COMP47350: Data Analytics (Conv)

Dr. Georgiana Ifrim
georgiana.ifrim@ucd.ie

Insight Centre for Data Analytics
School of Computer Science
University College Dublin

2018/19

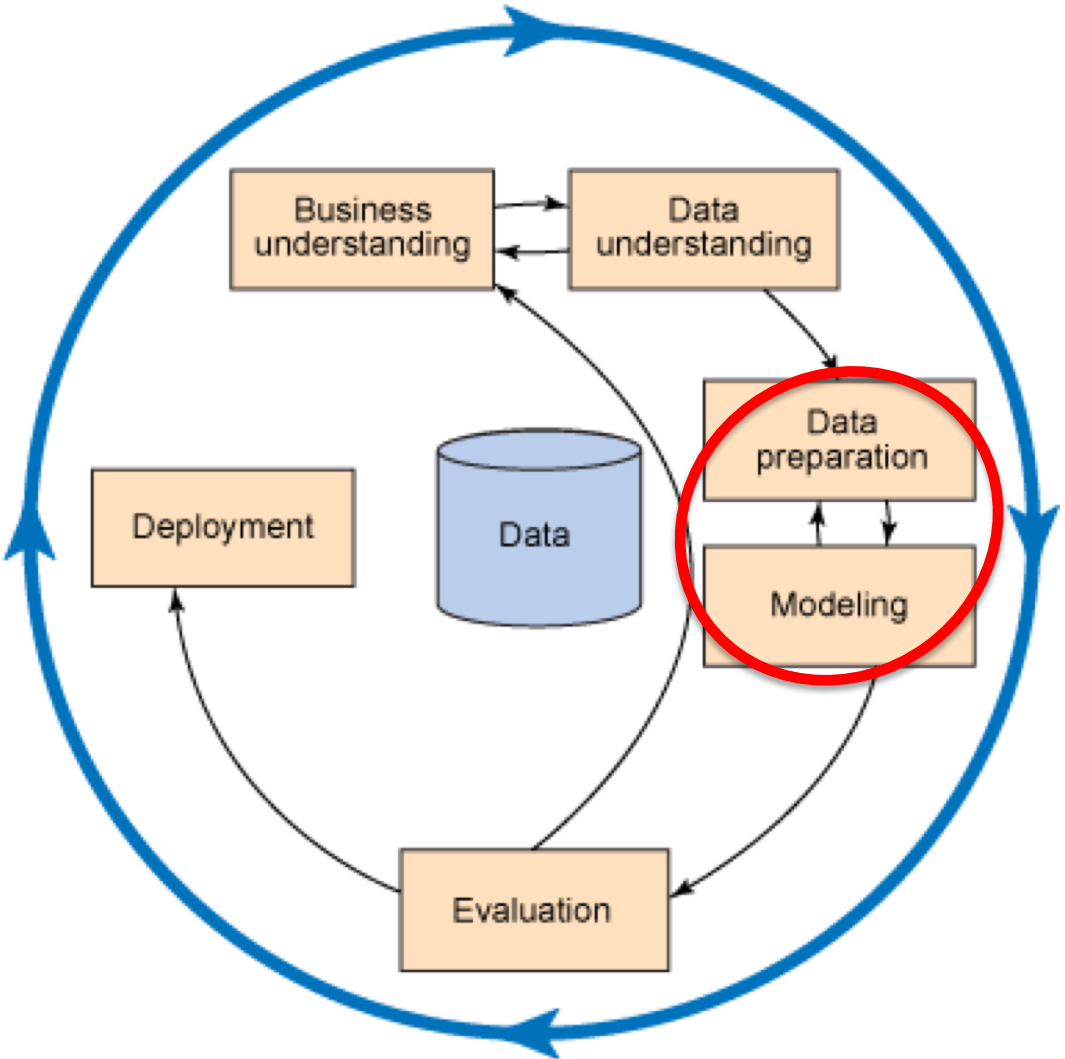
Module Topics

- **Python Environment** (Anaconda, Jupyter Notebook)
- **Getting Data** (Web scrapping, APIs, DBs)
- **Understanding Data** (slicing, visualisation)
- **Preparing Data** (cleaning, transformation)
- **Modeling & Evaluation** (machine learning)

