COMP30027 Machine Learning
Basics of Machine Learning

Semester 1, 2019

Jeremy Nicholson & Tim Baldwin & Karin Verspoor

Lecture Outline

1. Basics of ML: Instances, Attributes and Learning Paradigms

2. ML in the Wild

Terminology

- The input to a machine learning system consists of:
  - **Instances**: the individual, independent examples of a concept

    *also known as **exemplars***
  - **Attributes**: measuring aspects of an instance

    *also known as **features***
  - **Concepts**: things that we aim to learn

    *generally in the form of **labels** or **classes***

Source(s): Tan et al. [2006, pp19–35]

Example: `weather.nominal` Dataset

Outlook	Temperature	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
⋮	⋮	⋮	⋮	⋮

Example: `weather.nominal` Dataset

Outlook	Temperature	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
⋮	⋮	⋮	⋮	⋮

INSTANCE$_1$

INSTANCE$_2$

Example: `weather.nominal` Dataset

Outlook	Temperature	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
⋮	⋮	⋮	⋮	⋮

ATTRIBUTE_1

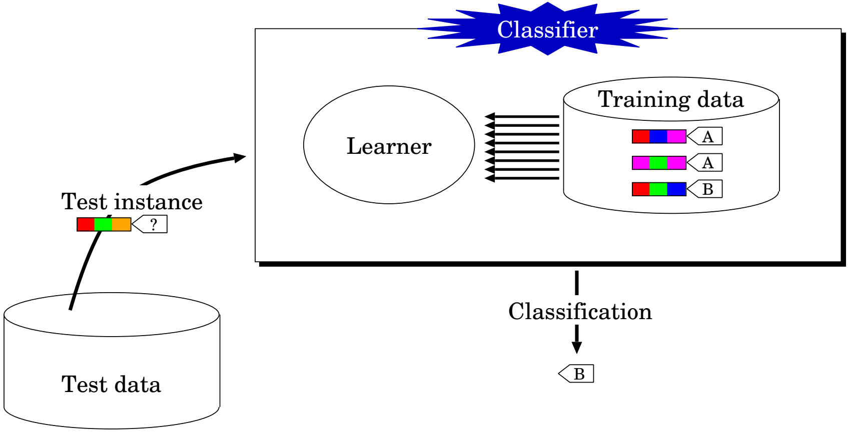
ATTRIBUTE_2

What's a Concept?

- Styles of "concepts" that we aim to learn:
  - Classification learning:
    predicting a discrete class
  - Clustering:
    grouping similar instances into clusters
  - Regression:
    predicting a numeric quantity
  - Association learning:
    detecting associations between attribute values

Classification Learning

- Scheme is provided with actual outcome or **class**
- The learning algorithm is provided with a set of classified **training data**
- Measure success on "held-out" data for which class labels are known (**test data**)
- Classification learning is **supervised**


Classifier
Learner
Training data
A
A
B
Test instance
?
Test data
Classification
B

Example Predictions for `weather.nominal`

Outlook	Temperature	Humidity	Windy	Actual	Classified
sunny	hot	high	FALSE	no	
sunny	hot	high	TRUE	no	
overcast	hot	high	FALSE	yes	
rainy	mild	high	FALSE	yes	
rainy	cool	normal	FALSE	yes	
rainy	cool	normal	TRUE	no	
overcast	cool	normal	TRUE	yes	
sunny	mild	high	FALSE	no	
sunny	cool	normal	FALSE	yes	
rainy	mild	normal	FALSE	yes	
sunny	mild	normal	TRUE	yes	no
overcast	mild	high	TRUE	yes	yes
overcast	hot	normal	FALSE	yes	yes
rainy	mild	high	TRUE	no	yes

Clustering

- Finding groups of items that are similar
- Clustering is **unsupervised** — the learner operates without a set of labelled training data
- The class of an example is not known ... or at least, not given to the classifier
- Success often measured subjectively; evaluation is problematic

Clustering over `weather.nominal`

Outlook	Temperature	Humidity	Windy	Play
sunny	hot	high	FALSE	~~no~~
sunny	hot	high	TRUE	~~no~~
overcast	hot	high	FALSE	~~yes~~
rainy	mild	high	FALSE	~~yes~~
rainy	cool	normal	FALSE	~~yes~~
rainy	cool	normal	TRUE	~~no~~
⋮	⋮	⋮	⋮	~~⋮~~

