Practical 1: Predicting Superconductivity critical temperature

CS5014 Machine Learning

Due date: Wednesday 11th March 2020 (Week 7) 21:00
40% of the coursework grade

Aims

The main aim of this practical is to gain experience in working with real data. You will read, clean and process data from an existing dataset. You will then create a regression model to predict output based on a set of inputs and evaluate its performance.

Task

On studres, you will find a dataset which documents the critical temperatures for a variety of different superconductors. The dataset is documented in the corresponding paper, also provided on studres.[1] You are asked to predict the final column in the data (the critical temperature) using a combination of other features. The solution is expected to consist of several steps:

1. loading and cleaning the data,
2. analysing and visualising the data,
3. preparing the inputs and choosing a suitable subset of features,
4. selecting and training a regression model,
5. evaluating the performance of the model, and
6. a critical discussion of the results and your approach.

Each of these steps should be clearly explained in the report. You may find some of the steps more relevant than others, e.g. you may choose to use a subset of features or all of them, as long as you provide a justification for either decision. In all cases, you should show that you understand the consequences of each decision on the performance of your model.

Try to keep the report informative and focussed on the important details and insights – the report also demonstrates an understanding of what is important. There is a maximum page limit of 15 pages, note that this is a limit not a target. If you have large amounts of (relevant!) data, you can move them to an appendix and refer from the main text.

You are not expected to outperform the published results. There are many legitimate ways to approach this task; treat it as an open problem on which you can test everything covered in the module so far.

[1] Hamidieh, Kam, A data-driven statistical model for predicting the critical temperature of a superconductor, Computational Materials Science, Volume 154, November 2018, Pages 346-354.

Deliverables

Hand in via MMS, by the deadline of 9pm on Wednesday of Week 7:

- The Python source code of your application.
- A report in PDF format which contains details of each step of the process, justification for any decisions you take, and an evaluation of the final model. This should also contain evidence of functionality and any notable figures you have produced.

Please create a `.zip` file containing both and submit this to MMS.

Marking and Extensions

This practical will be marked according to the guidelines at `https://info.cs.st-andrews.ac.uk/student-handbook/learning-teaching/feedback.html#Mark_Descriptor`
Some examples of submissions in various bands are:

- A *basic implementation* **in the 11–13 grade band** is a submission which implements a regression model in a straight-forward way and contains some evaluation, but is lacking in quality and detail, or is accompanied by a weaker report which does not evidence good understanding.
- An implementation **in the 14–16 range** should complete all parts of the specification, consist of clean and understandable code, and be accompanied by a good report which clearly describes the process and reasoning behind each step and contains a good discussion of the achieved results including graphs and evaluation measures.
- To achieve a grade of **17 and higher**, your solution should extend a solid basic solution *in a meaningful way*. Potential extensions include comparison of multiple algorithms (e.g. by comparing different optimisation strategies, different loss functions, different regularisation approaches, different subsets of data, etc.), or applying more advanced algorithms from course textbooks and research publications. Using more than one algorithm does not constitute an extension unless accompanied by additional insight and analysis gained from this.

Note that the goal is *solid machine learning methodology and understanding* rather than a collection of extensions – a good scientific approach and analysis are difficult, whereas running many different scikit-learn algorithms on the same data is easy. A basic solution can be based on a linear regression model, as long as the methodology and evaluation are sound. Be thorough in your basic solution and see extensions (e.g. a comparison with a different kind of regression model) as a means to strengthen your basic argument and methodology.
Also note that:

- We will not focus on software engineering practice and advanced Python techniques when marking, but your code should be sensibly organised, commented, and easy to follow.
- Overlength penalty: Scheme A, 1 mark for work that is 10% over-length, then a further 1 mark per additional 10% over. See `https://www.st-andrews.ac.uk/policy/academic-policies-assessment-examination-and-award-coursework-penalties/coursework_penalties.pdf`

- Standard lateness penalties apply as outlined in the student handbook at https://info.cs.st-andrews.ac.uk/student-handbook/learning-teaching/assessment.html
- Guidelines for good academic practice are outlined in the student handbook at https://info.cs.st-andrews.ac.uk/student-handbook/academic/gap.html