



COMP9033
DATA ANALYTICS

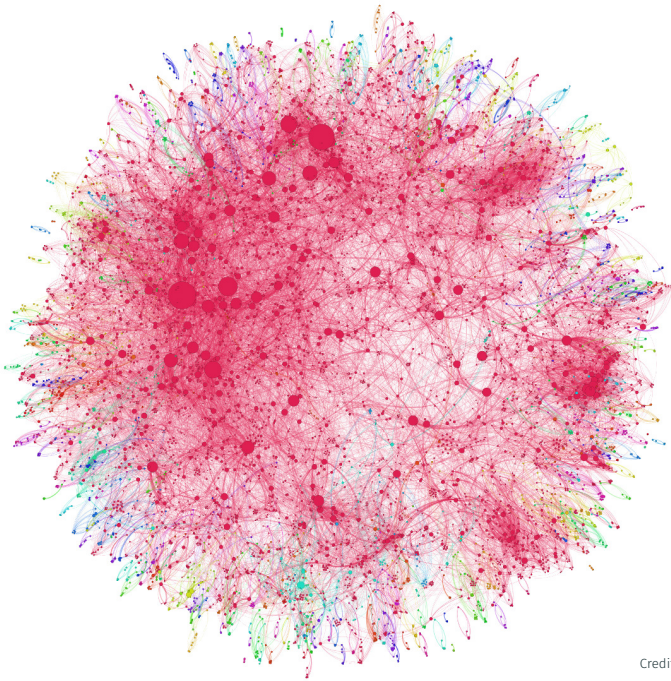
1/12

INTRODUCTION TO DATA ANALYTICS

DR. DONAGH HORGAN

DEPARTMENT OF COMPUTER SCIENCE
CORK INSTITUTE OF TECHNOLOGY

2017.02.01



Overview

1. Introduction to data analysis:

- What is it?
- How does it work?
- Real world examples.

2. Module outline:

- Overview of topics.
- Marking scheme.
- Lab work.
- Project work.
- Contact information.

3. Data analysis processes:

- What are they?
- Why use them?
- How do they work?
- Which one to use?

4. Data sampling:

- What is it?
- Why is it important?
- How to do it?

Introduction to data analysis

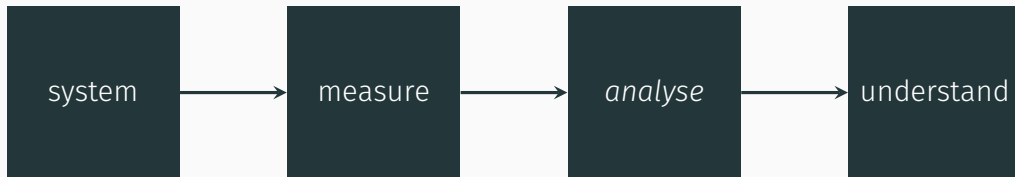
1.1 / WHAT IS DATA ANALYTICS?

- Data analytics is an area of science concerned with the analysis and understanding of data.
- In recent years, it has become a hot topic, with notable uses in areas such as
 - Search engines: Google, Yahoo, DuckDuckGo.
 - Speech recognition: Siri, Alexa, Cortana.
 - Music fingerprinting: Shazam, SoundHound.
 - Customer intelligence: Tesco, Amazon.
 - Recommendation systems: Netflix, Spotify, Audible.
 - Spam detection: Gmail, Outlook, Yahoo.
- It encompasses concepts such as statistics, visualisation and machine learning, but it is these things and more!

1.2 / WHAT IS DATA ANALYTICS?

