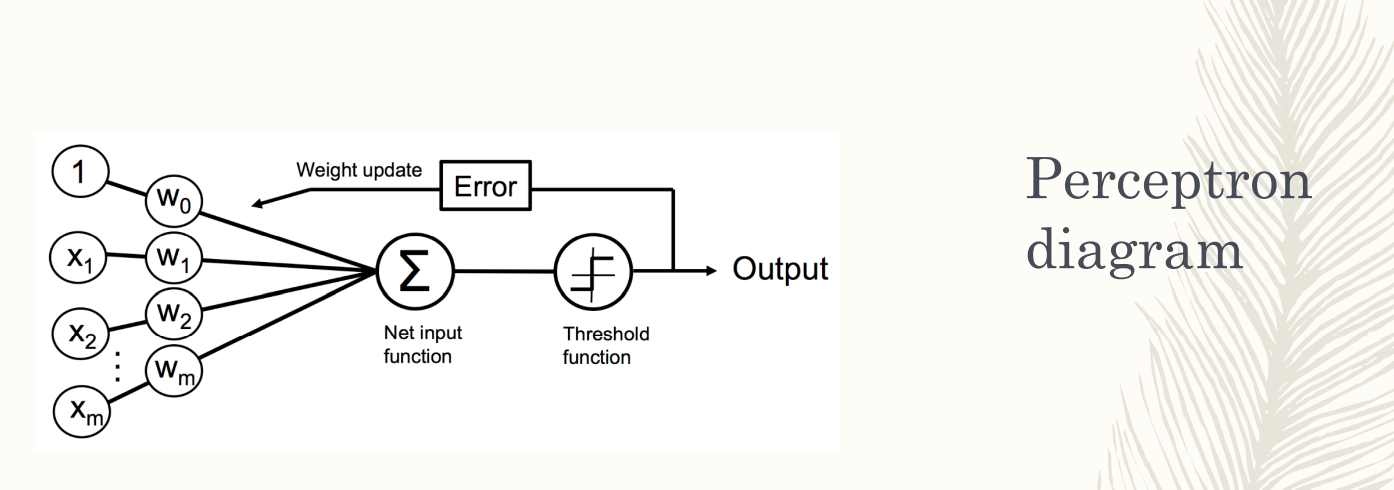
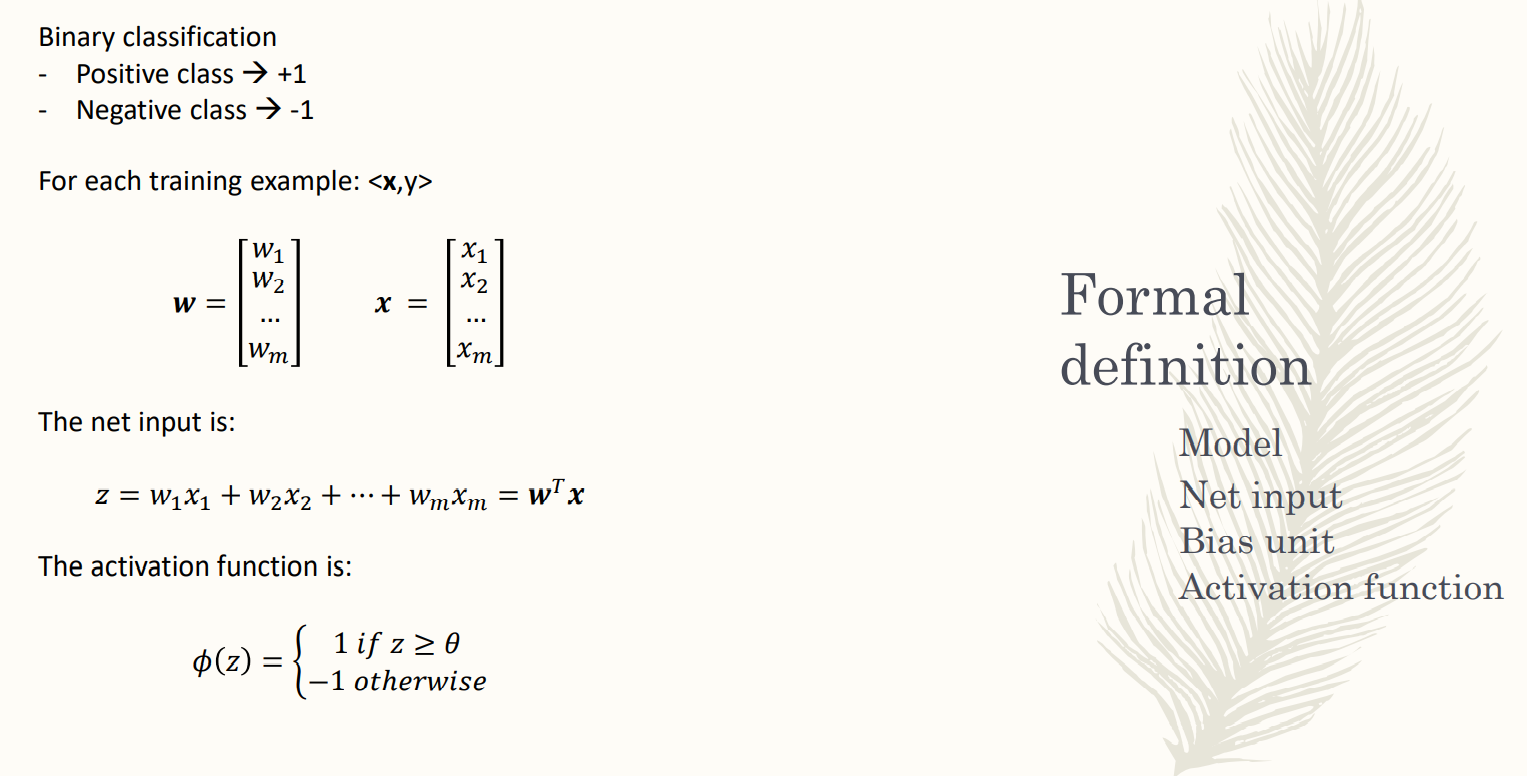
**Supervised classification**

Perceptron and adaline

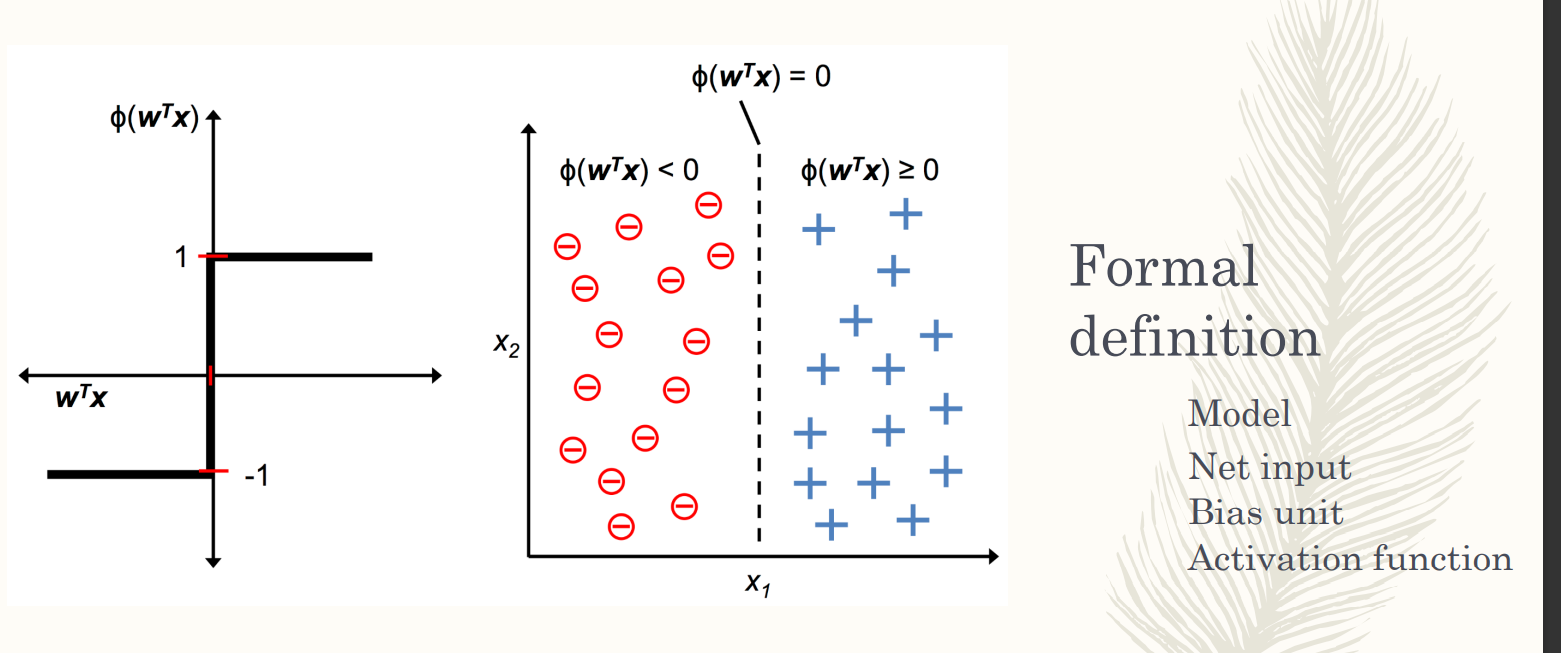
Perceptron take an input and features



we want to learn the weight



use activation function,



find hyperplane that divide the positive and negative

**ADALINE ADAPTIVE LINEAR NEURON**

It is similar to the perceptron but introduce activation function that is linear, we obtain that the cost function that we have to optimize become continuous and we can use the descent gradient



**LOGISTIC REGRESSION**

****

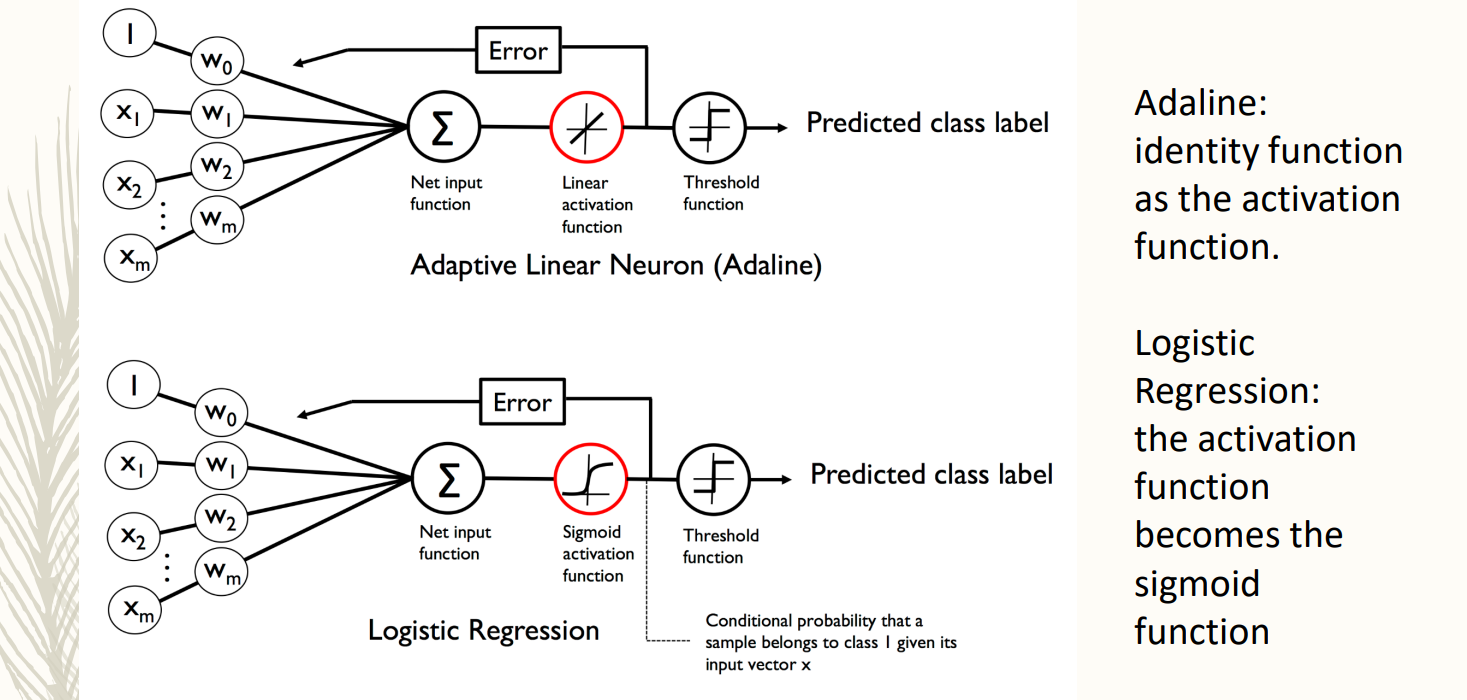
Perceptron problem: if the classes are not linearly separable the perceptron does not separate. Logistic regression perform very well on linear and almost-linear classes

It performs very well and is well adopted. When working in the industry and we don’t have much data you can use logistic regression for good results. Widely use even if it is old.

