The Linear Model 3: Return of the y_i

Lecture 9

Dr Milan Valášek
28 March 2022

UNIVERSITY OF SUSSEX

Stats wars

- **LM1:** A New Equation
- **LM2:** R^2 strikes back
- **LM3:** Return of the y_i

Today

- Extending the linear model
- Multiple predictors
- Transforming variables in a model
  - Mean-centring
  - Scaling
  - *z*-transforming

Basic linear model

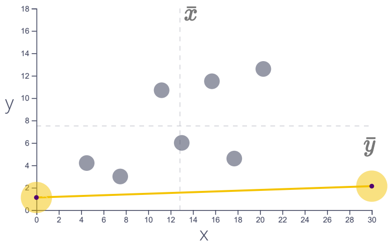
$$\text{outcome}_{\text{obs}} = \text{intercept} + \text{slope} \times \text{predictor}_{\text{obs}} + \text{residual}_{\text{obs}}$$

$$y_i = b_0 + b_1 \times x_{1_i} + e_i$$

- The model is a line through the scatter of data
- The line shows what the value of outcome for a given value of predictor *should* be according to the model
- Residual is the difference between prediction and observation

Mean as linear model

- The simplest linear model is **the mean**
- $y_i = b_0 + e_i$
- $b_0 = Mean(y)$
- That's *literally* the same as $y_i = b_0 + 0 \times x_{1_i} + e_i$
- Mean is the *intercept-only* model: a linear model where all b coefficients other than b_0 have been set (fixed) to zero



Other coefficients?

- Just like we can fix b_1 to zero in $y_i = b_0 + 0 \times x_{1_i} + e_i$, we can fix any other b coefficient as well
- We can think of the basic single-predictor linear model as

$$y_i = b_0 + b_1 \times x_{1_i} + 0 \times x_{2_i} + 0 \times x_{3_i} + \cdots + 0 \times x_{n_i} + e_i$$

- We're just ignoring all but one of the infinity possible predictors we could put in the model
- Not including a predictor in a model is **the same as saying that there is no relationship between that variable and the outcome**
  - It's just said *implicitly* rather than aloud
- We can include them in the model if we wish to so that their associated b coefficient gets estimated, rather than set to 0

Variables are dimensions

- We've been representing the mean as a line on a plot of 2 variables
- It can also be represented as a point on the number line
- Every predictor *adds a dimension*

$$y_i = b_0 + e_i$$

More complex models

- Including more than one predictor allows us to model the outcome variable in a more sophisticated way
- Every slope (b_n coefficient, for $n > 0$) expresses the relationship between a given predictor and the outcome *after the relationship of all other predictors has been accounted for*
- A relationship – causal or not – between two variables can drastically change when another variable is taken into account
- It's important to consider all variables with a known effect when modelling a relationship (especially in observational research)
    - Say we find a relationship between home environment and mental health
    - However, mental health has a strong genetic component
    - Parental predisposition to worse mental health is also linked to home environment
    - Can we *really* claim a relationship between environment and mental health if we don't consider genetics?

Breast is best but is it smartest?

- Lot of ink has been spilled over the claim that breastfeeding leads to increase in child IQ (BBC, The Guardian, The New York Times, FiveThirtyEigth)
- When assessed at face value breastfed children have higher IQ
- Whether or not a person breastfeeds their child is also linked to things like socio-economic status or the person's IQ
- When these effects are adjusted for, the effect shrinks substantially – 3 IQ points difference is a generous estimate and even that has been contested

The linear model allows us to build these more nuanced models and get closer to the Truth about the Universe™

Mutiple predictors in practice

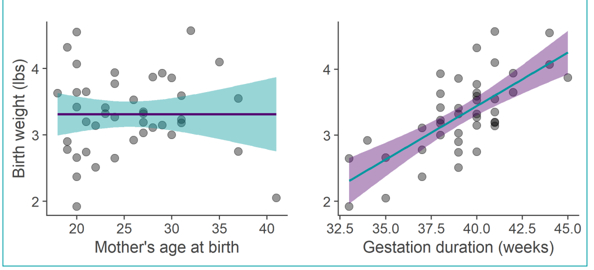
- Today's example focuses on data about babies' birth weights and parental characteristics (source)

ID	Length	Birthweight	Headcirc	Gestation	smoker
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1360	56	4.55	34	44	0
1016	53	4.32	36	40	0
462	58	4.10	39	41	0
1187	53	4.07	38	44	0
553	54	3.94	37	42	0
1636	51	3.93	38	38	0
820	52	3.77	34	40	0
1191	53	3.65	33	42	0
1081	54	3.63	38	38	0
822	50	3.42	35	38	0

