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>( (\mathbf{x}, y) )</td>
<td>( \mathbf{x} ) ( y ): discrete category</td>
<td>classification</td>
<td>supervised learning</td>
</tr>
<tr>
<td>( (\mathbf{x}, y) )</td>
<td>( \mathbf{x} ) ( y ): continuous value</td>
<td>regression</td>
<td>supervised learning</td>
</tr>
<tr>
<td>( \mathbf{x} )</td>
<td>( {\mathbf{x}} ) ( y ): groups (subsets)</td>
<td>clustering</td>
<td>unsupervised learning</td>
</tr>
</tbody>
</table>

where \( \mathbf{x} = (x_1, \ldots, x_d)^T \) : feature vector

\( y \) : target vector or scalar
Supervised learning

Test mode

Classification

New data

Label

N/A

Goal of training: develop a classifier of good generalisation
Supervised learning

Oranges:

Lemons:

Inf2b Learning and Data: Lecture 4
Classification and nearest neighbours
Classification

- The data has a feature vector \( x = (x_1, x_2, \ldots, x_d)^T \) and a label \( c \)
- **Training set**: A set of \( N \) feature vectors and their labels \((x_1, c_1), \ldots, (x_N, c_N)\)
- 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 (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

<table>
<thead>
<tr>
<th>Circumference/cm</th>
<th>Height/cm</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>15</td>
<td>20</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>15</td>
<td>20</td>
</tr>
</tbody>
</table>

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

- **Height/cm** vs. **Circumference/cm**

- The graph shows a scatter plot with blue dots and red triangles representing data points.

- The point marked with a star is used to illustrate the 1-nearest neighbour approach.
5-nearest neighbour

Inf2b Learning and Data: Lecture 4  Classification and nearest neighbours
Geometry of nearest neighbour perpendicular bisectors
Voronoi tessellation
Decision boundary: boundary (surface) that partitions the vector space into subsets of different classes.

K-NN classification forms *piecewise-linear decision boundary.*
What $K$ should we use?

An example where a large $K$ reduces noise

$K = 1$

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

$K = 15$

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 $K = 5$
Tips on pre-processing data

Different units

⇒ Standardise features unless understand units

Oranges: 10 8 6 4
Lemons: 10 8 6

height/cm

width/cm

height [m]

width [cm]
Tips on pre-processing data

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?
Poor generalisation: stories

In a competition:
http://blog.kaggle.com/2012/07/06/
the-dangers-of-overfitting-psychopathy-post-mortem/

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 $\sim 0\%$
  - $\Rightarrow$ 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

Data set

P1 P2 P3 P4

4-fold CV

Classification and nearest neighbours
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

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)