Main advantage of logistic regression. Assign each instance to a class, and model the probability belonging to a class. **The output gives you the probability that each instance belongs to a class or another**.

It is not 0 or 1 anymore, now the logistic regression tells which is the class to which an instance is assigned and the probability. Knowing the probability allows to have an assessment to how reliable it is

With logistic regression we don’t have a convergence problem. You still have the converge of the algorithm



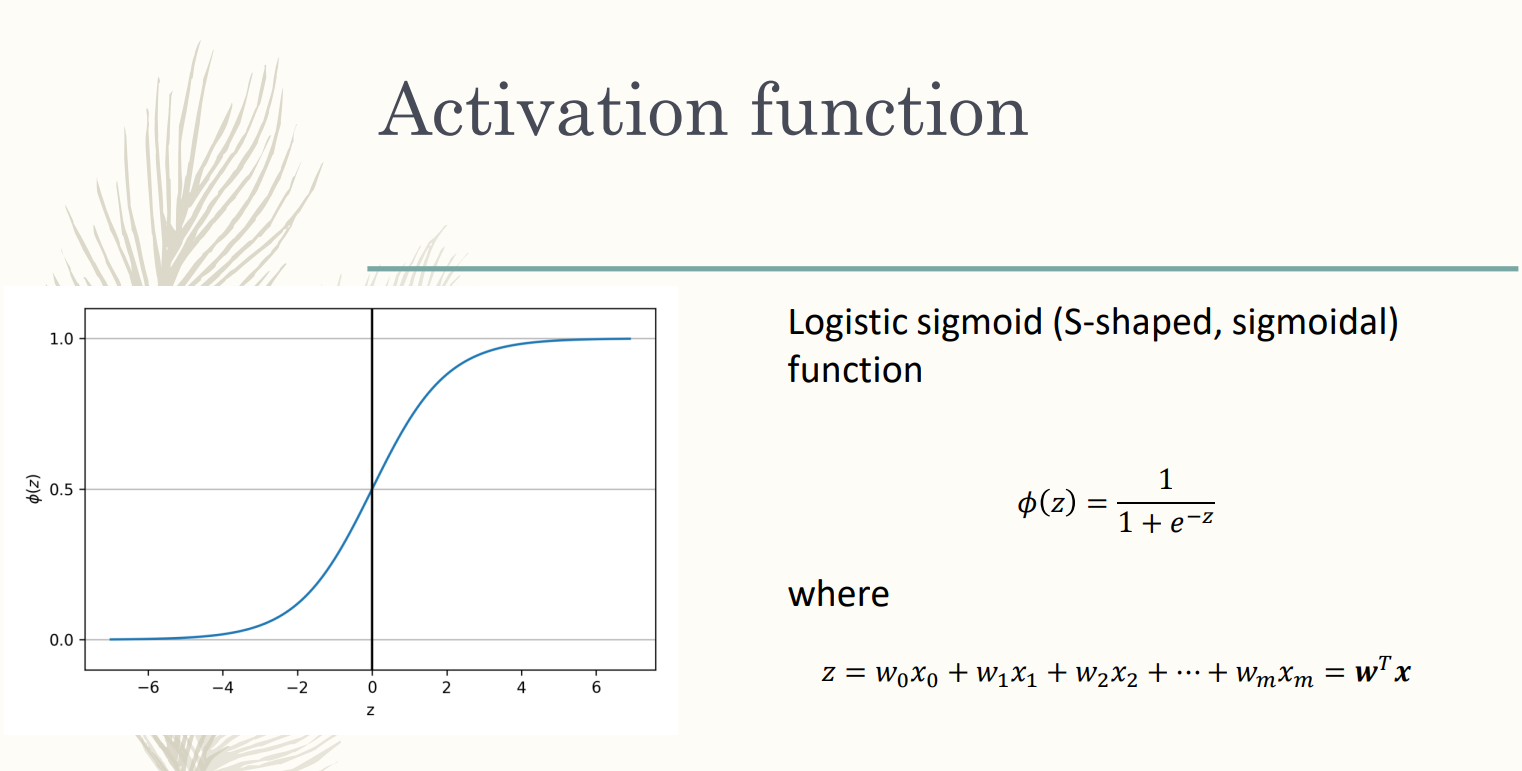
Logistic regression has a similar idea of perceptron.

The learning is about learning the optimal way of the feature.

As we do with the perceptron, the logistic regression still has the net input function. **It changes the activation function**. In adaline is linear activation function (does not change anything but makes the cost function continuous)

Logistic regression uses sigmoid activation function. Takes an input the weighted sum of the features and manipulates them by the sigmoid activation function. You have to introduce the threshold function to find which class it belongs to.

The main point for logistic regression is that it introduce the sigmoid activation function.

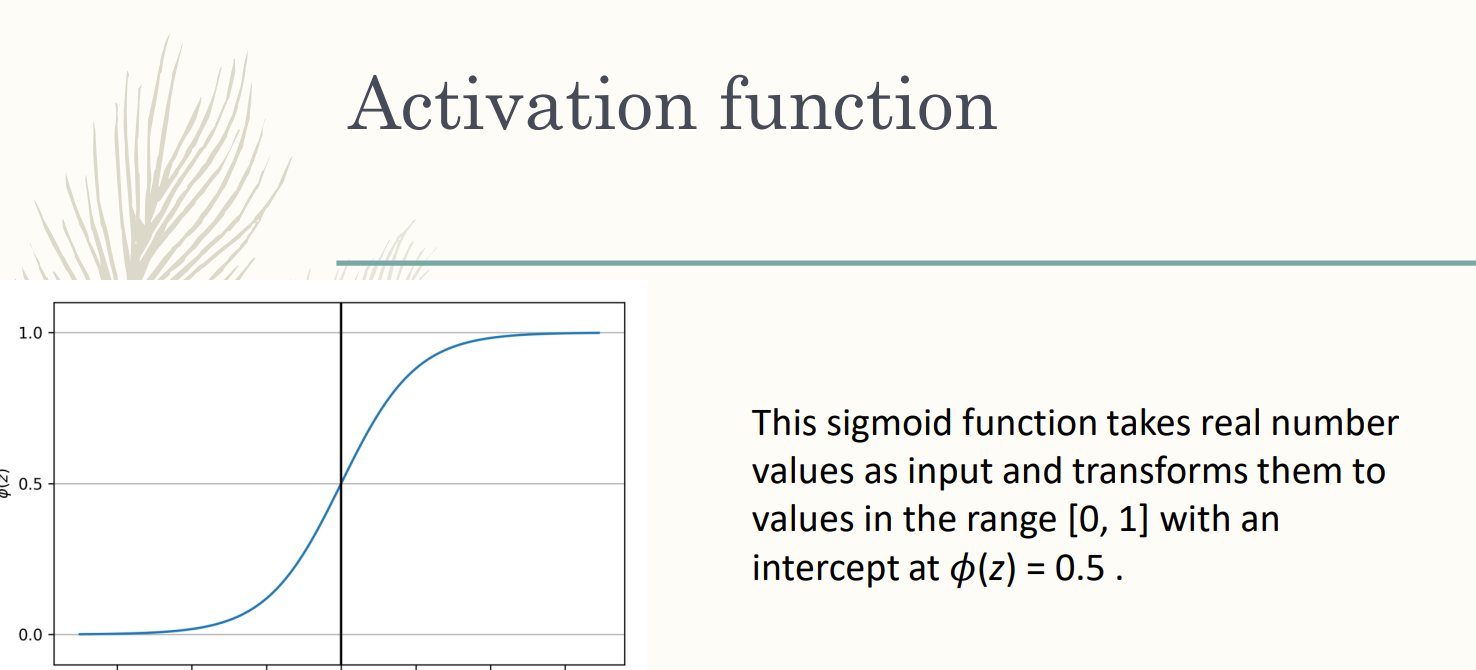


Sigmoid: S shape function/logistic function

takes value from all the real numbers (X) the input function is the weighted sum, and can take value on each possible number. The output (Y) is between 0 and 1. It means that you take the linear combination of the feature and transform each combination into a number in the range 0-1.

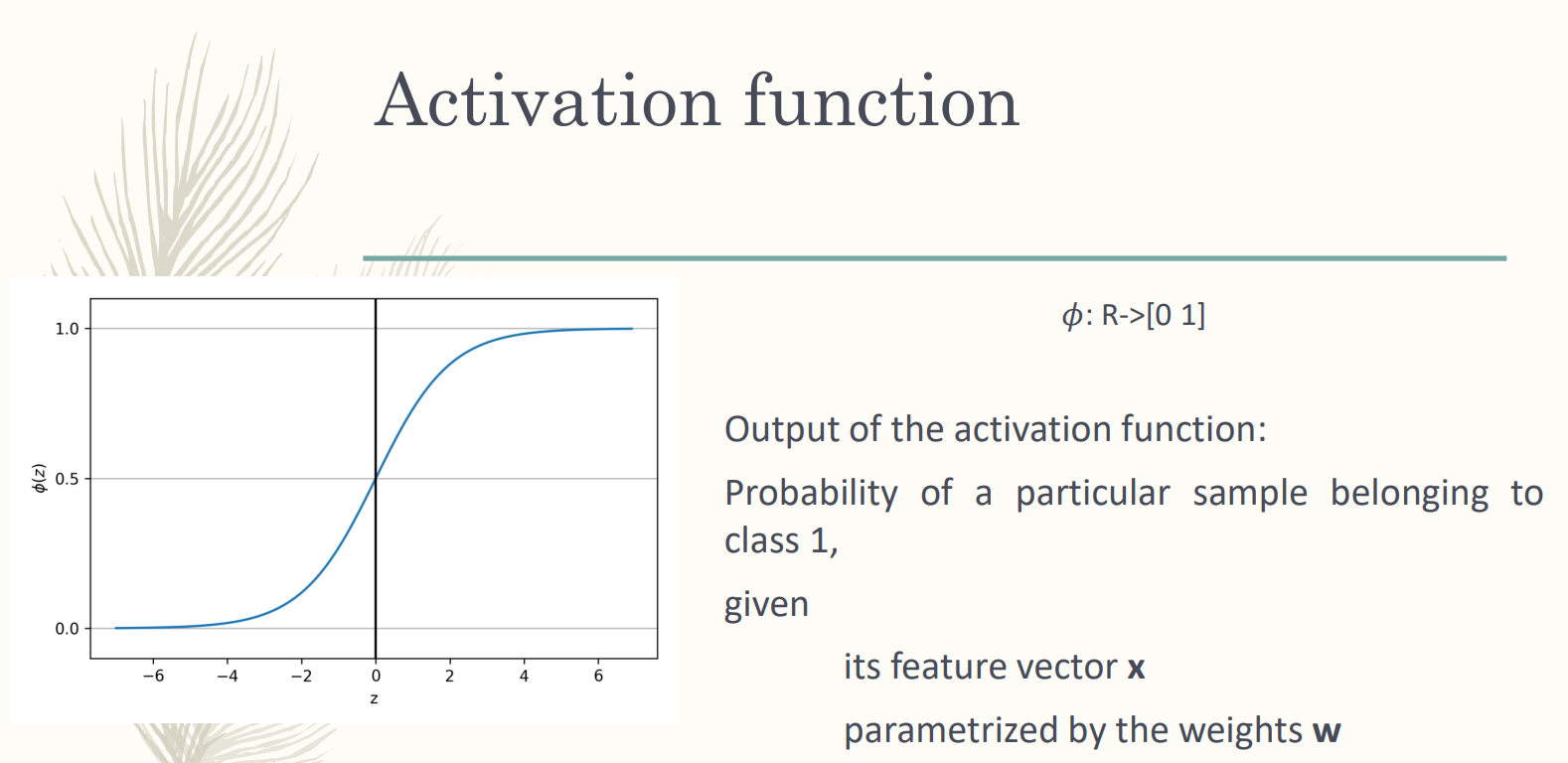
If we are able to transform each possible value of the weighted combination for a feature in a number between 0 and 1 we can say that this number is the probability.

