CORK INSTITUTE OF TECHNOLOGY
INSTITIÚID TEICNEOLAÍOCHTA CHORCAÍ

Machine Learning

Machine Learning

Lecture: Introduction to Machine Learning

Ted Scully

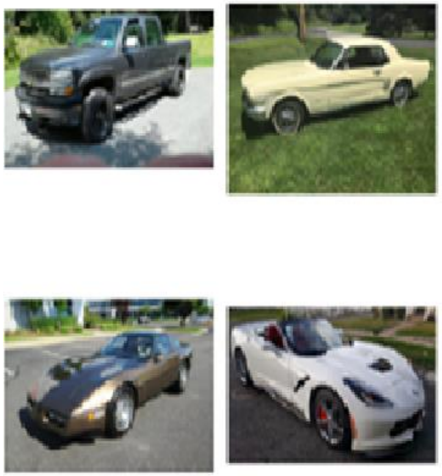
Challenges in Machine Learning

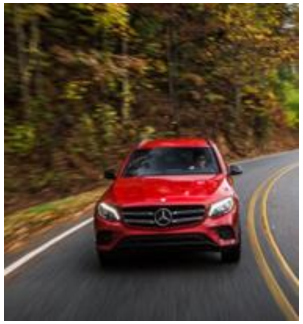
- There are a range of challenges that you may encounter when attempt to build a machine learning model and these can be largely categorized into either **data-based issues** or **model based issues**.

- **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.
  - This is often harder than it sounds: if the sample is too small, you will have sampling noise (i.e., non-representative data as a result of chance), but even very large samples 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
  - If your training data is full of errors. For example, your training set may contain some features that have little to no relationship with the class you are trying to predict. There could be **outliers in your data or missing values** (e.g., due to poor quality measurements, faulty sensors, etc).
  - It is very common to spend time cleaning up your training data.
  - For example, if some instances are clearly **outliers**, it may help to simply discard them.
  - If some instances are **missing** a few features (e.g., 5% of your customers did not specify their age), you must decide whether you want to ignore this feature altogether, ignore these specific instances with missing value, fill in the missing values.
  - A critical part of the success of a Machine Learning project is coming up with a good set of features to train on. **Feature selection**: selecting the most useful features to train on among existing features.
  - There are many other issues that must be faced such as **imbalanced** data, etc. More on this 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

- How do you **assess the performance** of a machine learning model.

- It is very important that you use the **correct methodology** for assessing your model performance. 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**.

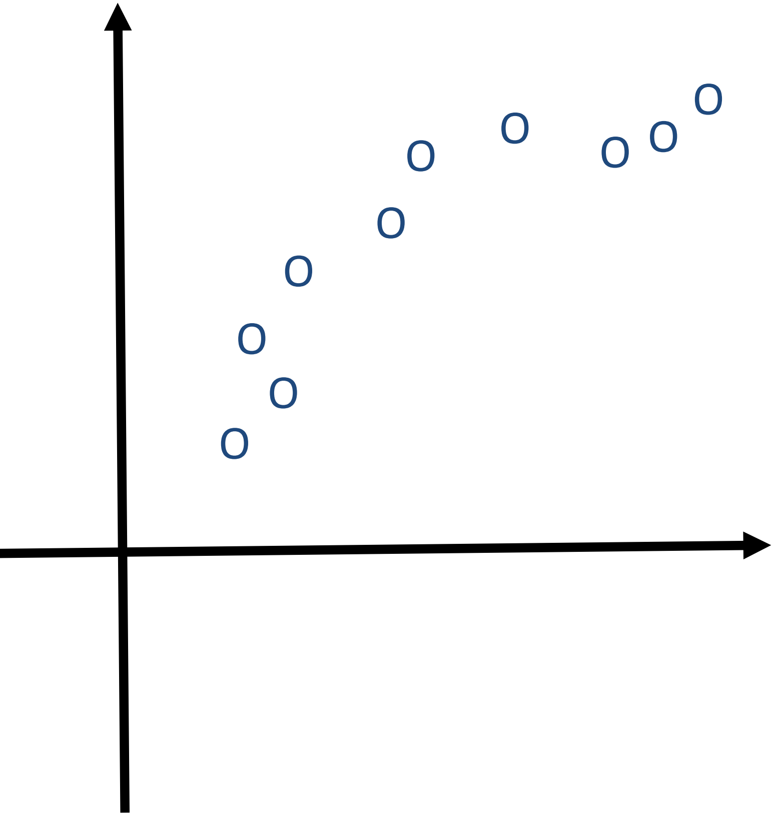
- The reason for this is that ML models are very powerful algorithms that can fit the training data very tightly and fail to generalize to unseen data. We refer to this issue as **overfitting**.

