Inf2b Learning and Data
Lecture 4: Classification and nearest neighbours

Hiroshi Shimodaira
(Credit: Iain Murray and Steve Renals)

Centre for Speech Technology Research (CSTR)
School of Informatics
University of Edinburgh

http://www.inf.ed.ac.uk/teaching/courses/inf2b/
https://piazza.com/ed.ac.uk/spring2017/infr08009learning

Jan-Mar 2017
Today’s topics

1. Classification
2. Nearest neighbour classification
3. Decision boundary
4. Tips on pre-processing data
5. Generalisation and over-fitting


Types of learning problems

<table>
<thead>
<tr>
<th>Data</th>
<th>System</th>
<th>Type of problem</th>
<th>Type of learning</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x )</td>
<td>( { x } )</td>
<td>groups (subsets)</td>
<td>clustering</td>
</tr>
<tr>
<td>((x, y))</td>
<td>( x )</td>
<td>( y ): discrete category</td>
<td>classification</td>
</tr>
<tr>
<td>((x, y))</td>
<td>( x )</td>
<td>( y ): continuous value</td>
<td>regression</td>
</tr>
</tbody>
</table>

where \( x = (x_1, \ldots, x_D)^T \): feature vector
\( y \): target vector or scalar
Supervised learning

Test mode

New data | Label | Goal of training: develop a classifier of good generalisation
Supervised learning

Classification and nearest neighbours
Classification

- The data has a feature vector $\mathbf{x} = (x_1, x_2, \ldots, x_D)^T$ and a label $c \in \{1, \ldots, C\}$

- **Training set**: A set of $N$ feature vectors and their labels $(\mathbf{x}_1, c_1), \ldots, (\mathbf{x}_N, c_N)$

- Use a learning algorithm to train a classifier from a training set

- **Test set**: a set of feature vectors to which the classifier must assign labels – used for evaluation. (NB: training and test sets should be mutually exclusive)

- Error function: how accurate is the classifier? One option is to count the number of misclassifications:

$$\text{Error rate} = \frac{\# \text{ of misclassified samples}}{\# \text{ of test samples}}$$
Nearest-neighbour classifier

- **Nearest neighbour classification**: label a test example to have the label of the closest training example
- **$K$-nearest neighbour ($K$-NN) classification**: find the $K$ closest points in the training set to the test example; classify using a majority vote of the $K$ class labels
- Training a $K$-nearest neighbour classifier is simple! — Just store the training set
- Classifying a test example requires finding the $K$ closest training examples
  - This is computationally demanding if the training set is large — potentially need to compute the Euclidean distance between the test example and every training example
  - Data structures such as the kD-tree can make finding nearest neighbours much more efficient (in the average case)
Classifying test data with $K$-nearest neighbours

- Classifying test data with $K$-nearest neighbours

- Inf2b Learning and Data: Lecture 4 - Classification and nearest neighbours
1-nearest neighbour

![Graph showing 1-nearest neighbour classification](image)

- **Height/cm**
- **Circumference/cm**

- **Inf2b Learning and Data: Lecture 4**
- **Classification and nearest neighbours**
5-nearest neighbour
For each test example $z \in Z$:

- Compute the distance $r(z, x)$ between $z$ and each training example $(x, c) \in X$
- Select $U_k(z) \subseteq X$, the set of the $k$ nearest training examples to $z$
- Decide the class of $z$ by the majority voting:

$$c(z) = \arg \max_{j \in \{1, \ldots, C\}} \sum_{(x, c) \in U_k(z)} \delta_{j,c}$$
Geometry of nearest neighbour
Geometry of nearest neighbour – decision boundary and decision regions
Geometry of nearest neighbour

Delaunay triangulation

Perpendicular bisectors of the edges of triangles
Voronoi tessellation

Voronoi diagram
Decision boundary: boundary (surface) that partitions the vector space into subsets of different classes.

*K-NN classification forms piecewise-linear decision boundary.*
Decision regions: regions separated by the decision boundaries
Decision boundaries for $C = 3$
What $K$ should we use?

An example where a large $K$ reduces noise

$K = 1$

$K = 15$

(Black curve: KNN decision boundary, broken purple curve: the Bayes decision boundary)

The Elements of Statistical Learning (2nd Ed.)
Hastie, Tibshirani, Friedman. §13.3 p463–

Learning curves

The Elements of Statistical Learning (2nd Ed.)
Hastie, Tibshirani, Friedman. §13.3 p463–
Predict land-usage from satellite data
KNN applied to 9 pixel patch in 4 spectral bands, with $K=5$

The Elements of Statistical Learning (2nd Ed.)
Hastie, Tibshirani, Friedman. §13.3 p463–
Tips on pre-processing data

different units

⇒ Standardise features unless understand units
Wisconsin Diagnostic Breast Cancer (WDBC) data set

http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)

Consider transformation, e.g. log-transform.
Generalisation and over-fitting

How reasonable is this decision boundary?
In a competition:

Classic stories:
http://neil.fraser.name/writing/tank/

http://www.j-paine.org/dobbs/neural_net_urban_legends.html
How reliable is the error rate?

- Error rate on training data set:
  - can be $\sim 0\%$
  - ⇒ useless to estimate generalisation error

- Error rate on a test data set (exclusive to the training set)
  - How large should the data set be?
  - How should it be collected?

  *Cross validation* is used to estimate generalisation error
  (swapping test and training data sets)

  - $k$-fold cross validation ($k$-fold CV)
    (2-fold CV is sometimes called 'holdout method')
  - leave-one-out cross validation (LOO CV)
Cross validation

Population

Sampling

4-fold CV

Data set
Training data set
Validation data set
Test data set
Summary

- **Classification with similarity based methods**
  - Represent items as feature vectors
  - Compute distances to other items and sort
  - Assign a class label to the feature vector
  - $k$-NN: an example-based approach that classifies a test point based on the classes of the closest training samples
  - Larger $k$ results in a smoother solution
  - Decision boundaries/regions, Voronoi diagram

- **Generalisation**
  - Overfitting: tuning a classifier to closely to the training set can reduce accuracy on the test set
  - Compare methods on held out data (validation set)
  - Estimate final performance on *really* new data (test set)
Simple recommender system and clustering

- Work on the questions in advance to identify what you understand and what you don’t. (avoid attending the tutorial without any preparation)
- Be active/positive - prepare topics that you’d like to discuss at the tutorial
- Try writing Matlab code of your own
08 Feb.  Lab-4  K-means clustering and visualisation with PCA.

15 Feb.  Lab-5  K-NN classification and Naive Bayes

• Wednesdays at 16:10-17:00 in FH-3.D01