MTU
Ollscoil Teicneolaíochta na Mumhan
Munster Technological University

Practical Machine Learning

Practical Machine Learning

Lecture: Introduction to Machine Learning (Part 2)

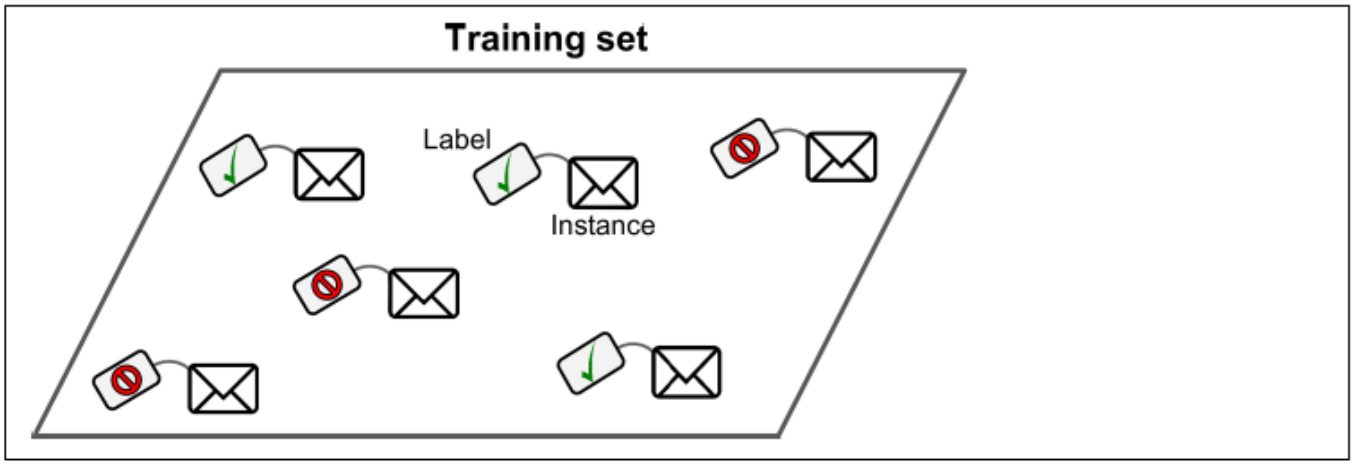
Ted Scully

Categories of Machine Learning Algorithms

- Machine learning algorithms can be divided into five main categories
  - Supervised Learning Algorithms
  - Unsupervised Learning Algorithms
  - Semi Supervised Learning Algorithms
  - Self Supervised Machine Learning Algorithms
  - Reinforcement Learning Algorithms

Supervised Learning Algorithms

- Supervised learning algorithms used **labelled training data** to train algorithms.
- In other words, the training data you feed to the algorithm includes the desired solutions, called labels.

