Project 1 FYS-STK4155

Mia Synnøve Frivik, Andrea Myrvang, Max Jan Willem Schuringa, Janita Ovidie Sandtrøen Willumsen
https://github.com/Mia-F/FYS_STK_Project_1.git
(Dated: September 29, 2023)

The aim of this report is to see how different regression method affects the data it is applied to. More concretely, we will look at the three different methods ordinary least squares (OLS), Ridge and LASSO. We will also apply bias variance trade of as well as cross validation on the data sets used to evaluate our models. What we found was that ... regression with parameter ... best fitted the topological data analysis...

I. INTRODUCTION

Machine learning is a powerfull tool usfull in many fields of reasarche. One illustration of its utility is its application to terrain data analysis. Through the creation of terrain models from real data of a specific geographic area, one can effectively anticipate high-risk avalanche zones[4], potentially leading to life-saving interventions. This methods can extends to addressing concerns related to floods, which has become a hot topic this past month following the storm Hans. It can also help in aiding with spatial planning challenges. which is usefull in big citys all over the world. It is fair to say machine learning possesses immense potential to contribute to the solutions of complex and relevant challenges in our modern society, encompassing climate-related issues, urban planning, and life-saving endeavors.

In this report er are going to study three different regression methods, ordinary least squares (OLS), Ridge and LASSO and see how these method compare to eachother when applied to different data sets. First we are going to use the Franke function to make dummy data to validate if our models works. When plotted in the interval $[0, 1]$ this function looks like a mountain and a valley, which is a perfect starting point when we later want to apply these methods on real digital terrain data taken from https://earthexplorer.usgs.gov/. To more accuraltly simulate the realism of practical machine learning scenarios, we will impose limitations on our datasets. Additionally, we will employ techniques such as bootstrapping and cross-validation to expand our dataset size and assess their impact on model validation.

II. THEORY

A. Linear regression methods

Linear regression is a foundational statistical modeling technique employed for the prediction of continuous target variables based on one or more input features. It operates under the assumption that there exists a linear relationship between the input features and the target variable, which is mathematically expressed as:

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p$$

In this equation, $\hat{y}$ denotes the predicted target variable, while $x_1, x_2, \ldots, x_p$ represent the input features. The coefficients $\beta_0, \beta_1, \beta_2, \ldots, \beta_p$ are parameters that correspond to each respective feature. The primary objective of linear regression is to determine these coefficients to establish the optimal linear model that best fits the given data.

Linear regression encompasses several variants, each offering unique characteristics and advantages. In this particular context, we will narrow our focus to three fundamental techniques, Ordinary Least Squares (OLS), Ridge regression, and Lasso regression. These techniques will be employed in the context of analyzing the two-dimensional Franke function.

1. *Ordinary least squares (OLS)*

Ordinary Least Squares (OLS) stands as a fundamental technique in regression analysis, where the aim is to create a model that minimize the diffrense from the observed data and the predicted model.

A linear regression model has the following form :

$$f(\mathbf{X}) = \beta_0 + \sum_{i=1}^{p} X_i \beta_i [2] \tag{1}$$

Where β is the coefficients and X the design matrix. It is importent to note that thise method assume that either the function is linear or approxematly linear.

The cost function for OLS is given by:

$$C(\beta) = \frac{1}{n} = \left\{ (\mathbf{y} - \mathbf{X}\beta)^T (\mathbf{y} - \mathbf{X}\beta) \right\} \tag{2}$$

From this we obtein that the expression for the optimal β is given by:

$$\beta = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \tag{3}$$

2. *Ridge*

$$\beta = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y} \tag{4}$$

Calculations for bothe the optimal β and variance can be found in the apendix

3. LASSO

B. MSE

C. Resampling techniques

The main restriction in machine learning is the amount of data points available to create the model out of. It may be the case where one have done a costfull and time consuming experiments and are left with a small number of data. It is therfull extremely useful to have methods where one can reuse the data multiple times thereby creating a relatively large dataset from the small number of datapoints. In this report we are going to use two different methods, the first is called bootstrap and the second one is cross validation.

1. Bootstrap

The bootstrap method is a resampling procedure that uses data from one sample to generate a sampling distribution by repeatedly taking random samples from the known sample, with replacement[3] This means that if we have a data set D with n data points. The elements in this data set can be representatet in the following way:

$$D = d_1, d_2, d_3, d_4, \ldots, d_n \tag{5}$$

Then by apllying the bootstarp method on this data set one possible output D^* can be:

$$D^* = d_3, d_n, d_4, d_4, \ldots, d_2 \tag{6}$$

From this example we see that one observation can appear multiple times in the new dataset. We can take this method a step further and create "new" data points by extracting multible data points from the dataset and take the mean of all thise values:

$$d_{new} = \frac{1}{k}(d_1, \ldots d_k) \tag{7}$$

This gives us a method of producing lots of "new" dataset from limited data points to train our model with. For each of this data-sets the mean and standard deviation can be calculated to evaluate the model statistically. [1]

One huge advatage of using the boostrap method is that the data can be split in to test and train before shuffeling the data, this means that the test data can bee kept entirely separate from the creation of the model. When we den test the model it will be on a dataset that has nothing to do with crating the model and will therfore show how god the model represent real data. **Write something about large numbers law, also disadvantages and advantage**

2. Cross validation

Cross validation is another method of creating "new" datasets from the original data. This method wrks by splitting the data in k-folds

D. Bias-variance trade-off

The Bias-varinace trade-off is a measuerment on how accuratly a model fit the real data.

