Data Mining and Machine Learning

Comp 3027J

Dr Catherine Mooney
Assistant Professor

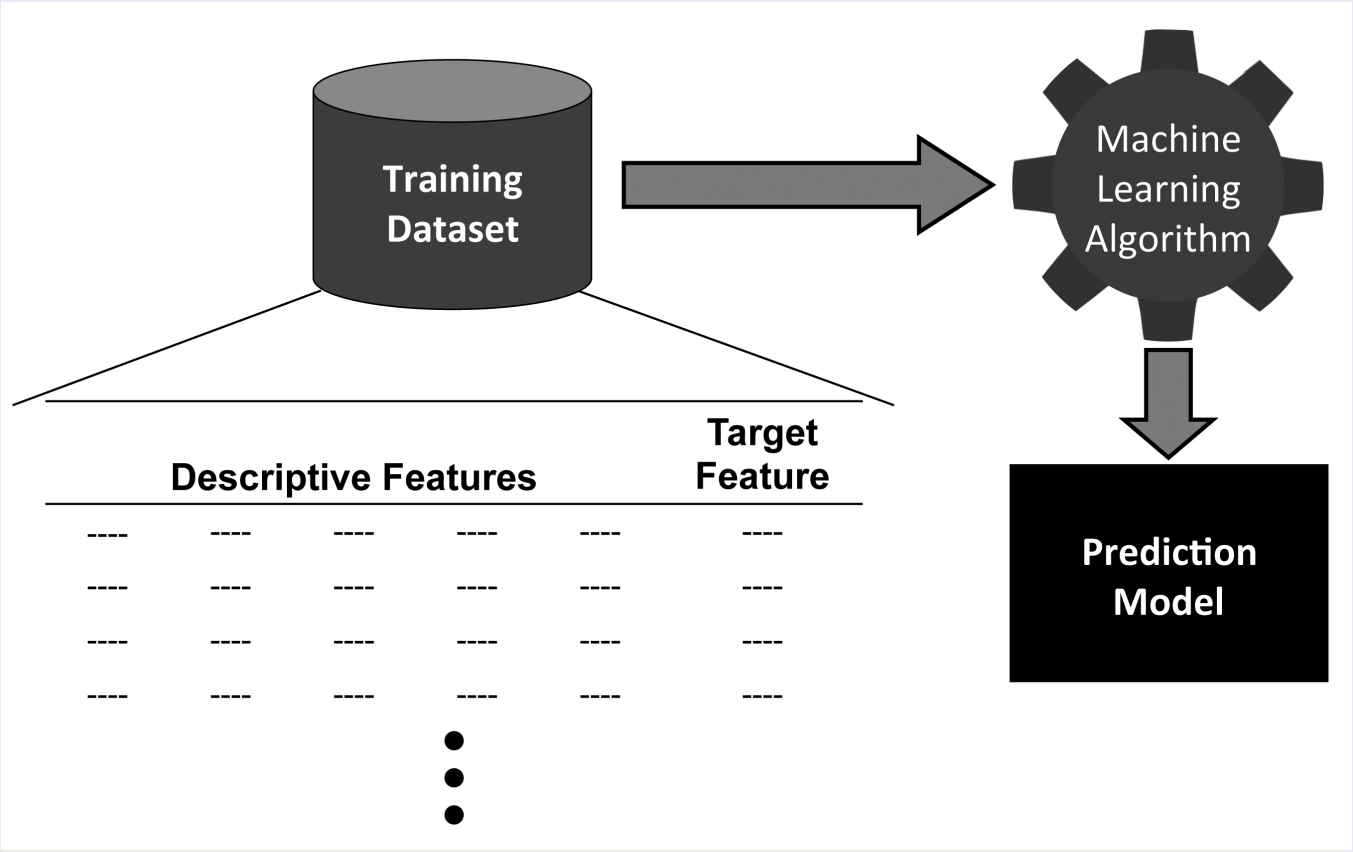
catherine.mooney@ucd.ie

Lectures and Text

- **Core Text:**
  Fundamentals of Machine Learning for Predictive Data Analytics
  By John D. Kelleher, Brian Mac Namee and Aoife D'Arcy
- Last lecture we covered Chapter 7, sections 7.4.4, 7.4.6 and 7.4.7 (Error-based Learning – Logistic Regression, Multinomial Logistic Regression, and Support Vector Machines).
- This week we will cover Chapter 11 – "The Art of Machine Learning for Predictive Data Analytics"

The Art of Machine Learning

Supervised machine learning techniques automatically learn the relationship between a set of **descriptive features** and a **target feature** from a set of historical **instances** (referred to as a **training dataset**) to build a **prediction model**.