1-10 of 42 rows | 1-6 of 16 columns Previous 1 2 3 4 5 Next

Birth weight, mother's age, and gestation time

Plot | R code



Fit model using `lm()`

```
## Intercept-only model
m_null <- lm(Birthweight ~ 1, bweight)

## Add mother's age as predictor
m_age <- lm(Birthweight ~ mage, bweight)
# alternatively update(m_null, ~ . + mage)

## Add gestation duration as predictor
m_gest <- lm(Birthweight ~ mage + Gestation, bweight)
# same as update(m_age, ~ . + Gestation)
```

Results - null model

```
summary(m_null)
```

```
##
## Call:
## lm(formula = Birthweight ~ 1, data = bweight)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.39286 -0.37286 -0.01786  0.33464  1.25714
##
## Coefficients:
##             Estimate Std. Error t value            Pr(>|t|)
## (Intercept)  3.31286    0.09318   35.55 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6039 on 41 degrees of freedom
```

Results - Mother's age as predictor

```
summary(m_age)
```

```
##
## Call:
## lm(formula = Birthweight ~ mage, data = bweight)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.39275 -0.37288 -0.01786  0.33473  1.25702
##
## Coefficients:
##               Estimate Std. Error t value      Pr(>|t|)
## (Intercept) 3.31238583 0.44072153   7.516 0.00000000362 ***
## mage        0.00001845 0.01685112   0.001         0.999
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6114 on 40 degrees of freedom
## Multiple R-squared:  2.996e-08,    Adjusted R-squared:  -0.025
## F-statistic: 1.199e-06 on 1 and 40 DF,  p-value: 0.9991
```

Results - M's age and gestation time

```
summary(m_gest)
```

```
##
## Call:
## lm(formula = Birthweight ~ mage + Gestation, data = bweight)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.77485 -0.35861 -0.00236  0.26948  0.96943
##
## Coefficients:
##               Estimate Std. Error t value    Pr(>|t|)
## (Intercept) -3.0092887  1.0567990  -2.848     0.00699 **
## mage        -0.0007953  0.0120469  -0.066     0.94770
## Gestation    0.1618369  0.0258242   6.267 0.000000221 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4371 on 39 degrees of freedom
## Multiple R-squared:  0.5017,    Adjusted R-squared:  0.4762
## F-statistic: 19.64 on 2 and 39 DF,  p-value: 0.00000126
```

Model prediction

- Linear model can tell us the expected value of outcome for any combination of predictor values
- According to our model, expected birth weight for a baby whose mother is 29 years old and whose gestation period was 38 weeks is:

$$
\begin{aligned}
\hat{y} &= -3.01 + 0 \times \text{age} + 0.16 \times \text{gestation} \\
&= -3.01 + 0 \times 29 + 0.16 \times 38 \\
&= -3.01 + 0 + 6.08 \\
&= 3.07
\end{aligned}
$$

- Let's compare to observations in sample

```
bweight %>% filter(mage == 29 & Gestation == 38) %>%
  rmarkdown::paged_table()
```

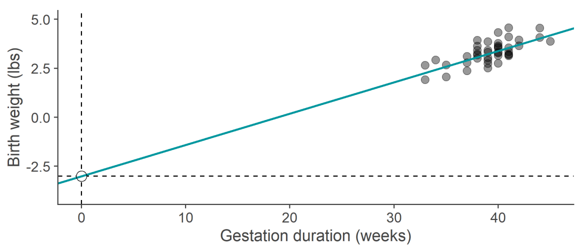
ID <dbl>	Length <dbl>	Birthweight <dbl>	Headcirc <dbl>	Gestation <dbl>	smoker <dbl>
1636	51	3.93	38	38	0

1 row | 1-6 of 16 columns

Negative intercept?!

- The intercept always tells us the value of the outcome when all predictors are 0
  - Not always sensible (instantaneous childbirth in women aged 0 is not a common occurrence)

Plot | R code



Transforming variables in the model

- We can apply various transformations to variables in the model
  - Centring, scaling, standardising
  - Non-linear transformations are also possible (*e.g.,* log-transform)
- Transforming variables **changes the interpretation of the coefficients**

Centring

- Centring *predictors* changes the interpretation of the intercept

```
# untransformed predictor
lm(Birthweight ~ Gestation, bweight)

##
## Call:
## lm(formula = Birthweight ~ Gestation, data = bweight)
##
## Coefficients:
## (Intercept)    Gestation
##     -3.0289       0.1618

# centred predictor
bweight <- bweight %>%
  mutate(gest_cntrd = Gestation - mean(Gestation, na.rm=TRUE))

lm(Birthweight ~ gest_cntrd, bweight)

##
## Call:
## lm(formula = Birthweight ~ gest_cntrd, data = bweight)
##
## Coefficients:
## (Intercept)   gest_cntrd
##      3.3129       0.1618
```