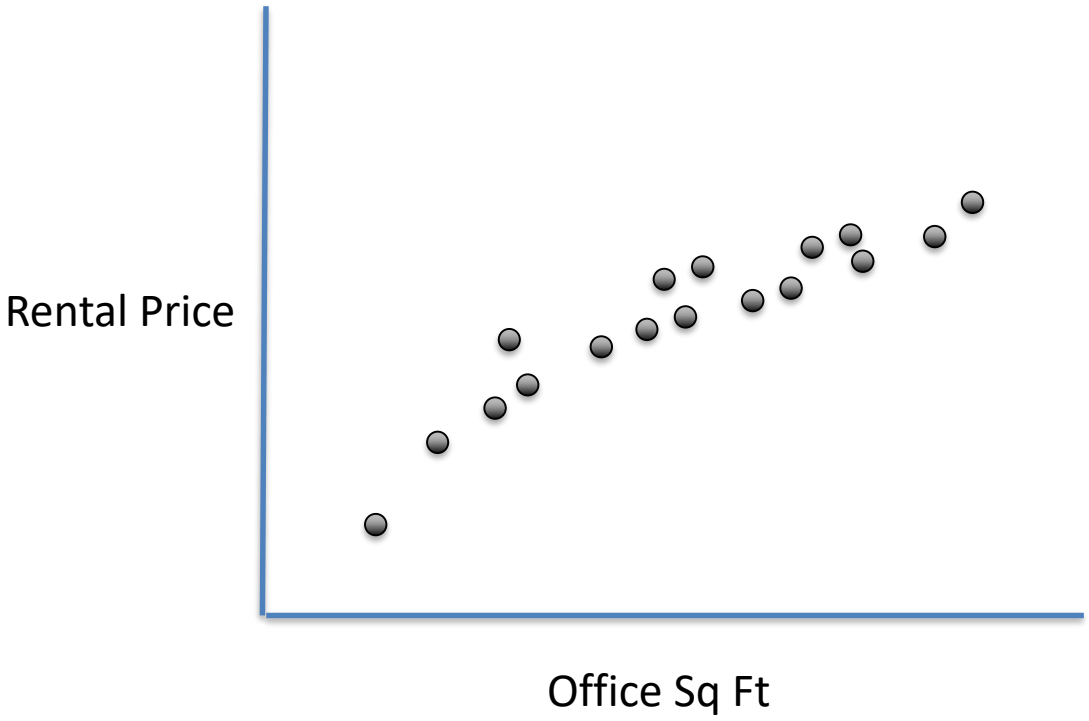


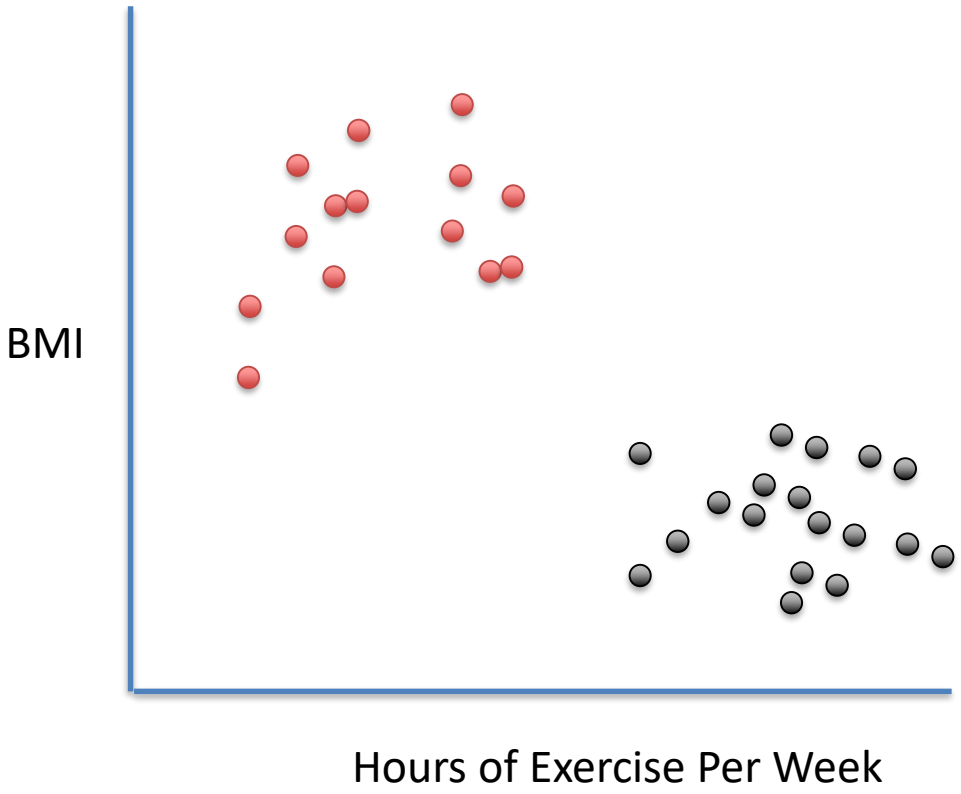
Supervised Learning Algorithms

- Supervised learning can be subdivided into either classification or regression algorithms.

- **Classification** algorithms are used to predict **discrete values** (for example Spam/Ham, Male/Female, Malignant/Benign.

- **Regression** algorithms are used to predict **continuous values** (for example, the price of a house, the concentration of a drug based on a chemical analysis or predict a person's lifespan based on feature information about their health and lifestyle).

