom_dotplot(**binaxis = "y", stackdir = "center"**)**
x, y, alpha, color, fill

g + **geom_violin(**scale = "area"**)**
x, y, alpha, color, fill, linetype, size, weight

Discrete X, Discrete Y

h <- ggplot(diamonds, aes(cut, color))

h + **geom_jitter()**
x, y, alpha, color, fill, shape, size

Continuous Bivariate Distribution

i <- ggplot(movies, aes(year, rating))

i + **geom_bin2d(**binwidth = c(5, 0.5)**)**
xmax, xmin, ymax, ymin, alpha, color, fill, linetype, size, weight

i + **geom_density2d()**
x, y, alpha, colour, linetype, size

i + **geom_hex()**
x, y, alpha, colour, fill size

Continuous Function

j <- ggplot(economics, aes(date, unemploy))

j + **geom_area()**
x, y, alpha, color, fill, linetype, size

j + **geom_line()**
x, y, alpha, color, linetype, size

j + **geom_step(**direction = "hv"**)**
x, y, alpha, color, linetype, size

Visualizing error

df <- data.frame(grp = c("A", "B"), fit = 4:5, se = 1:2)
k <- ggplot(df, aes(grp, fit, ymin = fit-se, ymax = fit+se))

k + **geom_crossbar(**fatten = 2**)**
x, y, ymax, ymin, alpha, color, fill, linetype, size

k + **geom_errorbar()**
x, ymax, ymin, alpha, color, linetype, size, width (also **geom_errorbarh()**)

k + **geom_linerange()**
x, ymin, ymax, alpha, color, linetype, size

k + **geom_pointrange()**
x, y, ymin, ymax, alpha, color, fill, linetype, shape, size

Maps

data <- data.frame(murder = USArrests$Murder,
  state = tolower(rownames(USArrests)))
map <- map_data("state")
l <- ggplot(data, aes(fill = murder))

l + **geom_map(**aes(map_id = state), map = map**)** +
expand_limits(x = map$long, y = map$lat**)**
map_id, alpha, color, fill, linetype, size

Three Variables

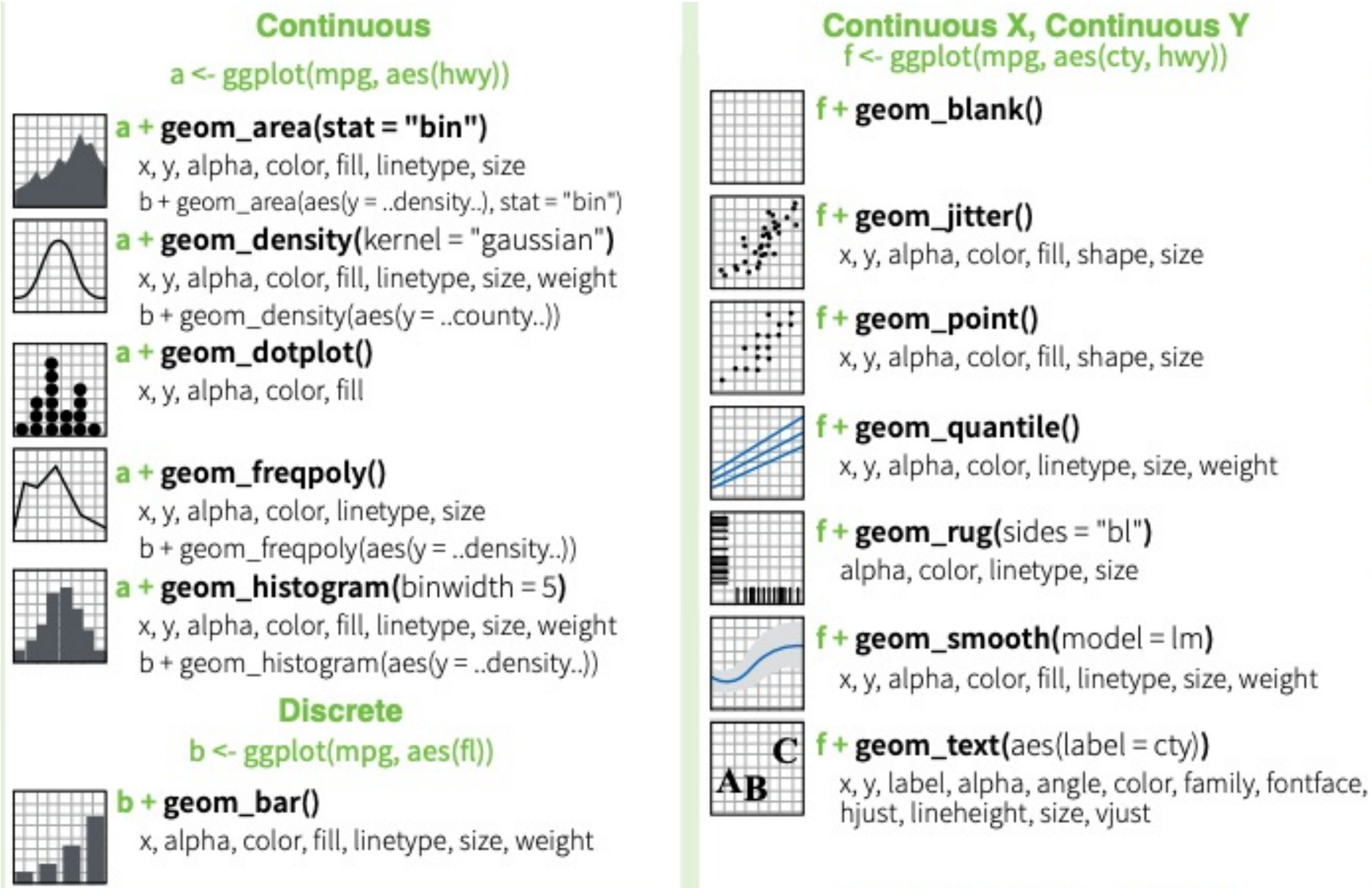
seals$z <- with(seals, sqrt(delta_long^2 + delta_lat^2))
m <- ggplot(seals, aes(long, lat))

m + **geom_contour(**aes(z = z)**)**
x, y, z, alpha, colour, linetype, size, weight

m + **geom_raster(**aes(fill = z), hjust=0.5, vjust=0.5, interpolate=FALSE**)**
x, y, alpha, fill

m + **geom_tile(**aes(fill = z)**)**
x, y, alpha, color, fill, linetype, size

ggplot2 (cheat sheet)



Continuous

a <- ggplot(mpg, aes(hwy))

a + **geom_area(stat = "bin")**
x, y, alpha, color, fill, linetype, size
b + geom_area(aes(y = ..density..), stat = "bin")

a + **geom_density(**kernel = "gaussian"**)**
x, y, alpha, color, fill, linetype, size, weight
b + geom_density(aes(y = ..county..))

a + **geom_dotplot()**
x, y, alpha, color, fill

a + **geom_freqpoly()**
x, y, alpha, color, linetype, size
b + geom_freqpoly(aes(y = ..density..))

a + **geom_histogram(**binwidth = 5**)**
x, y, alpha, color, fill, linetype, size, weight
b + geom_histogram(aes(y = ..density..))

Discrete

b <- ggplot(mpg, aes(fl))

b + **geom_bar()**
x, alpha, color, fill, linetype, size, weight

Continuous X, Continuous Y

f <- ggplot(mpg, aes(cty, hwy))

f + **geom_blank()**

f + **geom_jitter()**
x, y, alpha, color, fill, shape, size

f + **geom_point()**
x, y, alpha, color, fill, shape, size

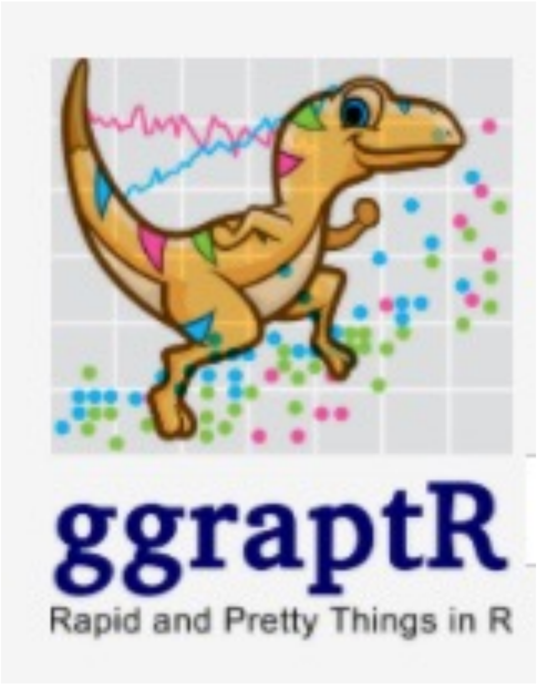
f + **geom_quantile()**
x, y, alpha, color, linetype, size, weight

f + **geom_rug(**sides = "bl"**)**
alpha, color, linetype, size

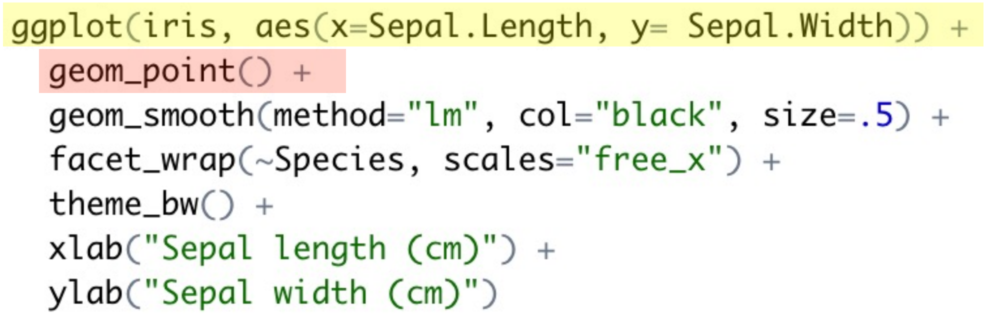
f + **geom_smooth(**model = lm**)**
x, y, alpha, color, fill, linetype, size, weight

f + **geom_text(**aes(label = cty)**)**
x, y, label, alpha, angle, color, family, fontface,
hjust, lineheight, size, vjust