We can then use this **prediction model** to make predictions for new instances

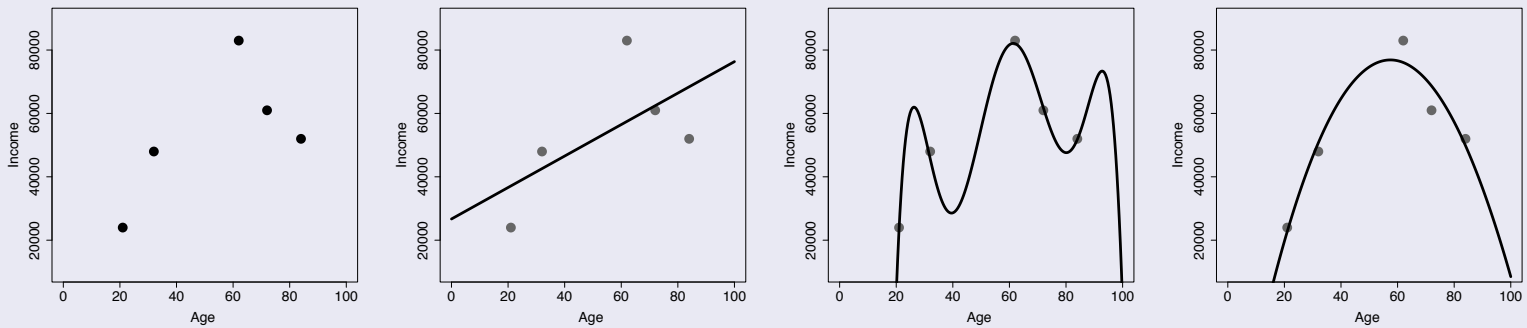
- A prediction model that makes the correct predictions for these queries is said to **generalise** well.
- The goal of machine learning is to find the predictive model that **generalises** best.
- To find the best prediction model, a machine learning algorithm must use some criteria for choosing among the candidate prediction models it considers during its search.

What criteria should we use?

- Lots of different machine learning algorithms.
- Each machine learning algorithm uses different model selection criteria to drive its search for the best **predictive model**.
- The set of assumptions that defines the model selection criteria of a machine learning algorithm is known as the **inductive bias** of the machine learning algorithm.
- It has been shown that there is no particular **inductive bias** that on average is the best one to use.
- **The ability to select the appropriate machine learning algorithm (and hence inductive bias) to use for a given predictive task is one of the core skills that a data analyst must develop!!**

What happens if we choose the wrong inductive bias:

- **underfitting**
- **overfitting**
- Striking the right balance between **model** simplicity and complexity (between underfitting and overfitting) is the hardest part of machine learning.



(a) Dataset (b) Underfitting (c) Overfitting (d) Just right

- Machine Learning $\approx$ inductive learning – learning entails inducing a general rule from a set of specific instances.
  1. A model learned by induction is not guaranteed to be correct – the general rule that is induced from a sample may not be true for all instances in a population.
  2. Learning cannot occur unless the learning process is biased in some way – we need to tell the learning process what types of patterns to look for in the data. This bias is referred to as inductive bias.
  3. The inductive bias of a learning algorithm comprises the set of assumptions that define the search space the algorithm explores and the search process it uses.

We also bias the outcome of a predictive data analytics project in lots of other ways, for example:

- What is the predictive analytics target?
- What descriptive features will we include/exclude?
- How will we handle missing values?
- How will we normalize our features?
- How will we represent continuous features?
- What types of models will we create?
- How will we set the parameters of the learning algorithms?
- What evaluation process will we follow?
- What performance measures will we use?

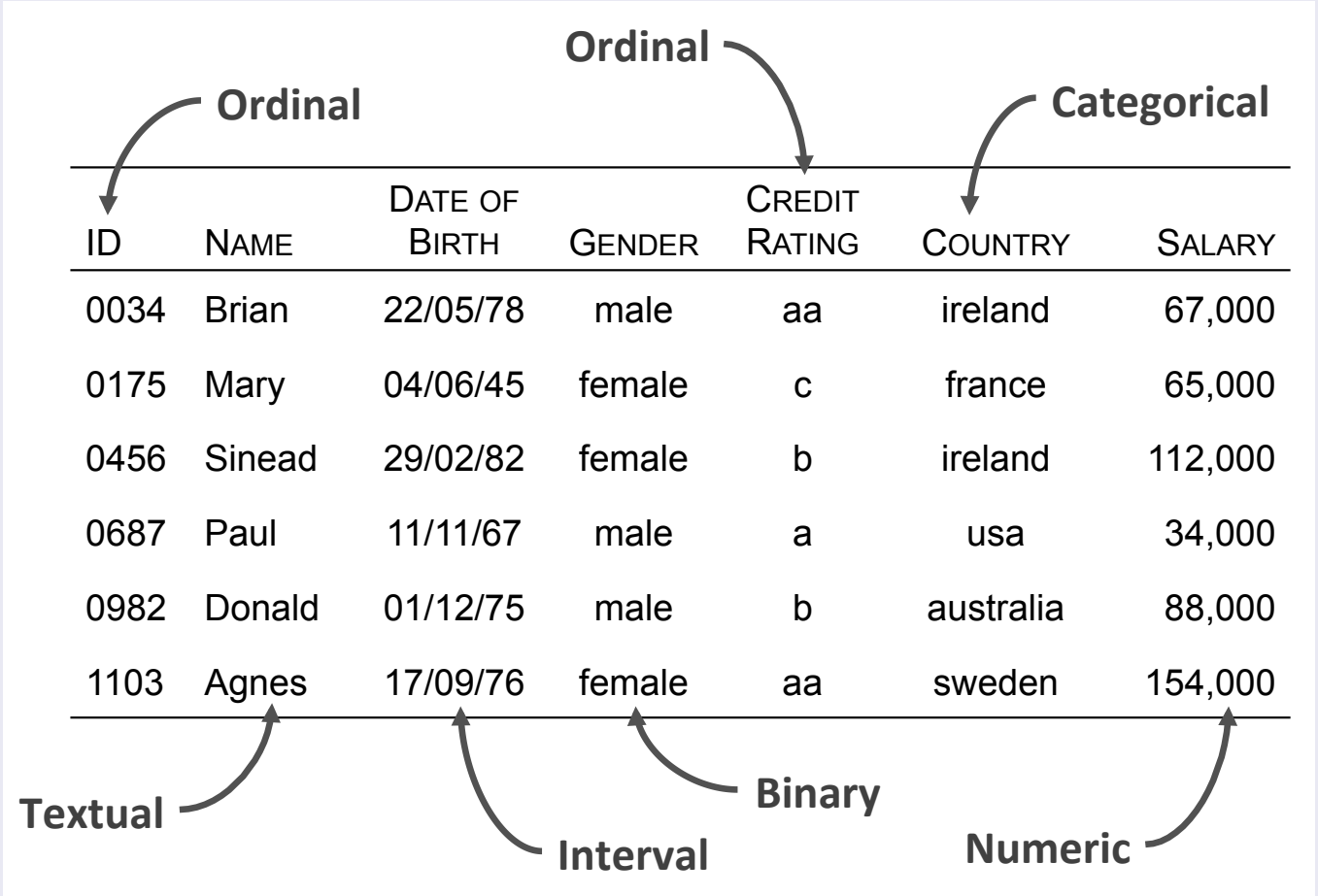
- These questions are relevant when building any prediction model, and the answer to each one introduces a specific bias.
- All of the questions that must be answered to successfully complete a predictive data analytics project can seem overwhelming!
- Often we are forced to answer these questions based on intuition, experience, and experimentation
- This is “the art of machine learning”
- It is also what makes machine learning such a fascinating and rewarding area to work in!!