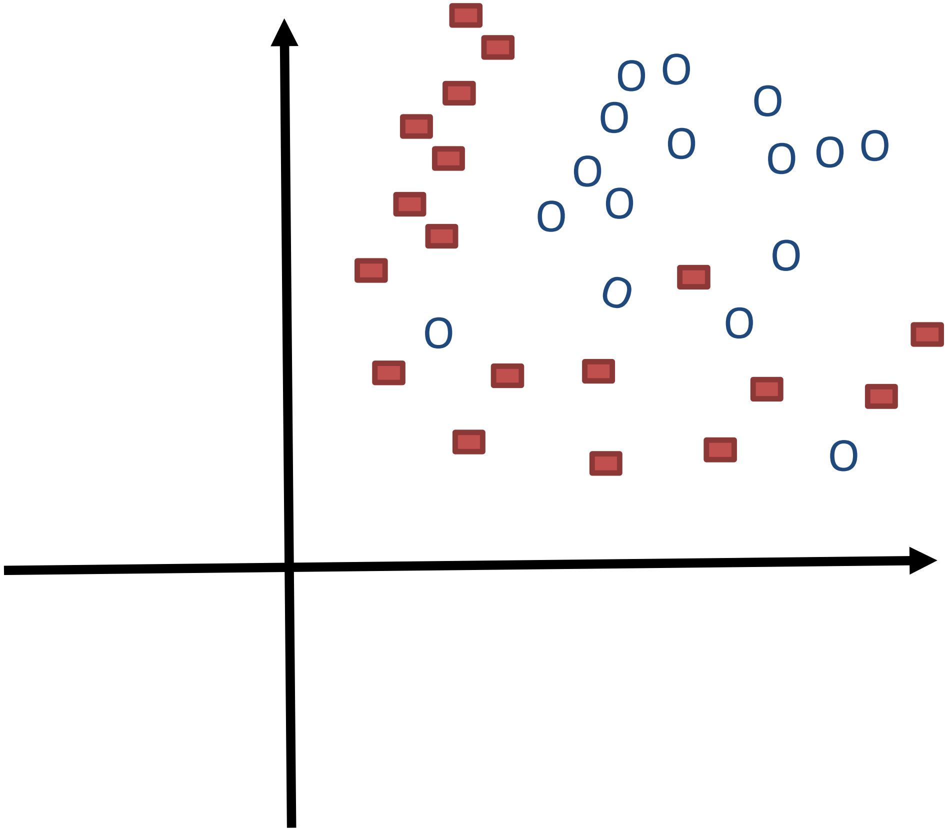
Overfitting

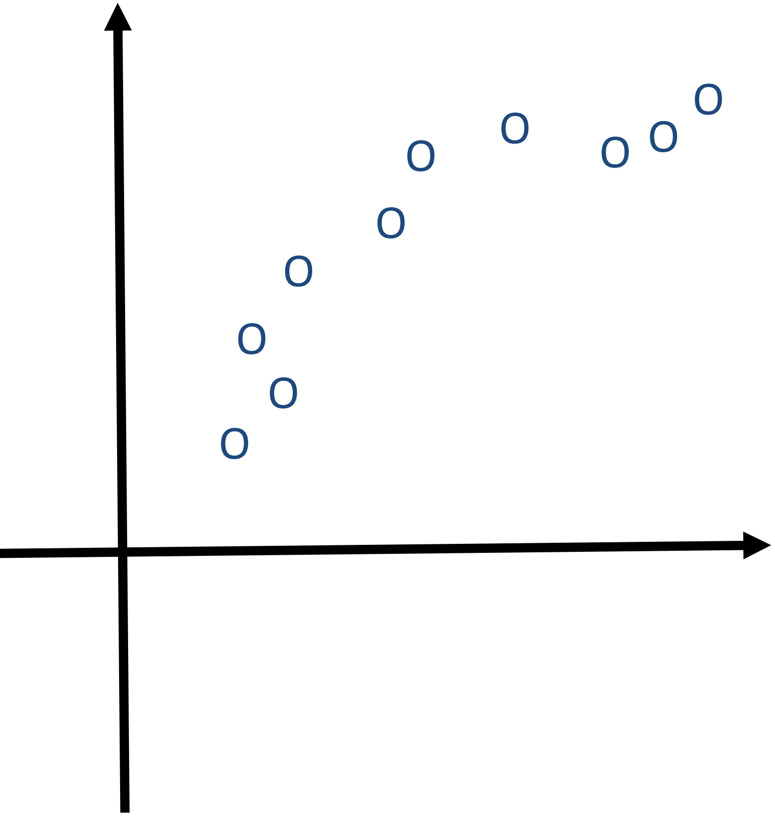
- Overfitting generally occurs when a **model/function is excessively complex and has fit too tightly to the training data.**

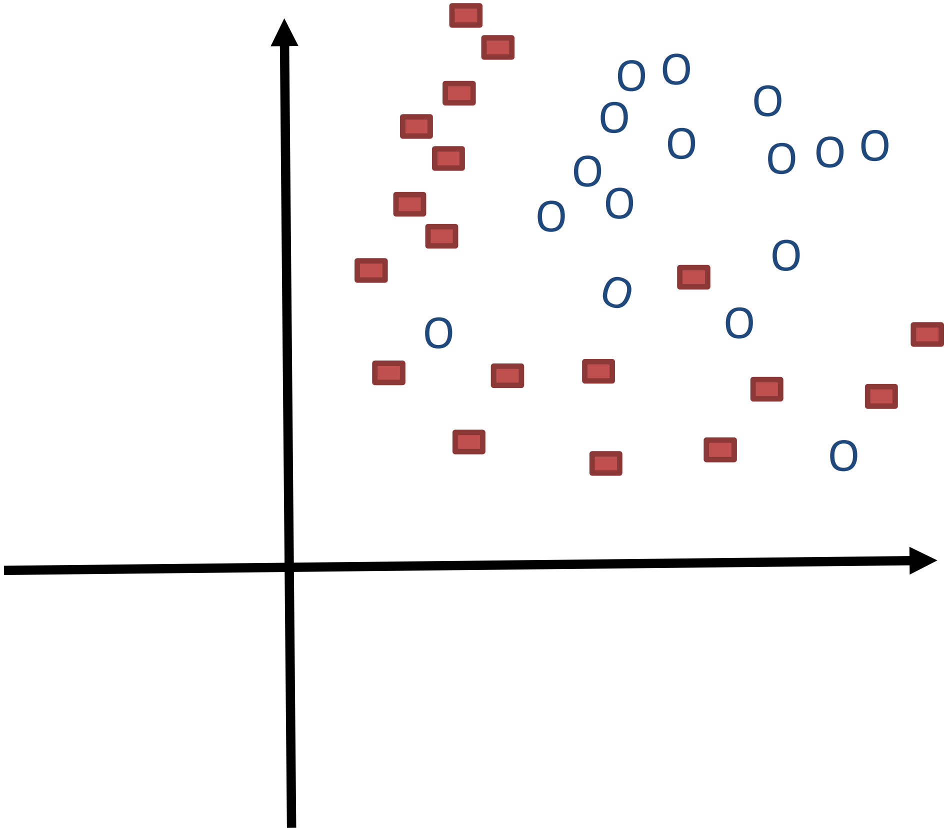
- A model/**function which has been overfit will generally have poor predictive performance on unseen data (it doesn't generalize well to unseen examples)**, as it can exaggerate minor fluctuations in the data.
  - A model is typically **trained** by maximizing its performance on some set of training data.
  - However, its overall performance is determined not by its performance on the training data but by its ability to perform well on **unseen data**.

- You can think of this as the difference between <u>memorizing</u> the data and <u>generalizing</u> from the data.