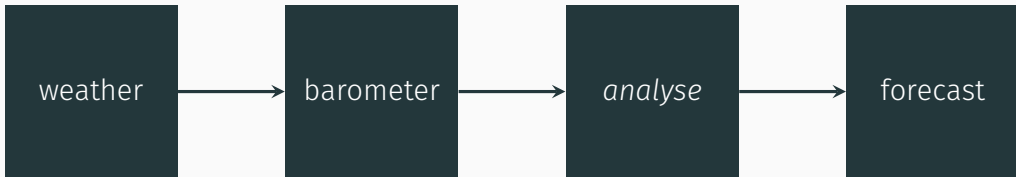
- Formally, data analytics is a systematic way of examining and manipulating data to discover new information.
- It is a *scientific* method, with defined steps and procedures:
 - Observations are made.
 - Hypotheses are formulated, refined, accepted and rejected.
 - Generally applicable conclusions are reached.
- It is also an *art*, often relying on subjective human judgement:
 - Should the data be manipulated and, if so, how?
 - Which technique or tool is best to use in a given situation?
 - What level of cost-performance trade off is acceptable?

1.3 / WHAT IS DATA ANALYTICS?



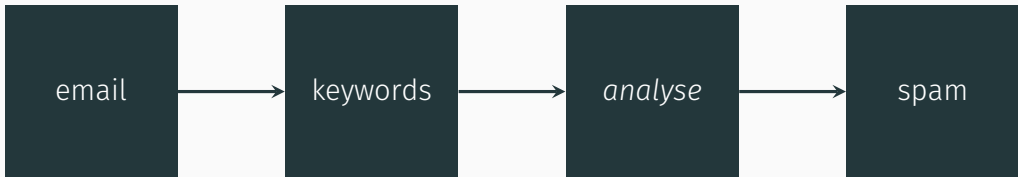
- We can think of data analysis as a step in a process:
 1. We have a system we want to understand.
 2. We measure some data related to the system.
 3. We *analyse* the data to better understand it.
 4. We gain understanding and draw conclusions.

1.4 / EXAMPLE: WEATHER FORECASTING



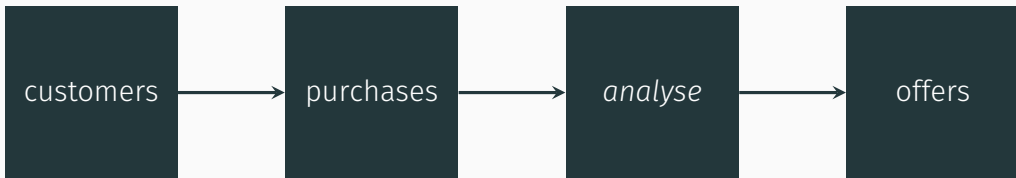
- Weather forecasting is a form of data analysis:
 1. We want to make predictions about the weather.
 2. We measure some data related to it, *e.g.* atmospheric pressure.
 3. We *analyse* the data to extract information about the relationship.
 4. We better understand how pressure affects weather, *e.g.* high pressure → sunshine!

1.5 / EXAMPLE: SPAM DETECTION



- Spam detection is a form of data analysis:
 1. We want to detect whether incoming emails are spam or not.
 2. We measure some data related to this, *e.g.* keywords in the email text.
 3. We *analyse* the data to extract information about the relationship.
 4. We better understand how certain keywords are good indicators of spam.

1.6 / EXAMPLE: DISCOUNT OFFERS



- Data analysis can be used to make offers to customers:
 1. We want to understand and/or predict customer behaviour.
 2. We measure some data related to this, *e.g.* purchase history.
 3. We *analyse* the data to extract information about the relationship.
 4. We better understand which items to offer discounts on to encourage purchases.

1.7 / WHAT IS DATA ANALYTICS?

- So what does data analysis actually involve?
- It can take a variety of forms, including:
 - Examination of statistical measures, *e.g.* mean, median, standard deviation.
 - Visualisation of the data, *e.g.* histogram, scatter plot matrix.
 - Transformation of the data into a form that makes further analysis more convenient.
 - Building and testing models of the data.
- Typically, multiple forms are used in combination to solve a given problem.
- The key to solving problems effectively is to know which of these tools to use and in what order.

