SCHOOL *of* ELECTRICAL ENGINEERING & COMPUTING
COLLEGE *of* ENGINEERING, SCIENCE & ENVIRONMENT
The UNIVERSITY *of* NEWCASTLE

Comp3330/Comp6380 Machine Intelligence, Semester 1, 2021

Project A and Written Assignment A: Introductory Machine Learning Project

Deadlines: Check dates and times fro both parts on Blackboard and submit via Blackboard)
Maximum possible marks: 20 = 10 (group project part) + 10 (individual paper)

Description

In this assignment we want to gain basic experience in testing out ANNs and SVMs for classification. There are two components for marking this assignment. Both components have to be submitted separately in Blackboard:

Project 1A [10 marks, group based]. The group's experimental work should be summarised in a brief group summary report that provides an overview what your team has done in each of the experimental components of your project. It should explain how the software and data is structured and detail your group's individual member contributions to the project outcome. Marked will be the quality, volume and depth of the experimental results achieved by the group. If the member contributions within the group are very different the marks will be weighted accordingly. The recommended length of the group summary report is about 2-3 pages. Include in your submission the summary report and all essential files and code that are required for reproducing and verifying your results. It is sufficient if one team member submits. Make sure that all other group member names are listed and each member has signed **(!)** the agreed group report with individual member contribution statements.

Report 2A [10 marks, individual]. Each student writes an individual paper that describes, analyses and critically discusses the project results achieved by the student and his/her team in detail. We expect about 4-8 pages from COMP3330 students, and about 6-12 pages from COMP6380 students. Aim at providing a high quality report that describes and discusses your results and approaches from part 1A clearly and concisely following instructions of the individual questions Q1, Q2, Q3 below. Your individual report should be formatted in Springer LNCS format. Any literature citations, e.g., in the background, method or discussion sections should follow a consistent citation style. For COMP6380 students the citations and paper presentation are expected to be of high professional standard. The LNCS conference paper template is available e.g. at the following link: `https://www.springer.com/gp/computer-science/lncs/conference-proceedings-guidelines` Please submit your individual paper in pdf format in Blackboard.

Notes: Be prepared that training ANNs can require some time. We recommend using Python, scikit-learn and PyTorch. However, any language/library combination is acceptable while it is expected that you are able to acquire the necessary details how to use the software or programming language of your choice from relevant on-line help or literature. Plot error curves that indicate convergence times (how many iterations did it take?). For demonstrating how well your trained ANN model generalises you can visualise the results of your tests (you can

submit several plots from different networks or different training schemes) or you may consider suitable statistical measures. Always discuss your results and highlight the most important outcomes.

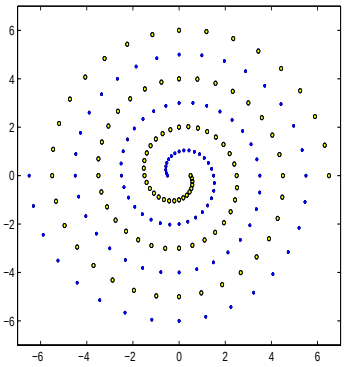
Warnings:

- You will find that some of the questions can lead into open-ended research and some of the experiments may take significant time on the computer. It is your responsibility to decide on a sensible balance of quality and depth of your investigation of each individual question so that the assignment can be completed within the given time. If computing resources become a severe issue provide pilot results as proof of concept, explain the issue and solution attempts and put more emphasis on the analysis and discussion.
- If you use or discuss results obtained by other team members or any details that you find in literature, alwayst fully cite or acknowledge your sources and contributors.

Q1 Variations of the Two-Spiral Task [3 marks in 1A]

Perform an experimental study on the following variations of the two-spiral task:

a) (ANN training): Start with the "original dataset" of Lang and Witbrock (1988) with 194 training points (see Figure below). How fast and how well can you solve this task using a feed-forward NN? (The (x, y)-coordinates of the points in the dataset will be supplied in blackboard.) [1 mark in 1A]



b) (ANN training): Generate a variation(s) of the 2-spiral task. Then solve the associated classification task using ANNs and discuss your approach and solution in comparison to a). You may consider, e.g., to generate three spirals or four spirals, or use code such as available at `https://gist.github.com/45deg/e731d9e7f478de134def5668324c44c5` [1 mark in 1A]

c) (ANN vs. SVM): Run experiments that allow to compare ANNs and SVMs on solving the two classification tasks above and report the outcome. [1 mark in 1A]

For each subquestion try out different architectures, parameters, and methods. Compare and discuss their performance (speed, generalisation). It is recommended that you focus for each part of your experiments on *about two* different aspects that you investigate in more detail (this could be e.g. variation of the step size, number of hidden layers/units, use of momentum, different kernels or kernel parameters in SVMs, ...). The performance of the solutions can be evaluated by visual inspection of a generalisation test applied to all pixels of a section of the (x, y)-plane (that for the 2-spiral data should result in two intertwined spiral shaped regions). You may also think about alternative performance measures.

