CSCI 556 Data Analysis & Visualization

Input: concepts, instances, attributes

Instructor: Dr. Jinoh Kim

Some contents have been imported/modified from the Weka materials.

Components of the input

- Concepts: things that can be learned
    - The output produced by a learning scheme is the concept description
- Instances: the individual, independent examples of a concept to be learned
    - More complicated forms of input with dependencies between examples are possible
- Attributes: measuring aspects of an instance
    - Will focus on nominal (categorical) and numeric ones

What’s a concept?

- Concept: thing to be learned
- Concept description: output of learning scheme
- Styles of learning:
  - Classification learning:
    predicting a discrete class
  - Association learning:
    detecting associations between features
  - Clustering:
    grouping similar instances into clusters
  - Numeric prediction:
    predicting a numeric quantity

Classification learning

- Classification learning is *supervised*
  - Scheme is provided with actual outcome
- Outcome is called the *class* of the example (instance)
- Measure success on fresh data for which class labels are known (*test data*)
- In practice success is often measured subjectively

	Sepal length	Sepal width	Petal length	Petal width	Type
1	**5.1**	**3.5**	**1.4**	**0.2**	**Iris setosa**
2	**4.9**	**3.0**	**1.4**	**0.2**	**Iris setosa**
…					
51	**7.0**	**3.2**	**4.7**	**1.4**	**Iris versicolor**
52	**6.4**	**3.2**	**4.5**	**1.5**	**Iris versicolor**
…					
101	**6.3**	**3.3**	**6.0**	**2.5**	**Iris virginica**
102	**5.8**	**2.7**	**5.1**	**1.9**	**Iris virginica**
…					

Clustering

- Finding groups of items that are similar
- Clustering is *unsupervised*
  - The class of an example (instance) is not known
- Success often measured subjectively

	Sepal length	Sepal width	Petal length	Petal width	Type
1	5.1	3.5	1.4	0.2	~~Iris setosa~~
2	4.9	3.0	1.4	0.2	~~Iris setosa~~
…					
51	7.0	3.2	4.7	1.4	~~Iris versicolor~~
52	6.4	3.2	4.5	1.5	~~Iris versicolor~~
…					
101	6.3	3.3	6.0	2.5	~~Iris virginica~~
102	5.8	2.7	5.1	1.9	~~Iris virginica~~
…					

Numeric prediction

- Variant of classification learning where "class" is numeric (also called "regression")
- Learning is supervised
  - Scheme is being provided with target value
- Measure success on test data

Outlook	Heat	Moisture	Windy	Play-time
Sunny	Hot	High	False	5
Sunny	Hot	High	True	0
Overcast	Hot	High	False	55
Rainy	Mild	Normal	False	40
…	…	…	…	…

What's in an example?

- Instance: specific type of example
    - Thing to be classified, associated, or clustered
    - Individual, independent example of target concept
    - Characterized by a predetermined set of attributes
- Input to learning scheme: set of instances/dataset
    - Represented as a single relation/flat file
- Rather restricted form of input
    - No relationships between objects

What’s in an attribute?

- Each instance is described by a fixed predefined set of features, its “attributes”
  - Number of attributes may vary in practice
  - Possible solution: “irrelevant value” flag
- Related problem: existence of an attribute may depend on value of another one
- Possible attribute types (“levels of measurement”):
  - Nominal, ordinal, interval and ratio

Nominal levels of measurement

- Values are distinct symbols
  - Values themselves serve only as labels or names
  - *Nominal* comes from the Latin word for name
- Example: attribute "outlook" from weather data
  - Values: "sunny", "overcast", and "rainy"
- No relation is implied among nominal values (no ordering or distance measure)
- Only equality tests can be performed

Ordinal levels of measurement

- Impose order on values
  - But no distance between values defined
- Example: attribute "Heat" in weather data
  - Values: "hot" > "mild" > "cool"
- Note: addition and subtraction don't make sense
- Example rule:
  Heat < hot ⇒ play = yes
- Distinction between nominal and ordinal not always clear (e.g., attribute "outlook")

Interval quantities

- Interval quantities are not only ordered but measured in fixed and equal units
- Example: attribute "year"
- Difference of two values makes sense
- Sum or product don't make sense
- Zero point is not defined!

