

Contents

1	Introduction	2
2	Objectives	2
2.1	Explore issues that arise from singularity.	2
2.2	Understand singular value decomposition (SVD).	3
2.3	Investigate how software deals with singularity when building linear models.	3
3	Methodology	3
4	Results and discussion	3
4.1	Python's statsmodel module	3
4.2	R's lm function	3
4.3	Impact of multicollinearity	4
4.4	Methods to detect collinearity	4
4.5	Confounding variables	4
5	Conclusion	4
6	Future recommendations	4
7	Appendix	4
8	References	4

1 Introduction

Linear regression is widely used in many applications to model linear relationships between a set of predictors and a response. One of the more common and simpler methods of fitting such a model is known as the ordinary least squares method (OLS), which is found to be a Best Linear Unbiased Estimator (BLUE) by the Gauss-Markov theorem, that is, it gives the unbiased estimates of each regression coefficient with the lowest possible variance, provided that assumptions are satisfied.

The motivation for this project arises from one of these assumptions, that is, the data matrix, \mathbf{X} , is full rank, i.e. none of the predictors are (nearly) perfectly correlated with each other. In cases where this is not true, we find that OLS fails to acquire any results. In such cases, there are several methods we can employ to address this issue, whether to drop one of the correlated variables, or to combine them, etc., based on the discretion of the data analyst. In this project, we investigate how two of the commonly used software for regression analysis, Python (`sklearn` and `statsmodel`) and R each deal with the collinearity problem.

Furthermore, we embrace this opportunity to explore further on the topic of multicollinearity, such as its impact on the accuracy of model prediction and model interpretation. We have also identified several known methods of detecting collinearity in the data, whether through hypothesis testing, or by using measures such as the variance inflation factor (VIF) and condition numbers/indices. In addition to aforementioned methods of addressing collinearity, shrinkage methods such as ridge regression and lasso were also considered. Another potential pitfall in regression analysis, that is confounding variables, were also briefly explored.

It should be noted that there are different conventions to what collinearity and multicollinearity mean according to varying sources. In this report, we will use these two terms interchangeably.

2 Objectives

There are three main objectives for this project:

2.1 Explore issues that arise from singularity.

In practical settings, it is unlikely to obtain a set of perfectly correlated data due to inevitable noise. However, even near perfect collinearity can lead to singularity and subsequently failure of OLS.

2.2 Understand singular value decomposition (SVD).

The least squares method estimates the regression coefficients of a model with p predictors by fitting it to a dataset with n observations using the following equation

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

where β is a $p \times 1$ column matrix, \mathbf{X} is a $n \times p$ data matrix, and \mathbf{y} is a $n \times 1$ column matrix containing the observed response values. With the presence of perfect collinearity, the matrix $\mathbf{X}^T \mathbf{X}$ is singular and hence $(\mathbf{X}^T \mathbf{X})^{-1}$ does not exist. However, the least squares solver of Python uses the Moore-Penrose inverse (or pseudoinverse) which gives the best approximation of $(\mathbf{X}^T \mathbf{X})^{-1}$, and can be easily computed using the SVD of $\mathbf{X}^T \mathbf{X}$.

2.3 Investigate how software deals with singularity when building linear models.

3 Methodology

4 Results and discussion

4.1 Python's statsmodel module

(include screenshot of results)

- Looks like it evenly distributes weights throughout the correlated predictors.
- Apparently uses pseudoinverse, and that the results obtained don't carry meaning.

4.2 R's lm function

(include screenshot of results)

- Looks like it dropped one collinear column and add the two coefficients together.
- Seems to combine collinear columns.
- Gives a warning when highly correlated variables are detected
- Option available to abort process upon detection of high multicollinearity.

- 4.3 Impact of multicollinearity**
- 4.4 Methods to detect collinearity**
- 4.5 Confounding variables**
- 5 Conclusion**
- 6 Future recommendations**
- 7 Appendix**
- 8 References**