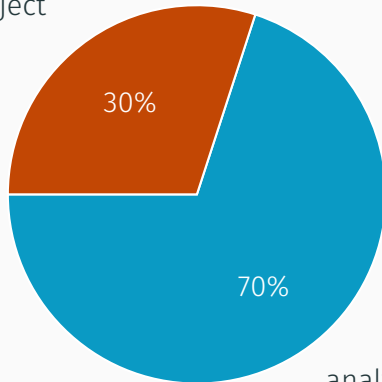
Module outline

- The aim of this module is to provide both a theoretical and a practical introduction to data analysis techniques.
- Over the coming weeks, we will cover a variety of topics:
 1. Introduction to data analytics.
 2. Exploratory data analysis.
 3. Data visualisation.
 4. Cleaning and transforming data.
 5. Data modelling.
 6. Decision tree classification.
 7. Linear regression.
 8. k -nearest neighbours regression.
 9. Clustering algorithms.
 10. Association rule mining.
 11. Big data processing.
 12. Data ethics.

2.2 / MARKING SCHEME

- The course marks are divided between a research project and a data analysis project:
 - The research project will be set in week 1 and is due in week 10.
 - The analysis project will be set in week 4 and is due in week 13.
- For more information, see the module description at bit.ly/2gxFuD2.

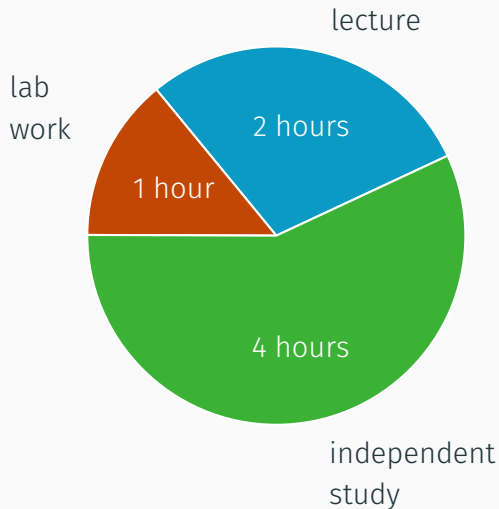
research
project



analysis
project

2.3 / SCHEDULE

- Each week, there will be a two hour lecture and a lab assignment.
- The lab work is *ungraded*, but you will need to understand it in order to do the projects.
- The course material is not hard, but there is a lot to cover — you should take time outside of class to study the material.



- Lectures take place every week, on Wednesday, from 20:00-22:00:
 - Daylight saving time: Irish time zone **changes** from GMT to IST on 26th March.
 - No classes during Easter break: 10th-21st April.
 - Official CIT student calendar: bit.ly/2iUmd46.
- Lecture audio and video are recorded with Adobe Connect and will be uploaded to Blackboard shortly after class.
- Lecture notes are available on GitHub: git.io/vDe5F.
- If you spot broken links, typos or other mistakes, please let me know!



- Except where indicated otherwise, the notes for this course are licensed to you under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International license.
- You are free to share these notes with anyone you like, as long as you:
 1. Give appropriate credit for the material and indicate the license.
 2. Don't use the material for commercial purposes.
 3. Don't redistribute any modified version of the material.
- For more information, see bit.ly/1weyPUN.

- Each week, there will be a lab assignment related to the lecture material.
- The lab assignments are *ungraded*, but you will need to understand them in order to complete the project work.
- Each lab involves the completion of data analysis exercises using Jupyter Notebook.
 - Already installed on your vDesktop environment.
 - Can run it on your personal computer too (more on this later).
- Lab assignments will also be posted to GitHub: [git.io/vDe5F](https://github.com/vDe5F).
- Again, please let me know if you spot broken links, typos or other mistakes.



- Except where indicated otherwise, the code for this course is licensed to you under the GNU General Public License (version 3).
- You are free to use, share and/or change this code however you like, as long as you:
 1. Give appropriate credit for the material and indicate the license.
 2. Make any modifications available under the same license.
- For more information, see bit.ly/2hqEPaZ.

2.8 / WHY NOT R?

