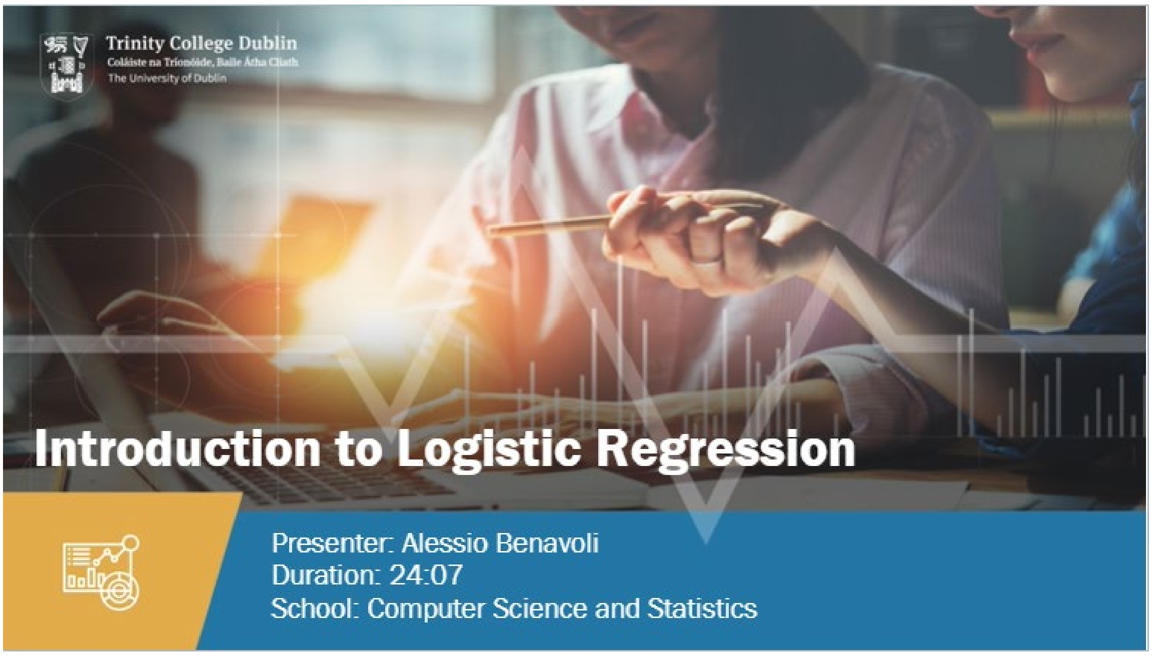
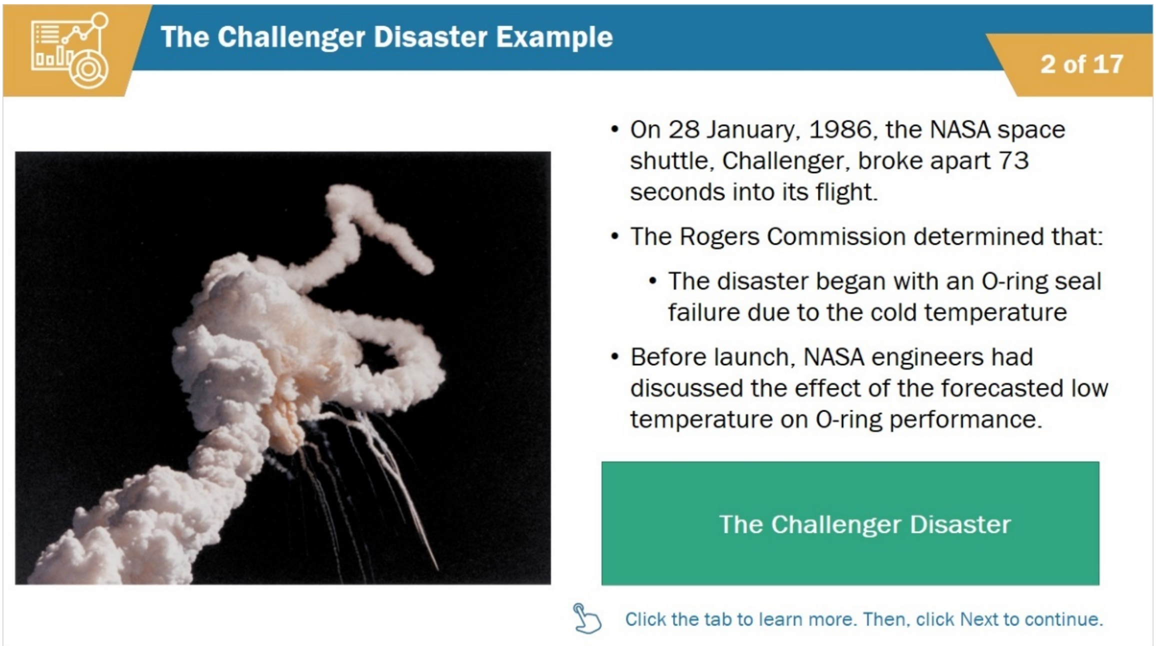
Introduction to logistic regression: Choosing a Regression Model

Slide 1: **Introduction**



My name is **Alessio Benavoli** and I am **the instructor for this session.** During this presentation I will introduce logistic regression. We will use a motivational example to introduce the topic: the disaster of the Space Shuttle Challenger. We will discuss the difference between ordinary linear regression and logistic regression. We will see how to fit the parameters of the logistic regression model, how to make predictions and perform statistical analysis. We will then move to multiple predictors and conclude the session with a brief overview of generalised linear model.

Slide 2: **The Challenger Disaster Example**



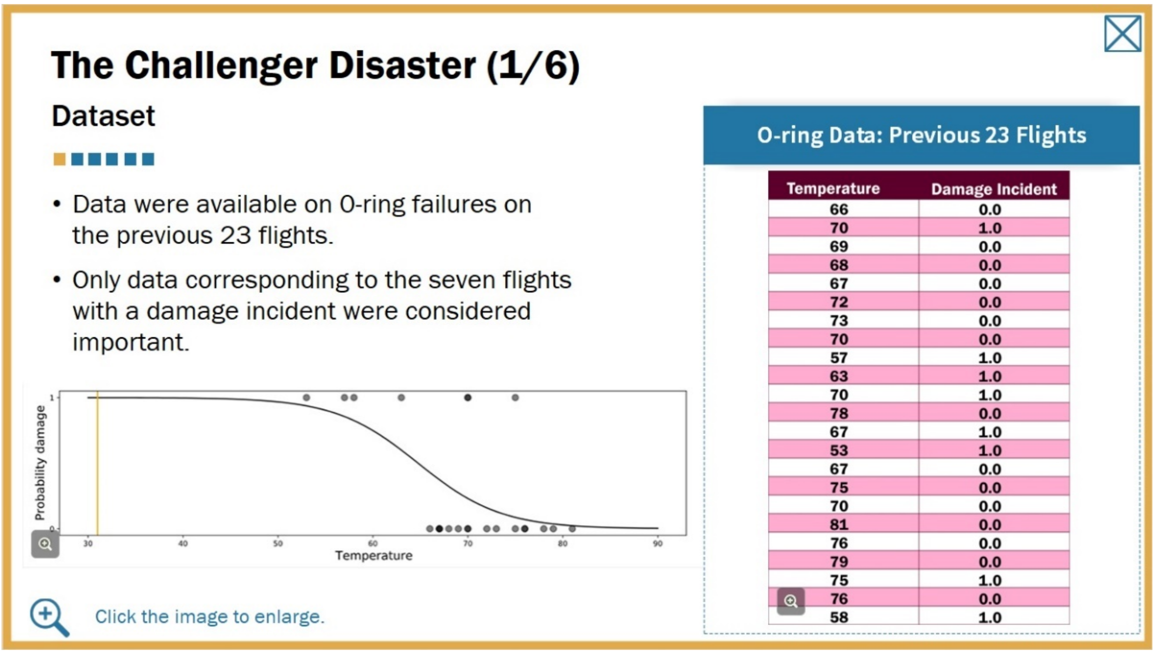
On January 28th, 1986, the NASA Space Shuttle Challenger broke apart 73 seconds into its flight.

NASA appointed members of the Rogers Commission to investigate the cause of the disaster. The commission determined that the disaster began with the failure of an O-ring seal in the solid rocket motor due to the cold temperature (-0.6 Celsius degrees) during the launch.

The problem with O-rings was known and the night before the launch, there was a three-hour teleconference between NASA engineers, discussing the effect of low temperature forecasted for the launch on the O-ring performance.

Click the tab to learn more about the consequences of excluding certain temperature data. When you are ready, click next to continue.

Tab 1: The Challenger Disaster: Dataset



Temperature	Damage Incident
66	0.0
70	1.0
69	0.0
68	0.0
67	0.0
72	0.0
73	0.0
70	0.0
57	1.0
63	1.0
70	1.0
78	0.0
67	1.0
53	1.0
67	0.0
75	0.0
70	0.0
81	0.0
76	0.0
79	0.0
75	1.0
76	0.0
58	1.0

Data were available on failures of O-rings on the previous 23 flights, and these data were discussed during the meeting, but unfortunately only the data corresponding to the seven flights on which there was a damage incident were considered important. The table shows the temperature at the launch and the right column indicates a damage incident (value 1) or no-damage (value 0).

Tab 1.1: NASA Engineers

The Challenger Disaster (2/6)

NASA Engineers

- The NASA engineers concluded that temperature data was not conclusive in predicting O-ring failure.
  - This was based only on using data for the seven flights with a damage incident.
    - Failures seemed to appear at any temperature.

Data For Seven Flights With a Damage Incident

Temperature	Damage Incident
70	1.0
57	1.0
63	1.0
70	1.0
53	1.0
75	1.0
58	1.0

The conclusion of NASA engineers based on these 7 data points was Temperature data is not conclusive on predicting primary O-ring failure. According to the engineers, the failures seem to appear at any temperature.

Tab 1.2: Rogers Commission Report

The Challenger Disaster (3/6)

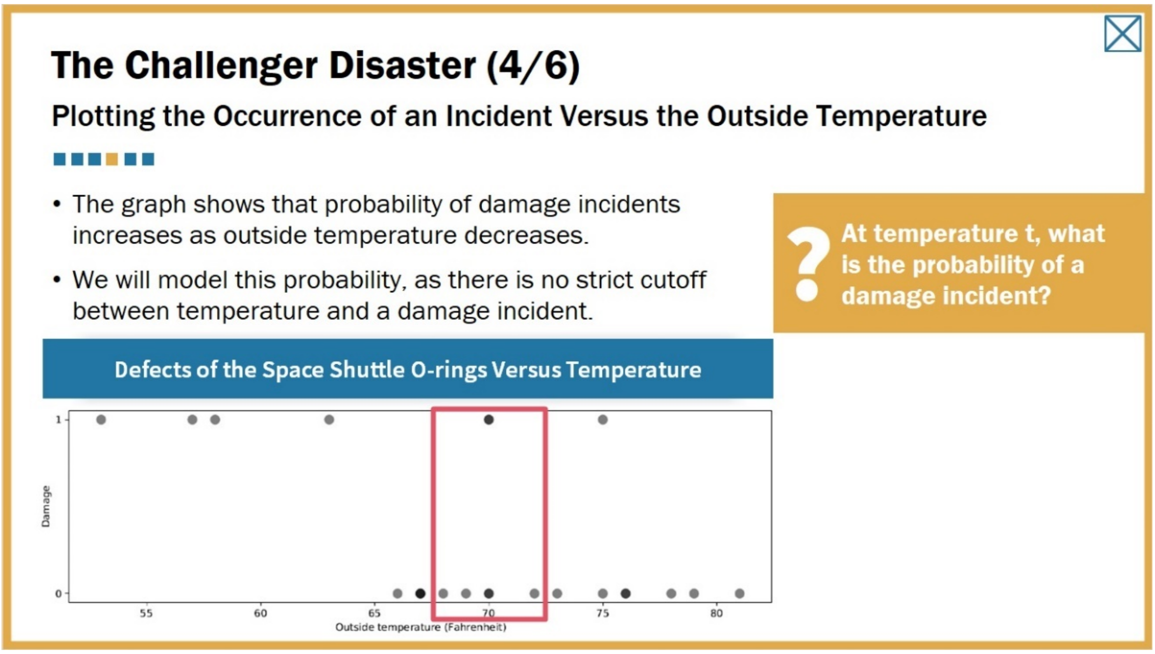
Rogers Commission Report

- The Commission observed the following major flaw in the NASA engineers conclusions:

> **The flights with zero incidents were excluded from the analysis because it was felt that these flights did not contribute any information about the temperature effect. However, the whole dataset clearly shows the correlation of O-ring damage in low temperature.**

The conclusion of the Rogers Commission was "the flights with zero incidents were excluded from the analysis because it was felt that these flights did not contribute any information about the temperature effect. However, the whole dataset clearly shows the correlation of O-ring damage in low temperature."

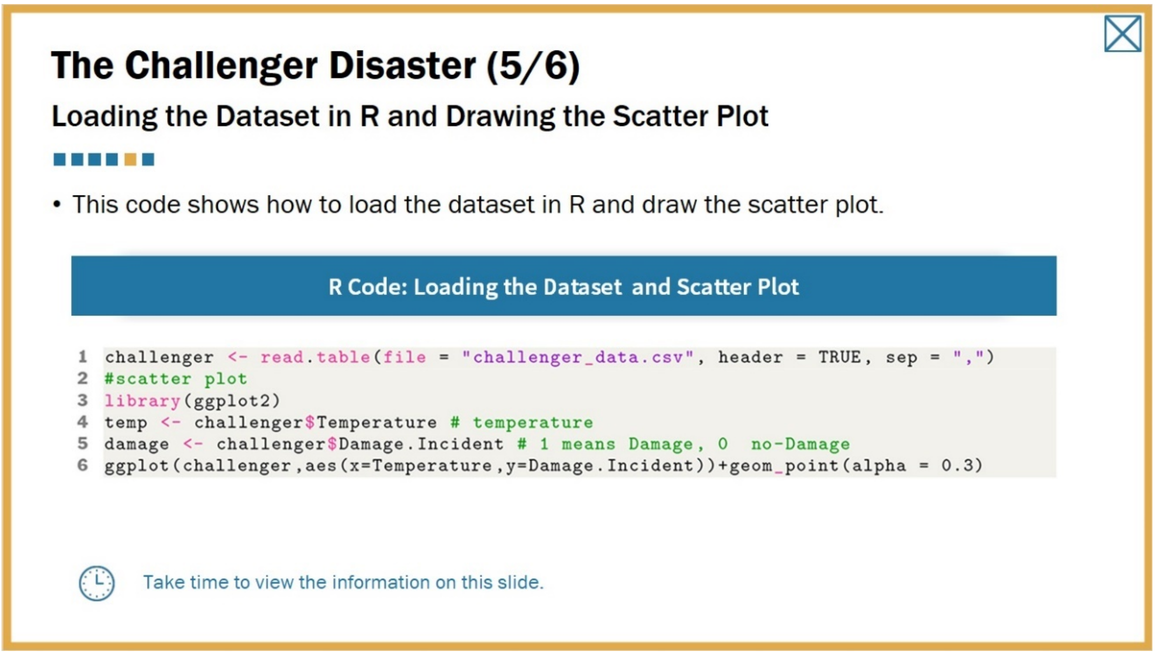
Tab 1.3: Plotting the Occurrence of an Incident Vs the Outside Temperature



We plot the occurrence of an incident versus the outside temperature to give you a rough idea of the relationship. It looks clear that the probability of damage incidents increases as the outside temperature decreases. We aim to model the probability because it does not look like there is a strict cutoff point between temperature and a damage incident, for instance note that at 70 degree we had two flights without damage and two flights with damage (roughly 50/50 chance).