The Analytics Base Table

- A simple, flat, tabular data structure made up of rows and columns.
- The columns are divided into a set of descriptive features and a single target feature.
- Each row contains a value for each descriptive feature and the target feature.
- Each row represents an instance about which a prediction can be made.

Descriptive Features						Target Feature
----	----	----	----	----	----	----
----	----	----	----	----	----	----
----	----	----	----	----	----	----
----	----	----	----	----	----	----
----	----	----	----	----	----	----

Different Types of Data



Different Types of Data

- **Numeric**: True numeric values that allow arithmetic operations (e.g. price, age)
- **Interval**: Values that allow ordering and subtraction, but do not allow other arithmetic operations (e.g. date, time)
- **Ordinal**: Values that allow ordering but do not permit arithmetic (e.g. size measured as small, medium, or large)
- **Categorical**: A finite set of values that cannot be ordered and allow no arithmetic (e.g. country, product type)
- **Binary**: A set of just two values (e.g. yes/no)
- **Textual**: Free-form, usually short, text data (e.g. name, address)

We often reduce this categorization to just two data types:

- **Continuous** (encompassing the numeric and interval types)
- **Categorical** (encompassing the categorical, ordinal, binary, and textual types)
  - We refer to the set of possible values that a categorical feature can take as the levels of the feature
  - For example, the levels of the CREDIT RATING feature are aa, a, b, c and the levels of the GENDER feature are male, female.

The presence of different types of descriptive and target features can have a big impact on how an algorithm works.

Derived Feature Types

- There are a number of common derived feature types:
  - **Aggregates** – measures defined over a group or period and are usually defined as the count, sum, average, minimum, or maximum of the values within a group.
  - **Flags** – binary features that indicate presence or absence of some characteristic within a dataset (e.g. whether or not a bank account has ever been overdrawn).
  - **Ratios** – capture the relationship between two or more raw data values (e.g. loan-to-value ratios).
  - **Mappings** – convert continuous features into categorical features and are often used to reduce the number of unique values that a model will have to deal with (e.g. salary –> low, medium, and high).
  - Other...

Data Exploration

There are two goals in data exploration:

1. To fully understand the characteristics of the data in the ABT.
   - It is important that for each feature in the ABT, we understand characteristics such as the types of values a feature can take, the ranges into which the values in a feature fall, and how the values in a dataset for a feature are distributed across the range that they can take. We refer to this as **getting to know the data**.
2. To determine whether or not the data in an ABT suffer from any **data quality issues** that could adversely affect the models that we build.
   - Examples of typical data quality issues include an instance that is *missing values* for one or more descriptive features, an instance that has an *extremely high value* for a feature, or an instance that has an *inappropriate level* for a feature.

The data quality report

- A data quality report includes tabular reports that describe the characteristics of each feature in an ABT using standard statistical measures of **central tendency** (mean, mode, and median) and **variation** (standard deviation and percentiles).
- The tabular reports are accompanied by data visualizations:
  - A **histogram** for each continuous feature in an ABT.
  - A **bar plot** for each categorical feature in an ABT.