Data Analytics Project Lifecycle:

CRISP-DM

CRISP-DM: **CR**oss-**I**ndustry **S**tandard **P**rocess for **D**ata **M**ining



Modeling Data

- Modeling:
  - How to build prediction models
  - How to evaluate prediction models

Learning Tasks (e.g., types of ML problems):

- **Regression:** predicting a numeric target feature
- **Classification:** predicting a categorical target feature

Today's Lecture

Regression Algorithm:

Linear Regression for predicting a **numeric** target feature; **looks for linear relationship between descriptive and target feature.** Could in principle be used for classification, but we will see pitfalls of this approach.

Classification Algorithm:

Logistic Regression for predicting a **categorical** target feature; **looks for linear separation between classes.**

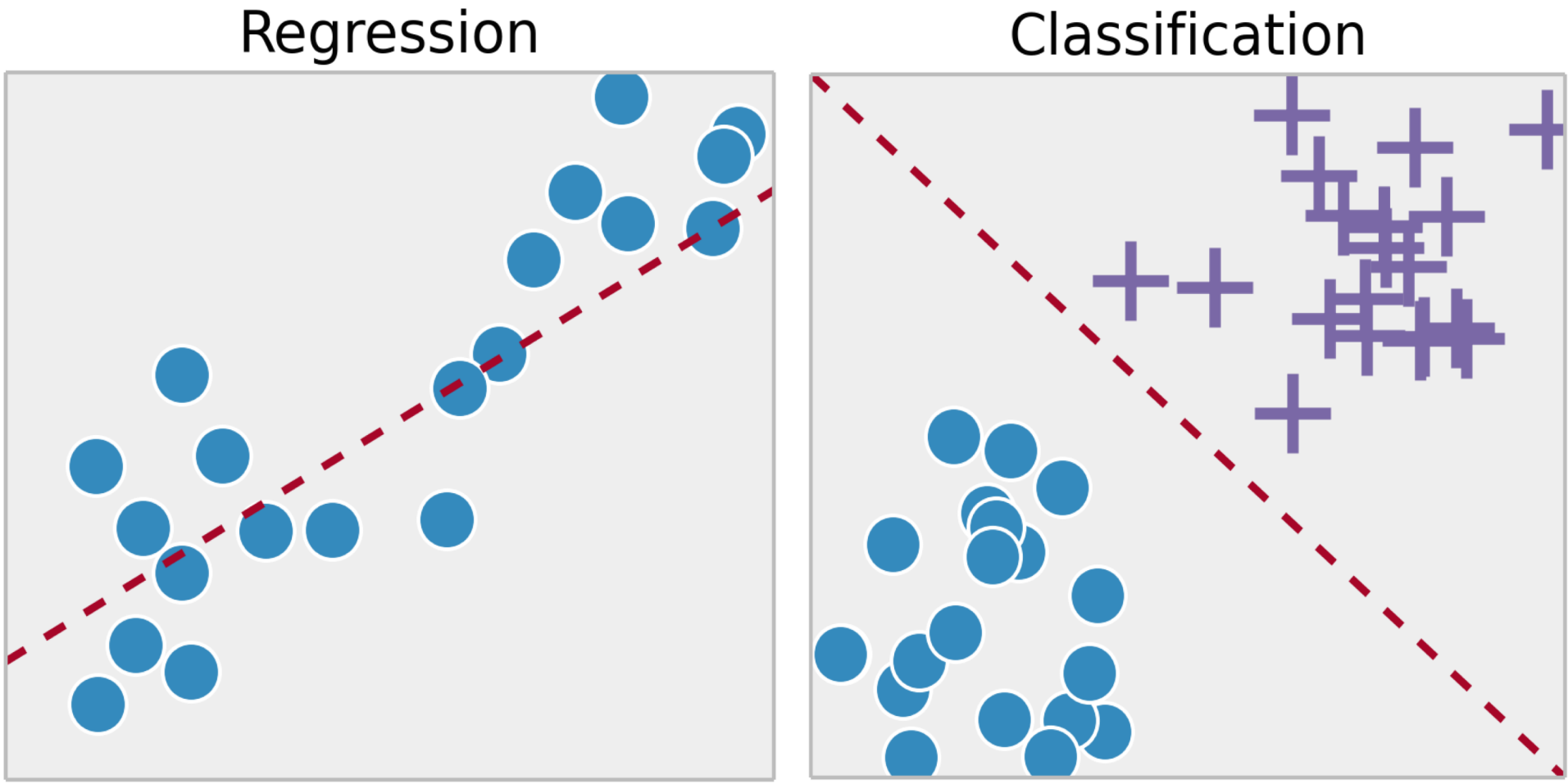
Regression vs Classification

Regression: Predict the RentalPrice given the Size of an office

Classification: Predict if the RentalPrice is High or Low given the Size of the office (the focus is on predicting the **Probability(RentalPrice=High| Size)**)

