

NUMERICAL OPTIMISATION PROJECT GUIDELINES

OPTIMISATION IN CLASSIFICATION WITH SUPPORT VECTOR MACHINES (SVMs)

Marta Betcke

Submit a **single PDF report** via TurnItIn by **Friday 1st Apr 2022 at 16:00 (UK time)**

Include the name of the optimisation method used **in the name of the file submitted to TurnItIn** and in the header/title of your report.

Scope:

This is a numerical optimisation and not a machine learning project. The emphasis here is on **optimisation methods and their convergence** not on classification performance. However, it is important to choose the right formulation for your problem with arguments to support your choices and at the end present the classification result and quantify it in terms of some classification metrics.

Report:

The report needs to be well structured (address each point completely in a dedicated section), clear and self contained i.e. include the description of the classification problem (it is absolutely fine to use synthetic data, if using real data keep the level of application specific details to minimum), the chosen formulation, the optimisation method used and any theorems you invoke to argue convergence. There is no need to include derivations or proofs. **Always provide references to sources of any claims, theorems, methods** etc (avoid Wikipedia, good Wiki contributions state original sources!).

Do not expect us to search in your text for arguments, guess what you may have meant, look things up, or accept any statements without coherent supporting argument with reference to sources.

Mathematical rigour on par with the lecture and clear and succinct writeup is expected. **3k words is total hard limit** (code excluded), please include the word count.

Code:

The optimisation method needs to be implemented by you from scratch (**no use of optimisation packages is allowed**) except for reusing codes that were implemented by you or made available by us in tutorials. For setting up the classification problem you can use dedicated packages, but do not use the built in optimisation routines. You can use Python or Julia but you may need to reimplement some routines that were provided in Matlab.

Include your implementation of the optimisation method and an excerpt from testing script showing how it is called (keep it to minimum necessary to validate that you used your own implementation) as an appendix in your report i.e. collate everything as one single PDF document for submission. The code will not count towards the word limit.

UCL rules and regulations on late submission and plagiarism apply.

SVM classification and the underlying optimisation problem:

Support vector machines (SVMs) are a well established and rigorously funded technique for solution of classification problems in machine learning.

- (a) Choose and describe a classification problem. [5 pt]
- (b) Choose the most appropriate SVM formulation for it (primal/dual, linear/nonlinear, choice of loss etc). Justify your choice for your classification problem. [5 pt]
- (c) Shortly introduce and explain the chosen formulation. There is no need for comprehensive derivation. [5 pt]
- (d) Identify the type and properties of the resulting optimisation problem and any challenges it poses. [5 pt]

Marks: 20 pt baseline + up to **5 pt** bonus (more complicated maths of the problem e.g. nonlinear, non-standard loss etc. Rigorous mathematical discussion is expected to get bonus points). [: 20 - 25 pt]

Optimisation method and convergence theory in the context of your problem:

Note: Theory will only be given marks if there is a serious attempt on numerical solution.

- (a) Propose an appropriate method for solution of your optimisation problem. Justify your choice w.r.t. applicability, convergence type and rate, CPU time/memory efficiency. [5 pt]
- (b) Describe the method in sufficient detail so it is clear how it works (state your sources, unify notation). You may want to introduce a method which was not on the syllabus and apply it to your optimisation problem. The bonus points awarded will depend on the level of difficulty of the chosen method and the theory involved. [5 pt]
- (c) Discuss if local or global convergence can be expected for your problem. For more details see below (†). [10 pt]
- (d) Discuss theoretical local convergence rates predicted for your problem. For more details see below (†). [10 pt]

(†) To argue convergence in (c,d) paraphrase theorems from the lecture or other respectable sources (books, journals, trusted lecture notes etc). Always state the complete result, this may involve combining multiple lemmas and theorems into a coherent theorem (reference all sources, unify notation!). Explain why this result applies to your problem i.e. check the assumptions against the properties of your problem. If you cannot exactly match the problem with the theory explain this clearly and discuss what is the departure from the theory in your case and what effect on convergence do you expect and why.

Marks: 30 pt baseline + up to **20 pt** bonus (method substantially different to those on the syllabus, different convergence theorems etc). [: 30 - 50 pt]

Solution and discussion of the results:

- (a) Solve the optimisation problem, state all relevant parameter choices. Critically discuss the obtained solution. [5 pt]

- (b) Provide relevant convergence plot(s) and discuss empirical convergence rate qualitatively (e.g. linear/quadratic/superlinear with qualitative assessment if particularly slow/fast; no need to calculate the precise rate). **[5 pt]**
- (c) Discuss the theoretical versus empirical convergence rates including all relevant checks e.g. was trust region active or not, was step size 1 attained or not etc. **[5 pt]**
- (d) Discuss the performance of the method in terms of complexity, CPU time, memory used. **[5 pt]**
- (e) Present the classification result and provide quantitative evaluation. For this project, this is purely for verification purposes, no need to optimise the classification performance. **[5 pt]**

Marks: 25 pt baseline

[: 25 pt]