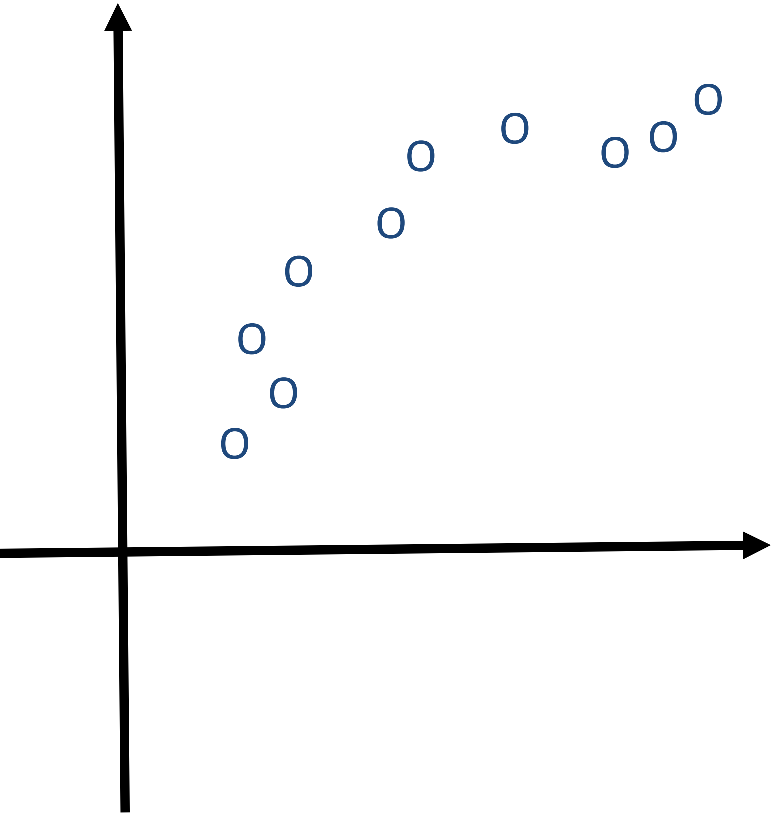


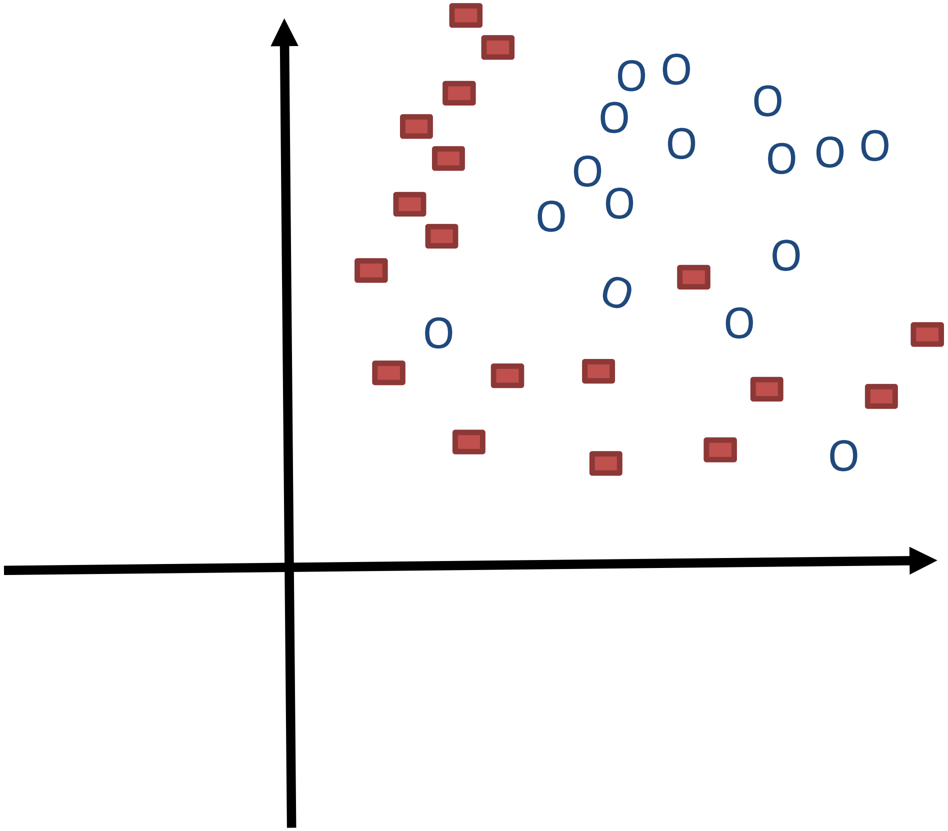


Underfitting

- As you might guess, underfitting is the opposite of overfitting: it occurs when your model is unable to learn the underlying structure of the data.

- Often this issue can be addressed by :
  - Selecting a more **complex model**, with more parameters
  - Feeding **better features** to the learning algorithm (feature engineering)