- We typically work with two classes and code one class with 0 and the other class with 1 (or with -1 and 1).
- If there are more than 2 classes, we can use the one-vs-all formulation to create **C** binary classification problems (where **C** is the number of classes).

Regression vs Classification



Linear Regression

Estimates a **linear relationship** between the descriptive features and the target feature (weights: w_0, w_1, …,w_n)

target_feature = w_0 + w_1 *feature_1 + w_2*feature_2 + + … + w_n*feature_n

Regression: Example

Table: The **office rentals dataset**: a dataset that includes office rental prices and a number of descriptive features for 10 Dublin city-centre offices.

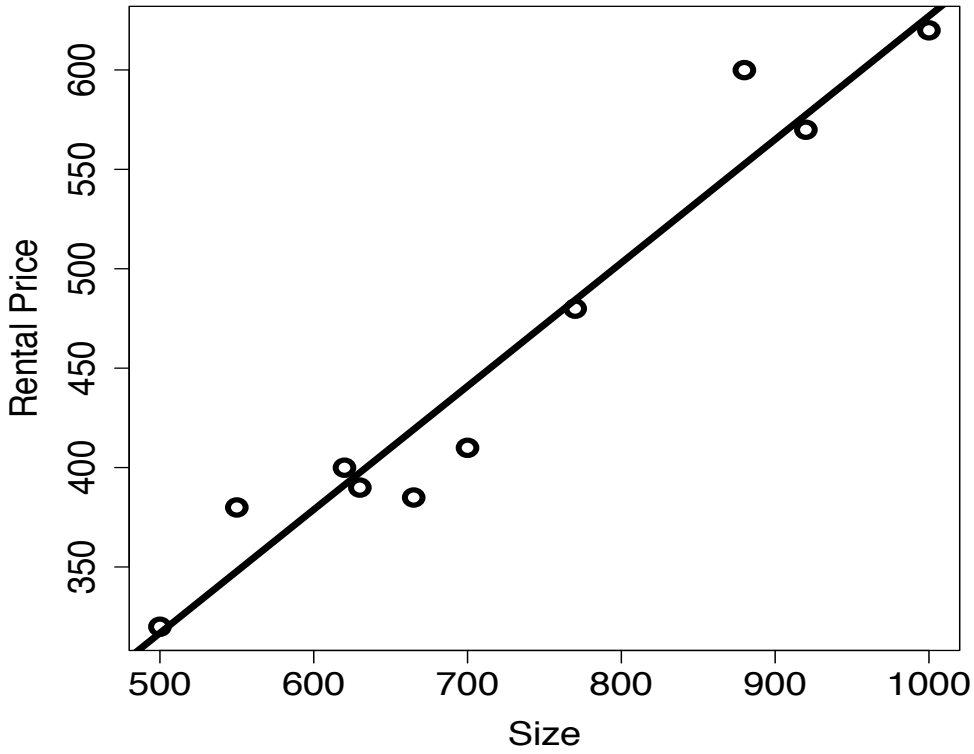
ID	Size	Floor	Broadband Rate	Energy Rating	Rental Price
1	500	4	8	C	320
2	550	7	50	A	380
3	620	9	7	A	400
4	630	5	24	B	390
5	665	8	100	C	385
6	700	4	8	B	410
7	770	10	7	B	480
8	880	12	50	A	600
9	920	14	8	C	570
10	1,000	9	24	B	620

Can we predict the **rental price**, given the descriptive features (**size, floor, broadband rate, energy rating**) for an office?