The data quality report – continuous features

Should include:

- the total number of instances in the ABT
- the percentage of instances in the ABT that are missing a value for each feature
- the cardinality of each feature (cardinality measures the number of distinct values present in the ABT for a feature)
- the minimum, 1st quartile, mean, median, 3rd quartile, maximum, and standard deviation statistics

Feature	Count	% Miss.	Card.	Min.	1^{st} Qrt.	Mean	Median	3^{rd} Qrt.	Max.	Std. Dev.
___	___	___	___	___	___	___	___	___	___	___
___	___	___	___	___	___	___	___	___	___	___
___	___	___	___	___	___	___	___	___	___	___

The data quality report – categorical features

Should include:

- the total number of instances in the ABT
- the percentage of instances in the ABT that are missing a value for each feature
- the cardinality of each feature
- for the two most frequent levels for the feature (the mode and 2nd mode) – the frequency with which these appear (both as raw frequencies and as a proportion of the total number of instances in the dataset)

Feature	Count	% Miss.	Card.	Mode	Mode Freq.	Mode %	2^{nd} Mode	2^{nd} Mode Freq.	2^{nd} Mode %
___	___	___	___	___	___	___	___	___	___
___	___	___	___	___	___	___	___	___	___
___	___	___	___	___	___	___	___	___	___

- A **data quality issue** is loosely defined as anything *unusual* about the data in an ABT.
- The most common data quality issues are:
  - **missing values**
  - **irregular cardinality**
  - **outliers**

- The data quality issues we identify from a data quality report will be of two types:
  - Data quality issues due to **invalid data** – take immediate action to correct them, regenerate the ABT, and recreate the data quality report.
  - Data quality issues due to **valid data** – record any data quality issues due to valid data in a data quality plan so that we remain aware of them and can handle them later if required.

Handling Data Quality Issues

- Handling Missing Values
- Handling Irregular Cardinality
- Handling Outliers

Handling Missing Values

- The **% Miss.** columns in the data quality report highlight the percentage of missing values for each feature
- Why are the values missing?
- Approach 1: Drop any features that have missing value (if % Miss. $>$60%).
- Approach 2: **Imputation**

Handling Missing Values

- **Imputation** replaces missing feature values with a plausible estimated value based on the feature values that are present.
- The most common approach to imputation is to replace missing values for a feature with a measure of the central tendency of that feature.
- It is not recommended to use imputation on features missing in excess of 30% of their values

Handling Outliers

- Outliers are values that lie far away from the central tendency of a feature.
- There are two kinds of outliers that might occur in an ABT: invalid outliers and valid outliers.
- Invalid outliers are values that have been included in a sample through error and are often referred to as noise in the data.
- Valid outliers are correct values that are simply very different from the rest of the values for a feature, for example, a billionaire who has a massive salary compared to everyone else in a sample.

Identifying Outliers

- Examine the minimum and maximum values for each feature and use domain knowledge to determine whether these are plausible values.
- Invalid outliers and should immediately be either corrected, if data sources allow this, or removed and marked as missing values if correction is not possible.
- In some cases we might even remove a complete instance from a dataset based on the presence of an outlier.

- The key outcomes of the **data exploration** process are that the practitioner should
  1. Have *gotten to know* the features within the ABT, especially their central tendencies, variations, and **distributions**.
  2. Have identified any **data quality issues** within the ABT, in particular **missing values**, **irregular cardinality**, and **outliers**.
  3. Have corrected any data quality issues due to **invalid data**.
  4. Have recorded any data quality issues due to **valid data** in a **data quality plan** along with potential handling strategies.
  5. Be confident that enough good quality data exists to continue with a project.

Data Preparation

- Some data preparation techniques change the way data is represented just to make it more compatible with certain machine learning algorithms.
  - Normalization
  - Binning
  - Sampling

Normalization

- Having continuous features that cover very different ranges can cause difficulty for some machine learning algorithms.
- For example, a feature representing customer ages might cover the range [16, 96], whereas a feature representing customer salaries might cover the range [10,000, 100,000].
- **Normalization** techniques can be used to change a continuous feature to fall within a specified range while maintaining the relative differences between the values for the feature.

Binning

- **Binning** involves converting a continuous feature into a categorical feature.
- To perform binning, we define a series of ranges (called **bins**) for the continuous feature that correspond to the levels of the new categorical feature we are creating.
- Two of the more popular ways of defining bins:
  - **equal-width binning**
  - **equal-frequency binning**

Sampling

- Sometimes the dataset we have is so large that we do not use all the data available to us in an ABT and instead **sample** a smaller percentage from the larger dataset.
- We need to be careful when sampling, however, to ensure that the resulting datasets are still representative of the original data and that no unintended **bias** is introduced during this process.
- Common forms of sampling include:
  - **top sampling**
  - **random sampling**
  - **stratified sampling**
  - **under-sampling**
  - **over-sampling**

- **Top sampling** simply selects the top $s\%$ of instances from a dataset to create a sample.
- Top sampling runs a serious risk of introducing bias, however, as the sample will be affected by any ordering of the original dataset.
- **Top sampling should be avoided.**

- **Random sampling** randomly selects a proportion of $s\%$ of the instances from a large dataset to create a smaller set.
- Random sampling is a good choice in most cases as the random nature of the selection of instances should avoid introducing bias.

- **Stratified sampling** is a sampling method that ensures that the relative frequencies of the levels of a specific **stratification feature** are maintained in the sampled dataset.
- To perform stratified sampling:
  - the instances in a dataset are divided into groups (or strata), where each group contains only instances that have a particular level for the stratification feature
  - s% of the instances in each stratum are randomly selected
  - these selections are combined to give an overall sample of s% of the original dataset.