A Word on Supervision

- **Supervised** methods have prior knowledge of a closed set of classes and set out to discover and categorise new instances according to those classes
- **Unsupervised** methods:
  - dynamically discover the "classes" (implicitly derived from grouping of instances) in the process of categorising the instances [STRONG] ... *OR* ...
  - categorise instances as certain labels without the aid of pre-classified data [WEAK]

Regression

- Classification learning, but class is continuous (**numeric prediction**)
- Learning is supervised
- Why is this distinct from Classification?
  - In Classification, we can exhaustively enumerate all possible labels for a given instance; a correct prediction entails mapping an instance to the label which is truly correct
  - In Regression, infinitely many labels are possible, we cannot conceivably enumerate them; a "correct" prediction is when the numeric value is acceptably close to the true value

Example Predictions for weather

Outlook	Humidity	Windy	Play	Actual Temp	Classified Temp
sunny	85	FALSE	no	85	
sunny	90	TRUE	no	80	
overcast	86	FALSE	yes	83	
rainy	96	FALSE	yes	70	
rainy	80	FALSE	yes	68	
rainy	70	TRUE	no	65	
overcast	65	TRUE	yes	64	
sunny	95	FALSE	no	72	
sunny	70	FALSE	yes	69	
rainy	80	FALSE	yes	75	
sunny	70	TRUE	yes	75	68.8
overcast	90	TRUE	yes	72	76.2
overcast	75	FALSE	yes	81	70.6
rainy	91	TRUE	no	71	76.5

Association Learning

- Detect "useful" patterns, associations, correlations, or causal structures among sets of items or objects in dataset
- "Good" pattern: combination of attribute values where the presence of one (or more) value(s) suggests that one (or more) other value(s) will also be attested for numerous instances in the dataset
- Any kind of structure is considered interesting, and no *a priori* sense of what we hope to predict; unsupervised; evaluation is problematic
- Potentially many, many association rules

Full `weather.nominal` Dataset

Outlook	Temperature	Humidity	Windy	Play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Top-10 Association Rules for weather.nominal

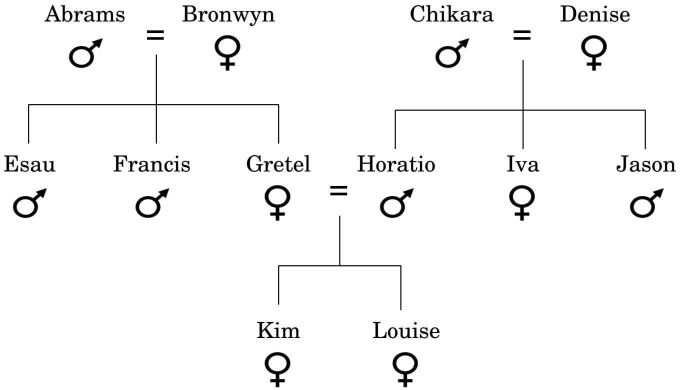
```
# java weka.associations.Apriori -t data/weather.nominal.arff
```

```
 1. humidity=normal windy=FALSE ==> play=yes
 2. temperature=cool ==> humidity=normal
 3. outlook=overcast ==> play=yes
 4. temperature=cool play=yes ==> humidity=normal
 5. outlook=rainy windy=FALSE ==> play=yes
 6. outlook=rainy play=yes ==> windy=FALSE
 7. outlook=sunny humidity=high ==> play=no
 8. outlook=sunny play=no ==> humidity=high
 9. temperature=cool windy=FALSE ==> humidity=normal play=yes
10. temperature=cool humidity=normal windy=FALSE ==> play=yes
```

Instance Topology

- Instances characterised as "feature vectors", defined by a predetermined set of attributes
- Input to learning scheme: set of instances/dataset
    - Flat file representation
    - No relationships between objects
    - No explicit relationship between attributes

A Family Tree



Family Tree Represented as a Table

Name	Gender	Parent1	Parent2
Abrams	Male	?	?
Bronwyn	Female	?	?
Chikara	Male	?	?
Denise	Female	?	?
Esau	Male	Abrams	Bronwyn
Francis	Male	Abrams	Bronwyn
Gretel	Female	Abrams	Bronwyn
Horatio	Male	Chikara	Denise
Iva	Female	Chikara	Denise
Jason	Male	Chikara	Denise
Kim	Female	Gretel	Horatio
Louise	Female	Gretel	Horatio