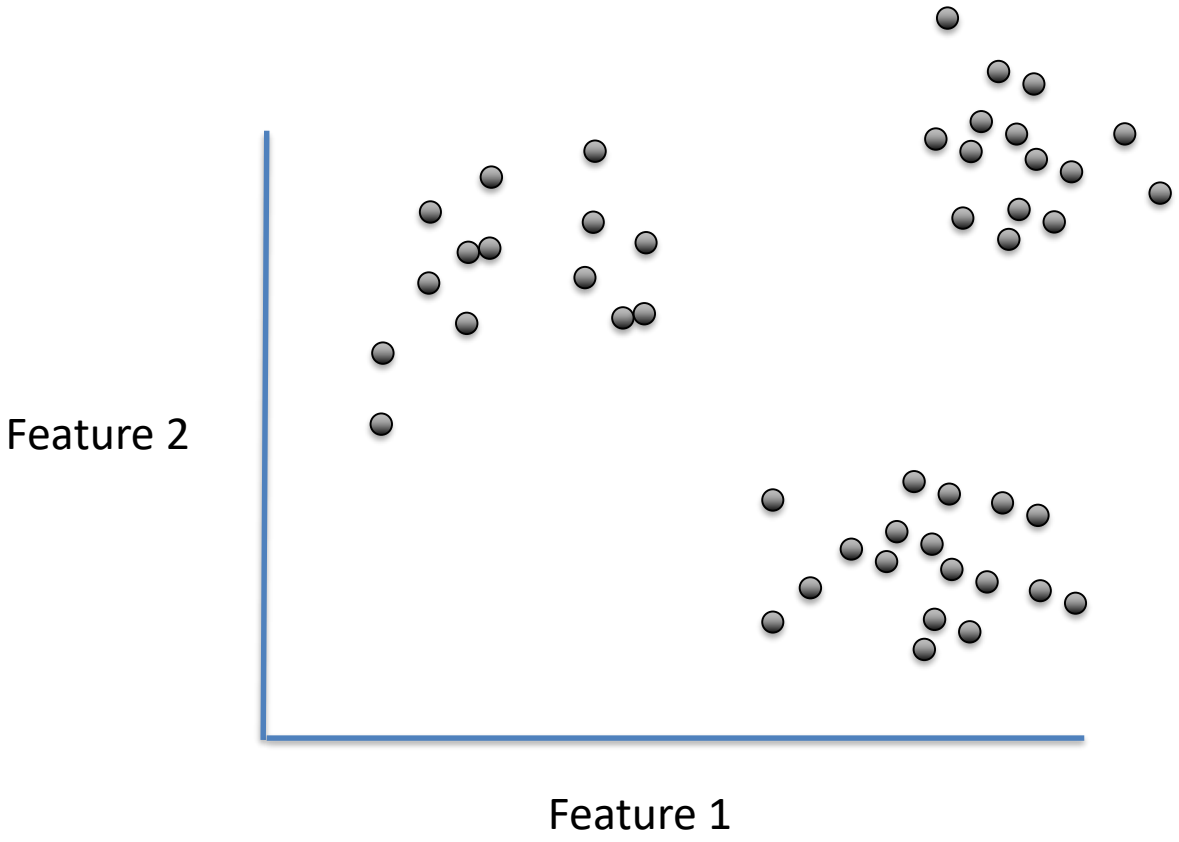

Rental Price
Office Sq Ft

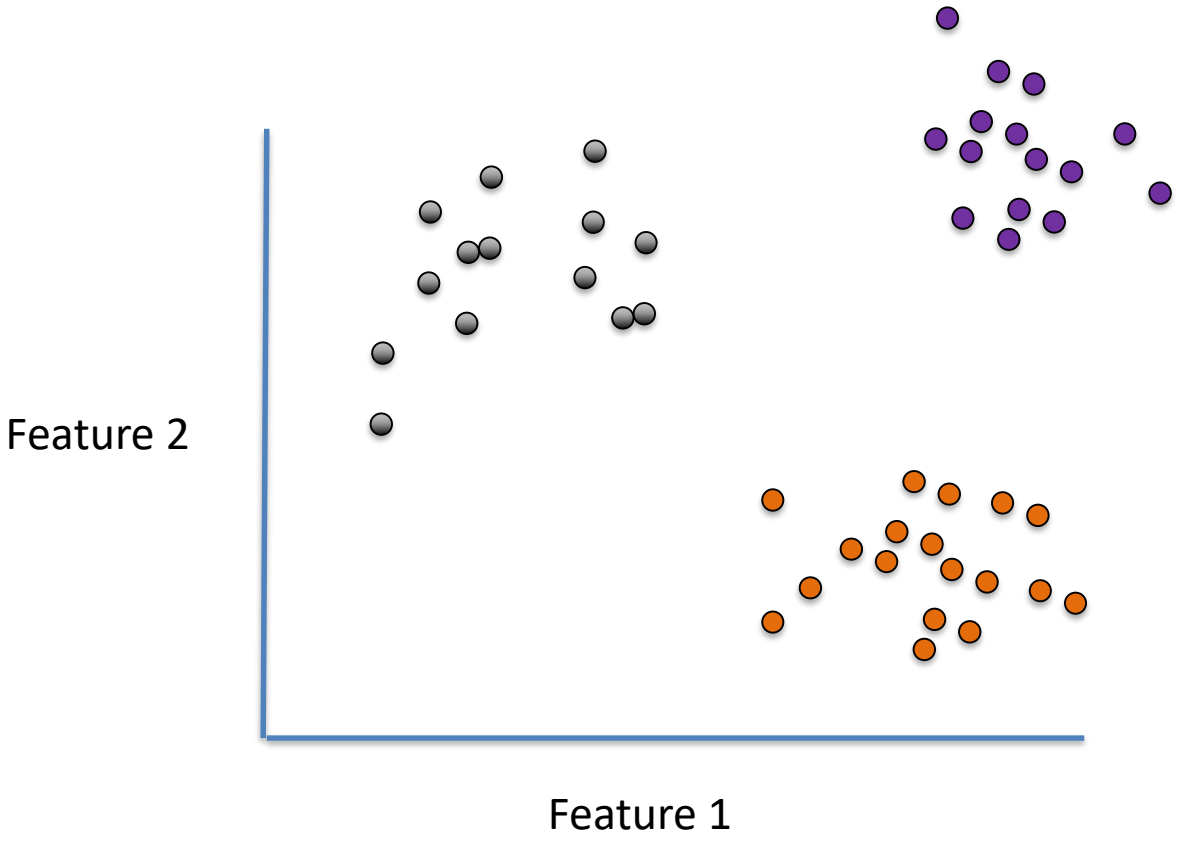

BMI
Hours of Exercise Per Week

Unsupervised Learning Algorithms

- In unsupervised learning the algorithm is not provided with any labelled training data and **must learn patterns from the data**.

