STATS 330

Handout 2
Explanatory terms and interpretation

Department of Statistics, University of Auckland

Interpretation and communication

Sometimes the main goal of our analysis is explaining the relationships between variables For example, we may have the following questions of interest:

- Does the injection of LSD cause a change in subjects' arithmetic skills? If so, how?
- Does the dose of Amphotericin B affect the number of *M. Ornithogaster* organisms shedded in chickens' faeces? If so, how?
- Does age play a role in the occcurrence of coronary heart disease? If so, how?

In this handout, we will recap how we answered these types of questions in STATS 20x.

Fitting GLMs in R

Linear regression: LSD analysis

```
## Either
lsd.fit <- lm(score ~ lsd)
## Or
lsd.fit <- glm(score ~ lsd, family = "gaussian")
```

Poisson regression: *M. Ornithogaster* analysis

```
chickens.fit <- glm(mo ~ weight, family = "poisson")
```

Logistic regression: CHD analysis

```
chd.fit <- glm(cbind(y, n - y) ~ age, family = "binomial")
```

Interpretation: numeric explanatory variables

Linear regression

$$y = mx + c$$

- when x is zero, y equals c
- For every one-unit increase in x, y increases by m

Interpretation: numeric explanatory variables

Linear regression

$$y = mx + c$$

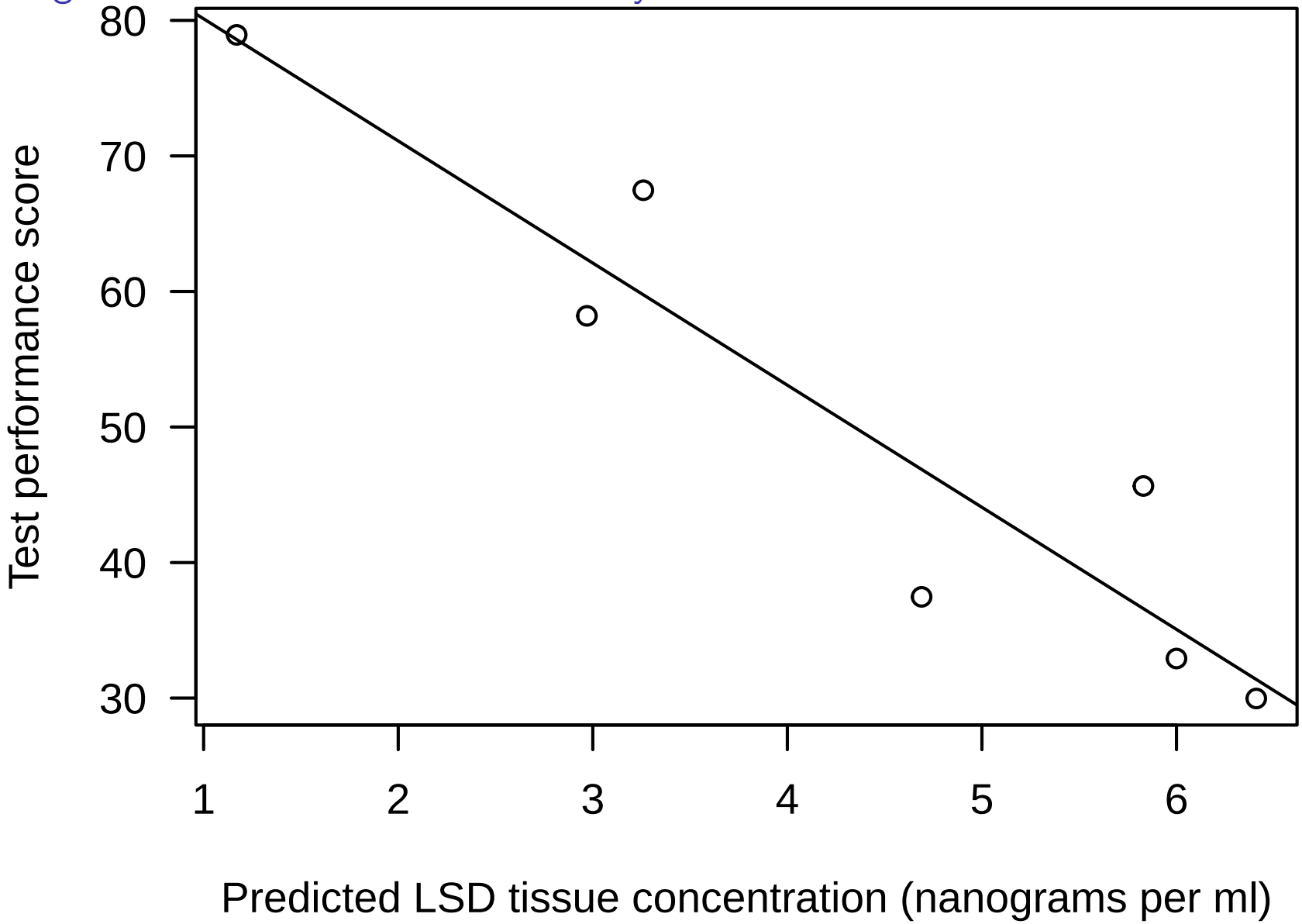
- when x is zero, y equals c
- For every one-unit increase in x, y increases by m

$$\mu = \beta_0 + \beta_1 x$$

- Same thing!
- when x is zero, the expected value of the response equals β_0
- For every one-unit increase in x, the expected value of the response increases by β_1

Interpretation: numeric explanatory variables

Linear regression: LSD effect on maths analysis



Interpretation: numeric explanatory variables

Linear regression: LSD effect on maths analysis

```
lsd.fit <- lm(score ~ lsd)
summary(lsd.fit)

## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   89.124      7.048  12.646 5.49e-05 ***
## lsd           -9.009      1.503  -5.994  0.00185 **
## ---
## Residual standard error: 7.126 on 5 degrees of freedom
## Multiple R-squared:  0.8778,Adjusted R-squared:  0.8534
## F-statistic: 35.93 on 1 and 5 DF,  p-value: 0.001854
```

Interpretation: numeric explanatory variables

Linear regression: LSD effect on maths analysis

```
coef(lsd.fit)

## (Intercept)         lsd
##   89.123874   -9.009466
```

So our fitted relationship is

$$\begin{aligned}\hat{\mu}_i &= \hat{\beta}_0 + \hat{\beta}_1 x_i \\ &= 89.12 - 9.01 x_i,\end{aligned}$$

where x_i is the LSD tissue concentration for the ith time point.

The hats above the parameters indicate that they are estimates. They are not the 'true' parameter values, and are subject to random sampling error.

Interpretation: numeric explanatory variables

Linear regression: LSD effect on maths analysis

We can make the following statements:

- We estimate that the expected average arithmetic test score for subjects prior to being injected with LSD is approximately 89.1 marks.
- We estimate that, for every 1 nanogram per ml increase in LSD tissue concentration, the expected average arithmetic test score decreases by approximately 9.0 marks.

However, these sentences to not communicate the uncertainty in our estimates. How far from 'truth' might they be? What is the potential magnitude of the sampling error?

We should instead focus on interpreting confidence intervals, rather than our point estimates.

LSD effect on maths analysis

Interpretation

```
confint(lsd.fit)

##                 2.5 %     97.5 %
## (Intercept)  71.00758 107.240169
## lsd         -12.87325  -5.145685
```

- We estimate that the average arithmetic test score for subjects prior to being injected with LSD is between 71.0 and 107.2 marks.
- We estimate that, for every 1 nanogram per ml increase in LSD tissue concentration, the average arithmetic test score decreases by between 5.1 and 12.9 marks.

Interpretation: numeric explanatory variables

Linear regression: LSD effect on maths analysis

We have answered our questions of interest:

- Does the injection of LSD cause a change in subjects' arithmetic skills?
  - We have evidence to suggest it does; it is not plausible that an increase in LSD tissue concentration does not affect the average arithmetic score, because it is not plausible that $\beta_1 = 0$.
- If so, how?
  - We estimate that, for every 1 nanogram per ml increase in LSD tissue concentration, the average arithmetic test score decreases by between 5.1 and 12.9 marks.

Alternatively, the first question could be achieved by a formal hypothesis test, rather than by observing whether or not the confidence interval includes zero.

Interpretation: numeric explanatory variables

Linear regression: LSD effect on maths analysis

```
summary(lsd.fit)

## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   89.124      7.048  12.646 5.49e-05 ***
## lsd           -9.009      1.503  -5.994  0.00185 **
## ---
## Residual standard error: 7.126 on 5 degrees of freedom
## Multiple R-squared:  0.8778,Adjusted R-squared:  0.8534
## F-statistic: 35.93 on 1 and 5 DF,  p-value: 0.001854
```

Interpretation: numeric explanatory variables

Linear regression: LSD effect on maths analysis

Each p-value in the rightmost column tests the null hypothesis that the true parameter value for that row is equal to zero.

The null hypothesis we wish to test is

$$H_0 : \beta_1 = 0,$$

because under this hypothesis there is no relationship between LSD tissue concentration and the mean arithmetic test score.

We have a p-value of 0.00185, so there is strong evidence against this null hypothesis; there appears to be a relationship between LSD tissue concentration and the mean arithmetic test score.