- In contrast to stratified sampling, sometimes we would like a sample to contain different relative frequencies of the levels of a particular feature to the distribution in the original dataset.
- To do this, we can use **under-sampling** or **over-sampling**.

- **Under-sampling** begins by dividing a dataset into groups, where each group contains only instances that have a particular level for the feature to be under-sampled.
- The number of instances in the *smallest* group is the under-sampling target size.
- Each group containing more instances than the smallest one is then randomly sampled by the appropriate percentage to create a subset that is the under-sampling target size.
- These under-sampled groups are then combined to create the overall under-sampled dataset.

- **Over-sampling** addresses the same issue as under-sampling but in the opposite way around.
- After dividing the dataset into groups, the number of instances in the *largest* group becomes the over-sampling target size.
- From each smaller group, we then create a sample containing that number of instances using **random sampling with replacement**.
- These larger samples are combined to form the overall over-sampled dataset.

Choosing a Machine Learning Approach

- A key step in any predictive analytics project is deciding which type of predictive analytics model to use.
- We have looked at three of the most commonly used prediction models and the machine learning algorithms used to build them.
  1. Information-based Learning
  2. Similarity-based Learning
  3. Error-based Learning

The mathematical foundation of these approaches can be described using three simple (but important) equations:

$$H(t, \mathcal{D}) = - \sum_{l \in levels(t)} (P(t = l) \times log_2(P(t = l)))$$

$$dist(\mathbf{q}, \mathbf{d}) = \sqrt{\sum_{i=1}^{m} (\mathbf{q}[i] - \mathbf{d}[i])^2}$$

$$L_2(\mathbb{M}_{\mathbf{w}}, \mathcal{D}) = \frac{1}{2} \sum_{i=1}^{n} (t_i - \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i))^2$$

Claude Shannon's model of entropy (ID3):

$$H(t, \mathcal{D}) = - \sum_{l \in levels(t)} (P(t = l) \times log_2(P(t = l)))$$

Euclidean distance (k nearest neighbor):

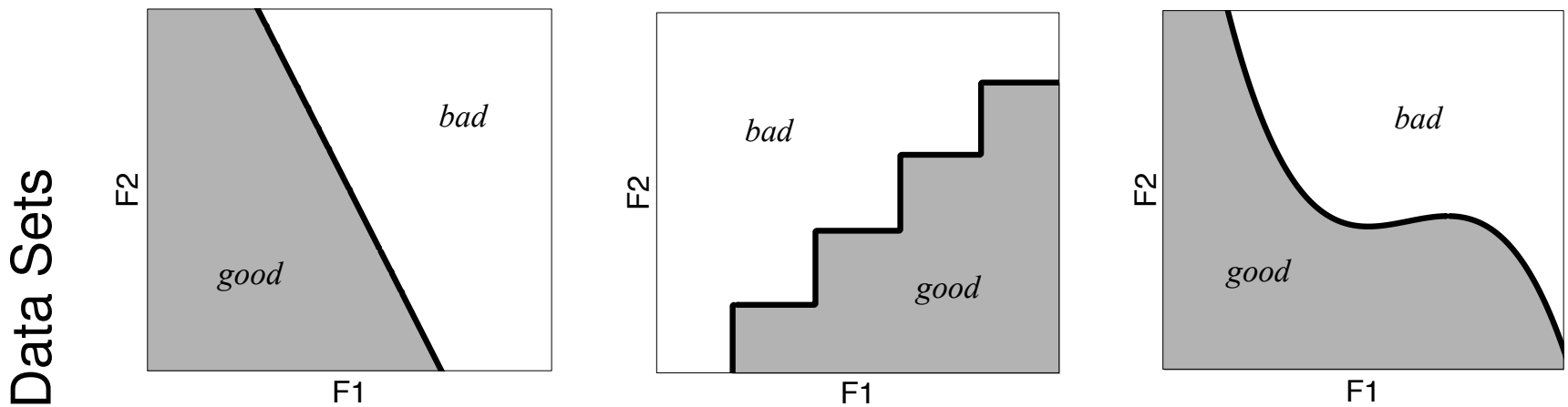
$$dist(\mathbf{q}, \mathbf{d}) = \sqrt{\sum_{i=1}^{m} (\mathbf{q}[i] - \mathbf{d}[i])^2}$$

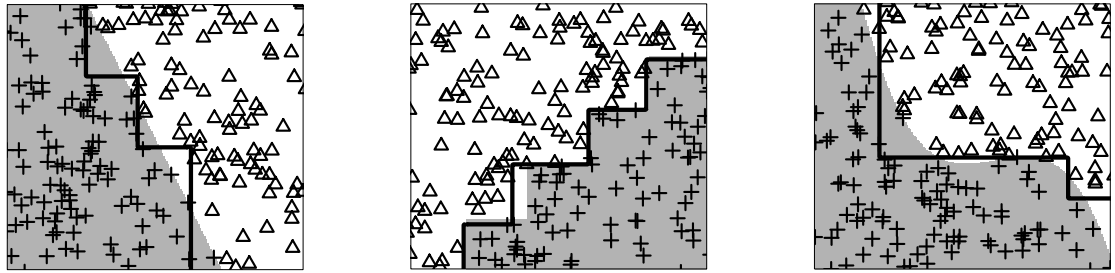
The sum of squared errors (multivariable linear regression with gradient descent):

$$L_2(\mathbb{M}_{\mathbf{w}}, \mathcal{D}) = \frac{1}{2} \sum_{i=1}^{n} (t_i - \mathbb{M}_{\mathbf{w}}(\mathbf{d}_i))^2$$