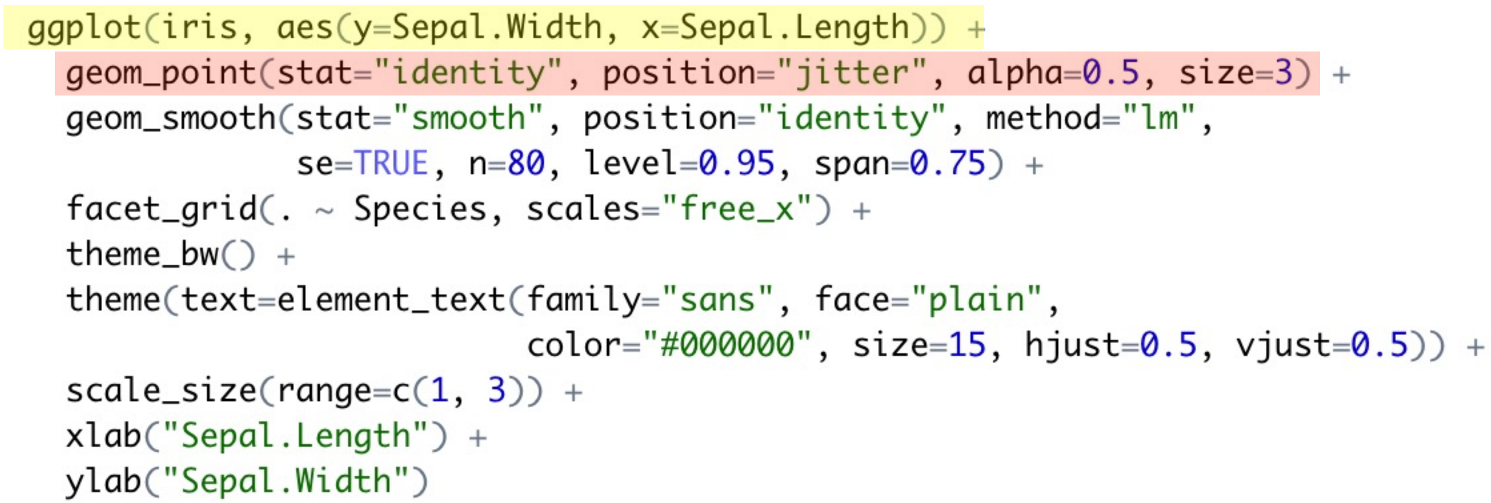
ggraptR – a gentle transition to ggplot



ggraptR – a gentle transition to ggplot



```
ggplot(iris, aes(x=Sepal.Length, y= Sepal.Width)) +
  geom_point() +
  geom_smooth(method="lm", col="black", size=.5) +
  facet_wrap(~Species, scales="free_x") +
  theme_bw() +
  xlab("Sepal length (cm)") +
  ylab("Sepal width (cm)")
```



```
ggplot(iris, aes(y=Sepal.Width, x=Sepal.Length)) +
  geom_point(stat="identity", position="jitter", alpha=0.5, size=3) +
  geom_smooth(stat="smooth", position="identity", method="lm",
              se=TRUE, n=80, level=0.95, span=0.75) +
  facet_grid(. ~ Species, scales="free_x") +
  theme_bw() +
  theme(text=element_text(family="sans", face="plain",
                          color="#000000", size=15, hjust=0.5, vjust=0.5)) +
  scale_size(range=c(1, 3)) +
  xlab("Sepal.Length") +
  ylab("Sepal.Width")
```

Homework

Using the betta data and ggplot2 make an awesome plot that includes 3 variables (2 continuous and 1 discrete) – Due by Tuesday class time.

Correlation vs Regression

- Both methods are ways to explore contingency between variables.
- Regression describes the degree to which we can predict the value of one variable based on the value of another.
- Regression calculates a line that describes this relationship between two variables.
- Use regression when you believe there is a strong case for causation.

Correlation vs Regression

- Both methods are ways to explore contingency between variables.
- Regression describes the degree to which we can predict the value of one variable based on the value of another.
- Regression calculates a line that describes this relationship between two variables.
- Use regression when you believe there is a strong case for causation.

Terminology

Linear regression vs.

OLS regression (Ordinary least squares regression) vs.

General linear models vs.

Generalized linear model

Terminology

Linear regression vs.

OLS regression (Ordinary least squares regression) vs.

General linear models vs.

Generalized linear model

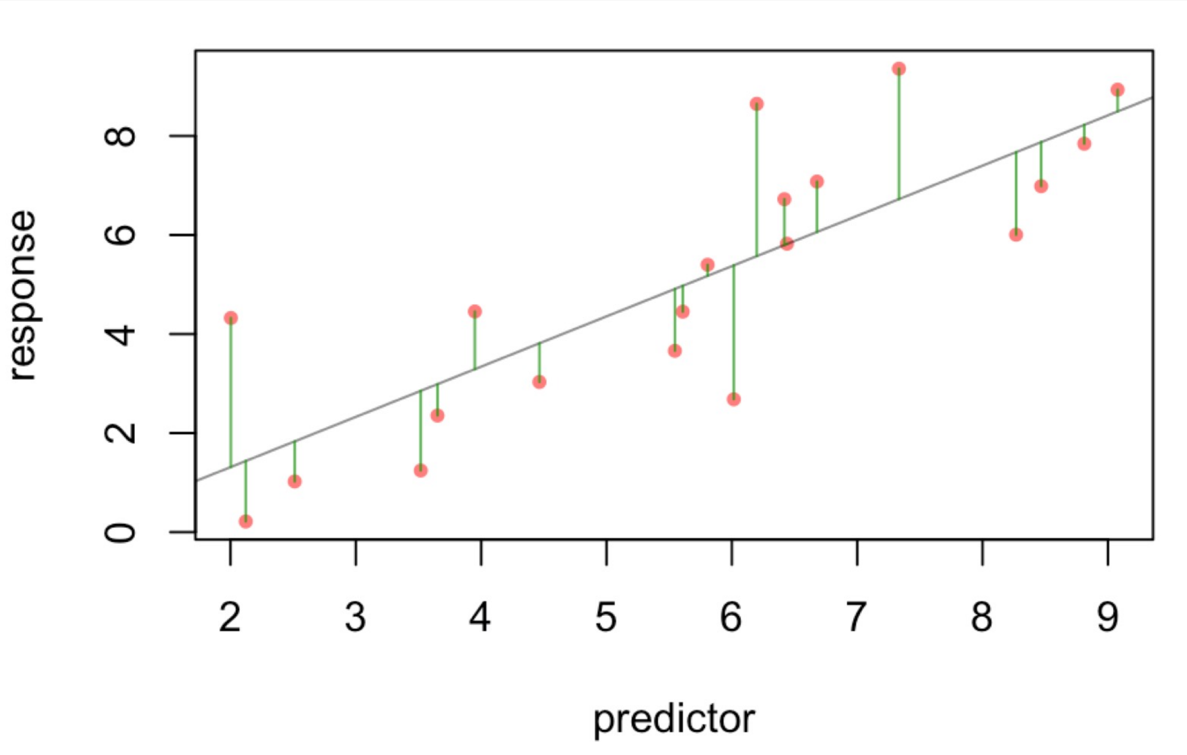
`glm`

Regression in R

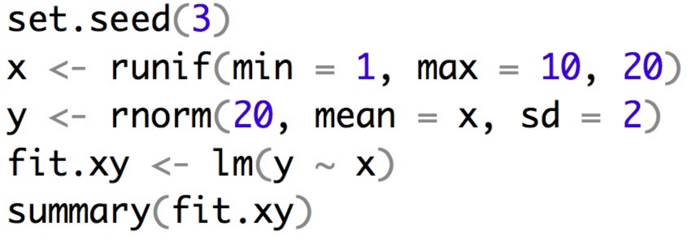
1) With linear regression we find the linear equation that best predicts the values of Y based on the values of X.

2) $y = bx + a$

3) Least-squares regression minimizes the squared deviations of the data points from that line.