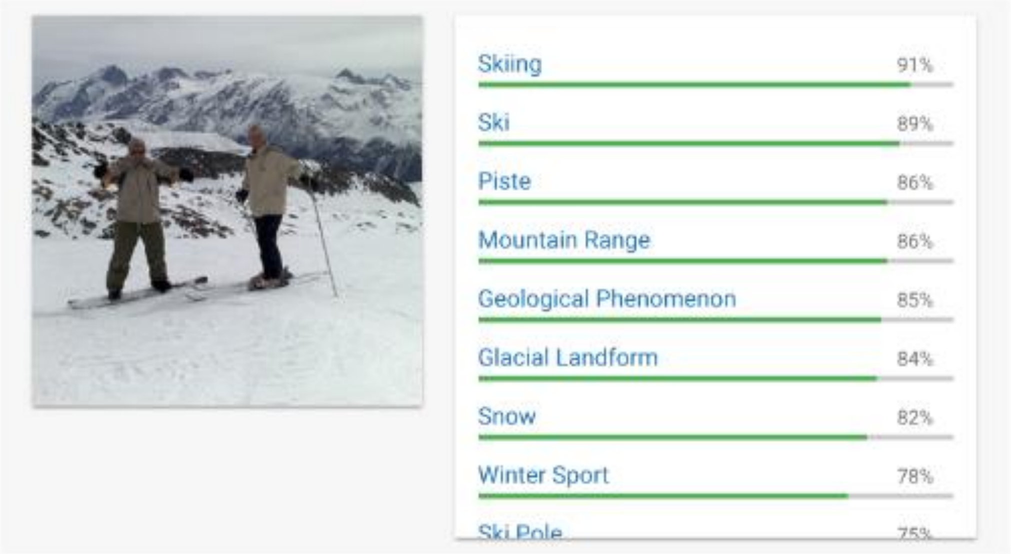




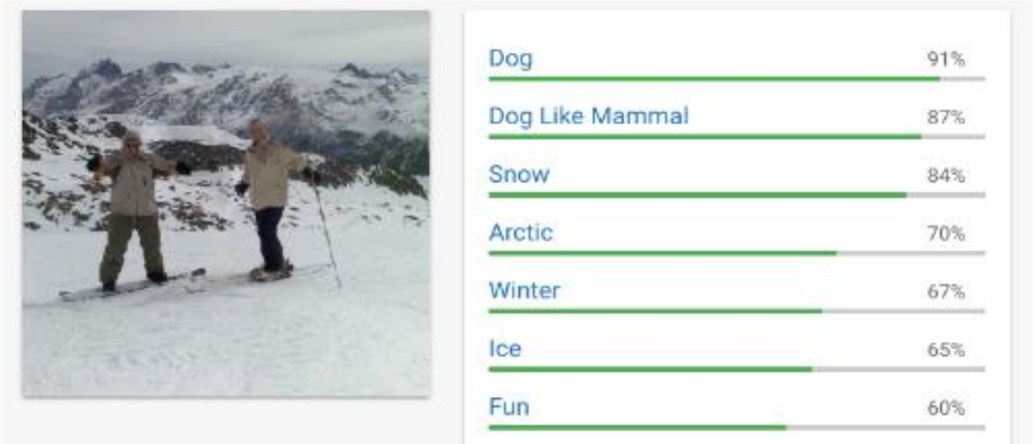
ML Security Challenge

- While the challenges outlined the previous slides are confined to the ML model and the data there an emergent concerns around the security of ML models.

- For example, research has demonstrated that they can ‘fool’ deep learning vision systems for perturbing some of the pixels in an image.



Skiing	91%
Ski	89%
Piste	86%
Mountain Range	86%
Geological Phenomenon	85%
Glacial Landform	84%
Snow	82%
Winter Sport	78%
Ski Pole	75%

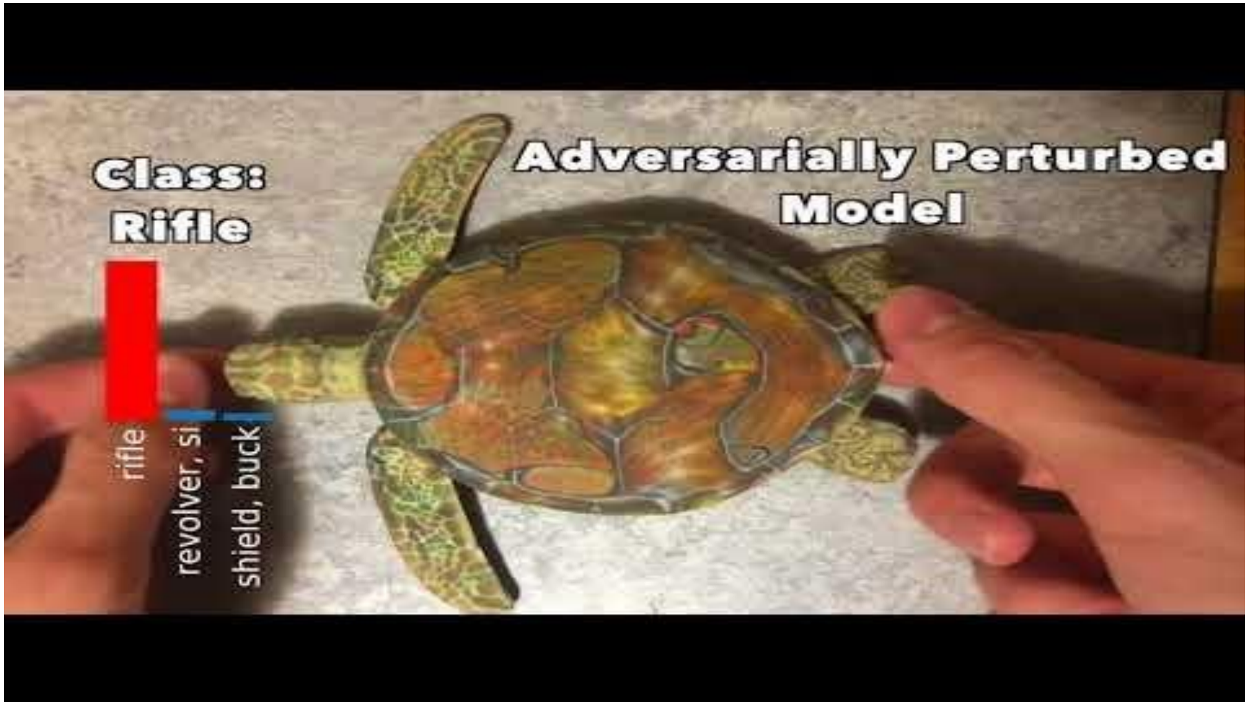


Dog	91%
Dog Like Mammal	87%
Snow	84%
Arctic	70%
Winter	67%
Ice	65%
Fun	60%

Andrew Ilyas, Logan Engstrom, Anish Athalye, Jessy Lin, Black-box Adversarial Attacks with Limited Queries and Information

Security Challenge

- The example is taken from the following paper.



Anish Athalye, Logan Engstrom, Andrew Ilyas, Kevin Kwok Synthesizing Robust Adversarial Examples 2018