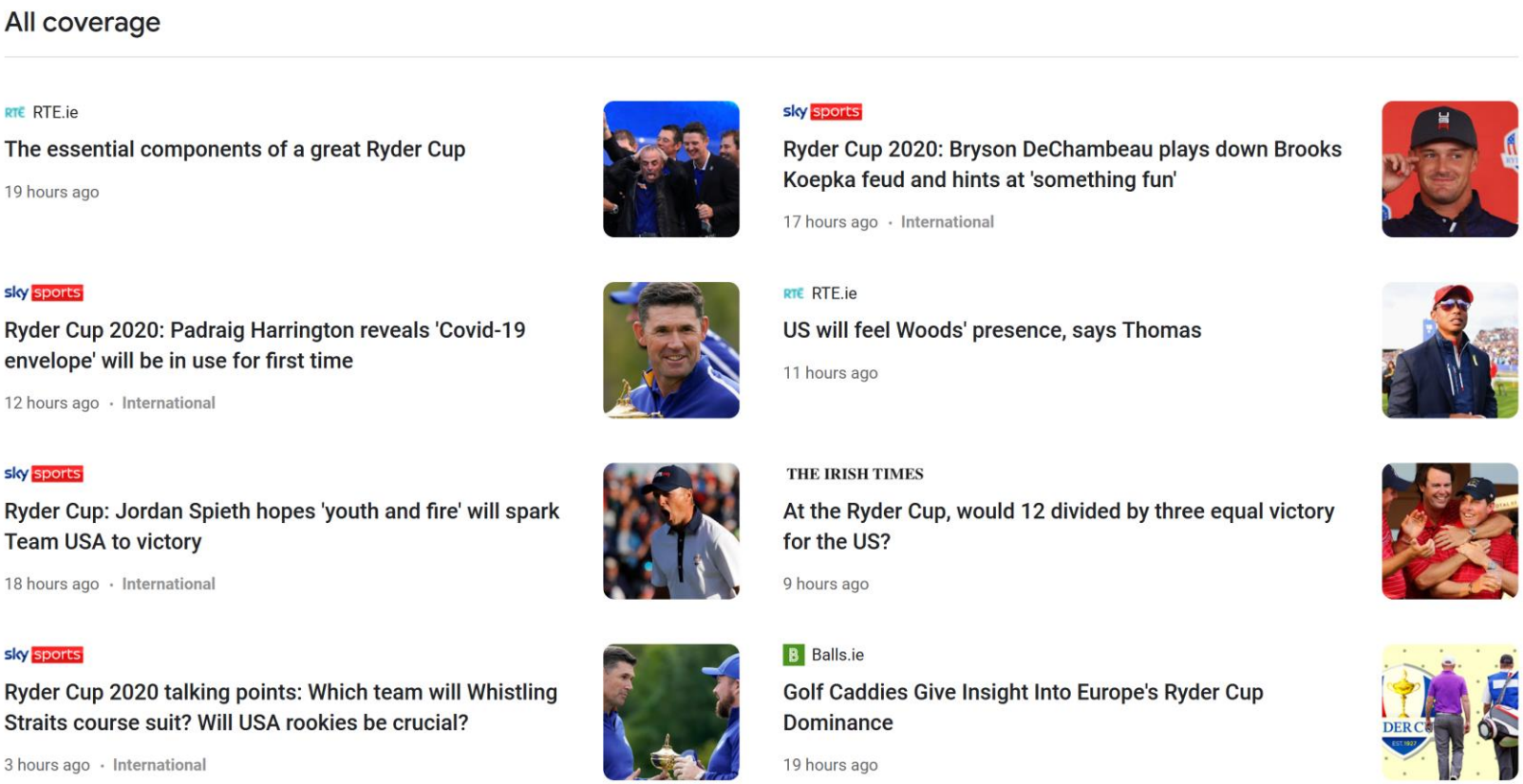
- Unsupervised algorithms seek out **patterns** or **data groupings** in unlabelled data (only features are available).
  - These groups are termed **clusters**.
  - There are a broad range of clustering machine learning techniques.
  - Example- K Means Clustering (is told in advance how many clusters it should form -- a potentially difficulty)


Feature 2
Feature 1


Feature 2
Feature 1

Applications

- Google news uses unsupervised learning to group new articles with related content.
- In this case it groups new articles from different sites in a single cluster due to their related content.

