Regression

Lecture 04

Topics today…..

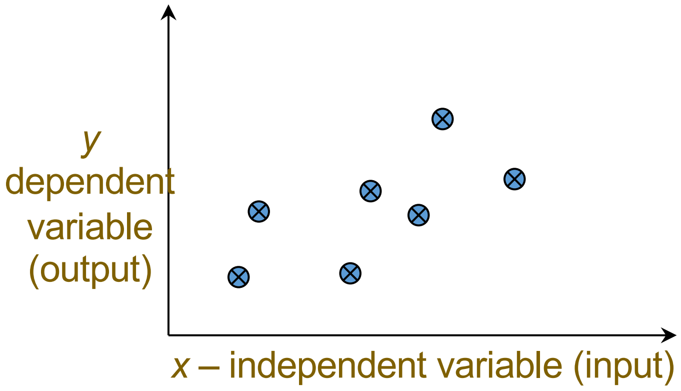
An Overview

- Introduction to Regression
- Types of Regression
- Key Concepts
- Applications
- Conclusion

Regression

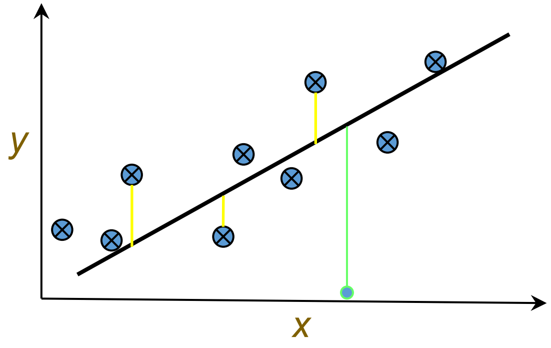
Definition

Regression analysis is a statistical method for modelling relationships between a dependent variable and one or more independent variables.



Purpose

To predict and forecast outcomes, and to understand the strength and type of relationships.



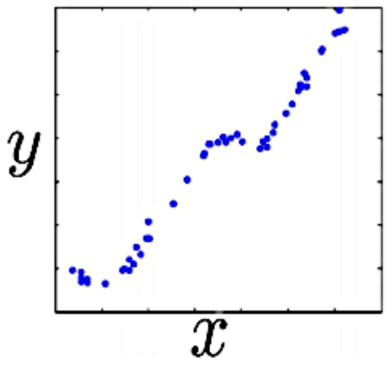
For classification the output(s) is nominal
In regression the output is continuous

Function Approximation
Many models could be used – Simplest is linear regression

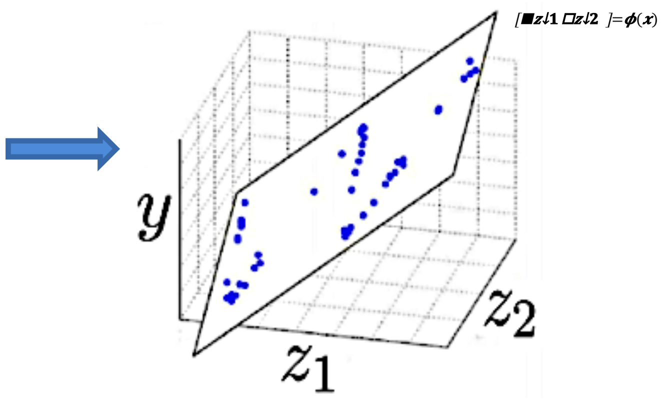
Fit data with the best hyper-plane which "goes through" the points
For each point the difference between the predicted point and the actual observation is the *residue*

Linear Regression

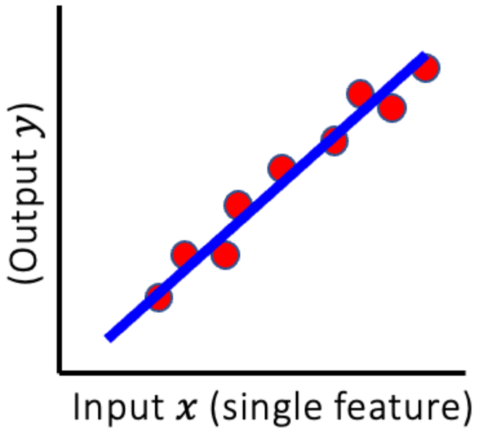
Linear regression is like fitting a line or (hyper)plane to a set of points



Original (single) feature
Nonlinear curve needed



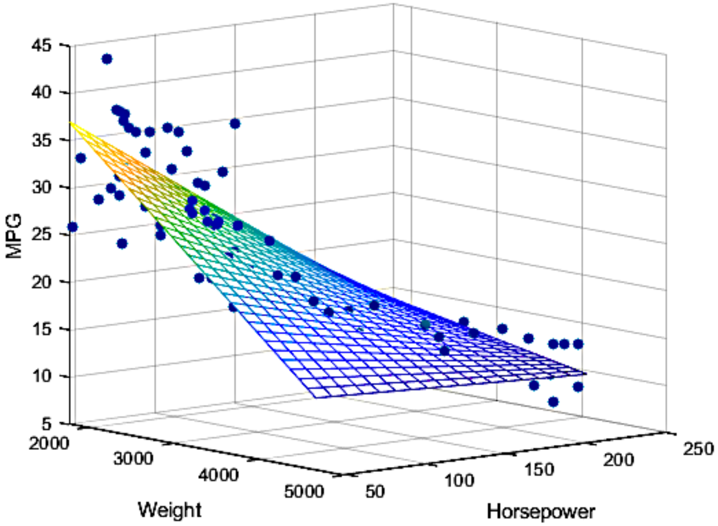
Two features
Can fit a plane (linear)



For now, assume just one (input) independent variable *x*, and one (output) dependent variable *y*

Multiple linear regression assumes an input vector **x**

Multivariate linear regression assumes an output vector **y**



Types of Regression

- Linear Regression
- Multiple Regression
- Logistic Regression
- Polynomial Regression
- Ridge Regression
- Lasso Regression
- Elastic Net Regression
- Quantile Regression
- Non-Linear Regression

Simple Linear Regression

We "fit" the points with a line (i.e. hyperplane)

Which line should we use?

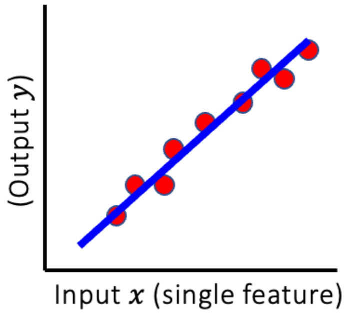
Choose an objective function

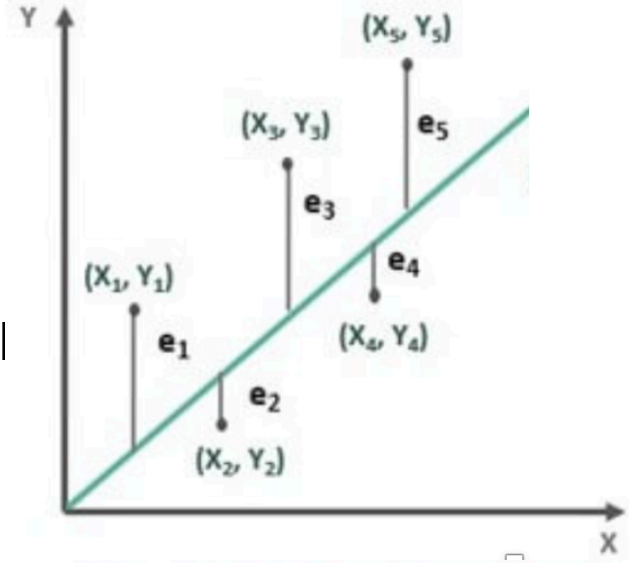
For simple linear regression we use sum squared residue (SSR)

$$\text{SS (predicted}_i - \text{actual}_i)^2 = \text{SS (residue}_i)^2$$

Thus, find the line which minimizes the sum of the squared residues (e.g. least squares)

This exactly mimics the case assuming data points were sampled from an actual target hyperplane witl Gaussian noise added





$$\text{RSS} = e1^2 + e2^2 + e3^2 + e4^2 + e5^2$$

e = Error or residue

Numerical

You are given the following dataset representing the relationship between the number of hours studied and the scores achieved by students in a test.

Hours Studied (X)	Test Score (Y)
2	50
3	60
5	80
7	90
8	95

A linear regression model is proposed as: Y=5X+40

1.Calculate the predicted test scores for each value of hours studied (X) using the given linear regression model.

2.Compute the Sum of Squared Errors (SSE) between the actual test scores and the predicted test scores

Learning parameters

For the 2-dproblem (line) there are coefficients for the bias and the independent variable (y-intercept and slope)

$$y = \beta_0 + \beta_1 x$$

To find the values for the coefficients (weights) which minimize the objective function we can take the partial derivatives of the objective function (SSE) with respect to the coefficients. Set these to 0 and solve

$$\beta_0 = \frac{\sum y - \beta_1 \sum x}{n}$$

$$\beta_1 = \frac{n\sum xy - \sum x \sum y}{n\sum x^2 - \left(\sum x\right)^2}$$

Numerical

Problem Statement:
Suppose we have a dataset that shows the number of hours studied by a student and their corresponding scores on a test. The goal is to predict the regression function for test score (y) based on the number of hours studied (x).

Hours Studied (x)	Test Score (y)
1	2
2	4
3	5
4	4
5	5

Multiple Linear Regression

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \ldots\ldots + + \beta_n x_n$$

There is a closed form for finding multiple linear regression weights which requires matrix inversion, etc.

There are also iterative techniques to find weights

$$\Delta w = c\left(t - net\right) \times x_i$$

One is the delta rule. For regression we use an output node which is not thresholded (just does a linear sum) and iteratively apply the delta rule – For regression net is the output

Delta rule will update until minimizing the SSE, thus solving multiple linear regression

There are other regression approaches that give different results by trying to better handle outliers and other statistical anomalies

$\triangle \boldsymbol{w_i}$: change in weight $\boldsymbol{w_i}$

c : is the learning rate

x_i : is the input for that weight

t: The target output (the actual label or value we want to predict).

net: The net input or the predicted output (this could be the output from a neural network or another model).

Linear Regression - Problem

$$\Delta w = c\left(t - net\right) \times x_i$$

Assume we start with all weights as 1 (don't use bias weight though you usually always will – else forces the line through the origin)

Remember for regression we use an output node which is not thresholded (just does a linear sum) and iteratively apply the delta rule – thus the net is the output

What are the new weights after one iteration through the following training set using the delta rule with a learning rate c = 1. How does it generalize for the novel input (-.3, 0)?

x_1	x_2	Target y
.5	-.2	1
1	1	0

$$\Delta w = c\left(t - net\right) \times x_i$$

Initial Setup
Initial Weights: w1=1, w2=1
Learning Rate: c=1

Input (x1,x2)	Target (t)	Predicted Output(net) = $w_1x_1+w_2x_2$	Error (t-net)	Δw_1= c(t-net).x_1	Δw_2= c(t-net).x_2	Updated w_1	Updated w_2
Initial Weights						1.00	1.00
(0.5, -0.2)	1	0.5*1+(−0.2)*1 =0.3	1−0.3=0.7	0.7·0.5=0.35	0.7·−0.2=−0.14	1+0.35=1.35	1−0.14=0.86
(1, 1)	0	1*1.35+1*0.86=2.21	0−2.21=−2.21	−2.21*1=−2.21	−2.21*1=−2.21	1.35−2.21=−0.86	0.86−2.21=−1.35

Final Weights

After processing all the inputs in the training set, the final weights are:

$w1=-0.86$

$w2=-1.35$

Generalization to Novel Input (–0.3,0)

$Y = w1\cdot(-0.3)+w2\cdot 0 = -0.86\cdot(-0.3)+(-1.35)\cdot 0 = 0.258$

Practice Numerical

$$\Delta w = c\left(t - net\right) \times x_i$$

Assume we start with all weights as 0

What are the new weights after one iteration through the following training set using the delta rule with a learning rate c = .2

How does it generalize for the novel input (1, .5)?

x_1	x_2	Target
.3	.8	.7
-.3	1.6	-.1
.9	0	1.3

Linear Regression- Summary

One advantage of linear regression models (and linear classification) is the potential to look at the weights to give insight into which input variables are most important in predicting the output

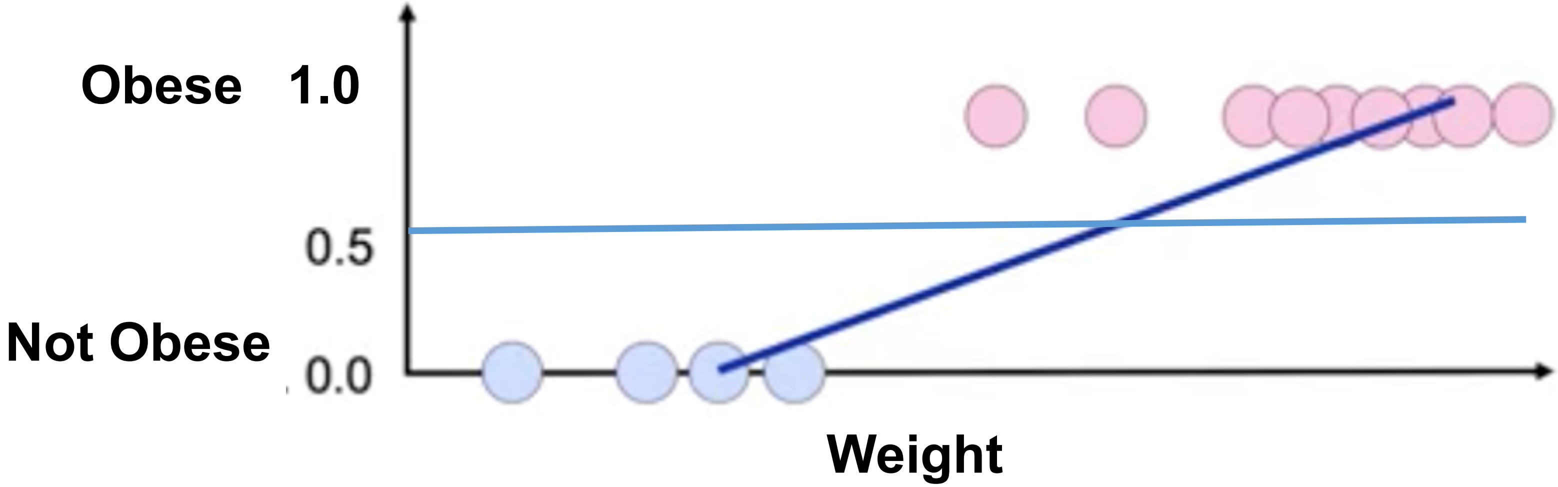
The variables with the largest weight magnitudes have the highest correlation with the output

1. A large positive weight implies that the output will increase when this input is increased (positively correlated)
2. A large negative weight implies that the output will decrease when this input is increased (negatively correlated)
3. A small or 0 weight suggests that the input is uncorrelated with the output (at least at the 1st order)

Linear regression/classification can be used to find best "indicators"

1. Be careful not to confuse correlation with causality
2. Linear models cannot detect higher order correlations! The power of more complex machine learning models!!