Interpretation: numeric explanatory variables

Poisson regression

For Poisson regression we have

$$\log(\mu) = \beta_0 + \beta_1 x$$

- when x is zero, the log of the expected value of the response equals β_0
- For every one-unit increase in x, the log of the expected value of the response increases by β_1
- ... But comprehending the effect of x on the log of the expected value is difficult, especially for a layperson

Interpretation: numeric explanatory variables

Poisson regression

$$\log(\mu) = \beta_0 + \beta_1 x$$
$$\mu = e^{\beta_0 + \beta_1 x}$$
$$= e^{\beta_0}(e^{\beta_1})^x$$

- When $x = 0$

$$\mu = e^{\beta_0}(e^{\beta_1})^0$$
$$= e^{\beta_0}$$

- When $x = 1$

$$\mu = e^{\beta_0}(e^{\beta_1})^1$$
$$= e^{\beta_0} e^{\beta_1}$$

Interpretation: numeric explanatory variables

Poisson regression

$$\log(\mu) = \beta_0 + \beta_1 x$$
$$\mu = e^{\beta_0 + \beta_1 x}$$
$$= e^{\beta_0}(e^{\beta_1})^x$$

- When $x = 2$

$$\mu = e^{\beta_0}(e^{\beta_1})^2$$
$$= e^{\beta_0} e^{\beta_1} e^{\beta_1}$$

- When $x = 3$

$$\mu = e^{\beta_0}(e^{\beta_1})^3$$
$$= e^{\beta_0} e^{\beta_1} e^{\beta_1} e^{\beta_1}$$

Interpretation: numeric explanatory variables

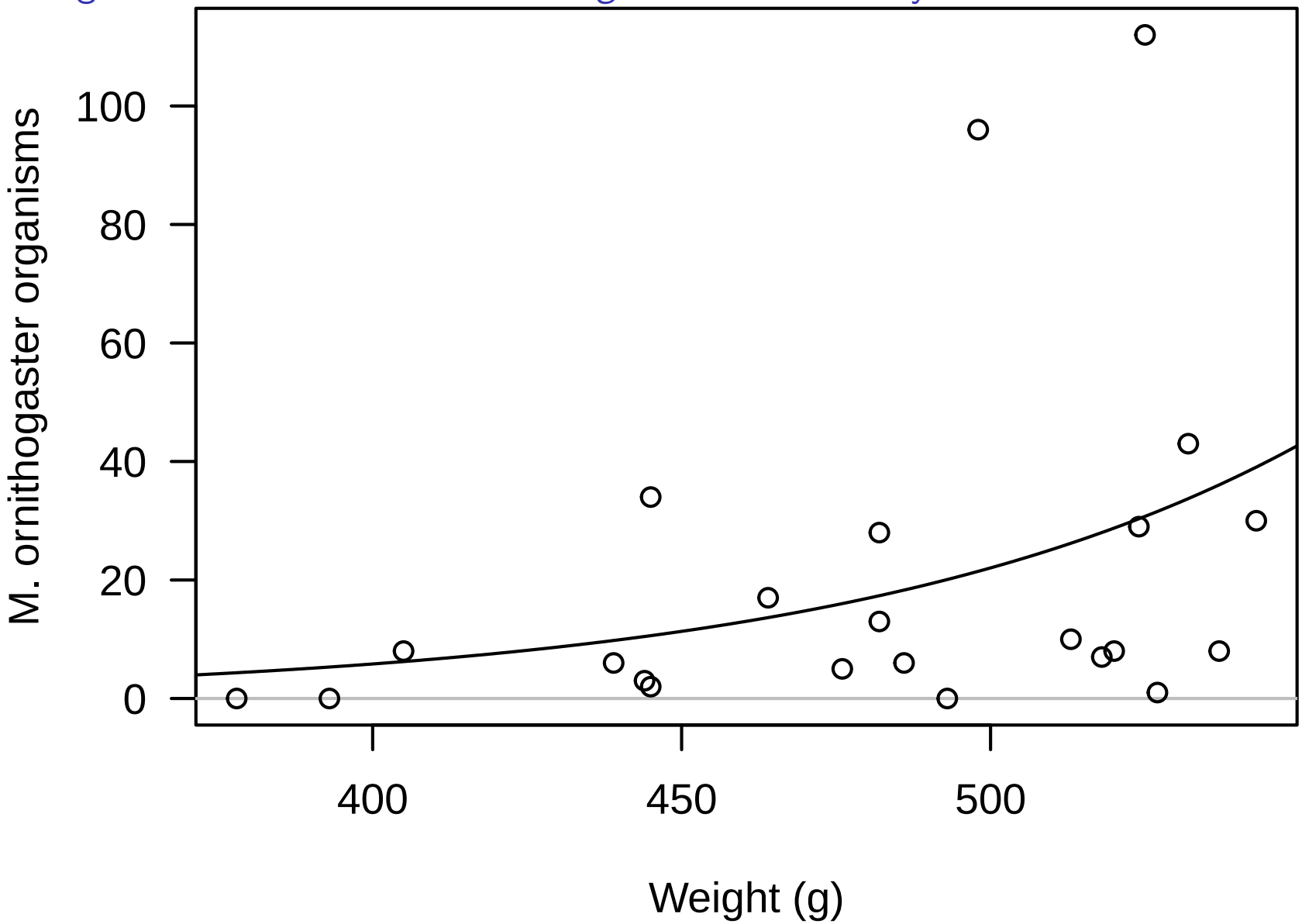
Poisson regression

$$\begin{aligned} \log(\mu) &= \beta_0 + \beta_1 x \\ \mu &= e^{\beta_0 + \beta_1 x} \\ &= e^{\beta_0}(e^{\beta_1})^x \end{aligned}$$

- when x is zero, the expected value equals $\exp(\beta_0)$
- For every one-unit increase in x, the expected value of the response is multiplied by $\exp(\beta_1)$
- For every ten-unit increase in x, the expected value of the response is multiplied by $\exp(10\beta_1)$
- For every n-unit increase in x, the expected value of the response is multiplied by $\exp(n\beta_1)$

Interpretation: numeric explanatory variables

Poisson regression: *Macrorhabdus ornithogaster* chicken analysis



Interpretation: numeric explanatory variables

Poisson regression: *Macrorhabdus ornithogaster* chicken analysis

```
chickens.fit <- glm(mo ~ weight, family = "poisson")
summary(chickens.fit)

## Call:
## glm(formula = mo ~ weight, family = "poisson")
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.558206   0.657140  -5.415 6.14e-08 ***
## weight       0.013302   0.001301  10.228  < 2e-16 ***
## ---
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 683.02  on 22  degrees of freedom
## Residual deviance: 554.49  on 21  degrees of freedom
```

Interpretation: numeric explanatory variables

Poisson regression: *Macrorhabdus ornithogaster* chicken analysis

```
coef(chickens.fit)

## (Intercept)      weight
## -3.55820556  0.01330221
```

So our fitted relationship is

$$\begin{aligned}\log(\hat{\mu}_i) &= \hat{\beta}_0 + \hat{\beta}_1 x_i \\ &= -3.56 + 0.013 x_i \\ \hat{\mu}_i &= \exp(-3.56 + 0.013 x_i),\end{aligned}$$

where x_i is the weight of the ith chicken.

Interpretation: numeric explanatory variables

Poisson regression: *Macrorhabdus ornithogaster* chicken analysis

```
coef(chickens.fit)

## (Intercept)      weight
## -3.55820556  0.01330221
```

We can make the following statements:

- We estimate that the log of the expected number of *Macrorhabdus ornithogaster* organisms in a faecal sample from a chicken that weighs 0 g is -3.6
  - Interpreting the intercept isn't often sensible
- We estimate that, for every 1 g increase in the weight of a chicken, the log of the expected number of *Macrorhabdus ornithogaster* organisms increases by 0.013.

Interpretation: numeric explanatory variables

Poisson regression: *Macrorhabdus ornithogaster* chicken analysis

```
exp(coef(chickens.fit))

## (Intercept)      weight
##   0.0284899   1.0133911
```

We can make the following statements:

- We estimate that the expected number of *Macrorhabdus ornithogaster* organisms in a faecal sample from a chicken that weighs 0 g is 0.028
  - Interpreting the intercept isn't often sensible
- We estimate that, for every 1 g increase in the weight of a chicken, the expected number of *Macrorhabdus ornithogaster* organisms in its faecal sample is multiplied by 1.013.

Interpretation: numeric explanatory variables

Poisson regression: *Macrorhabdus ornithogaster* chicken analysis

```
chickens.fit <- glm(mo ~ weight, family = "poisson")
exp(100*coef(chickens.fit)[2])

##   weight
## 3.781881
```

We can make the following statements:

- We estimate that the expected number of *Macrorhabdus ornithogaster* organisms in a faecal sample from a chicken that weighs 0 g is 0.028
  - Interpreting the intercept isn't often sensible
- We estimate that, for every 100 g increase in the weight of a chicken, the expected number of *Macrorhabdus ornithogaster* organisms in its faecal sample is multiplied by 3.78.

Interpretation: numeric explanatory variables

Poisson regression: *Macrorhabdus ornithogaster* chicken analysis

For a multiplicative interpretation of the effect of a one-unit change:

- We use $\exp(\beta_1)$

For a percentage-change interpretation:

- We use $100[\exp(\beta_1) - 1]$

```
100*(exp(coef(chickens.fit)[2]) - 1)

##    weight
## 1.339108
```

- We estimate that, for every 1 g increase in the weight of a chicken, the expected number of *Macrorhabdus ornithogaster* organisms in its faecal sample increases by 1.3%.

Interpretation: numeric explanatory variables

Poisson regression: *Macrorhabdus ornithogaster* chicken analysis

For a multiplicative interpretation of the effect of a 100-unit change:

- We use $\exp(100\beta_1)$

For a percentage-change interpretation:

- We use $100[\exp(100\beta_1) - 1]$

```
100*(exp(100*coef(chickens.fit)[2]) - 1)

##   weight
## 278.1881
```

- We estimate that, for every 100 g increase in the weight of a chicken, the expected number of *Macrorhabdus ornithogaster* organisms in its faecal sample increases by 278%.

Interpretation: numeric explanatory variables

Poisson regression: *Macrorhabdus ornithogaster* chicken analysis

Confidence intervals are better:

```
100*(exp(100*confint(chickens.fit)[2, ]) - 1)

## Waiting for profiling to be done...

##     2.5 %    97.5 %
## 194.5422 390.5354
```

- We estimate that, for every 100 g increase in the weight of a chicken, the expected number of *Macrorhabdus ornithogaster* organisms in its faecal sample increases by between 195 and 391%.

Interpretation: numeric explanatory variables

Logistic regression: Probability, odds, and log-odds recap

Probability	Odds	Log-Odds
p	$\frac{p}{1-p}$	$\log\left(\frac{p}{1-p}\right)$
0.50	1	0
0.75	3	≈ 1.1
0.25	≈ 0.33	≈ -1.1
0.9	9	≈ 2.2
0.1	≈ 0.11	≈ -2.2
0.9999	9999	≈ 9.2
0.0001	≈ 0.0001	≈ -9.2
1	∞	∞
0	0	$-\infty$

Interpretation: numeric explanatory variables

Logistic regression: Probability, odds, and log-odds recap

Some results:

- When p is small, the probability and the odds are similar
- Probability is always between 0 and 1
- Odds are always between 0 and ∞
- Log-odds are always between $-\infty$ and ∞

Interpretation: numeric explanatory variables

Logistic regression

For logistic regression we have

$$\begin{aligned} \text{logit}(p) &= \beta_0 + \beta_1 x \\ \log\left(\frac{p}{1-p}\right) &= \beta_0 + \beta_1 x \\ \log(\text{odds}) &= \beta_0 + \beta_1 x \end{aligned}$$

- when x is zero, the log-odds of success are equal to β_o
- For every one-unit increase in x, the log-odds of success increase by β_1
- ... But comprehending the effect of x on the log of the odds of success is difficult, especially for a layperson

Interpretation: numeric explanatory variables

Logistic regression

$$\begin{aligned}\log(\text{odds}) &= \beta_0 + \beta_1 x \\ \text{odds} &= e^{\beta_0+\beta_1 x} \\ &= e^{\beta_0}(e^{\beta_1})^x\end{aligned}$$