In your individual reports you can document the process of researching and creating the classifiers and discuss how well they perform. A background paper with literature links, description of the data and some hints about previously successful network architectures is, for example, the following survey (Chalup and Wiklendt, 2007).

Q2 Dry Bean Dataset [3.5 marks in 1A]

Seven different types of dry beans were used in this research, taking into account the features such as form, shape, type, and structure by the market situation. A computer vision system was developed to distinguish seven different registered varieties of dry beans with similar features in order to obtain uniform seed classification. For the classification model, images of 13,611 grains of 7 different registered dry beans were taken with a high-resolution camera. Bean images obtained by computer vision system were subjected to segmentation and feature extraction stages, and a total of 16 features; 12 dimensions and 4 shape forms, were obtained from the grains.

CLASSES

There are 7 decision classes:

1: SEKER

2: BARBUNYA

3: BOMBAY

4: CALI

5: HOROZ

6: SIRA

7: DERMASON

NUMBER OF EXAMPLES

- training data = 13611

ATTRIBUTES

There are 17 attributes:

1. Area (A): The area of a bean zone and the number of pixels within its boundaries.
2. Perimeter (P): Bean circumference is defined as the length of its border.
3. Major axis length (L): The distance between the ends of the longest line that can be drawn from a bean.
4. Minor axis length (l): The longest line that can be drawn from the bean while standing perpendicular to the main axis.
5. Aspect ratio (K): Defines the relationship between L and l.
6. Eccentricity (Ec): Eccentricity of the ellipse having the same moments as the region.

7. Convex area (C): Number of pixels in the smallest convex polygon that can contain the area of a bean seed.
8. Equivalent diameter (Ed): The diameter of a circle having the same area as a bean seed area.
9. Extent (Ex): The ratio of the pixels in the bounding box to the bean area.
10. Solidity (S): Also known as convexity. The ratio of the pixels in the convex shell to those found in beans.
11. Roundness (R): Calculated with the following formula: $(4piA)/(P^2)$
12. Compactness (CO): Measures the roundness of an object: Ed/L
13. ShapeFactor1 (SF1)
14. ShapeFactor2 (SF2)
15. ShapeFactor3 (SF3)
16. ShapeFactor4 (SF4)
17. Class (Seker, Barbunya, Bombay, Cali, Dermosan, Horoz and Sira)

DOWNLOAD

The data is available at the UCI Machine Learning Repository

`https://archive.ics.uci.edu/ml/datasets/Dry+Bean+Dataset`

TASK

Your task of your group is to use the training data to train your classifier and report the maximum accuracy that you can achieve in a 10-fold cross-validation experiment. For solving this you can train a SVM and/or a Neural Network, or some combination. Your experiments should also investigate if the data needs to be normalised or requires some alternative form of preprocessing and what impact this has on the results.

In your individual reports you can document the process of researching and creating this classifier and discuss how well your classifier performs by using some suitable form of metrics e.g. considering false positives and false negatives, confusion matrices, learning curves etc. You should also address and discuss the impact of preprocessing and if it was necessary.

Acknowledgment and paper:
`https://doi.org/10.1016/j.compag.2020.105507`

Q3 Select Your Own Data [3.5 marks in 1A]

For this question please perform a comparison study of SVMs and ANNs on a data set of your choice but different from the data already used in this assignment. You can find other possible data sets, e.g., at:

- UCI repository `https://archive.ics.uci.edu/ml/datasets.php`
- Kaggle `https://www.kaggle.com/datasets`

a) Group part: Submit your full study with all specifications so that the marker is able to verify it.

b) Individual part: Describe your approach in a concise report that is detailed enough to allow your solution to be replicated. Include a detailed analysis of your classifier and the outcome of your experiments.

Note

Marks will be awarded for the performance of the classifier, evidence of researching better solutions for the classifier, and evidence of understanding the training process and the effects of the various training parameters. For details please consult the marking guides that will be provided separately. Depending on the configuration of your solution you may be asked to give a demo to the tutors for evaluation. If you have any questions about the specific submission format of your solution please consult with the tutor. Make sure you submit before the deadline.

Literature

S. K. Chalup, and L. Wiklendt. Variations of the Two-Spiral Task. *Connection Science 19*(2), pp. 183-199, June 2007.
Available at http://hdl.handle.net/1959.13/808886

K. J. Lang and M. J. Witbrock. Learning to tell two spirals apart. In: Touretzky, D., Hinton, G., Sejnowski, T. (Eds.), *Proceedings 1988 Connectionist Models Summer School*. Morgan Kaufmann, Los Altos, CA, pp. 52–59, 1988.

T. Mitchell. Machine Learning, McGraw Hill, 1997.