FORMULA: not linear, we have to transform real number in number between 0 and 1.

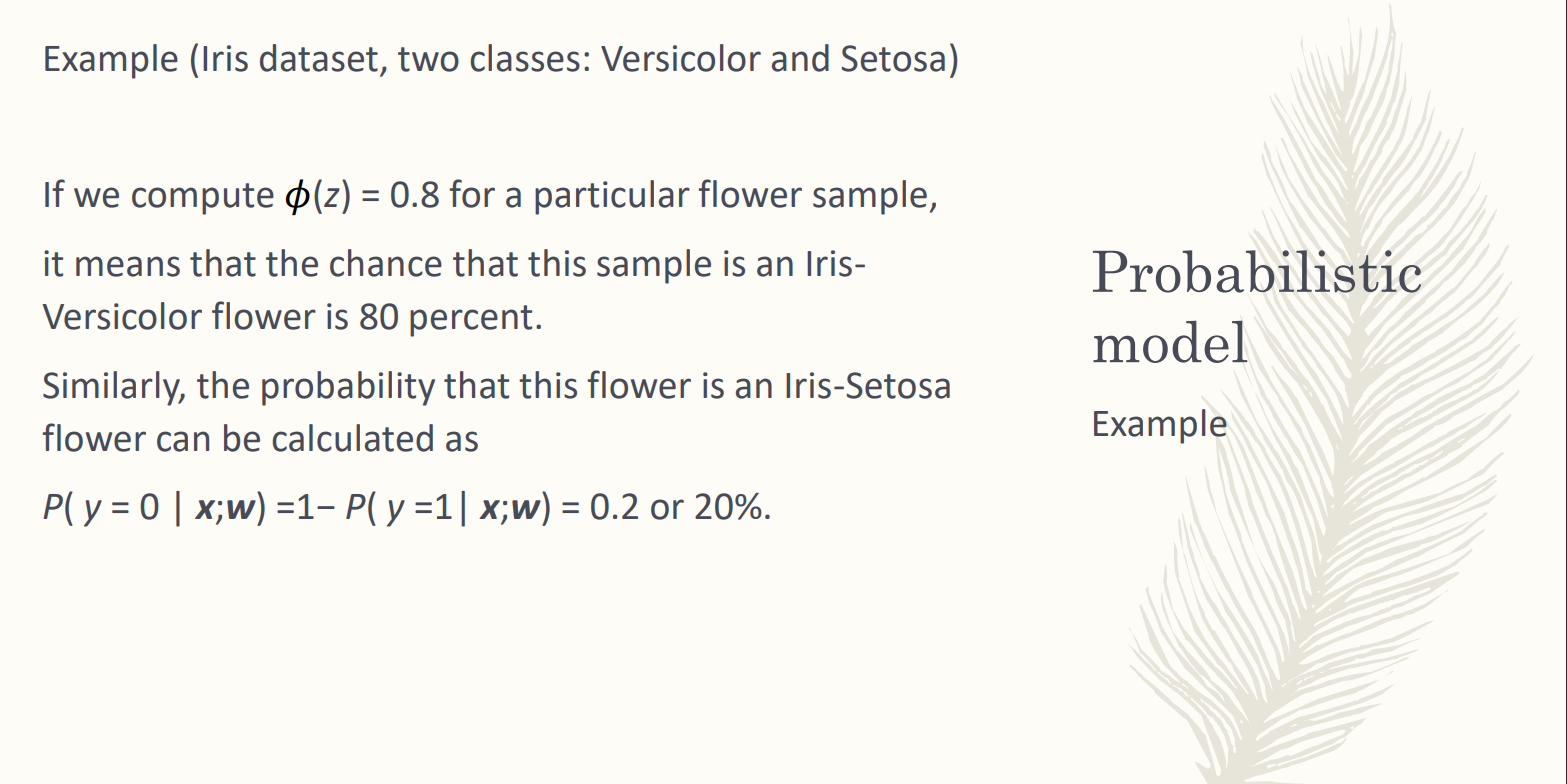


The intercept of this activation function is 0.5. ALl combustion of the feature is transformed by sigmoid function in a number between 0 and 1. If the combination is positive you transform it between 0,5 and 1

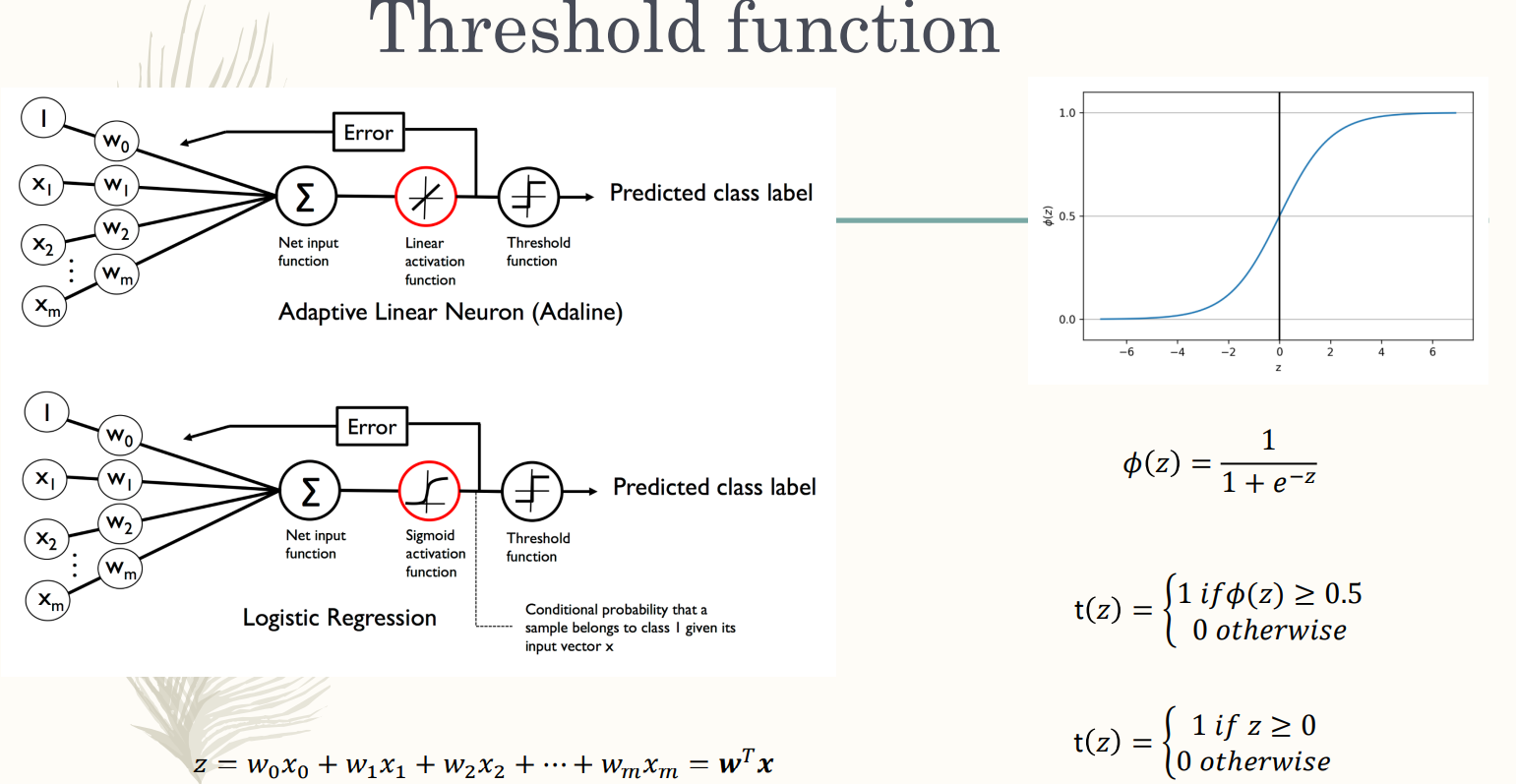
is not linear, you have different slopes around 0 and 1.



we have a number between 0 and 1. The output of the activation function gives you the prob, that the sample belongs to the class 1.



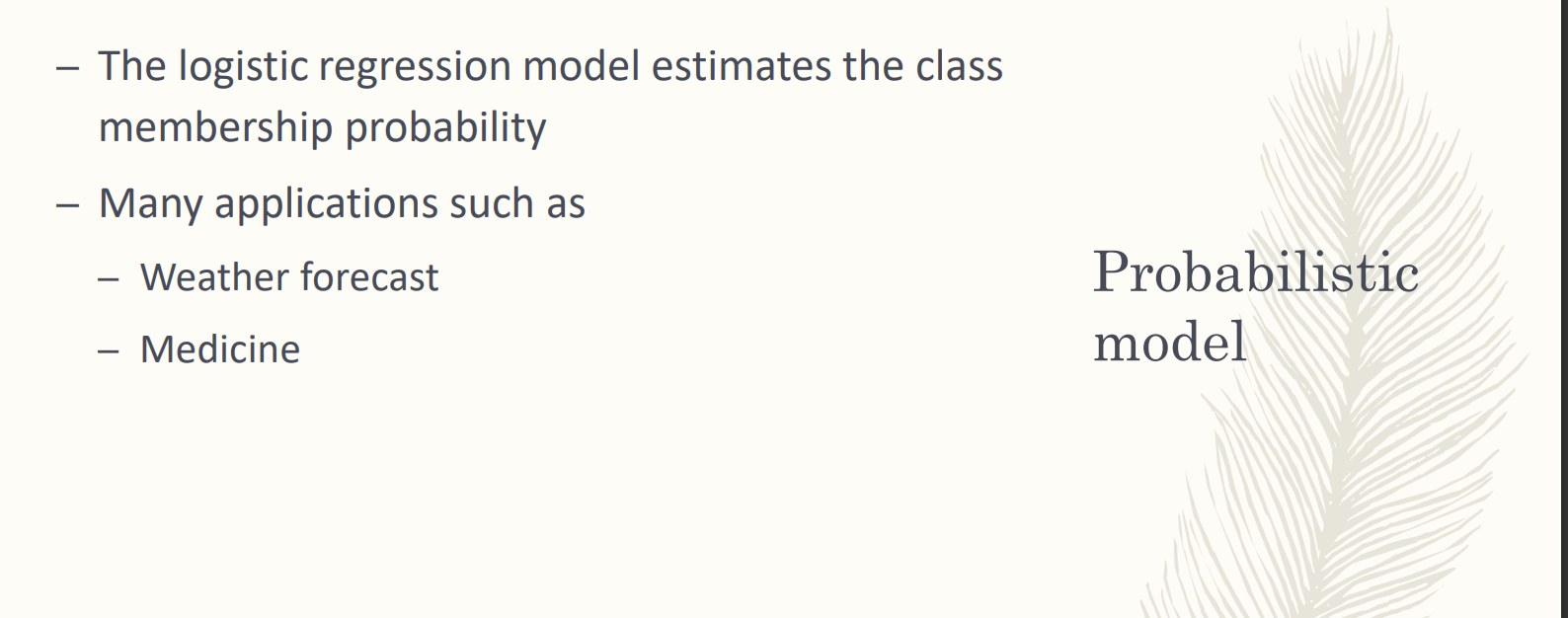
you train logistic regression, you have form a particular



Need threshold function to do a classification. Belongs to a class if the prob is greater than 0,5