	R	Python
Algorithm support	Very large	Large
Difficulty	Moderate	Easy
Useful outside domain	No	Yes

- R and Python are commonly used for data analysis:
 - Both are useful and have their pros and cons.
 - However, Python is far more widely used for general computing.
 - To save having to learn a single purpose language, we will use Python.
- If you don't know Python or haven't used it in some time, Codecademy offer a free introduction course that covers the basics: bit.ly/1DCO2hj.

2.9 / RUNNING LABS AT HOME

- To run the labs on your own computer, you will need to install the following software:

- Python 2.7.x
- IPython
- Jupyter Notebook

- NumPy
- SciPy
- pandas

- matplotlib
- scikit-learn

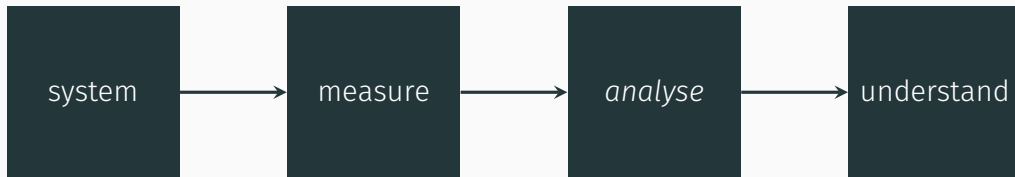
- Alternatively, you can use the *Jupyter Data Science Notebook* Docker image on GitHub: git.io/vDJb0.

- For this module, you are required to complete two projects:
 - The first project is research-based and focuses on distributed file systems.
 - The second project is practical and requires you to analyse a complex data problem and produce your own solution.
- Project work will be managed using Blackboard:
 - Project briefs will be posted to the *Assessment* folder.
 - You can submit your reports using the built-in submission tool.
- Again, please let me know if you spot broken links, typos or other mistakes.

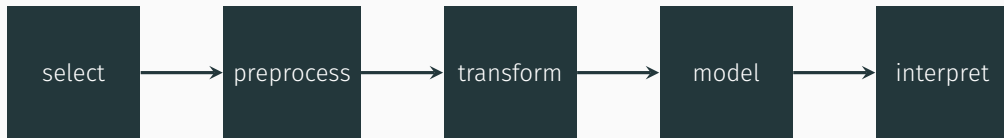
- If you have a question about lecture notes, lab work, project work, or even something not related specifically to course content (e.g. you want to share an interesting link or blog post), post it to the Blackboard forum.
 - Sharing information benefits everyone.
 - Someone else might answer your question faster than I will.
 - URL: blackboard.cit.ie.
- If you have a problem, send me an email.
 - Please don't send emails about course material — let's keep this on Blackboard.
 - Email: donagh.horgan@cit.ie.
- I will try to reply to Blackboard posts and emails within 48 hours, but sometimes it may take longer than this.

Data analysis processes

3.1 / DATA ANALYSIS PROCESSES



- Earlier, we discussed how data analysis is the *process* of manipulating data to discover new information.
- So what exactly is involved in this process? It depends on your goals!
 - Several standardised processes exist, each with their own area of speciality.
 - Some industries/organisations prefer one over the others.
 - However, *all* are driven by the same core principles.



- Knowledge Discovery in Databases (KDD) is an early data analysis paradigm.
 - Its main focus is on the extraction of knowledge from large enterprise databases.
 - While KDD was a popular paradigm initially, it has now largely been superseded by SEMMA and CRISP-DM.
- KDD consists of five stages: selection, preprocessing, transformation, data mining/modelling and interpretation/evaluation.

3.3 / KDD STEPS

1. Selection

- This consists of gathering the data to be analysed.
- Data may be sub-sampled to a more manageable level.

2. Preprocessing

- Fix data quality issues (e.g. missing data, outliers).