Centring

- What's the weight of a baby born to a "typical" mother in terms of age and pregnancy duration

```
# centre mother's age
bweight <- bweight %>%
  mutate(age_cntrd = mage - mean(mage, na.rm=TRUE))

lm(Birthweight ~ age_cntrd + gest_cntrd, bweight) %>% summary()
```

```
##
## Call:
## lm(formula = Birthweight ~ age_cntrd + gest_cntrd, data = bweight)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.77485 -0.35861 -0.00236  0.26948  0.96943
##
## Coefficients:
##              Estimate Std. Error t value             Pr(>|t|)
## (Intercept)  3.3128571  0.0674405  49.123 < 0.0000000000000002 ***
## age_cntrd   -0.0007953  0.0120469  -0.066                0.948
## gest_cntrd   0.1618369  0.0258242   6.267          0.000000221 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4371 on 39 degrees of freedom
## Multiple R-squared:  0.5017,    Adjusted R-squared:  0.4762
## F-statistic: 19.64 on 2 and 39 DF,  p-value: 0.00000126
```

Scaling

- Scaling *predictors* or *outcome* changes the interpretation of the slopes

```
# untransformed outcome
lm(Birthweight ~ gest_cntrd, bweight)

##
## Call:
## lm(formula = Birthweight ~ gest_cntrd, data = bweight)
##
## Coefficients:
## (Intercept)   gest_cntrd
##      3.3129       0.1618

# scaled outcome
bweight <- bweight %>%
  mutate(bweight_g = Birthweight / 2.205 * 1000) # 2.205 lbs in kg

lm(bweight_g ~ gest_cntrd, bweight)

##
## Call:
## lm(formula = bweight_g ~ gest_cntrd, data = bweight)
##
## Coefficients:
## (Intercept)   gest_cntrd
##     1502.43        73.39
```

Standardising

- Sometimes it's useful to talk about change in outcome associated with a 1 *SD* change in predictors

```
# untransformed predictor
lm(Birthweight ~ Gestation, bweight)

##
## Call:
## lm(formula = Birthweight ~ Gestation, data = bweight)
##
## Coefficients:
## (Intercept)    Gestation
##     -3.0289       0.1618
```

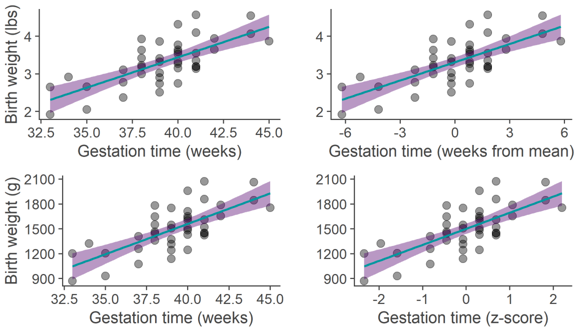
```
# standardised predictor

bweight <- bweight %>%
  mutate(gest_z = scale(Gestation))
lm(bweight_g ~ gest_z, bweight)

##
## Call:
## lm(formula = bweight_g ~ gest_z, data = bweight)
##
## Coefficients:
## (Intercept)       gest_z
##        1502          194
```

It's all the same model!

Plot | R code



Standardised coefficients

- Standardised coefficients are equivalent to b coefficients in a model **where *both* the predictors and the outcome have been *z*-transformed**
- We'll call them B to distinguish them from "raw" coefficients b but there is a lot of confusion in literature about the notation (you may see b, B, β, or $Beta$ used to mean either of the two)
- B expresses the change in outcome in terms of number of *SD* as a result of 1 *SD* change in predictor

Standardised coefficients

- Handy function – `QuantPsyc::lm.beta()`
- Only gives B for slopes, not intercept!

```
m_gest <- lm(Birthweight ~ mage + Gestation, bweight)

# raw coefficients (b)
m_gest %>% coef()

##   (Intercept)          mage     Gestation
## -3.0092887340 -0.0007952874  0.1618368592

# standardised coefficeints (B)
m_gest %>% QuantPsyc::lm.beta()

##         mage    Gestation
## -0.007462176  0.708383324

# same as if we z-transform everything ourselves
lm(scale(Birthweight) ~ scale(mage) + scale(Gestation), bweight) %>% coef() %>% r

##      (Intercept)      scale(mage) scale(Gestation)
##      0.000000000     -0.007462176      0.708383324
```

Take-home message

- Linear model can be easily extended to more than one predictor
- Each predictor entered into the model *adds an extra dimension* to the space in which the model exists
- Each b coefficient (except for b_0) is a slope of the regression plane in its dimension
- Both *including* and *omitting* a variable is a claim about its relationship with the outcome
- A b coefficient for a predictor tells us about the relationship between the predictor and the outcome **after accounting for** the relationship between all other predictors and the outcome
- Intercept may not be a sensible value if variables are not transformed
- Transforming variables *changes the interpretation* of the coefficients
- Standardised coefficients, B, express the change in outcome in terms of number of *SD* as a result of 1 *SD* change in predictor


That's all Fo