COMP9417 Project

July 6, 2020

Aims

Learning objectives of this assignment:

- a self-selected task to extend aspects of the course material
- involves practical aspects of the machine learning problem, i.e.
  - implementing or modifying algorithms and/or
  - experimental evaluation of algorithms on data set(s)
- exercise written communication skills in motivating, recording and summarising work done on a specified task

Submission

The hand-in for this assignment has two parts:

- files containing program code to do something interesting with data set(s) and/or results of running programs on data set(s)
  - compressed archive of files
  - any programming language can be used
- a report on what you did.
  - must be a single document in PDF format
  - must include names and zIDs of **ALL** team members

Note: **ONLY ONE** person on the team submits both parts of the assignment.

Note: currently there is a **2MB limit** on the size of `give` submissions. If you need to submit a larger file, submit your PDF via give and provide a link to a Google drive with the supporting material.

Submission

- Submission is via give. You need to submit the project files and your report.
- The project files should be submitted as a compressed tar or zip archive. If your submission exceeds the current limit (2MB) contact the course admins to arrange an alternative method of submission for the excess materials, such as a download link.
- The report must be in PDF format.
- Combinations accepted are (if you are running from the Linux command-line):
  - $ give cs9417 assignment files.tgz report.pdf
  - $ give cs9417 assignment files.zip report.pdf

Marking

Total: 30 marks available

- **Part 1:** [15 marks]
  - 8 marks: solving the basic problem as described in the topic
  - 7 marks: extra features, or 1 person solving most or all of a > 1 person problem
- **Part 2:** [10 marks]
  - 6 marks: describing the problem and your solution
  - 4 marks: good presentation and communication of results
- **Achievement:** [5 marks]

Part 1

Marks will be gained by:

- evidence of good design or planning by breaking down the problem into sub-components
- rigorous collection of results
- use of comments and notes to record decisions taken and reasons for them in the process of the work
- Motivating the choice of project and your approach (e.g. why was the project interesting? has it been done before? what is different about your approach?)

Part 1

Marks will be lost by:

- programs failing to compile or run
- missing results files
- no clear information on contents of files submitted (e.g. in README)
- evidence of plagiarism (including submissions that are very similar to existing implementations online)

Part 2

Marks will be gained by:

- evidence of thorough testing of an idea
- good presentation and summary of key results using tables, graphs, etc.
- simple, clear and relevant explanations
- well-formatted, well-organised, spell-checked and grammar-checked documents

Part 2

Marks will be lost by:

- inappropriate length (aim for around 2 pages per group member — extra figures, tables, etc. can go in an appendix of *reasonable* length)
- digression, rambling or waffling to fill space unnecessarily
- errors or inconsistencies in presentation, such as
  - incorrect description of algorithms or their properties
  - poor algorithm selection for a task
  - errors in evaluation like not using an independent test set or cross-validation if this is required
  - statements or conclusions not based either on your experimental results or referenced sources
  - incorrect or inappropriate use of statistical tests
- evidence of plagiarism

Group Configuration

Each team must be configured with 1-4 students currently enrolled in the course

- Teams can be made up of students from different tutorials
- Larger teams are expected to do more (achievement grade will be affected by this)
- Teams should submit a summary of work completed by each member. If missing, we will assume that all members contributed equally.
- You can use webcms forum to form groups if needed.
- **Add your group to the google sheet!. Deadline to do this is 17th July, 2020.** (You do not need to create a webcms group).

Report Structure

Giving a very strict set of guidelines to the format of the report for the project is difficult since the different projects are very varied. However, some things to keep in mind are:

- **Length:** Keep it concise, roughly speaking, about 2 pages per person in the group on average. Include a README file with the code so you don't have to put that type of information in the report.
- **Introduction:** You must explain the problem you have tackled, the basic approach taken to solving it, why you chose it, and any important aspects of that approach in terms of machine learning.
- **Implementation:** If your work was mostly implementation, focus on that. Otherwise briefly describe what you did.

Report Structure

- **Experimentation:** All methods must be tested on some data, so these results should be included. Additionally, if this was a major focus, you will need to explain the work done and what was accomplished, for example on setting up the learning task, choice of evaluation, and so on. Detailed statistical analyses are probably outwith the scope of the project, so don't include these unless you are already very familiar with this kind of thing.
- **References:** Should be there for algorithms used or other aspects of the work.
- **Appendix:** Should be used if you have a lot of experimental results. However, consider plotting graphs or using other visualizations like histograms to summarize a lot of results concisely.

Deadline

Sunday August 9, 2020 23:59:59

Topics: Topic 0 - Propose your own

The objective of this topic is to propose a machine learning problem, source the dataset(s) and implement a method to solve it. This will typically come from an area of work or research of which you have some previous experience.

- it must involve some practical work with some implementation of machine learning
- you must send an email to the course coordinator (Omar) with a description of what you are planning (a couple of paragraphs would usually be enough) that needs to be approved in an emailed reply before you start
- it must not involve double-dipping, i.e., be part of project for another course, or for research postgrads it must include a statement to the effect that it is not part of the main work planned for the thesis (although it can be related)

Topics: Topic 1 - Machine Learning Paper

The objective of this topic is to choose a journal or conference paper, summarise its findings, and implement the proposed algorithm on a new or simulated dataset.

- Good sources for papers are: NeurIPS, ICML, JMLR, JAIR, ICLR, or ArXiv
- You may also choose a series of papers and compare various approaches to the same problem.
- Email the course coordinator before you get started on this one too.

Topics: Topic 2.1 - Competitions & Challenges - Kaggle

A number of sites now host regular competitions. You can enter a live competition or work on the dataset from a past competition.

- The main site hosting machine learning competitions is Kaggle, with the competitions hosted here. Only competitions in categories other than "Getting started" are acceptable. However, you can select one from either Active or Completed competitions to work on.
- assess carefully the time you will need to understand the competition requirements, get familiar with the data and run the algorithm(s) you plan to use
- for live competitions you can include your submission's placing on the leaderboard at submission time!

Topics: Topic 2.2 - Competitions & Challenges - CodaLab

CodaLab hosts datasets, code and papers on some areas that may be of interest, including Question Answering and Image Processing. Question Answering is a task in the area of Natural Language Processing that requires mapping a natural language question along with some knowledge source (text or a structured database) into an answer. The Coda Lab worksheet on Image Classification contains some popular image classification datasets, such as ImageNet.

Topics: Topic 2.3 - Competitions & Challenges - Deep Reinforcement Learning/Transfer Learning

There are a number of tasks, from introductory up to pretty much state-of-the-art in reinforcement learning hosted at the OpenAI "Gym", such as the "Retro Contest".

- It is recommended that you do not attempt this topic unless you have experience in both deep learning and reinforcement learning.
- There are a number of options, from basic control tasks to learning to play Atari games to robot control tasks. Some tasks, such as "Toy text", are introductory and are **not sufficient** by themselves for this assignment.

Topics: Other Considerations

- Do not choose a project that needs a significant amount of data processing, or 'create' a dataset, as we are primarily interested in machine learning in this course, not data cleaning. Of course most tasks will require *some* preprocessing.
- A larger group is expected to achieve more, and group size will be taken into consideration when assigning marks for achievement and extra features.
- Choose a topic that interests you, but be pragmatic when it comes to time requirements and difficulty of the project.
- Use common sense when choosing competitions/datasets/models. Do not expect a good grade if you choose a very simple task.
- Before using advanced machine learning techniques, always use a simple baseline such as a decision tree or logistic regression.