CORK INSTITUTE OF TECHNOLOGY

INSTITIÚID TEICNEOLAÍOCHTA CHORCAÍ

Machine Learning

Machine Learning

Lecture: Bayesian Classification

Ted Scully

Contents

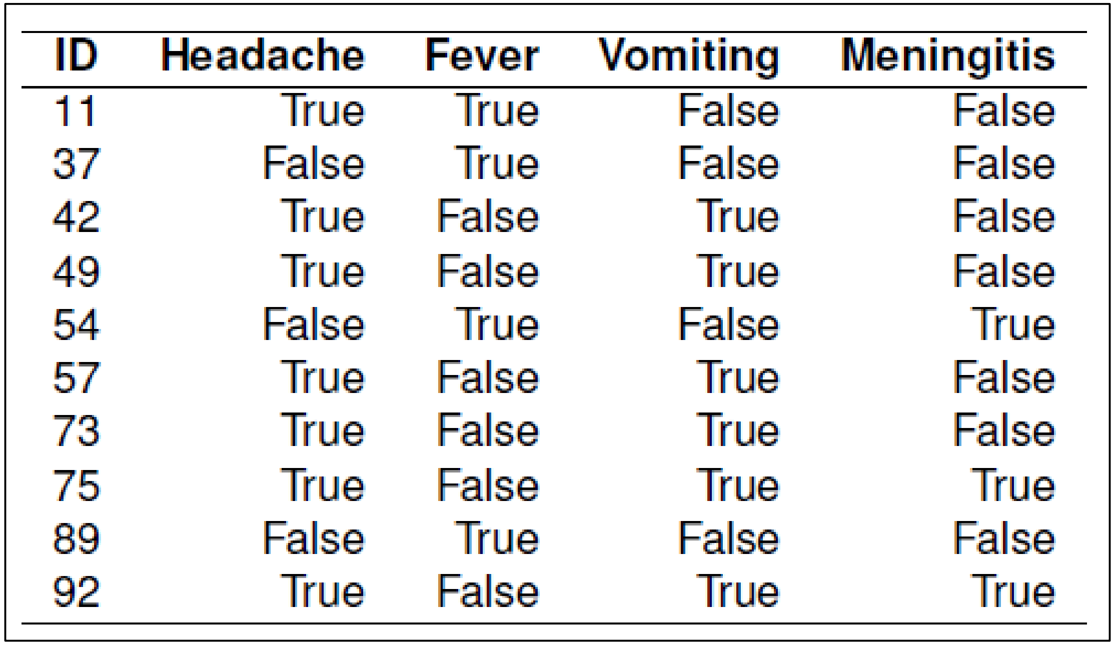
1. <u>Probability distributions, rules and Bayes theorem</u>

2. Classification Example using Naïve Bayes

3. Text Classification Using Naïve Bayes

Review of Basic Concepts

- Over the next few slides we will review some basic probability concepts and the use the following sample dataset to help illustrate.



ID	Headache	Fever	Vomiting	Meningitis
11	True	True	False	False
37	False	True	False	False
42	True	False	True	False
49	True	False	True	False
54	False	True	False	True
57	True	False	True	False
73	True	False	True	False
75	True	False	True	True
89	False	True	False	False
92	True	False	True	True

Machine Learning for Predictive Analytics – Kelleher et al.

Review of Basic Concepts and Terminology

- An **event** defines an **assignment of values** to the **features** in the domain; these assignments may define values for all the features in the domain (e.g. a full row in the dataset) or just to one or more features (Fever = True).

- A **probability function** is a function that takes an event (an assignment of values to features) as a parameter and returns the likelihood of that event (P(Fever = True)).

- The value returned by a probability function for an event is simply the **relative frequency** of that event in the dataset

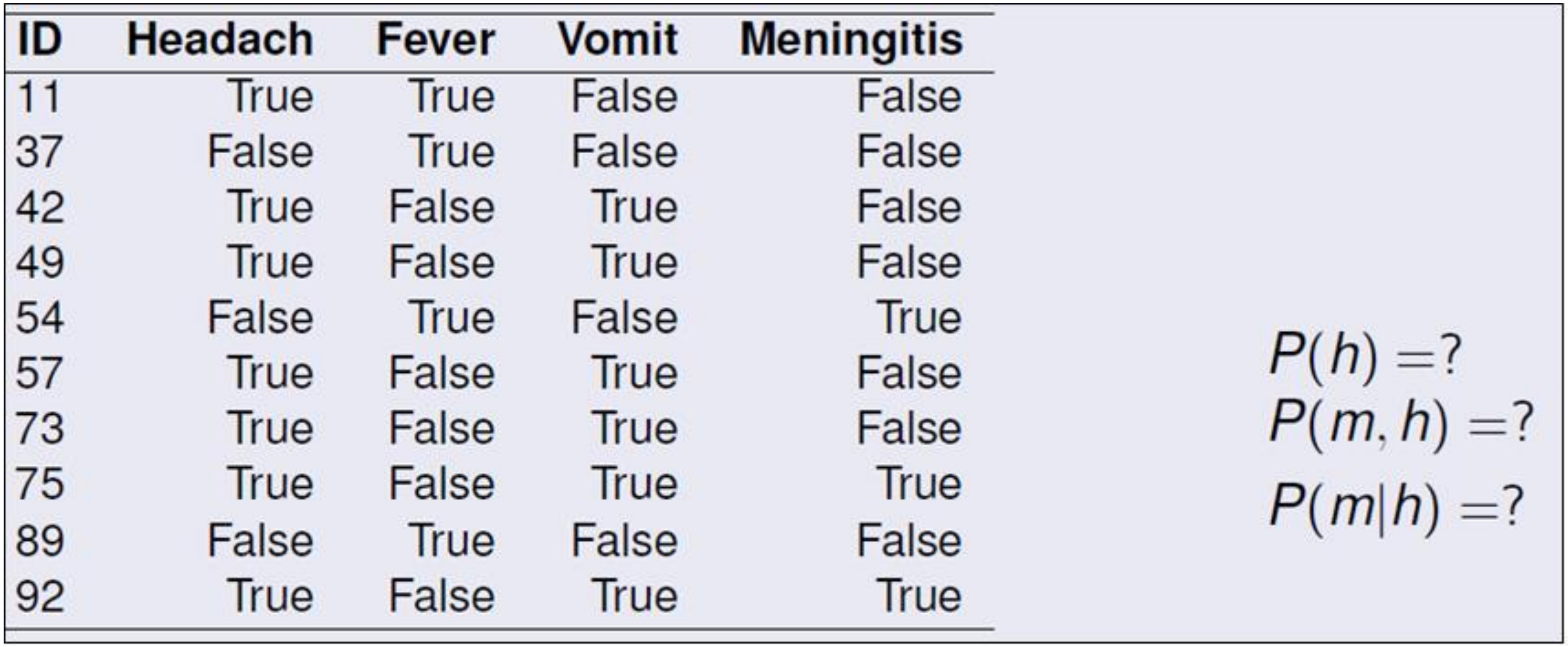
- In other words, how often the event happened divided by how often it could have happened.

Review of Basic Concepts

- **Prior probability** (**Unconditional probability**) : The probability of an event that is not dependent on any other feature.
  - The count of all the rows in the dataset where the feature is assigned the relevant value divided by the number of rows in the dataset.

