Introduction to Generalised Linear Models

Introduction and motivating examples

[video, videoid="5u1w6eROypI", duration="9m57s"] Introduction to GLMs

In this course we will extend the theory of linear regression models, covered in Predictive Modelling and Learning from Data. Before we begin with the first part, which covers a class of models known as Generalised Linear Models (GLMs), we will briefly illustrate why linear models are not sufficient for all types of data. Throughout the course, we will show you how to deal with a variety of situations where the linear model may not be adequate.

The main objective of this week's learning material is to introduce Generalised Linear Models (GLMs), which extend the linear model framework to outcome variables that don't follow the normal distribution. GLMs can be used to model non-normal continuous outcome variables, but they are most frequently used to model binary, categorical or count data. We will focus on these latter types of outcome variables. To see why extensions to the normal linear model are needed, let's look at a couple of examples, one where the normal linear model is appropriate and one where it's not.

[example]Bollywood box office revenue

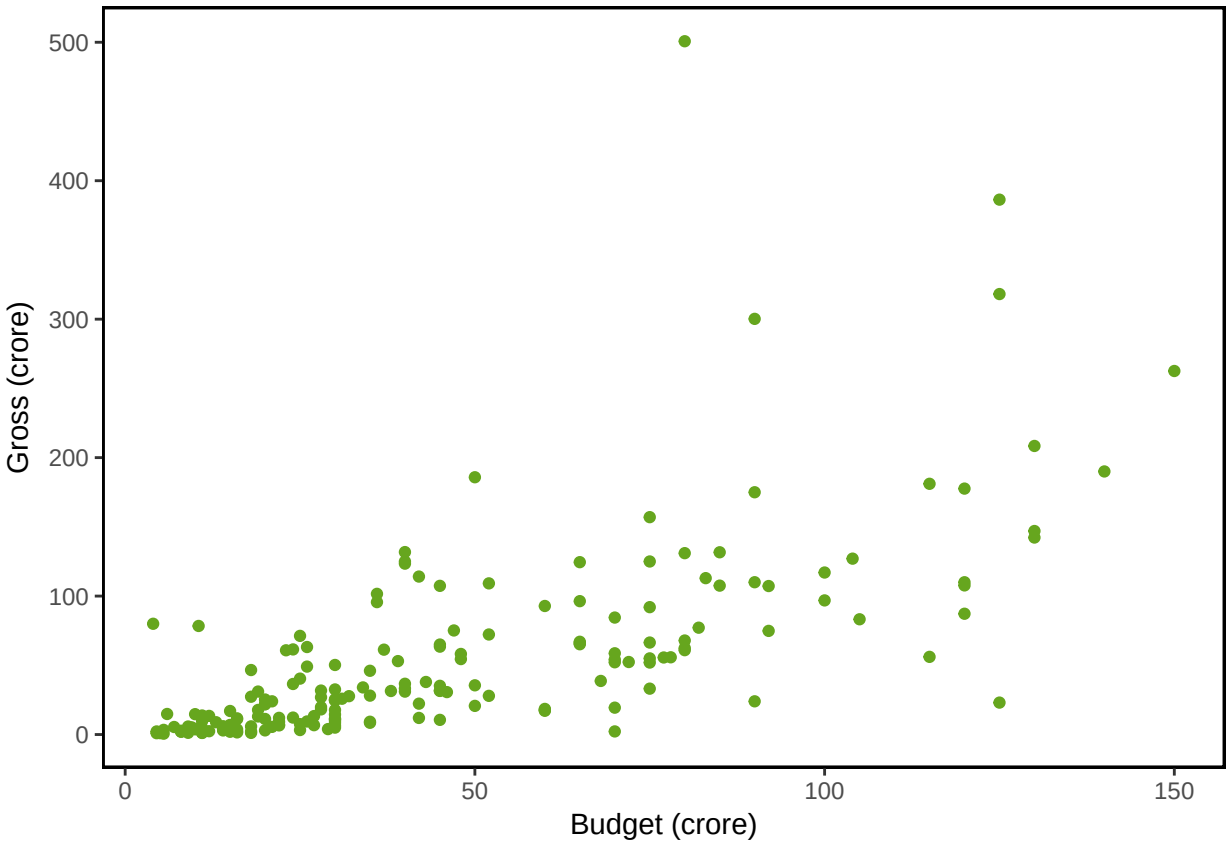
Possibly the simplest scenario of a predictive model is when we want to predict an outcome variable based on a predictor which displays a linear relationship to the variable of interest. Consider the following dataset on Bollywood film revenues (source:http://www.bollymoviereviewz.com) which contains data on 190 films made during the period 2013-2017. We would like to predict the gross revenue of a film from the film's budget. Both the gross revenue and the budget are measured in crore. Here are the first few rows of the data:

```
bollywood <-
  read.csv(url("http://www.stats.gla.ac.uk/~tereza/rp/bollywood_boxoffice.csv"))
head(bollywood)
```

Movie	Gross	Budget
Ek Villain	95.64	36.0
Humshakals	55.65	77.0
Holiday	110.01	90.0
Fugly	11.16	16.0
City Lights	5.19	9.5
Kuku Mathur Ki Jhand Ho Gayi	2.23	4.5