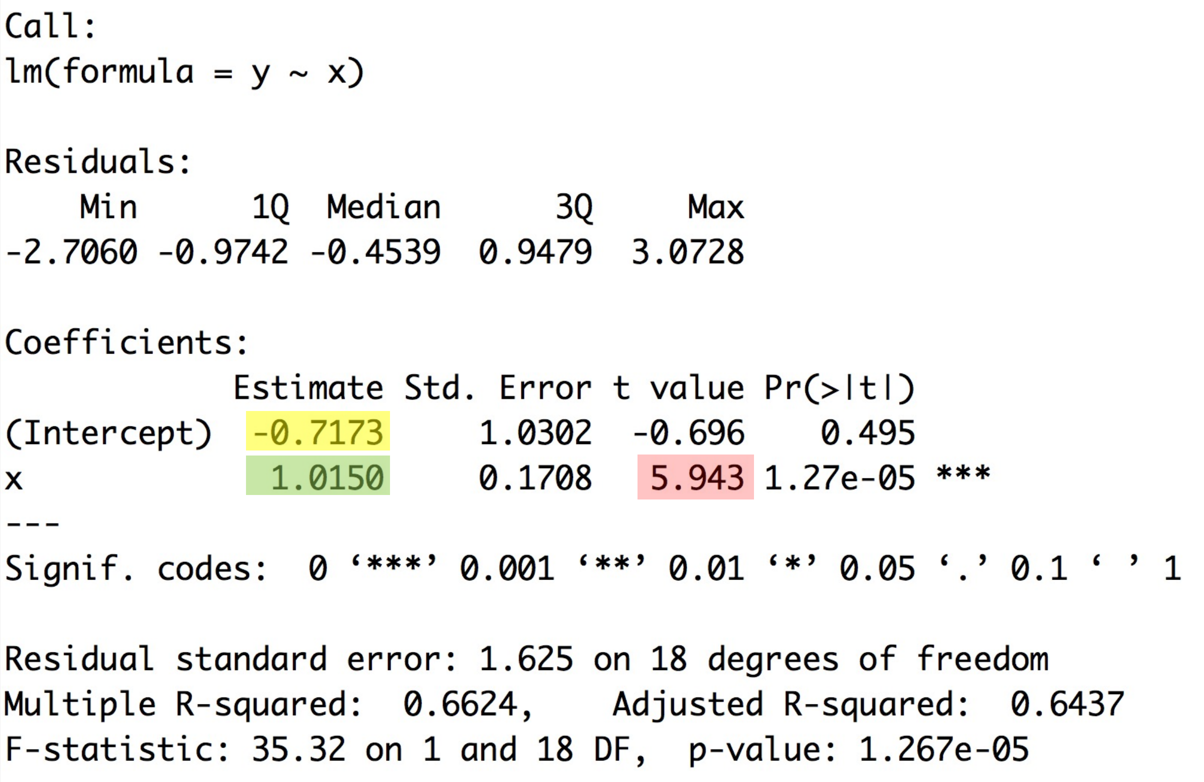
Example of regression



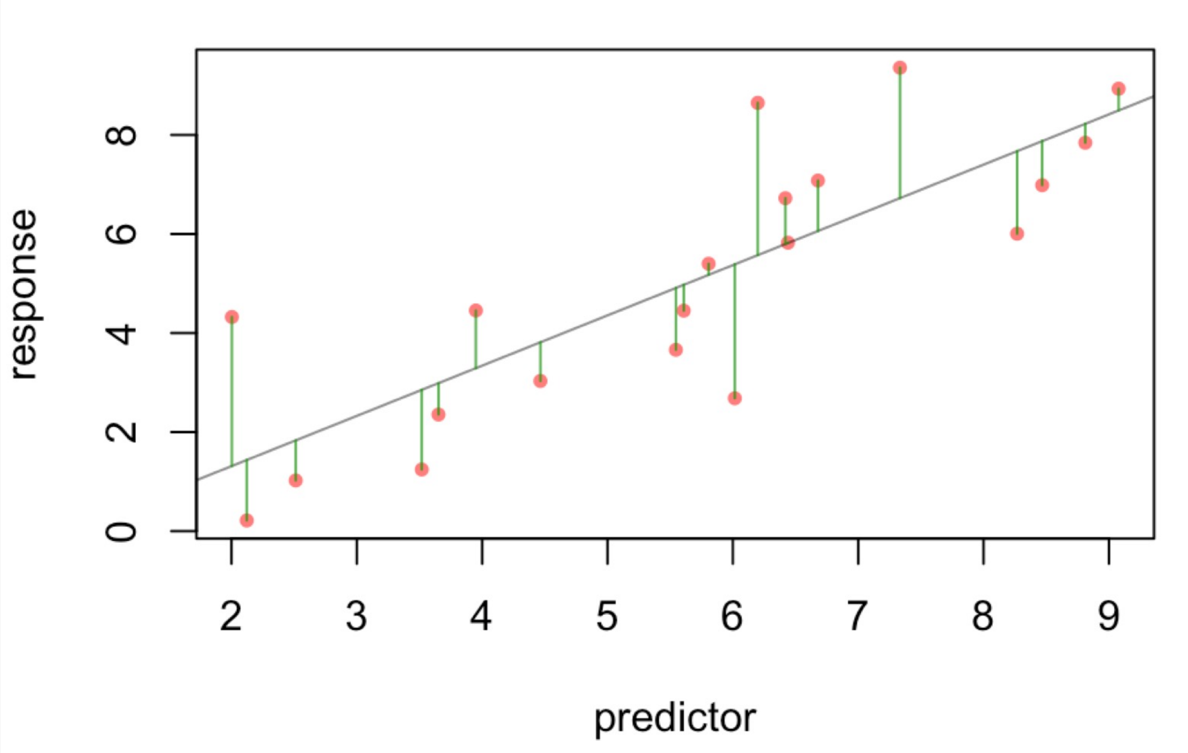
```r
set.seed(3)
x <- runif(min = 1, max = 10, 20)
y <- rnorm(20, mean = x, sd = 2)
fit.xy <- lm(y ~ x)
summary(fit.xy)
```

$$y = bx + a$$

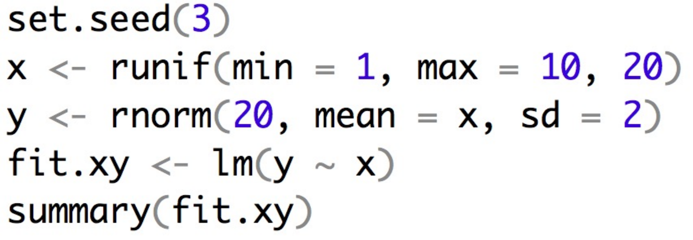
$$t = \frac{\beta_0}{SE_b}$$



```
Call:
lm(formula = y ~ x)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.7060 -0.9742 -0.4539  0.9479  3.0728 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  -0.7173     1.0302  -0.696    0.495    
x             1.0150     0.1708   5.943 1.27e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.625 on 18 degrees of freedom
Multiple R-squared:  0.6624,	Adjusted R-squared:  0.6437 
F-statistic: 35.32 on 1 and 18 DF,  p-value: 1.267e-05
```



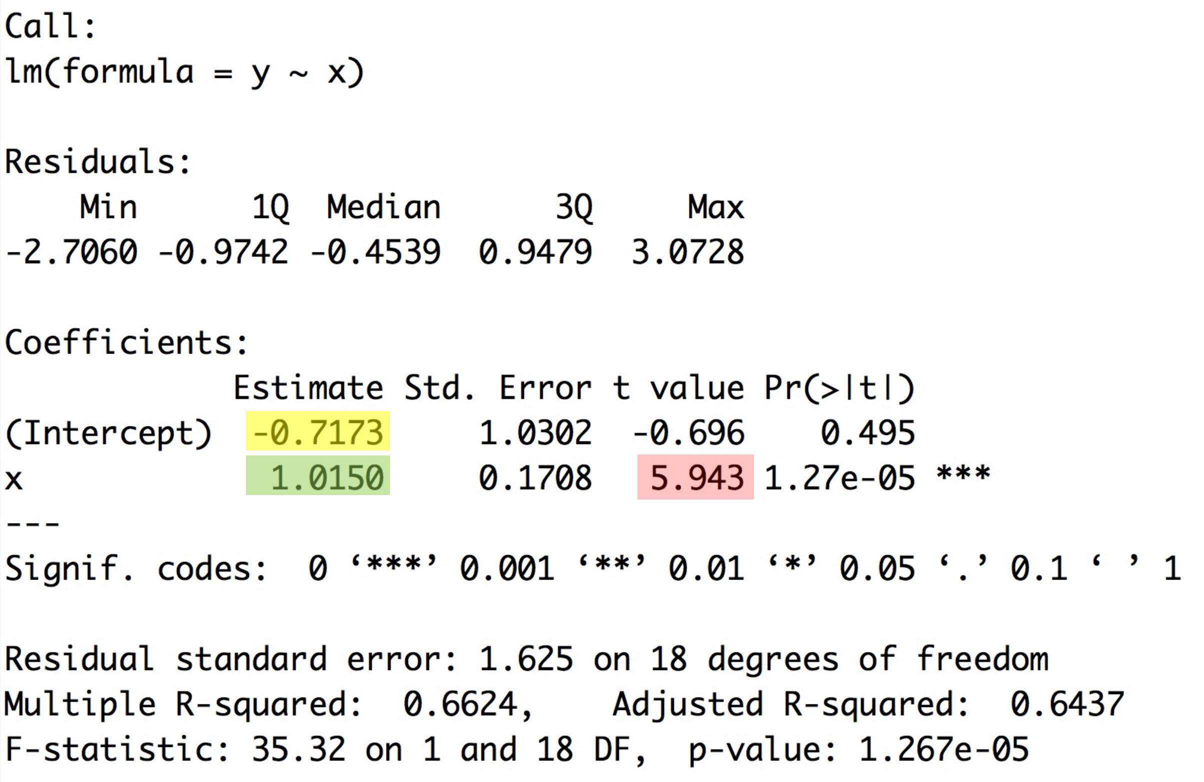
Example of regression



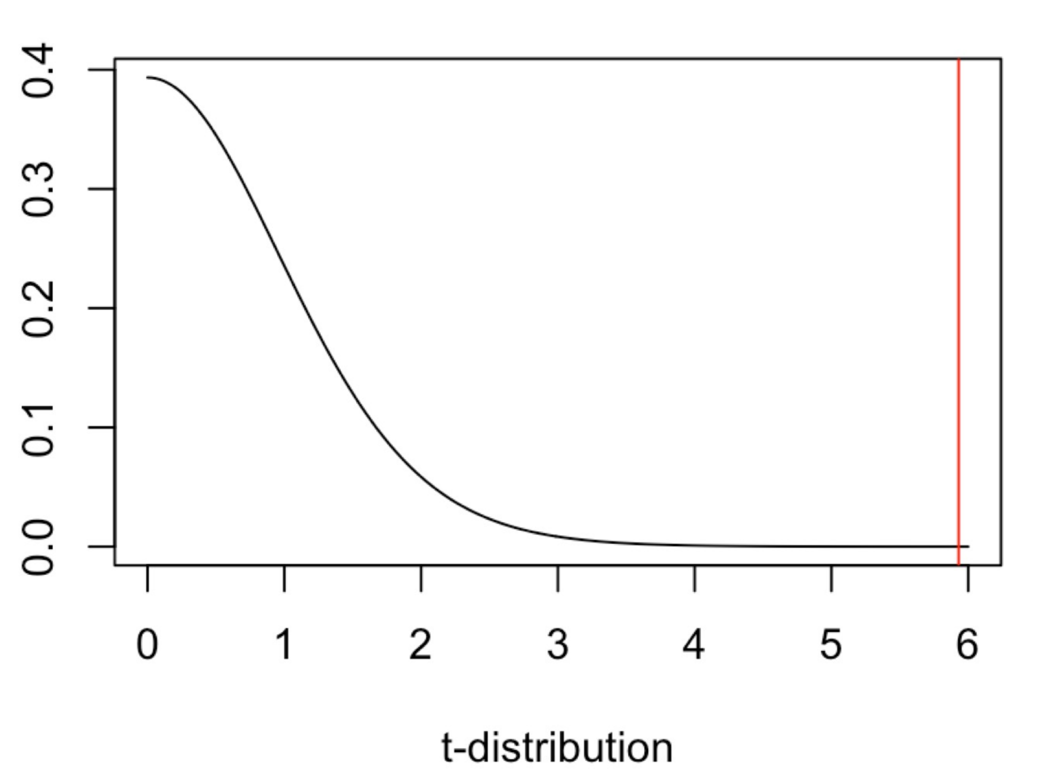
```
set.seed(3)
x <- runif(min = 1, max = 10, 20)
y <- rnorm(20, mean = x, sd = 2)
fit.xy <- lm(y ~ x)
summary(fit.xy)
```