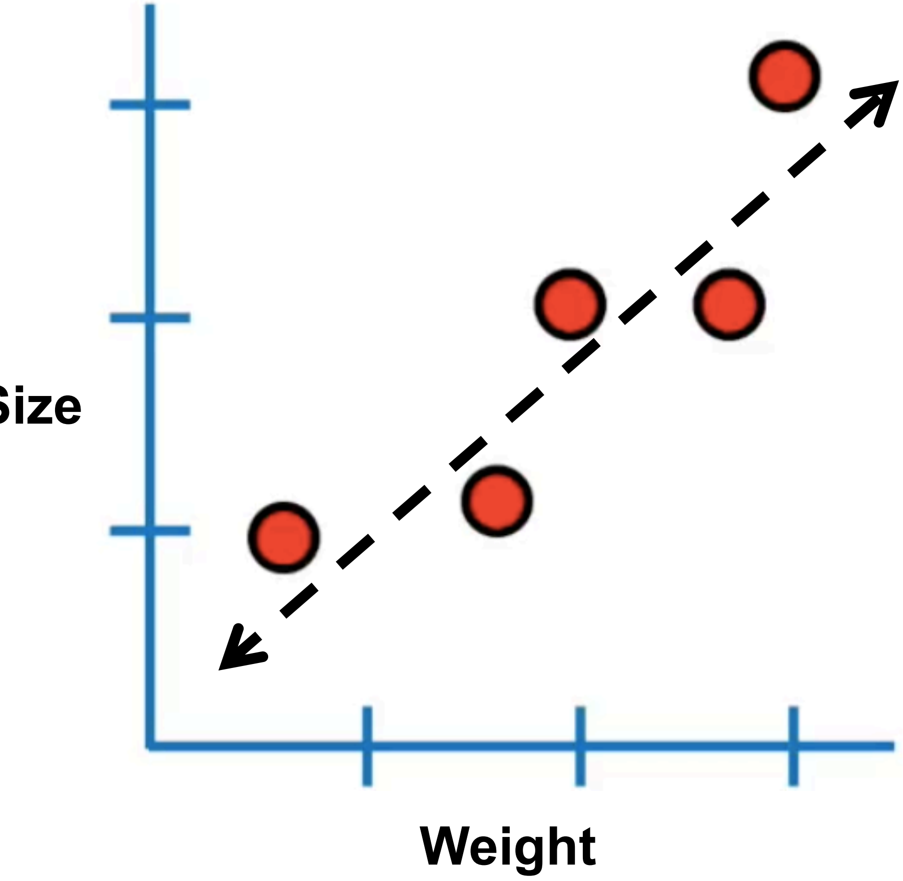
Linear regression for Classification



- If the model result > 0.5; predict Obese
- If the model result < 0.5; predict Not Obese

Linear Regression

- Using data to predict something falls under the category of "machine learning"

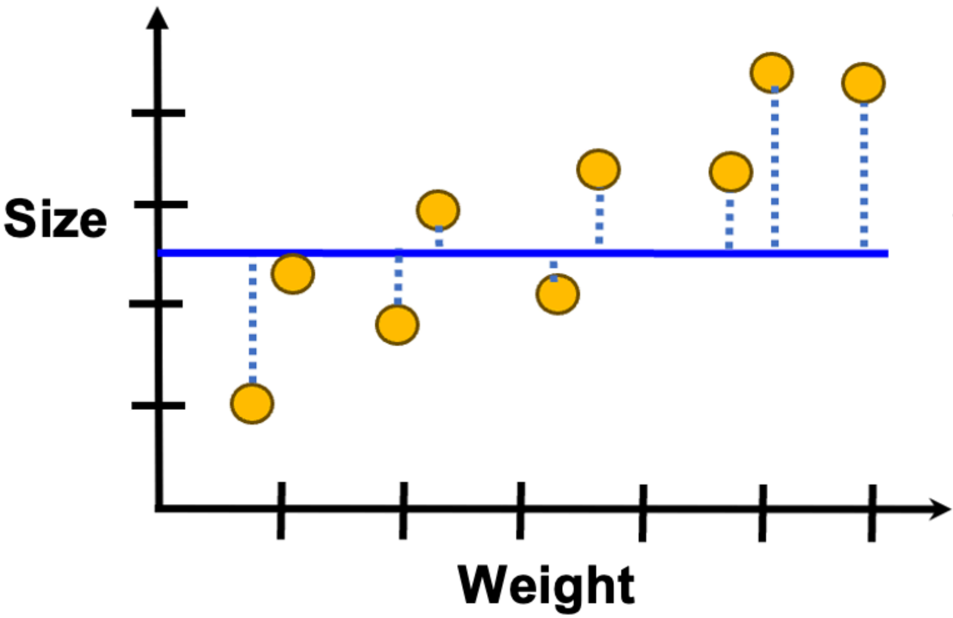


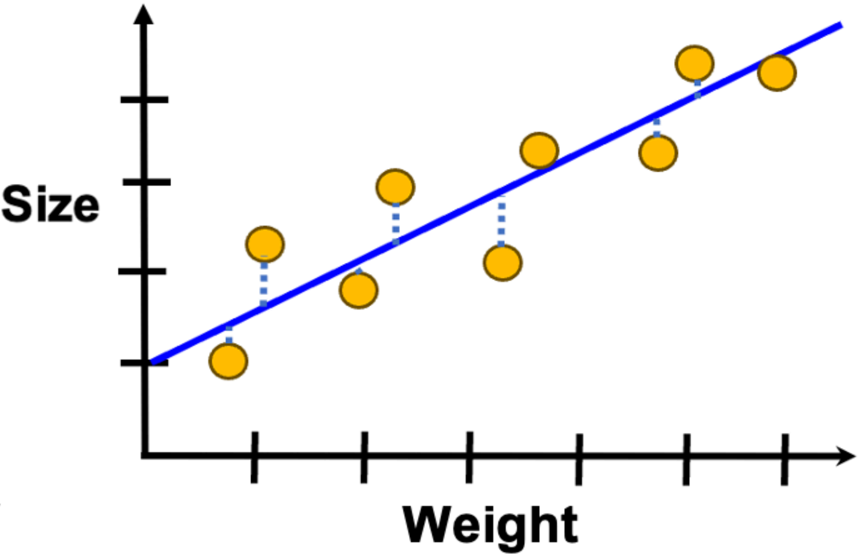
- Calculate R^2 and determine if weight and size are correlated. Large values imply a large effect
- Calculate a p-value to determine if the R^2 value is statistically significant.
- Use the line to predict size for give weight

R^2 compares a measure of a good fit, SS(fit) to a measure of a bad fit, SS(mean)