The best we can do is ask “At temperature t, what is the probability of a damage incident? **We will see how to answer this question in the next slides.**

Tab 1.4: Loading the Dataset in R and Drawing the Scatter Plot

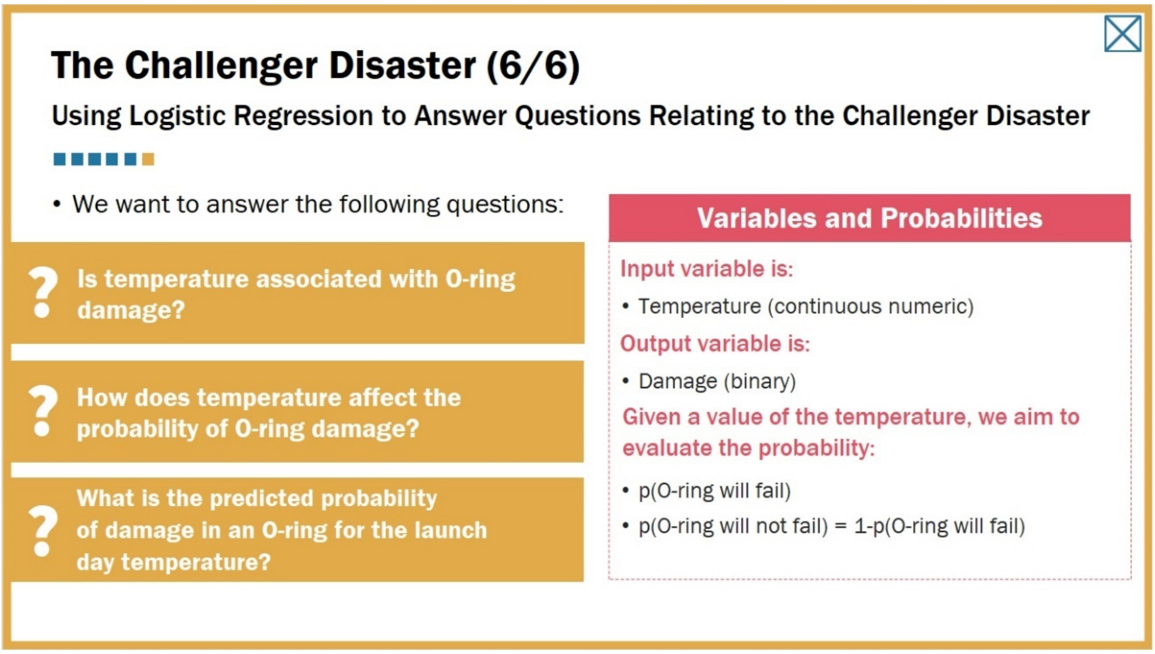


The following code shows how to load the dataset in R and draw the scatterplot.

```
challenger <- read.table(file = "challenger_data.csv", header = TRUE , sep = ",")
#scatter plot
library(ggplot2)
temp <- challenger$Temperature # temperature
damage <- challenger$Damage.Incident # 1 means Damage , 0 no-Damage
ggplot(challenger ,aes(x=Temperature ,y=Damage.Incident))+geom_point(alpha = 0.3)
```

Take time to view the information on this slide, then click next to continue.

Tab 1.5: Using Logistic Regression to Answer Questions Relating to the Challenger Disaster



Our goal is to address the following questions, with specific reference to the Challenger disaster:

1. Is the temperature associated with O-ring damage?

2. How is the temperature affecting the probability of O-ring damage?

3. What is the predicted probability of a damage in an O-ring for the temperature of the launch day?

The input variable in this case is

- temperature, which is a continuous (numeric) variable

The output variable is

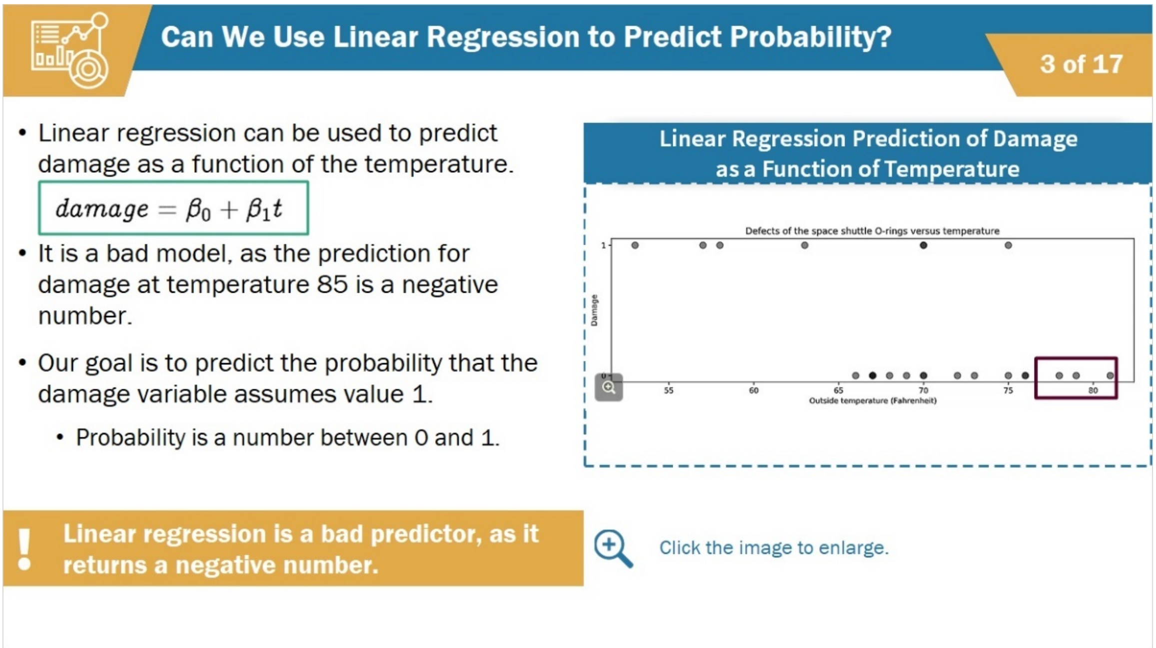
- damage, which is a binary (Yes 1, or No 0) variable

Given a value of the temperature we aim to evaluate the following probabilities:

- the probability that O-ring will fail

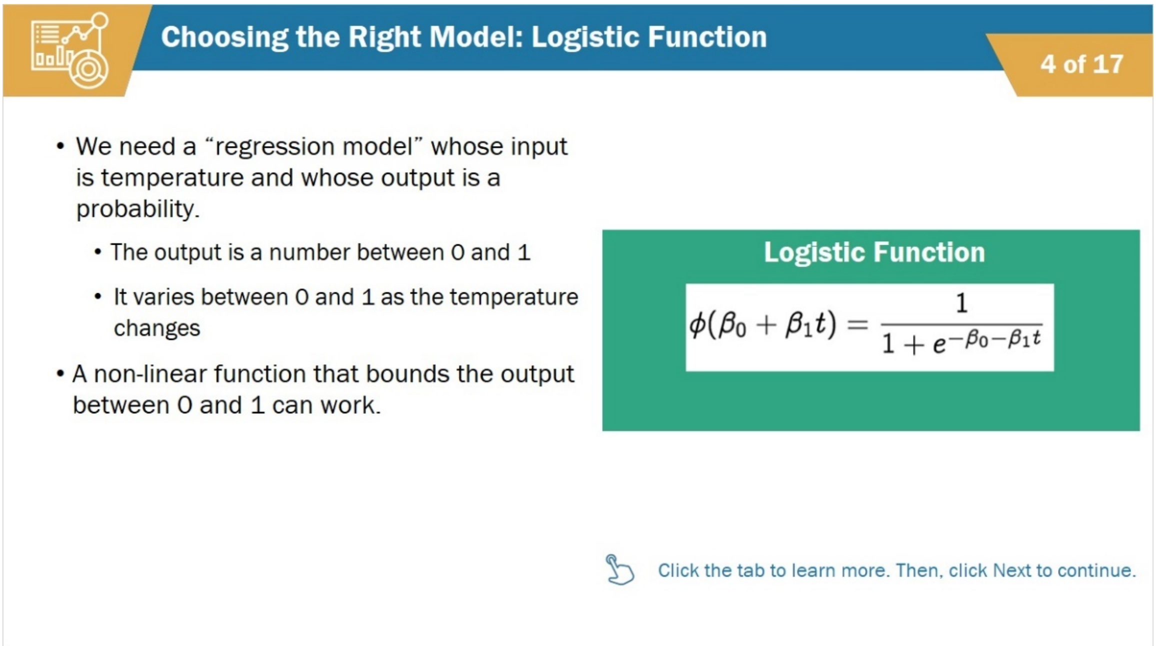
- the probability that O-ring won't fail, which is one minus the probability that O-ring will fail.

Slide 3: Can We Use Linear Regression to Predict Probability?



We could use linear regression to predict damage as a function of the temperature and the plot shows the resulting regression line. This is not a good model, note for instance that the prediction for damage at temperature 85 is a negative number! Our goal is to predict the probability of damage, that is the probability the damage variable assumes value one, and probability is a number between 0 and 1. So what linear regression returns is totally wrong: a probability cannot be negative.

Slide 4: Choosing the Right Model: Logistic Function

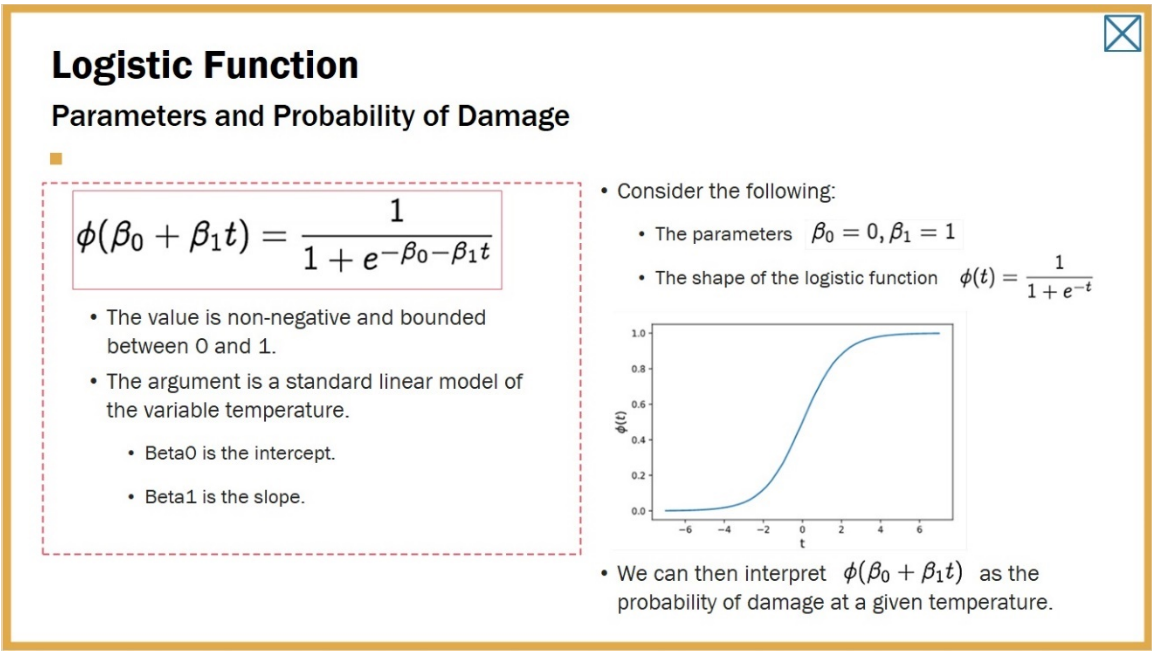


We need a “regression model” whose input is the temperature but whose output is a probability; that is, the output is a number between 0 and 1 and varies between 0 and 1 as the temperature changes.

We could squeeze our linear regression model through a nonlinear function that bounds the output between 0 and 1, for instance the logistic function does the job.

Click the tab to learn more about the logistic function. When you are ready, click next to continue.

Tab 1: **Slide 4, tab 1: Logistic Function**

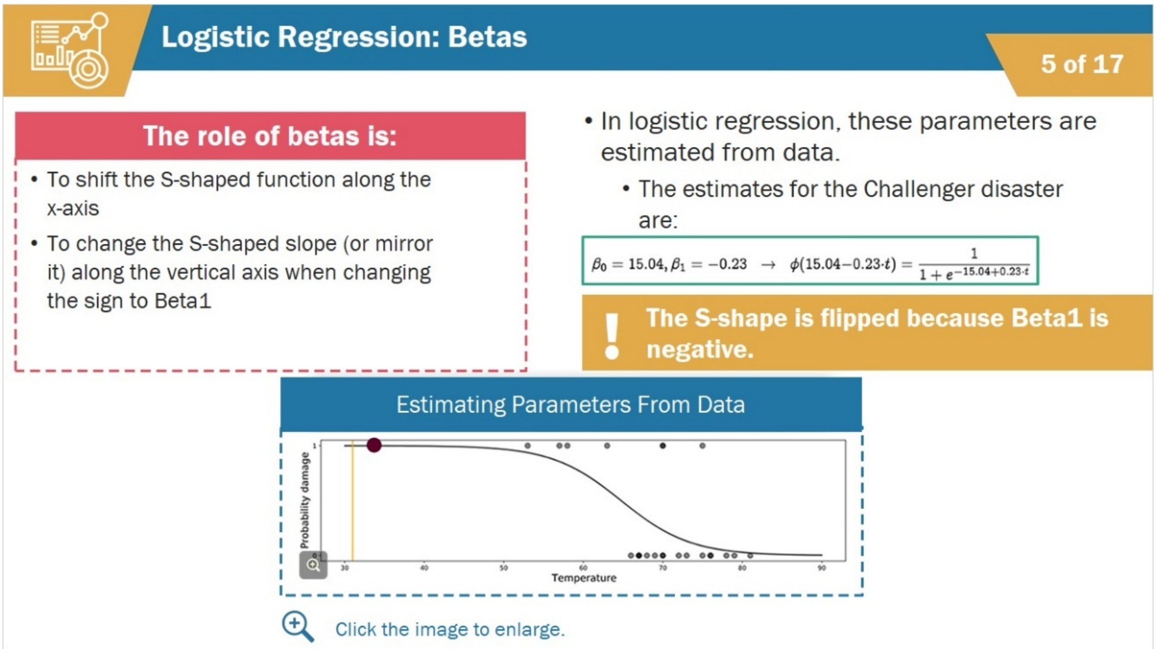


Note that the value of this function is nonnegative and bounded between 0 and 1. Note also that the argument of the function is a standard linear model of the variable temperature. Beta0 is the intercept and Beta1 is the slope.

For instance, consider the parameters for Beta0=0 and Beta1=1, and we can see the typical S-shape of the logistic function.

Given the output of the logistic function is a real number between 0 and 1, we can interpret it as the probability of a damage incident. For instance, from the plot, the probability of damage at temperature 0 is 0.5 and the probability of damage at temperature -6 is about 0.