3. Transformation

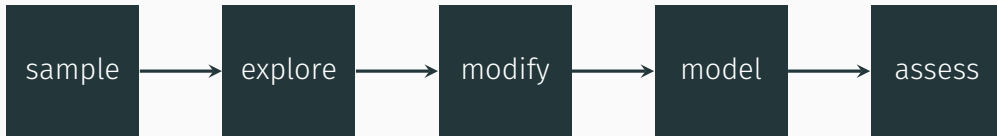
- Select features for further analysis.
- Transform data into forms more suitable for analysis.

4. Data mining/modelling

- Application of a data modelling technique (e.g. a machine learning algorithm).

5. Interpretation/evaluation

- Validate the model to ensure it works (e.g. testing a weather forecast).
- Analyse the results and draw conclusions.



- SEMMA is an alternative process flow model for data mining.
 - It is an evolution of the KDD process, developed by SAS Institute.
 - More commonly used than KDD, but less than CRISP-DM.
- SEMMA is an acronym for the five steps involved: Sample, Explore, Modify, Model, Assess.

3.5 / SEMMA STEPS

1. Sample

- If the total amount of data is large, then take a *representative* sample to speed up later analysis.
- If the total amount of data is not large, then don't sample.

2. Explore

- Explore the data to find patterns or trends.
- Identify data quality problems.
- Form hypotheses based on your findings.

3. Modify

- Select the data to be analysed.
- Fix data quality problems.
- Transform the data, if necessary.

4. Model

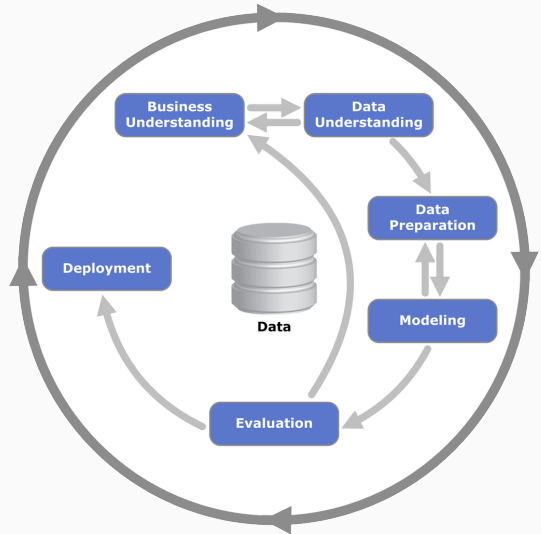
- Apply a data modelling technique to solve the problem.

5. Assess

- Evaluate how useful and reliable the generated model is.
- Estimate how well it will generalise to new situations.

3.6 / CRISP-DM

- CRISP-DM is the CROss Industry Standard Process for Data Mining.
 - Developed by a consortium of industry partners, although mainly associated with IBM.
 - Currently, the most widely used data mining paradigm.
- CRISP-DM consists of six distinct phases, which are repeated if requirements are not met.
- At the end of the evaluation phase, the entire process may be restarted if requirements are not met.



3.7 / CRISP-DM PHASES

1. Business understanding

- Define objectives based on business requirements.
- Translate the objectives into data analysis problems.
- Create a project plan.

2. Data understanding

- Gather the data to be analysed.
- Sample and explore the data to identify any problems and/or form hypotheses.

3. Data preparation

- Select and clean/transform data before modelling.

4. Modelling

- Apply data modelling or pattern matching techniques to solve the problem.
- Optimise algorithm parameters to maximise performance.

5. Evaluation

- Review the business requirements and determine whether the model meets them.

6. Deployment

- Use the model to solve the problem.
- Maintain the solution, if required.