$$R^2 = \frac{SS(mean) - SS(fit)}{SS(mean)}$$

- Size = 0.7 x weight + 0.86

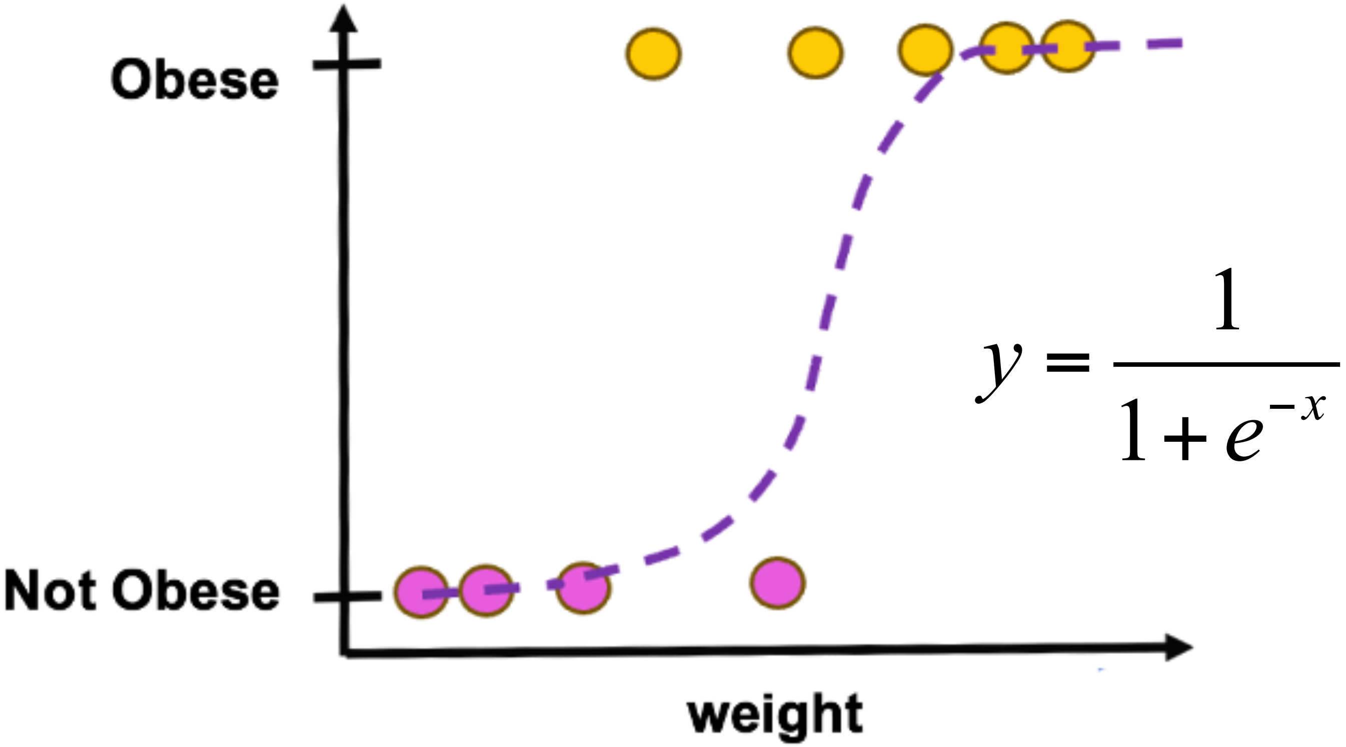




Weight

Logistic Regression

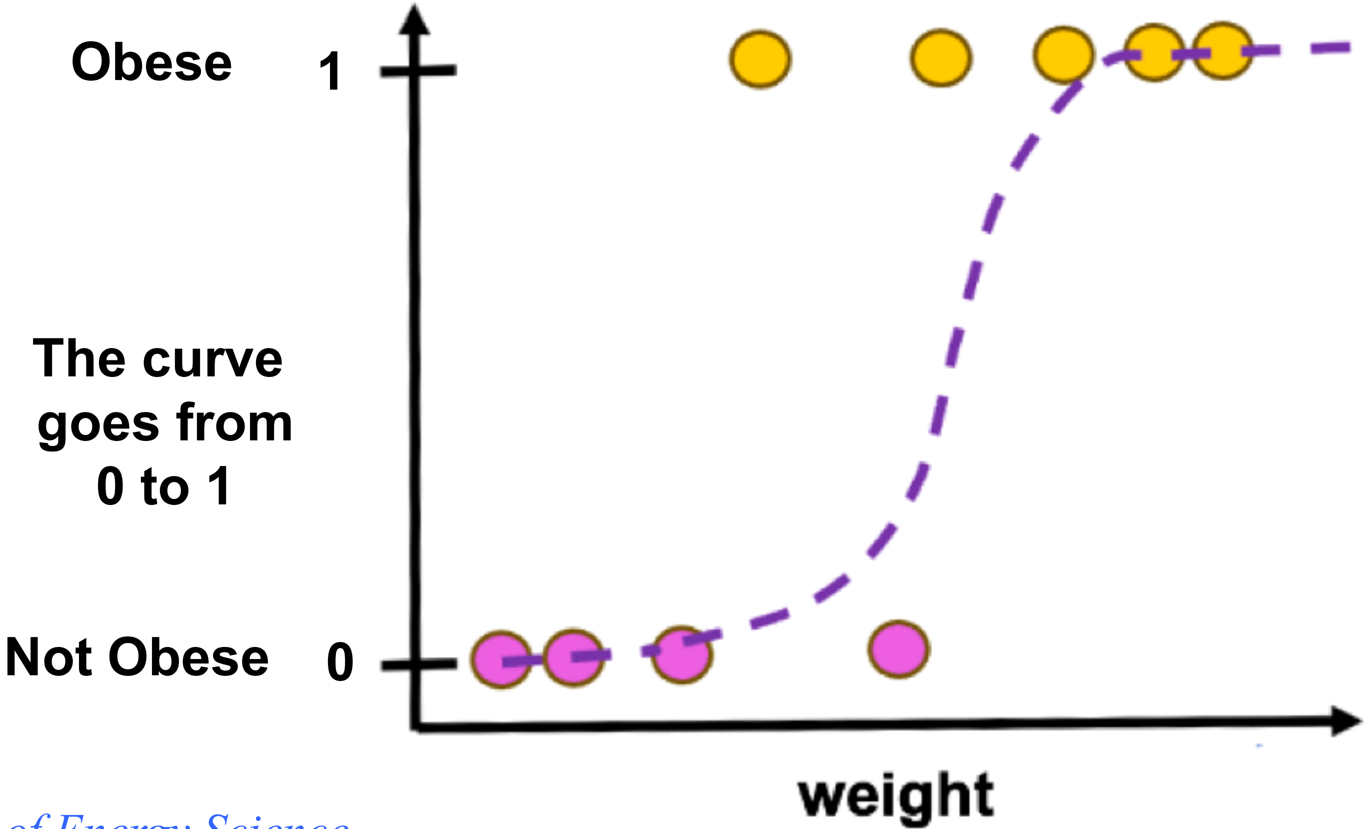
- Logistic regression is similar to linear regression, except
- Logistic regression predicts whether something is True or False, instead of predicting something continuous like size



$$y = \frac{1}{1+e^{-x}}$$

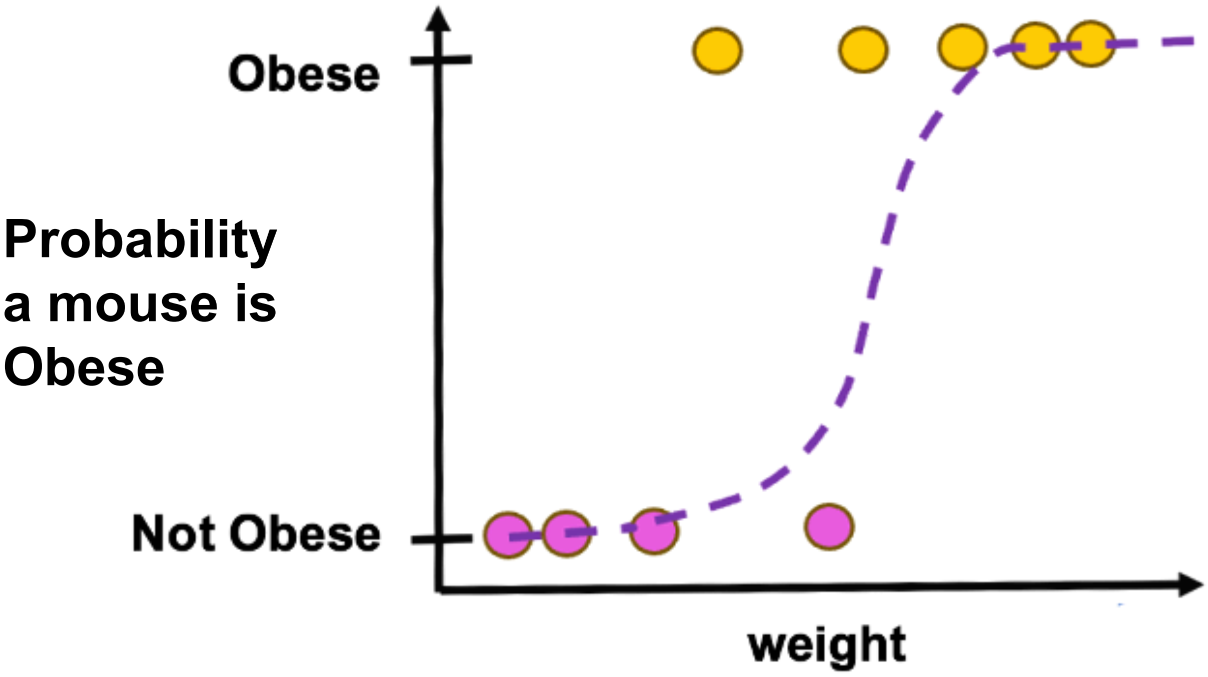
Logistic Regression

- Instead of fitting a line to the data, logistic regression fits an “S” Shaped “logistic function
- The curve tells you the probability that a mouse is obese based on its weight



Logistic Regression

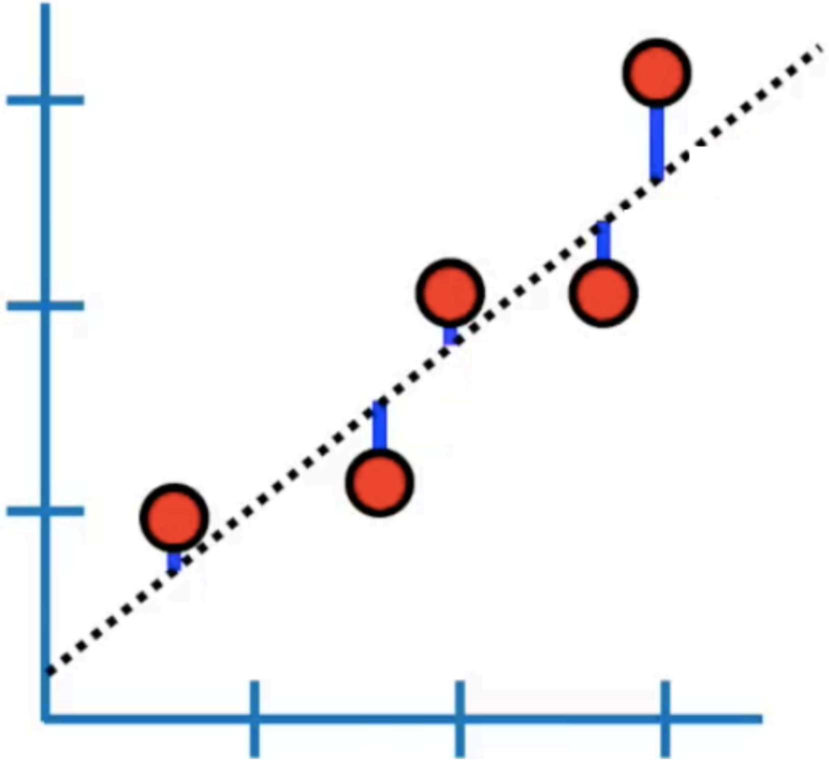
- We use a continuous variable (like weight) to predict obesity
- Although logistic regression tells the probability that a mouse is obese or not, its usually used for classification
- For example, if the probability a mouse is obese is >50%, then we'll classify it as obese, otherwise we'll classify it as "not obese"



Logistic Regression

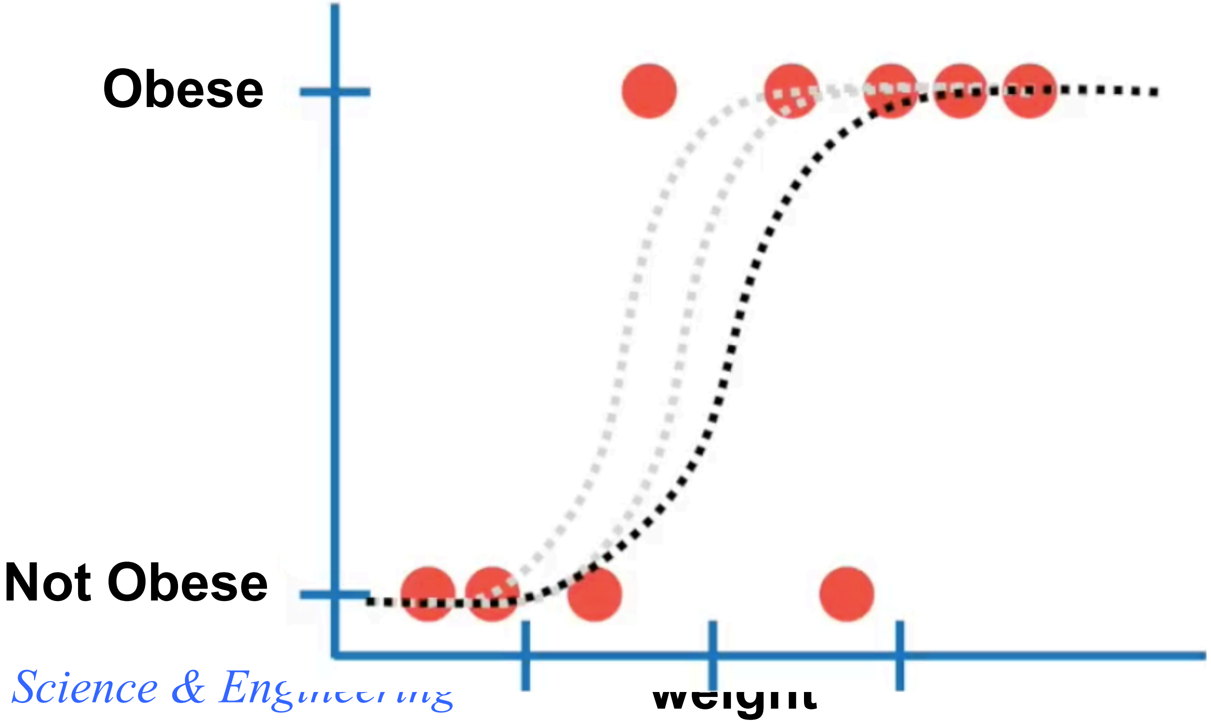
- Logistic regression's ability to provide probabilties and classify new samples using continuous and discrete measurements makes it a popular machine learning method
- One big difference between linear regression and logistic regression is how the line is fit to the data
- With linear regression, we fit the line using "least squares". In other words, we find the line that minimizes the sum of the squares of these residuals (SSR)
- We also use the residuals to calculate R^2 and to compare simple models to complicated models
- Logistic regression doesn't have the same concept of a "residual", so it can't use least squares and it can't calculate R^2.
- Instead it uses something called "maximum likelihood".

We find the line that minimizes the sum of the squares of these residuals



Logistic Regression

- pick a probability, scaled by weight, of observing an obese mouse and use that to calculate the likelihood of observing a non-obese mouse that weighs this much
- Then we calculate the likelihood of observing all remaining mouse
- Lastly we multiply all of those likelihoods together. That’s the likelihood of the data given as “S” Shaped line
- Finally, the curve with the maximum likelihood is selected



Log(odds)

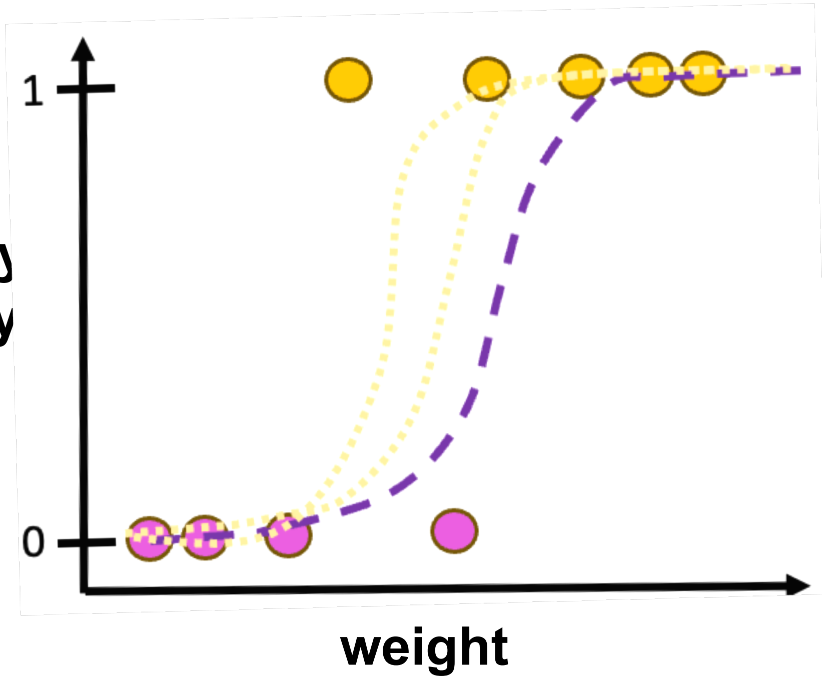
The Odds in favor of my team wining the game are 5 to 3:

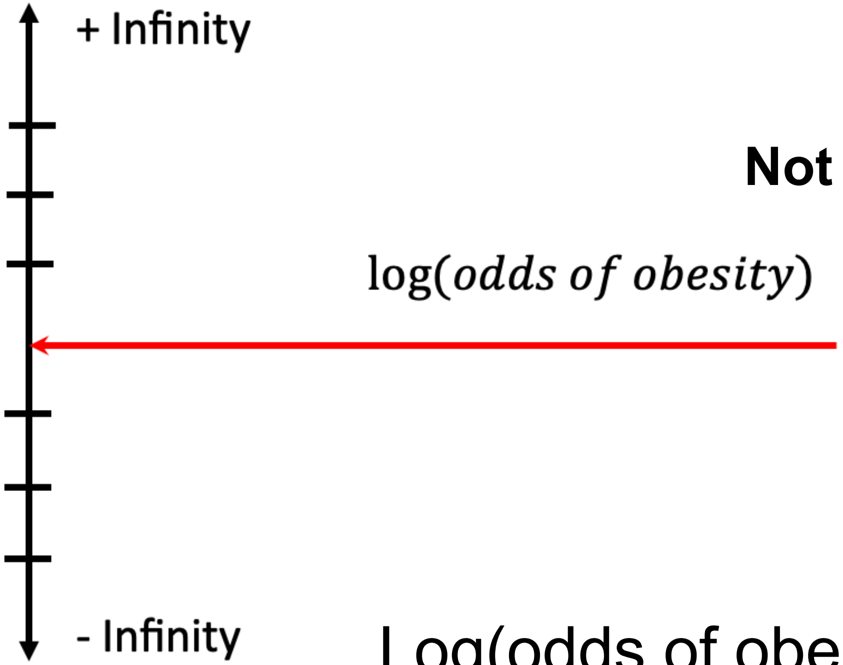
We can write this as log(odds) = log(5/3)

Probability of wining: p = 5/8

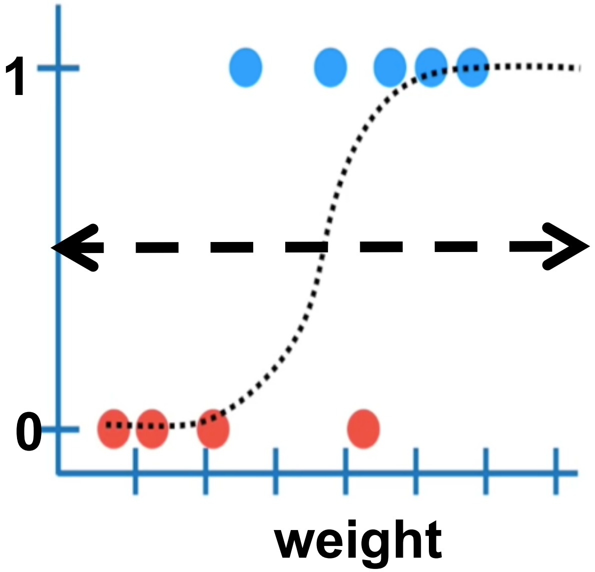
$$\log(odds) = \log\left(\frac{p}{1-p}\right) = \log\left(\frac{\frac{5}{8}}{1-\frac{5}{8}}\right) = \log\left(\frac{5}{3}\right)$$

The y-axis in logistic regression is transformed from the “probability of obesity to the log(odds of obesity



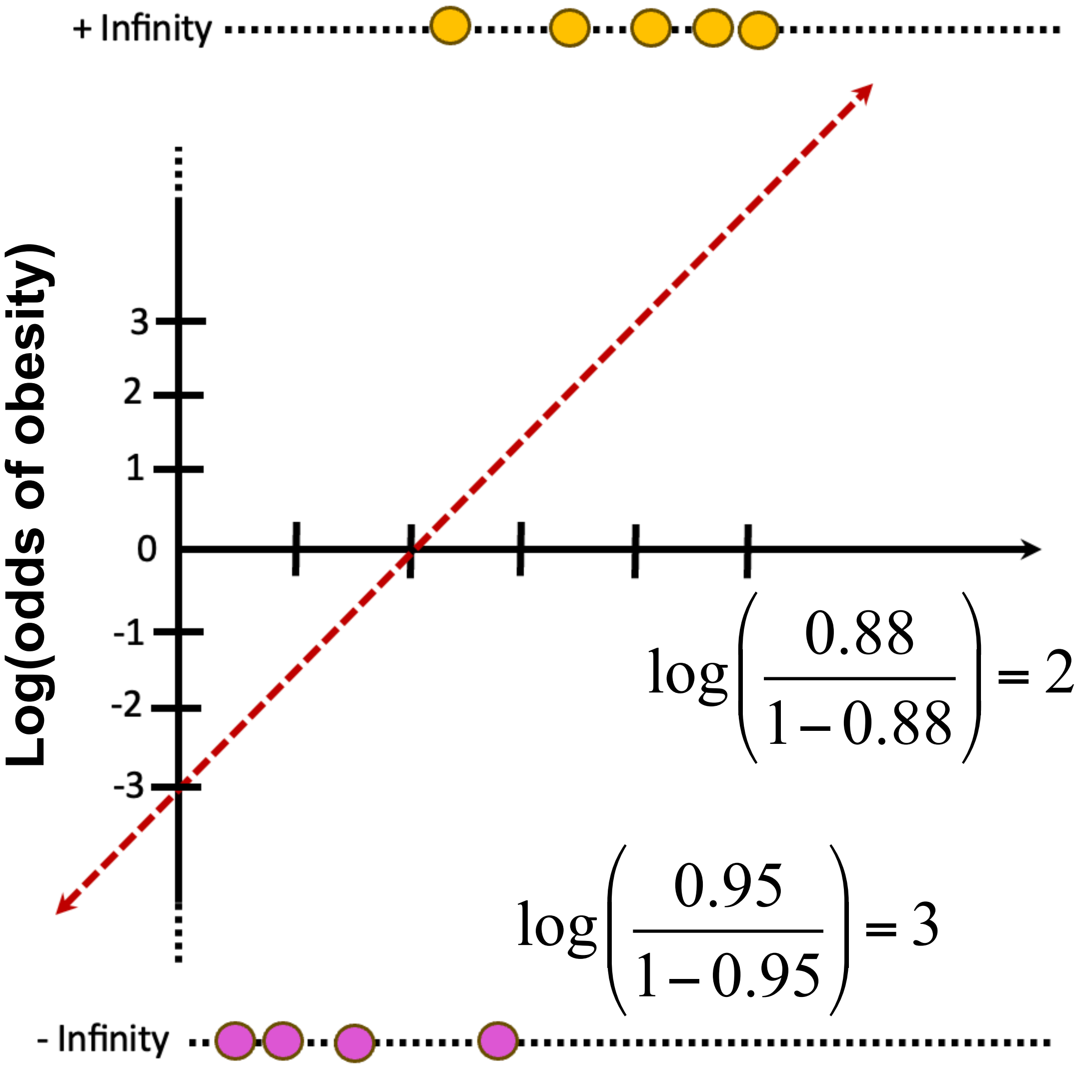


$$\text{Log(odds of obesity)} = \log(odds) = \log\left(\frac{p}{1-p}\right)$$



$$\log\left(\frac{0.5}{1-0.5}\right)=0$$

$$\log\left(\frac{0.731}{1-0.731}\right)=1$$



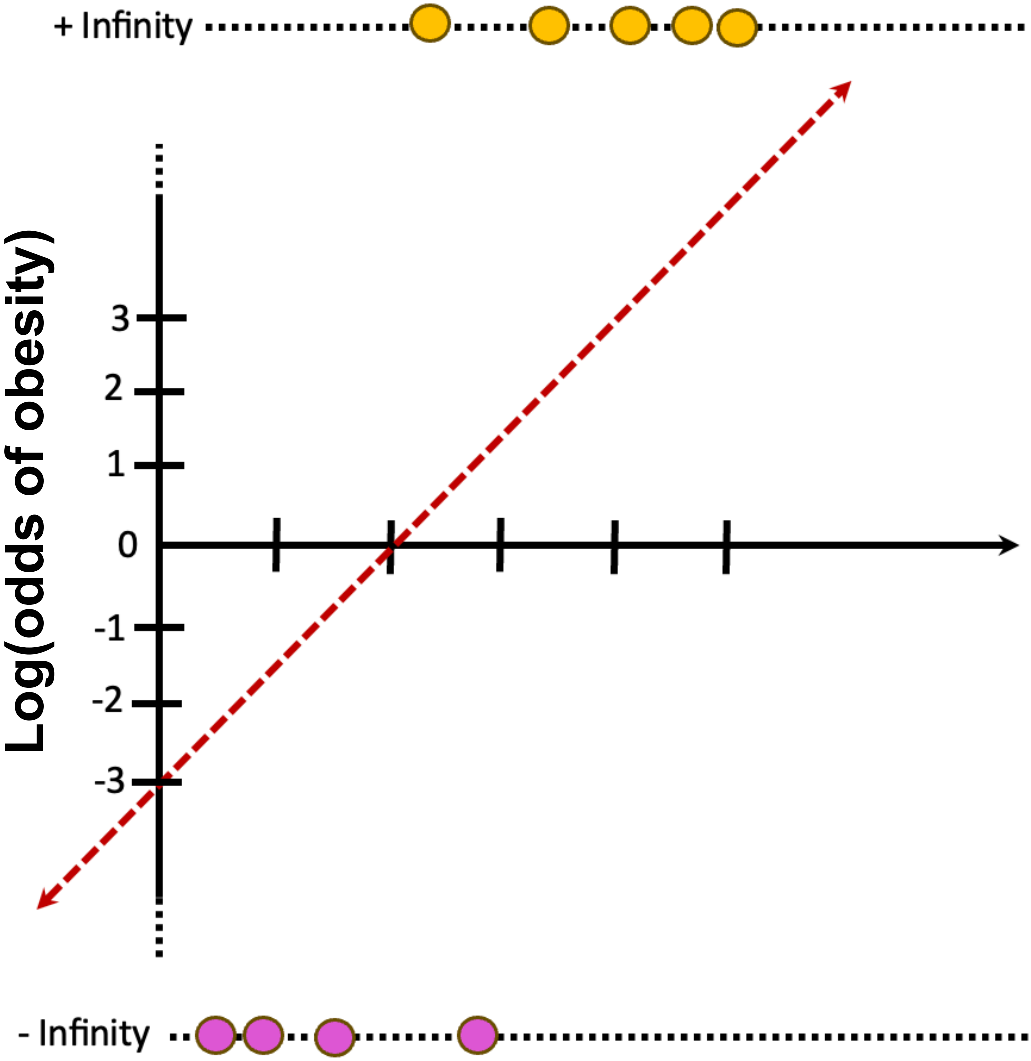
$$\log\left(\frac{0.88}{1-0.88}\right)=2$$

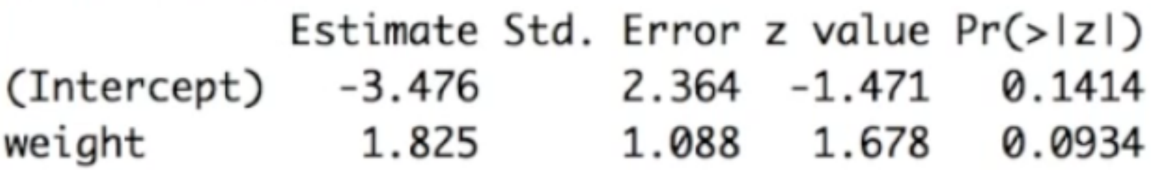
$$\log\left(\frac{0.95}{1-0.95}\right)=3$$