$$y = bx + a$$

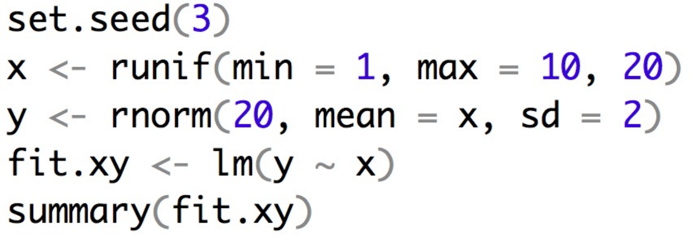
$$t = \frac{\beta_0}{SE_b}$$



```
Call:
lm(formula = y ~ x)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.7060 -0.9742 -0.4539  0.9479  3.0728 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  -0.7173     1.0302  -0.696    0.495    
x             1.0150     0.1708   5.943 1.27e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.625 on 18 degrees of freedom
Multiple R-squared:  0.6624,	Adjusted R-squared:  0.6437 
F-statistic: 35.32 on 1 and 18 DF,  p-value: 1.267e-05
```

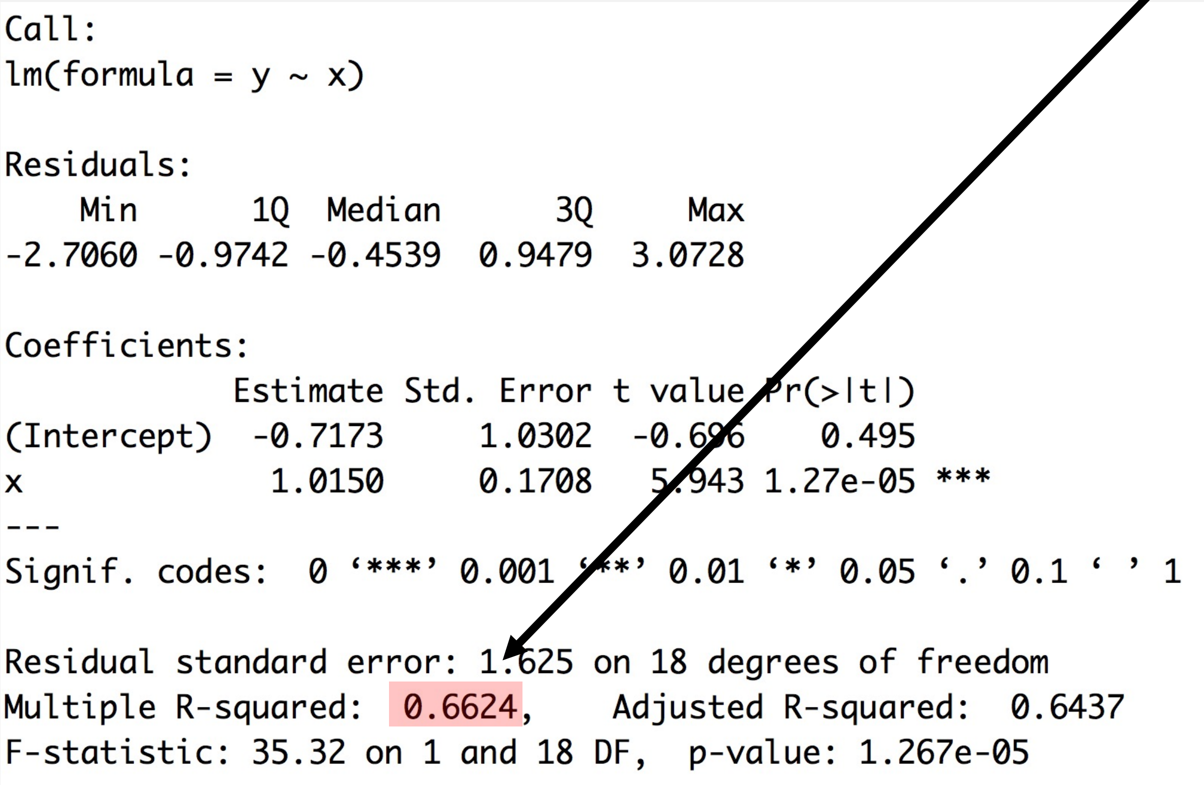


Example of regression

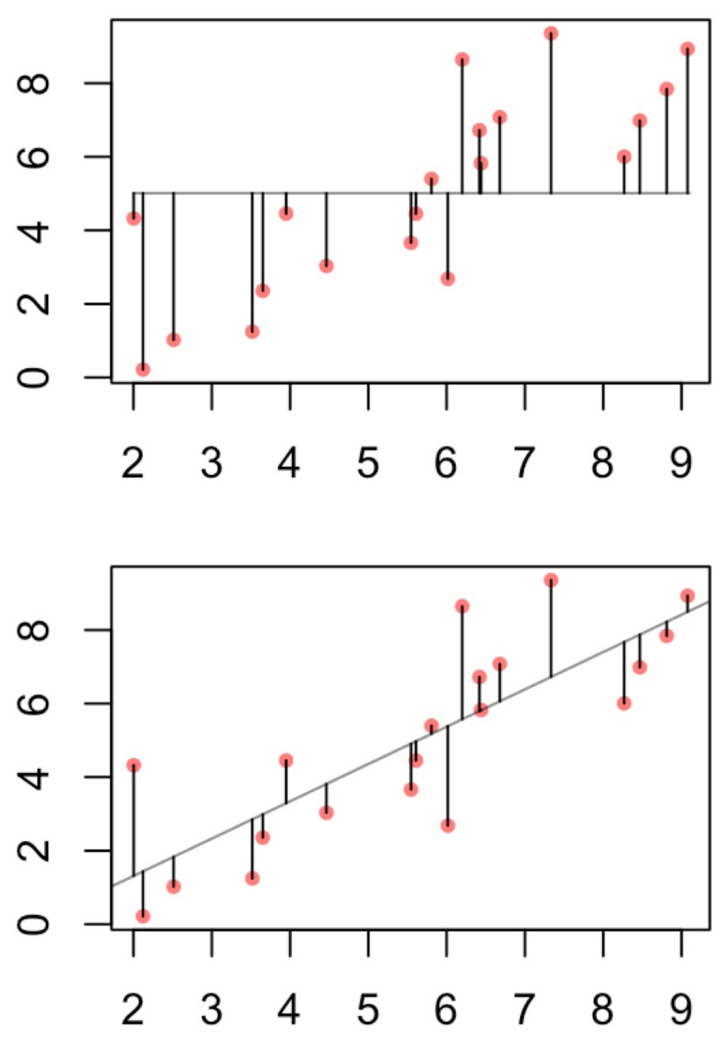


```
set.seed(3)
x <- runif(min = 1, max = 10, 20)
y <- rnorm(20, mean = x, sd = 2)
fit.xy <- lm(y ~ x)
summary(fit.xy)
```

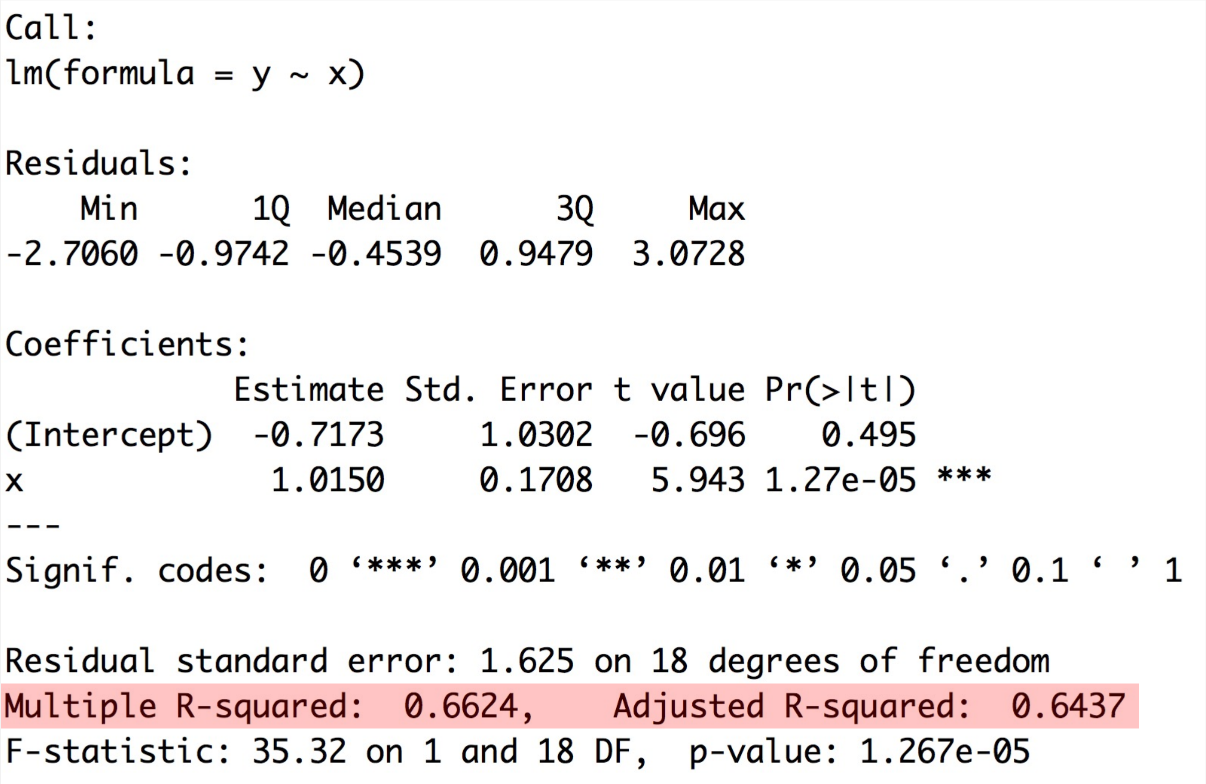
This can help to justify the biological importance assuming you have a regression that is significant. It is the proportion of total variance explained by the regression.



```
Call:
lm(formula = y ~ x)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.7060 -0.9742 -0.4539  0.9479  3.0728 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  -0.7173     1.0302  -0.696    0.495    
x             1.0150     0.1708   5.943 1.27e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.625 on 18 degrees of freedom
Multiple R-squared:  0.6624,	Adjusted R-squared:  0.6437 
F-statistic: 35.32 on 1 and 18 DF,  p-value: 1.267e-05
```