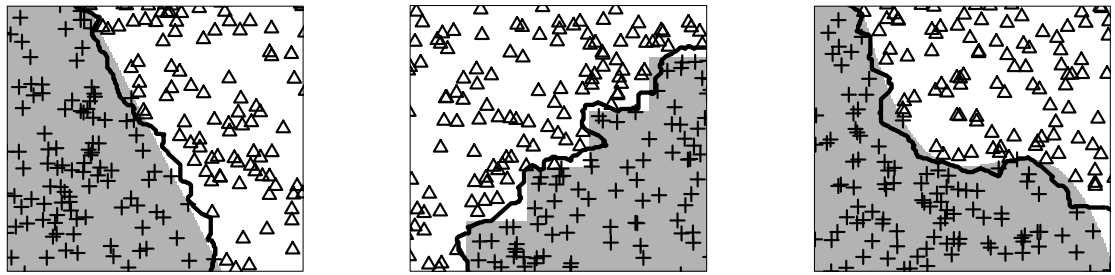
- How do we decide which machine learning approach to use?
- There is not one best approach that always outperforms the others.
- Each of the approaches we have covered induces distinct types of prediction models with different strengths and weaknesses.
- Each algorithm encodes a distinct set of assumptions (the inductive bias of the learning algorithm).
- A set of assumptions that are appropriate in one domain may not be appropriate in another domain.
- We can see the assumptions encoded in each algorithm reflected in the distinctive characteristics of the decision boundaries that they learn for categorical prediction tasks.

Three artificial datasets – Each of the images shows a feature space defined by two continuous descriptive features, F1 and F2, partitioned into good and bad regions by three different, artificially created decision boundaries.

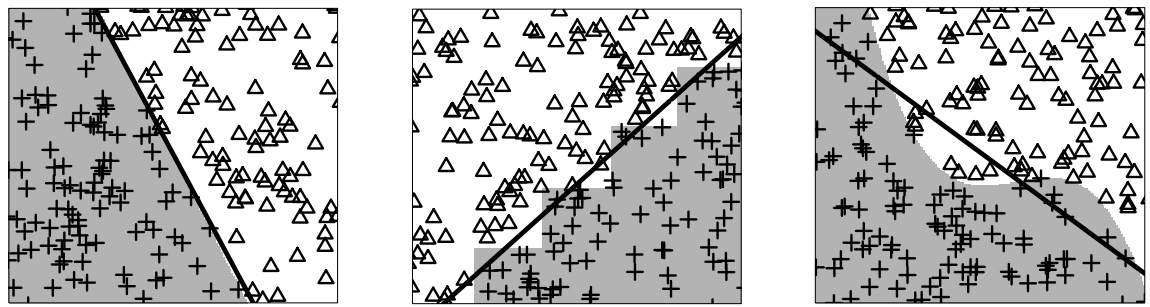




A decision tree (without pruning)



A *k*-NN model (with k = 3 and using majority voting)



A logistic regression model (using a simple linear model)

The training data instances are shown as triangles for good and crosses for bad, the decision boundaries learned by each algorithm are represented by thick black lines, and the underlying actual decision boundaries are shown by the background shading.

- The characteristic appearance of the decision boundaries is related to the representations used within the models and the inductive biases that the algorithms used to build them encode
- The decision boundaries associated with decision trees have a characteristic stepped appearance because of the way feature values are split in a decision tree
- The decision boundaries associated with k-NN models are noticeably jagged because of their local focus

- Some of the models do a better job of representing the underlying decision boundaries than others
- The decision boundary learned by the logistic regression model best matches the underlying decision boundary for the dataset in the first column
- The decision tree model seems most appropriate for the dataset in the second column
- The k-NN model appears best for the dataset in the third column

- Typically choose a number of different approaches and run experiments to evaluate which best suits the particular project.
- There are two questions to consider in the selection of a set of initial approaches:
  1. Does a machine learning approach match the requirements of the project?
  2. Is the approach suitable for the type of prediction we want to make and the types of descriptive features we are using?

Project Requirements

- Accuracy – In many cases the primary requirement of a project is to create an accurate prediction model
- A model must be accurate, but it must also meet the other requirements:
  - Prediction speed
  - Capacity for retraining
  - Interpretability

Prediction speed:

- Logistic regression models are very fast at making predictions as all that is involved is calculating the regression equation and performing a thresholding operation.
- *k* nearest neighbor models are very slow to make predictions as they must perform a comparison of a query instance to every instance in a, typically large, training set.
- The time differences arising from these different computational loads can have an influence on model selection.
- For example, in a real-time credit card fraud prediction system, it may be required that a model perform thousands of predictions per second.

