Analysis of regression methods
FYS-STK3155 - Project 1

Erik Berthelsen, Morten Taraldsten Brunes, Satu Innanen, Johanna Tjernström
University of Oslo

(Dated: September 25, 2025)

Regression methods are a starting point for learning machine learning as they bridge the gap between statistics. It is the simplest form for exploring relationships between variables by optimizing cost functions and acts as base knowledge for machine learning practitioners. There is a need to explore Ordinary Least Squares (OLS), Ridge and LASSO to broaden the knowledge for these regression methods. We used Runge's function as an example to analyze different values for hyperparameters such as polynomial degree, learning rate and number of data points. We analyzed the results by examining Mean Squared Error (MSE) and R2 metrics, and regression coefficients as a function of the polynomial degree. We explored the methods for choosing the learning rate and resampling techniques to analyze their influence on model prediction. The results show that **WE ARE NOT FINISHED YET**. These results imply that when analyzing datasets with machine learning and statistics, several regression methods and hyperparameters need to be explored to be able to choose the optimal model. The bias-variance trade-off is used as a valuable technique to determine the optimal hyperparameters for model generation.

I. INTRODUCTION

Regression analysis has been a cornerstone of statistics and machine learning. When used right, it provides insightful information about the relationships between different variables. Even if regression algorithms are considered to be more simplistic forms of machine learning, they provide the basics of how machine learning can be used in more complex cases.

In this report we consider the Ordinary Least Squares (OLS), Ridge regression and LASSO regression to compare and contrast these methods and their results when fitting to specific one-dimensional functions. For this purpose, we will use Runge's function. In addition to this, we will explore the optimizable parameters present in some of these methods, for example the lambda values in Ridge regression, as well as optimization through the usage of the gradient descent method. The Gradient descent method will also consider different ways of updating the learning rate, comparing update by momentum, ADAgrad, RMSprop and ADAM. Stochastic gradient descent will also be considered. In addition to this, we will link the regression analysis with a statistical analysis. To this end, we consider resampling, specifically using the bootstrap method, and other statistical methods such as the bias-variance tradeoff and cross-validation. These statistical methods give additional insight into the positive and negative aspects of the methods considered toward the purpose of fitting to a specific one-dimensional function.

The methods used in this report are described in greater detail in section II. The results of the regression and statistical analysis, as well as any insights taken from them, are presented in section III. Finally, the conclusions drawn will be described in section IV. Appendix A and B contains derivations for some of the equations described in section II.

II. METHODS

The function used in this project is the one-dimensional Runge's function:

$$f(x) = \frac{1}{1+25x^2}, \tag{1}$$

where $x \in [-1, 1]$ with a uniform distribution. The polynomial fit, and hence the design matrix, X, is created by $[x, x^2, \dots]$.

We use three different linear regression methods to this function: the OLS regression, Ridge regression and LASSO regression. The goal is to explain the aforementioned Runge's function with regard to X through a linear relationship [4].

A. Ordinary Least Squares

The cost function of the OLS method is defined as:

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

where θ are the optimal model parameters, n is the number of samples, $\boldsymbol{y}$ are the real data, $\boldsymbol{X}$ is the design matrix. The optimal parameters are found by minimizing the cost function and results in (see Appendix XXX):

$$\hat{\boldsymbol{\theta}}_{OLS} = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}. \tag{3}$$

B. Ridge Regression

In Ridge regression, a hyperparameter λ is added in the cost function resulting in:

$$C(\boldsymbol{X},\boldsymbol{\theta}) = \left\{(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\theta})^T(\boldsymbol{y}-\boldsymbol{X}\boldsymbol{\theta})\right\} + \lambda\boldsymbol{\theta}^T\boldsymbol{\theta}, \quad (4)$$

Thus, the optimal parameters are:

$$\hat{\boldsymbol{\theta}}_{\text{Ridge}} = \left(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y} \quad (5)$$

The hyperparameter, also called the penalty, causes the parameters to shrink and introduces bias to the estimate. This can improve the model by decreasing the variance, when the hyperparameter is tuned to its optimal value (cite XXX).

C. Gradient descent

The goal is to optimize parameters (design matrix) in the model in a way that the model produces predictions that are close to the targets (the data, Eq. 1) [1]. Gradient descent is an optimization method trying to minimize the cost function by XXXX.

D. Evaluation metrics

To evaluate the performance of the different methods, we use the mean squared error (MSE) and the R^2 as evaluations metrics. The MSE is given by:

$$MSE(\boldsymbol{y},\tilde{\boldsymbol{y}}) = \frac{1}{n}\sum_{i=0}^{n-1}(y_i - \tilde{y}_i)^2, \quad (6)$$

The R^2 is given by:

$$R^2(\boldsymbol{y},\tilde{\boldsymbol{y}}) = 1 - \frac{\sum_{i=0}^{n-1}(y_i-\tilde{y}_i)^2}{\sum_{i=0}^{n-1}(y_i-\bar{y})^2} \quad (7)$$

E. References added for exercise week 39

As mentioned in exercise week 39 a reference to Hastie: [2]. We also have a reference to Scikit-learn: [3]

F. Bias-variance tradeoff

We rewrite the cost function (MSE) with the variance of the model, the bias between the true data and the model, and the variance of the noise (see Appendix V B), resulting in:

$$\mathbb{E}[(y-\tilde{y})^2] = Bias[\tilde{y}] + Var[\tilde{y}] + \sigma^2 \quad (8)$$

Here, the bias-term describes the systematic offset between the predictions and the target. The bias is typically high, if the model is too simple. The variance of the model describes the fluctuation of predictions between different training sets. If the variance is high, the model is typically overfitting. The variance of the error ϵ describes the irreducible error.