Multiple vs Adjusted R-squared



```
Call:
lm(formula = y ~ x)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.7060 -0.9742 -0.4539  0.9479  3.0728 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  -0.7173     1.0302  -0.696    0.495    
x             1.0150     0.1708   5.943 1.27e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.625 on 18 degrees of freedom
Multiple R-squared:  0.6624,	Adjusted R-squared:  0.6437 
F-statistic: 35.32 on 1 and 18 DF,  p-value: 1.267e-05
```

Adjusted R-squared penalizes for additional parameters

Linear regression uses

- Depict the relationship between two variables in an eye-catching fashion

- Test the null hypothesis of no association between two variables
  - The test is whether or not the slope is zero

- Predict the average value of variable *Y* for a group of individuals with a given value of variable *X*
  - variation around the line can make it very difficult to predict a value for a given individual with much confidence
  - Predictions outside of the range of observed data is generally discouraged

- Used both for experimental and observational studies

What are Residuals

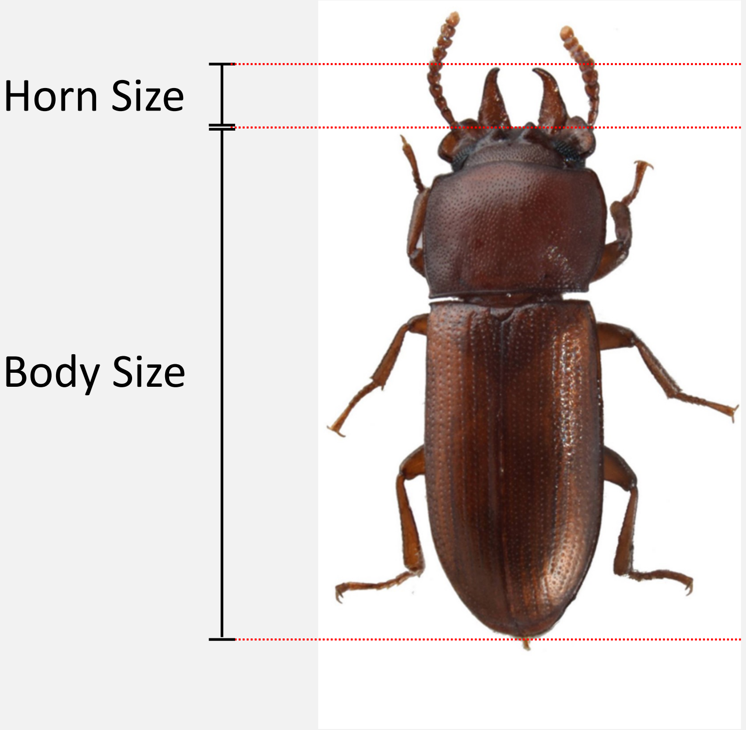
In general, the residual is the individual’s departure from the value predicted by the model

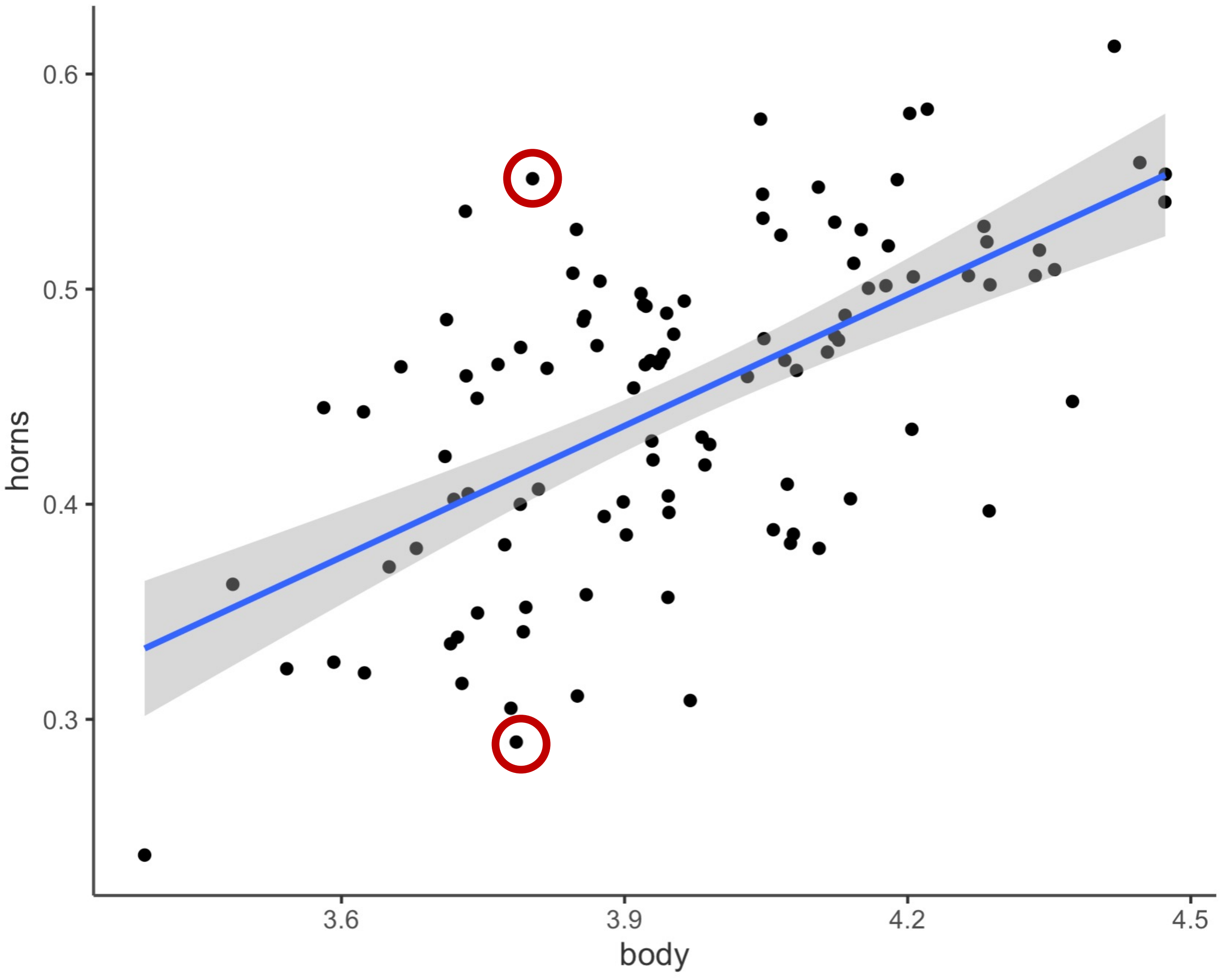
In this case the model is simple – the linear regression – but residuals also exist for more complex models

For a model that fits better, the residuals will be smaller on average

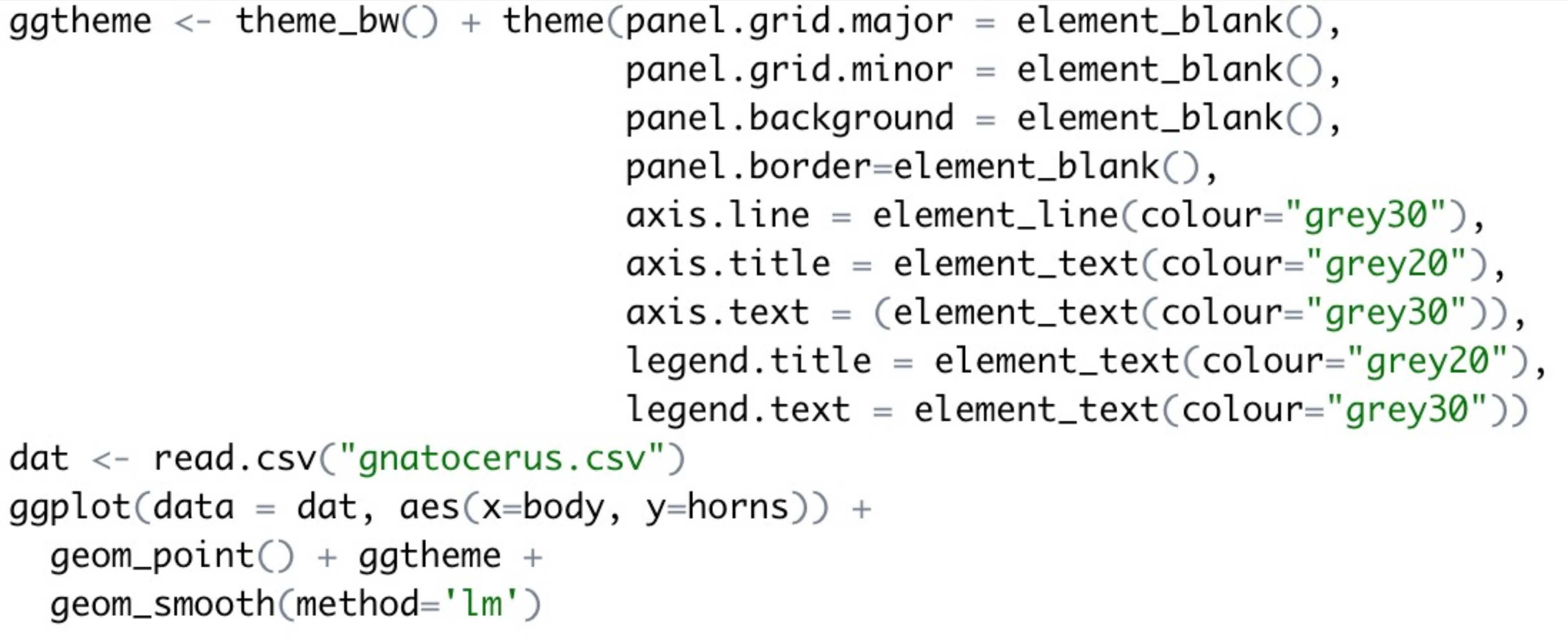
Residuals can be of interest in their own right, because they represent values that have been ***corrected*** for relationships that might be obscuring a pattern.