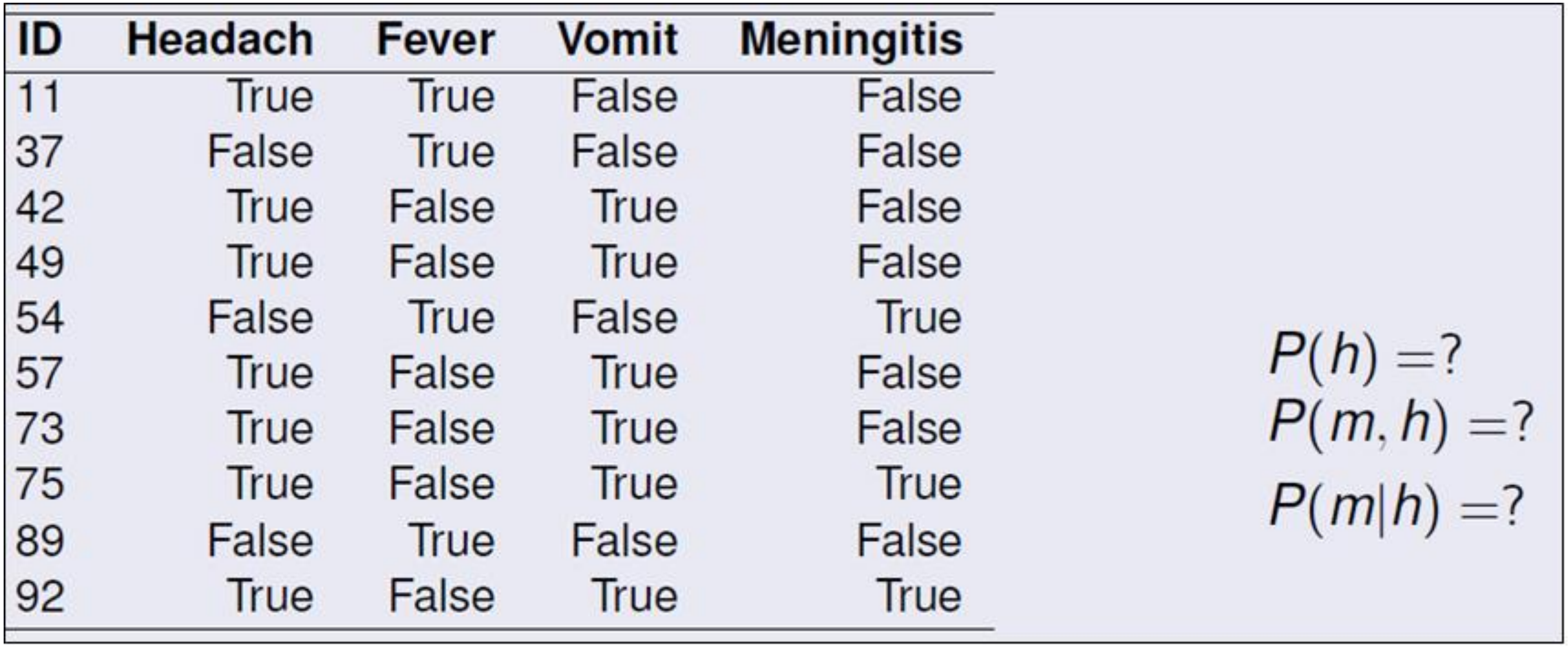
- **Joint probability**: The probability of two or more events happening together.
  - The number of rows in the dataset where the set of assignments listed in the joint event holds divided by the total number of rows in the dataset.

- **Posterior Probability** (**Conditional Probability**): The probability of an event where one or more other events are known to have happened.
  - The number of rows in the dataset where both events are true divided by the number of rows in the dataset where just the given event is true.



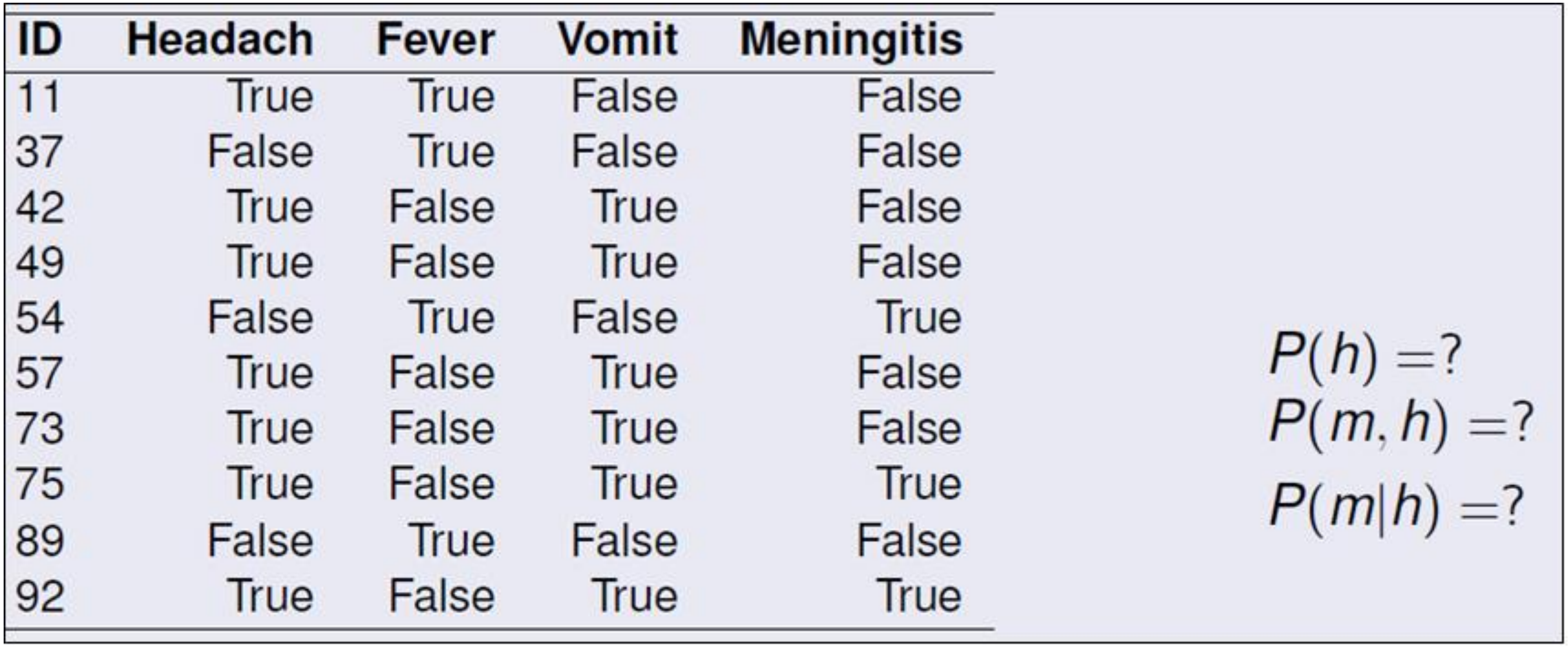
ID	Headach	Fever	Vomit	Meningitis
11	True	True	False	False
37	False	True	False	False
42	True	False	True	False
49	True	False	True	False
54	False	True	False	True
57	True	False	True	False
73	True	False	True	False
75	True	False	True	True
89	False	True	False	False
92	True	False	True	True