The `sister` Relation

X	Y	$\texttt{sister}(X, Y)$
Abrams	Abrams	No
Abrams	Bronwyn	No
Abrams	Chikara	No
⋮	⋮	⋮
Esau	Francis	No
Esau	Gretel	Yes
Esau	Horatio	No
⋮	⋮	⋮
Gretel	Denise	No
Gretel	Esau	No
⋮	⋮	⋮

X	Y	$\texttt{sister}(X, Y)$
Horatio	Iva	Yes
Horatio	Jason	No
Horatio	Kim	No
⋮	⋮	⋮
Jason	Iva	Yes
Jason	Jason	No
Jason	Kim	No
⋮	⋮	⋮
Kim	Kim	No
Kim	Louise	Yes
⋮	⋮	⋮

A Full Representation in One Table I

X				Y				sister
Name	**Gender**	**Parent1**	***Parent2***	**Name**	**Gender**	**Parent1**	***Parent2***	(X, Y)
Abrams	Male	?	?	Abrams	Male	?	?	No
Abrams	Male	?	?	Bronwyn	Female	?	?	No
Jason	Male	Chikara	Denise	Iva	Female	Chikara	Denise	Yes
Iva	Female	Chikara	Denise	Jason	Male	Chikara	Denise	No
Esau	Male	Abrams	Bronwyn	Gretel	Female	Abrams	Bronwyn	Yes
Esau	Male	Abrams	Bronwyn	Horatio	Male	Abrams	Bronwyn	No
Gretel	Female	Abrams	Bronwyn	Denise	Female	?	?	No
Kim	Female	Gretel	Horatio	Louise	Female	Gretel	Horatio	Yes
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

- What we would like to be able to extract:

  IF Y.Gender = Female AND (X.Parent1 = Y.Parent1 AND X.Parent2 = Y.Parent2) OR (X.Parent1 = Y.Parent2 AND X.Parent2 = Y.Parent1) AND X $\neq$ Y THEN sister(X,Y) = yes

A Full Representation in One Table II

- What the supervised classifiers we will look at **actually** generate:

```
IF Y.Gender = Female AND X.Parent1 =
Gretel AND
Y.Parent1 = Gretel THEN sister(X,Y) = yes

IF X.Gender = Male AND Y.Name = Gretel
THEN sister(X,Y) = yes
```

- How can we convert the table into a "classifier-friendly" format?

A Classifier-friendly Representation

X.Gender	Y.Gender	X.Parent1 = Y.Parent1	X.Parent2 = Y.Parent2	
Male	Female	Yes	Yes	Yes
Female	Female	Yes	Yes	Yes
Male	Female	No	No	No
Male	Male	Yes	Yes	No
Female	Male	Yes	Yes	No
⋮	⋮	⋮	⋮	⋮

- The importance of **feature engineering**

What's in an Attribute?

- Each instance is described by a fixed feature vector
- Possible attribute types (levels of measurement):

  nominal
  ordinal
  continuous

Nominal Quantities

- Values are distinct symbols (e.g. {`sunny`,`overcast`,`rainy`})
  - values themselves serve only as labels or names
- Also called **categorical**, or **discrete** (NB. "discrete" implies an order which tends not to exist)
- Special case: dichotomy ("Boolean" attribute)
- No relation is implied among nominal values (no ordering or distance measure), and only equality tests can be performed

Ordinal Quantities

- An explicit order is imposed on the values (e.g. $\{\texttt{hot},\texttt{mild},\texttt{cool}\}$ where $\texttt{hot} > \texttt{mild} > \texttt{cool}$)
- No distance between values defined and addition and subtraction don't make sense
- Example rule: $\texttt{temperature} < \texttt{hot} \rightarrow \texttt{play} = \texttt{yes}$
- Distinction between nominal and ordinal not always clear (e.g. `outlook`)

Continuous Quantities

- Continuous quantities are real-valued attributes with a well-defined zero point and no explicit upper bound
- Example: attribute `distance`
  - Distance between an object and itself is zero
- All mathematical operations are allowed (of which addition, subtraction, scalar multiplication are most salient, but other operations are relevant in some contexts)