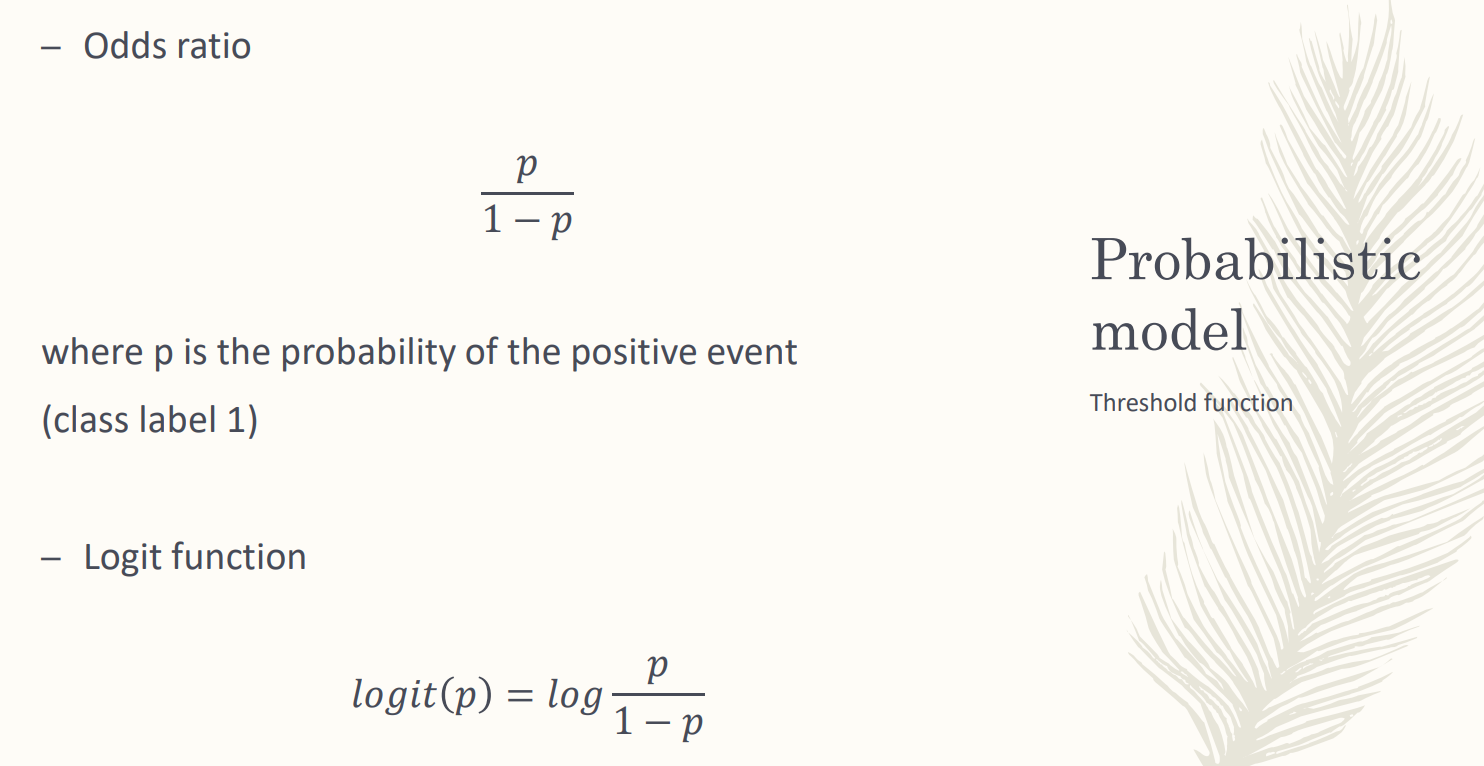
you could also use a different value for the threshold (can use 0.9 to be very precise).

can always fix the threshold function.



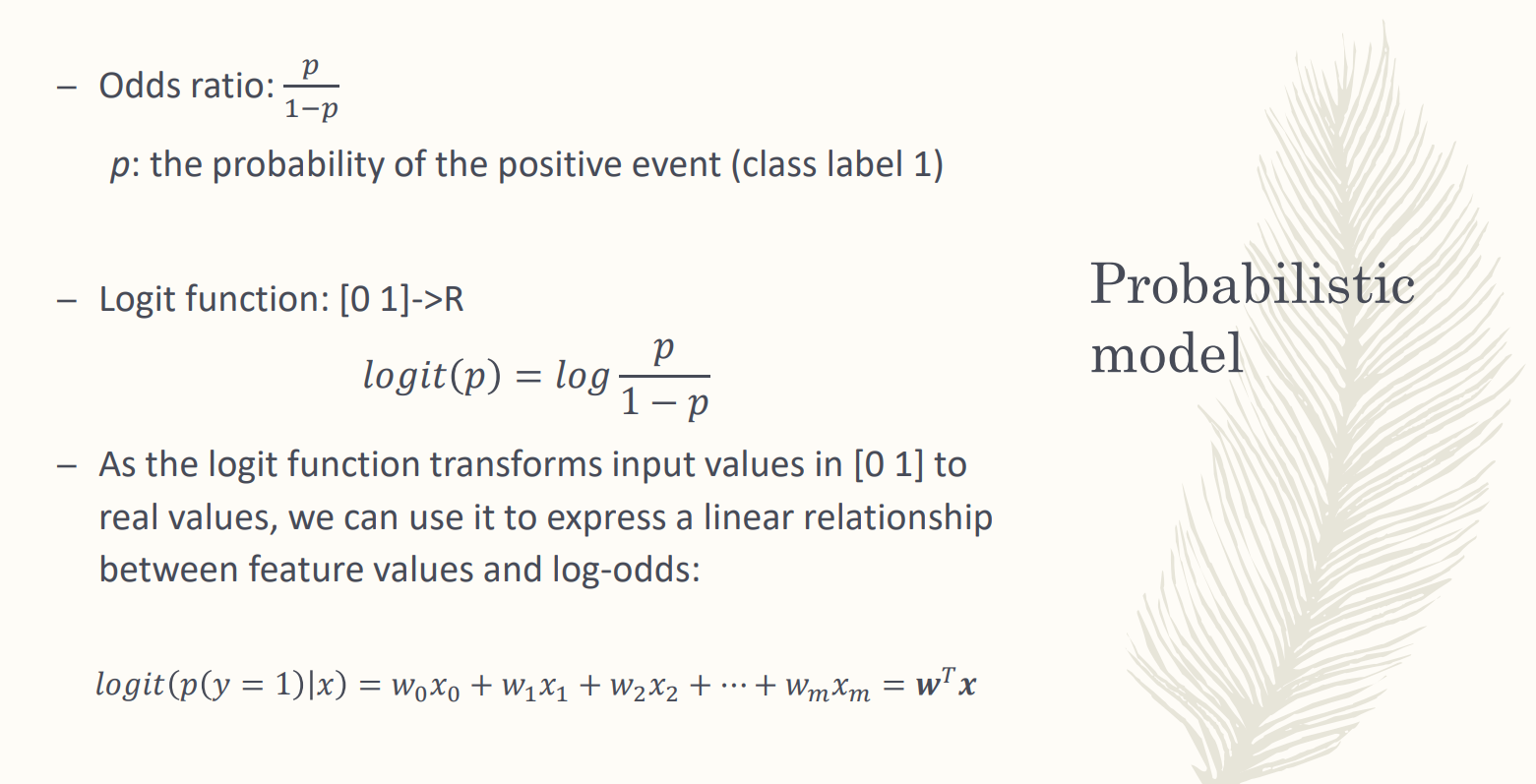
Many application areas that we need are probability and not only classification. More important to have the prob that someone has to take a drug or in the weather forecasting. I want to know the probability if it will rain or not.

You need to have the condition also.



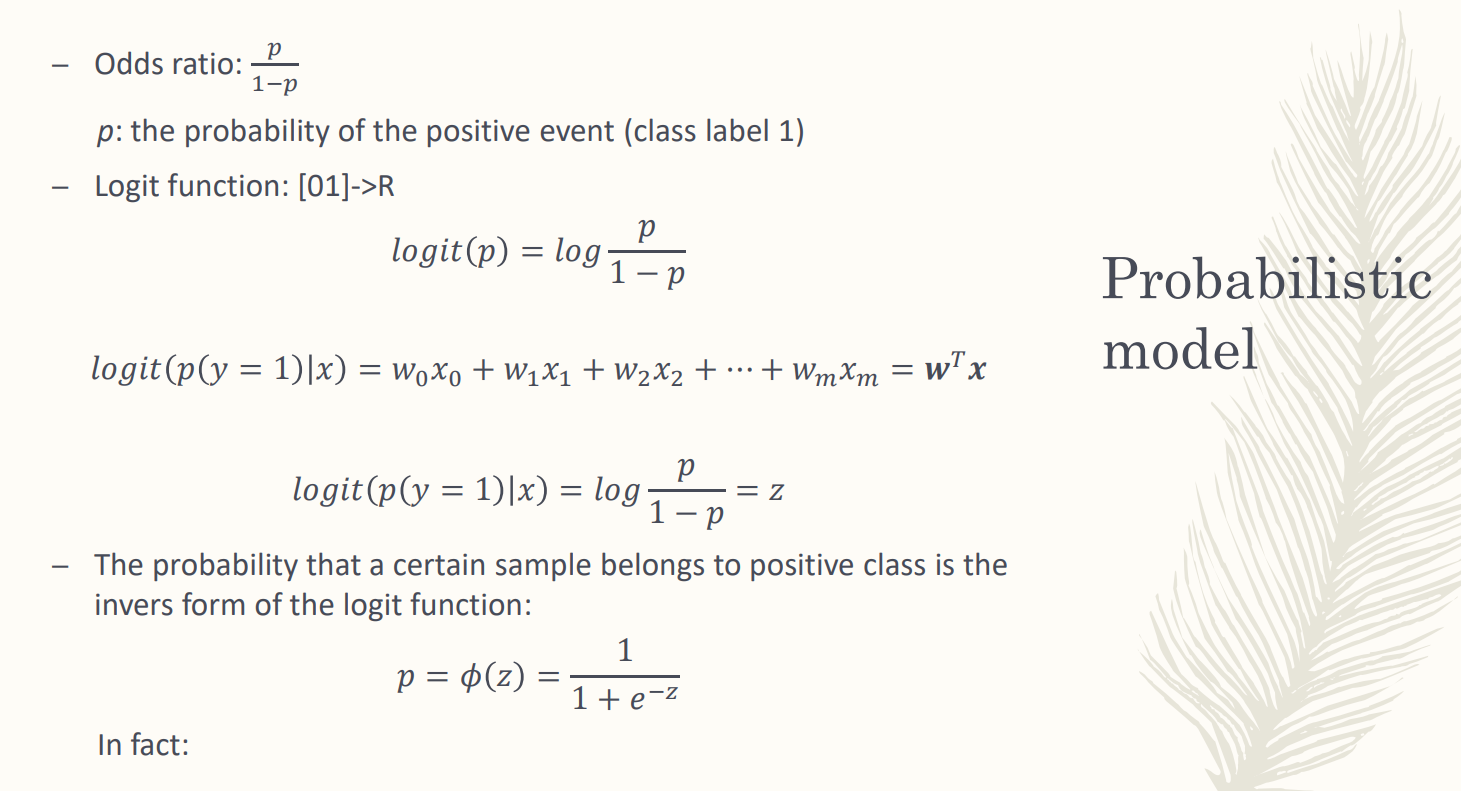
odds ratio: the ratio between the prob *p* of belonging to a class divided by 1-p (not belonging to the class).

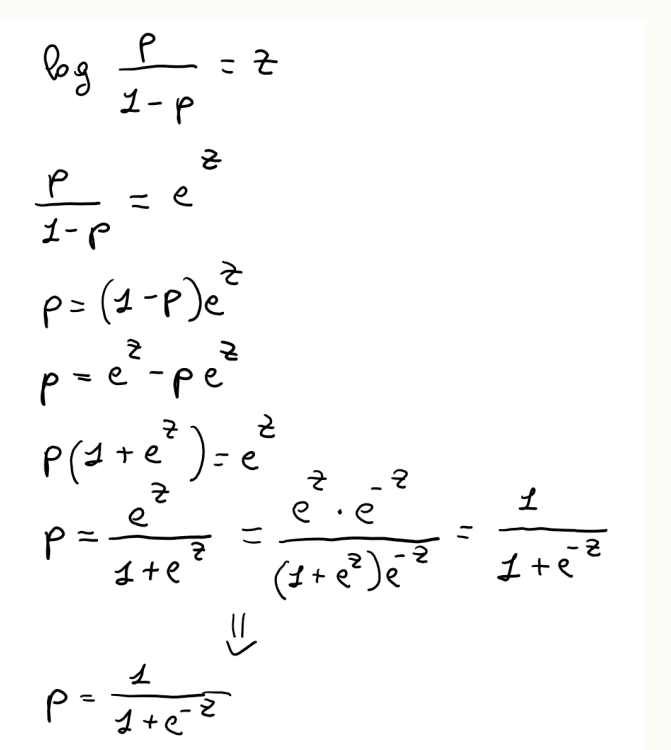
Usually we use the log (base e) off the odds ration.