$$P(h) = ?$$
$$P(m, h) = ?$$
$$P(m|h) = ?$$



ID	Headach	Fever	Vomit	Meningitis
11	True	True	False	False
37	False	True	False	False
42	True	False	True	False
49	True	False	True	False
54	False	True	False	True
57	True	False	True	False
73	True	False	True	False
75	True	False	True	True
89	False	True	False	False
92	True	False	True	True

$$P(h) = ?$$
$$P(m, h) = ?$$
$$P(m|h) = ?$$

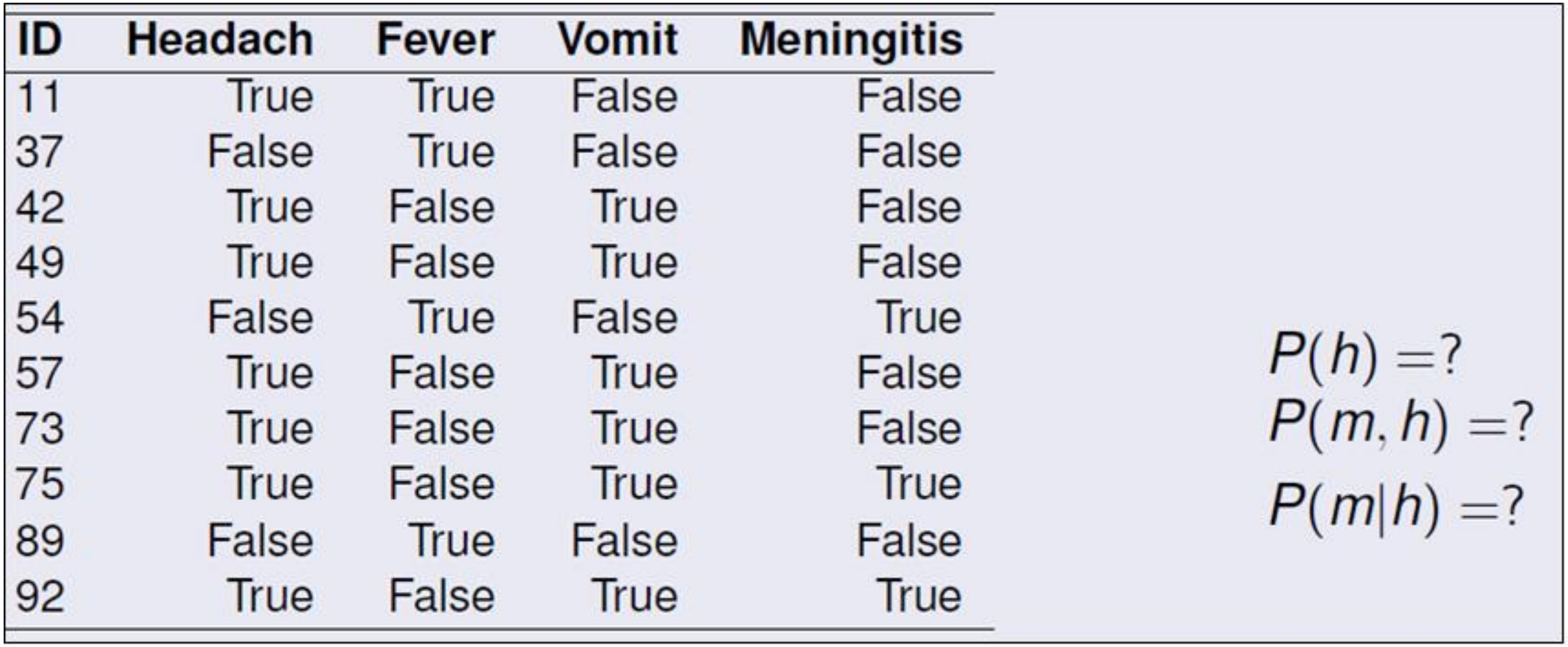


ID	Headach	Fever	Vomit	Meningitis
11	True	True	False	False
37	False	True	False	False
42	True	False	True	False
49	True	False	True	False
54	False	True	False	True
57	True	False	True	False
73	True	False	True	False
75	True	False	True	True
89	False	True	False	False
92	True	False	True	True