What are Residuals





Making that plot



```r
ggtheme <- theme_bw() + theme(panel.grid.major = element_blank(),
                              panel.grid.minor = element_blank(),
                              panel.background = element_blank(),
                              panel.border=element_blank(),
                              axis.line = element_line(colour="grey30"),
                              axis.title = element_text(colour="grey20"),
                              axis.text = (element_text(colour="grey30")),
                              legend.title = element_text(colour="grey20"),
                              legend.text = element_text(colour="grey30"))
dat <- read.csv("gnatocerus.csv")
ggplot(data = dat, aes(x=body, y=horns)) +
  geom_point() + ggtheme +
  geom_smooth(method='lm')
```

Strong Inference for Observational Studies

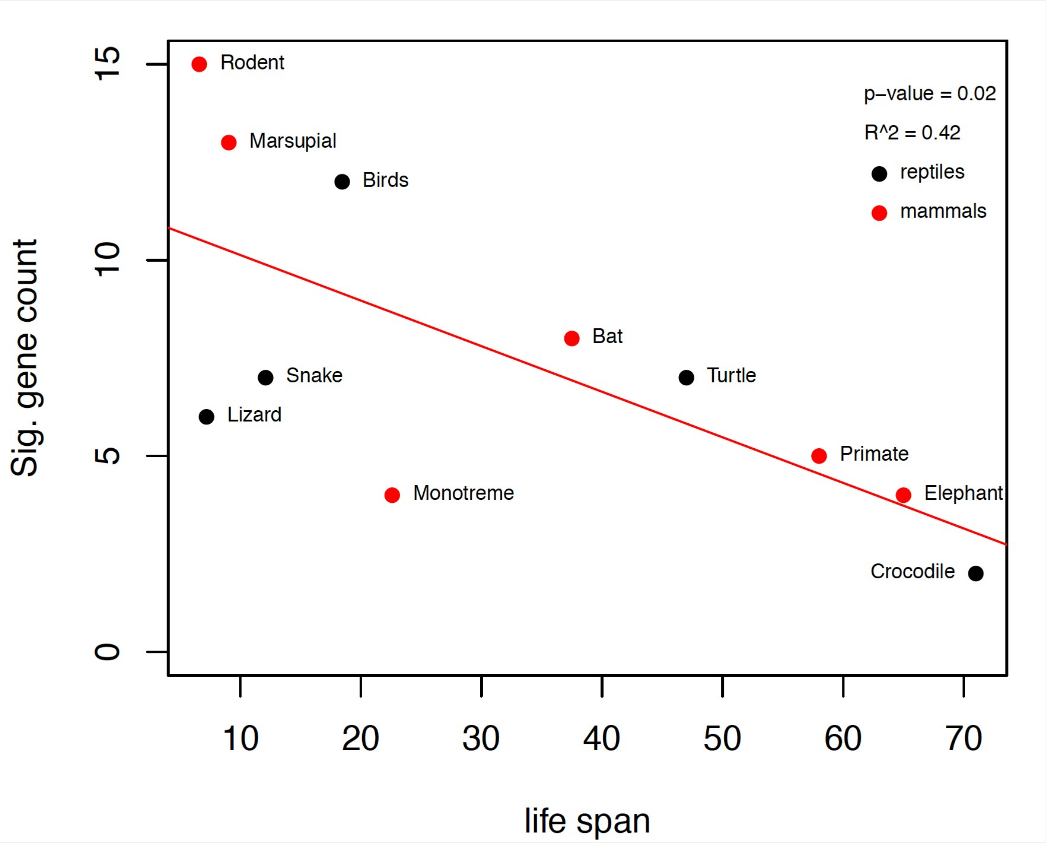
- Noticing a pattern in the data and reporting it represents a post hoc analysis
- This is not hypothesis testing
- The results, while potentially important, must be interpreted cautiously

What can be done?

- Based on a post-hoc observational study, construct a new hypothesis for a novel group or system that has not yet been studied

Example

1) We already knew that the P53 network is important in guarding against cancer in long lived species.
2) We also knew that primates and elephants show rather little change in this network when compared to rodents.
3) Collect data on many more species and test apriori hypothesis that there will be a significant and negative regression coefficient.



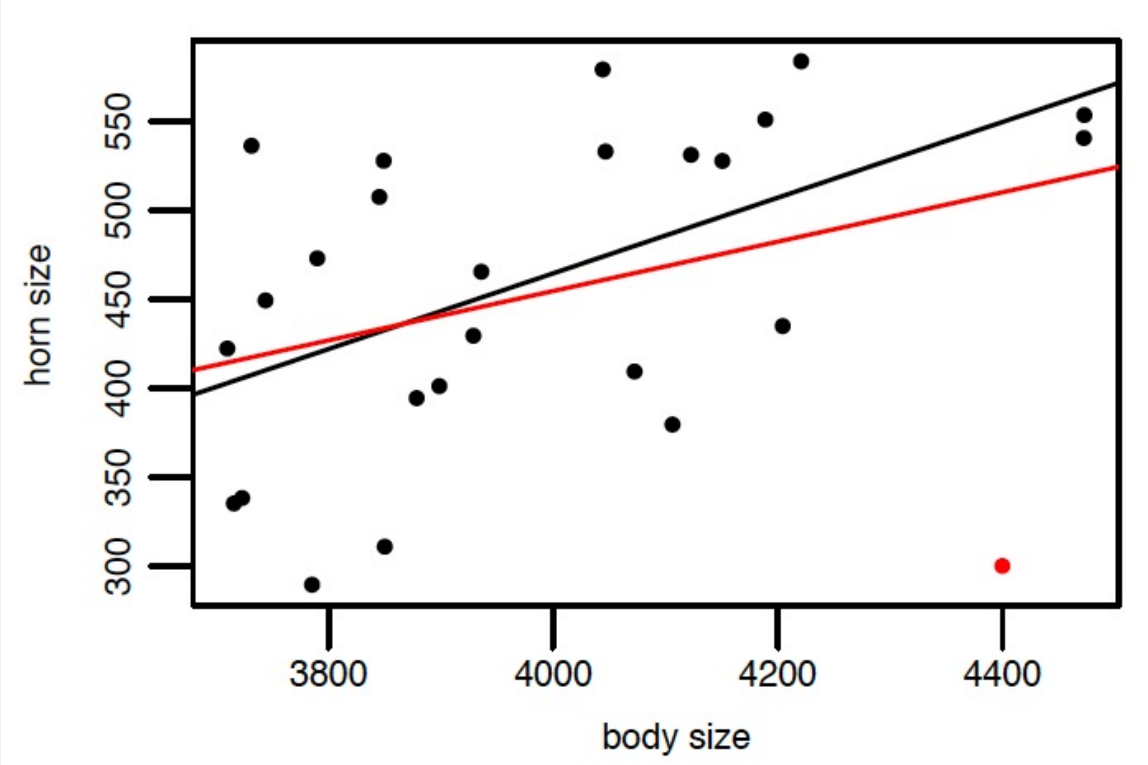
Assumptions of Linear Regression

- The true relationship must be linear
- At each value of *X*, the distribution of *Y* is normal (i.e., the residuals are normal)
- The variance in *Y* is independent of the value of *X*
- **Note that there are no assumptions about the distribution of *X***

Common Problems

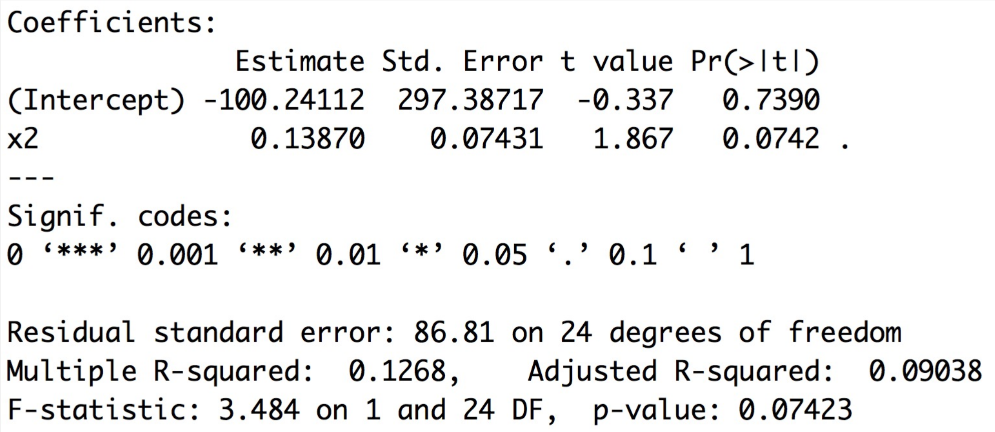
- Outliers
  - Regression is extremely sensitive to outliers
  - The line will be drawn to outliers, especially along the x-axis
  - Consider performing the regression with and without outliers
- Non-linearity
  - Best way to notice is by visually inspecting the plot and the line fit
  - Try a transformation to get linearity [often a log transformation]
- Non-normality of residuals
  - Can be detected from a residual plot
  - Possibly solved with a transformation
- Unequal variance
  - Usually visible from a scatterplot or from a residual plot