- When $x = 0$

$$\begin{aligned}\text{odds} &= e^{\beta_0}(e^{\beta_1})^0 \\ &= e^{\beta_0}\end{aligned}$$

- When $x = 1$

$$\begin{aligned}\text{odds} &= e^{\beta_0}(e^{\beta_1})^1 \\ &= e^{\beta_0}e^{\beta_1}\end{aligned}$$

Interpretation: numeric explanatory variables

Logistic regression

$$\log(\text{odds}) = \beta_0 + \beta_1 x$$

$$\begin{aligned} \text{odds} &= e^{\beta_0 + \beta_1 x} \\ &= e^{\beta_0}(e^{\beta_1})^x \end{aligned}$$

- When $x = 2$

$$\begin{aligned} \text{odds} &= e^{\beta_0}(e^{\beta_1})^2 \\ &= e^{\beta_0} e^{\beta_1} e^{\beta_1} \end{aligned}$$

- When $x = 3$

$$\begin{aligned} \text{odds} &= e^{\beta_0}(e^{\beta_1})^3 \\ &= e^{\beta_0} e^{\beta_1} e^{\beta_1} e^{\beta_1} \end{aligned}$$

Interpretation: numeric explanatory variables

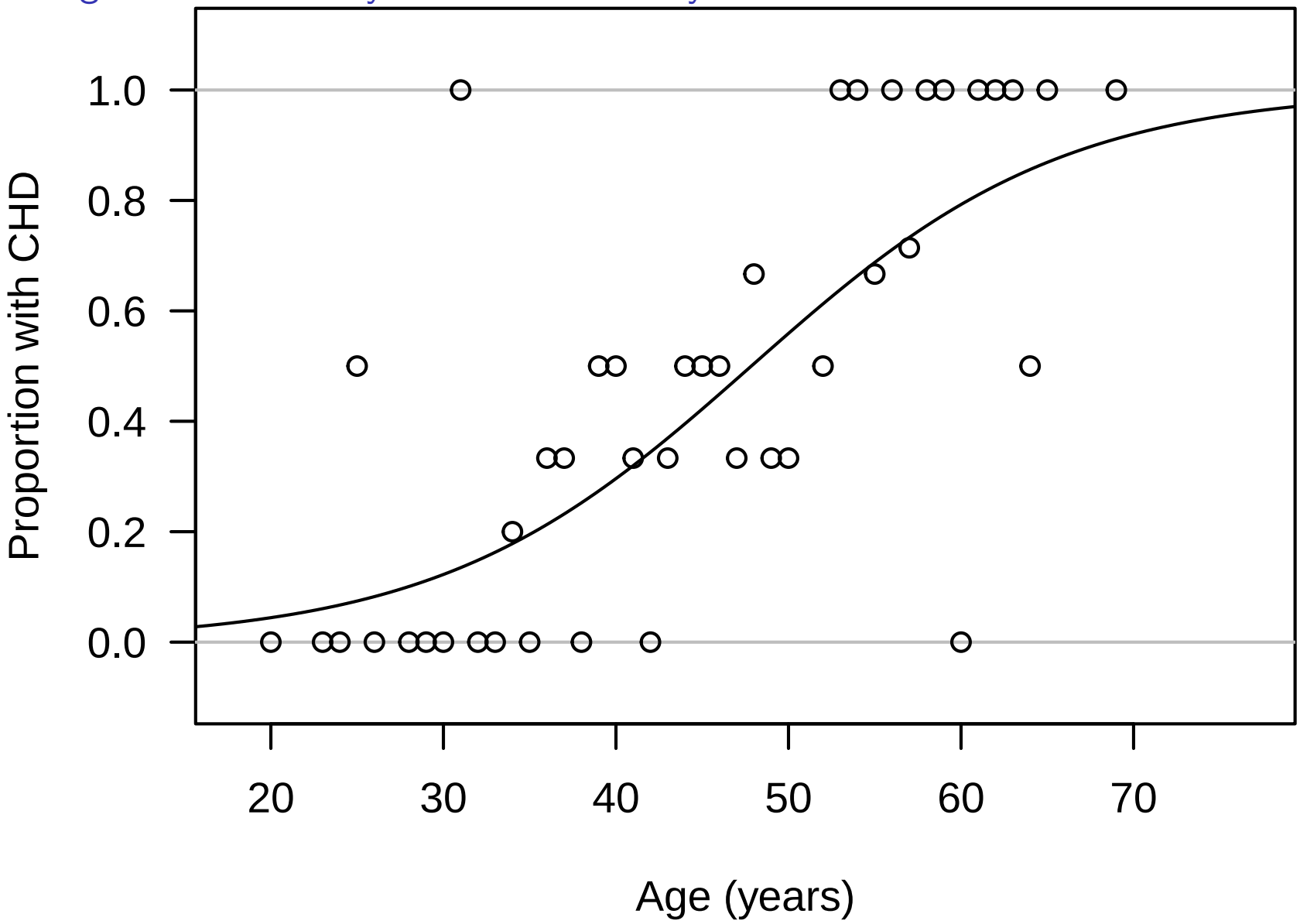
Logistic regression

$$\begin{aligned} \log(\text{odds}) &= \beta_0 + \beta_1 x \\ \text{odds} &= e^{\beta_0 + \beta_1 x} \\ &= e^{\beta_0}(e^{\beta_1})^x \end{aligned}$$

- when x is zero, the odds of success equal $\exp(\beta_0)$
- For every one-unit increase in x, the odds of success are multiplied by $\exp(\beta_1)$
- For every ten-unit increase in x, the odds of success are multiplied by $\exp(10\beta_1)$
- For every n-unit increase in x, the odds of success are multiplied by $\exp(n\beta_1)$

Interpretation: numeric explanatory variables

Logistic regression: Coronary heart disease analysis



Interpretation: numeric explanatory variables

Logistic regression: Coronary heart disease analysis

```
chd.fit <- glm(cbind(y, n - y) ~ age, family = "binomial")
summary(chd.fit)

## Call:
## glm(formula = cbind(y, n - y) ~ age, family = "binomial")
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.27844    1.13053  -4.669 3.03e-06 ***
## age          0.11032    0.02402   4.593 4.36e-06 ***
## ---
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 63.958  on 42  degrees of freedom
## Residual deviance: 34.976  on 41  degrees of freedom
```

Interpretation: numeric explanatory variables

Logistic regression: Coronary heart disease analysis

```
coef(chd.fit)

## (Intercept)         age
##  -5.2784444   0.1103208
```

We can make the following statements:

- We estimate that the log-odds of a newborn baby having coronary heart disease are -5.3
  - Interpreting the intercept isn't often sensible
- We estimate that, for every 1-year increase in age, the log-odds of having coronary heart disease increase by 0.11.

Interpretation: numeric explanatory variables

Logistic regression: Coronary heart disease analysis

```
exp(coef(chd.fit))

## (Intercept)         age
## 0.005100359 1.116636221
```

We can make the following statements:

- We estimate that the odds of a newborn baby having coronary heart disease are 0.005
  - Interpreting the intercept isn't often sensible
- We estimate that, for every 1-year increase in age, the odds of having coronary heart disease are multiplied by 1.12.

Interpretation: numeric explanatory variables

Logistic regression: Coronary heart disease analysis

```
exp(10*coef(chd.fit)[2])

##      age
## 3.013819
```

We can make the following statements:

- We estimate that the odds of a newborn baby having coronary heart disease are 0.005
  - Interpreting the intercept isn't often sensible
- We estimate that, for every 10-year increase in age, the odds of having coronary heart disease are multiplied by 3.0.

Interpretation: numeric explanatory variables

Logistic regression: Coronary heart disease analysis

```
exp(10*confint(chd.fit)[2, ])

## Waiting for profiling to be done...

##    2.5 %   97.5 %
## 1.942264 5.017448
```

We can make the following statements:

- We estimate that the odds of a newborn baby having coronary heart disease are 0.005
  - Interpreting the intercept isn't often sensible
- We estimate that, for every 10-year increase in age, the odds of having coronary heart disease are multiplied by between 1.9 and 5.0.

Interpretation: numeric explanatory variables

Logistic regression: Coronary heart disease analysis

```
100*(exp(10*coef(chd.fit)[2]) - 1)

##       age
## 201.3819
```

We can make the following statements:

- We estimate that the odds of a newborn baby having coronary heart disease are 0.005
  - Interpreting the intercept isn't often sensible
- We estimate that, for every 10-year increase in age, the odds of having coronary heart disease increase by 201%.

Interpretation: numeric explanatory variables

Logistic regression: Coronary heart disease analysis

```
100*(exp(10*confint(chd.fit)[2, ]) - 1)

## Waiting for profiling to be done...

##     2.5 %    97.5 %
##  94.22639 401.74481
```

We can make the following statements:

- We estimate that the odds of a newborn baby having coronary heart disease are 0.005
  - Interpreting the intercept isn't often sensible
- We estimate that, for every 10-year increase in age, the odds of having coronary heart disease increase by between 94 and 402%.

Interpretation: numeric explanatory variables

Logistic regression: Coronary heart disease analysis

We have answered our questions of interest:

- Does age play a role in the occcurrence of coronary heart disease?
  - We have evidence to suggest it does; it is not plausible that an increase in age does not affect the average arithmetic score, because it is not plausible that $\beta_1 = 0$.
- If so, how?
  - We estimate that, for every 10-year increase in age, the odds of having coronary heart disease increase by between 94 and 402%.

Alternatively, the first question could be achieved by a formal hypothesis test, rather than by observing whether or not the confidence interval includes zero.

Interpretation: numeric explanatory variables

Logistic regression: Coronary heart disease analysis

```
summary(chd.fit)

## Call:
## glm(formula = cbind(y, n - y) ~ age, family = "binomial")
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.27844    1.13053  -4.669 3.03e-06 ***
## age          0.11032    0.02402   4.593 4.36e-06 ***
## ---
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 63.958  on 42  degrees of freedom
## Residual deviance: 34.976  on 41  degrees of freedom
```

Interpretation: numeric explanatory variables

Logistic regression: Coronary heart disease analysis

Same procedure as the linear regression model.

Each p-value in the rightmost column tests the null hypothesis that the true parameter value for that row is equal to zero.

The null hypothesis we wish to test is

$$H_0 : \beta_1 = 0,$$

because under this hypothesis there is no relationship between age and the probability of coronary heart disease.

We have a very small p-value, so there is strong evidence against this null hypothesis; there appears to be a relationship between age and the probability of having coronary heart disease.