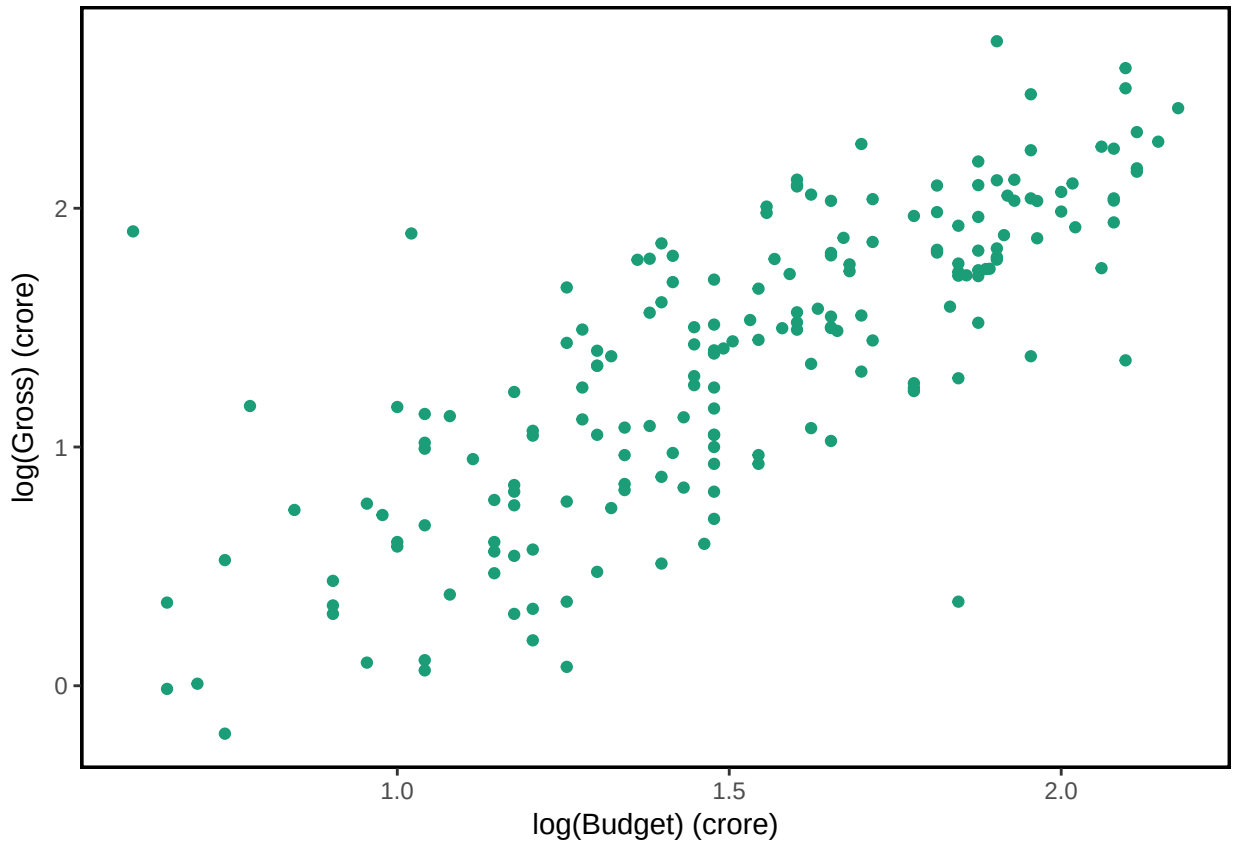
We can plot the gross revenue against the budget to explore the relationship between the two variables.

```
b.plot <- ggplot(data = bollywood, aes(y = Gross, x = Budget)) +
          geom_point( col = "#66a61e") +
          scale_x_continuous("Budget (crore)") + scale_y_continuous("Gross (crore)")
```



Looking at the scale of the values on both the horizontal and vertical axes, we might want to transform the data by taking logs.

```
b.plot.l <- ggplot(data = bollywood, aes(y = log10(Gross), x = log10(Budget))) +
          geom_point( col = "#1b9e77") +
          scale_x_continuous("log(Budget) (crore)") +
          scale_y_continuous("log(Gross) (crore)")
```



Now let's fit a model with the $\log_{10}$ transformed *gross revenue* as the response (Y_i) and the $\log_{10}$ transformed *budget* (x_i) as the explanatory/predictor variable. We can use the `lm` function to fit this linear model in R.

```
bol.lm <- lm( log10(Gross) ~ log10(Budget), data = bollywood)
```

The model equation in mathematical notation is

$$E(Y_i) = \mu_i = \beta_0 + \beta_1 x_i; \quad \text{where the } Y_i \text{ are independent } N(\mu_i, \sigma^2), \quad i = 1, ..., 190$$

The model fit is shown below:

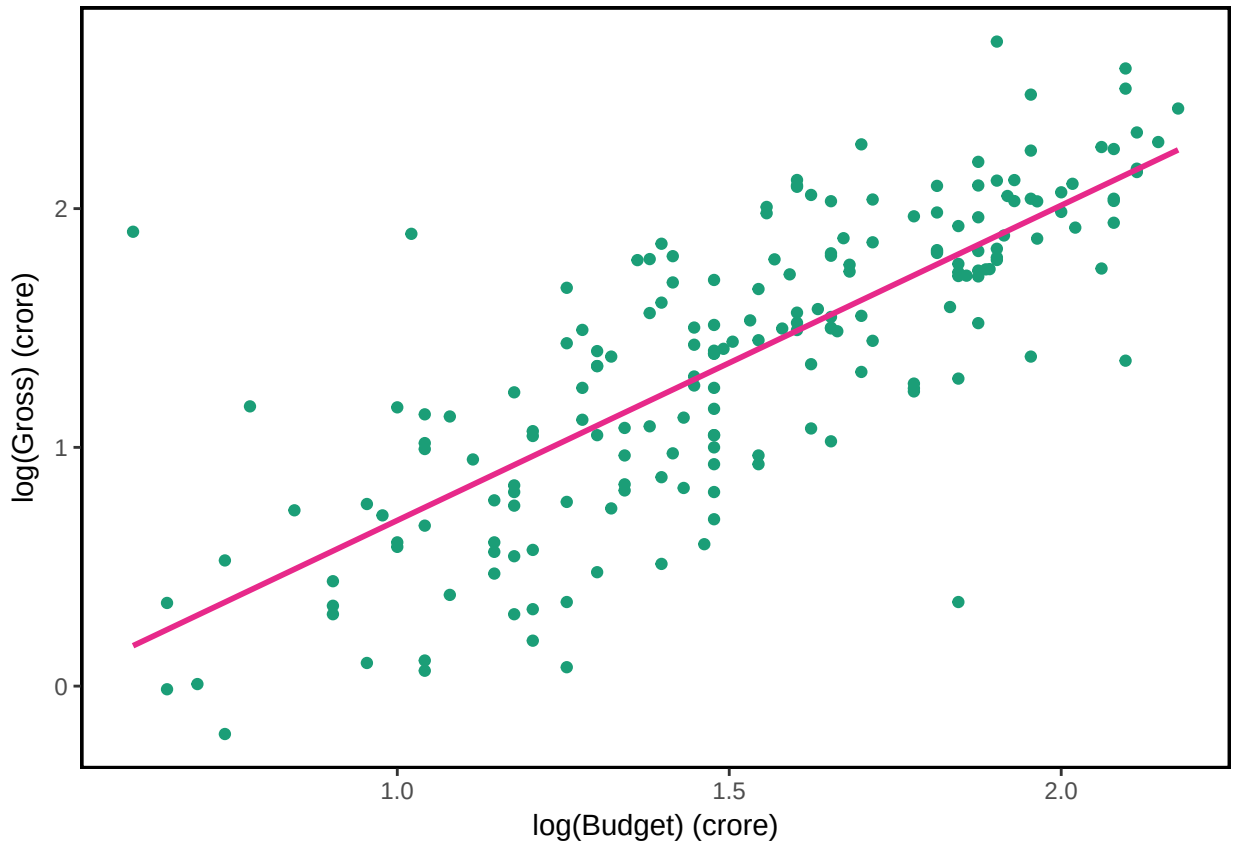
```
summary(bol.lm)
```

```
Call:
lm(formula = log10(Gross) ~ log10(Budget), data = bollywood)

Residuals:
     Min       1Q   Median       3Q      Max
-1.45702 -0.24470  0.00807  0.24600  1.73413

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   -0.62549    0.12338  -5.069 9.51e-07 ***
log10(Budget)  1.31955    0.07887  16.730  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.3921 on 188 degrees of freedom
Multiple R-squared:  0.5982,    Adjusted R-squared:  0.5961
F-statistic: 279.9 on 1 and 188 DF,  p-value: < 2.2e-16
```

We can visualise this regression model by plotting the data and fitted regression line:

```
b.plot.lm <- ggplot(data = bollywood, aes(y = log10(Gross), x = log10(Budget))) +
          geom_point(col ="#1b9e77") +
          scale_x_continuous("log(Budget) (crore)") +
          scale_y_continuous("log(Gross) (crore)") +
          geom_smooth(method = lm, colour="#e7298a", se=FALSE)
```



[/example]

[task]

Using the fitted model equation, predict the gross revenue for a film with a budget of (i) 10, (ii) 50, and (iii) 100 crore.

Hint: Remember that the variables have been log-transformed!