Outliers

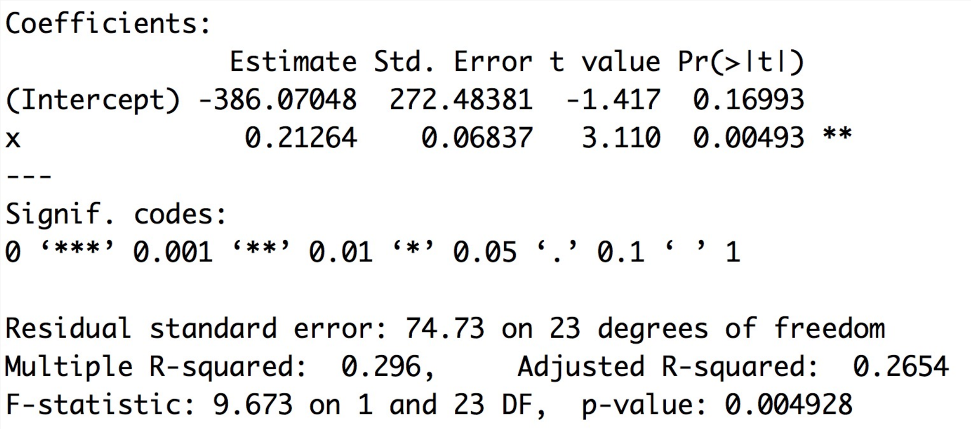


Leverage and cooks distance

Theil–Sen estimator



```
Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept) -100.24112  297.38717  -0.337   0.7390
x2             0.13870    0.07431   1.867   0.0742 .
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 86.81 on 24 degrees of freedom
Multiple R-squared:  0.1268,	Adjusted R-squared:  0.09038
F-statistic: 3.484 on 1 and 24 DF,  p-value: 0.07423
```



```
Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept) -386.07048  272.48381  -1.417  0.16993
x              0.21264    0.06837   3.110  0.00493 **
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 74.73 on 23 degrees of freedom
Multiple R-squared:  0.296,	Adjusted R-squared:  0.2654
F-statistic: 9.673 on 1 and 23 DF,  p-value: 0.004928
```

Moving past simple models

- The reason ANOVA is so widely used is that it provides a framework to simultaneously test the effects of multiple factors

- ANOVA also makes it possible to detect ***interactions*** among the factors

- ANOVA is a special case of a ***general linear model***

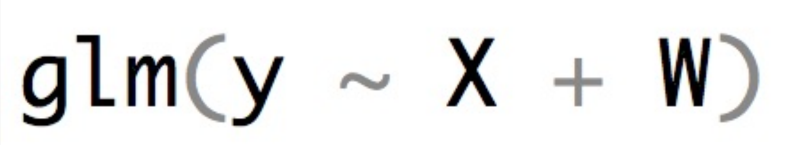
- Linear regression is a special case of a ***general linear model***

GLM and LM function in R

- The GLM and LM function in R takes equations that can be described with the following operators

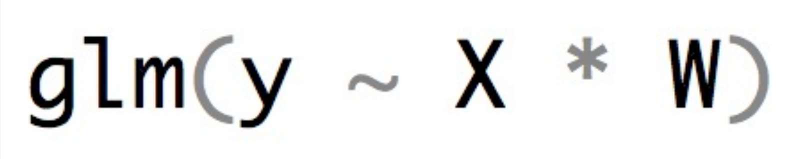
+	+X include this variable
:	X:Z include the interaction between these variables
∗	X∗Y include these variables and the interactions between them
^	(X + Z + W)^3 include these variables and all interactions up to three way

R versus the math implied



```
glm(y ~ X + W)
```

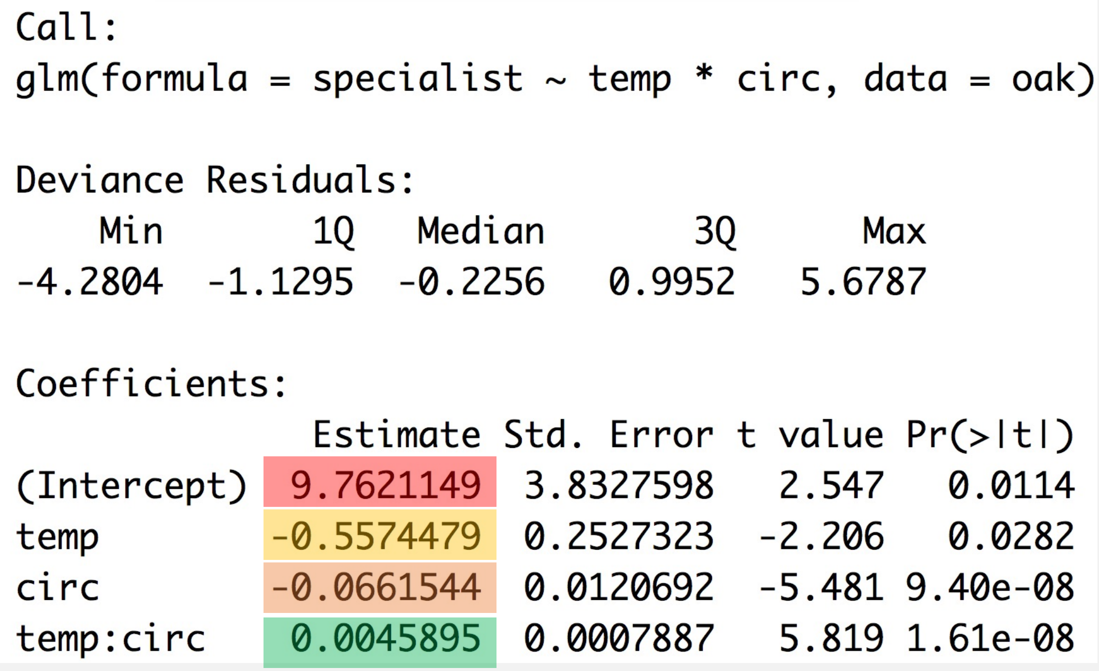
$$y_i = \beta_0 + \beta_1 X_i + \beta_2 W_i + \epsilon_i$$