Simple Linear Regression (1 feature)

ID	Size	Rental Price
1	500	320
2	550	380
3	620	400
4	630	390
5	665	385
6	700	410
7	770	480
8	880	600
9	920	570
10	1,000	620

$$\text{Rental Price} = 6.47 + 0.62 \times \text{Size}$$

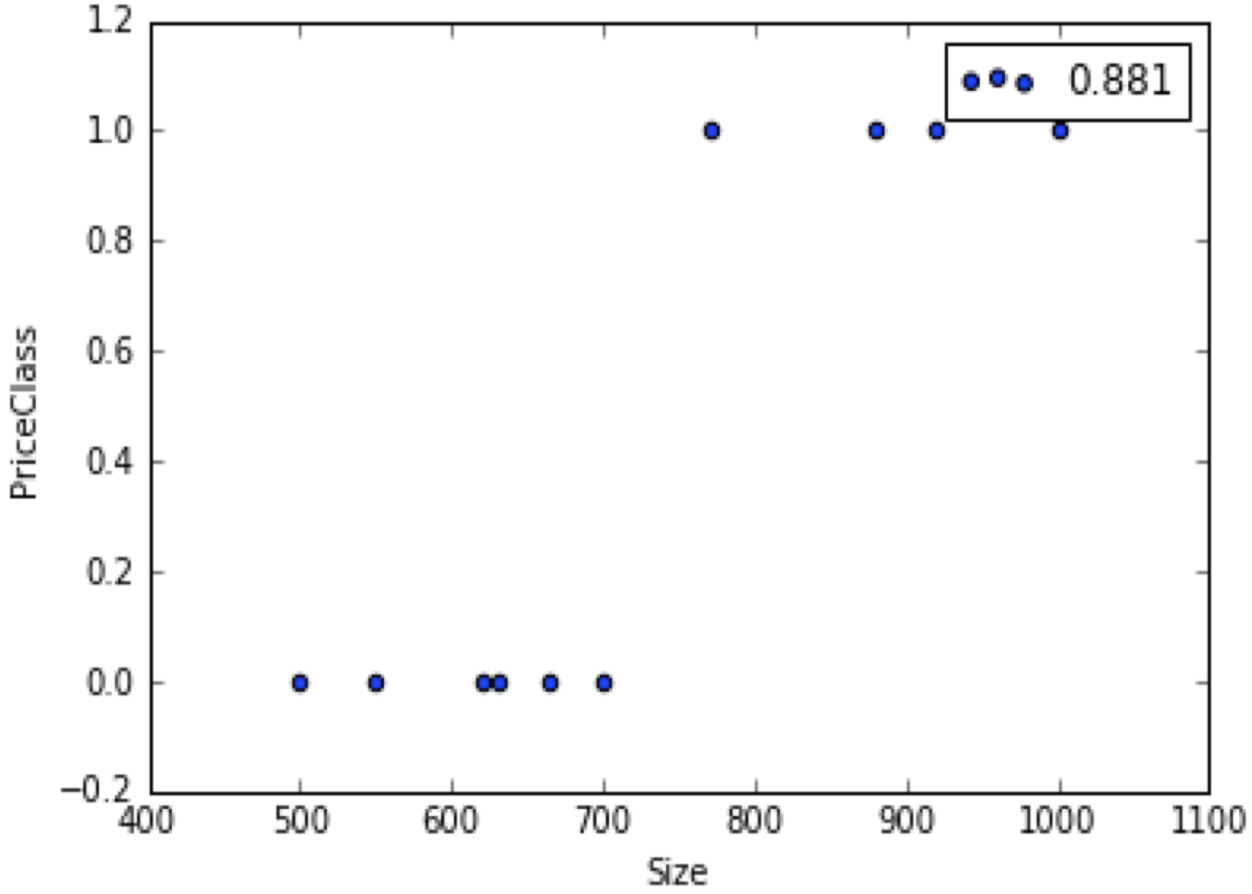


Classification: Example

	ID	Size	Floor	BroadbandRate	EnergyRating	PriceClass
0	1	500	4	8	C	0.0
1	2	550	7	50	A	0.0
2	3	620	9	7	A	0.0
3	4	630	5	24	B	0.0
4	5	665	8	100	C	0.0
5	6	700	4	8	B	0.0
6	7	770	10	7	B	1.0
7	8	880	12	50	A	1.0
8	9	920	14	8	C	1.0
9	10	1000	9	24	B	1.0

Can we predict the probability of the **price class being 1**, given the descriptive features (**size, floor, broadband rate, energy rating**) for an office? We code with 0 for PriceClass Low and 1 for PriceClass High.