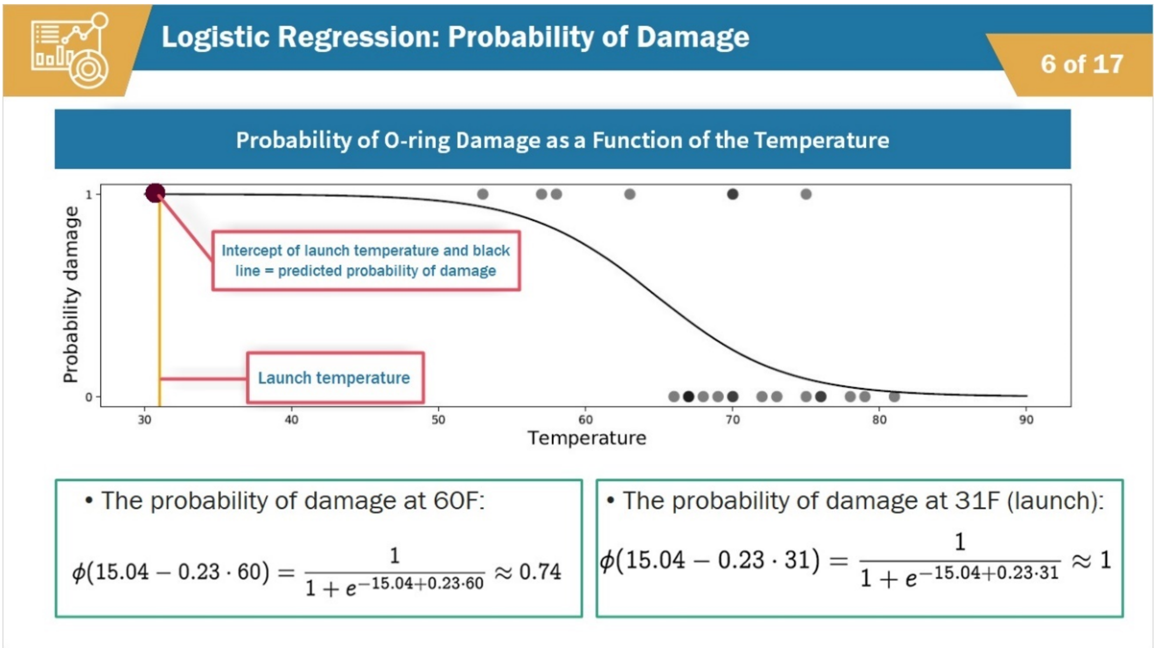
Slide 5: **Logistic Regression: Betas**



When thinking about logistic regression, we need to consider betas. What is their role? They allow us to shift the S-shaped function along the x-axis and to change its slope, or flip it around the vertical axis when we change the sign to Beta1. In logistic regression, we estimate these parameters from data. For instance, using the 23 data points from the Challenger dataset, the resulting estimates are Beta0=15.04 and Beta1=-0.23. We can use these values to compute the probability of damage at any temperature using the formula you see in the slides. We can also plot the function resulting in the black line in the plot.

Note that the S-shape is flipped, this happens because Beta1 is negative.

Slide 6: Logistic Regression: Probability of Damage

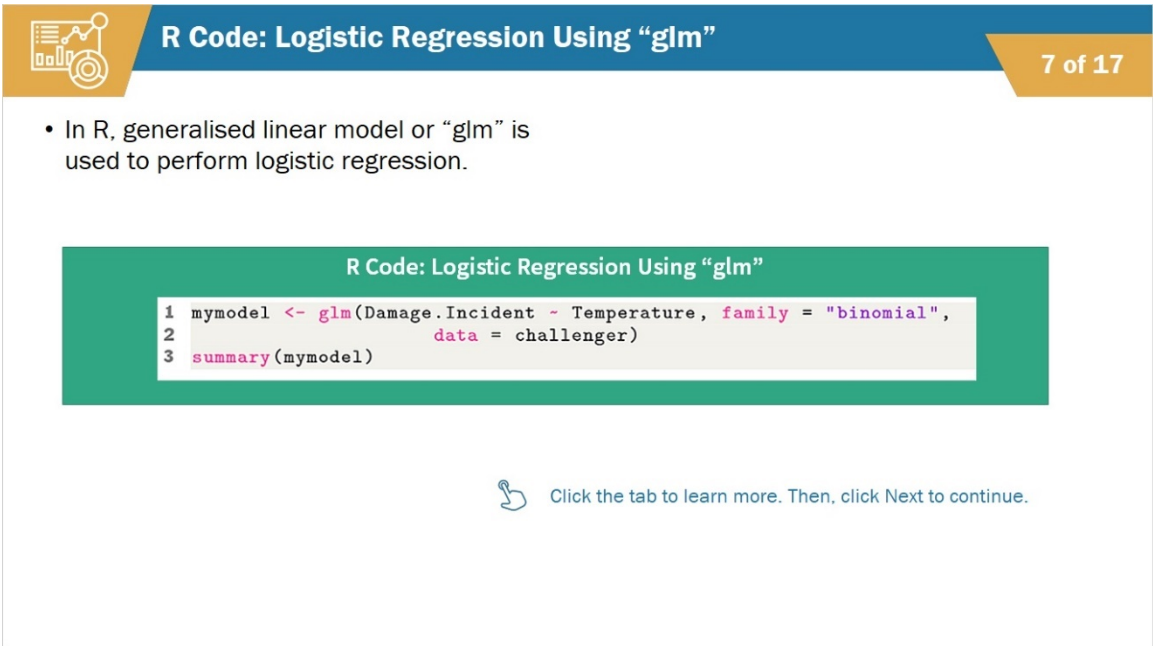


The continuous black line seen here is the probability of damage of the O-ring as a function of the temperature.

For instance, the probability of damage at 60F can be computed as shown in the first equation and it is equal to approximatively to 0.74.

The probability of damage at launch temperature (31F) is approximatively equal to 1, see the second equation. This means we predict damage with almost certainty. The orange vertical line represents the launch temperature and its intercept with the black line is the predicted probability of damage we have just computed.

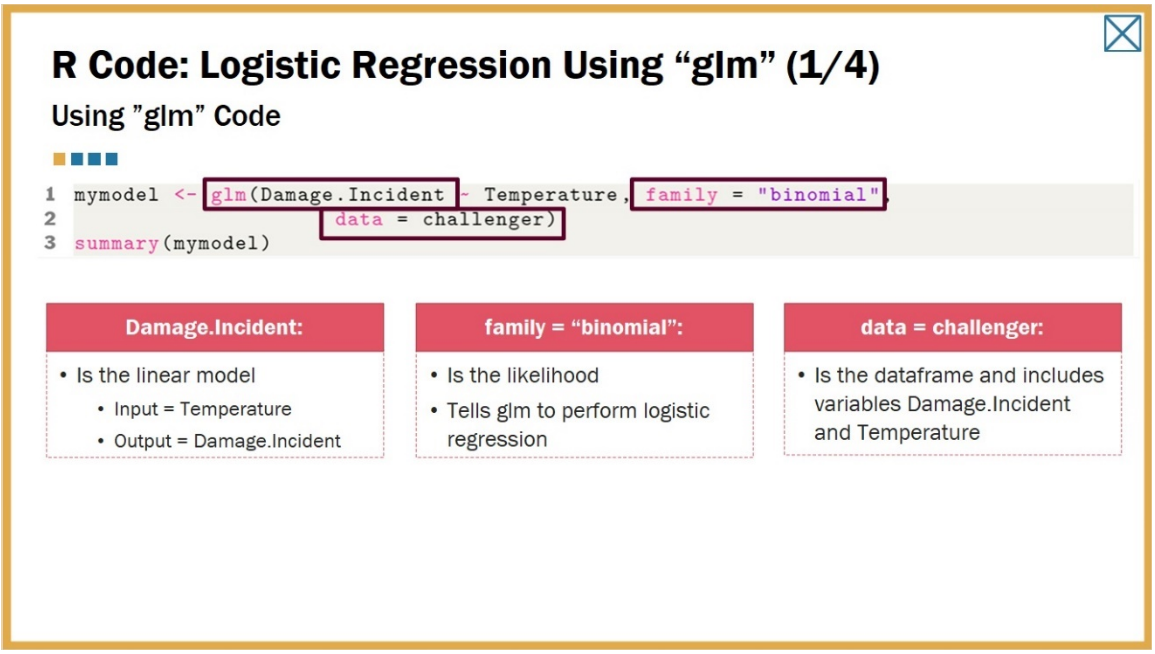
Slide 7: R Code: Logistic Regression Using “glm”



In R, we use glm to perform logistic regression. glm stands for generalised linear model.

Click the tab to learn more.

Tab 1: R Code: Logistic Regression Using "glm"




```
mymodel <- glm(Damage.Incident ~ Temperature , family = "binomial",
data = challenger)
summary(mymodel)
```

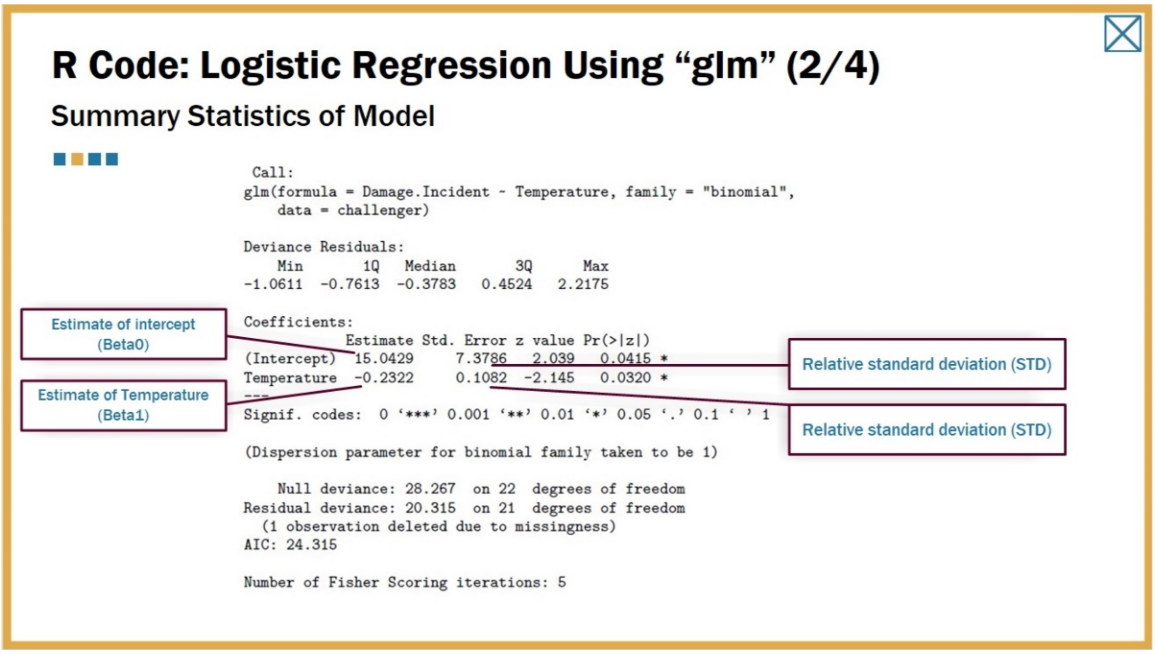
Note that the code here denotes:

Damage.Incident ~ Temperature denotes the linear model, input is temperature and output is Damage.Incident

family = "binomial" is the likelihood (as we will see in the next slides). By selecting binomial we tell glm to perform logistic regression.

data = challenger is the dataframe which includes the variables Damage.Incident, and Temperature

Tab 1.1: Summary Statistics of Model



Call:

glm(formula = Damage.Incident ~ Temperature, family = "binomial",

data = challenger)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.0611 -0.7613 -0.3783 0.4524 2.2175

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) 15.0429 7.3786 2.039 0.0415 *

Temperature -0.2322 0.1082 -2.145 0.0320 *

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 28.267 on 22 degrees of freedom

Residual deviance: 20.315 on 21 degrees of freedom

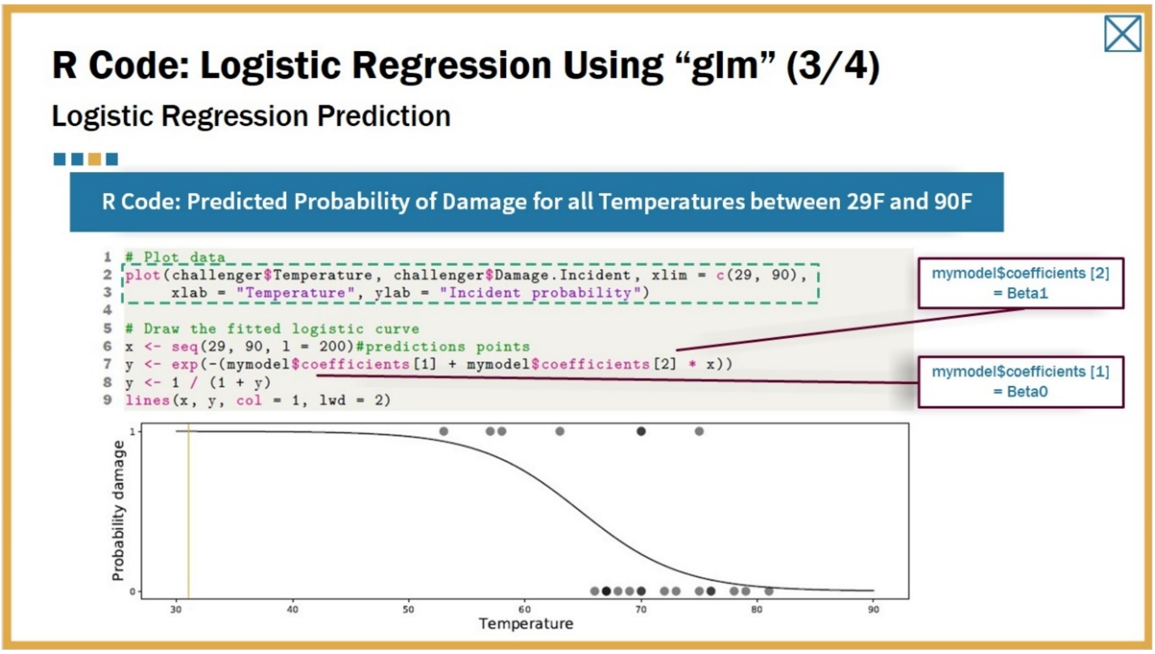
(1 observation deleted due to missingness)

AIC: 24.315

Number of Fisher Scoring iterations: 5

We also show the summary statistics of the model. For the moment, just note the Estimate of Intercept (Beta0, 15.0429) and Temperature (Beta1, -0.2322) and relative standard deviations (Std. Error, 7.3786 and 0.1082). We will go back to the other statistics, in particular deviance, later on.

Tab 1.2: Logistic Regression Prediction

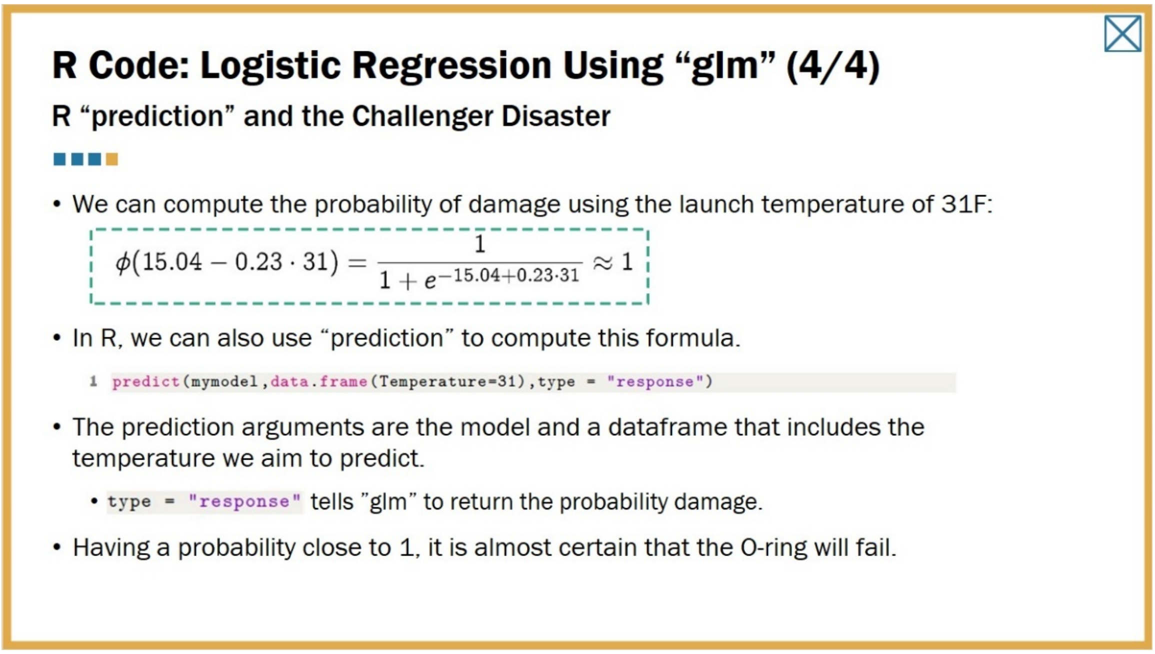


The following code plots the predicted probability of damage for all the temperatures between 29F and 90F. Note that mymodel$coefficients [1] is Beta0 and mymodel$coefficients [2] is Beta1.

```
# Plot data
plot(challenger$Temperature , challenger$Damage.Incident , xlim = c(29, 90),
xlab = "Temperature", ylab = "Incident probability")

# Draw the fitted logistic curve
x <- seq(29, 90, l = 200)#predictions points
y <- exp(-(mymodel$coefficients [1] + mymodel$coefficients [2] * x))
y <- 1 / (1 + y)
lines(x, y, col = 1, lwd = 2)
```

Tab 1.3: R "Prediction" and the Challenger Disaster



The temperature of the launch was 31F, and the probability of damage can be computed using the following formula.