The coefficient for the line in logistic Regression

The new y-axis transform the squiggly line into a straight line

Y = -3.48 + 1.83 x weight

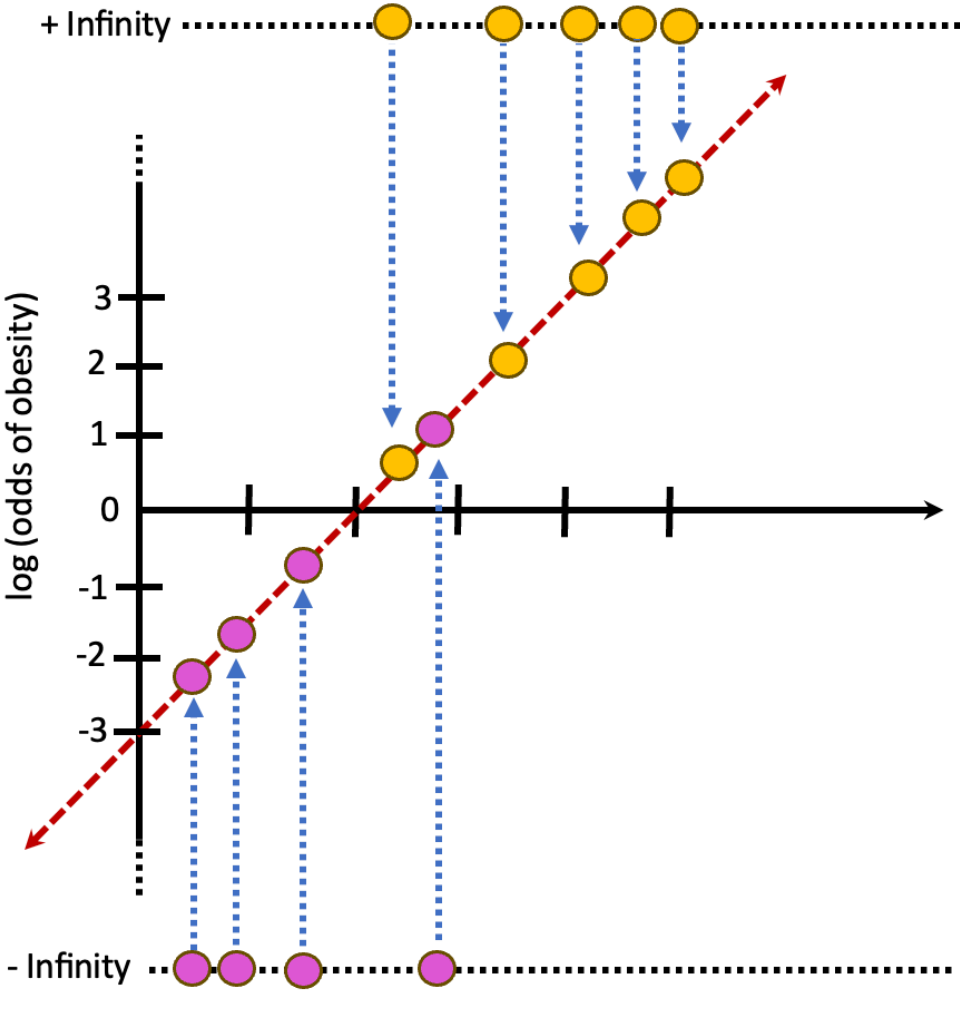


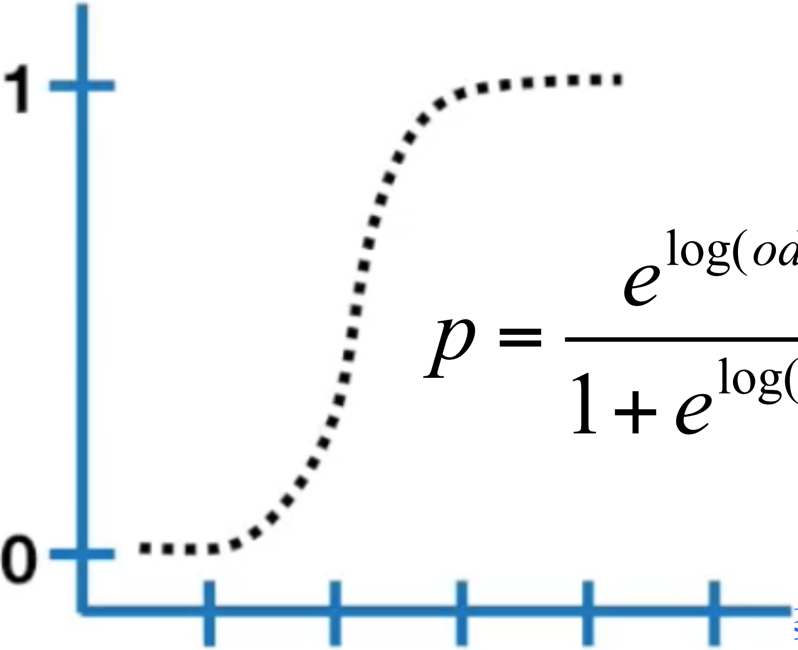


```
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   -3.476      2.364  -1.471   0.1414
weight         1.825      1.088   1.678   0.0934
```

Can't use least-squares to find the best fitting line, instead use the maximum likelihood

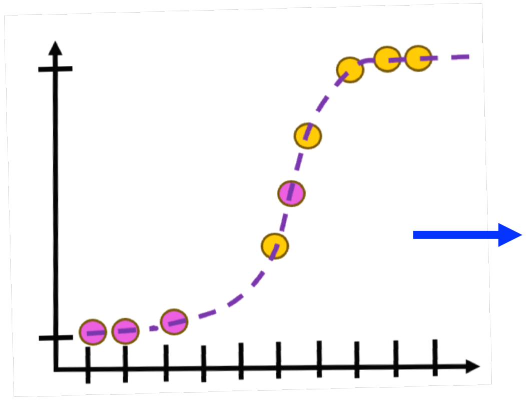
- First, project the original data points onto candidate line
- This gives each sample a candidate log(odds) value
- Then transform the candidate log(odds) to candidate probabilities using this formula

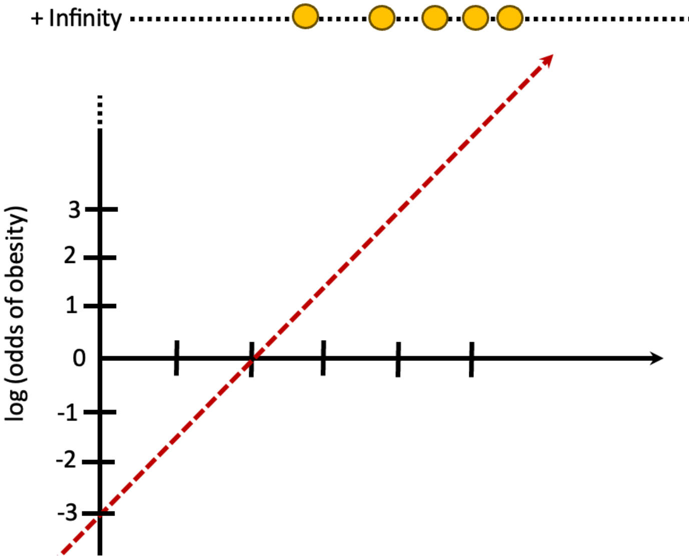




$$p = \frac{e^{\log(odds)}}{1 + e^{\log(odds)}}$$

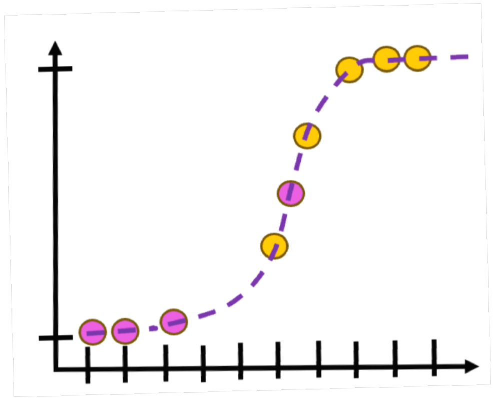
$$\log\left(\frac{p}{1-p}\right)=\log(odds)$$

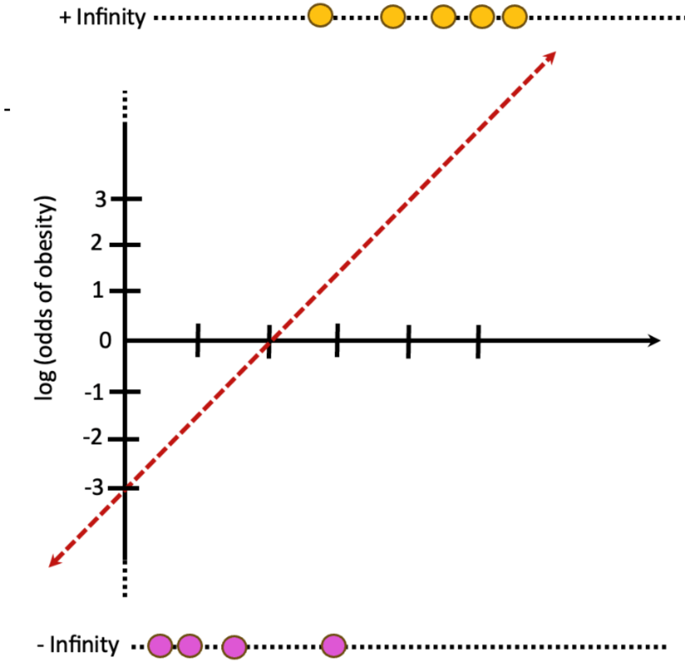




An equation that takes probability as input and outputs log(odds)

$$p=\frac{e^{\log(odds)}}{1+e^{\log(odds)}}$$

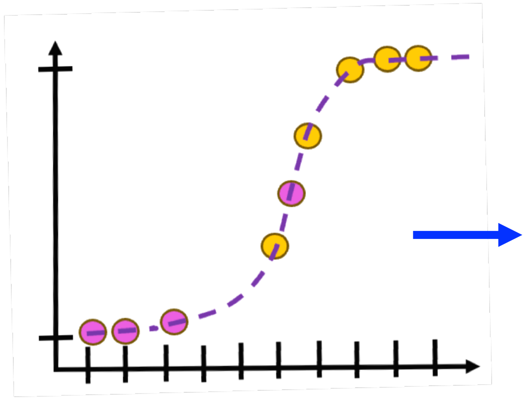


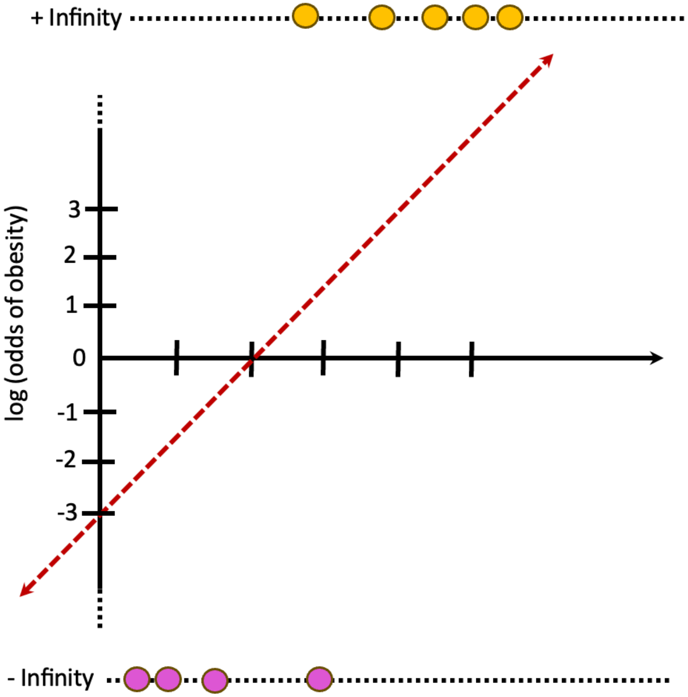


An equation that takes log(odds) as input and outputs probability

An equation that takes probability as input and outputs log(odds)

$$\log\left(\frac{p}{1-p}\right)=\log(odds)$$





$$\frac{p}{1-p}=e^{\log(odds)}$$

$$p=(1-p)e^{\log(odds)}$$

$$p=e^{\log(odds)}-pe^{\log(odds)}$$

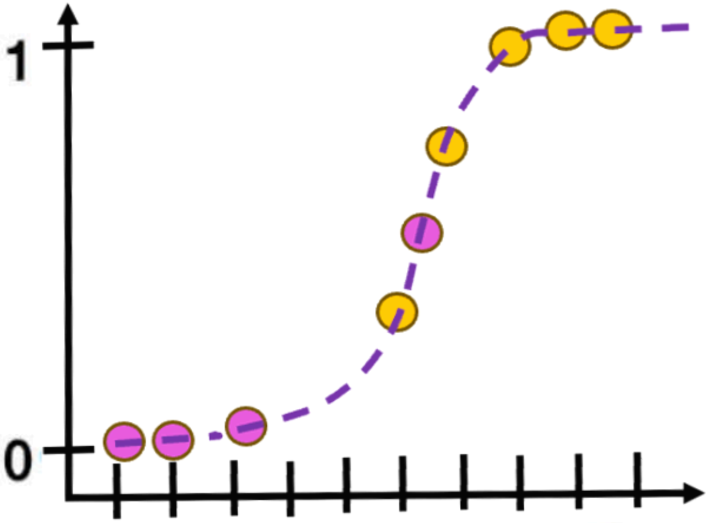
$$p+pe^{\log(odds)}=e^{\log(odds)}$$

$$p\left(1+e^{\log(odds)}\right)=e^{\log(odds)}$$

$$p=\frac{e^{\log(odds)}}{1+e^{\log(odds)}}$$

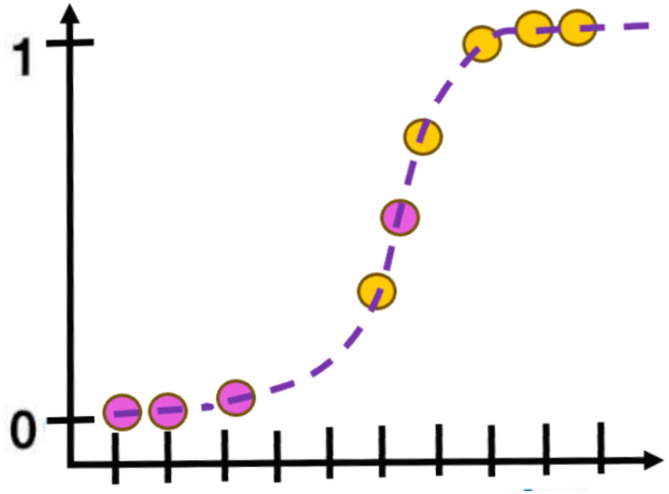
An equation that takes log(odds) as input and outputs probability