**Logit function transform values from [0,1] to real values** (similar to the opposite of the sigmoid function that take real values and transform the in the range [0,1]) logic function takes values [0,1] and transform them in real values

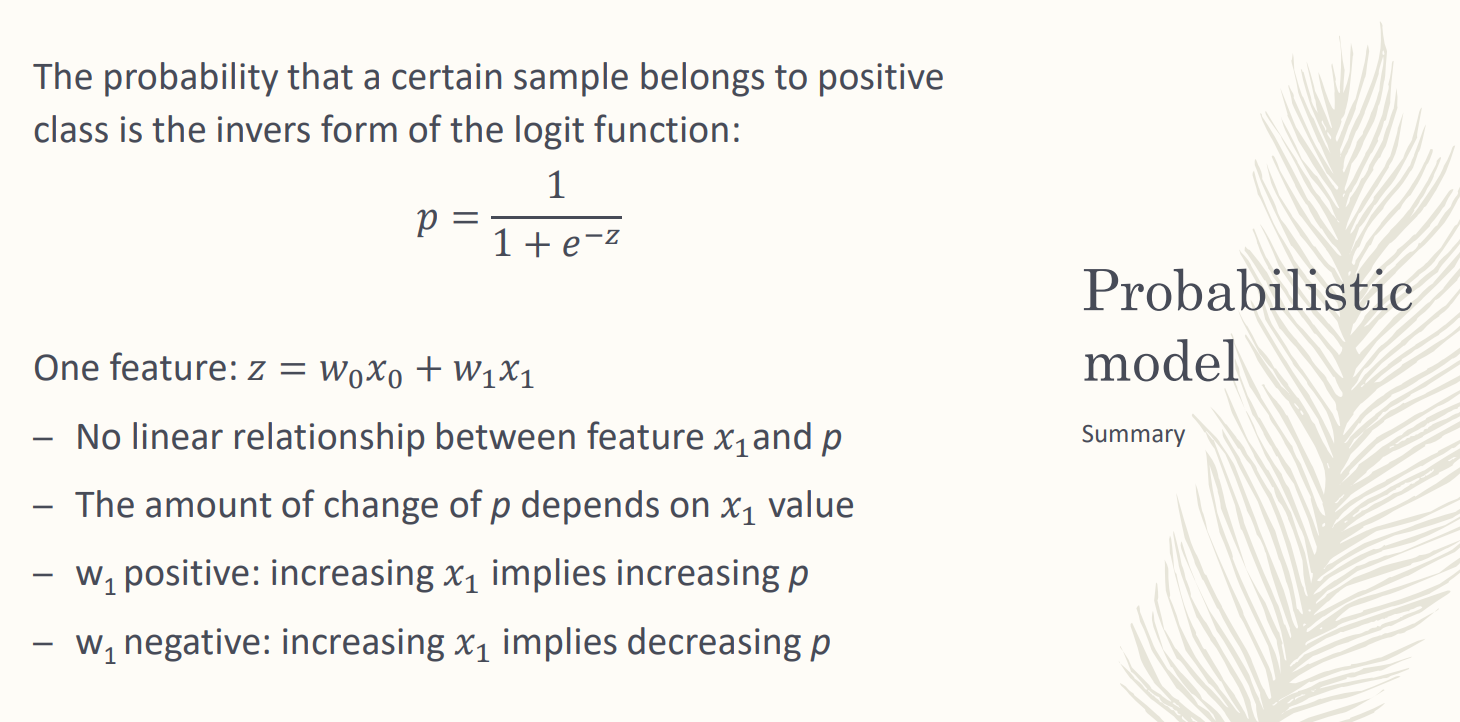
the logit function is the ned input (linear combination of the features)





The output of the activation function is equal to the sigmoid function (?)

the output of the activation function is the probability.



interpreting the output of the activation function as the probability. Sample with 1 feature, measure only the length of the pedal

This probability is not lineart with the net input. The prob. to belong to the +xompl

one feature ⟹ the net input is given by the bias unit and the feate x1 weighted by the weight of x1. no relation between the value of the feature and the probability of belonging to the class

there is a relation, but is not linear.

The probability to belong to the positive class depends on the value of the feature. Which is the difference in meaning of positive and negative weight.

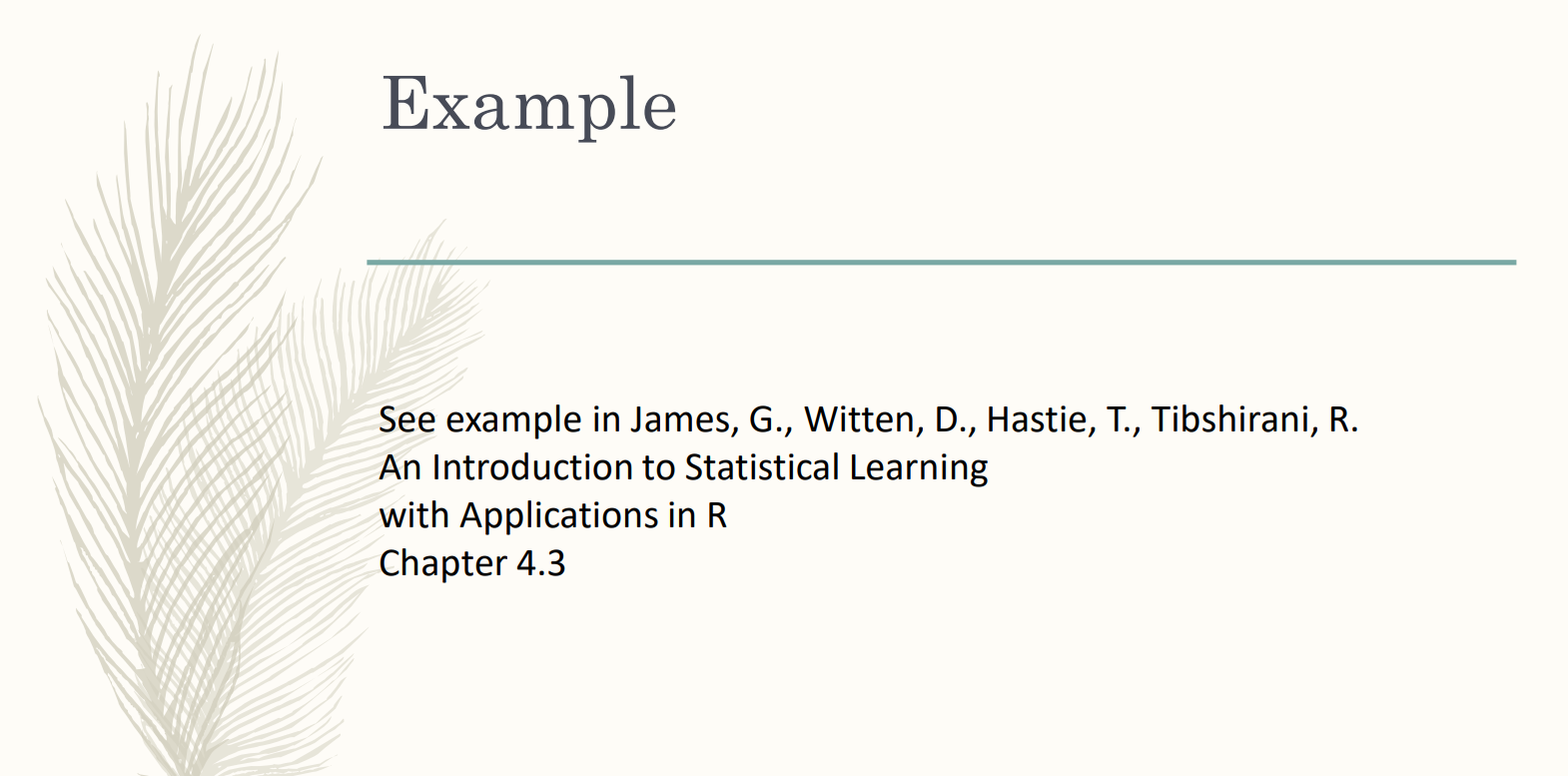
If the weight of the feature is positive, increasing the value of the feature. +compl

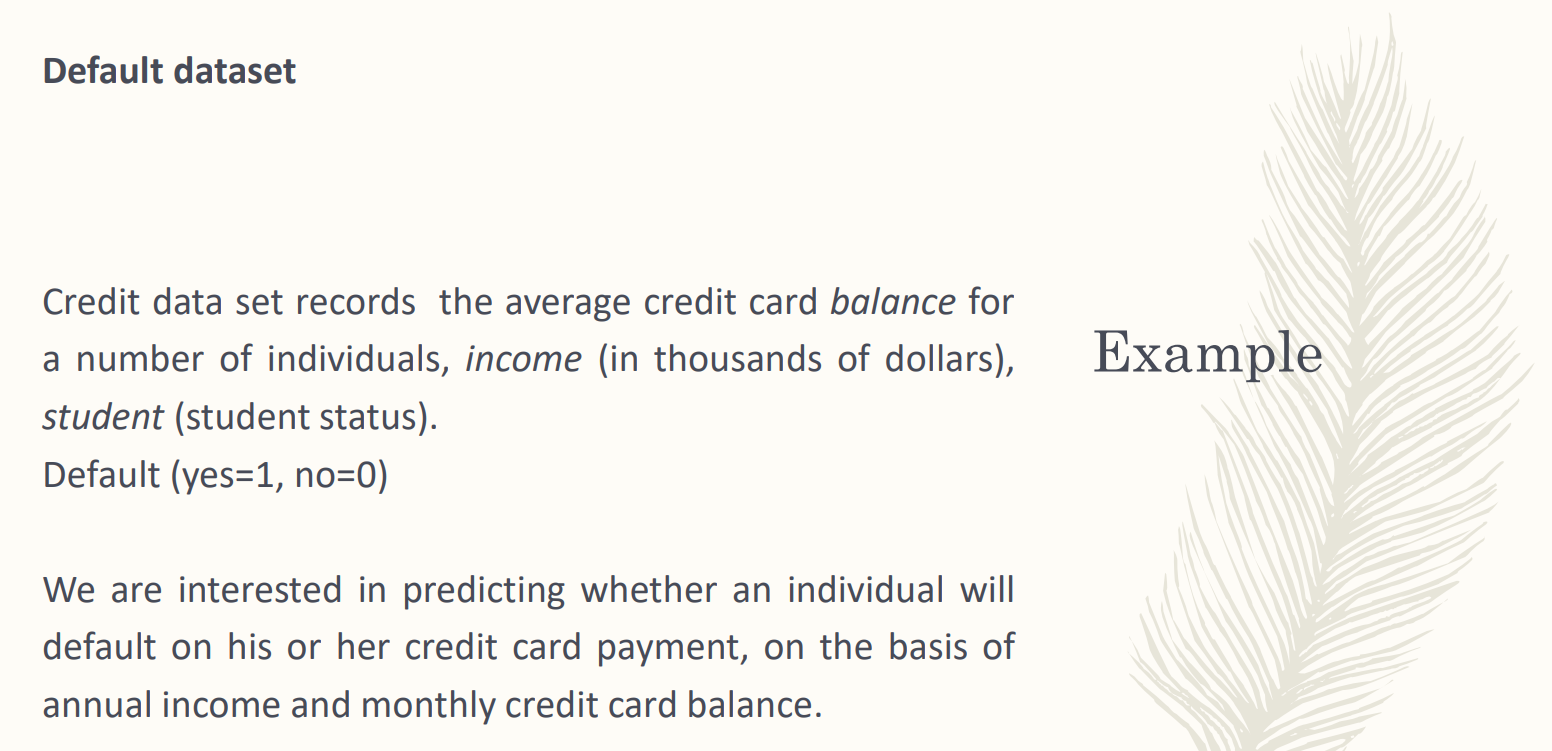
After training you get the model and you can interpret the weight immediately.

Negative weight for a feature: the feature works against the positive class. increasing value of feature implies increasing value of the probability to the positive class.