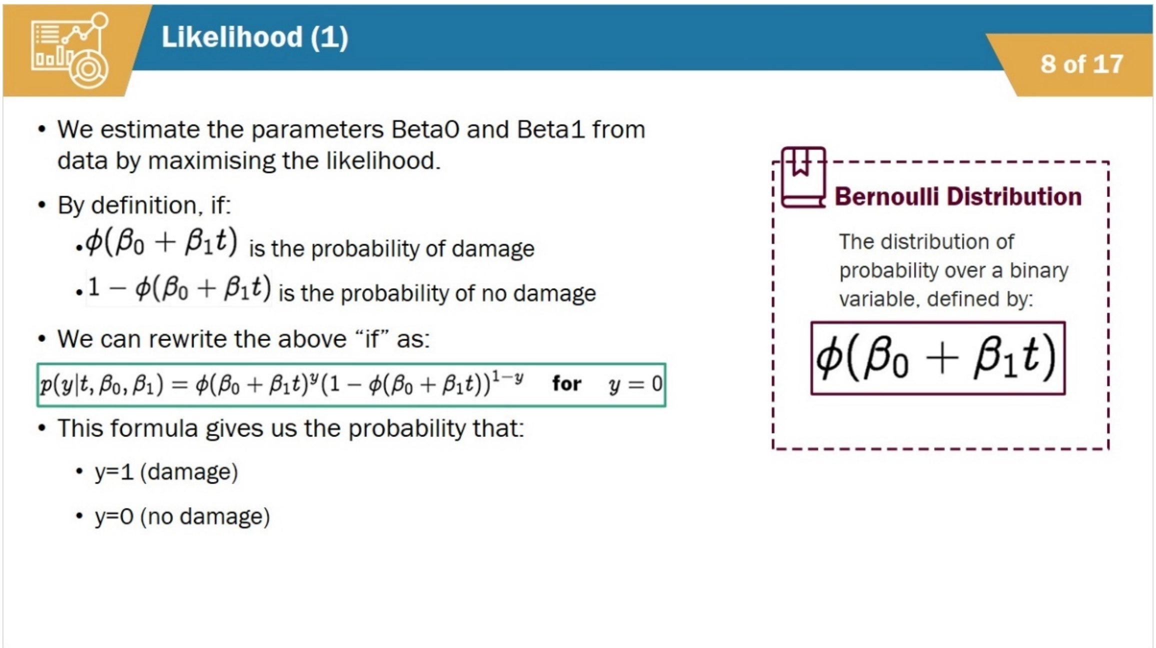
In R we can also use "prediction" which computes the above formula for us.

```
1 predict(mymodel ,data.frame(Temperature =31),type = "response")
```

Note that the arguments of prediction are the model and a dataframe including the value of the temperature we aim to predict. type = "response" tells glm to return the value of the response, that is the probability of damage.

The value of the probability close to one means we are practically certain that the O-ring will fail. We can sadly conclude that the disaster could be avoided!

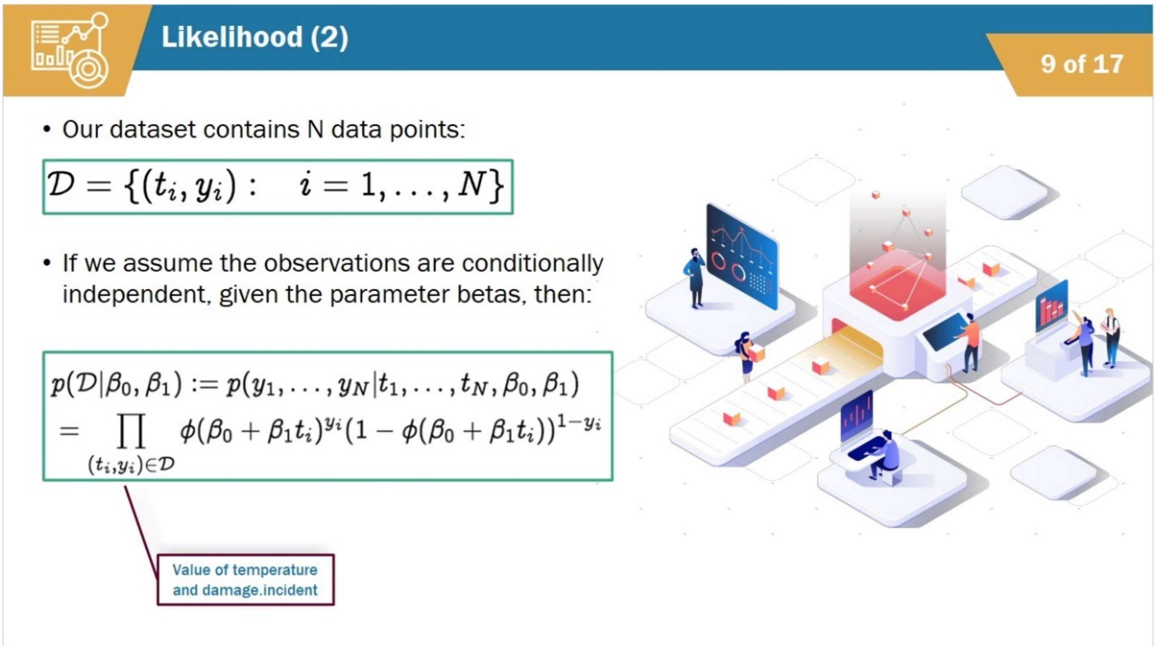
Slide 8: **Likelihood (1)**



How do we estimate the parameters Beta0 and Beta1 from data? We do it by maximising the likelihood. By definition Phi(beta0+beta1*t) is the probability of damage and so 1-Phi is the probability of no-damage. We can rewrite these two probabilities in a single row as showed in the formula.

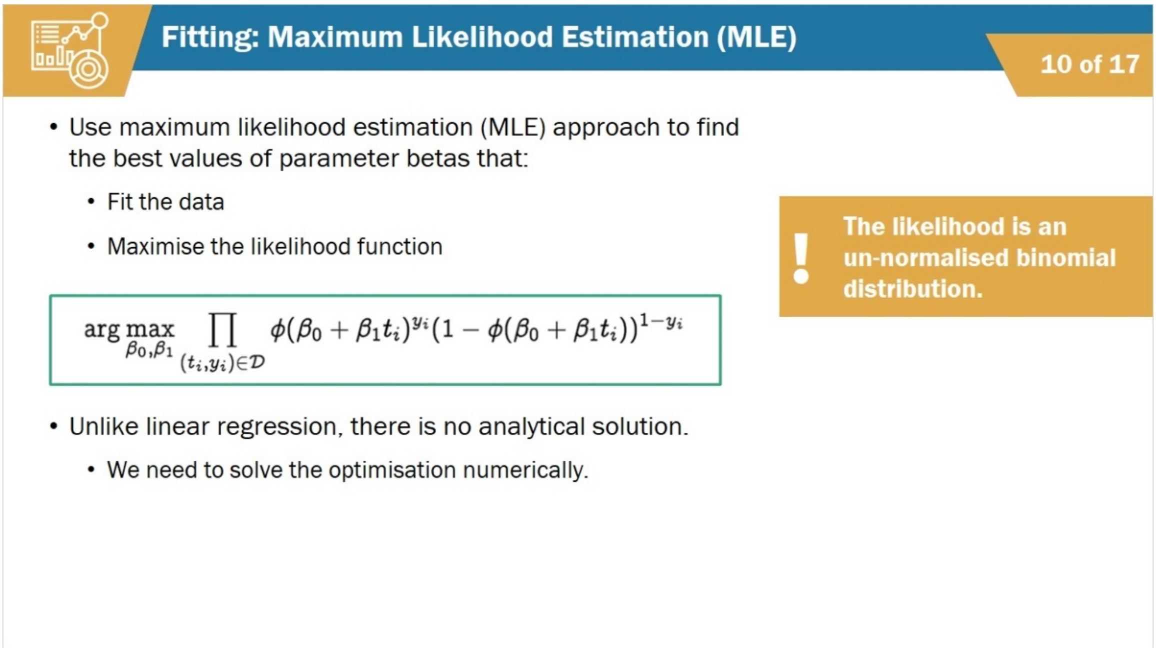
Here y is a binary variable representing damage. When y=1, we obtain Phi(beta0+beta1*t) and when y=0 we obtain 1-Phi(beta0+beta1*t). Therefore, the formula gives us the probability that y=1, that is damage and the probability that y=0, that is no-damage. This distribution of probability over a binary variable, defined by the probability Phi(beta0+beta1*t), is called Bernoulli distribution.

Slide 9: **Likelihood (2)**



We have N data points in our dataset. If we assume the observations are conditionally independent given the parameters beta then the likelihood of the whole dataset is a product of Bernoulli distribution resulting in the following probability. Note that t_i and y_i denotes the value of the temperature and damage.incident in the i-th row of the dataset.

Slide 10: Fitting: Maximum Likelihood Estimation (MLE)

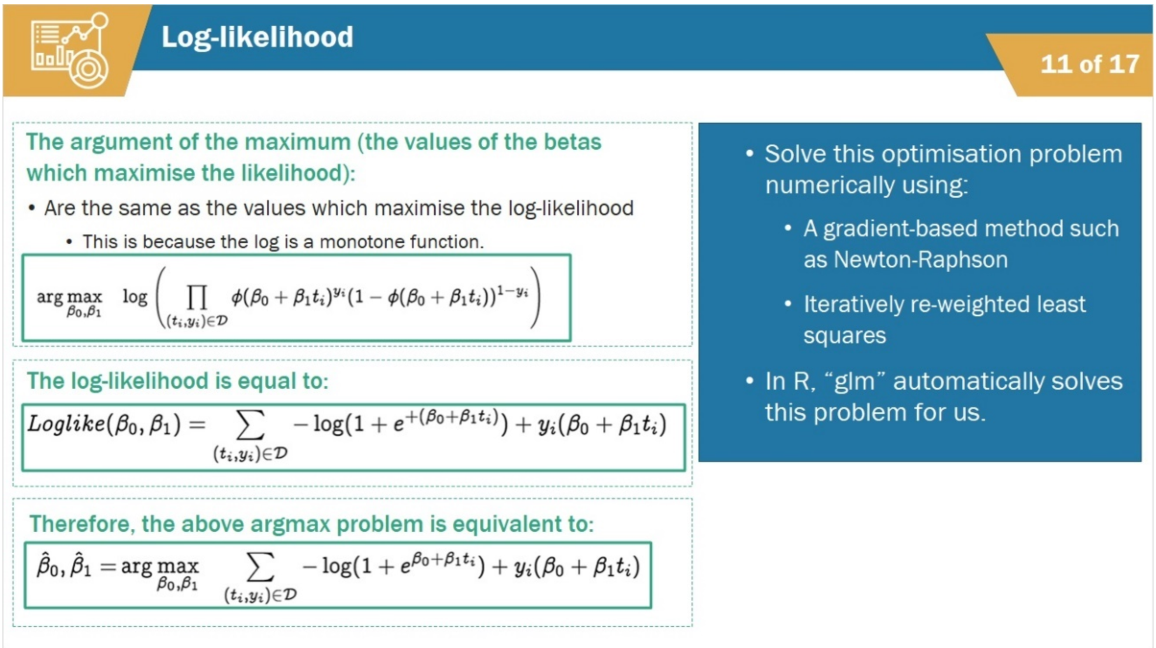


According to the Maximum Likelihood Estimation approach, we find the values of the Betas which maximise the likelihood function. These are the values which are mostly in agreement with the observed dataset.

We aim to find the best values of parameters betas that fit the data by maximising the likelihood function. Note that the likelihood is an unnormalised Binomial distribution. We will denote the estimated parameters with hat betas.

Contrarily to linear regression, there is no analytical solution, we need to solve that optimisation numerically. We will see how to do that from scratch in another session.

Slide 11: **Log-likelihood**



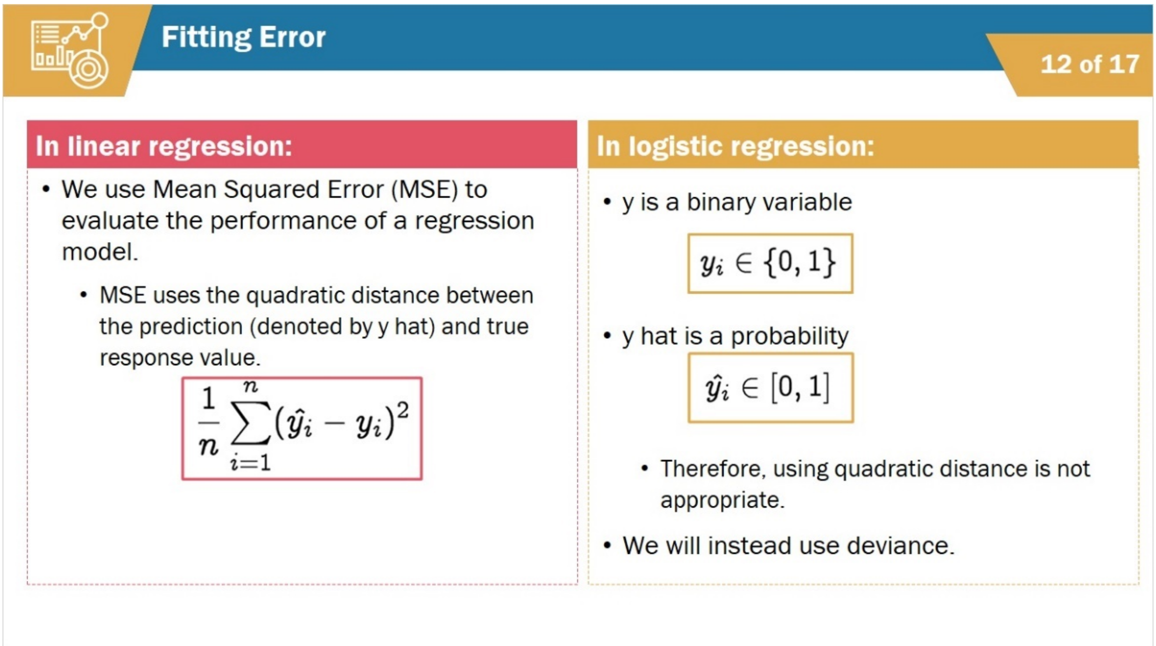
The argument of the maximum, that is the values of the betas which maximise the likelihood, are the same as the values which maximise the log-likelihood, because the log is a monotone function.