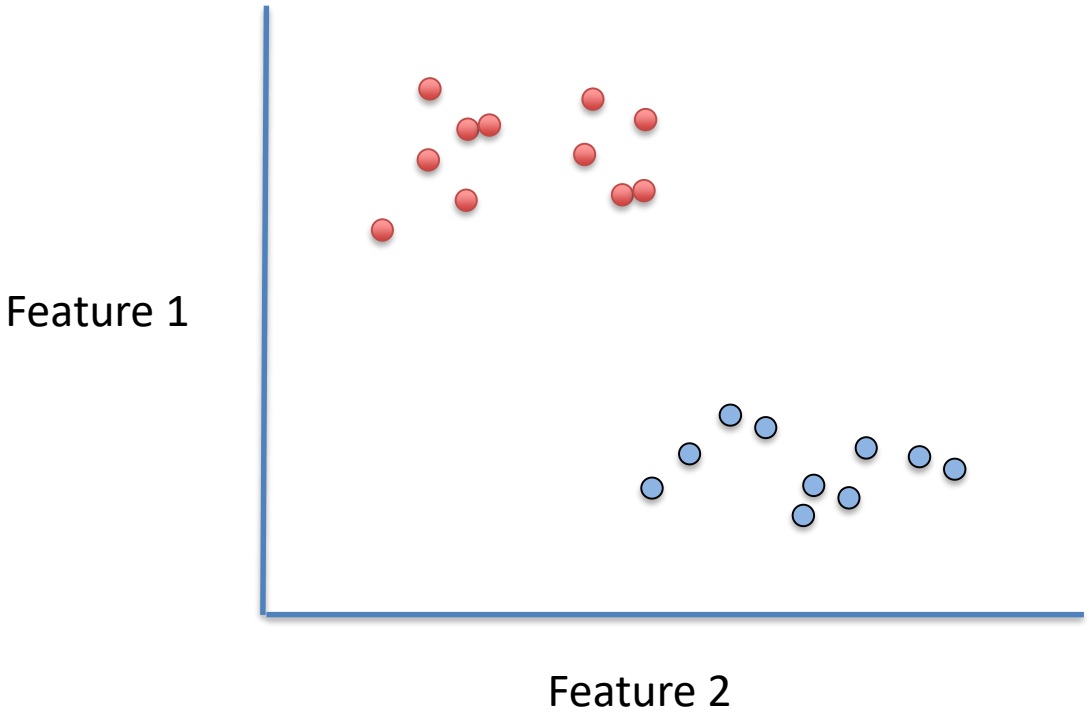


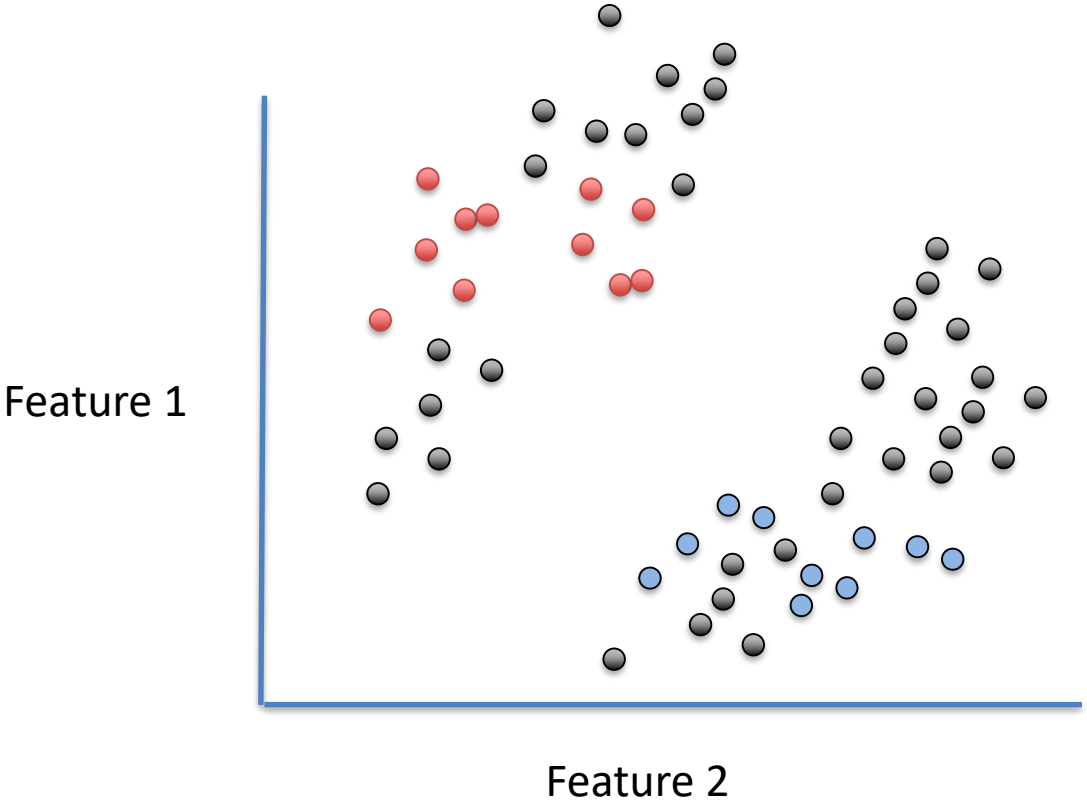
Semi-Supervised Learning

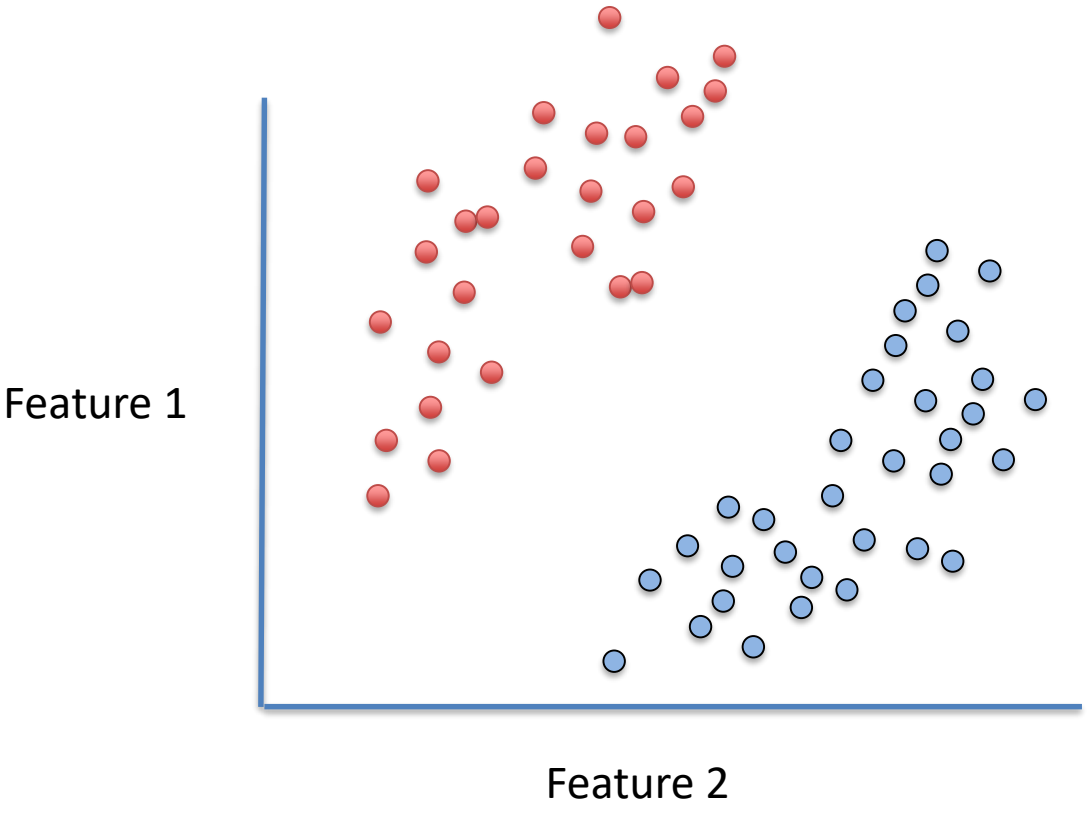
- The semi-supervised learning approach to machine learning **combines** supervised and unsupervised learning techniques.

- Remember supervised learning uses a labelled training set, while unsupervised learning techniques use unlabelled data.

- A semi-supervised approach **utilises both labelled and unlabelled data** for training.
  - Normally a small amount of labelled data is used along with a large amount of unlabelled data


Feature 1
Feature 2


Feature 1
Feature 2


Feature 1
Feature 2

Self Supervised Machine Learning Algorithms

- **Self supervised** learning occurs when we have an **unlabelled** dataset.
- However, the learning process we use is a **supervised learning process**.
- This seems counterintuitive! How can we use supervised learning techniques when the data isn't labelled???
- This is a little more challenging to grasp if you have limited exposure to ML.
- A good example of self supervised learning is a standard **generative adversarial network** (see next few slides).
- [Generative Adversarial Networks can be a little difficult to grasp so don't worry if you don't fully understand this now. We will be covering it in much more detail in the Deep Learning module next semester]

Real Dataset

Real Dataset

Artificial Dataset

Generator Network

The objective of the **generator** is to produce **new images** from the same distribution as the original dataset.

When the generator starts generating images initially they will be very poor. They won't look at all like the images from the real dataset.

We need to **teach the generator** how to produce authentic looking images.

To do this we will use a **supervised learning** algorithm.

But don't supervised learning algorithms take in labelled data?

