Data Representation

Danushka Bollegala

UNIVERSITY OF LIVERPOOL

Types of data we encounter

- Categorical data
  - discrete categories (colour of a flower petal)
- Numerical data
  - integer values (number of petals in a flower)
  - real values (length of a petal)
- String/textual data
  - words in a document
- time series data / sequential data
  - continuous, chronological, flows in one direction

Things to consider

- What is the best data type to represent the value of a given variable?
- What type of data does a particular algorithm can handle?
- How can we convert one data type to a different data type?
- What is the scale/range of the data type that we are using and is it appropriate?

Case Study

- In what ways can we represent the following sentence?

The burger I ate was an awesome burger!

Case Study

- In what ways can we represent the following sentence?

The burger I ate was an awesome burger!

Method 1: By a list of words?

[“the”, “burger”, “i”, “ate”, “was”, “an”, “awesome”, “burger”]

Method 2: By the set of words?

{“the”, “burger”, “i”, “ate”, “was”, “an”, “awesome”}

Method 3: By a vector of word frequency?

(“the”:1, “burger”:2, “i”:1, “ate”:1, “was”:1, “an”:1, “awesome”:1)

Method 4: By a vector of letter frequency???

{‘a’: 3, ' ': 7, 'b': 2, 'e': 6, 'g': 2, 'i': 2, 'h': 1, 'm': 1, 'o': 1, 'n': 1, 's': 2, 'r': 4, 'u': 2, 't': 2, 'w': 1}

Data Representation

- Data representation is one of the first things we must do in data mining.
- What we can mine is largely determined by our data representation.
- There is no one best data representation method for all data mining tasks
- For example, unsuitable data representations will lead to poor classification performance irrespective of the classification algorithm.
- We represent a data point using a set of features

What is a feature?

- features are attributes of data points that we can use to represent the data points
- For example, “colour” can be a feature when representing a “flower”. The value (feature value) of the feature “colour” could be “red”.
- Features are central to data mining
- Features allow us to abstract data points and learn rules that can be used to predict things for unseen/future data points
    - If we learn the rule that
        - if colour== red then flower= rose
    - we can use this rule to classify not only flowers in our train dataset, but also all flowers including ones that we do not have in our train dataset
- Coming up of good features is an art!
    - feature-engineering
- Recent work in machine learning such as deep learning focus on automatically discovering good features from data, without any human intervention