PriceClass given Size



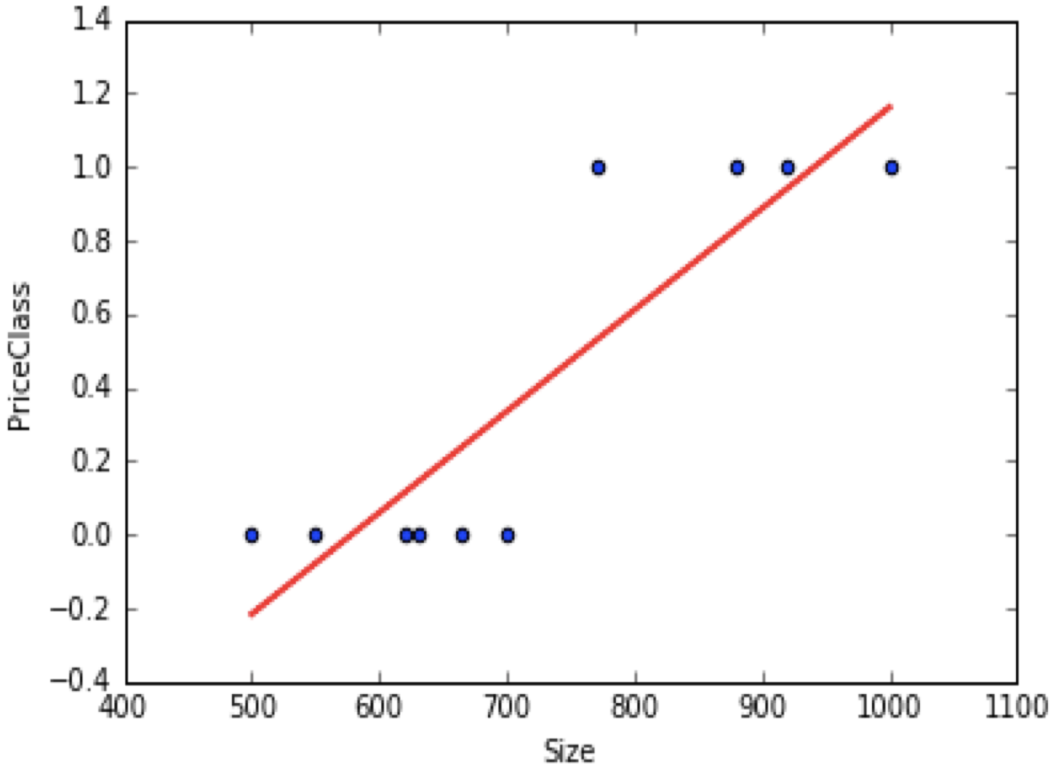
There seems to be a threshold on the values of the Size (at about 700) that separates the offices with Low Price (class 0) from High Price (class 1)

PriceClass given Size

Using linear regression directly for classification (target can be interpreted as a probability P(PriceClass=High|Size)):

$$PriceClass = -1.59 + 0.0027 * Size$$

- The regression line in the plot under-estimates or over-estimates the PriceClass. It also produces negative predictions < 0 or positive predictions > 1.
- We can use a threshold on the prediction, to get a class label: If predicted_PriceClass > 0.5 then PriceClass = High, else PriceClass= Low.