Real Dataset

Class Label 1

Artificial Dataset

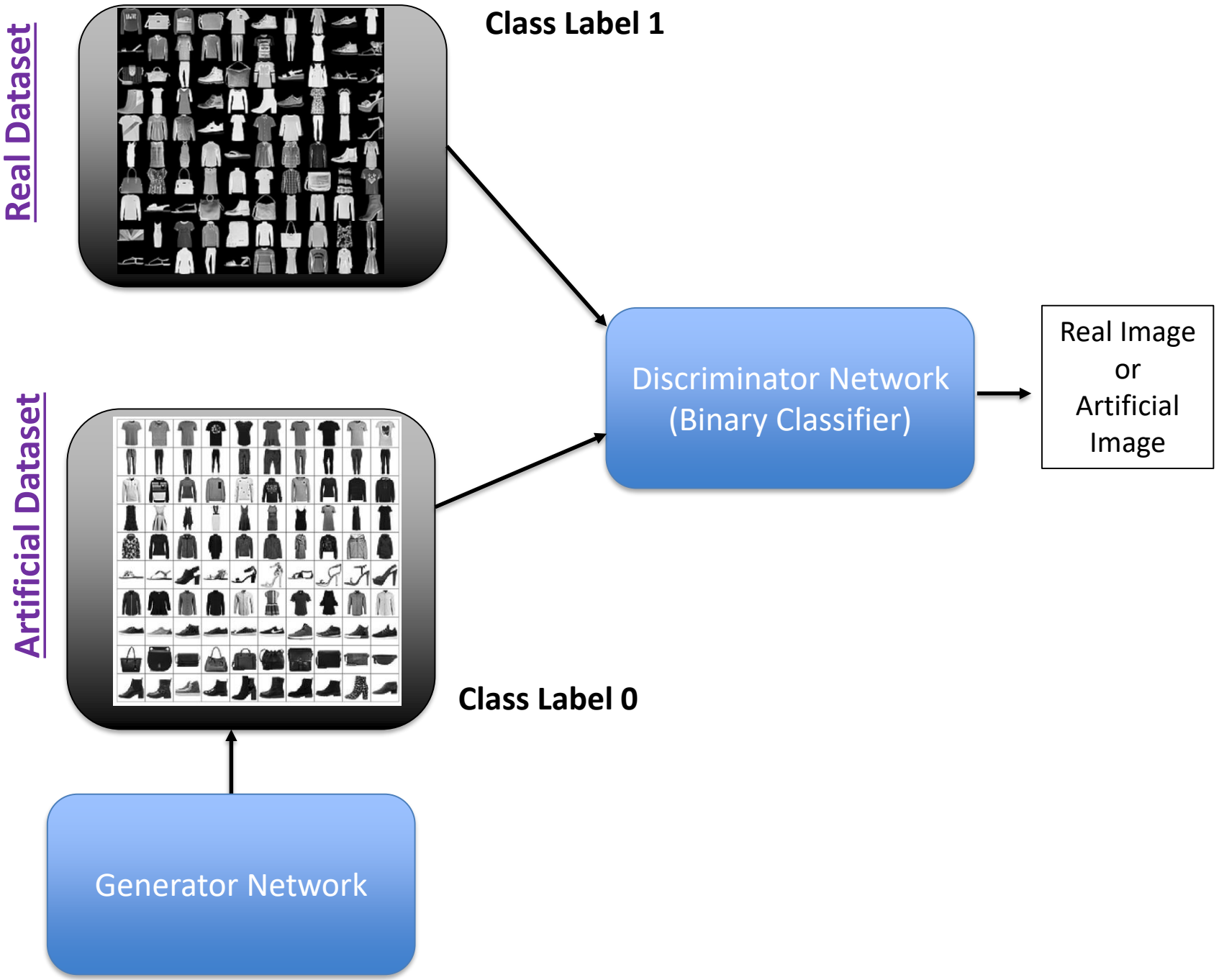
Class Label 0

Generator Network

To do this we will use a supervised learning algorithm. But don't **supervised** learning algorithms take in **labelled** data?

Correct.

So we label all the images in the true dataset as 1 and all the images in the artificial dataset as 0.


Real Dataset
Class Label 1
Artificial Dataset
Class Label 0
Discriminator Network
(Binary Classifier)
Real Image
or
Artificial
Image
Generator Network

Reinforcement Learning Algorithms

- The objective of Reinforcement learning algorithms is to utilize observed rewards to **learn an optimal (or near-optimal) set of actions (or policy)** for each state in a given environment.

- Unlike most other forms of machine learning the learner **knows nothing about their environment** and just has a set of available actions that it can take.

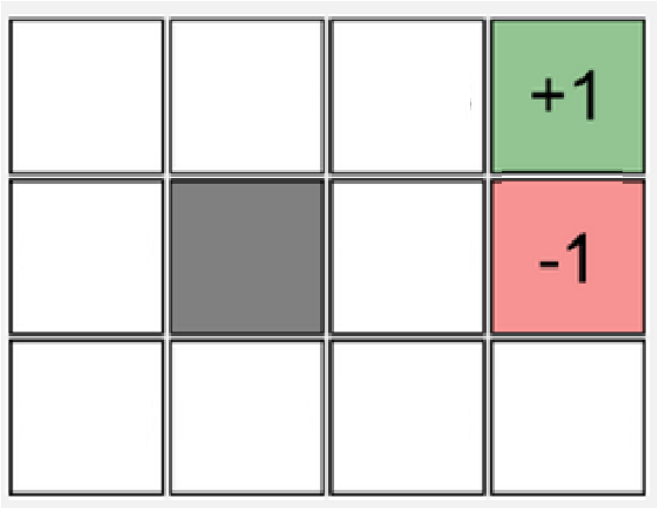
- The learner is not told which actions to take, but instead must **discover which actions yield the most reward** by trying them in the environment.