[answer]

For the *Bollywood box office revenue* example, we can write down the fitted model equation from the `summary(bol.lm)`:

$$\log 10(\text{Gross}) = -0.62549 + 1.31955 \times \log 10(\text{Budget})$$

We can use this equation to predict the gross revenue of a film by simply substituting the relevant budget value, and transforming the result from the log10 scale. Thus:

(i) budget = 10:

$$\log 10(\text{Gross}) = -0.62549 + 1.31955 \times \log 10(10) = 0.69406 \Rightarrow \text{Gross} = 10^{0.69406} = 4.94379$$

(ii) budget = 50:

$$\log 10(\text{Gross}) = -0.62549 + 1.31955 \times \log 10(50) = 1.616386 \Rightarrow \text{Gross} = 10^{1.616386} = 41.34148$$

(iii) budget = 100:

$$\log 10(\text{Gross}) = -0.62549 + 1.31955 \times \log 10(100) = 2.01361 \Rightarrow \text{Gross} = 10^{2.01361} = 103.1834$$

###[/answer]

[/task]

[example] GPA and admission to medical school

Now let's look at a different kind of dataset, where the outcome we want to predict is not continuous-valued but binary. This is a dataset on admissions to US medical schools which you have first seen in Predictive Modelling. The dataset gives the admission status, GPA and standardised test scores for 55 medical school applicants from a liberal arts college in the US Midwest and it can be loaded from the Stat2Data package in R.

```
library(Stat2Data)
data(MedGPA)
```

The first few rows of the data are given below.

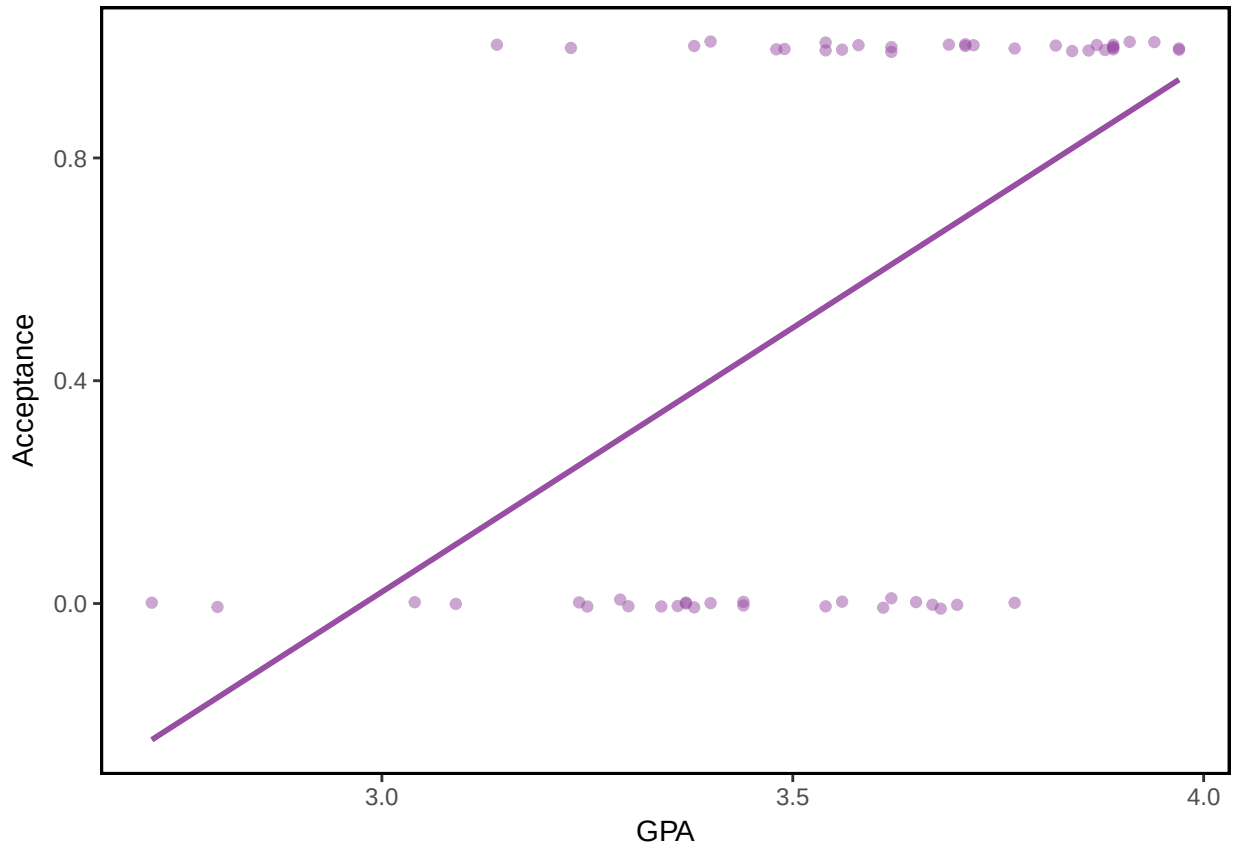
Accept	Acceptance	Sex	BCPM	GPA	VR	PS	WS	BS	MCAT	Apps
D	0	F	3.59	3.62	11	9	9	9	38	5
A	1	M	3.75	3.84	12	13	8	12	45	3
A	1	F	3.24	3.23	9	10	5	9	33	19
A	1	F	3.74	3.69	12	11	7	10	40	5
A	1	F	3.53	3.38	9	11	4	11	35	11
A	1	M	3.59	3.72	10	9	7	10	36	5

Let us look at a plot of acceptance against GPA, adding a bit of jitter to make overlapping points more visible.

```
medgpa.plot <- ggplot(data = MedGPA, aes(y = Acceptance, x = GPA)) +
               geom_jitter(width =0, height =0.01, alpha =0.5, colour ="#984ea3")
```