- Now we use the observed status (obese or not obese) to calculate their likelihood given the shape of the squiggly line
- Calculate the likelihood of the obese mice, given the shape of the squiggle
- The likelihood that this mouse is obese, given the shape of the squiggle, is the same as the predicted probability
- The likelihood that these mice are obese are 0.4,0.85,0.9, 0.98, 0.99



The likelihood for all of the obese mice is just the product of the individual likelihoods

= = 0.4 x 0.85 x 0.9 x 0.98 x 0.99



The probability that these mice are obese are both 0.01, so the probability and likelihood that they are not obese is (1 - 0.01)

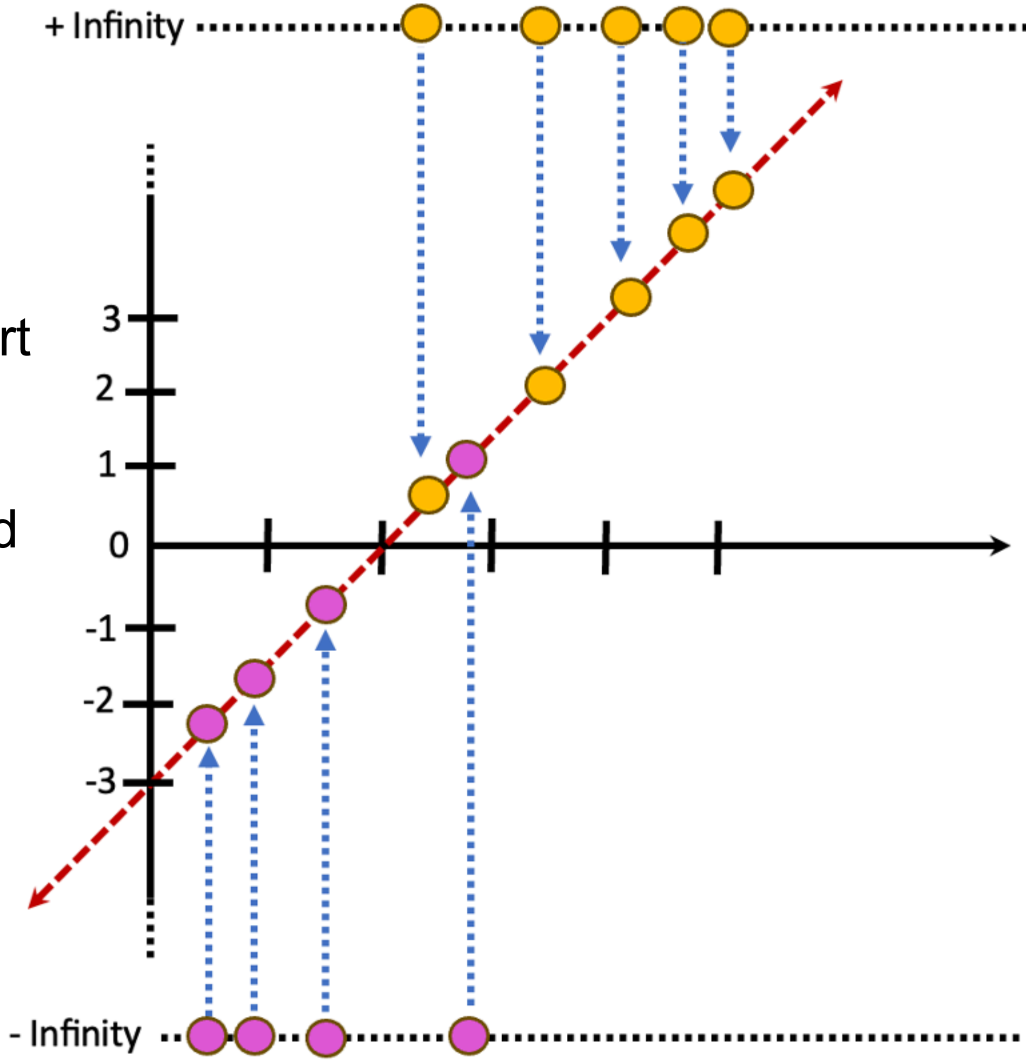
The likelihood for the mice that are not obese:
(1 - 0.6), (1 – 0.03), (1 – 0.01), (1 – 0.01)

likelihood of data given the squiggle
= 0.4 x 0.85 x 0.9 x 0.98 x 0.99 x(1 - 0.6) x (1 – 0.03) x (1 – 0.01) x (1 – 0.01)

Log(likelihood of data given the squiggle)
= log(0.4) + log(0.85) + log(0.9) + log(0.98) + log(0.99) + log(1 - 0.6) + log(1 – 0.03) + log(1 – 0.01) + log(1 – 0.01)

Log(likelihood of data given the squiggle) = -2.1813

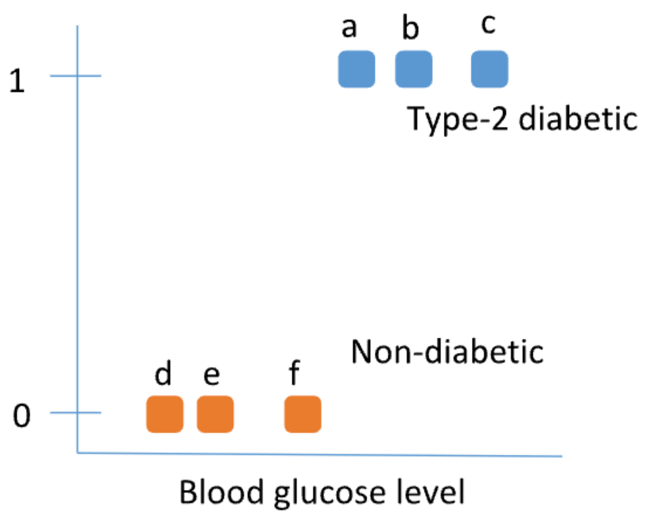
The algorithm that finds the line with the maximum likelihood is pretty smart – each time it rotates the line, it does so in a way that increases the log-likelihood. Thus the algorithm can find the optimal fit after a few rotations

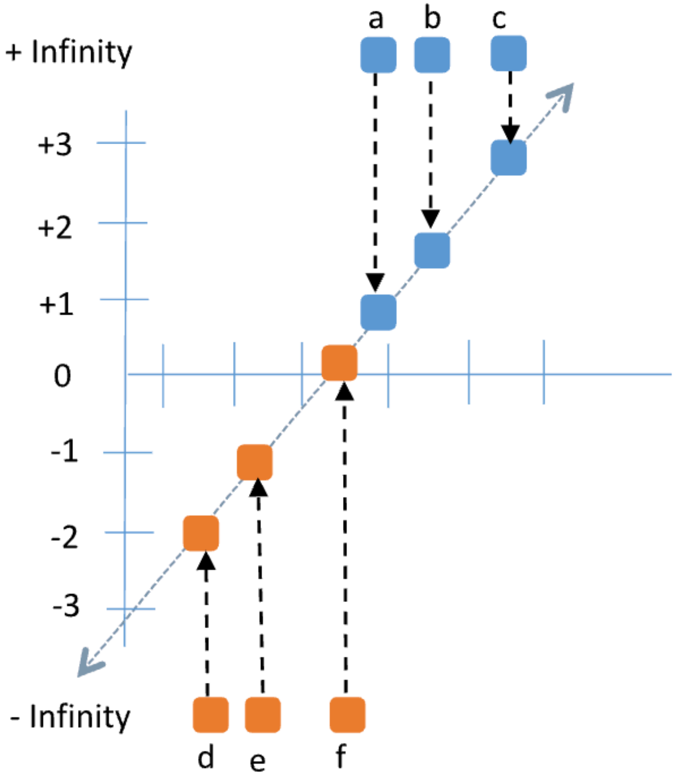


Numerical

Consider a data set of 6 individual where 3 people are diagnosed with type 2 diabetics (T2D) and 3 are non-diabetic control. Below given plot shows the probability of having T2D on the y axis and blood glucose level on the x axis. Now, we have transformed the probability of T2D into log(odds of T2D) and draw the candidate best fitting line.

If the log odds value of candidate data point is -1.9 what will be the candidate probability of sample ‘d’ being non-diabetic ? Also calculate the log of likelihood of overall probability of T2D.





Numerical

Consider a data set with 6 mice where 3 are obese and 3 are not obese. We have calculated the log(odds) of obesity for each candidate date by fitting line X.

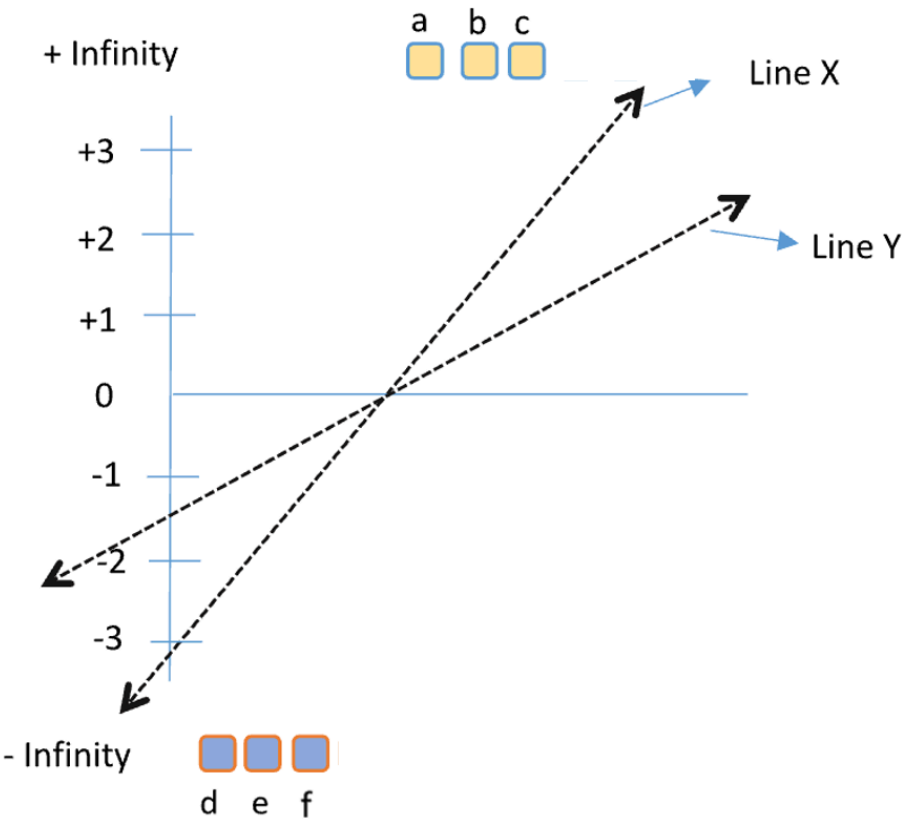
The log (odds) of each candidate data point for line X is as follows:

log(odds) of a = +.3, log(odds) of b = +1.2, log(odds) of c = +2, log(odds) of d = -1.8, log(odds) of e = -1.2, log(odds) of f = -0.1

Next, we rotate the line (Y) and calculate the log(odds of obesity) for all the candidate data.

The log (odds) of each candidate data point for line Y is as follows:

log(odds) of a = +0.2, log(odds) of b = +0.5, log(odds) of c = +0.8, log(odds) of d = -0.9, log(odds) of e = -0.5, log(odds) of f = -0.2



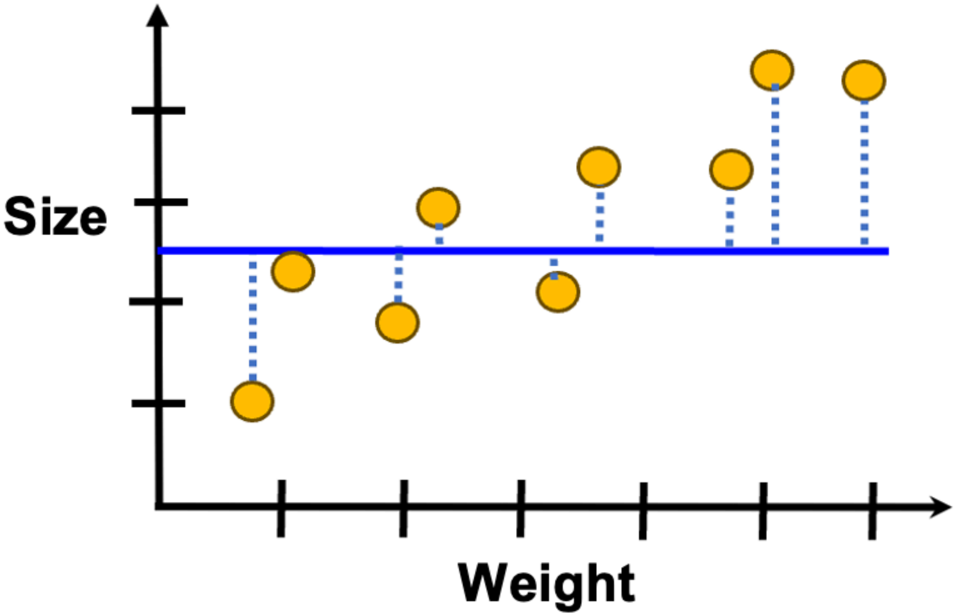
(a) Calculate the log(likelihood) of all the given data points when fitting line X

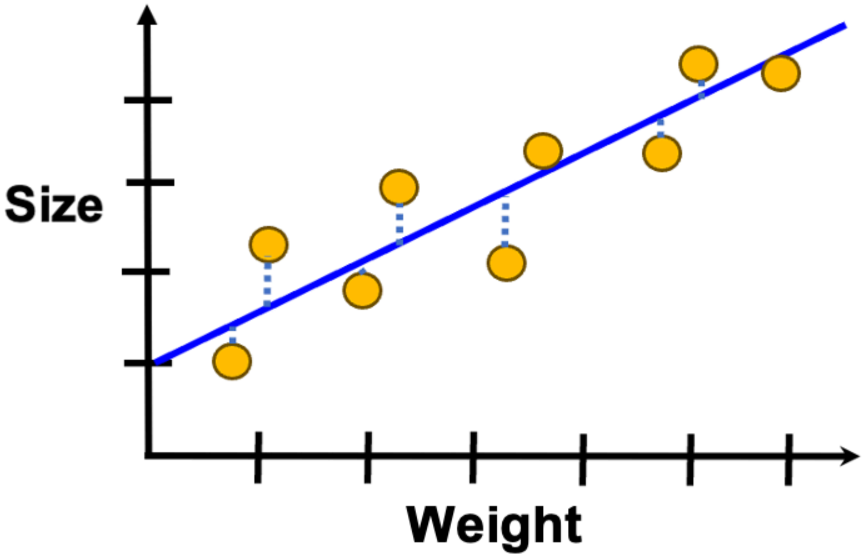
(b) Calculate the log(likelihood) of all the given data points when fitting line Y

(c) Which line can be considered the best fitting line for the above scenario and why?