Logistic Regression

- Learning algorithm designed specifically for the classification task
- We assume only 2 classes (binary classification); can extend to many classes with one-vs-all approach

- We model the probability of class membership, e.g., if **p(PriceClass = High| Size) > 0.5**, then predict class **High**,
  else predict class **Low**
- We use a logistic function to make sure the predictions are in the [0,1] interval (proper probabilities)

- **Linear regression:**
  p(PriceClass=High| Size)**= w_0 + w_1** * Size

- **Logistic regression:**
  p(PriceClass=High| Size)**= logistic(w_0 + w_1** * Size**)**

Logistic Regression

Logistic function: takes in any real-number and outputs a number between [0, 1]

logistic function

$$logistic(x) = \frac{e^x}{1+e^x} = \frac{1}{1+e^{-x}}$$

$$Logistic(x) = \frac{1}{1 + e^{-x}} \tag{8}$$

where x is a numeric value and e is **Euler's number** and is approximately equal to 2.7183.

Logistic Regression

When performing linear regression, we use the following function:

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

When performing logistic regression, we use the following form:

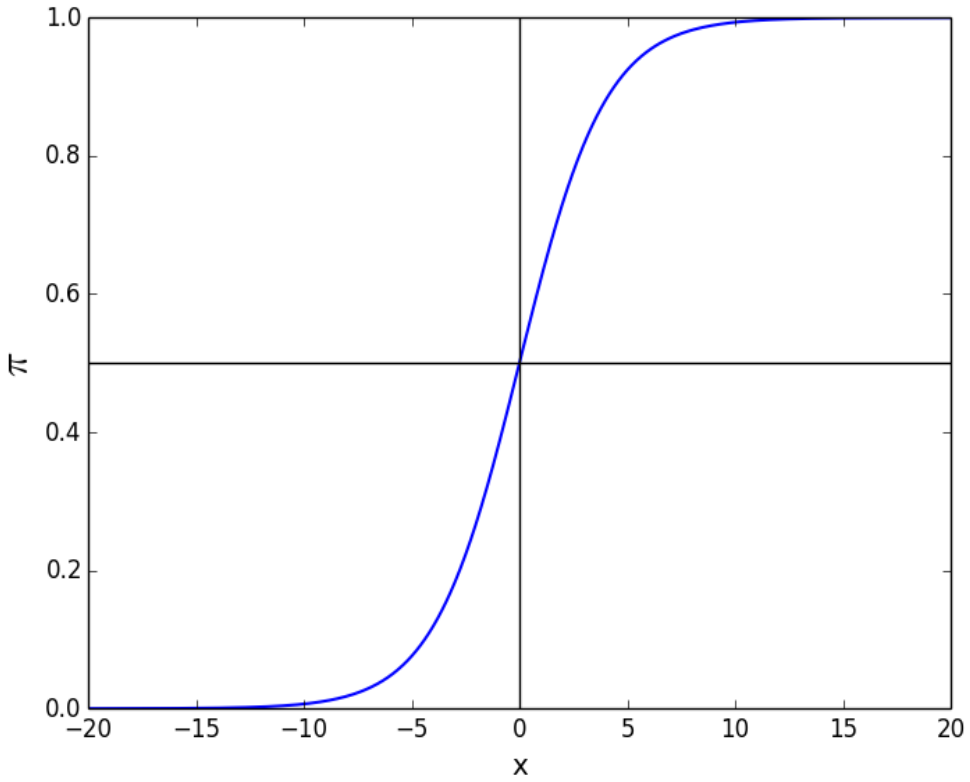
$$\pi = \Pr(y = 1 \mid x) = \frac{e^{\beta_0 + \beta_1 x}}{1 + e^{\beta_0 + \beta_1 x}}$$

Probability of y = 1, given x

Logistic Regression

The logistic function takes on an "S" shape, where y is bounded by [0, 1]