$$P(h) = ?$$

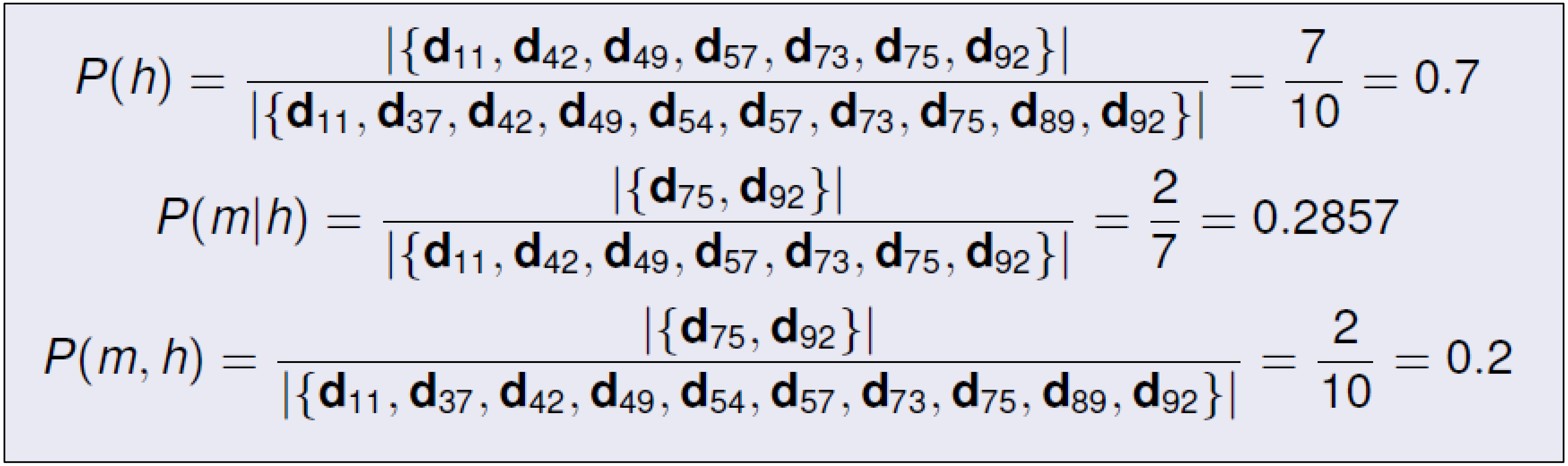
$$P(m, h) = ?$$

$$P(m|h) = ?$$



ID	Headach	Fever	Vomit	Meningitis
11	True	True	False	False
37	False	True	False	False
42	True	False	True	False
49	True	False	True	False
54	False	True	False	True
57	True	False	True	False
73	True	False	True	False
75	True	False	True	True
89	False	True	False	False
92	True	False	True	True

$$P(h) = ?$$
$$P(m, h) = ?$$
$$P(m|h) = ?$$



$$P(h) = \frac{|\{\mathbf{d}_{11}, \mathbf{d}_{42}, \mathbf{d}_{49}, \mathbf{d}_{57}, \mathbf{d}_{73}, \mathbf{d}_{75}, \mathbf{d}_{92}\}|}{|\{\mathbf{d}_{11}, \mathbf{d}_{37}, \mathbf{d}_{42}, \mathbf{d}_{49}, \mathbf{d}_{54}, \mathbf{d}_{57}, \mathbf{d}_{73}, \mathbf{d}_{75}, \mathbf{d}_{89}, \mathbf{d}_{92}\}|} = \frac{7}{10} = 0.7$$

$$P(m|h) = \frac{|\{\mathbf{d}_{75}, \mathbf{d}_{92}\}|}{|\{\mathbf{d}_{11}, \mathbf{d}_{42}, \mathbf{d}_{49}, \mathbf{d}_{57}, \mathbf{d}_{73}, \mathbf{d}_{75}, \mathbf{d}_{92}\}|} = \frac{2}{7} = 0.2857$$

$$P(m, h) = \frac{|\{\mathbf{d}_{75}, \mathbf{d}_{92}\}|}{|\{\mathbf{d}_{11}, \mathbf{d}_{37}, \mathbf{d}_{42}, \mathbf{d}_{49}, \mathbf{d}_{54}, \mathbf{d}_{57}, \mathbf{d}_{73}, \mathbf{d}_{75}, \mathbf{d}_{89}, \mathbf{d}_{92}\}|} = \frac{2}{10} = 0.2$$

Review of Basic Concepts

- **Probability Distribution** : For all the possible values of a feature it describes the probability of the feature taking that value.
- A probability distribution of a categorical feature is a vector that lists the probabilities associated with the values in the domain of the feature.

- **Joint Probability Distribution** : It gives us an exhaustive list of probabilities for all possible combinations of values for a specified set of features.
- A **full joint probability distribution** is simply a joint probability distribution over all the features in a domain.

$$\mathbf{P}(H,F,V,M) = \left[\begin{array}{ll} P(h,f,v,m), & P(\neg h,f,v,m) \\ P(h,f,v,\neg m), & P(\neg h,f,v,\neg m) \\ P(h,f,\neg v,m), & P(\neg h,f,\neg v,m) \\ P(h,f,\neg v,\neg m), & P(\neg h,f,\neg v,\neg m) \\ P(h,\neg f,v,m), & P(\neg h,\neg f,v,m) \\ P(h,\neg f,v,\neg m), & P(\neg h,\neg f,v,\neg m) \\ P(h,\neg f,\neg v,m), & P(\neg h,\neg f,\neg v,m) \\ P(h,\neg f,\neg v,\neg m), & P(\neg h,\neg f,\neg v,\neg m) \end{array} \right]$$

Review of Basic Concepts

- Can you think of any problem we might encounter with calculating a joint probability distribution as we **increase the number of features** (and the number of values for those features also increases).

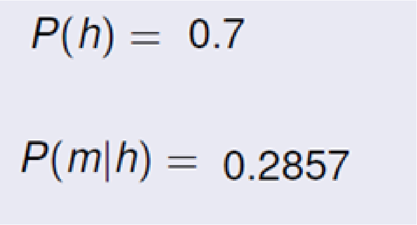
Review of Basic Concepts

- Unfortunately the size of a **full joint probability distribution** grows exponentially as the number of features and the number of values in the domain of the features grow. Consequently, they are difficult to generate.

- Remember computing each probability value in the joint probability distribution requires a set of instances.

- As we add additional features the size of the **distribution grows exponentially** but so too does the **size of the dataset required** to generate the joint probability distribution.

- Therefore, for domains of any reasonable complexity it is **not tractable** to build a full joint probability distribution.



$P(h) = 0.7$

$P(m|h) = 0.2857$

Product Rule

- The product rule is shown below. In this form it allows us to calculate the joint probability of two events.

$$\underline{P(a, b) = P(a|b)\,P(b) = P(b|a)\,P(a)}$$

- Using the product rule let's calculate the joint probability of P(m, h)

- P(m,h)

Bayes Rule

- Over the next few slides we are going to look at Bayes rule.
- We are going to look at this in a classification setting and I'm going to use the following notation.
- When using the **notation** ***c*** we refer to a specific class
- When using ***d*** we are referring to a new data instance (data to be classified).

Bayes’ Rule

P(c **,** d) = **P(c | d) P(d) = P(d|c)P(c) [Product Rule]**

Bayes’ Rule can be easily derived from the product rule

$$P(c|d) = \frac{P(d|c)P(c)}{P(d)}$$