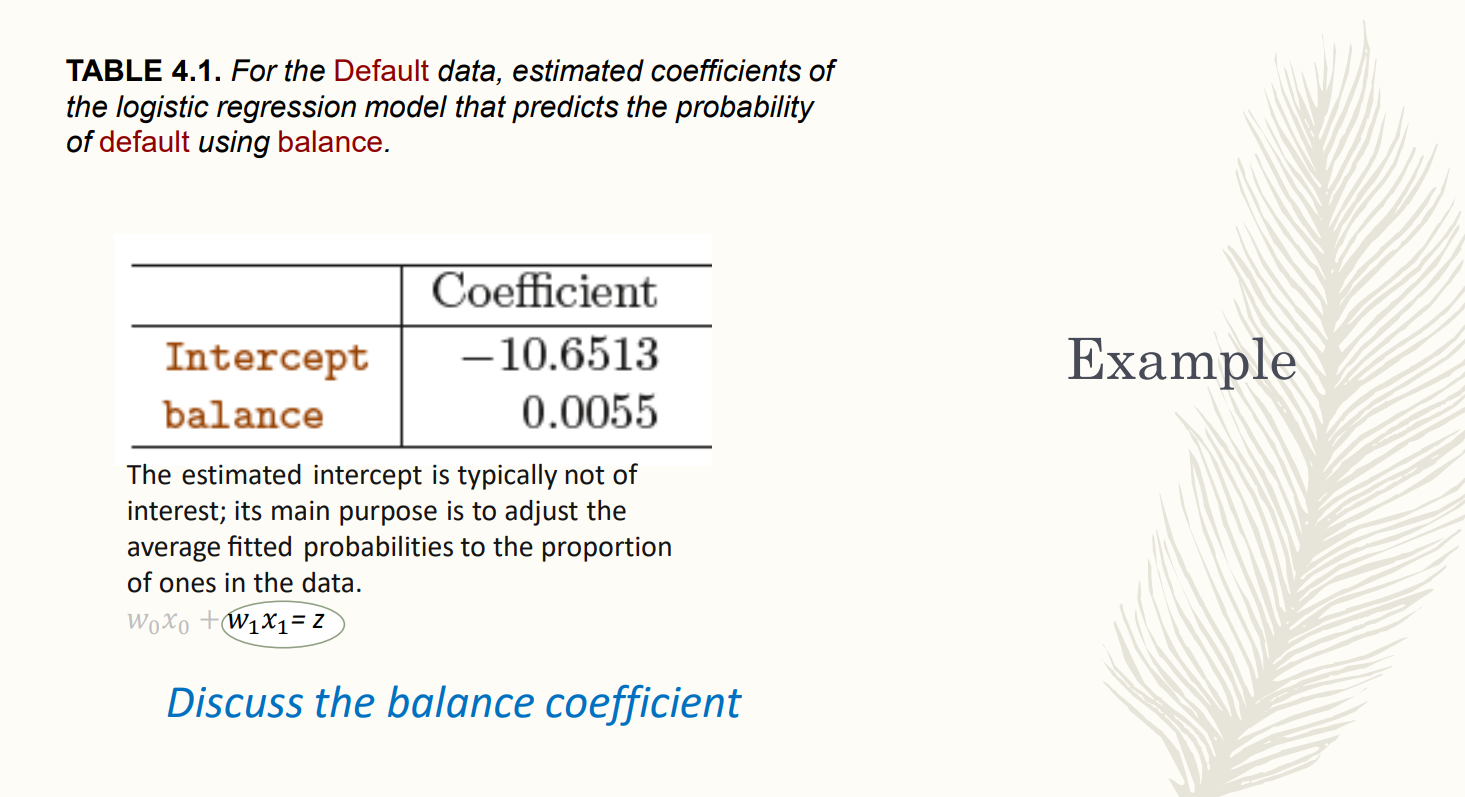
positive weight: the feature helps passing the class.

You can compute the exact probability by using that equation.



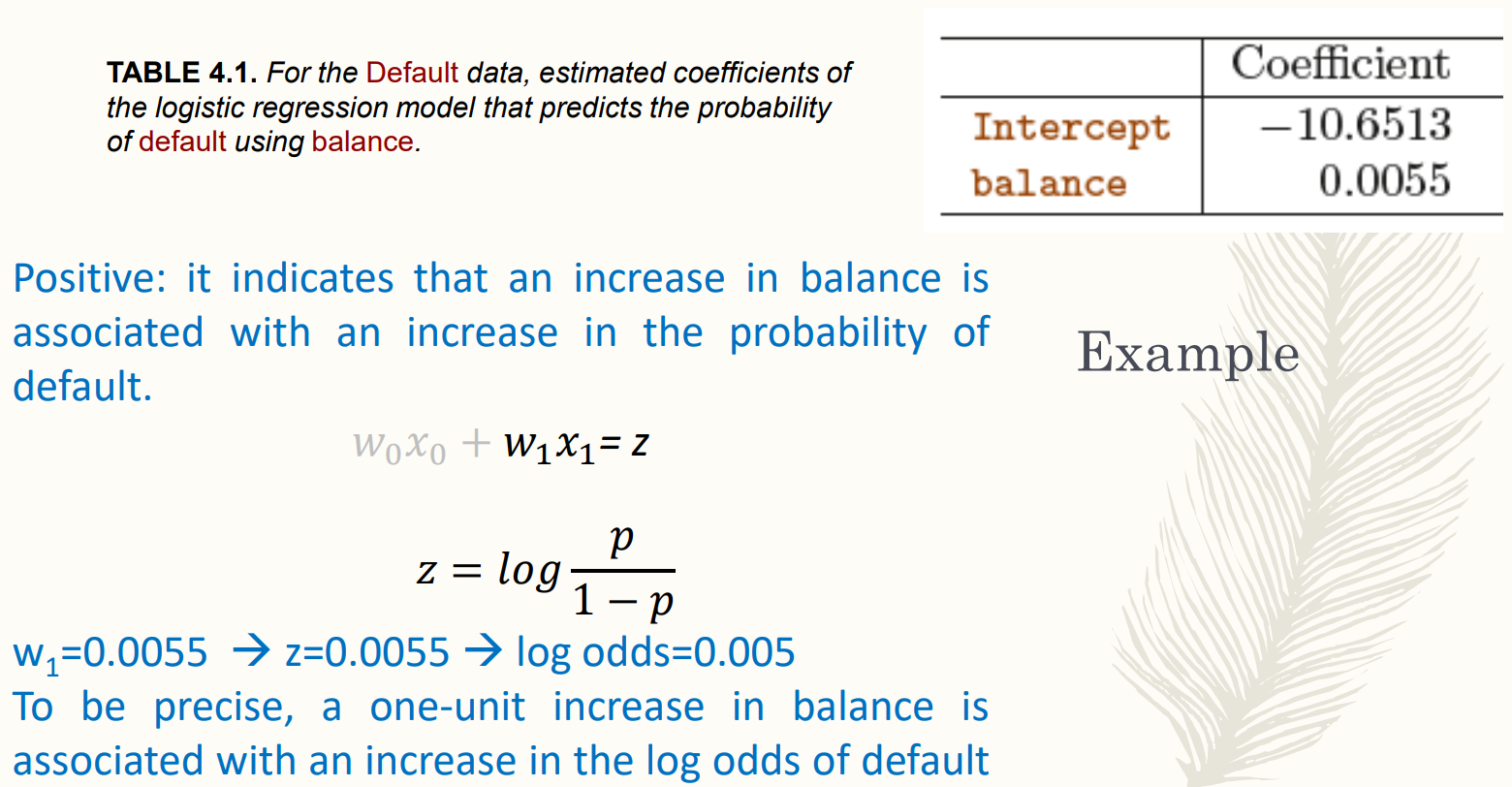


set of individuals for whom you record the average credit card balance, the income and if they are students or not.



increasing the value of the balence of the individual is increasing the value of default. Positive weight of a feature ⟹ the feature works for the positive class.

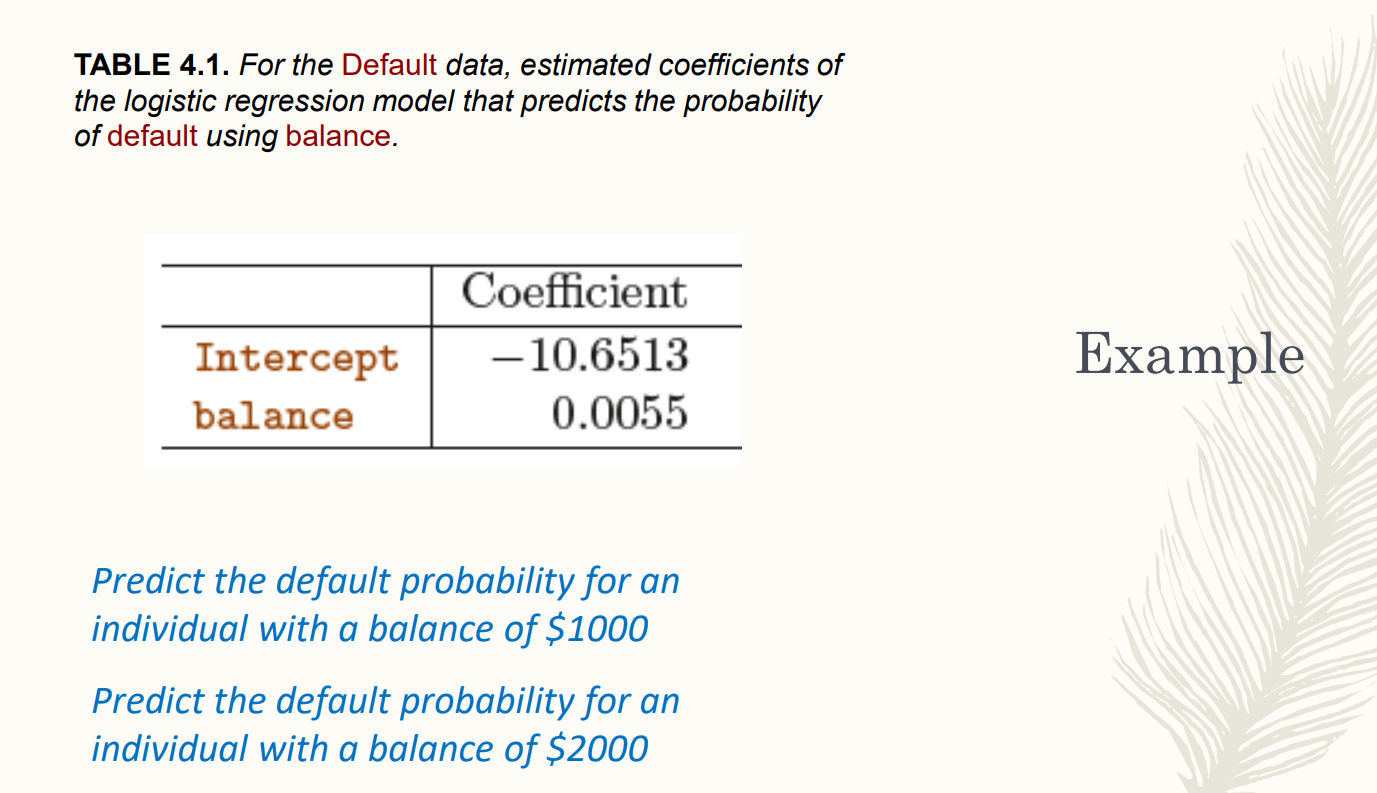
individual with high values on the balance more likely defaulting.

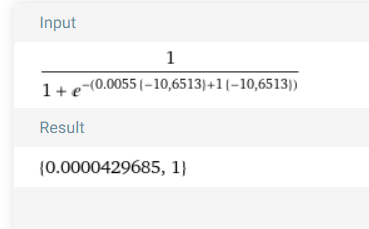


We have a non-linear activation function.