Categorical data

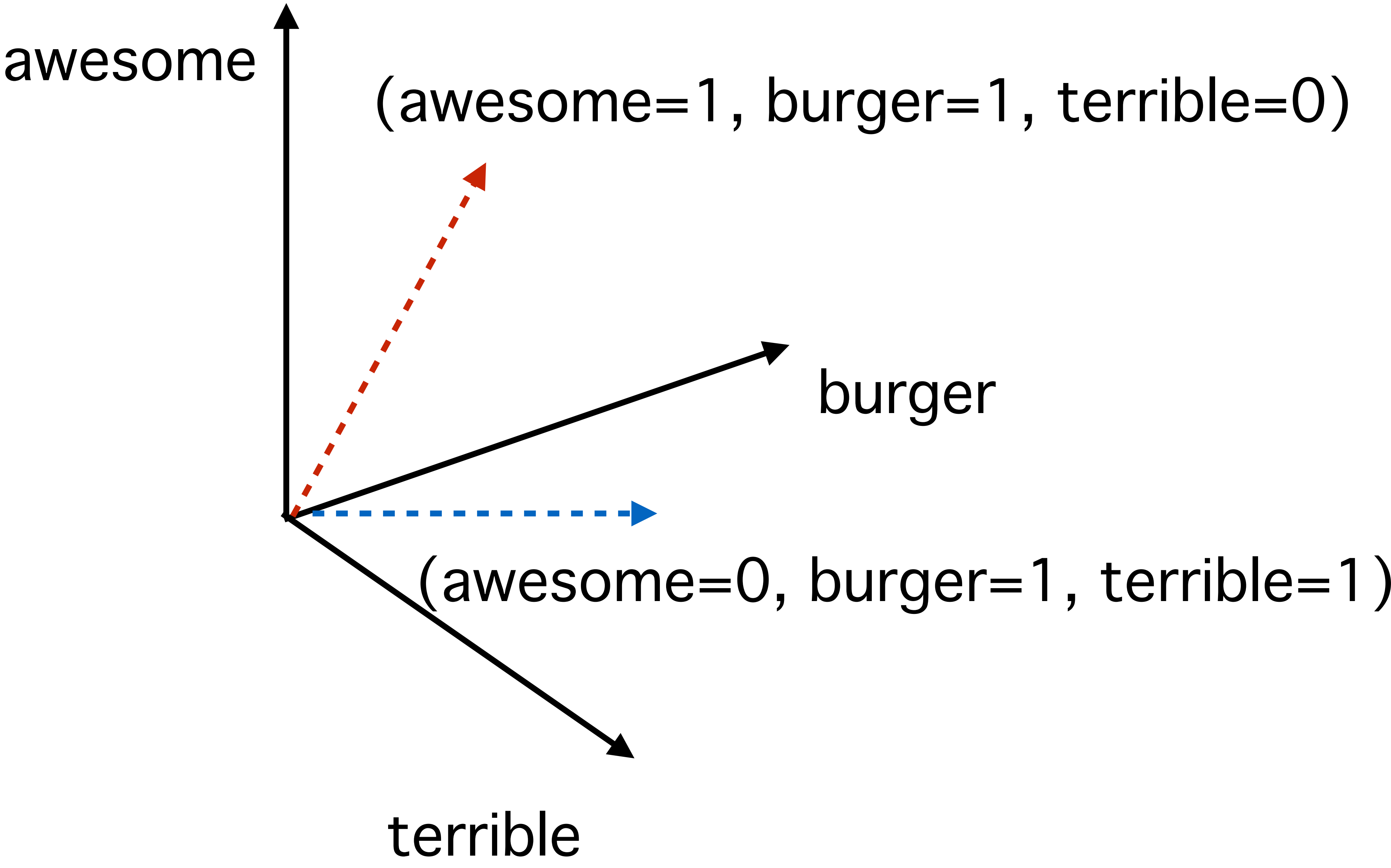
- Examples: colour of a flower
  - red, blue, green, ...
- Simple to represent and often consist of a small finite set of categories
- Easy to work when learning rules for data classification
  - if (colour == red) then flower = rose
- The number of categories need not necessarily be small
  - e.g. tags people use to label images such as names of people, locations, events or #tags is an open set

Challenges when using categorical data

- Algebraic comparisons are undefined!
  - rose(colour=red) vs. orchid(colour=yellow) ?
- No notion of feature correlation
  - “yellow” and “amber” are closely related colours. But there is no way we can guess this by looking at the data values
- Compare the above situations to
  - text_1(awesome=4) vs. text_2(awesome=0)
  - if awesome > 0 then POSITIVE_SENTIMENT

Representing categorical data

- If you must represent categorical data, then you could do so by representing each category as a separate dimension of a vector.


awesome
(awesome=1, burger=1, terrible=0)
burger
(awesome=0, burger=1, terrible=1)
terrible

Numerical data

- The number of possible values a variable can take can be infinite
    - all real values in range [0,1]
- There is a clearly defined ordering among the values (eg. 0.5 > 0.3)
- Algebraic operations are well-defined
- “most” machine learning algorithms assume you have your data points represented in (for example) n-dimensional real space

$$\boldsymbol{x} \in \mathbb{R}^n$$

Challenges when handing numerical data

- Different features will be in different ranges.
  - height ∈ [110, 230] (in centimetres)
  - weight ∈ [40, 120] (in kilograms)
- If we learn some classification rule for body mass index (BMI), then we must consider the differences in ranges in each data dimension

Feature normalization

- There are various ways to normalize (scale) a numerical features into a common scale
- Method 1: [0,1] scaling
  - compute the minimum and maximum value of a feature over the data points

$$\hat{x} = \frac{x - \min(x)}{\max(x) - \min(x)}$$

Method 2: Gaussian Normalization

- Compute the mean (μ) and the standard deviation (**σ**) for the feature and apply the following transformation

$$\hat{x} = \frac{x - \mu}{\sigma}$$

After this transformation each feature will have a zero mean and a unit variance.
Therefore, it is "easier" to compare two features, ignoring their absolute scales.

Quiz

- Assume that the height of a student takes the following values [s_1=170cm, s_2=160cm, s_3=155cm, s_4=165cm]
- Use method 1 ([0,1] normalization) to transform the four data points.
- Use method 2 (Gaussian normalization) to transform the four data point.