Lecture Outline

1. Basics of ML: Instances, Attributes and Learning Paradigms

2. ML in the Wild

Attribute Types Used in Practice

- Many data schemes/learners accommodate nominal attributes (perhaps with some awkwardness), and they are very commonly observed
- Many support continuous attributes, and they are commonly observed
- Some support ordinal attributes, which are occasionally observed (but often treated as one of the other types)

Transforming attributes to Boolean is one commonly-used work-around (more in later weeks)

Preparing the Input

- Problem: different data sources (e.g. sales department, customer billing department, ...)
  - Differences: styles of record keeping, conventions, time periods, data aggregation, primary keys, errors
  - Data must be assembled, integrated, cleaned up
  - Data warehouse: consistent point of access
- External data/storage may be required
- Critical: type and level of data aggregation

Sample Representation: ARFF

```
@relation weather
@attribute outlook {sunny, overcast, rainy}
@attribute temperature real
@attribute humidity real
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}
@data
sunny,85,85,FALSE,no
sunny,80,90,TRUE,no
overcast,83,86,FALSE,yes
rainy,70,96,FALSE,yes

      ⋮
```

Missing Values

- The number of attributes may vary in practice
  - missing values
  - inter-dependent attributes
- Frequently indicated by out-of-range entries
  - Types: unknown, unrecorded, irrelevant
  - Reasons:
    - malfunctioning equipment
    - changes in experimental design
    - collation of different datasets
    - measurement not possible
- Missing value may have significance in itself (e.g. missing test in a medical examination)
- Most schemes assume that is not the case →`missing` may need to be coded discretely

Inaccurate Values

- Cause: a given data mining application is often not known at the time logging is set up
- Result: errors and omissions that don't affect original purpose of data (e.g. age of customer)
- Typographical errors in nominal attributes values need to be checked for consistency
- Typographical and measurement errors in numeric attributes $\rightarrow$outliers need to be identified
- Errors may be deliberate (e.g. wrong post codes)

Getting to Know the Data

- Simple visualization tools are very useful
  - Nominal attributes: histograms (distribution consistent with background knowledge?)
  - Numeric attributes: scatter plots (any obvious outliers?)
- 2-D and 3-D plots show dependencies
- Need to consult domain experts
- Too much data to inspect? Take a sample!
- You can never know your data **too** well

Machine Learning and Ethics

discriminate:

1. To make distinctions.

For example, in supervised ML, for a given instance, we might try to discriminate between the various possible classes.

Source(s): Wiktionary contributors [2019]

Machine Learning and Ethics

discriminate:

2. To make decisions based on prejudice.

Digital computers have no volition, and consequently cannot be prejudiced.

However, the data may contain information which leads to an application where the ensuing behaviour is prejudicial, intentionally or otherwise.

Source(s): Wiktionary contributors [2019]

Machine Learning and Ethics I

ML has the potential to *discriminate* [def 2.] people

- some uses of data are unethical, some plainly illegal
  - race & sex in medical applications: OK
  - race & sex in loan applications: unethical
  - race & sex in student applications: ??? (affirmative action vs. racial/sex discrimination)
- legal frameworks are still being defined

Machine Learning and Ethics II

Not everything that *can* be done, *should* be done

- attributes in the data can encode information in an indirect way
  - For example, home address and occupation can be used (perhaps with other seemingly-banal data) to infer age and social standing of an individual
- potential legal exposure due to implicit "knowledge" used by a classifier
- just because you didn't realise doesn't mean that you shouldn't have realised, or at least, made reasonable efforts to check

Questions to Ask

- Who is permitted to access the data?
- For what purpose was the data collected?
- What kinds of conclusions are legitimate?
- If our conclusions defy common sense, are there confounding factors?
    - car insurance & young male drivers?
    - car loans & owners of red cars?

Summary

- What are instances, attributes and concepts?
- What styles of learning are there and what are their similarities/differences?
- Define supervised and unsupervised learning
- What are the basic attribute types?

References I

Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to Data Mining*. Addison Wesley, 2006.

Wiktionary contributors. discriminate, in *Wiktionary, the free dictionary*. `https://en.wiktionary.org/w/index.php?title=discriminate&oldid=49576494`, 2019.