- This process continues until it reaches a positive or negative terminal or goal state. It then **positively or negatively reinforces** the actions that led to that state.

- Through repeated interaction with it's environment the agent develops a policy, which **defines what action the agent should choose when it is in a given situation.**

Reinforcement Learning Algorithms

- Take the simple environment on the left. An agent knows nothing about this environment but wants to navigate to the successful state.



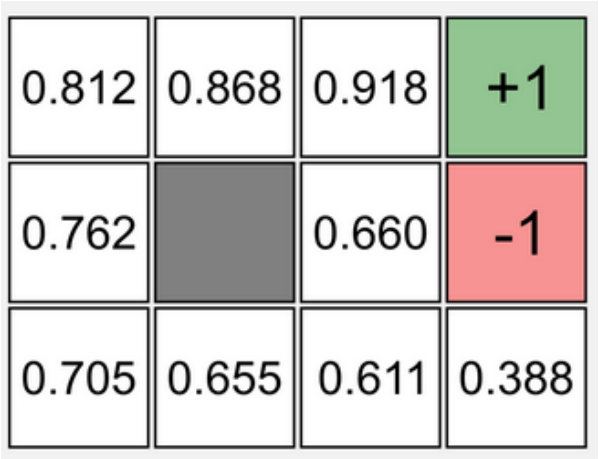
- They must start by exploring their environment. They have a certain number of actions available to them. (Up, down, left or right). Each action will take them to a new date on the grid.

- The **intended direction of movement occurs with a probability of 0.8**. With a 0.2 probability you will make a move at right angle to the intended direction. The terminal states have rewards of +1 and -1 respectively.

- **A reinforcement learning program will play this game many times and will develop and optimal policy over time.**

Reinforcement Learning Algorithms

- Take the simple environment on the left. An agent knows nothing about this environment but wants to navigate to the successful state.
- They must start by exploring their environment. They have a certain number of actions available to them. (Up, down, left or right). Each action will take them to a new date on the grid.
- The intended direction of movement occurs with a probability of 0.8. With a 0.2 probability you will make a move at right angle to the intended direction. The terminal states have rewards of +1 and -1 respectively.
- **A reinforcement learning program will play this game many times and will develop and optimal policy over time.**



Challenges in Machine Learning

- There are a range of challenges that you may encounter when attempting to build a machine learning model and these can be largely categorized into either **data-based issues** or **model based issues**.

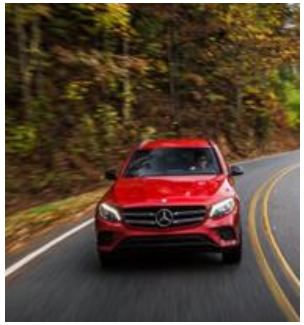
- **Insufficient Amount of Training Data**
  - To work well ML algorithms commonly need quite a lot of data.
  - Even for very simple problems you may often need many hundreds of training examples, and for complex problems such as image or speech recognition you may need **millions** of examples (unless you can reuse parts of an existing model).
  - In some cases **poor model performance** could be due to a **lack of data**. It is important to be able to **diagnose** you ML model to determine if a lack of data may improve it's overall level of accuracy.

Challenges in Machine Learning

- **Non-representative Training Data**
  - In order to generalize well, it is crucial that your training data be representative of the new cases you want to generalize to.
  - By using a non-representative training set, we will train a model that is unlikely to make accurate predictions.
  - Very large datasets can be non-representative if the sampling method is flawed. This is often referred to as sampling bias.







Challenges in Machine Learning

- Issues with your Data (**Pre-processing**)
  - After you have collected data it will likely not be suitable for a ML algorithm. For example, your training set may contain some **features** that have little to no relationship with the target you are trying to predict. Also there could be **outliers** in your data or **missing values** (e.g., due to poor quality measurements, faulty sensors, etc), the distribution of class labels might be very **imbalanced**.
  - It is very common to spend time **pre-processing** and cleaning up your training data.
  - We will delve into this in much detail later in the module.

Model Challenges in ML

- Issues with your models
- Of course once you address the challenges inherit in the data you must now also tackle the challenges associated with the **models**.
- There are a broad range of models that we can use. How do we determine **which model to use**?
- Each model has many **parameters** that we can use to tune it's performance. How do we decide on what parameters to set?

Model Challenges in ML

- **Evaluating** your model.

- It is very important that you use the **correct methodology** for **evaluating** the performance of your model. We will talk about this in much more detail later in the module.

- However, one basic rule is that you should **never assess the performance of a model using the data that used to train it**.

- ML models are very powerful algorithms that can fit the training data very tightly and fail to generalize to unseen data. We refer to this issue as **overfitting**.