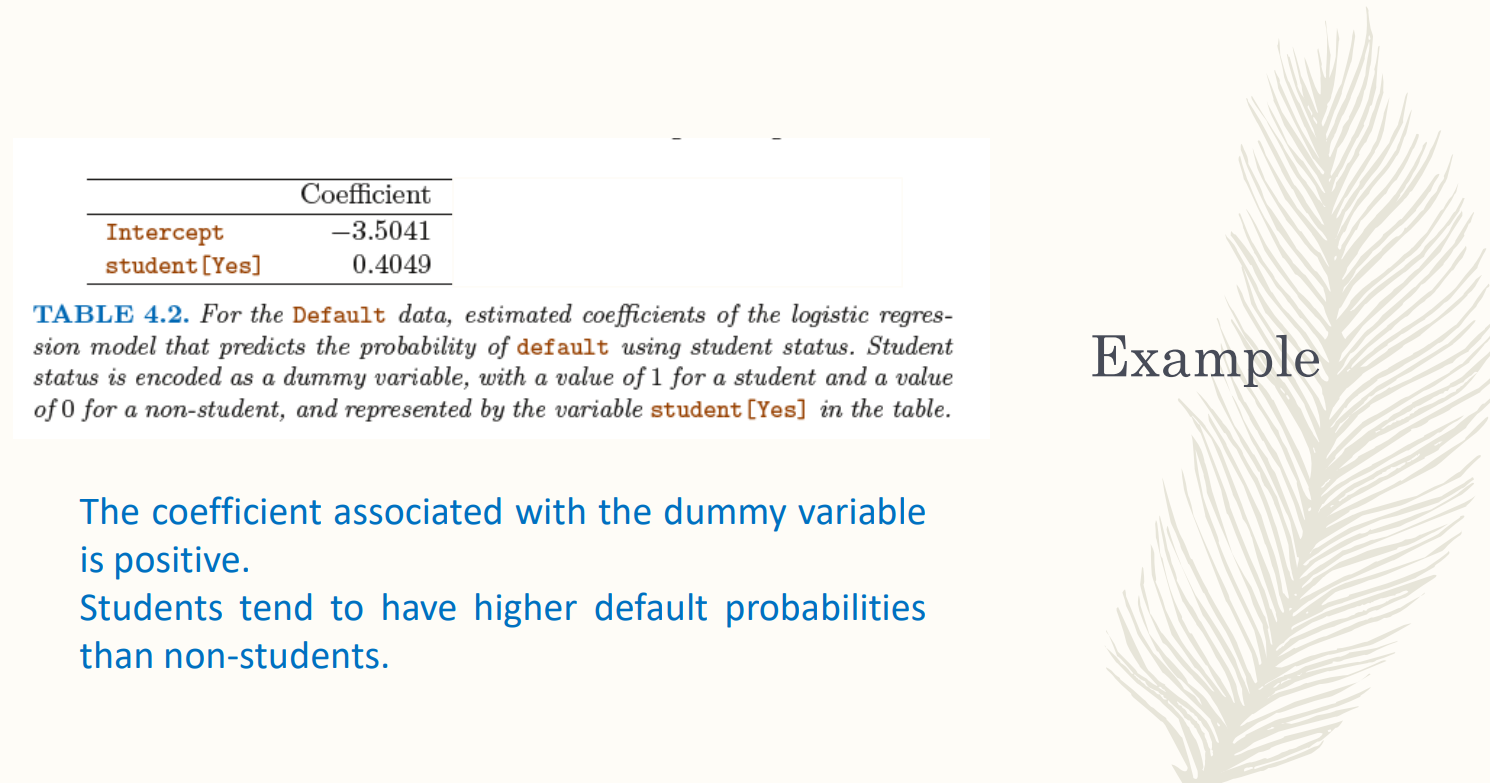
0.0055

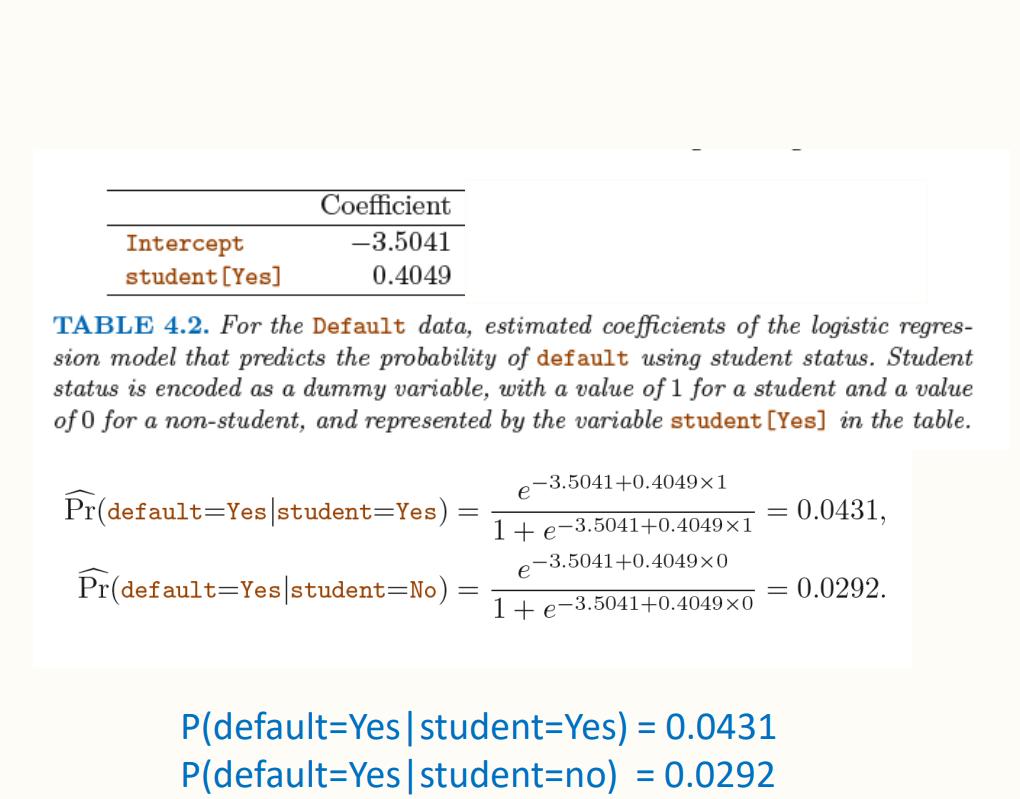
how much given a parameter of a feature and increasing of the value of a feature increase the probability.



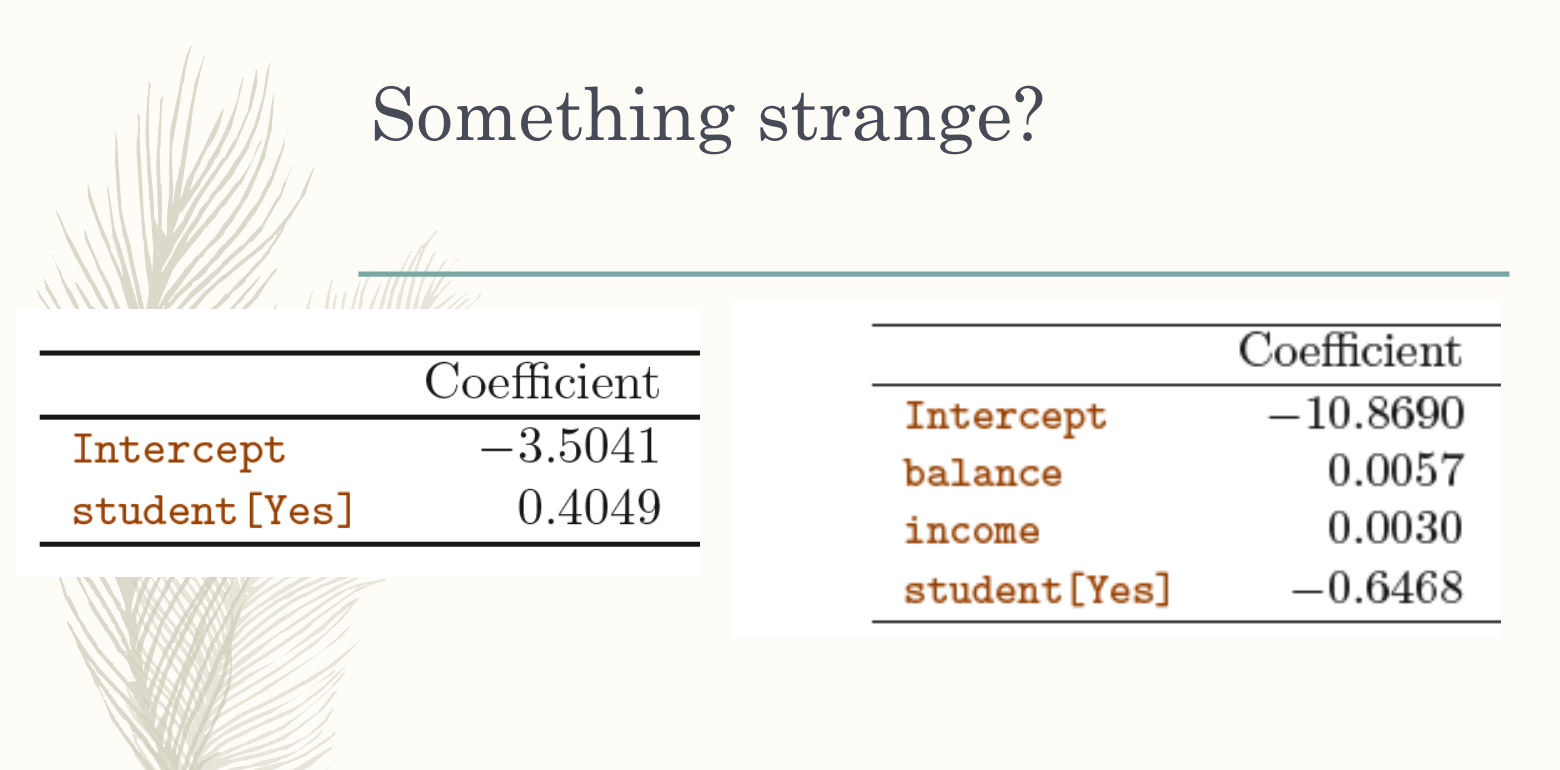


????