Capacity for retraining:

- Concept drift is a phenomenon that occurs when the relationship between the target feature and the descriptive features changes over time
- Approaches can be used to monitor the performance of a model so as to flag the occurrence of concept drift and indicate if a model has gone stale.
- When this occurs, the model needs to be changed in some way to adapt to the new scenario.
- For some modeling approaches this is quite easy, while for others it is almost impossible to adapt a model, and the only option is to discard the current model and train a new one using an updated dataset.
- *k* nearest neighbor models are good examples of the former type, while decision trees and regression models are good examples of the latter.

Interpretability:

- In many instances a business will not be happy to simply accept the predictions made by a model and incorporate these into their decision making.
- They will require the predictions made by a model to be explained and justified.
- Different models offer different levels of explanation capacity and therefore different levels of interpretability.
- Decision trees and linear regression models are very easily interpreted, while support vector machines and ensembles are almost entirely uninterpretable (because of this, they are often referred to as a black box).

Data Considerations

- continuous target $\rightarrow$ error based models
- categorical target $\rightarrow$ information and probability models
- continuous descriptive features $\rightarrow$ (+cat. target) similarity based models / (+cont. target) error based models
- categorical descriptive features $\rightarrow$ information and probability models
- lots of descriptive features (curse of dimensionality) $\rightarrow$ feature selection

Decision trees – Advantages

- The main advantage of decision tree models is that they are interpretable.
- Decision tree models can be used for datasets that contain both categorical and continuous descriptive features.
- They have the ability to model the interactions between descriptive features.

Decision trees – Disadvantages

- Although decision trees can handle both categorical and continuous features, they tend to become quite large when dealing with continuous descriptive features. This can result in trees becoming difficult to interpret.
- If dealing with purely continuous data, other prediction models may be more appropriate, for example, the error-based models
- Decision trees also have difficulty with domains that have a large number of descriptive features, particularly if the number of instances in the training dataset is small.
- They are eager learners – they are not suitable for modeling concepts that change over time (concept drift), because they will need to be retrained.

Similarity-based models (nearest neighbor algorithm)

- Similarity-based models attempt to mimic a way of reasoning that is natural to humans makes them easy to interpret and understand
- The nearest neighbor algorithm delays abstracting from the data until it is asked to make a prediction (lazy learner), as the number of instances becomes large, the model will become slower because it has more instances to check when defining the neighborhood. A nearest neighbor model may not be appropriate in domains where speed of prediction is a priority
- An advantage of the lazy learning strategy, however, is that similarity-based approaches are robust to concept drift. A nearest neighbor algorithm can be updated without retraining as it is relatively straightforward to update the model when new labeled instances become available

Error-based models – Advantages

- One of the advantages of using a logistic regression model is that it works well for datasets in which the instances with target features set to different levels overlap with each other in the feature space.
- The main advantages of SVM models are that they are robust to overfitting and perform well for very high-dimensional problems.

Error-based models – Disadvantages

- The logistic regression approach (and the SVM approach), in its basic form, can only handle categorical target features with two levels.
- In order to handle categorical target features with more than two levels, that is multinomial prediction problems, we need to use a one-versus-all approach in which multiple models are trained.
- This is one reason that other approaches are often favored over logistic regression (and the SVM approach) for predicting categorical targets with many levels.
- Support vector machines are not very interpretable, and, especially when kernel functions are used, it is very difficult to understand why a particular prediction has been made.

Summary

In summary, ensembles and support vector machines are, in general, more powerful machine learning approaches. However, these approaches are more complex, take a longer time to train, leverage more inductive bias, and are harder to interpret than the simpler approaches (e.g. k-NN). Furthermore, the selection of a machine learning approach also depends on the aspects of an application scenario described above (speed, capacity for retraining, interpretability), and often, these factors are a bigger driver for the selection of a machine learning approach than prediction accuracy.

Your Next Steps

- The easy part of a predictive data analytics project is building the models.
- What makes predictive data analytics difficult, but also fascinating, is figuring out how to answer all the questions that surround the modelling phase of a project.
- Intuition, experience, and experimentation!

Key tasks for an analyst

- become situationally fluent so that we can converse with experts in the application domain;
- explore the data to understand it correctly;
- spend time cleaning the data;
- think hard about the best ways to represent features;
- spend time designing the evaluation process correctly.