Interpretation: numeric explanatory variables

A summary

Linear regression:

- For every one-unit increase in x, the expected value of Y increases by β_1

Poisson regression:

- For every one-unit increase in x, the expected value of Y is multiplied by $\exp(\beta_1)$
- For every one-unit increase in x, the expected value of Y increases by $100 \times [\exp(\beta_1) - 1]\%$

Logistic regression:

- For every one-unit increase in x, the odds of success are multiplied by $\exp(\beta_1)$
- For every one-unit increase in x, the odds of success increase by $100 \times [\exp(\beta_1) - 1]\%$

Interpretation: Factors

To fit a model with k factors, we include $k-1$ dummy variables. For example, if we have a factor with levels A, B, C, and D, then

$$g(\theta_i) = \beta_0 + \beta_1 b_i + \beta_2 c_i + \beta_3 d_i,$$

where

- $b_i = 1$ if the ith observation has level B, otherwise $b_i = 0$;
- $c_i = 1$ if the ith observation has level C, otherwise $c_i = 0$; and
- $d_i = 1$ if the ith observation has level D, otherwise $d_i = 0$.

The level without a dummy variable is known as the baseline level.

Interpretation: Factors

If the ith observation has level A:

$$g(\theta_i) = \beta_0 + \beta_1 \times 0 + \beta_2 \times 0 + \beta_3 \times 0$$
$$g(\theta_i) = \beta_0$$

If the ith observation has level B:

$$g(\theta_i) = \beta_0 + \beta_1 \times 1 + \beta_2 \times 0 + \beta_3 \times 0$$
$$g(\theta_i) = \beta_0 + \beta_1$$

If the ith observation has level C:

$$g(\theta_i) = \beta_0 + \beta_1 \times 0 + \beta_2 \times 1 + \beta_3 \times 0$$
$$g(\theta_i) = \beta_0 + \beta_2$$

If the ith observation has level D:

$$g(\theta_i) = \beta_0 + \beta_1 \times 0 + \beta_2 \times 0 + \beta_3 \times 1$$
$$g(\theta_i) = \beta_0 + \beta_3$$

Interpretation: Factors

So,

- β_0 is the value of $g(\theta)$ for observations with level, A;
- β_1 is the difference in $g(\theta)$ between observations with level B and those with level A;
- β_2 is the difference in $g(\theta)$ between observations with level C and those with level A; and
- β_3 is the difference in $g(\theta)$ between observations with level D and those with level A.

In general,

- β_0 is the value of $g(\theta)$ for observations with the baseline level, and
- The coefficient of the dummy variable for the jth level is the difference in $g(\theta)$ between observations with the jth level and those with the baseline level.

Interpretation: Factors

We would rather make statements about θ than $g(\theta)$:

- For Poisson regression, we would rather make statements about μ than about $\log(\mu)$
- For logistic regression, we would rather make statements about the odds of success than the log-odds of success.

We achieve this in the same way we did with numeric explanatory variables:

- For Poisson regression, we exponentiate coefficients to give multiplicative effects on the expected value.
- For logistic regression, we exponentiate coefficients to give multiplicative effects on the odds of success.

We can also do the same calculations to obtain multiplicative effects in terms of percentage change.

Interpretation: Factors

Macrorhabdus ornithogaster chicken data

M. ornithogaster is a bacterial disease that affects birds. The drug Amphotericin B is often used for treatment, but its efficacy has not yet been established. It was of interest to determine if the Amphotericin B dose is related to the number of *M. ornithogaster* organisms shedded in chickens' faeces, while correcting for the size of the chicken. An experiment was conducted on 23 infected chickens.

The data frame contains the following variables:

`mo`: The number of *M. ornithogaster* in a sample of ten faecal slides from the chicken

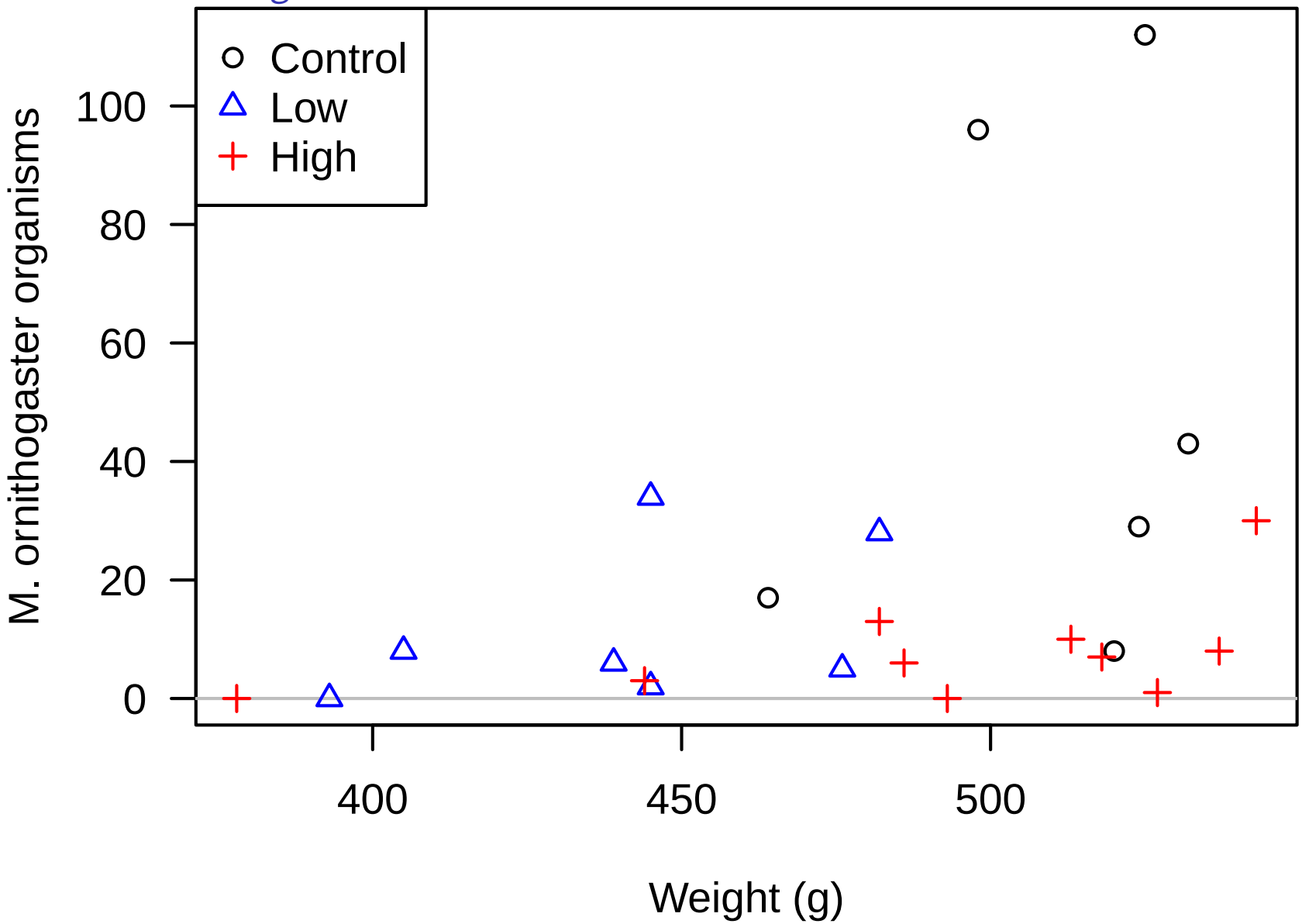
`dose`: The dose of Amphotericin B given to the chicken; either `control`, `low`, or `high`

`weight`: The weight of the chicken in grams

Let's investigate the relationship between `dose` and `mo`

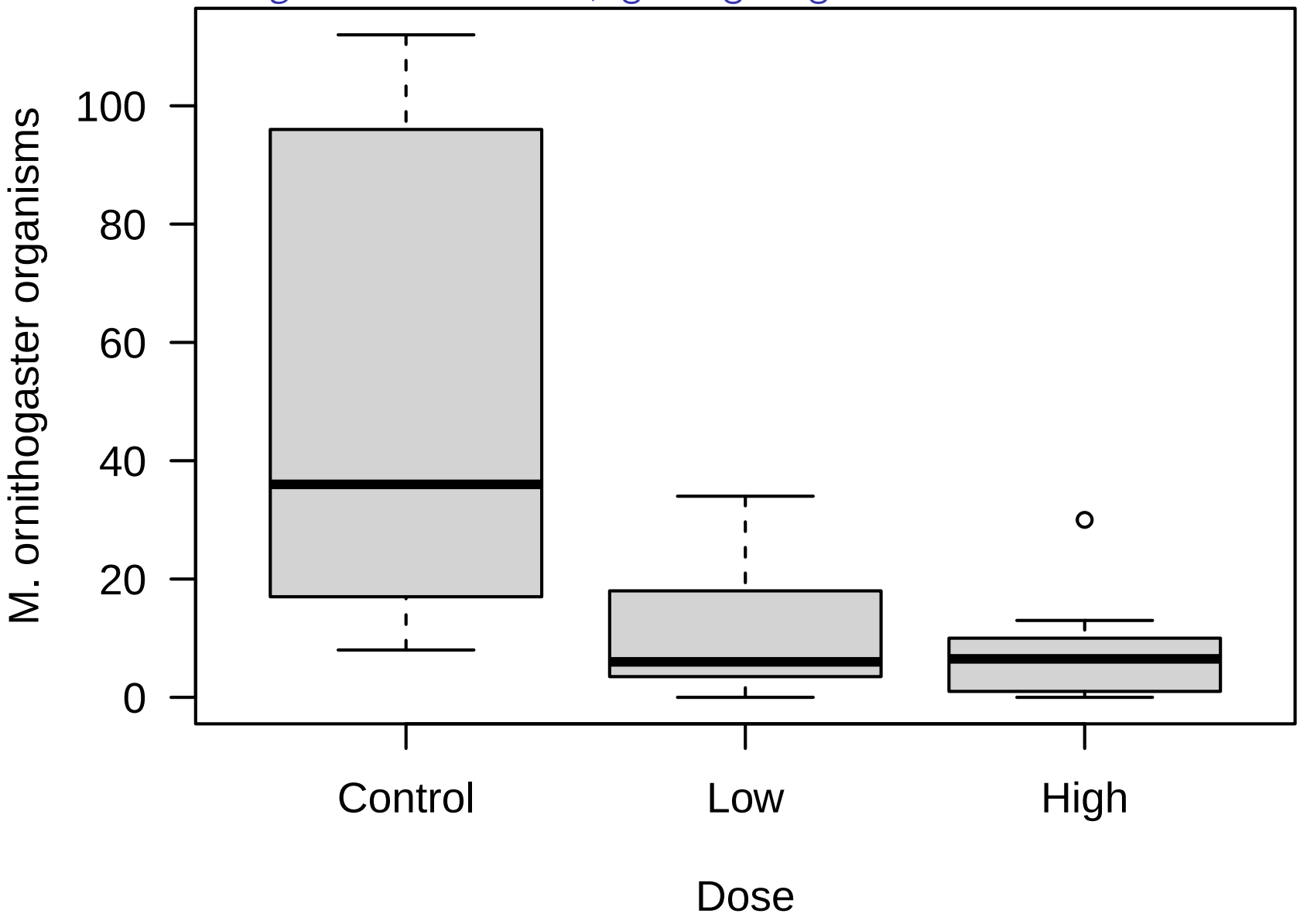
Interpretation: Factors

Macrorhabdus ornithogaster chicken data



Interpretation: Factors

Macrorhabdus ornithogaster chicken data, ignoring weight



Interpretation: Factors

Macrorhabdus ornithogaster chicken analysis

```
chickens.dose.fit <- glm(mo ~ dose, family = "poisson")
summary(chickens.dose.fit)

## Call:
## glm(formula = mo ~ dose, family = "poisson")
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.92855    0.05726   68.61   <2e-16 ***
## doseHigh    -1.87443    0.12688  -14.77   <2e-16 ***
## doseLow     -1.45562    0.12380  -11.76   <2e-16 ***
## ---
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 683.02  on 22  degrees of freedom
## Residual deviance: 359.75  on 20  degrees of freedom
```

Interpretation: Factors

Macrorhabdus ornithogaster chicken analysis

We have fitted the following model:

$$\log(\mu_i) = \beta_0 + \beta_1 h_i + \beta_2 l_i$$

$$Y_i \sim \text{Poisson}(Y_i)$$

But how do we interpret the coefficients β_0, β_1, and β_2?