We can add the linear regression line for `Acceptance` as a function of `GPA` to the plot.

```
medgpa.plot + geom_smooth(method = "lm", se = FALSE,
                          fullrange = TRUE, colour = "#984ea3")
```



The R code for fitting the model and the model output is shown below.

```
med.lm <- lm(Acceptance ~ GPA, data=MedGPA)
summary(med.lm)
```

```
Call:
lm(formula = Acceptance ~ GPA, data = MedGPA)

Residuals:
    Min      1Q  Median      3Q     Max
-0.7510 -0.3717  0.1352  0.3059  0.8464

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -2.8240     0.7226  -3.908 0.000266 ***
GPA           0.9483     0.2027   4.678 2.04e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4267 on 53 degrees of freedom
Multiple R-squared:  0.2922,    Adjusted R-squared:  0.2788
F-statistic: 21.88 on 1 and 53 DF,  p-value: 2.043e-05
```

In mathematical notation, we have independent responses $Y_i = 1$ if the ith applicant is accepted, $Y_i = 0$ otherwise, with x_i equal to the ith applicant's college GPA for $i = 1, \ldots, 55$. The normal linear model assumes that the Y_i are independent $N(\mu_i, \sigma^2)$ with $\mu_i = \beta_0 + \beta_1 x_i$ with the fitted model equation given by

$\hat{\mu}_i = -2.8240 + 0.9483x_i$.

One issue with this fit is that the predicted values of the response can take any real values, while acceptance can only take the value 0 or 1. And it is hard to argue that a variable taking values of 0 or 1 is normally distributed. Instead, we can use a logistic regression model for the *probability* of acceptance. Let's first write it down in mathematical notation by letting $p_i = P(Y_i = 1)$. This is the probability of acceptance for the ith applicant. We assume that the Y_i are independent $Bin(1, p_i)$ (or Bernoulli(p_i)) with

$$\log\left(\frac{p_i}{1-p_i}\right) = \beta_0 + \beta_1 x_i.$$

This is equivalent to:

$$p_i = \frac{\exp(\beta_0 + \beta_1 x_i)}{1 + \exp(\beta_0 + \beta_1 x_i)}.$$

We can fit this model in R using the `glm` function:

```
med.glm <- glm( Acceptance ~ GPA, data = MedGPA, family = binomial)
```

The argument `family=binomial` specifies that `Acceptance` follows a `binomial` distribution, with probability of success p, and the probability p is a function of `GPA`. The default link function, corresponding to the logit link $\log\left(\frac{p}{1-p}\right)$, is used here. That is, `family=binomial` implies `family = binomial(link="logit")`.

The model fit is shown below.

```
summary(med.glm)
```

```
Call:
glm(formula = Acceptance ~ GPA, family = binomial, data = MedGPA)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.7805  -0.8522   0.4407   0.7819   2.0967

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -19.207      5.629  -3.412 0.000644 ***
GPA            5.454      1.579   3.454 0.000553 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 75.791  on 54  degrees of freedom
Residual deviance: 56.839  on 53  degrees of freedom
AIC: 60.839

Number of Fisher Scoring iterations: 4
```

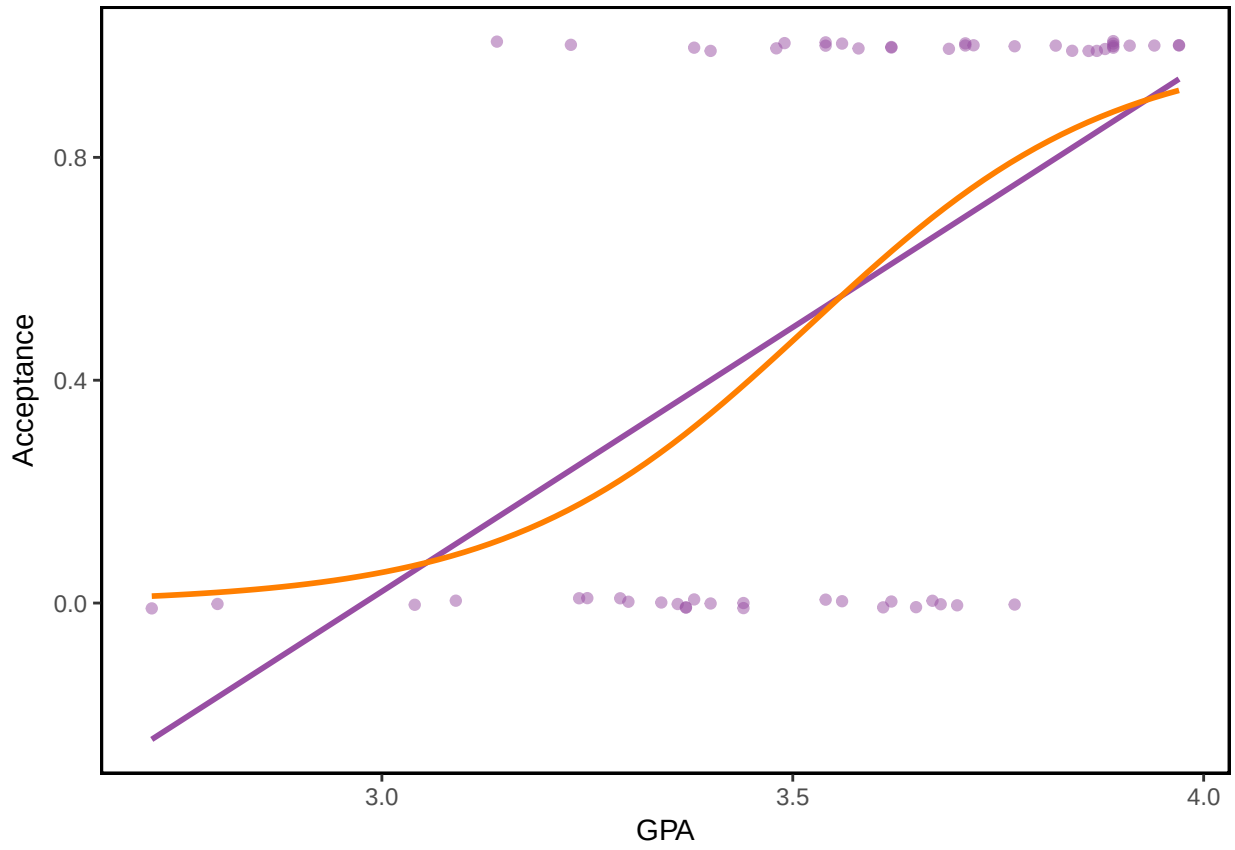
The regression equation for the fitted model is