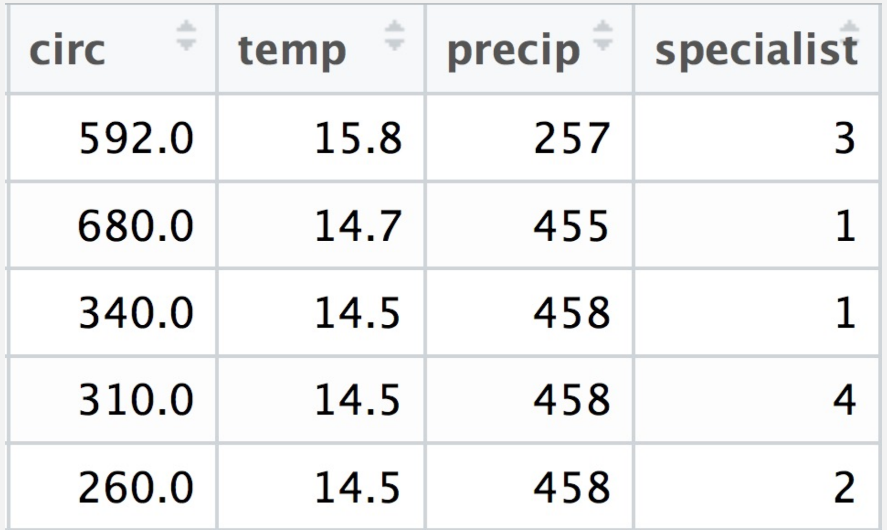
```
glm(y ~ X * W)
```

$$y_i = \beta_0 + \beta_1 X_i + \beta_2 W_i + \beta_3 X_i W_i + \epsilon_i$$

R versus the math oak example



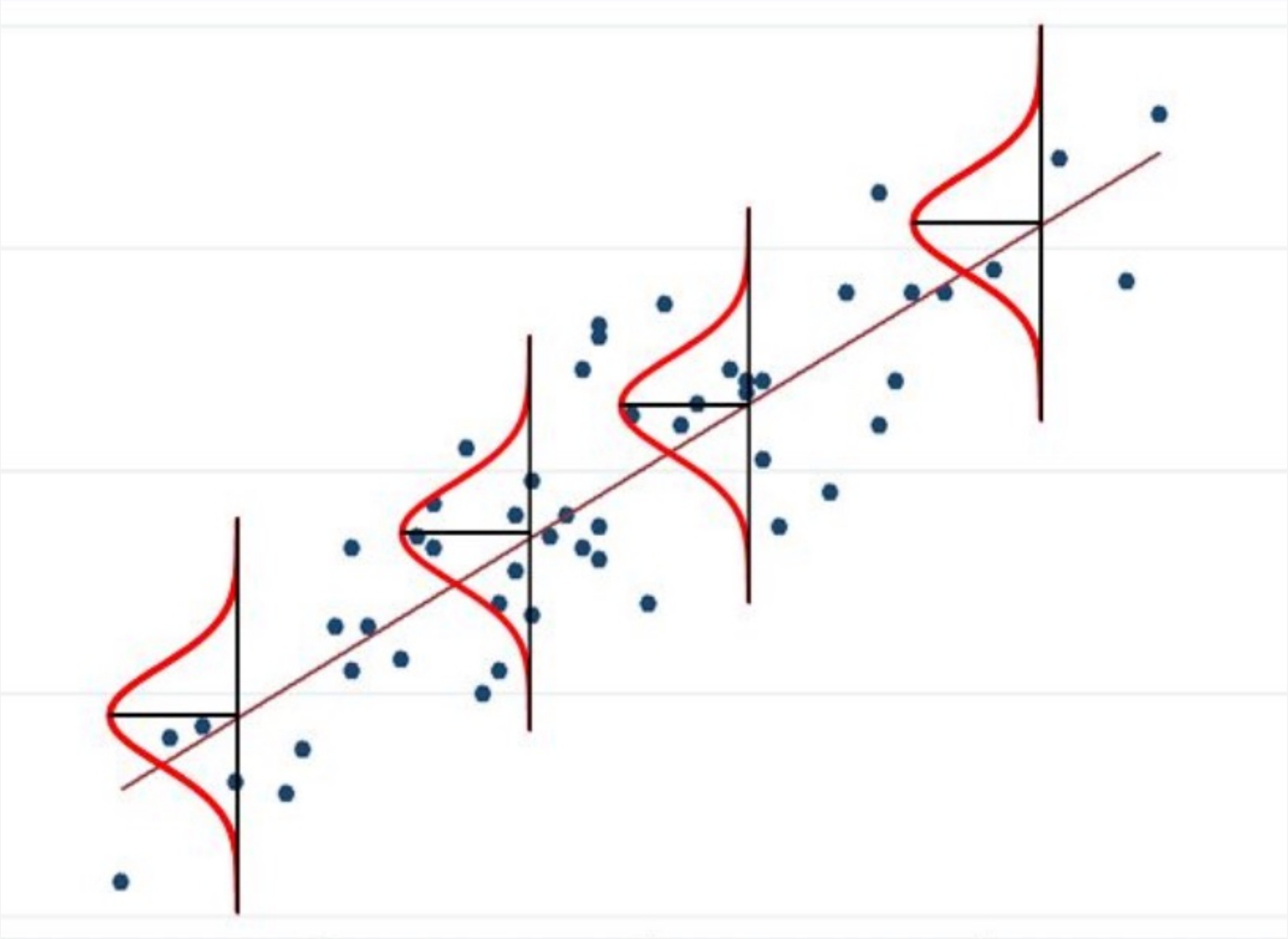
```
Call:
glm(formula = specialist ~ temp * circ, data = oak)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-4.2804  -1.1295  -0.2256   0.9952   5.6787

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  9.7621149  3.8327598   2.547   0.0114
temp        -0.5574479  0.2527323  -2.206   0.0282
circ        -0.0661544  0.0120692  -5.481 9.40e-08
temp:circ    0.0045895  0.0007887   5.819 1.61e-08
```



circ	temp	precip	specialist
592.0	15.8	257	3
680.0	14.7	455	1
340.0	14.5	458	1
310.0	14.5	458	4
260.0	14.5	458	2

$$y_i = \beta_0 + \beta_1 temp_i + \beta_2 circ_i + \beta_3 temp_i circ_i$$

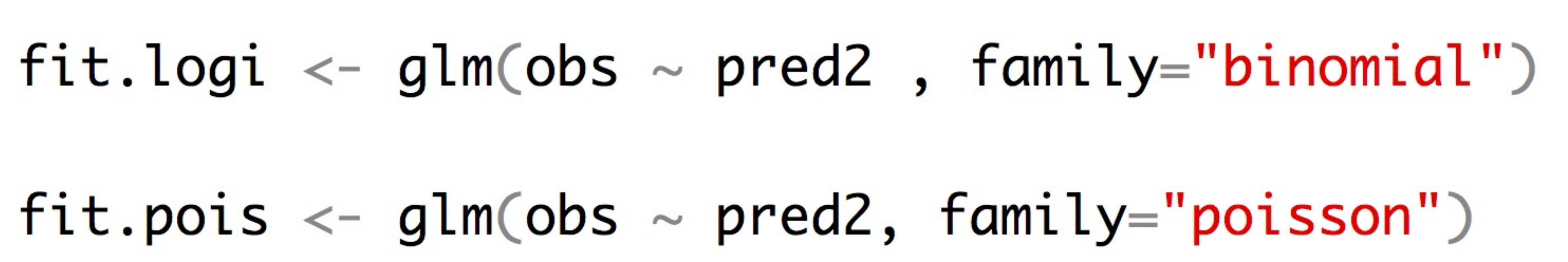
When the response variable isn’t normal



Other kinds of regression

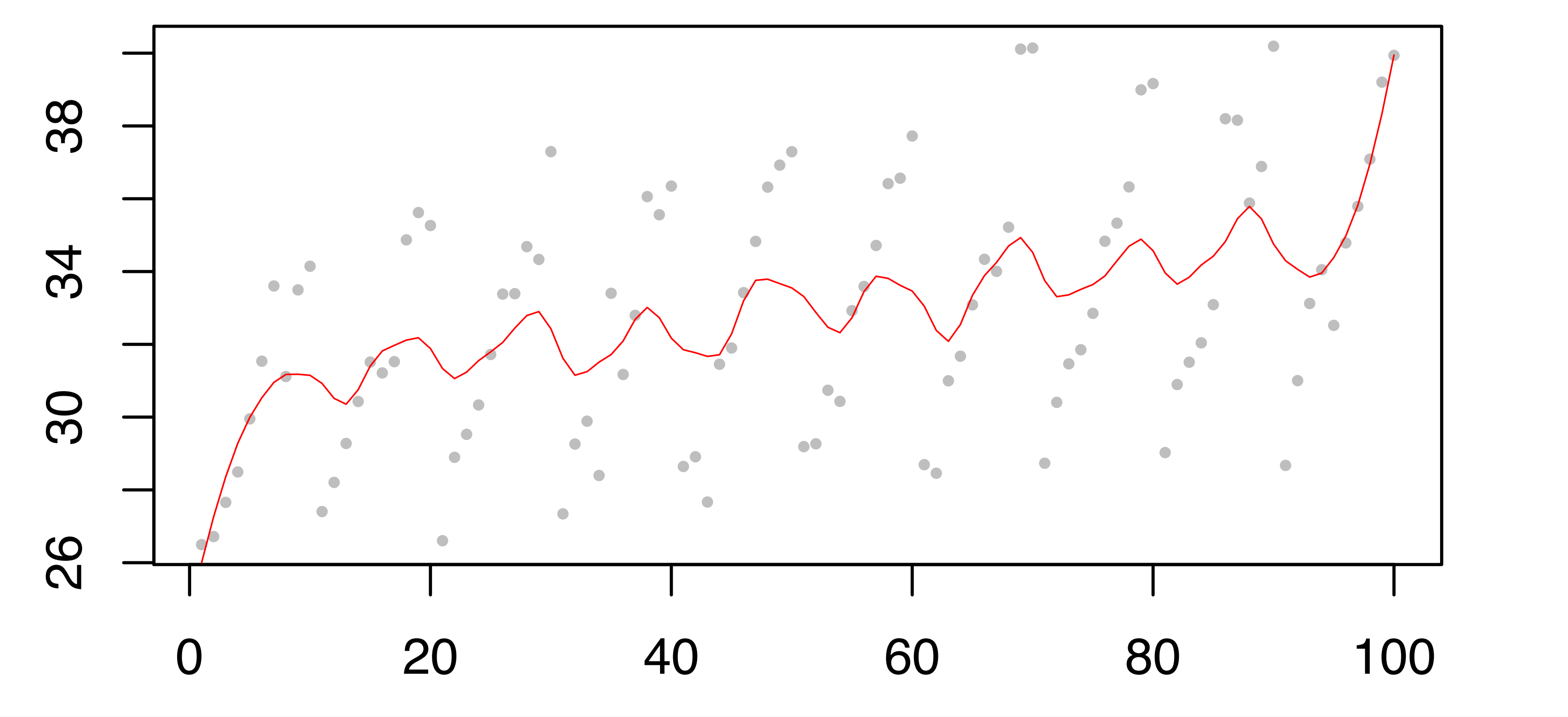
Logistic regression allows us to fit a binary response variable (absent/present; alive/dead) with one or more categorical or continuous predictor variables.

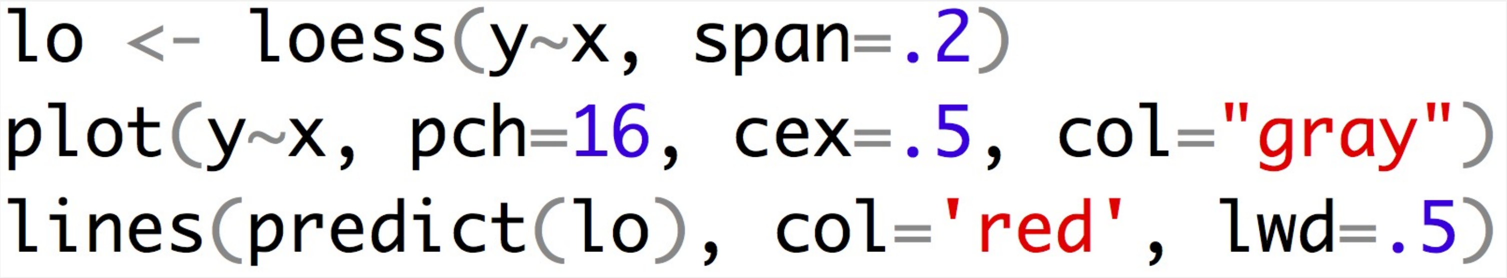
Poisson regression allows us to fit a response variable that is Poisson distributed (number of extinctions in a unit of time, number of colonies per plate, (number of occurrences for rare events)) with one or more categorical or continuous predictor variables.



```
fit.logi <- glm(obs ~ pred2 , family="binomial")

fit.pois <- glm(obs ~ pred2, family="poisson")
```

Sometimes regression isn’t best choice





```
lo <- loess(y~x, span=.2)
plot(y~x, pch=16, cex=.5, col="gray")
lines(predict(lo), col='red', lwd=.5)
```