Interpretation: Factors

Macrorhabdus ornithogaster chicken analysis

```
coef(chickens.dose.fit)

## (Intercept)    doseHigh     doseLow
##    3.928552   -1.874429   -1.455622
```

We estimate that the log of the expected number of *Macrorhabdus ornithogaster* organisms in a faecal sample from

- a chicken in the control group is equal to 3.93
- a chicken in the high-dose group is 1.87 lower than for those in the control group
- a chicken in the low-dose group is 1.46 lower than for those in the control group

Interpretation: Factors

Macrorhabdus ornithogaster chicken analysis

```
exp(coef(chickens.dose.fit))
```

```
## (Intercept)    doseHigh     doseLow
##   50.8333333   0.1534426   0.2332553
```

We estimate that the expected number of *Macrorhabdus ornithogaster* organisms in a faecal sample from

- a chicken in the control group is equal to 50.8
- a chicken in the high-dose group is 0.153 times that of those in the control group
- a chicken in the low-dose group is 0.233 times that of those in the control group

Interpretation: Factors

Macrorhabdus ornithogaster chicken analysis

```
100*(exp(coef(chickens.dose.fit)[c(2, 3)]) - 1)
```

```
##  doseHigh   doseLow
## -84.65574 -76.67447
```

We estimate that the expected number of *Macrorhabdus ornithogaster* organisms in a faecal sample from

- a chicken in the control group is equal to 50.8
- a chicken in the high-dose group is 84.7% lower than for those in the control group
- a chicken in the low-dose group is 76.7% lower than for those in the control group

Interpretation: Factors

Hypothesis tests

Each p-value in the `summary()` table tests the null hypothesis that the corresponding coefficient is equal to zero.

```
summary(chickens.dose.fit)

## Call:
## glm(formula = mo ~ dose, family = "poisson")
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.92855    0.05726   68.61   <2e-16 ***
## doseHigh    -1.87443    0.12688  -14.77   <2e-16 ***
## doseLow     -1.45562    0.12380  -11.76   <2e-16 ***
## ---
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 683.02  on 22  degrees of freedom
## Residual deviance: 359.75  on 20  degrees of freedom
```

Interpretation: Factors

Hypothesis tests

Each p-value in the `summary()` table tests the null hypothesis that the corresponding coefficient is equal to zero.

So we can use this table to test the following null hypotheses:

- $H_0 : \beta_1 = 0$. There is no difference between chickens in the control group and chickens given a high dose.
- $H_0 : \beta_2 = 0$. There is no difference between chickens in the control group and chickens given a low dose.

However, we cannot use it to test the following null hypotheses:

- $H_0 : \beta_1 = \beta_2$. There is no difference between chickens given a low dose and chickens given a high dose.
- $H_0 : \beta_1 = \beta_2 = 0$. There are no differences between any of the levels.

Interpretation: Factors

Hypothesis tests

We can use the `anova()` function to test the hypothesis

- $H_0 : \beta_1 = \beta_2 = 0$. There are no differences between any of the levels.

```
anova(chickens.dose.fit, test = "Chisq")

## Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: mo
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                    22     683.02
## dose  2   323.27         20     359.75 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpretation: Factors

Hypothesis tests

To test for differences from a different level, we can respecify the baseline level:

```
dose <- factor(dose, levels = c("High", "Low", "Control"))
chickens.dose.fit <- glm(mo ~ dose, family = "poisson")
summary(chickens.dose.fit)

## Call:
## glm(formula = mo ~ dose, family = "poisson")
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   2.0541     0.1132  18.142  < 2e-16 ***
## doseLow       0.4188     0.1577   2.656  0.00791 **
## doseControl   1.8744     0.1269  14.773  < 2e-16 ***
## ---
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 683.02  on 22  degrees of freedom
## Residual deviance: 359.75  on 20  degrees of freedom
```

Interpretation: Factors

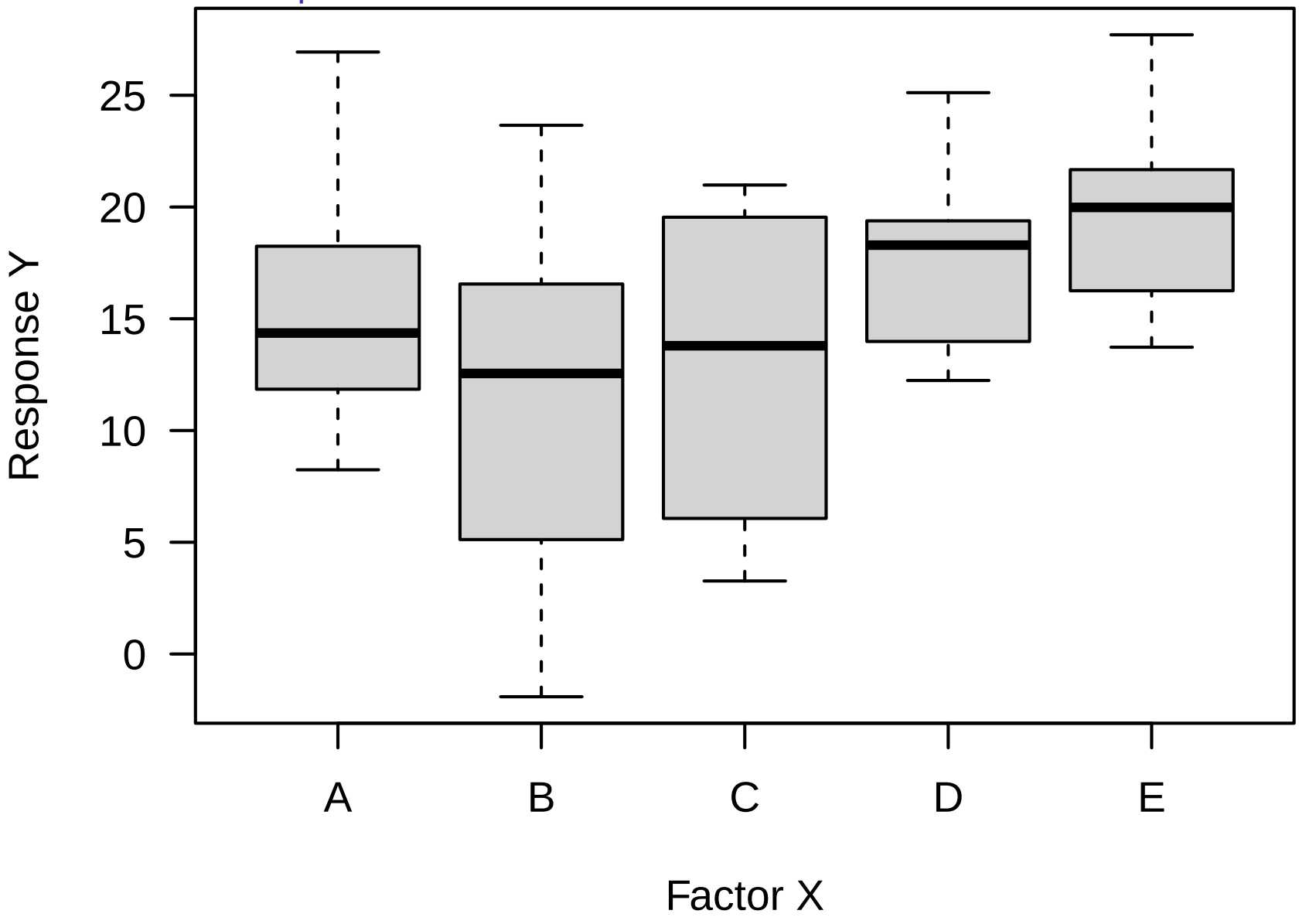
Hypothesis tests

A word of warning:

- If we wish to determine if a factor is related to the response variable, we should use the `anova()` function, rather than relying on the *p*-values in the `summary()` table.
- It is possible for all *p*-values in the `summary()` to be large, even though there is evidence of differences between some of the levels of the factor.
- This is because the `summary()` table only compares the baseline level to others. There may be no evidence of differences from the baseline level, but there may be evidence of differences between non-baseline levels.

Interpretation: Factors

Hypothesis tests: Example data



Interpretation: Factors

Hypothesis tests: Example data analysis

Large p-values in `summary()`...

```
example.fit <- lm(y ~ x)
summary(example.fit)

## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   15.323      1.885   8.131 2.22e-10 ***
## xB            -3.682      2.665  -1.382    0.174
## xC            -2.468      2.665  -0.926    0.359
## xD             2.613      2.665   0.980    0.332
## xE             4.371      2.665   1.640    0.108
## ---
## Residual standard error: 5.959 on 45 degrees of freedom
## Multiple R-squared:  0.2214,Adjusted R-squared:  0.1522
## F-statistic: 3.199 on 4 and 45 DF,  p-value: 0.02145
```

Interpretation: Factors

Hypothesis tests: Example data analysis

... But a small p-value in `anova()`...

```
anova(example.fit)

## Analysis of Variance Table
##
## Response: y
##           Df  Sum Sq Mean Sq F value  Pr(>F)
## x          4  454.44 113.609  3.1989 0.02145 *
## Residuals 45 1598.19  35.515
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '
```

Interpretation: Factors

Hypothesis tests: Example data analysis

... Due to evidence of a difference between non-baseline levels.