3.8 / COMPARISON

KDD	SEMMA	CRISP-DM
-	-	Business understanding
Selection	Sample	Data understanding
Preprocessing and Transformation	Explore	
	Modify	Data preparation
Data mining/modelling	Model	Modelling
Interpretation/evaluation	Assess	Evaluation
-	-	Deployment

3.9 / COMPARISON

- SEMMA evolved from KDD and is more comprehensive in certain areas:
 - Greater emphasis on exploratory data analysis.
 - All data modification actions are grouped under one step (Modify).
- CRISP-DM also evolved from KDD, but has some distinctions from SEMMA:
 - Focuses on business requirements as well as data analysis.
 - Emphasises the iterative nature of data analysis more strongly.
 - Focuses on post-analysis model deployment and maintenance.
- All three paradigms are useful and, in a given situation, adopting one may be a better choice than adopting another.
- For the remainder of this course, however, we will focus on the core steps shared by all three.

Data structures

- The first stage in any analytics project is to gather the data to be analysed:
 - We might write a SQL query to extract it from a database.
 - We might parse the data from a CSV file.
 - We might extract features from an MP3 file (e.g. Shazam).
- Generally, data is classified as belonging to one of three categories:
 1. Structured data.
 2. Semi-structured data.
 3. Unstructured data.
- Usually, the techniques that we use to gather the data depend on which category it belongs to.

4.2 / STRUCTURED DATA

- The term *structured data* describes data that is highly organised.
- Typically, structured data follows a defined schema or data model (e.g. relational databases), which imposes strict organisational rules:
 - The organisation of data is strictly defined, e.g. the number of columns in a relational database table is generally fixed when the table is created.
 - The types of data are strictly defined, e.g. the data type of a column in a relational database table is generally fixed when the column is created.
- This has both advantages and disadvantages:
 - The strict order facilitates optimisation and predictability, allowing the data to be queried quickly and easily.
 - However, it is also inflexible, leading to major redesign effort in order to accommodate relatively small changes in the data structure.

4.3 / SEMI-STRUCTURED DATA

- The term *semi-structured data* refers to data that is organised in a flexible structure, which generally does not impose restrictions on data organisation and types in as strict a manner as structured data.
 - File formats: CSV, JSON, XML and more.
 - NoSQL databases: MongoDB, InfluxDB, Redis and more.
- Because there is no schema or data model, semi-structured data is generally harder to query than structured data.
 - In a database schema, you might disallow null values in a column.
 - You can't do this in (schema-less) JSON — queries become more complicated.
- However, the flexible structure allows for variations in the structure of the data being stored, and so schema redesign is not required.
 - Adding a new field to a JSON object is easier than rewriting a database schema.

4.4 / SEMI-STRUCTURED DATA: CSV

Title	Author	Year	Price
Slaughterhouse 5	Kurt Vonnegut	1969	9.99
The Bonfire of the Vanities	Tom Wolfe	1987	12.99

- Here's an example of a CSV file containing some data about books.
- CSV stands for *Comma Separated Values*, although the definition is not strict and so the separator can be any character, *e.g.* "&".

```
Title,Author,Year,Price
Slaughterhouse 5,Kurt Vonnegut,1969,9.99
The Bonfire of the Vanities,Tom Wolfe,1987,12.99
```

```
Title| Author| Year| Price
Slaughterhouse 5| Kurt Vonnegut| 1969| 9.99
The Bonfire of the Vanities| Tom Wolfe| 1987| 12.99
```

4.5 / SEMI-STRUCTURED DATA: JSON

Title	Author	Year	Price
Slaughterhouse 5	Kurt Vonnegut	1969	9.99
The Bonfire of the Vanities	Tom Wolfe	1987	12.99

- Here's an example of a JSON file containing the same book data.
- JSON stands for *JavaScript Object Notation*, although it is widely used outside of JavaScript too because of its low overhead and easily read structure.

```
[
  {
    "title": "Slaughterhouse 5",
    "author": "Kurt Vonnegut",
    "year": 1969,
    "price": 9.99
  },
  {
    "title": "The Bonfire of the Vanities",
    "author": "Tom Wolfe",
    "year": 1987,
    "price": 12.99
  }
]
```

4.6 / SEMI-STRUCTURED DATA: XML

Title	Author	Year	Price
Slaughterhouse 5	Kurt Vonnegut	1969	9.99
The Bonfire of the Vanities	Tom Wolfe	1987	12.99

