UiO : Department of Physics
University of Oslo

Regression analysis and resampling methods

Erik Skaar
Sondre Torp
Mikael Kiste

Contents

Abstract

[1]

1 Introduction

2 Theory

2.1 Standard

$$\beta = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{y}$$

2.2 Ridge

$$\beta = \left(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}\right)^{-1}\mathbf{X}^T\mathbf{y}$$

2.3 Lasso

$$\beta = \text{argmin}_\beta \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j|^q \right\}$$

2.4 k-fold and bootstrap

3 Method

4 Implementation

The three different algorithms discussed in section xxx was implemented in our script. It is a few different versions, but the ëversion contains all you need. All the scripts discussed in this report can be found at our github.

The program was tested on the Frank-function, see equation 1. With an known solution we did a k-fold test and an degree and λ/α test. Both tested was done with the script descriped earlier. The tables below shows the different results.

$$f(x,y) = \frac{3}{4}e^{\left(-\frac{(9x-2)^2}{4}-\frac{(9y-2)^2}{4}\right)} + \frac{3}{4}e^{\left(-\frac{(9x+1)^2}{49}-\frac{(9y+1)}{10}\right)} + \frac{1}{2}e^{\left(-\frac{(9x-7)^2}{4}-\frac{(9y-3)^2}{4}\right)} - \frac{1}{5}e^{\left(-(9x-4)^2-(9y-7)^2\right)} \quad (1)$$

Table 1: This tables shows how the MSE evoles for different degrees. Scikit OLS is to confirm that our implementation is not retarded. For lasso and ridge the λ/α was set to 1e-5. Also, if we go beyond fifth order the OLS solutions starts to crumble.

degree	OLS	SCIKIT	RIDGE	SCIKIT LASSO
2	0.014132	0.014132	0.014132	0.014132
3	0.009929	0.009929	0.009929	0.010263
4	0.007400	0.007400	0.007400	0.010319
5	0.006056	0.006056	0.006056	0.010172

Table 2: The grids ran for 50'000 Monte Carlo cycles. The test ran on a macbook pro 13. It has a dual core CPU. Expected difference is 2.

Size	Normal	MPI	Expected difference	Actual difference
40x40	15.251s	5.991 s	2.000	2.546
60x60	33.923s	13.351 s	2.000	2.541
100x100	92.584s	36.245 s	2.000	2.554

5 Result & Discussion

6 Conclusion

References

[1] Morten Hjorth-Jensen. *Computational Physics*. Lecture notes. 2015. URL: https://github.com/CompPhysics/ComputationalPhysics/blob/master/doc/Lectures/lectures2015.pdf.

7 Appendix