$$\log\left(\frac{\hat{p}_i}{1-\hat{p}_i}\right) = -19.207 + 5.454x_i,$$

or equivalently

$$\hat{p}_i = \frac{\exp(-19.207 + 5.454x_i)}{1 + \exp(-19.207 + 5.454x_i)}.$$

The fitted curve for the probability of acceptance is shown in orange below.



We can see that that this curve fits the data better than the linear regression line, and that it gives predicted probabilities between 0 and 1, as desired. We could add predictors to the model to improve predictive performance – we'll see more about that later.

The regression equation we have obtained allows us to predict the acceptance probability for a given GPA.

[/example]

[task]

Predict the acceptance probability for an applicant with a GPA of (i) 2.5, (ii) 3 (iii) 4. First do this "by hand" using the regression equation, then in R using the `predict` function.

Hint: The `predict` *function will return values on the linear predictor scale unless you specify* `type='response'` *which returns probabilities instead.*

[answer]

In the *GPA and admission to medical school* example, we can write down the fitted model equation from the `summary(med.glm)`:

$$\log\left(\frac{p_i}{1-p_i}\right) = -19.207 + 5.454 \times \text{GPA}$$

From the fitted equation, we can obtain the acceptance probability by solving for p_i:

$$p_i = \frac{\exp(-19.207 + 5.454 \times \text{GPA})}{1 + \exp(-19.207 + 5.454 \times \text{GPA})}$$

To predict the acceptance probability for an applicant we just need to substitute the specified GPA in the equation for p_i:

(i) GPA = 2.5:

$$p_i = \frac{\exp(-19.207 + 5.454 \times 2.5)}{1 + \exp(-19.207 + 5.454 \times 2.5)} \Rightarrow p_i = 0.00378$$

(ii) GPA = 3:

$$p_i = \frac{\exp(-19.207 + 5.454 \times 3)}{1 + \exp(-19.207 + 5.454 \times 3)} \Rightarrow p_i = 0.05494$$

(iii) GPA = 4:

$$p_i = \frac{\exp(-19.207 + 5.454 \times 4)}{1 + \exp(-19.207 + 5.454 \times 4)} \Rightarrow p_i = 0.93143$$

Alternatively, we can use the `predict` function in `R` as folows:

```
predict( med.glm, data.frame( GPA = c(2.5, 3, 4) ), type = 'response')
```

```
          1           2           3
0.003791903 0.054992029 0.931512655
```

[/answer]

[/task]

What do the Bollywood box office and medical school admission examples have in common?

[weblink,target="https://glasgow.summon.serialssolutions.com/#!/search?bookMark=ePnHCXMw42LgTQStzc4rAe_hSmGGzJCCTlUH348J2q1rYgps5nPARkJAc2HmBoacDKa-mRWpKQrQhQEJwFraPApLMZEKAEACYU39A", icon=book]

- Section 2.3 from *Mixed effects models and extensions in ecology with R -Zuur et al.* discusses the appropriateness of the assumptions of the linear model.

[/weblink]

In both cases we have independent observations and we want to predict an outcome of interest (gross revenue/acceptance) based on an explanatory variable (budget/GPA). In both cases we have a regression equation allowing us to predict the response from a given value of the predictor. However, in one case our response is assumed to follow the normal distribution, in the other the binomial distribution. In both cases we fit a model to the *mean* of the response: in the normal linear model the mean $E(Y_i) = \mu_i$ is assumed to be a linear function of x_i: $\mu_i = \beta_0 + \beta_1 x_i$ and in the logistic regression the mean $\mu_i = E(Y_i) = p_i$ is modelled through the *logit link function*. That is, in logistic regression $\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_i$. In a little bit more general notation we have $g(\mu_i) = \boldsymbol{x}_i^T\boldsymbol{\beta}$ where $\mu_i = E(Y_i)$ and $g(\mu_i)$ for each distribution is given in the following table.