```
x <- factor(x, levels = c("B", "A", "C", "D", "E"))
summary(lm(y ~ x))

## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   11.641      1.885   6.177  1.7e-07 ***
## xA             3.682      2.665   1.382  0.17392
## xC             1.214      2.665   0.456  0.65090
## xD             6.295      2.665   2.362  0.02257 *
## xE             8.053      2.665   3.022  0.00414 **
## ---
## Residual standard error: 5.959 on 45 degrees of freedom
## Multiple R-squared:  0.2214,Adjusted R-squared:  0.1522
## F-statistic: 3.199 on 4 and 45 DF,  p-value: 0.02145
```

Interpretation: Multiple explanatory variables

We wish to determine the effectiveness of Amphotericin B, after accounting for the role that weight plays:

```
chickens.full.fit <- glm(mo ~ dose + weight, family = "poisson")
anova(chickens.full.fit, test = "Chisq")

## Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: mo
##
## Terms added sequentially (first to last)
##
##
##        Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                      22     683.02
## dose    2   323.27        20     359.75 < 2.2e-16 ***
## weight  1    49.14        19     310.61 2.382e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpretation: Multiple explanatory variables

```
summary(chickens.full.fit)

## Call:
## glm(formula = mo ~ dose + weight, family = "poisson")
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.578171   1.015763  -2.538 0.011144 *
## doseHigh    -1.743623   0.127181 -13.710  < 2e-16 ***
## doseLow     -0.604736   0.177085  -3.415 0.000638 ***
## weight       0.012670   0.001964   6.450 1.12e-10 ***
## ---
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 683.02  on 22  degrees of freedom
## Residual deviance: 310.61  on 19  degrees of freedom
```

Interpretation: Multiple explanatory variables

Interpretation is the same as for a single-variable model, but must acknowledge that the statements only hold if all other variables in the model are kept constant.

```
100*(exp(confint(chickens.full.fit)[c(2, 3), ]) - 1)

## Waiting for profiling to be done...

##                2.5 %    97.5 %
## doseHigh -86.45792 -77.68954
## doseLow  -61.45552 -22.80070
```

Holding the weight of a chicken constant, we estimate that the number of *Macrorhabdus ornithogaster* organisms in a faecal sample from a chicken

- in the high-dose group is between 78 and 86% lower than a chicken in the control group.
- in the low-dose group is between 23 and 61% lower than a chicken in the control group.

Interpretation: Interactions

We use interaction effects when we think the effect of one explanatory variable on the response depends on other explanatory variables.

Some examples:

- If it is rush hour, my commute to work is quicker on the bus due to bus lanes, but if it is not rush hour then it is faster to drive.
  - Interaction between time of day and mode of transport affects commuting time.
- Ice cream is tastier with chocolate sauce than tomato sauce, but you'd rather have tomato sauce on a sausage.
  - Interaction between food type and condiment affects tastiness.

Interpretation: Interactions

If the answer to the question "what is the effect of a variable A on the response" is "it depends on variable B", then we should fit an interaction between variables A and B.

What is the effect on commuting time of taking a bus rather than driving to work?

- It depends: is it rush hour or not?

What is the effect on food tastiness of adding tomato sauce rather than chocolate sauce?

- It depends: what am I eating?

Does the effect of treatment depend on the weight of a chicken?

Interpretation: Interactions

We can fit an interaction by using an asterisk in the model formula.

```
chickens.int.fit <- glm(mo ~ weight*dose, family = "poisson")
```

This results in the following model:

$$\log(\mu_i) = \beta_0 + \beta_1 w_i + \beta_2 h_i + \beta_3 l_i + \beta_4 w_i h_i + \beta_5 w_i l_i,$$

where w_i is the weight of the ith chicken, and h_i and l_i are dummy variables for the high- and low-dose factor levels, respectively.

We have two kinds of effects:

- Main effects, which only involve a single variable, and
- Interaction effects, which involve more than one variable.

We will have an interaction term for every possible pair of terms that can be enumerated by taking a main effect from each variable.

Interpretation: Interactions

```
summary(chickens.int.fit)

## Call:
## glm(formula = mo ~ weight * dose, family = "poisson")
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)       0.450933   1.392638   0.324  0.74609
## weight            0.006789   0.002708   2.507  0.01217 *
## doseHigh         -8.320869   2.602745  -3.197  0.00139 **
## doseLow          -5.565791   2.315474  -2.404  0.01623 *
## weight:doseHigh   0.012771   0.005032   2.538  0.01115 *
## weight:doseLow    0.010141   0.004878   2.079  0.03764 *
## ---
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 683.02  on 22  degrees of freedom
## Residual deviance: 302.06  on 17  degrees of freedom
```

Interpretation: Interactions

Main effects

We can interpret the main effects in the same way as we can for models that do not have an interaction, but these interpretations only hold

- At the baseline level of the other variable, if the other variable is a factor; or
- When the other variable is equal to zero, if the other variable is numeric. This is not necessarily sensible to interpret.

```
exp(100*confint(chickens.int.fit)[2, ])
```

```
##     2.5 %    97.5 %
## 1.175891 3.404126
```

For chickens in the control group, we estimate that, for every 100 g increase in the weight of a chicken, the expected number of *Macrorhabdus ornithogaster* organisms in its faecal sample is multiplied by between 1.18 and 3.40.

Interpretation: Interactions

Main effects

We can interpret the main effects in the same way as we can for models that do not have an interaction, but these interpretations only hold

- At the baseline level of the other variable, if the other variable is a factor; or
- When the other variable is equal to zero, if the other variable is numeric. This is not necessarily sensible to interpret.

```
100*(exp(confint(chickens.int.fit)[3, ]) - 1)
```

```
##      2.5 %     97.5 %
## -99.99988 -96.67849
```

For chickens that weigh 0 g, we estimate that being in the high-dose treatment rather than the control is associated with a decrease in the expected number of *Macrorhabdus ornithogaster* organisms in its faecal sample of between 96.68 and 100.00%

Interpretation: Interactions

Main effects

We can interpret the main effects in the same way as we can for models that do not have an interaction, but these interpretations only hold

- At the baseline level of the other variable, if the other variable is a factor; or
- When the other variable is equal to zero, if the other variable is numeric. This is not necessarily sensible to interpret.

```
100*(exp(confint(chickens.int.fit)[4, ]) - 1)
```

```
##      2.5 %     97.5 %
## -99.99615 -65.66166
```

For chickens that weigh 0 g, we estimate that being in the low-dose treatment rather than the control is associated with a decrease in the expected number of *Macrorhabdus ornithogaster* organisms in its faecal sample of between 65.66 and 100.00%

Interpretation: Interactions

Interaction effects

An interaction effect, on the other hand, measures how the effect of one variable changes depending on the other. Let's consider the effect of a one-unit increase in weight both

- For chickens in the control group (the baseline), and
- For chickens in the high-dose group.

$$\log(\mu_i) = \beta_0 + \beta_1 w_i + \beta_2 h_i + \beta_3 l_i + \beta_4 w_i h_i + \beta_5 w_i l_i,$$

For chickens in the control group:

$$\log(\mu_i) = \beta_0 + \beta_1 w_i.$$

For chickens in the high-dose group:

$$\begin{aligned} \log(\mu_i) &= \beta_0 + \beta_1 w_i + \beta_2 + \beta_4 w_i \\ &= (\beta_0 + \beta_2) + (\beta_1 + \beta_4) w_i \end{aligned}$$

Interpretation: Interactions

Interaction effects

For chickens in the control group, the relationship between weight and the log of the expected value of the response has

- An intercept of β_0
- A slope of β_1

For chickens in the high-dose group, the relationship between weight and the log of the expected value of the response has

- An intercept of $(\beta_0 + \beta_2)$
- A slope of $(\beta_1 + \beta_4)$

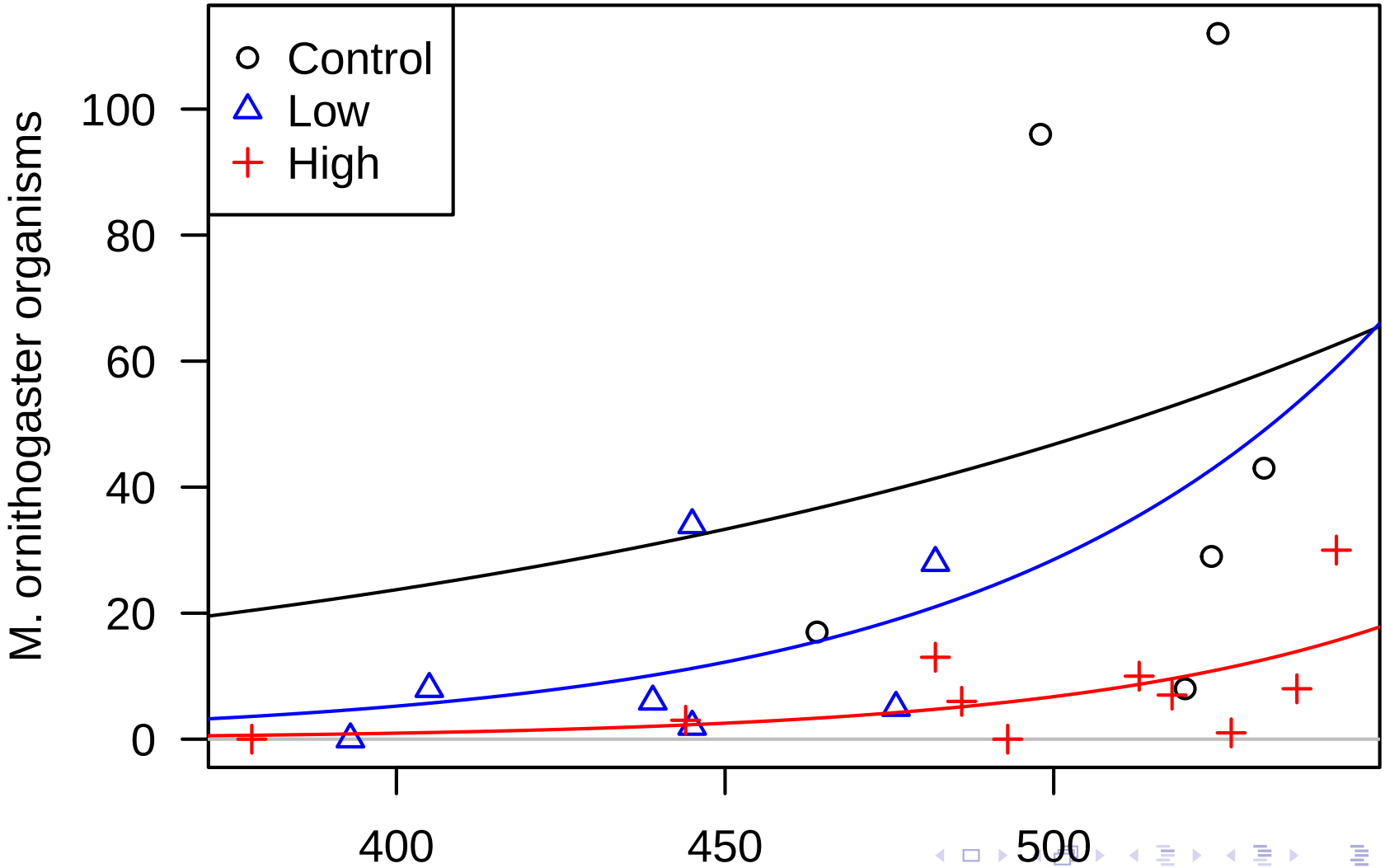
So the main effect, β_2, measures the difference between intercepts, and the interaction effect, β_4, measures the difference in the slopes.

