University of Oslo

FYS-STK4155
Project 1

October 10th, 2020

Kjell R. Christensen

ABSTRACT

Machine learning as a research field, has for the last decade bounced back, “again”, due to waste amount of data that is being generated, and today an overwhelming processing power. Never, has so much data been produced by social medias, banks, insurance, sales organizations, and research is again spread wildly on many different disciplines.
But still, some data is hard to produce, and need planning and huge amount of resources to extract. A good example are experiments done in CERN. It is extremely costly, limited and restricted access and produces huge amount of data, for post processing. Due to the latter, a lot of research within e.g. nuclear and astrophysics might at time be based on limited amount of datasets.
This project will make an analysis on small datasets by using linear regressions. We will compare three common methods (OLS, Ridge & Lassos) and show that one of them is more favorable that the other two, when it comes to regression analysis, for limited numbers of datasets.

INTRODUCTION

-Hva du har gjort?
-Har jobbet med å finne et skjæringspunkt for komplekistet i datamodellen, slik at processeringen kan begrenses. S
-Polynomial fitting is and procesing is costly and conform to the n(p-1) formula.
-

METHODS

The analyses is done in a msCode environment and is using standard libraries as specified in the *.py heading. All “imports” are based on standard libraries, commonly used for ML, and routines and code generated for this project, is located only in the python file.

I have use two datasets, one test-set, generated by extracting random numbers within [0,1] and scaling them by subtracting the mean. Second dataset is based on real topographic data. I have been scaling these data in the same manner as for the test-set.

Both datasets are feed into the Franke’s Function f(x,y) and returning a vector z, for further testing.

The program is modularized, based on separate tasks, and I have modularized in such a way that the program itself, can run one, several or all modules, in one go for analyses. A separate test environment is, hence, not needed.

The reader can at any time reproduce any test-results or plots, by activating the selected part in the MainModule().

All plots are located in the ./Plots folder and they are named, based on function, parameters and methods. The naming convention is also indicating the size of the sample data, to ease the setup, and re-run the tests.

Part of the analysis is to benchmark the resampling methods, k-fold and bootstrap.

I have expanded both the k-fold and bootstrap routine for cross-validation to also include the encapsulation of the outer-loop for, polynomials and set of lamdas, respectively. This makes the code more re-usable for the different test-cases.

RESULTS & DISCUSSIONS

My finding has a clear indication that

Results and Discussion
-Implementation

CONSLUSIONS & EXPECTATIONS

Our aim for the project was to test and compare three regression methods for limited amounts of datasets. A clear objective to find the most suitable polynomial for the linear regression method, based on OLS.
For the OLS method, our prediction model/f(x) is based on polynomial approximation. Our scope was limited, in the first part is limited to polynomial of degree 5, and falls a bit short of the optimal degree that is more likely to be 6 or seven. Since the scope for this part is polynomial p >=5, we see that the optimal value

The analysis for
Conclusion…
Perspective for future work…
Pros and cons of methods and …
Before reacing an absolute conclusion, future work should include more methods for linear regression. One should also include more graphs like heat-maps to visualize the findings and optimal values.

Even though, I have reached an intermediate conclusion, at current time, I don't see this analysis as finalized and is subject to further studies in the future.

Items to look further into:

1) Computational cost vs MSE for higher polynomials. Some tests have not been finalized, due to lack of computational "power".
2)
3) Bias vs Variance relationship.

REFERENCES

[1] Morten Hjorth-Jensen, Computational Physics, Lecture Notes and sample code, Fall 2020, https://github.com/CompPhysics/MachineLearning

[2] Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements of Statistical Learning, Second Edition.
DOI https://**doi**.org/**10.1007**/978-0-387-84858-7; Copyright Information Springer-Verlag New York 2009; Publisher Name Springer, New York, NY; eBook ...

[3Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow, Aurelien Geron. Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow [2nd ed.] 978-1-492-03264-9. 174 47 66MB. English Pages 1150 Year 2019.

[4]Data Science from Scratch-First Principles with Python, Grus.
Afficher les collections Cacher les collections. Identifiant. ISBN : **978-1-491-90142-7**. ISBN : **978-1-491-90142-7**. ISBN : 1-491-90142-X. ISBN : 1-491-90142-X.

[5] Practical Statistics for Data Scientists - Peter Bruce & Andrew Bruce

978-1-491-95296-2 [M] www.allitebooks.com Dedication We would like to dedicate this book to the memories of our parents Victor G. Bruce and Nancy C. Bruce, ...

[6]Python Machine Learning, Sebastian Raschka & Vahid Mirjalili
23. sep. 2017 — Python Machine Learning - von Sebastian Raschka, Vahid Mirjalili (ISBN **978-1-78712-593-3**) bestellen. Schnelle Lieferung, auch auf ...

[7]A Primer on Scientific Programming with Python, Hans Petter Langtangen
ISBN **978-3-662-49886-6**; Free shipping for individuals worldwide. Please be advised Covid-19 shipping restrictions apply. Please review prior to ordering.