Model	**Random component**	**Systematic component**	**Link function**
Normal model	$y_i \overset{\text{indep}}{\sim} N(\mu_i, \sigma^2)$, $E(Y_i) = \mu_i$	$\boldsymbol{x}_i^T\boldsymbol{\beta} = \beta_0 + \beta_1 x_i$	Identity link $g(\mu_i) = \mu_i$
Logistic regression model	$y_i \overset{\text{indep}}{\sim} Bin(1, p_i)$, $E(Y_i) = p_i$	$\boldsymbol{x}_i^T\boldsymbol{\beta} = \beta_0 + \beta_1 x_i$	Logit link: $g(\mu_i) = \log\left(\frac{\mu_i}{1-\mu_i}\right) = \log\left(\frac{p_i}{1-p_i}\right)$

Exponential family of distributions

[weblink,target="https://glasgow.summon.serialssolutions.com/#!/search?bookMark=ePnHCXMw42LgTQStzc4rAe__hSmGGzJCCTlUH348J2q1rYgps5nPARkJAc2HmBoacDKa-mRWpKQrQhC EJwFraPApLMZEKAEACYU39A", icon=book]

- Chapter 8 from *Mixed effects models and extensions in ecology with R -Zuur et al.* contains a more in-depth discussion about the exponential family.

[/weblink]

It turns out that the normal and binomial distributions also have something else in common: they are both members of the *exponential family of distributions*. (And so is the Poisson, the negative binomial, gamma distribution and many others.)

[definition] Exponential family of distributions

Consider a random variable Y whose probability density function (p.d.f.) or probability mass function (p.m.f.) depends on parameter θ. The distribution belongs to the exponential family if it can be written as

$$f(y;\theta) = \exp\left[a(y)b(\theta) + c(\theta) + d(y)\right].$$

The term $b(\theta)$ is called the *natural parameter*. If $a(y) = y$ the distribution is said to be in *canonical form*.
###[/definition]

[example] Normal distribution is a member of exponential family

Consider $Y \sim N(\theta, \sigma^2)$ with p.d.f.

$$f(y;\theta) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2\sigma^2}(y-\theta)^2\right], \quad -\infty < y < \infty. \tag{1}$$

If we are interested in estimating θ, the variance, σ^2, can be regarded as a nuisance parameter. By rewriting the p.d.f. as

$$f(y;\theta) = \exp\left[-\frac{y^2}{2\sigma^2} + \frac{y\theta}{\sigma^2} - \frac{\theta^2}{2\sigma^2} - \frac{1}{2}\log(2\pi\sigma^2)\right] \tag{2}$$

we can see that this is of exponential family form with $a(y) = y$ (hence in canonical form) and natural parameter $b(\theta) = \theta/\sigma^2$.

[/example]

[task]

Show that the binomial distribution $Bin(n, p)$ is a member of the exponential family. ###[/task]

The exponential family of distributions has several interesting and useful properties. It can be proven that the expectation and variance for members of the exponential family can be expressed as

$$E[a(Y)] = -\frac{c'(\theta)}{b'(\theta)}$$

and

$$\mathrm{Var}[a(Y)] = \frac{b''(\theta)c'(\theta) - c''(\theta)b'(\theta)}{[b'(\theta)]^3}.$$

Some further useful properties of exponential family distributions relate to the **score function**.

[definition] Score statistic

$U = \dfrac{dl(\theta;y)}{d\theta}$ is called the **score statistic**, and is equal to the derivative of the log-likelihood $l(\theta;y)$ with respect to the parameter θ. ###[/definition]

For exponential family distributions with log-likelihood $l(\theta;y) = a(y)b(\theta) + c(\theta) + d(y)$, the score is

$$U(\theta;y) = \frac{dl(\theta;y)}{d\theta} = a(y)b'(\theta) + c'(\theta) \tag{3}$$

Remember that in *Learning from Data* (Week 4 and Week 5) you used the score to solve the likelihood equation $U = \dfrac{dl(\theta;y)}{d\theta} = 0$ to obtain the maximum likelihood estimate $\hat{\theta}$ for a number of distributions.

We can think of the score statistic, $U = a(Y)b'(\theta) + c'(\theta)$ as a random variable in its own right, which means we can calculate its expectation

$$\mathrm{E}(U) = b'(\theta)E[a(Y)] + c'(\theta) = b'(\theta)\left[-\frac{c'(\theta)}{b'(\theta)}\right] + c'(\theta) = 0,$$

and its variance

$$\mathrm{Var}(U) = [b'(\theta)^2]\mathrm{Var}[a(Y)].$$

This leads us to a very important concept in statistical inference, called **Fisher information**, which we can use to obtain standard errors for maximum likelihood estimates of our GLM coefficients. ###[definition] Fisher's information

The **Fisher Information**, denoted as $\mathcal{I}$, is given by:

$$\mathcal{I} = \text{Var}[U] = E(U^2) = E\left[\left(\frac{dl(\theta; y)}{d\theta}\right)^2\right] = E\left[\frac{d^2 l(\theta; y)}{d\theta^2}\right]. \tag{4}$$

[/definition]

The variance of the maximum likelihood estimates tells us about the amount of *information* that an observed random variable carries about an unknown parameter in the model that is linked to a distribution.