You can easily verify that the log-likelihood is equal to the equation in the middle.

So, the above argmax problem is equivalent to the third equation.

This optimisation problem can be solved numerically using a gradient-based method (such as Newton-Raphson) or, equivalently, using iteratively reweighted least squares. In R, “glm” automatically solves this problem for us.

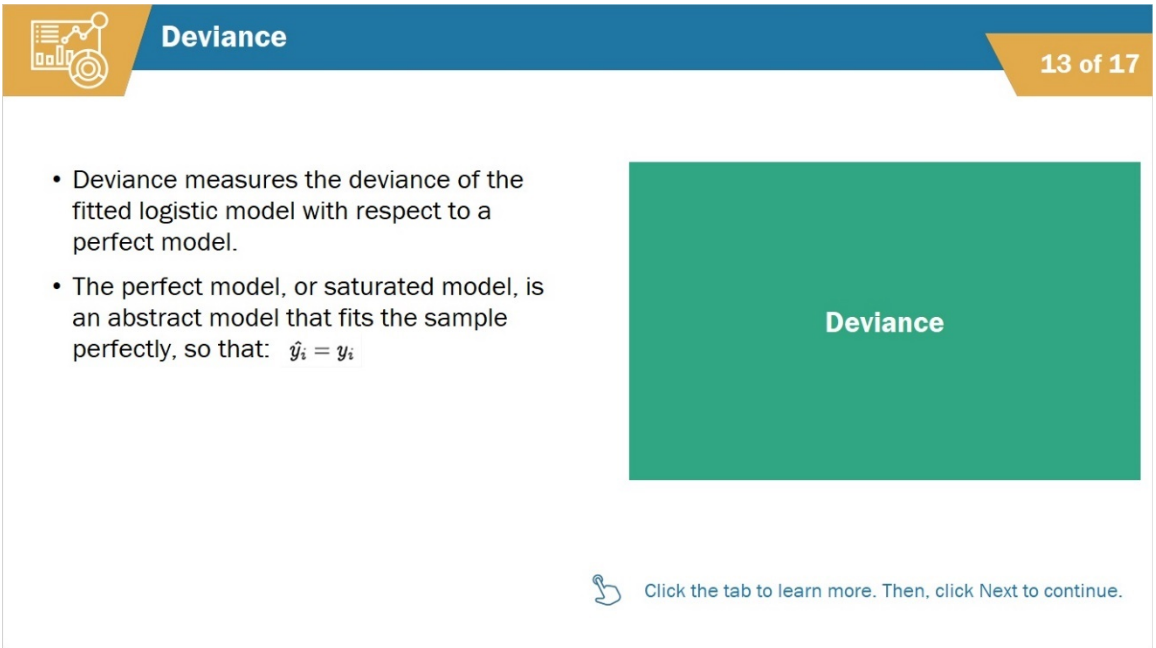
Slide 12: **Fitting Error**



In linear regression, we use Mean Squared Error (MSE) to evaluate the performance of a regression model. MSE uses the quadratic distance between the prediction (denoted by y hat) and true value.

For logistic regression, y is a binary variable and y hat is a probability. Therefore, the quadratic distance is not appropriate, we will instead use another performance metric called deviance.

Slide 13: Deviance



Deviance measures the deviance of the fitted logistic model with respect to a perfect model. The perfect model, known as the saturated model, denotes an abstract model that fits perfectly the sample, this is, the model such that y is equal to y hat.

Click the tab to learn more about deviance. When you are ready, click next to continue.

Tab 1: Deviance

Deviance (1/6)

Deviance: Introduction

- Deviance is always greater than, or equal to, zero.

$$2\sum_{i=1}^{N} y_i \log\left(\frac{y_i}{\hat{y}_i}\right) + (1-y_i)\log\left(\frac{1-y_i}{1-\hat{y}_i}\right)$$

- It is only zero if the fit is perfect.
  - The best model is not necessarily the one with zero deviance.
  - Zero deviance is not usually attainable.

- Deviance is useful:
  - For checking model fitting
  - To statistically compare models

The deviance is always larger or equal than zero, being zero only if the fit is perfect. As for MSE in linear regression, the best model is not necessarily the model which has zero deviance and also zero deviance is not usually attainable.

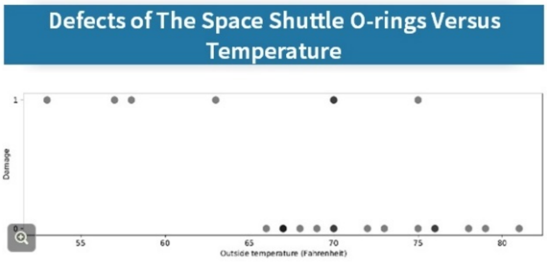
The deviance is useful to check model fitting and then statistically compare models as we will see in the next slides.

Tab 1.1: Deviance Example: Challenger

Deviance (2/6)

Deviance Example: Challenger

- Consider the flight temperature data of 70F.
  - Two flights had a damage incident.
  - Two flights had no damage incident.
- If only considering these four cases, the minimum deviance is obtained by the model that predicts $\hat{y}_i = 0.5$

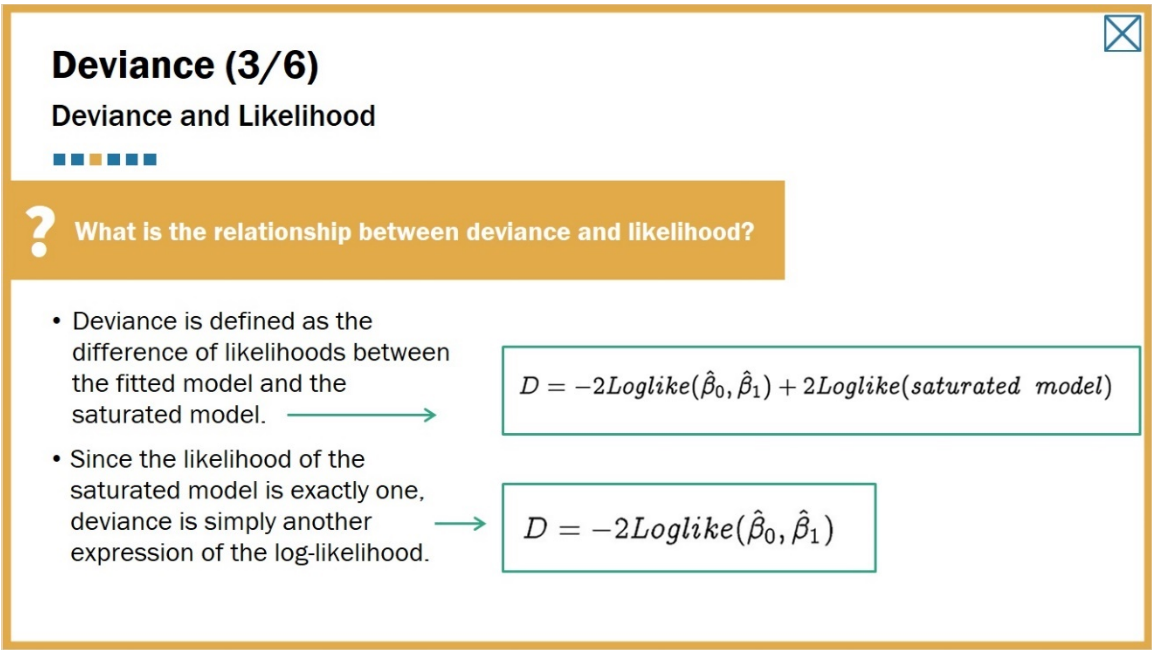


Click the image to enlarge.

For instance, consider the flight temperature data of 70F. There were two flights with damage incident and two flights with no damage incident.

If we consider only these 4 cases, the minimum deviance is obtained by the model that predicts 50% probability of damage.

Tab 1.2: Deviance and Likelihood

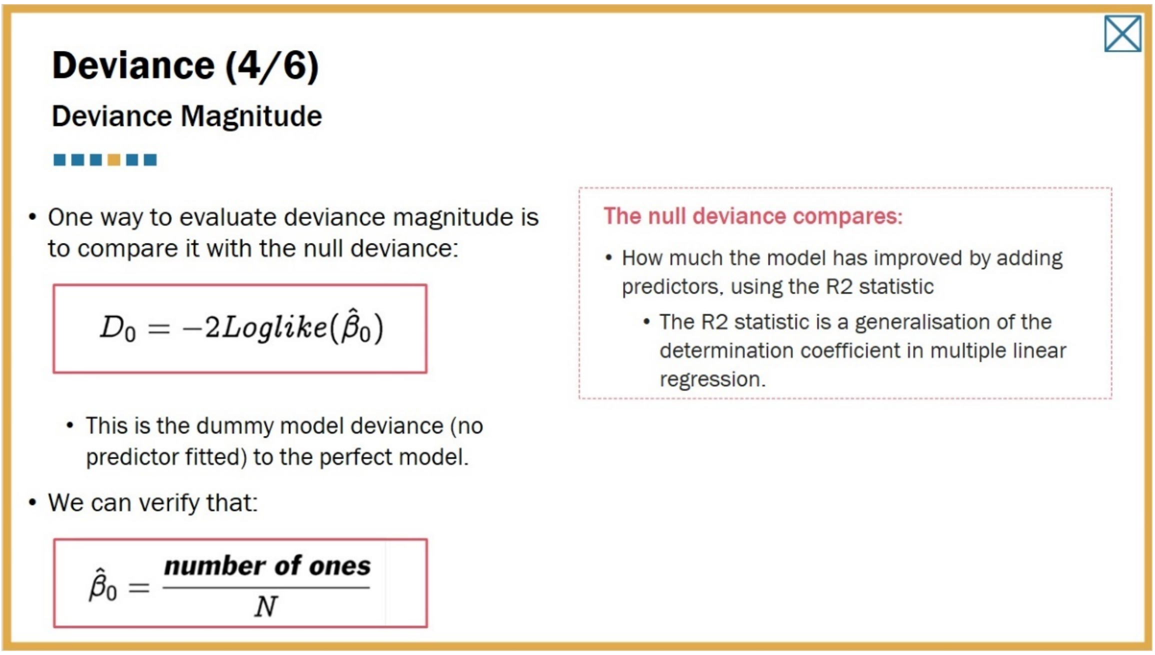


What is the relationship between deviance and likelihood?

As I mentioned earlier, deviance is defined as the difference of likelihoods between the fitted model and the saturated model, You can in fact verify that the deviance is equal to the formula shown here.

Since the likelihood of the saturated model is exactly one (why? Can you prove it!), then the deviance is simply another expression of the log-likelihood.

Tab 1.3: Deviance Magnitude



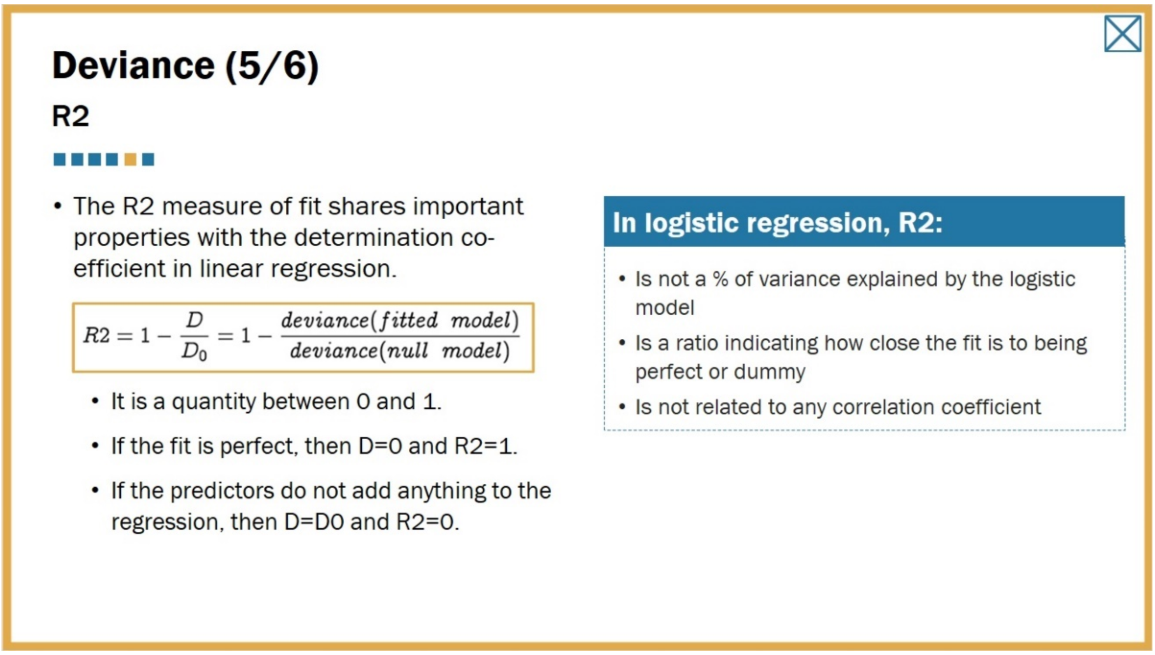
A way for evaluating the magnitude of the deviance is to compare it with the null deviance, see this formula

which is the deviance of the dummy model (the one fitted without any predictor) to the perfect model.

In this case, we can easily verify that Beta0 is simply equal to the number of ones on the dataset (for the Challenger disaster is the number of damages) divided by the number of all data points which was N=23

The null deviance serves for comparing how much the model has improved by adding the predictors. This can be done by means of the R2 statistic, which is a generalization of the determination coefficient in multiple linear regression.

Tab 1.4: Slide 13, tab 1, branch 5: R2



R2 is a measure of fit which shares some important properties with the determination coefficient in linear regression:

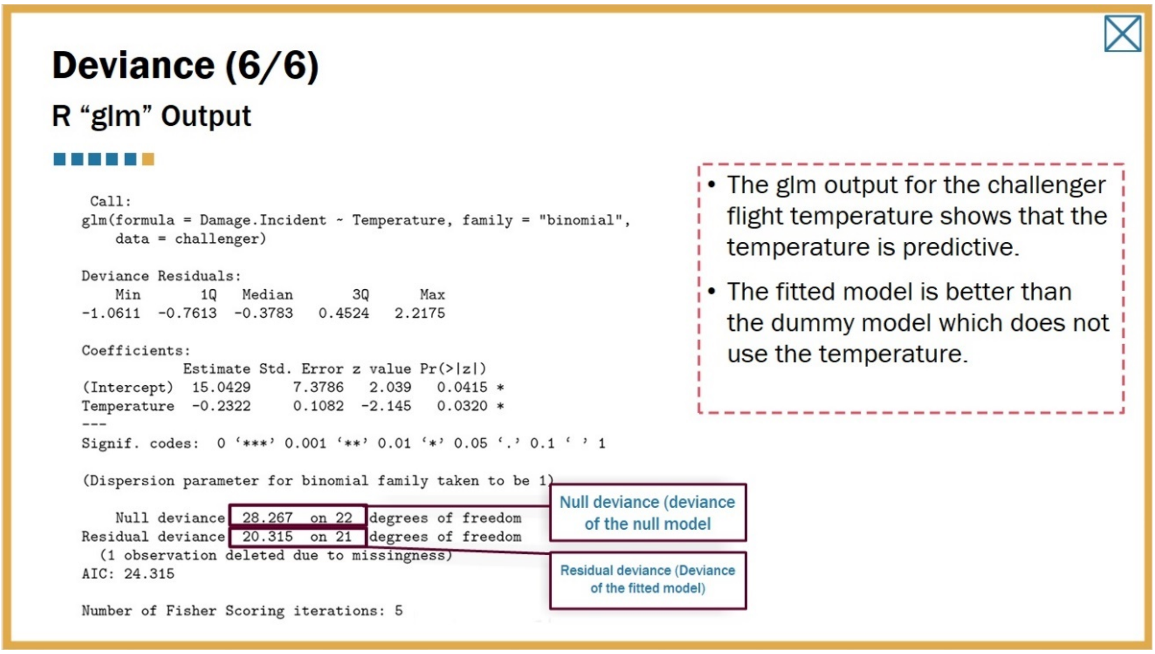
- It is a quantity between 0 and 1

- If the fit is perfect, then D=0 and R2=1.