Ratio quantities

- Ratio quantities are ones for which the measurement scheme defines a zero point
- Example: attribute “distance”
  - Distance between an object and itself is zero
- Ratio quantities are treated as real numbers
  - All mathematical operations are allowed

Attribute types used in practice

- Practically just two levels of measurement: nominal and ordinal
- Nominal attributes are also called “categorical”, ”enumerated”, or “discrete”
  - But: “enumerated” and “discrete” imply order
- Special case: dichotomy (“boolean” attribute)
- Ordinal attributes are sometimes coded as “numeric” or “continuous”
  - “continuous” implies mathematical continuity

ARFF data format

```
%
% ARFF file for weather data with some numeric features
%
@relation weather

@attribute outlook {sunny, overcast, rainy}
@attribute temperature numeric
@attribute humidity numeric
@attribute windy {true, false}
@attribute play? {yes, no}

@data
sunny, 85, 85, false, no
sunny, 80, 90, true, no
overcast, 83, 86, false, yes
...
```

Additional attribute types

- ARFF data format also supports *string* attributes:

```
@attribute description string
```

  - Similar to nominal attributes but list of values is not pre-specified

- Additionally, it supports *date* attributes:

```
@attribute today date
```

  - Uses the ISO-8601 combined date and time format *yyyy-MM-dd-THH:mm:ss*

Sparse data

- In some applications most attribute values are zero and storage requirements can be reduced
- ARFF supports sparse data storage

```
0, 26, 0,  0, 0 ,0, 63, 0, 0, 0, “class A”
0,  0, 0, 42, 0, 0,  0, 0, 0, 0, “class B”
```

```
{1 26, 6 63, 10 “class A”}
{3 42, 10 “class B”}
```

- This also works for nominal attributes (where the first value of the attribute corresponds to “zero”)
- Some learning algorithms work very efficiently with sparse data

Nominal vs. ordinal

- Attribute “age” nominal

```
If age = young and astigmatic = no
   and tear production rate = normal
   then recommendation = soft
If age = pre-presbyopic and astigmatic =
   no
   and tear production rate = normal
   then recommendation = soft
```

- Attribute “age” ordinal
  (e.g. “young” < “pre-presbyopic” < “presbyopic”)

```
If age ≤ pre-presbyopic and astigmatic =
   no
   and tear production rate = normal
   then recommendation = soft
```

Missing values

- Missing values are frequently indicated by out-of-range entries for an attribute
  - There are different types of missing values: 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 and "missing" may need to be coded as an additional, separate attribute value

Inaccurate values

- Reason: data has not been collected for mining it
- Result: errors and omissions that affect the accuracy of learning (mining)
- These errors may not affect the original purpose of the data (e.g., age of customer)
- Typographical errors in nominal attributes $\Rightarrow$ values need to be checked for consistency
- Typographical or measurement errors in numeric attributes $\Rightarrow$ outliers need to be identified
- Errors may be deliberate (e.g., wrong zip codes)
- Other problems: duplicates, stale data

Unbalanced data

- Unbalanced data is a well-known problem in classification problems
  - One class is often far more prevalent than the rest
  - Example: detecting a rare disease
- Class imbalance problem: simply predicting the majority class yields high accuracy but is not useful
  - Predicting that no patient has the rare disease gives high classification accuracy
- Unbalanced data requires techniques that can deal with unequal misclassification costs
  - Misclassifying an afflicted patient may be much more costly than misclassifying a healthy one

Getting to know your data

- Simple visualization tools are very useful
  - Nominal attributes: histograms (Is the distribution consistent with background knowledge?)
  - Numeric attributes: graphs (Any obvious outliers?)
- 2-D and 3-D plots show dependencies
- May need to consult domain experts
- Too much data to inspect manually? Take a sample!

Summary

- Components of input – concepts, instances, attributes
- Concepts – classification, clustering, prediction
- Examples (instances)
- Attributes – types, levels of measurements
- ARFF format
- Missing, inaccurate values, unbalanced data