G. Implementation

H. Use of AI tools

In this project, we have used AI tools ChatGPT and Microsoft Copilot in the following ways in the production of the code and the report:

- For exercise week 39 a prompt to Microsoft Copilot with LLM GPT-5 is executed with prompt: "Tell me in depth about Lasso regression". Text file with prompt and answer available for download

- **Remember to add information about used LMM for Lasso regression in project 1 report. Questions and provided code form LLM available as a jupyter notebook, create link to github file**

III. RESULTS AND DISCUSSION

A. Heatmap added for exercise week 39

Heatmap added for exercise in week 39 as an example

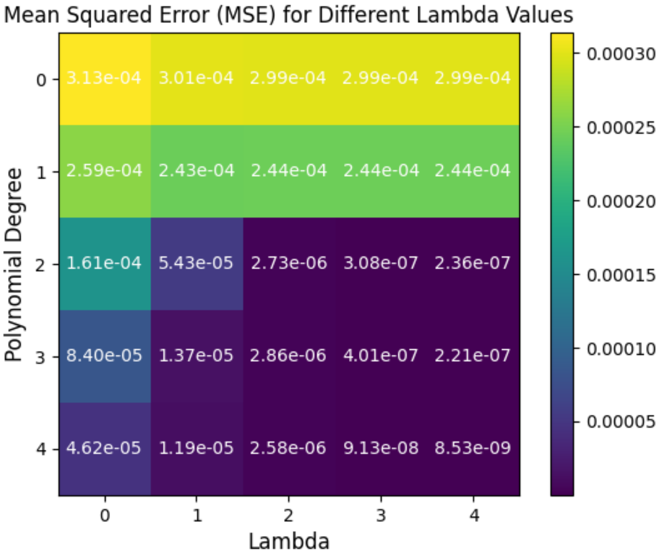

Figure 1: Mean Square Error (MSE) for different Lambda values with respect to polynomial degree

As shown in Figure 1, we have included a heat map figure and a reference in the text.

B. Bias-variance tradeoff

Figure 2 shows the bias-variance tradeoff (here for week 39 exercises).

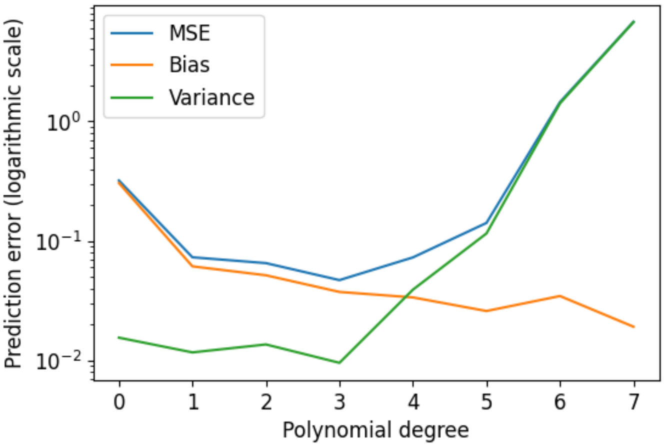

Figure 2: Mean squared error (MSE), bias and variance as a function of the OLS model complexity (from exercises week 38).

IV. CONCLUSION

V. CODE AVAILABILITY

The codes used in this project can be found at: `https://github.com/johtj/data_analysis_ml_projects`.

[1] Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

[2] Hastie, T., R.Tibshirani, and J.Friedman (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition. Springer Series in Statistics.* Springer, New York.

[3] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

[4] van Wieringen, W. N. (2023). *Lecture notes on ridge regression.*

APPENDIX

A. Derivatives of OLS and Ridge

B. Bias-variance tradeoff

We start deriving the bias-variance expression with the equation of the MSE:

$$MSE = \mathbb{E}[(y-\tilde{y})^2] \tag{9}$$

We expand Equation 9:

$$MSE = \mathbb{E}[y^2] - 2\mathbb{E}[y\tilde{y}] + \mathbb{E}[\tilde{y}^2]$$

Next, we consider each of these terms of the expanded MSE equation:

$$\mathbb{E}[y^2] = \mathbb{E}[(f+\epsilon)^2] = \mathbb{E}[(f^2] + 2\mathbb{E}[f\epsilon] + \mathbb{E}[\epsilon^2] = f^2 + \sigma^2$$

$$\mathbb{E}[y\tilde{y}] = \mathbb{E}[(f+\epsilon)\tilde{y}] = \mathbb{E}[f\tilde{y}] + \mathbb{E}[\epsilon\tilde{y}] = f\mathbb{E}[\tilde{y}]$$

$$\mathbb{E}[\tilde{y}^2] = var[\tilde{y}] + (\mathbb{E}[\tilde{y}])^2$$

We write the Equation 9 again as:

$$\mathbb{E}[(y-\tilde{y})^2] = f^2+\sigma^2-2f\mathbb{E}[\tilde{y}]+var[\tilde{y}] = \mathbb{E}[(f-\mathbb{E}[\tilde{y}])^2]+var[\tilde{y}]+\sigma^2$$

This leads to:

$$\mathbb{E}[(y-\tilde{y})^2] = Bias[\tilde{y}] + Var[\tilde{y}] + \sigma^2$$