-If the predictors do not add anything to the regression, then D=D0 and R2=0.

However, note that, for logistic regression, R2 is not the percentage of variance explained by the logistic model, but rather a ratio indicating how close is the fit to being perfect or dummy. It is not related to any correlation coefficient.

Tab 1.5: **Slide 13, tab 1, branch 6: R “glm” Output**



Here we see the output of glm for the challenger data.

Call:

glm(formula = Damage.Incident ~ Temperature, family = "binomial",

data = challenger)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.0611 -0.7613 -0.3783 0.4524 2.2175

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) 15.0429 7.3786 2.039 0.0415 *

Temperature -0.2322 0.1082 -2.145 0.0320 *

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 28.267 on 22 degrees of freedom

Residual deviance: 20.315 on 21 degrees of freedom

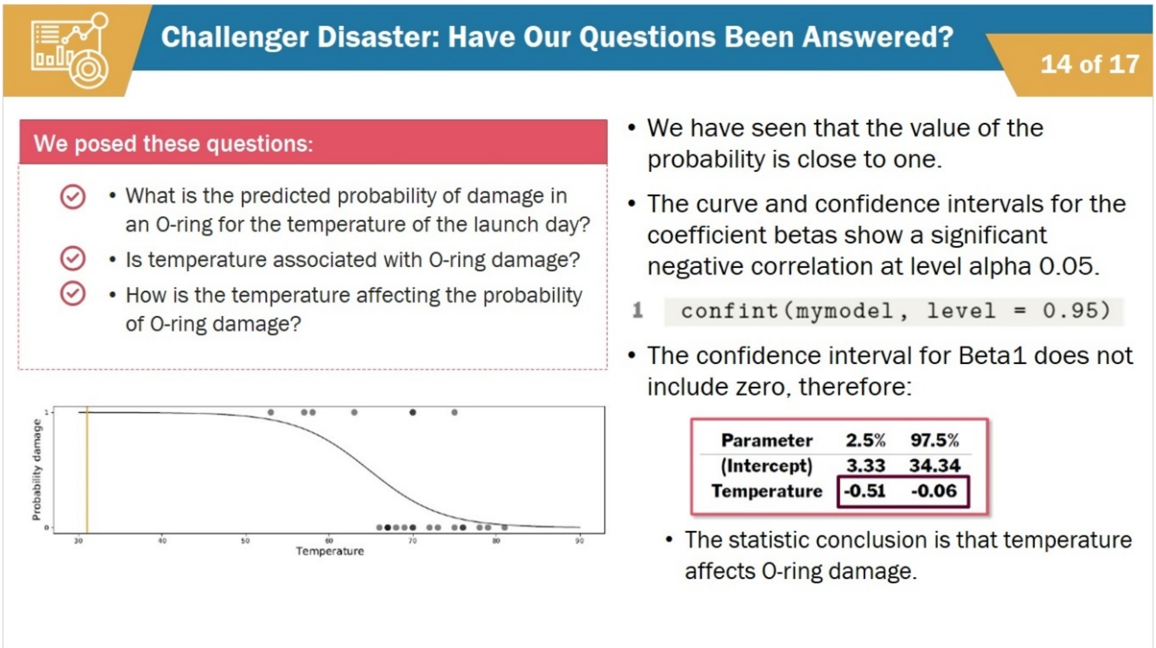
(1 observation deleted due to missingness)

AIC: 24.315

Number of Fisher Scoring iterations: 5

At the bottom you can see ``Residual deviance'' which is what we called deviance of the fitted model and ``Null deviance'' which is the deviance of the null model. Given that 20.315 is less than 28.267, we can conclude that the temperature is indeed predictive. The fitted model is better than the dummy model which does not use the temperature.

Slide 14: Challenger Disaster: Have Our Questions Been Answered?



At the beginning of this presentation we asked three questions. As we have seen that the value of the probability is close to one. We have now answered the question “what is the predicted probability of damage in an O-ring for the temperature of the launch day?” but we will now answer the other two questions

Is the temperature associated with O-ring damage?

How is the temperature affecting the probability of O-ring damage?

more formally.

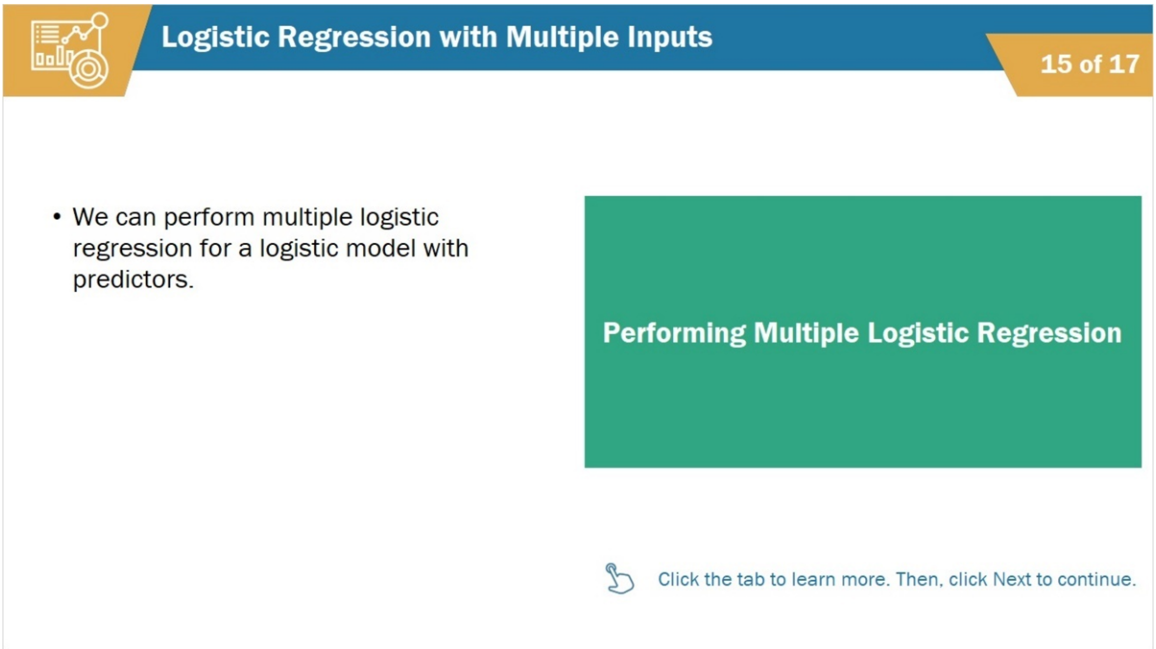
Take a look at This curve and also the confidence intervals for the coefficients betas which show a significant negative correlation at level alpha 0.05.

```
1 confint(mymodel , level = 0.95)
```

We can compute it in R using the function **confint**

Note that the confidence interval for Beta1 (Temperature) does not include zero and, therefore, we can statistically conclude that temperature affects O-ring damage.

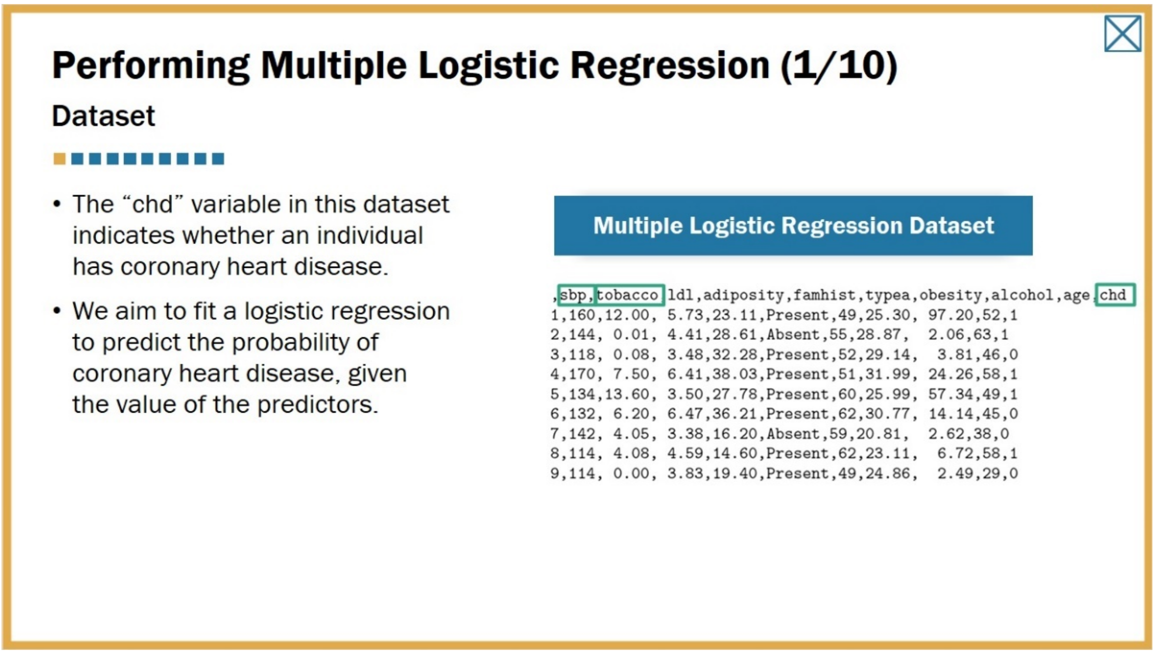
Slide 15: **Logistic Regression with Multiple Inputs**



Just as we were able to perform multiple linear regression for a linear model with multiple predictors, we can perform multiple logistic regression for a logistic model with predictors. Over the following slides we will go through this process using one particular dataset.

Click the tab to learn more about performing multiple logistic regression.

Tab 1: **Performing Multiple Logistic Regression: Dataset**



,sbp,tobacco,ldl,adiposity,famhist,typea,obesity,alcohol,age,chd

1,160,12.00, 5.73,23.11,Present,49,25.30, 97.20,52,1

2,144, 0.01, 4.41,28.61,Absent,55,28.87, 2.06,63,1

3,118, 0.08, 3.48,32.28,Present,52,29.14, 3.81,46,0

4,170, 7.50, 6.41,38.03,Present,51,31.99, 24.26,58,1

5,134,13.60, 3.50,27.78,Present,60,25.99, 57.34,49,1

6,132, 6.20, 6.47,36.21,Present,62,30.77, 14.14,45,0

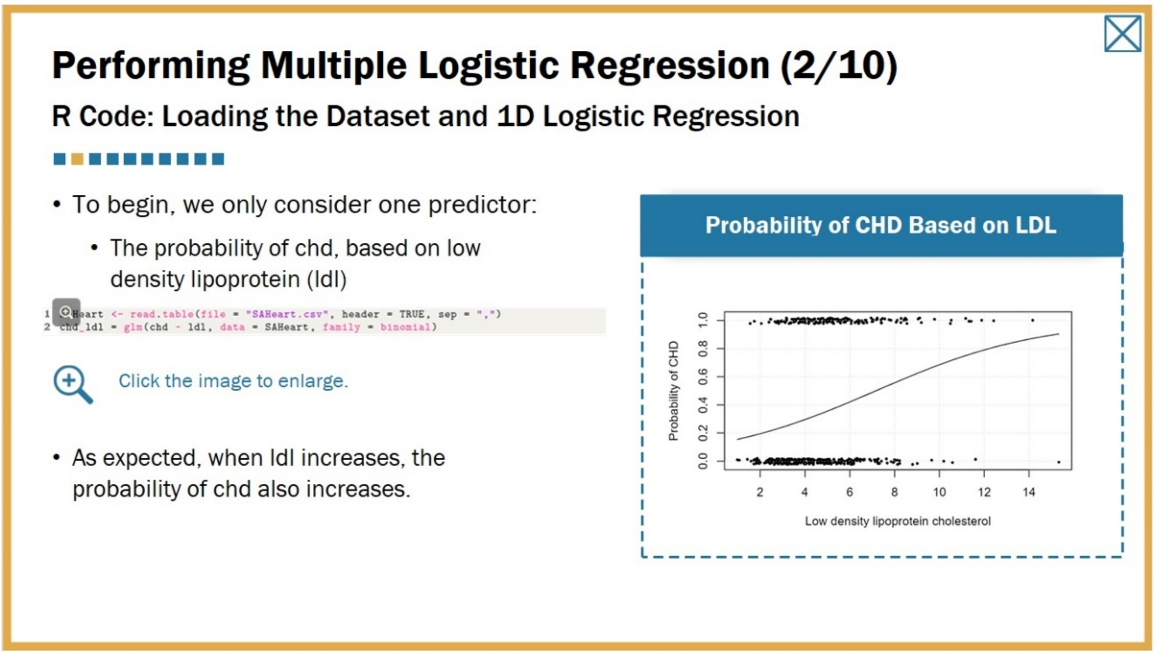
7,142, 4.05, 3.38,16.20,Absent,59,20.81, 2.62,38,0

8,114, 4.08, 4.59,14.60,Present,62,23.11, 6.72,58,1

9,114, 0.00, 3.83,19.40,Present,49,24.86, 2.49,29,0

Consider for instance this dataset where the “chd” variable indicates whether or not coronary heart disease is present in an individual. We have multiple predictors (sbp, tobacco..). We aim to fit a logistic regression model to predict the probability of “coronary heart disease” given the values of the predictors.

Tab 1.1: R Code: Loading the Dataset And 1D Logistic Regression

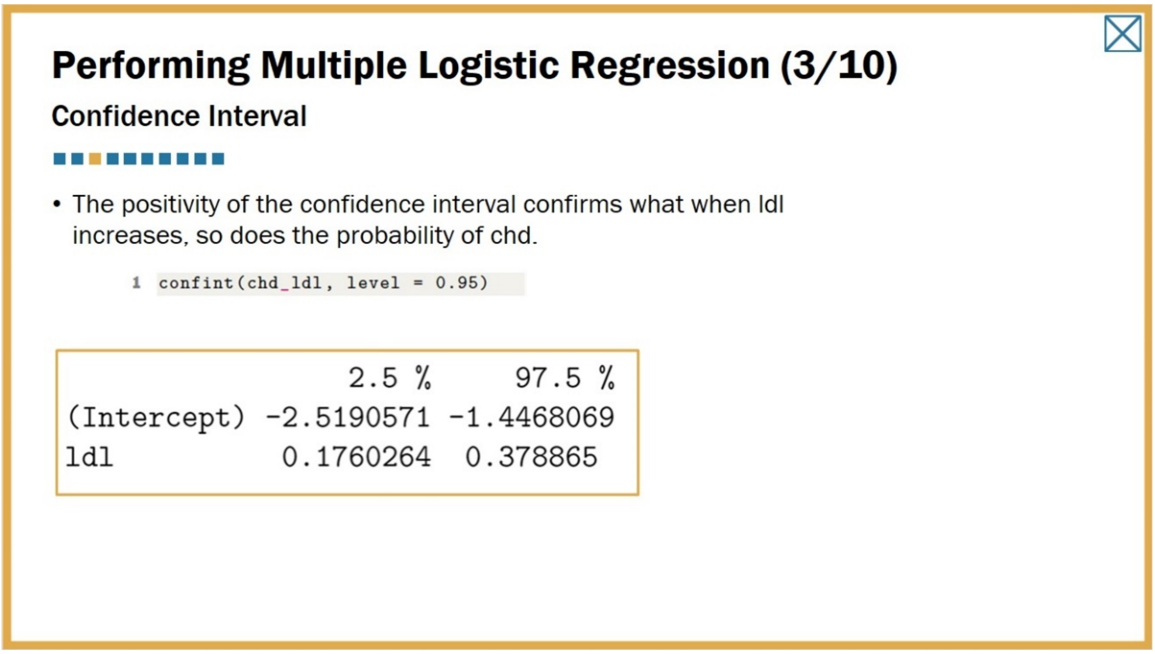


We begin by considering one predictor only. In particular, we model the probability of coronary heart disease based only on low density lipoprotein cholesterol (ldl). This is the R code.

```
SAHeart <- read.table(file = "SAHeart.csv", header = TRUE , sep = ",")
chd_ldl = glm(chd ~ ldl , data = SAHeart , family = binomial)
```

As one would expect, when ldl increases so does the probability of chd.