- Here's an example of a XML file containing the same book data.
- XML stands for *eXtensible Markup Language*.
- XML is still widely used, although has been superseded by JSON in many cases.

```
<?xml version="1.0" encoding="UTF-8"?>
<bookshelf>
  <book>
    <title>Slaughterhouse 5</title>
    <author>Kurt Vonnegut</author>
    <year>1969</year>
    <price>9.99</price>
  </book>
  <book>
    <title>The Bonfire of the Vanities</title>
    <author>Tom Wolfe</author>
    <year>1987</year>
    <price>12.99</price>
  </book>
</bookshelf>
```

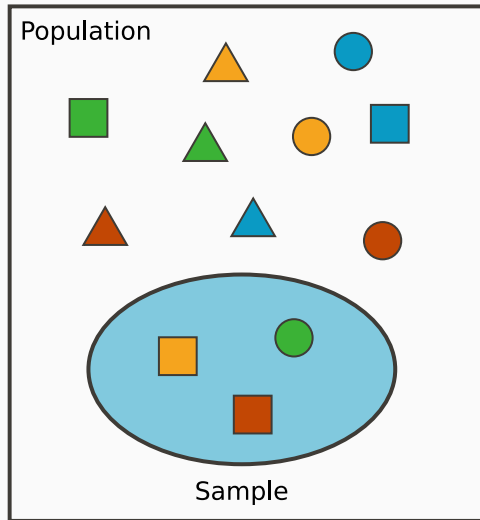

4.7 / UNSTRUCTURED DATA

- The term *unstructured data* refers to data that does not follow any defined structure.
 - Text: documents, emails, tweets.
 - Audio: human speech, music.
 - Images: photographs, video.
- Generally, unstructured data cannot be queried directly and must be preprocessed in order to extract data in a form that we can analyse:
 - Natural language processing extracts linguistic features from text, so that we can analyse its meaning using standard techniques.
 - Fingerprinting allows us to extract summary details from audio, which we can then use to identify the track (e.g. Shazam).
 - Pattern matching enables computers to detect human faces in images and video, allowing us to work with concrete identifiers rather than raw bits.

Data sampling

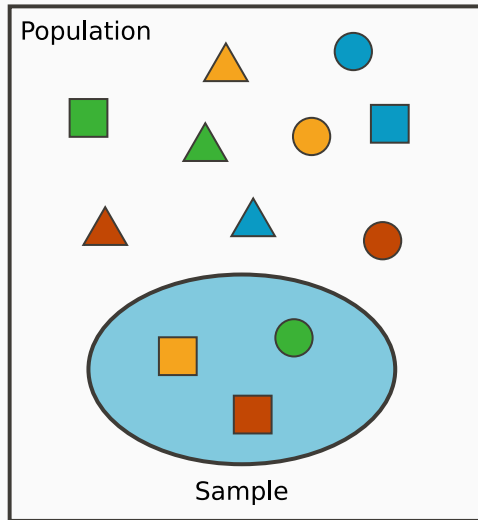
5.1 / POPULATIONS AND SAMPLES

- A statistical *population* is a complete set, representing the entire space of possible outcomes.
- A statistical *sample* is a subset of a population and so represents a number of the possible outcomes, but not the total.



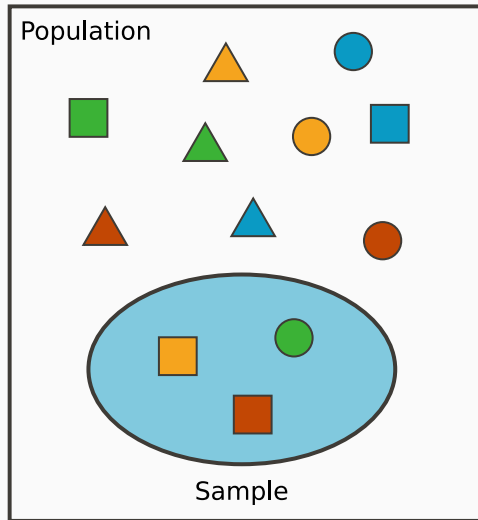
5.2 / POPULATIONS AND SAMPLES

- Populations describe all possible outcomes, while samples describe only a few.
- By random chance, a sample may look extremely different from the population it was drawn from.
 - For instance, in the image on the right, the sample contains no blue shapes or triangles.
- While conclusions based on an analysis of a population will always generalise, conclusions based on a sample may not.



