Advanced Machine Learning Algorithms
"A comprehensive study"

Person #1

School of Engineering (STI)

École polytechnique fédérale de Lausanne (EPFL)

Lausanne, Switzerland

Email: you@epfl.ch

Person #1

School of Engineering (STI)

École polytechnique fédérale de Lausanne (EPFL)

Lausanne, Switzerland

Email: you@epfl.ch

Abstract—**An abstract is an abbreviated/condensed version of your full paper. In most scientific conferences and journals the abstract should usually be below 200 words. The main purpose of an abstract is to give a quick overview to a potential reader. In this way he can decide if it is worth while for him to read your paper. An abstract should vaguely follow this structure; (1) A high level introduction of the topic you cover in your paper highlighting why it is an important area of research. Basically why do we care about doing research on such and such; (2) Your experimental design and hypothesis; (3) Your major findings including the key quantitative results or trends; (4) A very brief interpretation and conclusion. As you can can see the abstract is an extremely compact version of your full paper/report.**

There are many detailed resources regarding scientific writing. I am not going to write another one, but instead give you some good references I found.

1) http://abacus.bates.edu/~ganderso/biology/resources/
   writing/HTWsections.html#introduction
2) http://www.nature.com/scitable/ebooks/
   english-communication-for-scientists-14053993/
   writing-scientific-papers-14239285

I. INTRODUCTION

Typically an introduction at the start is fairly broad and introduce the topic to an non-expert reader. In this section you introduce the problem domain your work is part of and you make a case why research in this area is important. You introduce what is known about the domain, what research has already been done and what are still the open problems you seek to address with this paper you are writing. At the end of the introduction you state your main hypothesis and you give an outline of the paper.

II. METHODS

When comparing machine learning algorithms it is important to keep the following aspects in mind.

1) How many parameters does your algorithm have? This is important, some algorithms especially generative models (like GMM) which estimate a distribution of the data require approximately an order of 10 samples per parameter to be able to estimate them reliably. Other algorithms are less sensitive to this problem.
2) How sensitive is the algorithm to very few samples ?

3) What is the sensitivity of your algorithm with respect to outliers and missing data.
4) What is the **space** and **time** complexity at **train** and **test** ? Typically in papers you will see claims that the authors novel algorithm takes far less **epochs** during the training than another algorithm X. What is critical though is the **time** taken to do one epoch.
5) Sensitivity to initialization. Some machine learning techniques are much more sensitive than others when considering the initial choice of parameters. It is important to test for it. It is important for the reader to know that if he wants to achieve the same accuracy as you, he should also resort to *black magic*, i.e. extensive parameter tuning.
6) Effect of learning rate. If the optimization of machine learning method is based on gradient descent or some variant make sure to evaluate the effect different learning rates have on the train and test performance.

III. RESULTS

A. *Your analysis*

In your results you should show both **qualitative** and **quantitative** figures. It is good to first show examples of the algorithms you are comparing on toy data (2D-3D). In this way you can give a sense to the reader of the particular strengths and weaknesses of the algorithms you are evaluating. Then you should move on to real datasets which are used for benchmarking. Here is the link to the UCI Machine Learning Repository http://archive.ics.uci.edu/ml/. It has many datasets which can be used to benchmark your algorithms. Please feel free to look for other dataset repositories as well!

B. *Your figures*

Every figure you show should have the following properties.

1) All plots should be in vector graphics format (.pdf or .svg). Both in python and matlab you can save your figures in this format.
2) The **fontsize** of the x-y-z axis should be readable. I typically use a **fontsize** of 16 (in matlab) for the axis and 20 for the title. Also choose the correct resolution and number of ticks in your plots (XTick and YTick both in matlab and python).

3) All figures will have a self contained caption, explaining what we see and what is the significant. The first thing a reader will do is to glance over your figures. He will not want to go and look into the text to understand the figure. A figure should be in best case scenario independent of the text and fully self-contained.

IV. Discussion

V. Conclusion

VI. Appendix

References

[1] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.