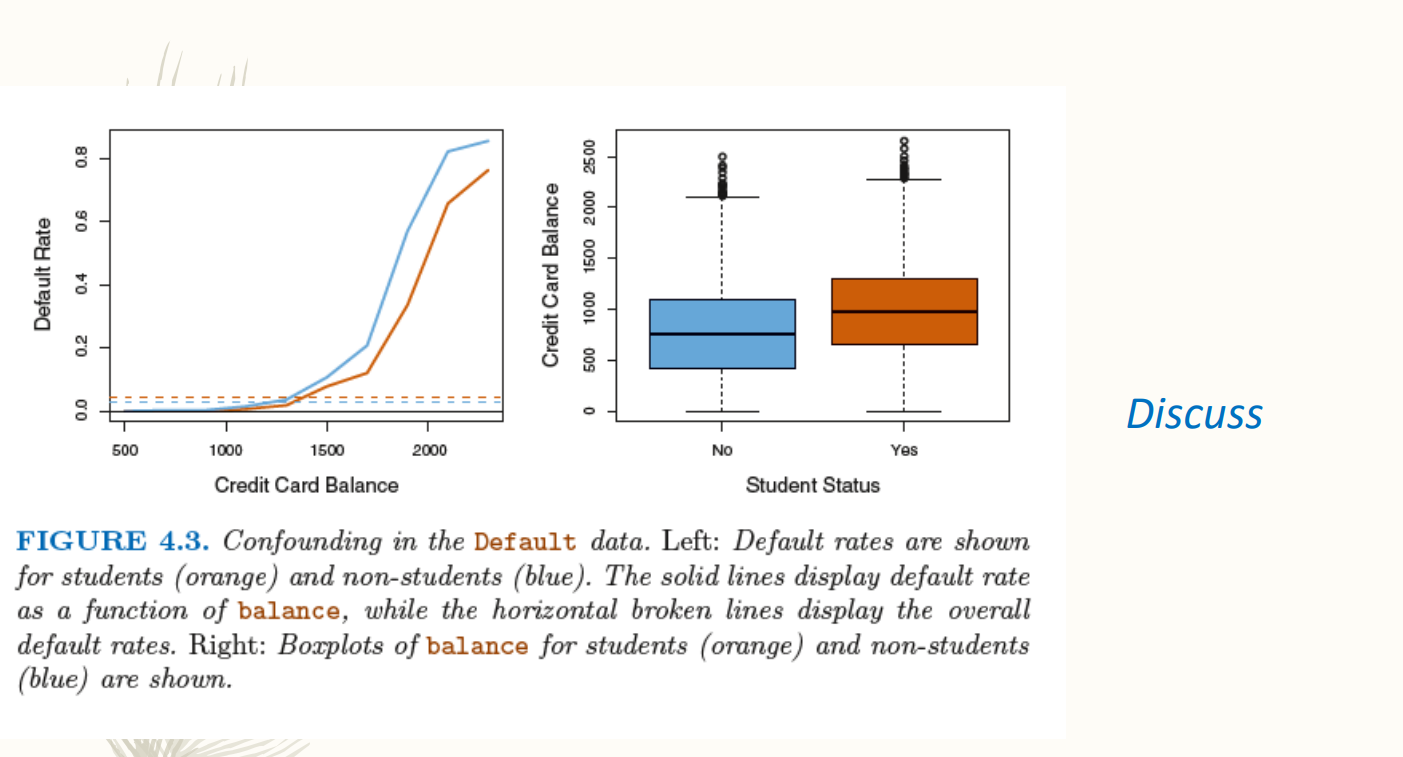




COMPLETARE



with multiple feature we have different models

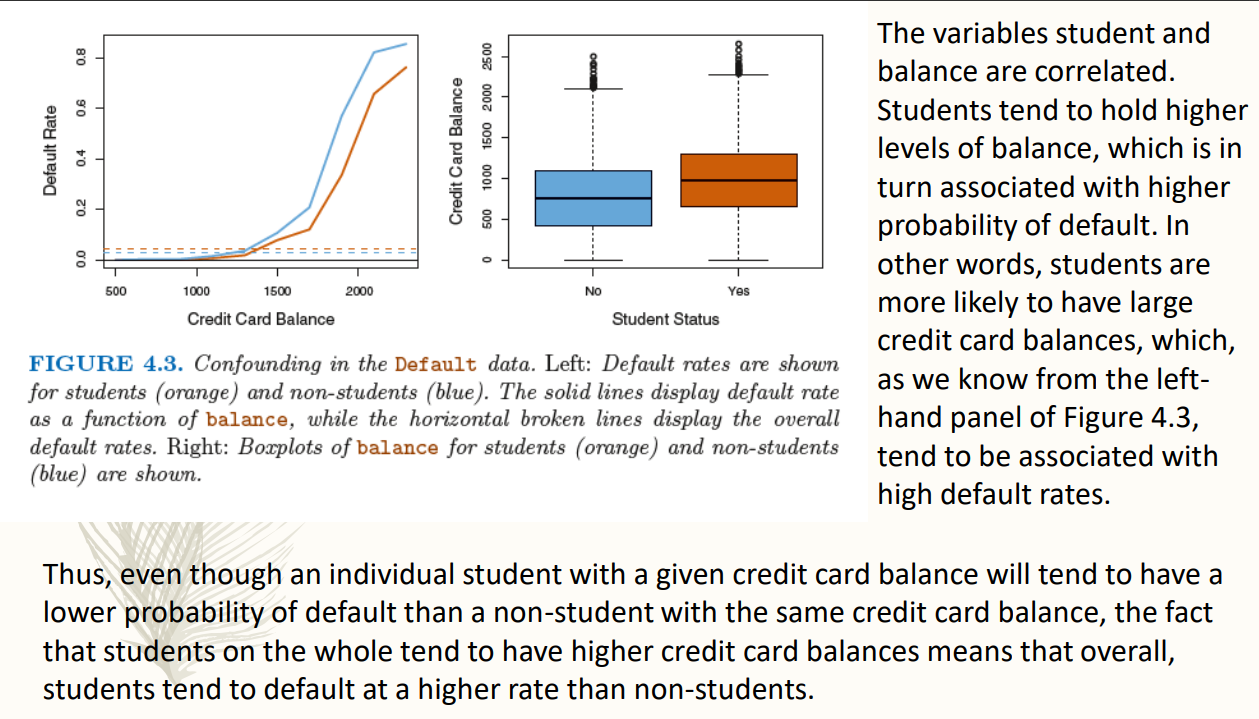


default rate for student and not students

blue line is above the orange line, increasing the credit card balance of not students makes the default rate higher

the credit card balance for student have a mean value higher than not student

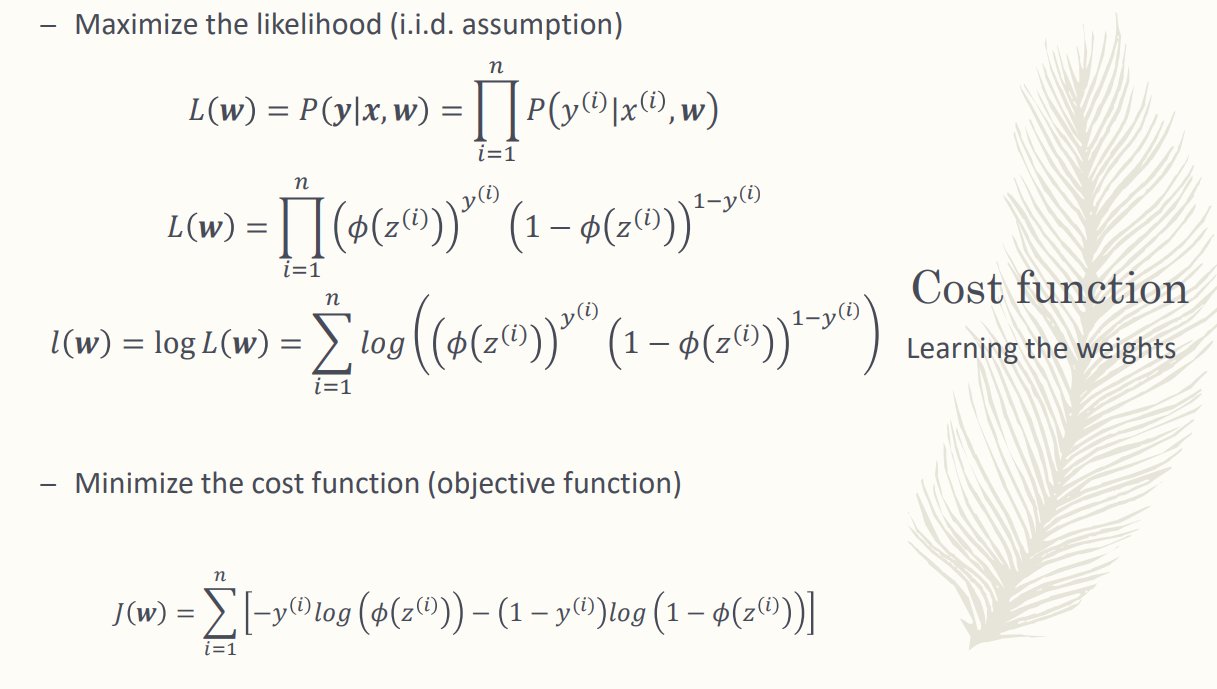
student have a higher credit card balance than not students.



student tends to have higher level of balance, high lebel of balance have high balance of probsability. students seems to be more risky than not students. but the model finds the correct weight and for each single individual is higher if it is a student

if you know the balance and the income of individuals you obtain that students are less risky than not students

on average students have higher balance.



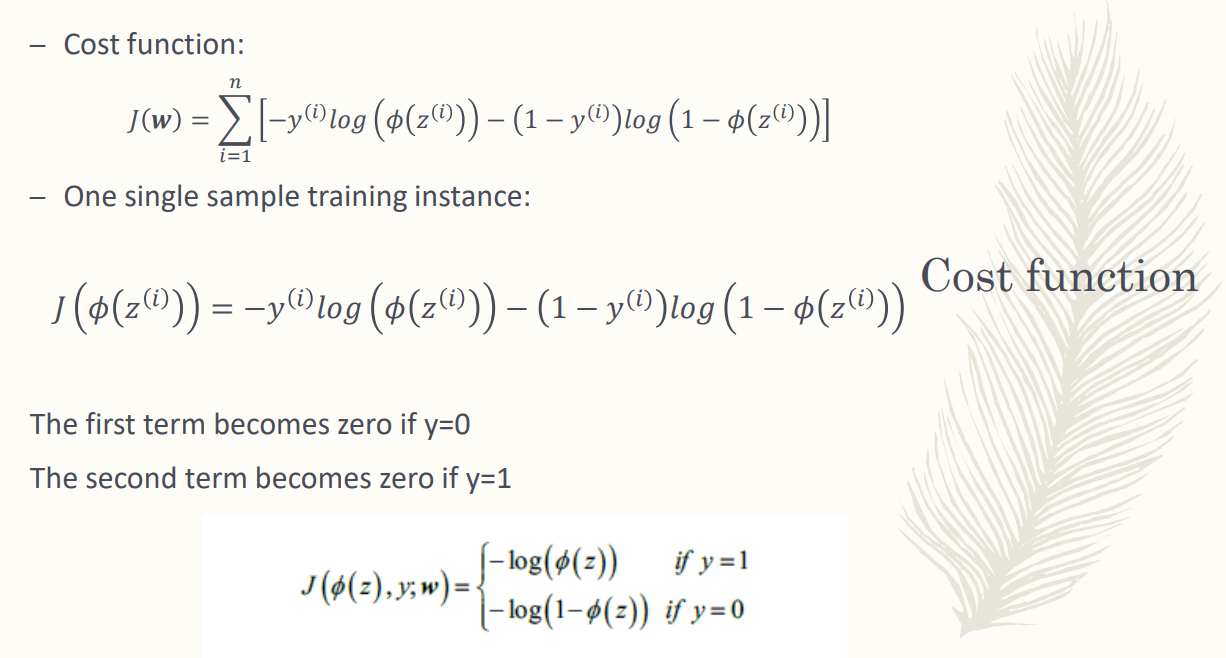
if we assume that all sample are independent and identically distributed

want to maximise the prob observer by your set of +compl

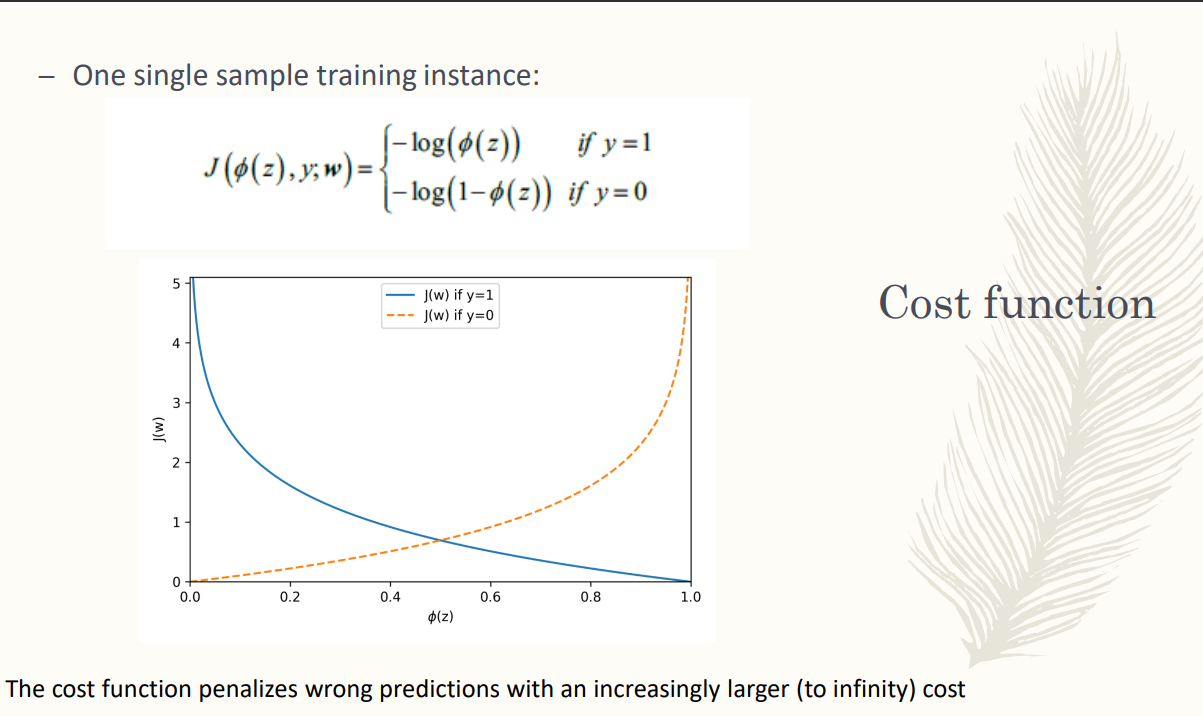
is the product of the probability over the sample.

in logistic regression the prob is given bt the Φ

product not very easy to manage. much better having sum then product.



cost function have only one term



-------------

Logistic regression: one of the most popular and widely used learning algorithms today