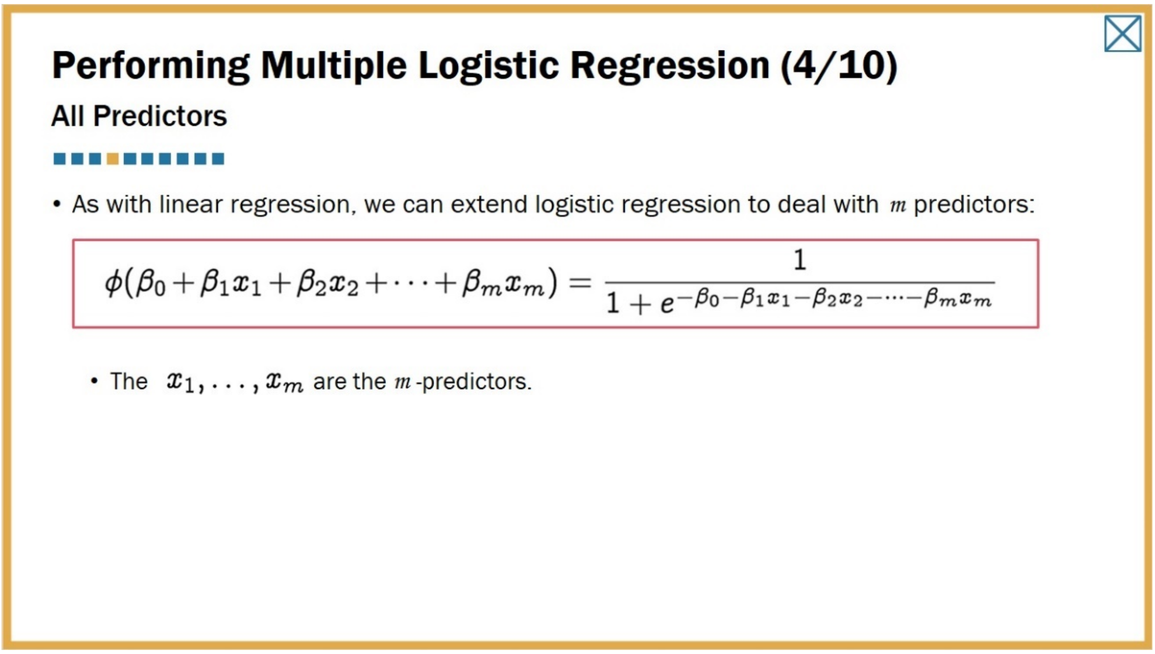
Tab 1.2: Slide 15, Tab 1, branch 3: Confidence Interval



The positivity of the confidence interval confirms that when ldl increases so does the probability of chd.

1 confint(chd_ldl , level = 0.95)

Tab 1.3: All Predictors



We can extend logistic regression to deal with m predictors as for linear regression. We simply squeeze a multiple linear regression model through the logistic function. The xs are the m predictors.

You see it is like having a multiple linear regression model inside the logistic function.

Tab 1.4: R Code: All Predictors

Performing Multiple Logistic Regression (5/10)

R Code: All Predictors

- In R, we can create a logistic regression model that considers all the predictors.

```
chd_all = glm(chd ~ ., data = SAHeart , family = binomial)
```

- Although we cannot plot the line (as we are solving a multi-dimensional problem), we can make predictions.

```
predict(chd_all,SAHeart[1:3,],type = "response")
```

- Note type = “response” returns the probability of chd for the dataset’s first three rows.

	,sbp,tobacco,ldl,adiposity,famhist,typea,obesity,alcohol,age,chd
1 0.7121829	1,160,12.00, 5.73,23.11,Present,49,25.30, 97.20,52,1
2 0.3310109	2,144, 0.01, 4.41,28.61,Absent,55,28.87, 2.06,63,1
3 0.2809570	3,118, 0.08, 3.48,32.28,Present,52,29.14, 3.81,46,0

In R we can create a logistic regression model using all the predictors by simply using dot inside glm. “chd ~ .” means use all predictors.

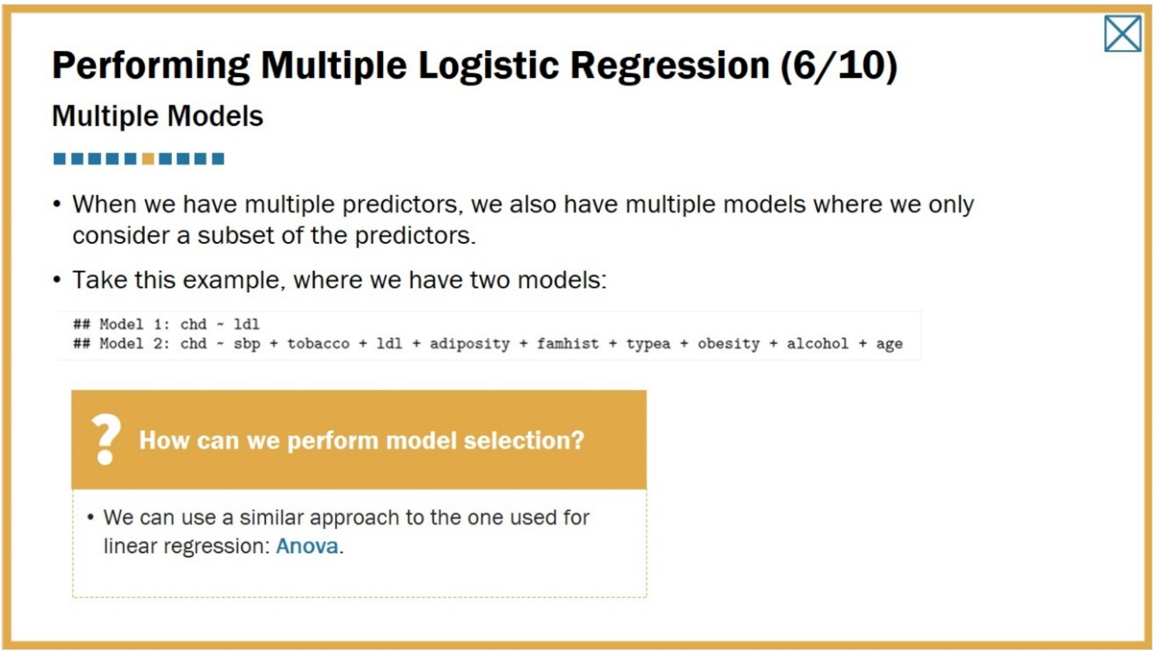
```
chd_all = glm(chd ~ ., data = SAHeart , family = binomial)
```

In this case we cannot plot the regression line because we are considering a multi-dimensional problem, but we can make predictions.

```
predict(chd_all ,SAHeart[1:3,],type = "response")
```

Note type = “response” which returns the probability of chd for the first three rows (adults) in the dataset. By changing the data frame part, we can make any predictions.

Tab 1.5: Multiple Models



When we have multiple predictors we also have multiple models where we only consider a subset of the predictors. Let's look at the two models shown here,

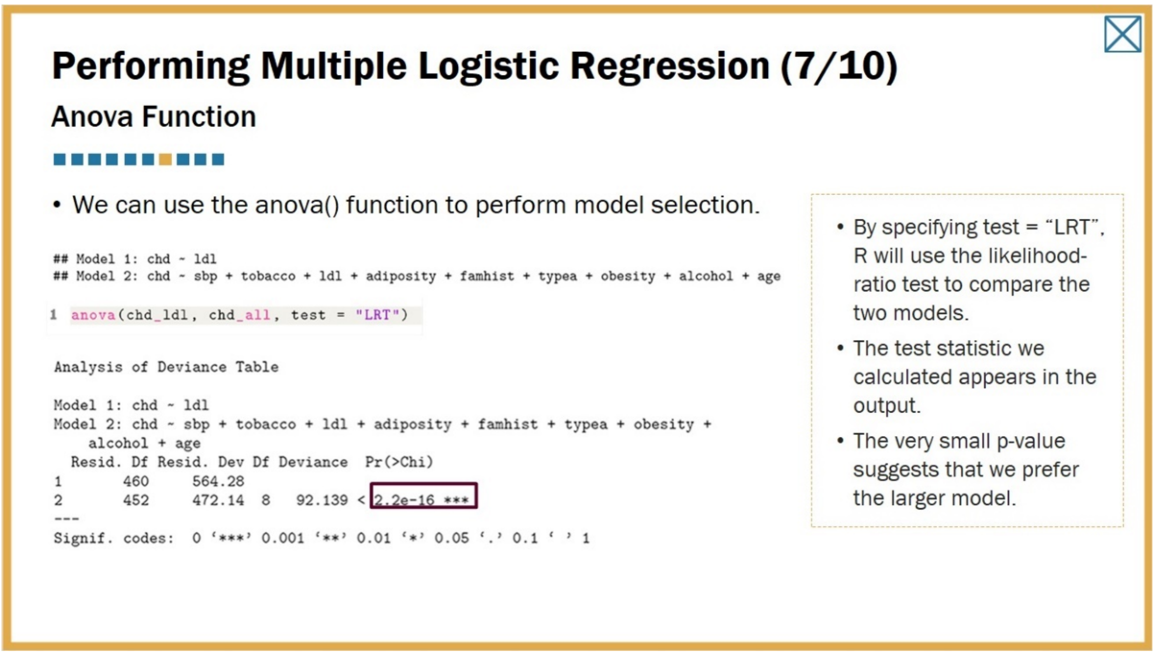
Model 1: chd ~ ldl

Model 2: chd ~ sbp + tobacco + ldl + adiposity + famhist + typea + obesity + alcohol + age

So, we would like to perform model selection. How can we do that?

We use an approach similar to the one we used for linear regression: Anova.

Tab 1.6: Anova Function



If we want to perform model selection, We can utilize the anova() function.

```
anova(chd_ldl , chd_all , test = "LRT")
```

By specifying test = "LRT", R will use the likelihood-ratio test to compare the two models shown here.

Analysis of Deviance Table

Model 1: chd ~ ldl

Model 2: chd ~ sbp + tobacco + ldl + adiposity + famhist + typea + obesity +

alcohol + age

Resid. Df Resid. Dev Df Deviance Pr(>Chi)

1 460 564.28

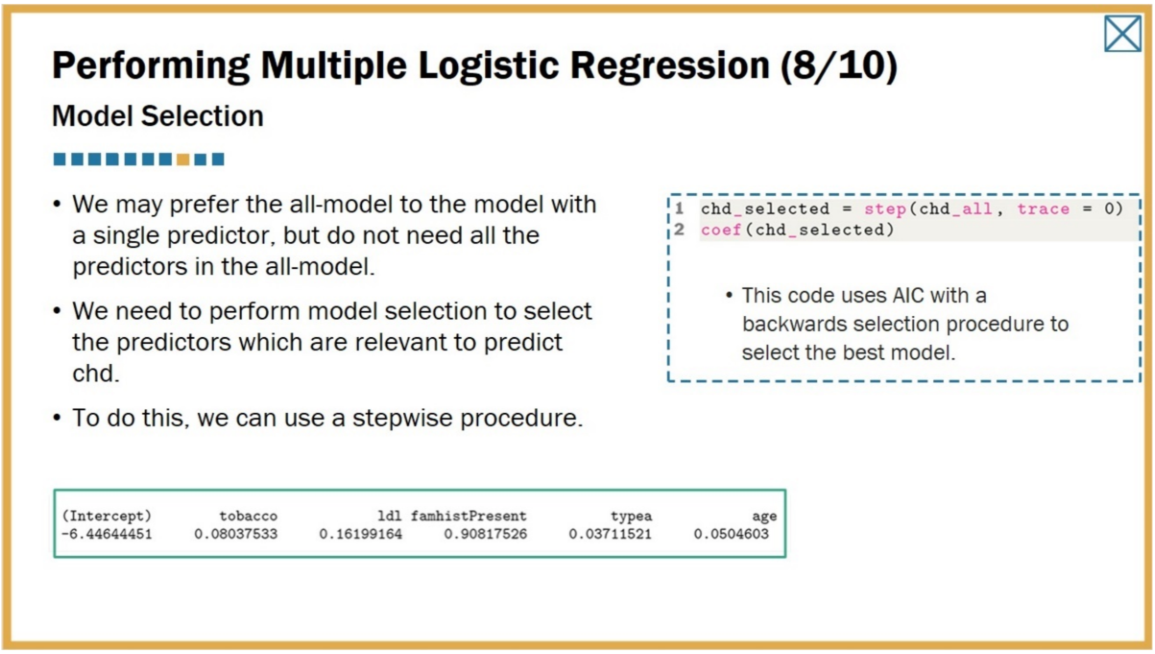
2 452 472.14 8 92.139 < 2.2e-16 ***

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Note that we use the deviance as a performance measure and the hypothesis test is based on this statistic.

We see that the test statistic that we had just calculated appears in the output. The very small p-value (2.2e-16) suggests that we prefer the larger model.

Tab 1.7: Model Selection



In the previous slide, we concluded we prefer the all-model compared to the model with only a single predictor, but do we actually need all of the predictors in the all-model?

We would like to perform model selection to choose the predictors which are relevant to predict chd. How can we do that?

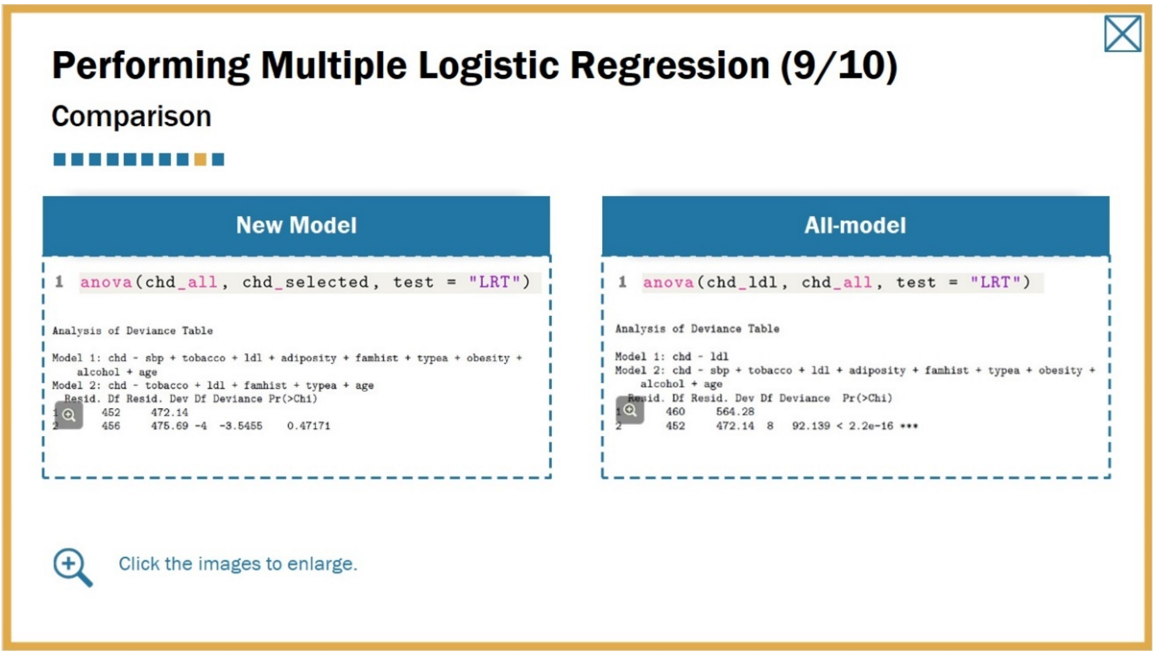
```
chd_selected = step(chd_all , trace = 0)
```

```
coef(chd_selected)
```

To select a subset of predictors, we can use a stepwise procedure as we did for linear regression, which we see on this slide.