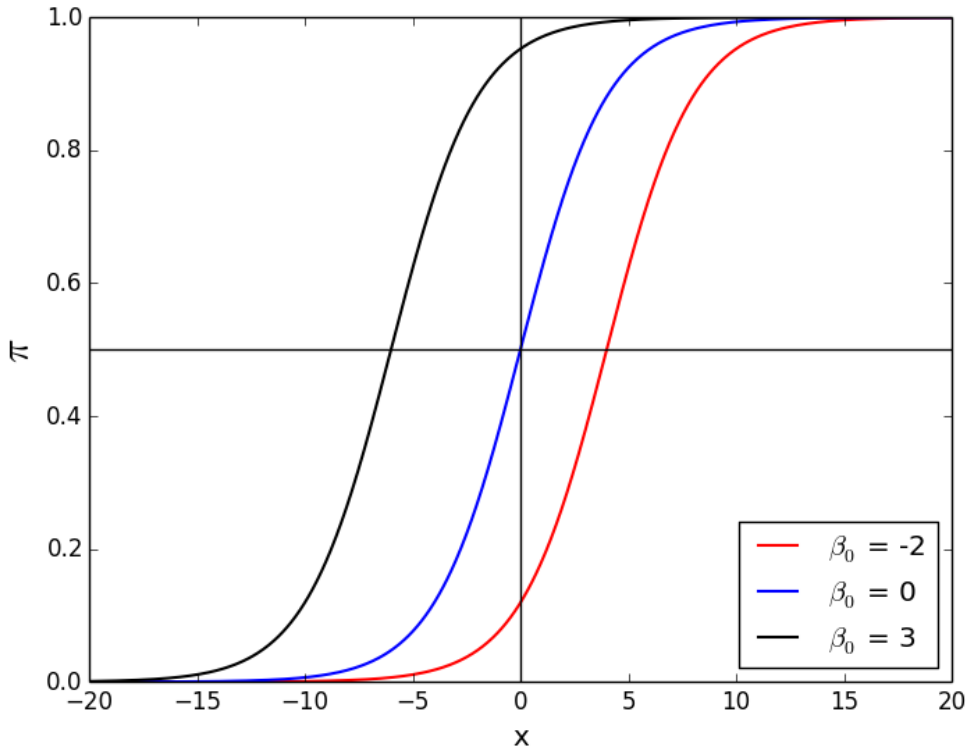
$$\pi = \frac{e^{\beta_0 + \beta_1 x}}{1 + e^{\beta_0 + \beta_1 x}}$$



Logistic Regression

Changing the β_0 value shifts the function horizontally.

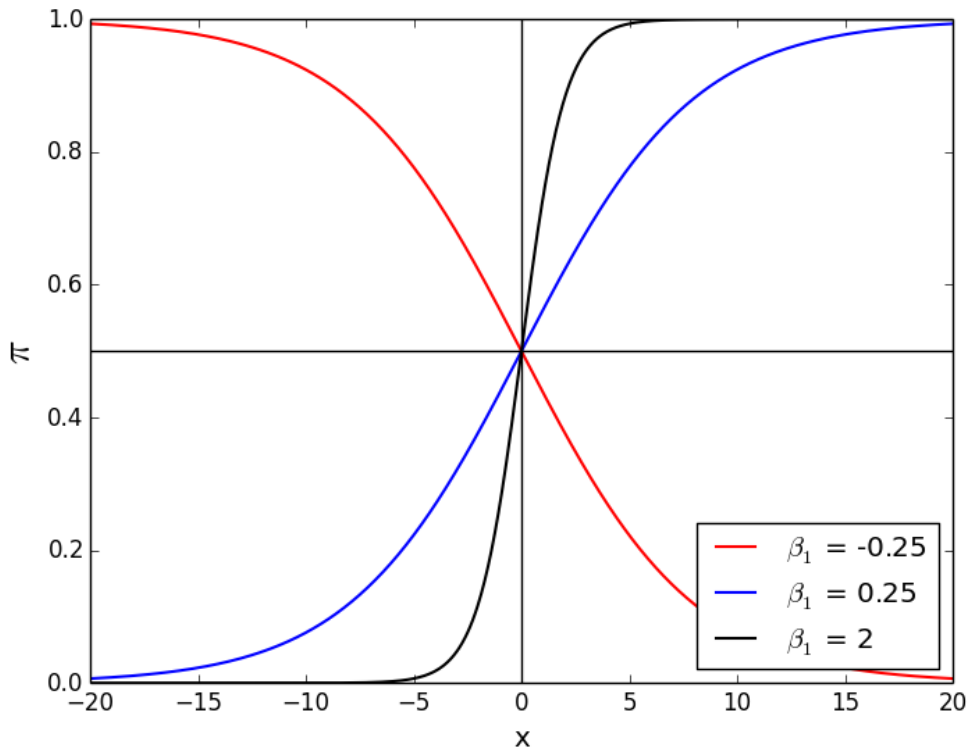
$$\pi = \frac{e^{\beta_0 + \beta_1 x}}{1 + e^{\beta_0 + \beta_1 x}}$$