As we will see shortly, the score and information play a key role in parameter estimation and in obtaining standard errors for the coefficient estimates of a GLM.

Having defined the exponential family of distributions, we are now ready to formally define a GLM.

Generalised Linear Models

[weblink,target="http://encore.lib.gla.ac.uk/iii/encore/record/C___Rb2939999?lang=eng",icon=book]

- Chapter 6 from *Extending linear models with R: generalized linear, mixed effects and nonparametric regression models - Faraway*

[/weblink]

[definition] Generalised Linear Models

Let Y_i be independent responses from an exponential family distribution in canonical form and $\mu_i = E(Y_i)$, $i = 1, \ldots, n$. A *generalised linear model* (GLM) is a model of the form $g(\mu_i) = \boldsymbol{x}_i^T \boldsymbol{\beta}$ where $\boldsymbol{\beta}$ is a p-dimensional parameter vector, $\boldsymbol{x}_i^T$ is the ith row of the design matrix $\boldsymbol{X}$, and $g()$ is a monotonic, differentiable function called the *link function.* ###[/definition]

A GLM generalises the normal linear model by allowing

1. a response variable with a distribution other than normal, but a member of the exponential family of distributions; and
2. a relationship between the response and the linear component of the form $g(\mu_i) = \boldsymbol{x}_i^T \boldsymbol{\beta}$ where g is the *link function.*

Components of a generalised linear model

1. The *random component*: Suppose $Y_1, \ldots, Y_n$ are independent random variables which follow an exponential family distribution such that $f(y_i; \theta_i) = \exp[y_i b(\theta_i) + c(\theta_i) + d(y_i)]$ for $i = 1, \ldots n$. The joint p.d.f. of the Y_i is

$$f(y_1, \ldots, Y_n; \theta_1, \ldots, \theta_n) = \prod_{i=1}^{n} \exp[y_i b(\theta_i) + c(\theta_i) + d(y_i)]$$
$$= \exp\left[\sum_{i=1}^{n} y_i b(\theta_i) + \sum_{i=1}^{n} c_i(\theta_i) + \sum_{i=1}^{n} d(y_i)\right] \quad (5)$$

The distribution of each Y_i is in canonical form and depends on a single parameter θ_i.

2. The *systematic component*: Associated with each y_i is a vector $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})^T$ of values of p explanatory variables. The response, Y_i, depends on the explanatory variables through a linear component, $\eta_i = \boldsymbol{x}_i^T \boldsymbol{\beta} = \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$ for $i = 1, \ldots, n$ where $\boldsymbol{x}_i^T$ is the ith row of the design matrix $\boldsymbol{X}$ and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^T$ is the parameter vector. As in linear models, the design matrix is given by
$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1^T \\ \vdots \\ \boldsymbol{x}_n^T \end{bmatrix} = \begin{bmatrix} x_{11} & \ldots & x_{1p} \\ \vdots & & \vdots \\ x_{n1} & \ldots & x_{np} \end{bmatrix}.$$

3. The *link function*: The parameters θ_i in equation (5) are usually not of direct interest. Instead, we are interested in a smaller set of parameters $(\beta_1, \ldots, \beta_p)$, and assume that Y_i depends on these through the linear predictor η_i. The link between the distribution of the Y_i and the linear predictor η_i is provided by the link function g, for which $g(\mu_i) = \eta_i = \boldsymbol{x}_i^T \boldsymbol{\beta}$. Here $\mu_i = E(Y_i)$ and g is a monotone, differentiable function. Although any one-to-one function could be used in principle, certain choices of link function can offer great simplification. In particular, the link function can be chosen so that the natural parameter, $b(\theta_i)$, is proportional to the linear component $\eta_i = \boldsymbol{x}_i^T \boldsymbol{\beta}$. Such a link function is known as the *canonical link*. The following table shows the canonical link function for some of the most common distributions.

Distribution	**Natural parameter**	**Canonical link**
Normal	$\dfrac{\theta}{\sigma^2}$	$g(\mu) = \mu$
Poisson	$\log \theta$	$g(\mu) = \log(\mu)$
Binomial	$\log\left(\dfrac{\theta}{1-\theta}\right)$	$g(\mu) = \log\left(\dfrac{\mu}{1-\mu}\right)$

[weblink,target="https://glasgow.summon.serialssolutions.com/#!/search?bookMark=ePnHCXMw42LgTQStzc4rAe__hSmGGzJCCTlUH348J2q1rYgps5nPARkJAc2HmBoacDKa-mRWpKQrQhQEJwFraPApLMZEKAEACYU39A", icon=book]

- Section 9.1 and 9.2 from *Mixed effects models and extensions in ecology with R -Zuur et al.* cover the general formulation of GLMs.

[/weblink]