Repeating the above for the low-dose group reveals that β_3 and β_5 measure the differences in the intercept and slope, respectively, between the low-dose and control groups.

Interpretation: Interactions

Interaction effects

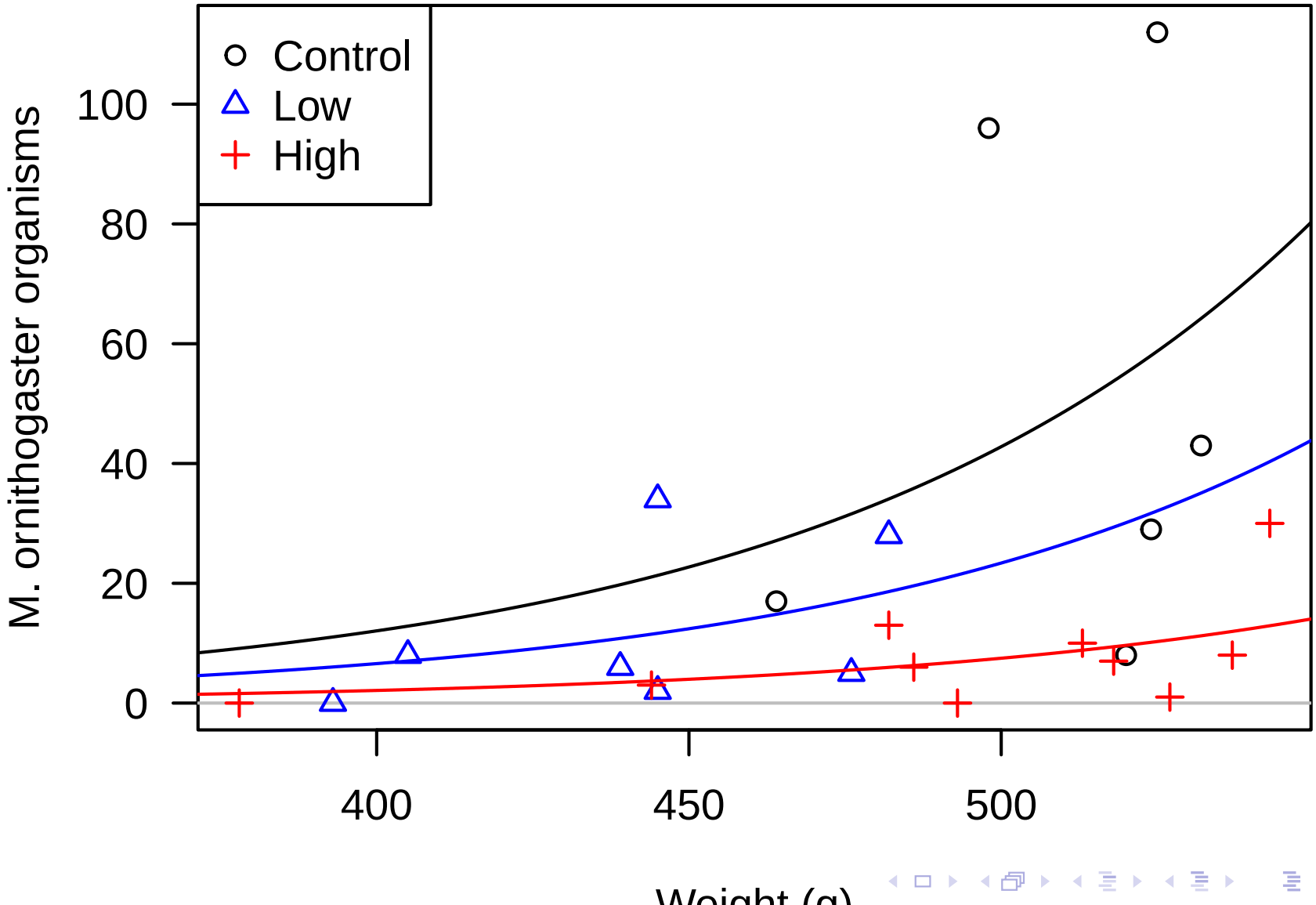
Directly interpreting the interaction effects concisely can be difficult. It can often be easier to plot how the effect of one variable changes across different settings for the other:



Interpretation: Interactions

Interaction effects

For comparison, the estimated relationships for a model without an interaction:



Interpretation: Interactions

Interaction effects

The model with an interaction suggests the following:

- For light chickens, there is not much of a difference between the high and the low dose.
- For heavy chickens, the low dose is not much different to the control, but is quite different to the high dose.

So, what dose should we give a chicken to minimise the number of organisms shedded?

- It depends!
  - We can give light chickens a low dose.
  - We should give heavy chickens a high dose.

Interactions

So far, we have only dealt with interactions between two variables. It is possible to have higher order interactions, such as three-way interactions.

- A three-way interaction allows the interaction between any two variables to depend on the level of the third.

Things can get complicated very quickly! As you begin to consider higher-order interactions, the number of possible models you can fit increases rapidly.
Higher-order interactions are less likely to exist than lower-order interactions or main effects, so often we only consider fitting interactions that we think might exist *a priori*.

How to do it in R

Interactions

To fit a GLM with an interaction between variables `A` and `B` with all possible main effects and interaction effects, we can use

```
fit <- glm(y ~ A + B + A:B, ...)
```

where

- the term `A` indicates we wish to fit main effects for variable A,
- the term `B` indicates we wish to fit main effects for variable B
- the term `A:B` indicates we wish to fit all possible interaction effects between the variables A and B.

A shortcut to fit the same model is

```
fit <- glm(y ~ A*B, ...)
```

which fits interaction effects between `A` and `B`, along with their main effects.

How to do it in R

But, for example, the model

```
fit <- glm(y ~ A + A:B, ...)
```

assumes that the main effects for variable `B` are equal to zero:

- If `A` is a factor, it assumes that there is no effect of variable `B` on the response when factor `A` is set to the baseline level.
- If `A` is a numeric variable, it assumes that there is no effect of variable `B` on the response when variable `A` is set to zero.

But it does allow `B` to have an effect when `A` is not set to the baseline (if `A` is a factor) or if `A` is not set to zero (if `A` is a numeric variable).

It rare that we'd expect this type of scenario, so if we fit a model with an interaction, we almost always fit the lower-order main effects.

How to do it in R

To fit a three way interaction between factors `A`, `B`, and `C`, we can use

```
fit <- glm(y ~ A*B*C, ...)
```

This fits main effects for all factors, two-way interactions between each pair of variables (A and B, A and C, B and C) and a three-way interaction between all factors.

If we wish to fit only all possible two-way interactions we can use

```
fit <- glm(y ~ A*B + A*C + B*C, ...)
```

Or a shortcut is

```
fit <- glm(y ~ (A + B + C)^2, ...)
```

Other model statement syntax

Putting a 1 in your model statement explicitly tells R to fit an intercept term so you'll sometimes see people use it, but R includes it automatically if you leave it out. Both of the following therefore do the same thing:

```
lm(y ~ 1 + x)
lm(y ~ x)
```

If you *don't* want to include an intercept term, you can use -1 in your model statement. For example, the code below will fit a linear model assuming $\mu_i = \beta_1 x_i$:

```
lm(y ~ -1 + x)
```

Offsets

Sometimes a variable has a *known* relationship with the response variable such that its estimation of its effect is not necessary. In this scenario we can sometimes use an offset. Consider the following simple, made-up example.

We wish to determine whether is there is a difference in the average height in adults born in the North Island of NZ and the South Island of NZ. We collect a random sample from each, and measure their heights:

```
head(height.df, 5)

##   height island
## 1  186.7  south
## 2  164.9  south
## 3  179.4  north
## 4  193.9  north
## 5  180.4  north
```

Offsets

Linear regression

... So we fit a linear model:

$$\mu_i = \beta_0 + \beta_1 \text{south}_i$$

```
fit.height <- lm(height ~ island, data = height.df)
summary(fit.height)

## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 173.4739     0.3241 535.299  < 2e-16 ***
## islandsouth  -1.2729     0.4583  -2.777  0.00553 **
## ---
## Residual standard error: 10.25 on 1998 degrees of freedom
## Multiple R-squared:  0.003846,Adjusted R-squared:  0.003347
## F-statistic: 7.714 on 1 and 1998 DF,  p-value: 0.00553
```

... And determine that we have evidence to suggest people from the North Island are taller, on average.

Offsets

Linear regression

Then we realise that we measured people *while they were wearing their shoes*. Maybe people from the North Island are not truly taller, on average, but they wear higher shoes!

We get in touch with our participants and ask them to provide the height of their shoes—in other words, how much their shoes add to their overall height.

```
head(height.full.df, 5)

##   height island shoe
## 1  186.7  south  1.3
## 2  164.9  south  1.5
## 3  179.4  north  1.6
## 4  193.9  north  1.7
## 5  180.4  north  2.7
```

Offsets

Linear regression

The easiest way to approach this problem would be to compute everyone's 'raw' height from their measured height an their shoe height, and rerun the analysis:

```
raw.height <- height - shoe
fit.raw <- lm(raw.height ~ island, data = height.df)
summary(fit.raw)

## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 171.1682     0.3207 533.661   <2e-16 ***
## islandsouth  -0.3396     0.4536  -0.749    0.454
## ---
## Residual standard error: 10.14 on 1998 degrees of freedom
## Multiple R-squared:  0.0002805,Adjusted R-squared:  -0.000219
## F-statistic: 0.5605 on 1 and 1998 DF,  p-value: 0.4541
```

This reveals no significant difference between islands.