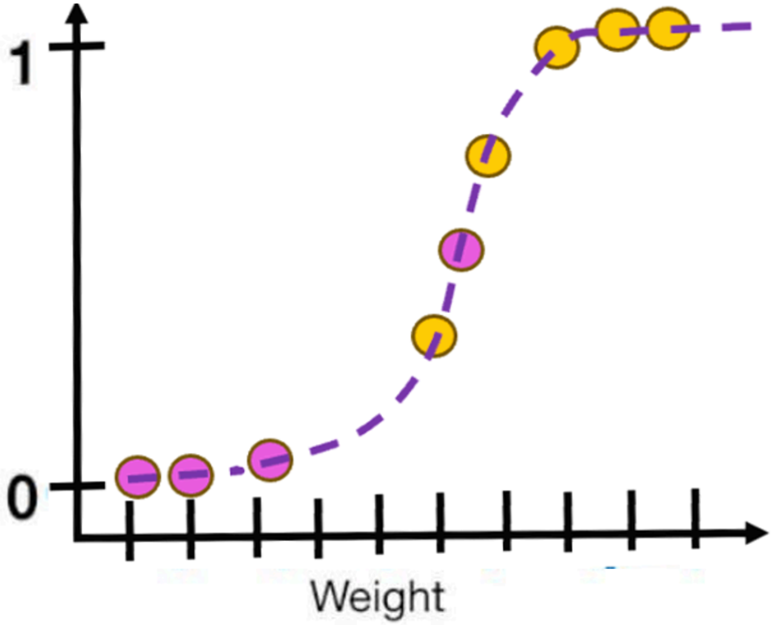
R^2 compares a measure of a good fit, SS(fit) to a measure of a bad fit, SS(mean)

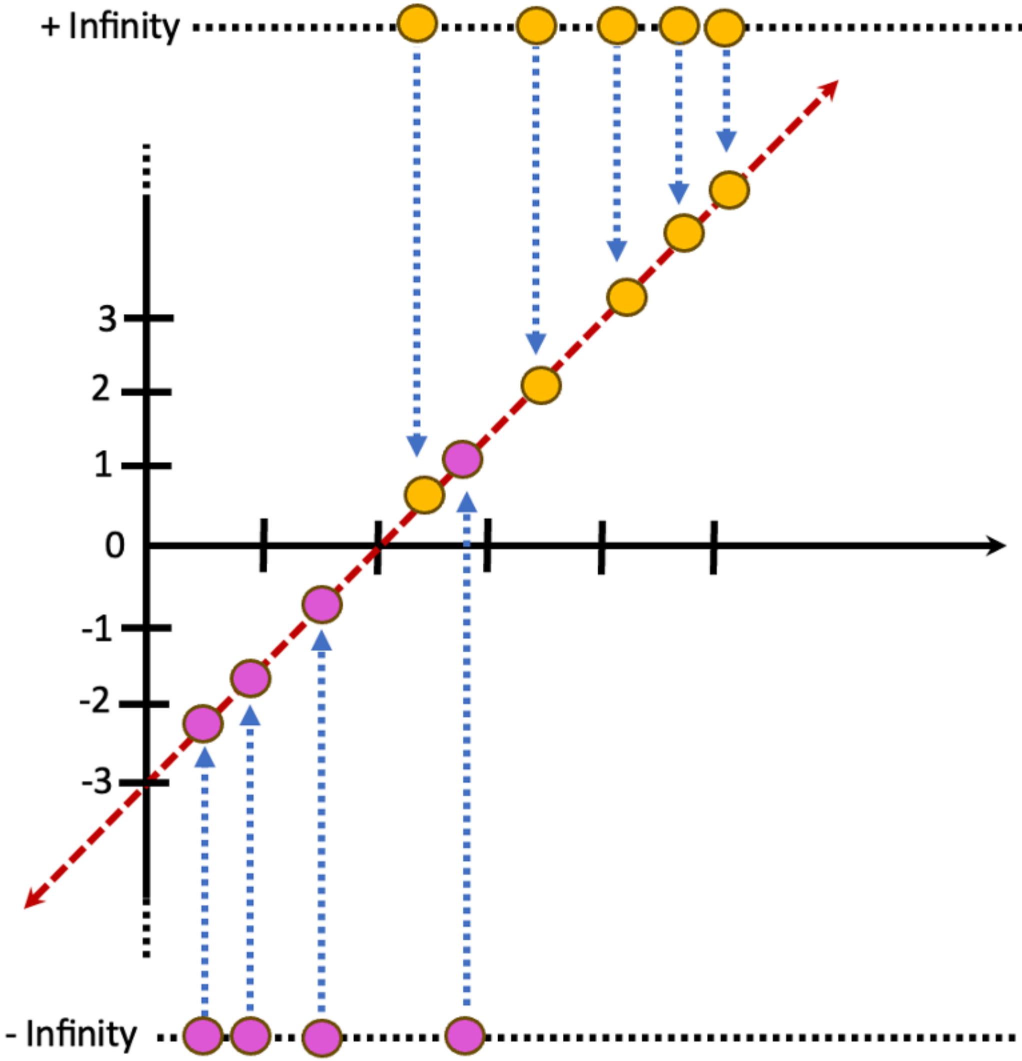
$$R^2 = \frac{SS(mean) - SS(fit)}{SS(mean)}$$





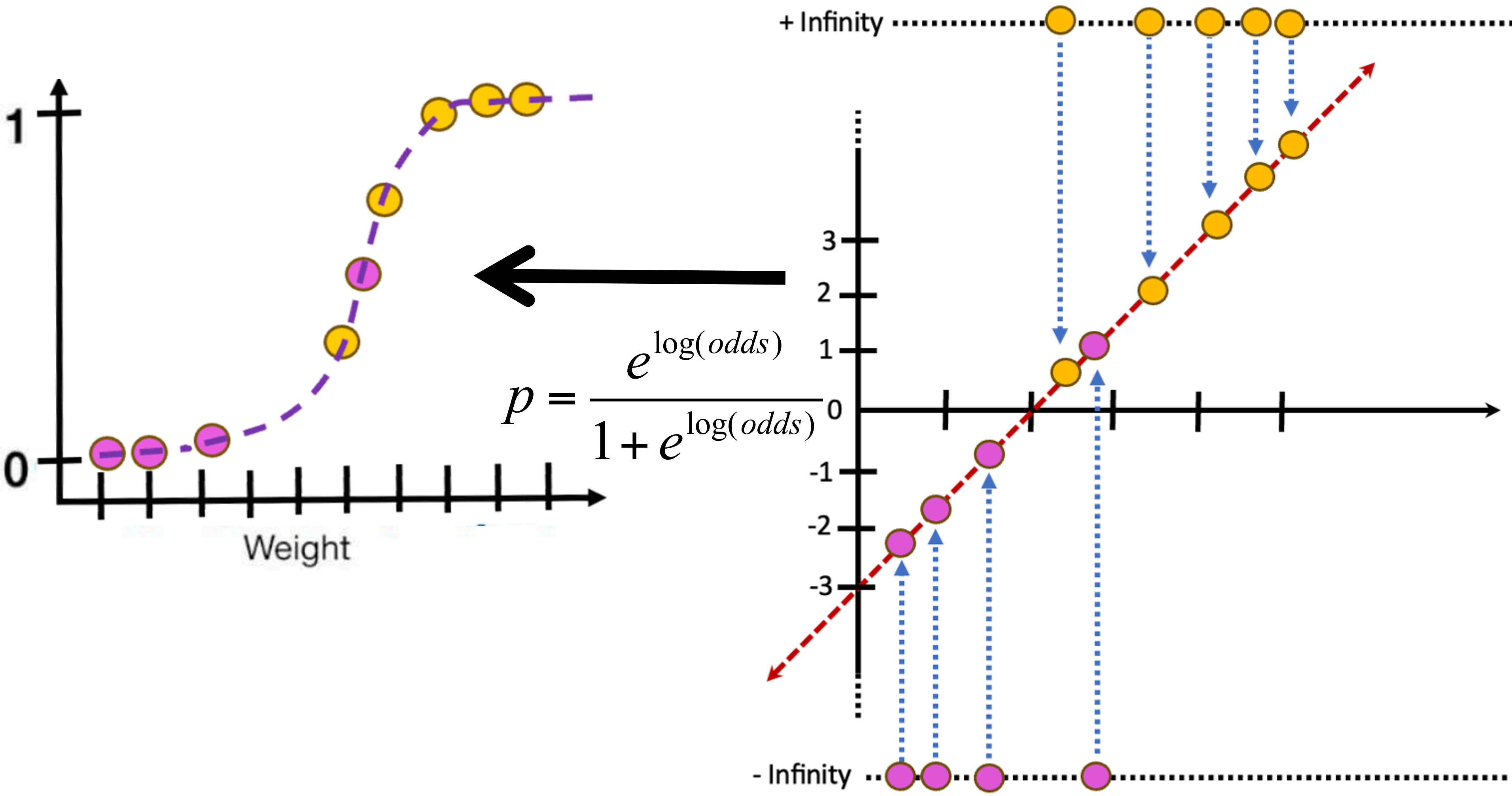
Like linear regression, we need to find a measure of a good fit to compare with bad fit





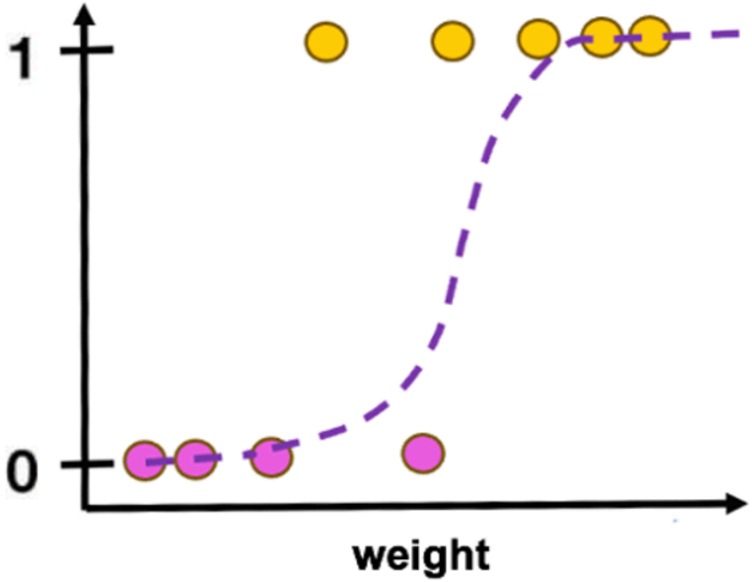
Unfortunately, the residuals fo
Logistic Regression are all infi
so we can't use them

Project the data onto the best fitting line and then translate the log(odds) to probabilities



$$p = \frac{e^{\log(odds)}}{1 + e^{\log(odds)}}$$

Lastly, calculate the log-likelihood of the data given the best fitting squiggle



Log(likelihood of data given the squiggle)
= log(0.4) + log(0.85) + log(0.9) + log(0.98) + log(0.99) +
log(1 - 0.6) + log(1 – 0.03) + log(1 – 0.01) + log(1 – 0.01) = -2.1813

We can call this LL(fit), for the log-likelihood of the fitted line, and use it as substitute for SS(fit)

LL(fit)= -2.1813

We need a measure of a poorly fitted line that is analogous to SS(mean)