Logistic Regression

Changing the β_1 value changes the slope of the curve

$$\pi = \frac{e^{\beta_0 + \beta_1 x}}{1 + e^{\beta_0 + \beta_1 x}}$$



Logistic Regression

In order to interpret the outputs of a logistic function we must understand the difference between probability and odds.

The odds of an event are given by the ratio of the probability of the event by its complement:

$$Odds = \frac{\pi}{1-\pi}$$

Logistic Regression

- You're trying to determine whether a customer will convert or not. The customer conversion rate is 33.33%. What are the odds that a customer will convert?

$$Odds = \frac{\pi}{1-\pi} = \frac{.3333}{.6666} = \frac{1}{2}$$

NOTE

This means that for every customer that converts you will have two customers that do not convert

Logistic Regression

- What would happen if we took the odds of the logistic function?

$$\frac{\pi}{1-\pi} = \frac{e^{\beta_0+\beta_1 x}/(1+e^{\beta_0+\beta_1 x})}{1-e^{\beta_0+\beta_1 x}/(1+e^{\beta_0+\beta_1 x})}$$

$$= \frac{e^{\beta_0+\beta_1 x}/(1+e^{\beta_0+\beta_1 x})}{(1+e^{\beta_0+\beta_1 x})/(1+e^{\beta_0+\beta_1 x}) - e^{\beta_0+\beta_1 x}/(1+e^{\beta_0+\beta_1 x})} = e^{\beta_0+\beta_1 x}$$

Logistic Regression

- If we take the logarithm of the odds, we return a linear equation

$$\log\left(\frac{\pi}{1-\pi}\right) = \log\left(e^{\beta_0+\beta_1 x}\right) = \beta_0 + \beta_1 x$$

- The above is known as the logit transformation or the log-odds function

Interpretation

- In linear regression, the parameter β_1 represents the change in the **target feature** for a unit change in x.
- In logistic regression, parameter β_1 represents the change in the **log-odds** for a unit change in x.

*This means that e^{β_1} gives us the change in the **odds** for a unit change in x.*

- A tiny change in β_1 leads to a big change in the odds

Logistic Regression: Multiple Features

Once we understand the basic form for logistic regression, we can easily extend the definition to include multiple input values.

$$\log\left(\frac{\pi}{1-\pi}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p$$

Logistic function

$$\pi = \frac{e^{\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p}}{1 + e^{\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p}}$$

Probability(class=1|example x described by features feature_1…feature_n) =
logistic(w_0 + w_1 *feature_1 + w_2*feature_2 + … + w_n*feature_n)

Logistic Regression

Tools that we used for linear regression can be used for logistic regression (which is also a linear model):

- Categorical descriptive features need to be turned to continuous (binary encoding or integer encoding)
- Non-linear relationships can be captured by creating new features (e.g., $Size^2$ or Size * Floor)

References

- Chapter4 from **An Introduction to Statistical Learning**, by G. James, D. Witten, T. Hastie, R. Tibshirani, Springer, 2016 (free book: http://www-bcf.usc.edu/~gareth/ISL/)

- Web: https://github.com/justmarkham/DAT4/blob/master/slides/09_logistic_regression.pdf