Offsets

Linear regression

Alternatively, we could keep the measured height as the response and consider using an offset:

$$\mu_i = \beta_0 + \beta_1 \text{south}_i + \text{shoe}_i$$

Where

- μ_i is the expected value of the measured height, our response;
- $\lambda_i = \beta_0 + \beta_1 \text{south}_i$ is the expected value of the raw height, for which we model with the same linear combination as before; and
- shoe_i is the observed shoe height.

Offsets

Linear regression

Alternatively, we could keep the measured height as the response and consider using an offset:

$$\mu_i = \beta_0 + \beta_1 \text{south}_i + \text{shoe}_i$$

Where

- μ_i is the expected value of the measured height, our response;
- $\lambda_i = \beta_0 + \beta_1 \text{south}_i$ is the expected value of the raw height, for which we model with the same linear combination as before; and
- shoe_i is the observed shoe height.

Offsets

Linear regression

Alternatively, we could keep the measured height as the response and consider using an offset:

$$\mu_i = \beta_0 + \beta_1 \text{south}_i + \text{shoe}_i$$

Where

- μ_i is the expected value of the measured height, our response;
- $\lambda_i = \beta_0 + \beta_1 \text{south}_i$ is the expected value of the raw height, for which we model with the same linear combination as before; and
- shoe_i is the observed shoe height.

Offsets

Linear regression

Alternatively, we could keep the measured height as the response and consider using an offset:

$$\mu_i = \beta_0 + \beta_1 \text{south}_i + \text{shoe}_i$$

Where

- μ_i is the expected value of the measured height, our response;
- $\lambda_i = \beta_0 + \beta_1 \text{south}_i$ is the expected value of the raw height, for which we model with the same linear combination as before; and
- shoe_i is the observed shoe height.

Offsets

Linear regression

Alternatively, we could keep the measured height as the response and consider using an offset:

$$\mu_i = \beta_0 + \beta_1 \text{south}_i + \text{shoe}_i$$

Where

- μ_i is the expected value of the measured height, our response;
- $\lambda_i = \beta_0 + \beta_1 \text{south}_i$ is the expected value of the raw height, for which we model with the same linear combination as before; and
- is the observed shoe height.

An offset can be thought of as a variable in the linear combination that does not have a coefficient to be estimated, or, equivalently, has a coefficient fixed at 1.

Offsets

Linear regression

We can fit an offset in R using the `offset` argument:

```
fit.offset <- lm(height ~ island, offset = shoe,
                 data = height.df)
summary(fit.offset)

## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 171.1682     0.3207 533.661   <2e-16 ***
## islandsouth  -0.3396     0.4536  -0.749    0.454
## ---
## Residual standard error: 10.14 on 1998 degrees of freedom
## Multiple R-squared:  0.03006,Adjusted R-squared:  0.02957
## F-statistic: 61.92 on 1 and 1998 DF,  p-value: 5.817e-15
```

Note that our output here is equivalent to that from our previous model, where we directly used raw height as the response. We can directly interpret the coefficient in terms of the expected raw height, not the expected measured height.

Offsets

Poisson regression

Offsets aren't particularly useful for linear regression, because we can always directly model a new response with the offset substracted off, like our raw heights above.

Offsets are most commonly used in Poisson regression, because they allow us to incorporate variables that measures the 'exposure' of each observation to the events being counted.

We will illustrate this approach with a real example.

Offsets

Poisson regression

These data that come from a study investigating a particular type of minor damage caused by waves to the forward sections of ships' hulls.

In total, 60 ships were inspected for hull damage, and the number of damage incidents were recorded from each. Hull construction engineers are interested in determining if the design of the hull is related to the number of observed damage incidents. Hull designs also potentially improve from year to year.

Offsets

Poisson regression

The data frame contains the following variables:

incidents: The number of damage incidents.

time: The year of manufacture; either 2001, 2002, or 2003.

type: The type of hull design; either A, B, C, or D.

service: The number of months the ship was in service.

```
head(ships.df, 5)

##   incidents time type service
## 1         2 2001    A       7
## 2         4 2001    B      13
## 3         2 2001    C      13
## 4         2 2001    D       9
## 5         2 2001    A      13
```

Offsets

Poisson regression

In this case, the service time of a ship measures its exposure to damage. It might be reasonable to believe that the expected number of damage incidents is directly proportional to service time: all else being equal, a ship in service for twice is long can be expected to have twice as many damage incidents.

In other words, we *know* the effect of the exposure variable on the expected value of the response, and it does not need to be estimated.

We can incorporate this known effect into our model by fitting the log of the exposure variable as an offset.

Offsets

Poisson regression

$$Y_i \sim \text{Poisson}(\mu_i)$$
$$\log(\mu_i) = \beta_0 + \beta_1 t_i + ... + \log(s_i)$$

- Y_i is the observed number of damage incidents for the ith ship, with expectation μ_i.
- $\beta_0 + \beta_1 t_i + ...$ includes the effects of the explanatory variables.
- $\log(s_i)$ is an offset, where s_i is the service time in months.

Offsets

Poisson regression

$$
\begin{aligned}
Y_i &\sim \text{Poisson}(\mu_i) \\
\log(\mu_i) &= \beta_0 + \beta_1 t_i + \ldots + \log(s_i) \\
\mu_i &= \exp[\beta_0 + \beta_1 t_i + \ldots + \log(s_i)]
\end{aligned}
$$

- Y_i is the observed number of damage incidents for the ith ship, with expectation μ_i.
- $\beta_0 + \beta_1 t_i + \ldots$ includes the effects of the explanatory variables.
- $\log(s_i)$ is an offset, where s_i is the service time in months.

Offsets

Poisson regression

$$
\begin{aligned}
Y_i &\sim \text{Poisson}(\mu_i) \\
\log(\mu_i) &= \beta_0 + \beta_1 t_i + \ldots + \log(s_i) \\
\mu_i &= \exp[\beta_0 + \beta_1 t_i + \ldots + \log(s_i)] \\
\mu_i &= \exp(\beta_0 + \beta_1 t_i + \ldots) \times \exp[\log(s_i)]
\end{aligned}
$$

- Y_i is the observed number of damage incidents for the ith ship, with expectation μ_i.
- $\beta_0 + \beta_1 t_i + \ldots$ includes the effects of the explanatory variables.
- $\log(s_i)$ is an offset, where s_i is the service time in months.

Offsets

Poisson regression

$$
\begin{aligned}
Y_i &\sim \text{Poisson}(\mu_i) \\
\log(\mu_i) &= \beta_0 + \beta_1 t_i + \ldots + \log(s_i) \\
\mu_i &= \exp[\beta_0 + \beta_1 t_i + \ldots + \log(s_i)] \\
\mu_i &= \exp(\beta_0 + \beta_1 t_i + \ldots) \times \exp[\log(s_i)] \\
\mu_i &= \exp(\beta_0 + \beta_1 t_i + \ldots) \times s_i
\end{aligned}
$$

- Y_i is the observed number of damage incidents for the ith ship, with expectation μ_i.
- $\beta_0 + \beta_1 t_i + \ldots$ includes the effects of the explanatory variables.
- $\log(s_i)$ is an offset, where s_i is the service time in months.

Offsets

Poisson regression

So we have

$$\mu_i = \exp(\beta_0 + \beta_1 t_i + ...) \times s_i,$$

or equivalently

$$\frac{\mu_i}{s_i} = \exp(\beta_0 + \beta_1 t_i + ...)$$

Where $\lambda_i = \exp(\beta_0 + \beta_1 t_i + ...)$ is the expected number of damage incidents per month, given by exponentiating the linear combination of explanatory terms (not including the offset).

So the expected number of damage incidents, μ_i, is given by the expected number of damage incidents per month, λ_i, multiplied by the number of months the ship is in service, s_i.

Offsets

Poisson regression

We can fit a model incorporating effects of manufacture time and hull type, along with the offset, as follows:

```
ships.fit <- glm(incidents ~ time + type, offset = log(service),
                 family = "poisson", data = ships.df)
summary(ships.fit)

## Call:
## glm(formula = incidents ~ time + type, family = "poisson", data = ships.df,
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 373.44624  177.49665   2.104   0.0354 *
## time         -0.18723    0.08866  -2.112   0.0347 *
## typeB         0.09418    0.19186   0.491   0.6235
## typeC        -0.47699    0.21768  -2.191   0.0284 *
## typeD         0.06967    0.18480   0.377   0.7062
## ---
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 65.425  on 59  degrees of freedom
## Residual deviance: 51.066  on 55  degrees of freedom
```

Offsets

Poisson regression

Our model is

$$Y_i \sim \text{Poisson}(\mu_i)$$
$$\log(\mu_i) = \beta_0 + \beta_1 t_i + \beta 2 b_i + \beta_3 c_i + \beta_4 d_i + \log(s_i)$$

Here, μ_i is the expected number of incidents over the ith boat's s_i months of service. If we wish to calculate its expected number of incidents per month, we can use

$$\lambda_i = \exp(\beta_0 + \beta_1 t_i + \beta_2 b_i + \beta_3 c_i + \beta_4 d_i)$$

We can interprete the coefficients as we normally do for Poisson regression, referring to their effects on μ_i, the expected value of the response. Alternatively, the same interpretation can be used to refer to their effects on λ_i, the expected damage incidents per month.

Other glm() arguments

subset

The subset function allows us to fit our model to only a subset of the observations in a data frame. Similarly to subsetting vectors in R, this can be done by position or via a logical statement. Here some common ways to use it, using the previous example:

Fitting a model to only the first twenty observations:

```
sub1.fit <- glm(incidents ~ time + type, offset = log(service),
                family = "poisson", data = ships.df,
                subset = 1:20)
```

Fitting a model to all observations except the 17th and the 23rd:

```
sub2.fit <- glm(incidents ~ time + type, offset = log(service),
                family = "poisson", data = ships.df,
                subset = -c(17, 23))
```

Other glm() arguments

subset

Fitting a model to obserations from all ships built in 2002 or later:

```
sub3.fit <- glm(incidents ~ time + type, offset = log(service),
                family = "poisson", data = ships.df,
                subset = time >= 2002)
```

Fitting a model to obserations from all ships that aren't of type D:

```
sub4.fit <- glm(incidents ~ time + type, offset = log(service),
                family = "poisson", data = ships.df,
                subset = type != "D")
```

Other `glm()` arguments

`na.action`

The `na.action` argument controls what happens if you try to fit a model to a data set with missing values:

- `na.action = na.omit` will remove any observations with missing values and fit the model. This is sensible enough, but you might not realise this has happened.
- `na.action = na.fail` will result in `glm()` returning an error if there are any missing values.

Summary

We have interpreted effects in each of the models we have considered so far. We have also explored various ways of including explanatory terms in GLMs, including interactions and offsets.

However, we haven't checked the adequacy of any of our models, so we shouldn't necessarily rely on the conclusions we've made here.

Testing for model goodness-of-fit is coming up in Handout 4.