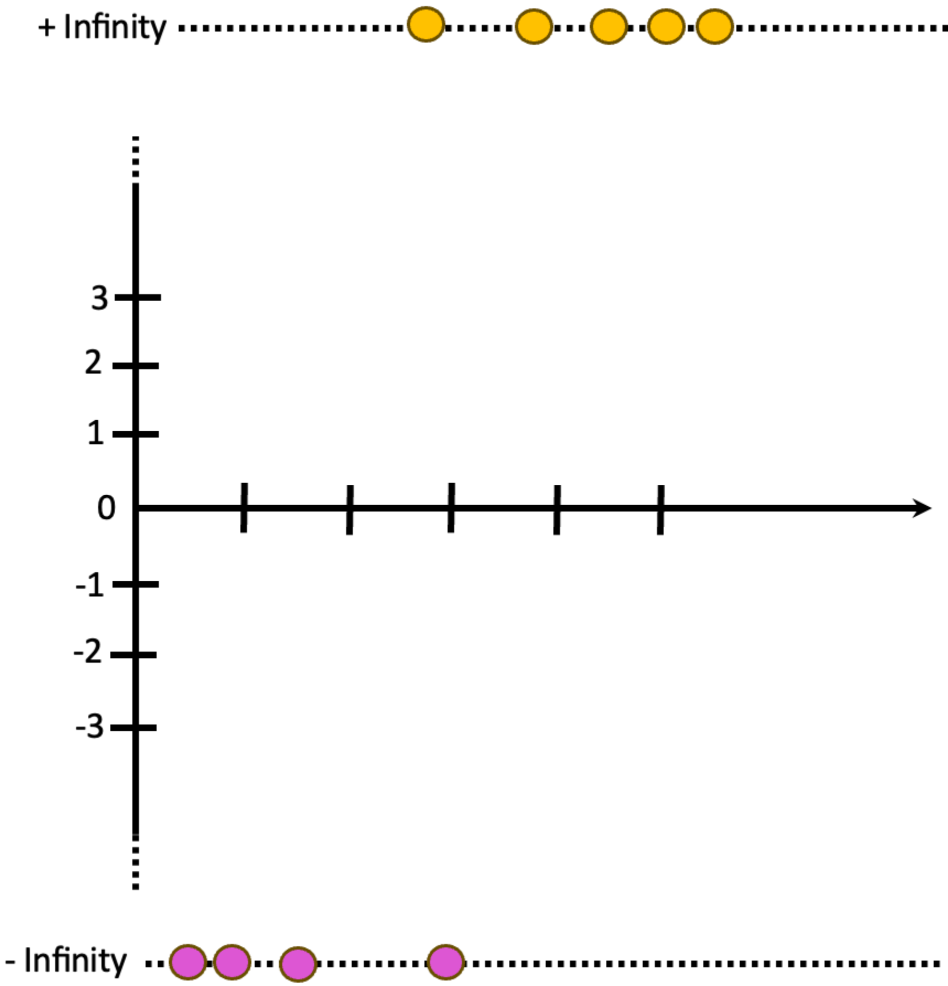
$$R^2 = \frac{SS(mean) - SS(fit)}{SS(mean)}$$

$$R^2 = \frac{??? - LL(fit)}{???}$$

Log(odds of obesity)

$$= \log\left(\frac{no\ of\ \ obese\ mice}{total\ \text{no}\ of\ \ mice\ not\ obese}\right)$$

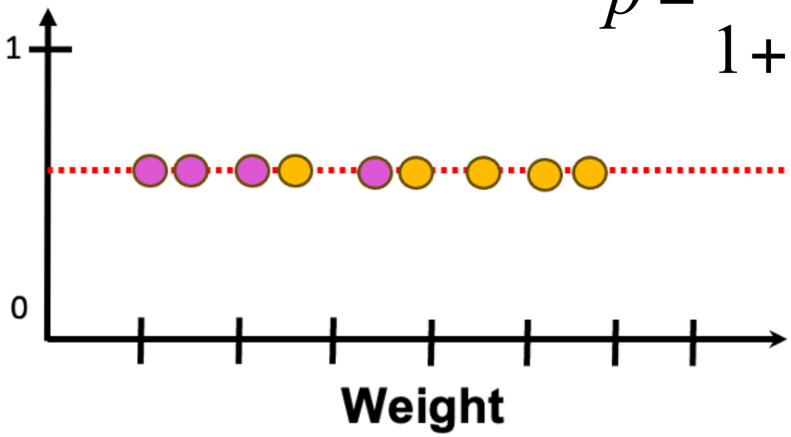
$$= \log\left(\frac{5}{4}\right) = 0.22$$

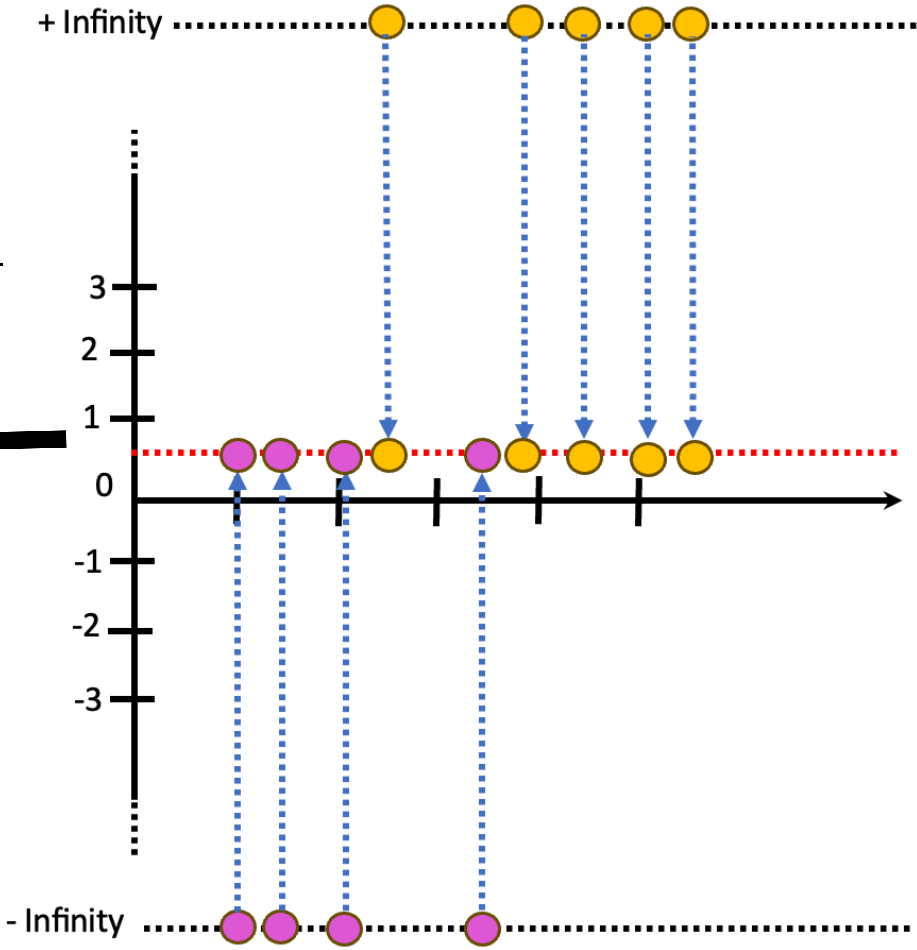


Translate the log(odds) back to probabilities

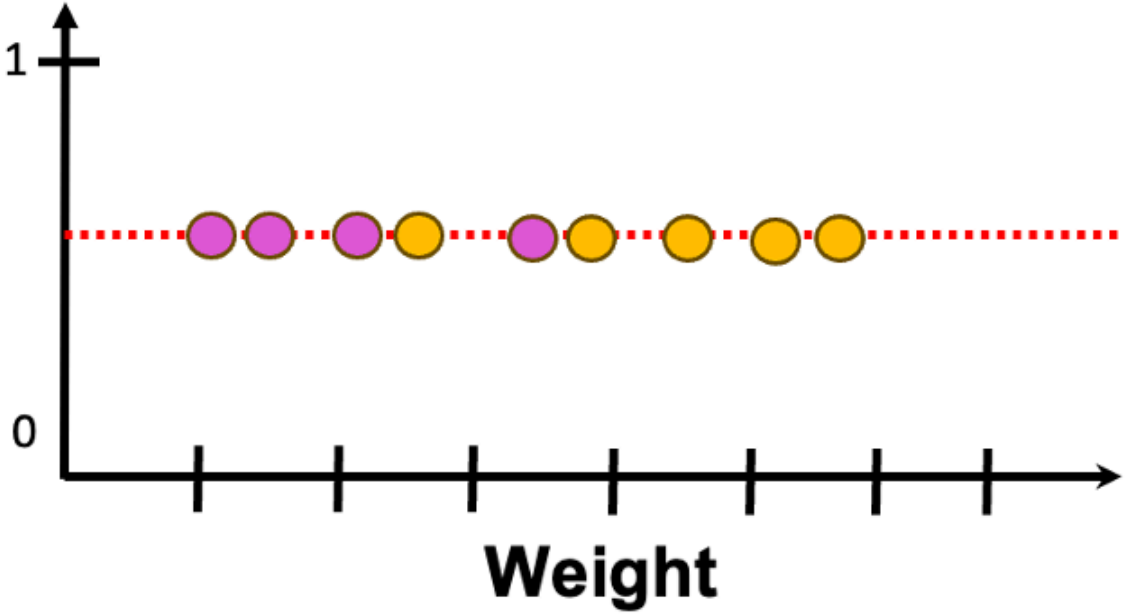
$$p = \frac{e^{0.22}}{1 + e^{0.22}} = 0.56$$

$$p = \frac{e^{\log(odds)}}{1 + e^{\log(odds)}}$$



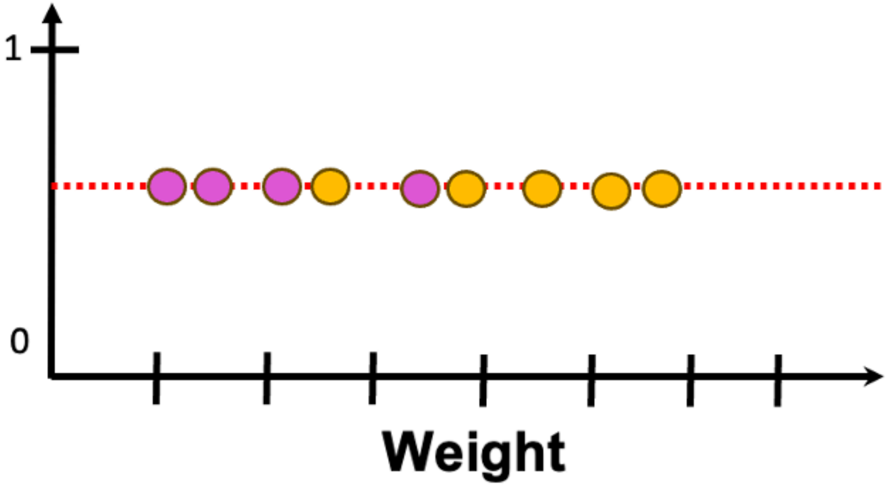


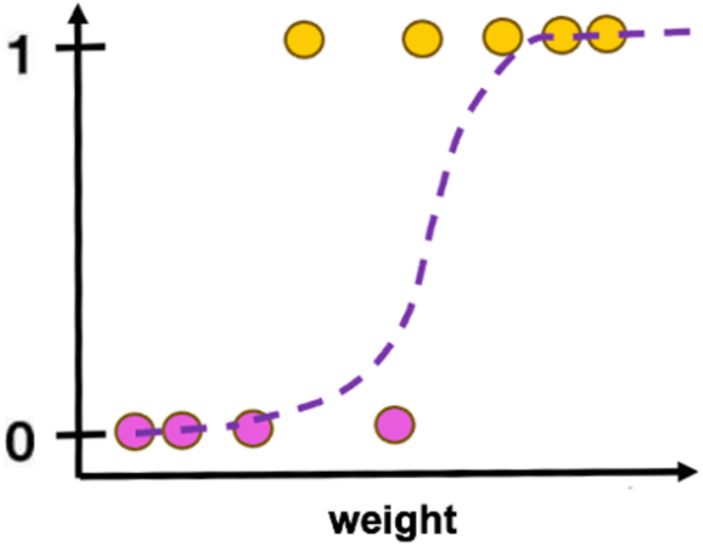
Calculate the log-likelihood of the data given the overall probability of obesity



Log(likelihood of data given overall probability of obesity)
= log(0.55) + log(0.55) + log(0.55) x log(0.55) + log(0.55) +
log(1 – 0.55) + log(1 – 0.55) + log(1 – 0.55) + log(1 – 0.55)
= -6.18

LL(overall probability)= -6.18





LL(overall probability), a measure of a bad fit

LL(fit), hopefully a measure of a good fit

$$R^2 = \frac{LL(overall\ \text{probability}) - LL(fit)}{LL(overall\ \text{probability})}$$

$$R^2 = \frac{-6.18-\left(-2.1813\right)}{-6.18} = 0.6475$$

Numerical

Calculate the log-likelihood of the data given the the best fitting squiggle for malignant tumour. Then calculate for R^2.

Malignant	Non-Malignant
0.45	0.001
0.9	0.002
0.91	0.005
0.95	0.2
0.99	0.34