5.3 / POPULATIONS AND SAMPLES

- For example, consider the population of a country:
 - A well chosen sample will reflect the general trends in the greater population: age, gender, income and so on.
 - A poorly chosen sample will not, *e.g.* if our sample contains only teenagers or criminals.



5.4 / WHY SAMPLE?

- In many cases, we want to understand the behaviour of a *large* population:
 - Voters in an election.
 - Customers of a business (e.g. Tesco).
 - Spam vs. non-spam emails.
- But generally we are constrained:
 - The financial cost of polling voters is high.
 - It's difficult to get customers to agree to let us analyse their data.
 - The computational effort involved in processing billions of emails is large.
- If we *sample* the population, then we can avoid these high costs — but we need to be careful to choose our sample well.

- The term *data sampling* refers to a concrete step in the analytics process, the aim of which is to reduce the *quantity* of data to be analysed, while maintaining the *quality*.
 - In KDD, sampling falls under the Selection stage.
 - In SEMMA, there is a step called Sample.
 - In CRISP-DM, sampling is a part of the Data Understanding phase.
- Sampling can be done during the data gathering process:
 - Selecting a subset of rows from a database table.
 - Selecting a subset from a collection of JSON objects.
 - Downsampling images to reduce their size.
- Sampling can also be done after data gathering, *e.g.* by selecting a subset of the gathered data.

- Do we *have* to sample?
 - Sampling isn't mandatory, if the cost of analysis is small.
 - However, if analysing the whole population is not feasible, then we *must* sample.
- If we decide to sample, then we must do so carefully to avoid choosing subsets of the data that do not represent the population.
- The two most common causes of sampling error are:
 1. Lack of randomness.
 2. Small sample size.

5.7 / LACK OF RANDOMNESS

- If we choose a sample in a way that isn't random, then it might not reflect the population that it was drawn from:
 - Polling voters in a single area to estimate national preferences.
 - Issuing discount offers in June based on sales information from December.
 - Deciding that all emails containing the phrase “online pharmacy” are spam.
- By choosing samples randomly, we can minimise the chance that our sample does not resemble the population it was drawn from.

5.8 / SAMPLE SIZE

- In general, larger randomly selected samples tend to reflect their populations better than smaller ones¹.
- However, larger samples are also more costly to analyse.
- Need to be able to choose a “Goldilocks” sample size, *i.e.* one that is large enough to be representative of its population, but small enough to be analysed cheaply.
- Choosing a specific number is often subjective! This is part of the *art* of data science.
- Generally, you should choose as large a sample size as possible, given the resources available to you (*e.g.* time, computing power).

¹Due to law of large numbers — for more information, see bit.ly/2hq7i0s.

Summary

- This week, we covered lots of the fundamentals of data analysis:
 - Formal processes for data analysis: KDD, SEMMA, CRISP-DM.
 - Data structures: structured, semi-structured and unstructured data.
 - Data sampling: populations and samples, randomness, sample size.
- Lab work:
 - This week's lab covers computing statistics and making basic plots in Python.
 - If you have questions, post on Blackboard!
- Next week, we'll look at
 - Statistics.
 - Exploratory data analysis.

1. Module information: bit.ly/2gxFuD2.
2. Lecture notes and labs: git.io/vDe5F.
3. Blackboard discussion forum: blackboard.cit.ie.
4. Codecademy introductory Python course: bit.ly/1DCO2hj.
5. Jupyter documentation: bit.ly/2gPFfe7p.