The code shown here uses Akaike Information Criterion (AIC) with a backwards selection procedure to select the best model.

Tab 1.8: **Comparison**



We can compare this new model to the all-model.

```
anova(chd_all , chd_selected , test = "LRT")
```

Analysis of Deviance Table

Model 1: chd ~ sbp + tobacco + ldl + adiposity + famhist + typea + obesity +

alcohol + age

Model 2: chd ~ tobacco + ldl + famhist + typea + age

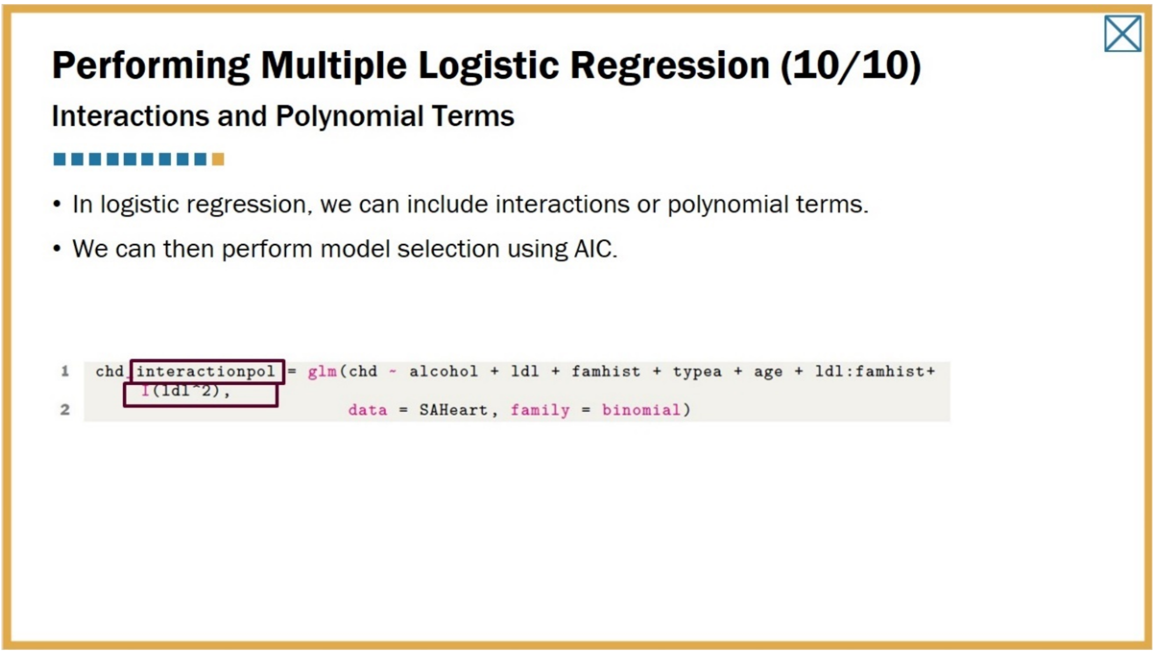
Resid. Df Resid. Dev Df Deviance Pr(>Chi)

1 452 472.14

2 456 475.69 -4 -3.5455 0.47171

We see that the selected model has a smaller deviance although the difference is not statistically significant.

Tab 1.9: Interactions and Polynomial Terms

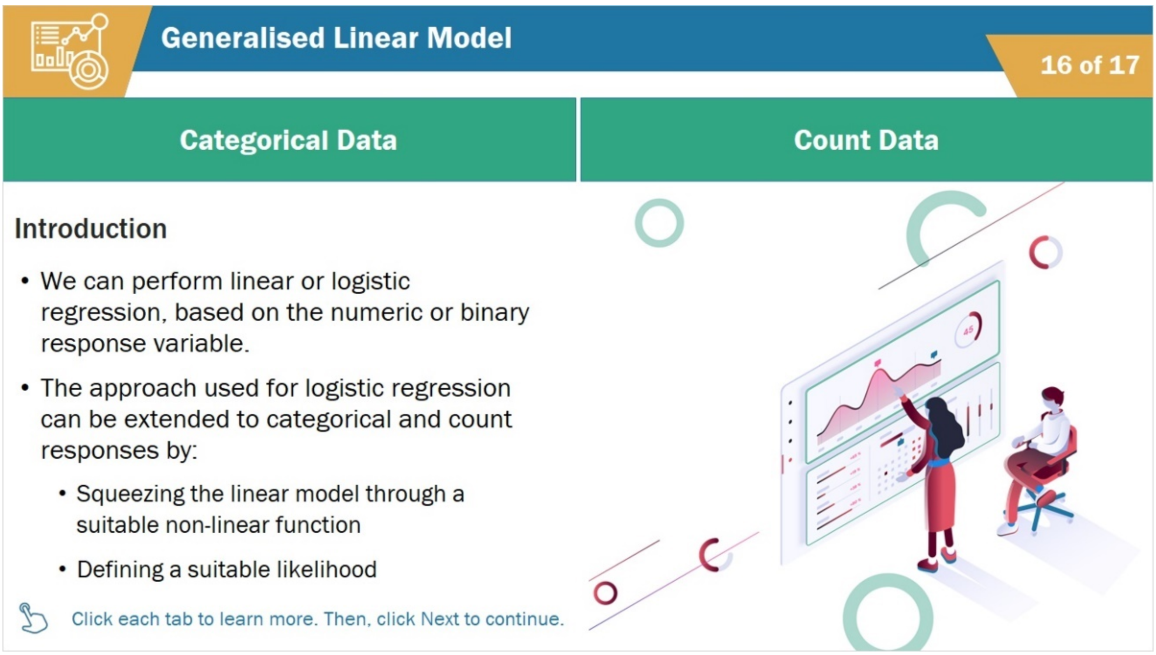


As in linear regression, we can also include interactions or polynomial terms and then we can perform model selection using AIC.

Note the interaction term ldl:famhist and the quadratic term I(ldl^2) in the regression model.

```
chd_interactionpol = glm(chd ~ alcohol + ldl + famhist + typea + age + ldl:famhist+
I(ldl^2),
data = SAHeart , family = binomial)
```

Slide 16: Generalised Linear Model



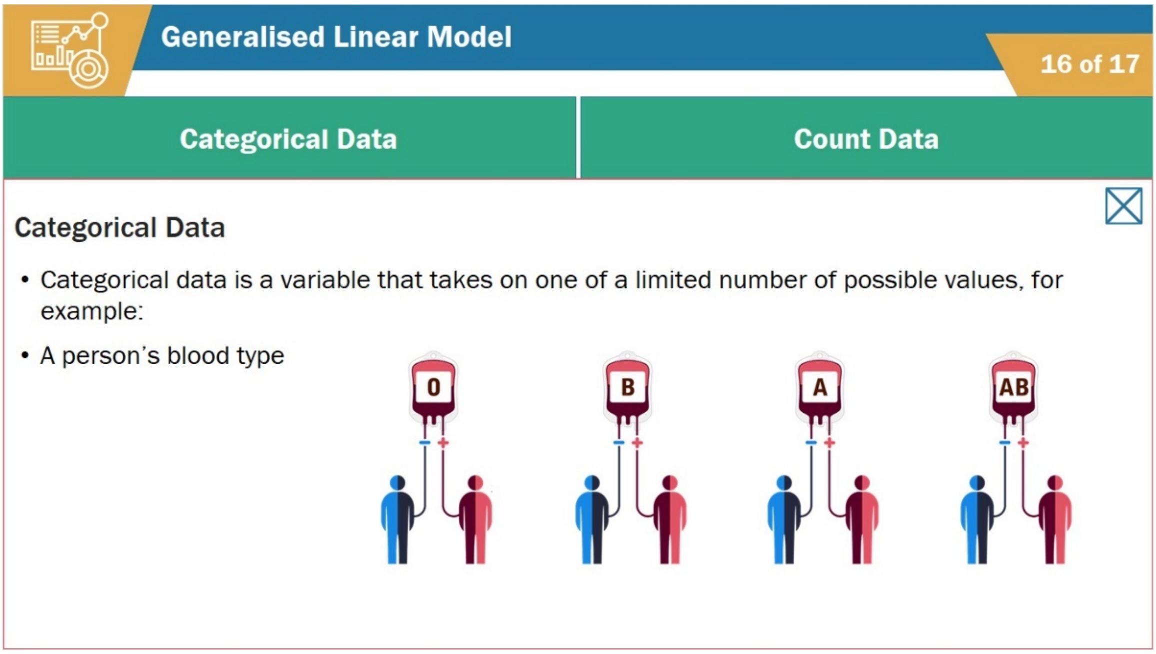
Depending on the type of response variable, numeric or binary, we can perform linear or logistic regression.

There are other types of responses, categorial and count.

We can extend the approach we used for logistic regression to these types of data by (i) squeezing the linear model through a suitable nonlinear function (also called link function); (ii) defining a suitable likelihood.

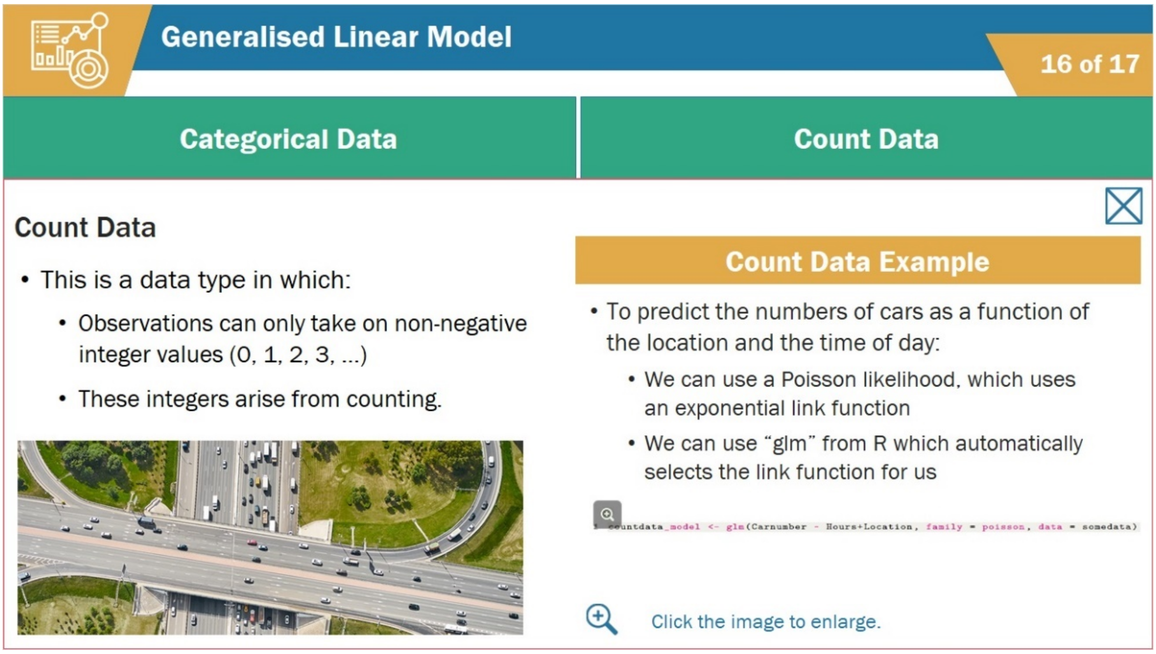
Click each tab to learn more. When you are ready, click next to continue.

Tab 1: Categorical Data



Categorical data is a variable that can take on one of a limited number of possible values (for example the blood type of a person: A, B, AB or O.)

Tab 2: **Count Data**



Count data refers to a data type in which the observations can take only the non-negative integer values (0, 1, 2, 3, ...), and where these integers arise from counting.

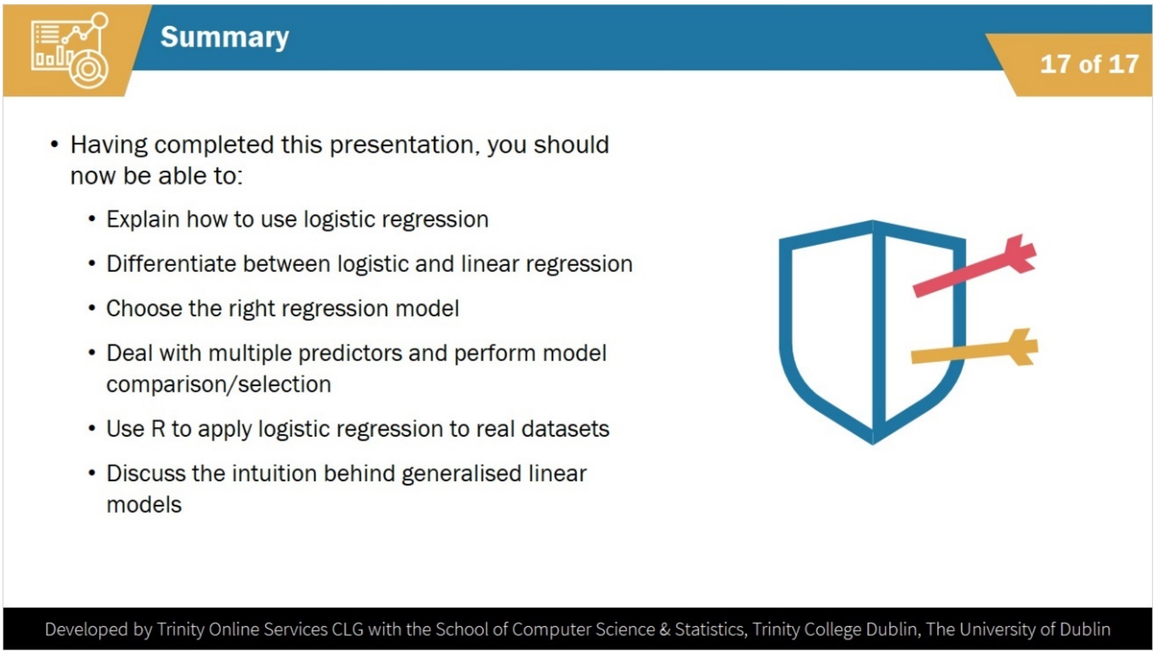
Let’s Assume we want to predict the number of cars as a function of the location (GPS coordinates) and time of the day. This is useful for predicting traffic and vehicle routing planning.

In this case we can use the Poisson likelihood.

```
countdata_model <- glm(Carnumber ~ Hours+Location , family = poisson , data = somedata)
```

This uses an exponential link function, but we can just use glm from R, which, based on the likelihood, automatically selects the link function for us.

Slide 17: **Summary**



The approach we have seen in these slides is valid for the whole family of generalized linear models, for which linear and logistic regression are particular cases.

We only need to change the likelihood to reflect the different type of response variable (numerical, binary, count data …)

Having completed this presentation, you should now be able to:

Explain how to use logistic regression,

Differentiate between logistic and linear regression,

Choose the right regression model,

Deal with multiple predictors and perform model comparison/selection,

Use R to apply logistic regression to real datasets, and

Discuss the intuition behind linear models.