CS3920/CS4920/CS5920 Assessed Coursework Assignment 1

October 7, 2017

This assignment must be submitted by 23 October 2017, 17:00. Feedback will be provided within ten working days of the submission deadline.

Learning outcomes assessed

Be able to implement machine-learning algorithms, using the Nearest Neighbours algorithm as an example. Have an understanding of ways to apply the ideas and algorithms of machine learning in science and technology.

Instructions

The coursework assignment must be completed strictly individually. You should not use in your submission any downloaded code or existing R implementations of the learning algorithms. The submission is entirely electronic.

In order to submit, copy all your submission files into a directory, e.g., `Learning` on the server `teaching` and run the script

```
submitCoursework Learning
```

from the parent directory. Choose your course (`CS3920` or `CS5920`, as appropriate) and assignment (`1` for all students) when prompted by the script. You will receive an automatically generated e-mail with a confirmation of your submission. Please keep the e-mail as a submission receipt in case of a dispute; it is your proof of submission. No complaints will be accepted later without a submission receipt. If you have not received a confirmation e-mail, please contact the IT Support team.

You should submit files with the following names:

- `NN.R` should contain the source code for the Nearest Neighbours method (if there are more than one file, you should use names of the form `NN<digit>.R`, such as `NN1.R` and `NN3.R`);
- all other auxiliary files with scripts and functions (please avoid submitting unnecessary files such as the data sets, old versions, back-up copies made by the editor, etc.);
- `report.pdf` should contain the numerical results and discussion.

The files you submit cannot be overwritten by anyone else, and they cannot be read by any other student. You can, however, overwrite your submission as often as you like, by resubmitting, though only the last version submitted will be kept. Submissions after the deadline will be accepted but they will

be automatically recorded as late and subject to College Regulations on late submissions. Please note that all your submissions will be graded anonymously.

The deadline for submission is **Monday, 23 October, 17:00**. An extension can only be given by the office or academic advisor (not the lecturer).

> **Note:** All the work you submit should be solely your own work. Coursework submissions are routinely checked for this.

Coursework

This assignment requires an implementation of the K-Nearest Neighbours method for classification. The method should be implemented in R (or, if you prefer, in MATLAB; if you would like to use any other programming language, please send me an email).

Datasets

Download the three datasets, `iris.txt`, `ionosphere.txt`, and `USPSsubset.txt`, from the course's Moodle page. Each line of these files represents one observation with comma-separated attributes (except for `USPSsubset.txt`).

- `iris.txt` is perhaps one of the best known data sets to be found in the pattern recognition literature. Each observation has 4 attributes describing sepal length, sepal width, petal length, and petal width of an iris plant. The last number in the line describes the label, $+1$ standing for Iris Versicolour and -1 standing for Iris Setosa. Use the first 70 lines as the training set and the last 30 lines as the test set.

- `ionosphere.txt` contains data collected by a radar system in Goose Bay, Labrador. This system consists of a co-phased array of 16 high-frequency antennas with a total transmitted power on the order of 6.4 kilowatts. The last number in the line describes the classification. "Good" $(+1)$ radar returns are those showing evidence of some type of structure in the ionosphere. "Bad" (-1) returns are those that do not. Use the first 200 lines as the training set and the rest as the test set.

- `USPS.txt` contains normalized handwritten digits, automatically scanned from envelopes by the US Postal Service. The original scanned digits are binary and of different sizes and orientations; the images have been deslanted and size normalized, resulting in 16×16 grayscale images. The data are in two files, and each line consists of the 256 grayscale values for the pixels followed by the true label (0–9). The original data set consisted of two parts, 7291 training observations and 2007 test observations, but they have been merged into one file, `USPS.txt`.

The datasets are available from the course's Moodle page and are based on:

- A. Frank and A. Asuncion (2010). UCI Machine Learning Repository http://archive.ics.uci.edu/ml. Irvine, CA: University of California, School of Information and Computer Science.
- Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard and L. D. Jackel. Handwritten digit recognition with a back-propagation network. In: D. Touretzky (ed.), *Advances in Neural Information Processing Systems* (NIPS 1989), 2, Morgan Kaufman, Denver, CO, 1990. The USPS data set itself can be found on

  `http://statweb.stanford.edu/~tibs/ElemStatLearn/data.html`

  (the web site of Hastie et al.'s textbook; see the bottom of the page, "ZIP code"). Made available by the neural network group at AT&T research labs.

For each dataset do the following:

1. Load it into R using a command like

```
data <- read.table("filename.txt")
```

2. Split the dataset into the training and test sets.
3. Run the K-Nearest Neighbours method with different parameters (the value of K and perhaps metric) as described below.
4. Write the results in your report (see below for a description of what to report).

Details of implementation

The K-Nearest Neighbours algorithm is described in Chapter 2. When $K = 1$, the predicted label for a test object x is the same as the label of the nearest training object (in Euclidean metric $\|x_1 - x_2\|$ or tangent metric). When $K = 3$, the predicted label is obtained by majority vote among the three nearest neighbours. Implement the algorithm for $K = 1$ and $K = 3$. You are not allowed to use any existing implementations (such as `knn()` in the `class` library); please also avoid using powerful R functions such as `sort` and `order` (which makes the task significantly easier).

Calculate predicted labels for all test objects and compare them with the true labels for the test objects for the `iris.txt` and `ionosphere.txt` data sets. Use Euclidean metric and $K = 1$ and $K = 3$. Calculate the percentage of correct predictions on both data sets (using basic R commands and not using functions such as `table`). Give the results in your report (there should be 4 numbers overall: `iris.txt` with $K = 1$ and $K = 3$; `ionosphere.txt` with $K = 1$ and $K = 3$).

Optional part

Calculate the percentage of correct predictions on the `USPS` data set using both Euclidean and tangent metrics. Make your training and test sets are large as possible (extracting them from `USPSsubset.txt` or even from `USPS.txt`). In the case of Euclidean metric, try normalization to see if your results improve. Try numbers $K = 1, 2, 3$ of nearest neighbours. Give your results in your report.

Marking criteria

To be awarded full marks you need both to submit correct code and to obtain correct results on the given data sets. Even if your results are not correct, marks will be awarded for correct or partially correct code (up to a maximum of 70%). Correctly implementing 1-Nearest Neighbour will give you 60%.

Extra marks

There are several ways to get extra marks (at most 10%) that will be added to your overall mark (the sum will be truncated to 100% if necessary). Extra marks will be given for experiments with the USPS data set, especially involving tangent distance and implementation in languages other than R to make computations more efficient, for implementing the algorithm for a general K, and for any interesting observations about the datasets and the method (discuss these in your report).