III. METHOD

In the first part of this project a function called Franke function was used as the data analysed. The Franke function is given by the following equation:

$$\begin{aligned} f(x,y) = &\frac{3}{4} exp\left(-\frac{(9x-2)^2}{4} - \frac{(9y-2)^2}{4}\right) \\ &+ \frac{3}{4} exp\left(-\frac{(9x+1)^2}{49} - \frac{(9y+1)}{10}\right) \\ &+ \frac{1}{2} exp\left(-\frac{(9x-7)^2}{4} - \frac{(9y-3)^2}{4}\right) \\ &- \frac{1}{5} exp\left(-(9x-4)^2 - (9y-7)^2\right) \end{aligned}$$

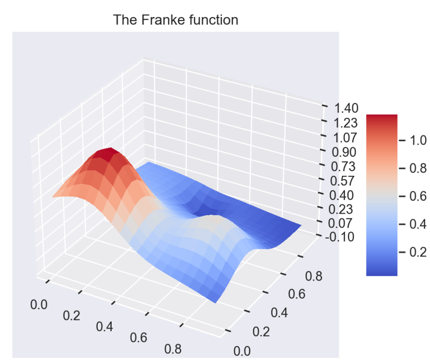

Figure 1. A plot of the Franke function

This function was fitted with the OLS method, and polynomials with varying degrees was used to create the design matrix. Since the design matrix in this case was noninvertible, singular value decomposition was used to create the β-values needed to create a model of the dataset. The mean square error and the R2 score was calculated for both the testing and training datasets. The result of this analyses is shown if figures (3)...... and table...

Next Ridge and LASSO regression was used on the Franke function, to see if these methods have a better fit than what was obtained with OLS. Diffrent values for λ was used to obtain the best fit as possible for each polynomial degree.

IV. RESULTS

The Following result are form the OLS regression on the Franke function. From the OLS regression on the franke

Figure 2.

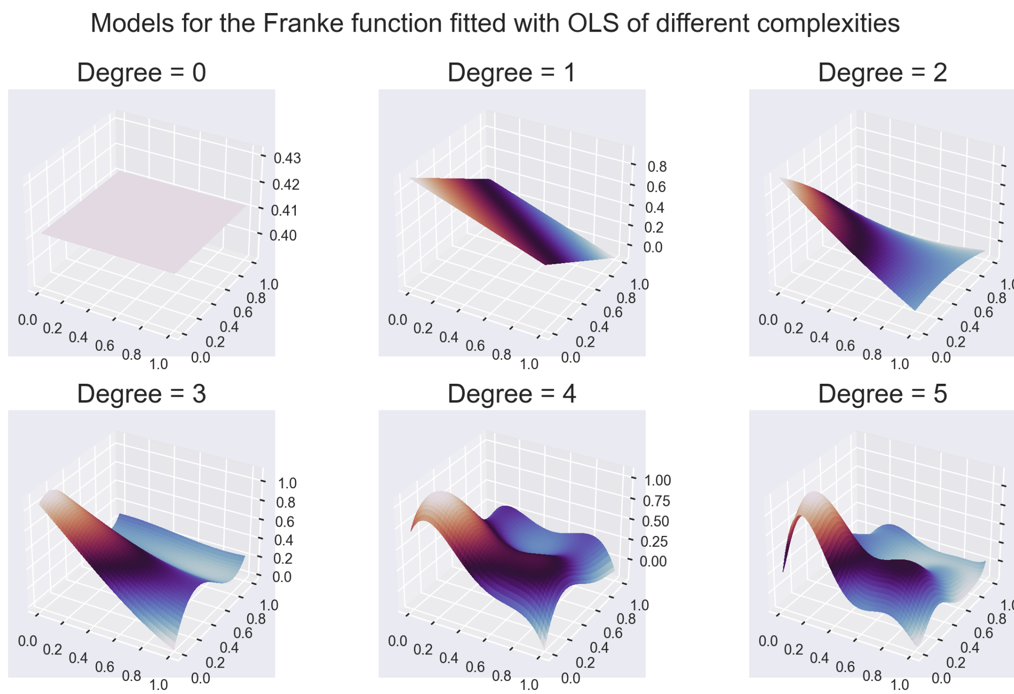

Figure 3. A plot showing how model with different complexities fit the franke function when OLS regession has been used.

V. DISCUSSION

VI. CONCLUSION

REFERENCES

[1] Jason Brownlee. A Gentle Introduction to the Bootstrap Method. https://machinelearningmastery.com/a-gentle-introduction-to-the-bootstrap-method/, 2019.

[2] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference and prediction*. Springer, 2 edition, 2009.