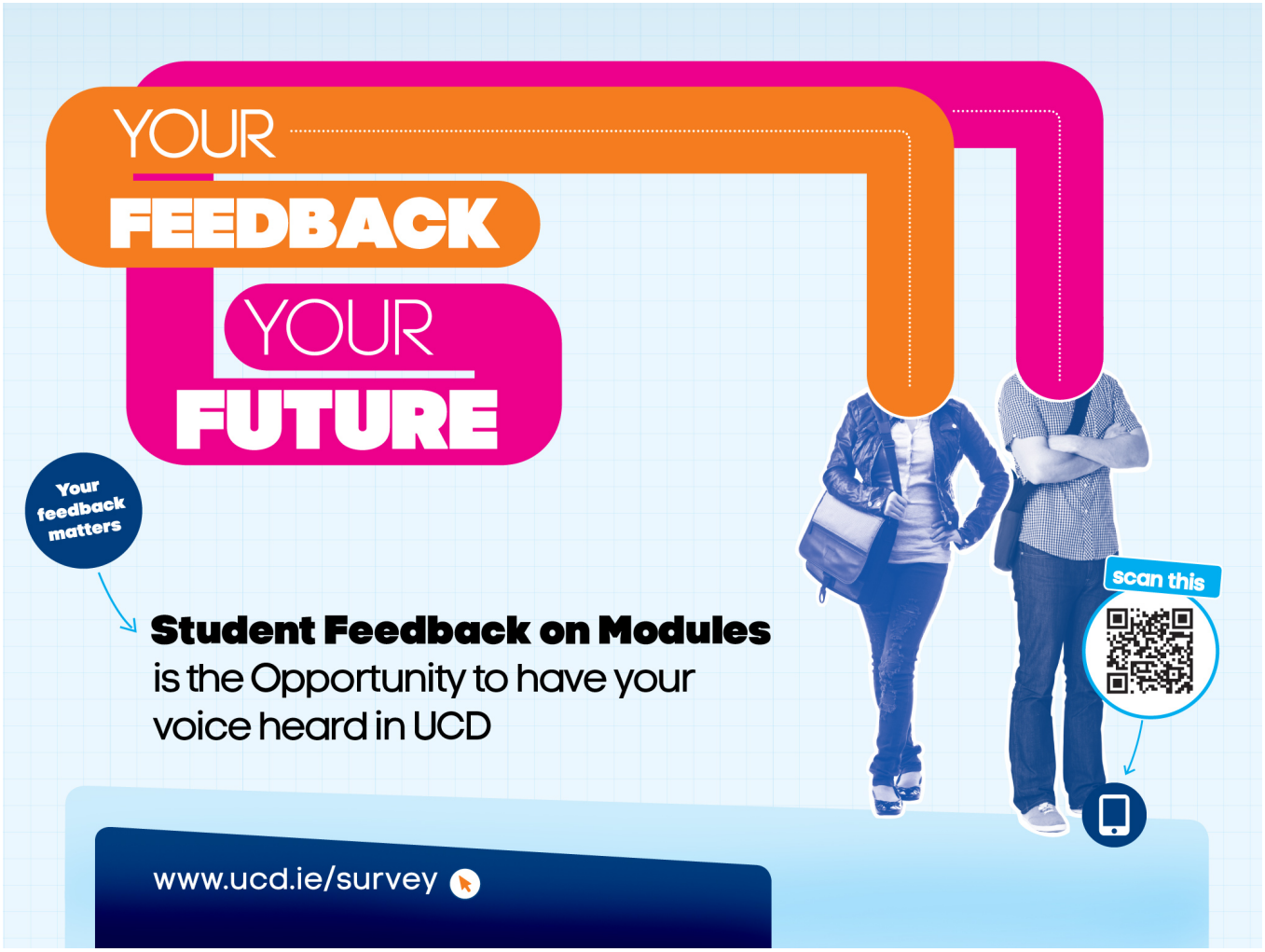
- Machine learning is a huge topic.
- There are lots of topics we haven't covered: **deep learning**, **graphical models**, **multi-label classification**, **association mining**, **clustering**, . . .
- But, I hope this course has given you the knowledge and skills that you will need to explore machine learning yourself.

Exam Instructions

- Answer Question 1 (worth maximum of 40 points)
- Answer any other two questions (worth maximum of 30 points each)
- Total marks available is 100
- Calculators are allowed

Questions?

Feedback


YOUR
FEEDBACK
YOUR
FUTURE
Your feedback matters
Student Feedback on Modules
is the Opportunity to have your
voice heard in UCD
scan this
www.ucd.ie/survey

Provide feedback on this module by completing the survey at www.ucd.ie/survey

Module Feedback for COMP3027J - Data Mining & Machine Learning

This should take approximately 3 minutes to complete. Click on the module code above to see details of this module. Your responses will remain anonymous and the results will not be made available to your lecturer or Head of School until after this semester's examination results have been issued.

Please complete all questions.

1	**I have a better understanding of the subject after completing this module.**	○Strongly Agree ○Agree ○Not Sure ○Disagree ○Strongly Disagree
2	**The assessments to date were relevant to the work of the module.**	○Strongly Agree ○Agree ○Not Sure ○Disagree ○Strongly Disagree
3	**I achieved the learning outcomes for this module.**	○Strongly Agree ○Agree ○Not Sure ○Disagree ○Strongly Disagree
4	**The teaching on this module supported my learning.**	○Strongly Agree ○Agree ○Not Sure ○Disagree ○Strongly Disagree
5	**Overall I am satisfied with this module.**	○Strongly Agree ○Agree ○Not Sure ○Disagree ○Strongly Disagree

Your comments are very important and valued by lecturers. Please ensure that neither the language nor content will cause personal offense to any individual lecturer.

6 **Identify up to three aspects of the module that most helped your learning**

7 **Suggest up to three changes to the module that would enhance your learning.**

Thank you for your feedback. Click SUBMIT below if you are happy with your responses.

Module Learning Outcomes for Comp 3027J: Data Mining and Machine Learning

On completion of this module, you will be able to:

- Distinguish between the different categories of data mining and machine learning algorithms
- Identify a suitable data mining/machine learning algorithm for a given application or task
- Run and evaluate the performance of a range of algorithms on real datasets using a standard machine learning toolkit