[3] Eberly College of Science. Bootstrapping Methods. https://online.stat.psu.edu/stat500/lesson/11/11.2/11.2.1.

[4] Hong Wen, Xiyong Wu, Xin Liao, Dong Wang, Kaiyang Huang, and Bernd Wünnemann. Application of machine learning methods for snow avalanche susceptibility mapping in the parlung tsangpo catchment, southeastern qinghai-tibet plateau. *Cold Regions Science and Technology*, 198:103535, 2022.

Appendix A: Mean values and variances calculations

The main regression method used in this report is the ordinary least squares method. This appensix shows the calculations for some of the equations used to produce the results shiwn in this report.

We have assumed that our data can be described by the continous function $f(\boldsymbol{x})$, and an error term $\boldsymbol{\epsilon}$ $N(0, \sigma^2)$. If we approximate the function with the solution derived from a model $\tilde{\boldsymbol{y}} = X\boldsymbol{\beta}$ the data can be described with $\boldsymbol{y} = X\boldsymbol{\beta} + \boldsymbol{\epsilon}$. The expectation value

$$
\begin{aligned}
\mathbb{E}(\boldsymbol{y}) &= \mathbb{E}(X\boldsymbol{\beta} + \boldsymbol{\epsilon}) \\
&= \mathbb{E}(X\boldsymbol{\beta}) + \mathbb{E}(\boldsymbol{\epsilon}) && \text{where the expected value } \boldsymbol{\epsilon} = 0 \\
\mathbb{E}(y_i) &= \sum_{j=0}^{P-1} X_{i,j}\beta_j && \text{for the each element} \\
&= X_{i,*}\beta_i && \text{where }_* \text{ replace the sum over index } i
\end{aligned}
$$

The variance for the element y_i can be found by

$$
\begin{aligned}
\mathbb{V}(y_i) &= \mathbb{E}\big[(y_i - \mathbb{E}(y_i))^2\big] \\
&= \mathbb{E}(y_i^2) - (\mathbb{E}(y_i)^2) \\
&= \mathbb{E}((X_{i,*}\beta_i + \epsilon_i)^2) - (X_{i,*}\beta_i)^2 \\
&= \mathbb{E}((X_{i,*}\beta_i)^2 + 2\epsilon_i X_{i,*}\beta_i + \epsilon^2) - (X_{i,*}\beta_i)^2 \\
&= \mathbb{E}((X_{i,*}\beta_i)^2) + \mathbb{E}(2\epsilon_i X_{i,*}\beta_i) + \mathbb{E}(\epsilon^2) - (X_{i,*}\beta_i)^2 \\
&= (X_{i,*}\beta_i)^2 + \mathbb{E}(\epsilon^2) - (X_{i,*}\beta_i)^2 \\
&= \mathbb{E}(\epsilon^2) = \sigma^2
\end{aligned}
$$

The expression for the optimal parameter

$$
\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}
$$

We find the expected value of $\hat{\boldsymbol{\beta}}$

$$
\begin{aligned}
\mathbb{E}(\hat{\boldsymbol{\beta}}) &= \mathbb{E}((\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}) \\
&= (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\mathbb{E}(\boldsymbol{y}) && \text{using that } \boldsymbol{X} \text{ is a non-stochastic variable} \\
&= (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta} && \text{using } \mathbb{E}(\boldsymbol{y}) = \boldsymbol{X}\boldsymbol{\beta} \\
&= \boldsymbol{\beta}
\end{aligned}
$$

we can find the variance by

$$
\begin{aligned}
\mathbb{V}(\hat{\boldsymbol{\beta}}) &= \mathbb{E}\big[(\hat{\boldsymbol{\beta}} - \mathbb{E}(\hat{\boldsymbol{\beta}}))^2\big] \\
&= \mathbb{E}(\hat{\boldsymbol{\beta}}\hat{\boldsymbol{\beta}}^T) - \mathbb{E}(\hat{\boldsymbol{\beta}})^2 \\
&= \mathbb{E}(((\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y})((\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y})^T) - \hat{\boldsymbol{\beta}}\hat{\boldsymbol{\beta}}^T \\
&= \mathbb{E}((\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}\boldsymbol{y}^T\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}) - \hat{\boldsymbol{\beta}}\hat{\boldsymbol{\beta}}^T \\
&= (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\mathbb{E}(\boldsymbol{y}\boldsymbol{y}^T)\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1} - \hat{\boldsymbol{\beta}}\hat{\boldsymbol{\beta}}^T \\
&= (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T(\boldsymbol{X}\boldsymbol{\beta}\boldsymbol{\beta}^T\boldsymbol{X}^T + \sigma^2)\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1} - \hat{\boldsymbol{\beta}}\hat{\boldsymbol{\beta}}^T \\
&= \boldsymbol{\beta}\boldsymbol{\beta}^T + \sigma^2((\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}) - \hat{\boldsymbol{\beta}}\hat{\boldsymbol{\beta}}^T \\
&= \sigma^2(\boldsymbol{X}^T\boldsymbol{X})^{-1}
\end{aligned}
$$

Appendix B: Bias-variance trade-off