Comparing the performance of parametric and non-parametric linear regression models and documenting the efficacy of a multinomial logistic regression model on an obesity dataset

Elias Crum, Genevieve Baddoo, Daniel Araujo

Introduction

The linear regression models presented by Ogutu *et al.* (2012) are utilized to perform the estimation of simultaneous effects of all genes in a select genome and combine those estimates to predict the total genomic breeding value (GEBV). Genomic Selection (GS), which is the prediction of GEBVs using dense molecular markers, is a becoming a key component of efficient and cost-effective breeding programs in scientific research. Predicting GEBVs can be undertaken using multiple linear regression models that vary in complexity, computational efficiency, and predictive accuracy. The accurate prediction of GEBVs is a recurring challenge in plant and animal breeding, thus it is important to identify linear regression methods best suited to GS and determine when certain models are optimal. Therefore, Ogutu *et al.* (2012) evaluate the relative performance of six regularized (penalized) linear regression models for GS: ridge regression (RR), ridge regression best linear unbiased prediction (RR-BLUP), the least absolute shrinkage and selection operator (lasso), elastic net, adaptive lasso and adaptive elastic net (ADAENET). The success of using these regularization methods, especially in genomic selection, relates to their use of penalties. These penalties facilitate fitting models with predictors that run into thousands of variables, including many irrelevant to the response, exceeding the sample size, or are highly correlated, with high efficiency and prediction accuracy.

The data used by the researchers in Ogutu *et al.* (2012) is comprised of an outbreed of 1000 individuals. These individuals were simulated over 1000 generations, followed by 150 individuals over 30 generations. Biallelic SNP markers (n = 9990) were distributed on 5 chromosomes for a total of 1998 SNPs. For the analysis, data corresponding to the last generation of the simulated pedigree of 20 sires, each mated to 10 different dams and yielding 15 progenies per dam were selected, for a total of 3000 progenies in the dataset. For every full-sib family of 15 progenies, 10 progenies were genotyped and phenotyped (n = 2000 progenies). The remaining 5 progenies were genotyped but not phenotyped (n = 1000 progenies). The 3000 progenies served as the candidates for genomic prediction in this study. One of the aims of this statistical analysis is to predict the true genomic value (TGV), which is the true expectation of the phenotypes of the 1000 non-phenotyped candidates. The other aim of this analysis to predict the true breeding value (TBV), which is the true expectation of the phenotypes of the progenies of the 1000 non-phenotyped candidates.

The authors then describe each of the 6 models utilized for the dataset of 3000 progenies. The first model, **Ridge regression (RR)**, is ideal to use if there are many predictors with non-zero coefficients and drawn from a normal distribution. RR performs well with many predictors each having small effect and prevents coefficients of linear regression models with many correlated variables from being poorly determined and exhibiting high variance. This type of estimator solves the regression problem in aim of the paper by using $\ell 2$ penalized least squares. Λ (lambda) is an important variable in the RR equation, as it is the tuning parameter that regulates the strength of the penalty (linear shrinkage) by determining the relative importance of

the data-dependent empirical error and the penalty term. While this variable is dependent on the type of dataset, it can be determined by using cross-validation for example. **Ridge regression BLUP** is similar to RR except that it estimates the penalty parameter by Restricted Maximum Likelihood (REML). **Lasso regression** is a type of linear model that performs an $\ell 1$ penalty, which adds a penalty equal to the absolute value of the magnitude of coefficients. This type of regularization can result in sparse models with few coefficients. This model is widely used with large datasets such as the genomics one used in Ogutu *et al.* Although this model provides a fast and efficient algorithm, it's shortcomings include lacking the oracle property (estimates a subset of true parameters with zero coefficients as exactly zero with probability tending to 1), being unstable with high-dimensional, and not selecting more variables than the sample size. **Adaptive lasso regression** solves the lack of the oracle property in lasso by adding data-driven weights in its equation. Adding these weights allows adaptive lasso to apply different amounts of shrinkage to different coefficients and hence to more severely penalize coefficients with small values. The **elastic net (ENET) regression** model uses a mixture of the $\ell 1$ (lasso) and $\ell 2$ (ridge regression) penalties to handle extreme correlations among predictors. $\ell 1$ performs automatic variable selection, while $\ell 2$ encourages grouped selection and stabilizes the solution paths with respect to random sampling, thereby improving prediction. An ENET model is able to select groups of correlated features when the groups are not known in advance. Lastly, **adaptive elastic net** is a mixture of adaptive lasso and elastic net that is able to confer the oracle property to elastic net regression. It alleviates the instability of the adaptive lasso with high-dimensional data inherited from lasso regression. Together, these 6 models are applied to the GS dataset to test each of their predictive accuracies for true genomic value (TGV) and true breeding value (TBV).

Methods

To test the various regression models presented by Ogutu *et al.* (2012), we utilized a public dataset from the UCI Machine Learning Repository. The data set is titled "Estimation of obesity levels based on eating habits and physical condition Data Set." This dataset contains data regarding the obesity levels of individuals from Mexico, Peru, and Colombia based on their eating habits and physical condition. The data contain 17 attributes and 2,111 records. For each individual, the column 'NObesity' denotes the participant's actual obesity level on a continuum -- Insufficient Weight, Normal Weight, Overweight Level I, Overweight Level II, Obesity Type I, Obesity Type II and Obesity Type III. Notably, 77% of the data were generated synthetically using the Weka tool and the SMOTE filter. The remaining 23% of the data were collected directly from users through a web platform.

Using the programming language R, we first applied the methods of Ogutu *et al.* (2012) to compare the different linear regression (ridge, lasso, elastic net) models. To begin, we fixed the obesity dataset to contain only numeric variable responses (see code below for details). Once the data was reformatted, we split the 2111 rows into 5 training / testing sets to be used in building the models using the 'glmnet' R package (Friedman; Hastie; Tibshirani, 2010). We then created a recursive loop to build the different models tested by Ogutu *et al* (2012). The loop allowed for the creation of 10 different models, and each model was designated by a different alpha value (ranging from 0 to 1 by increments of 0.1). When alpha=0, a Ridge model is fit to the

data, when alpha=1, a Lasso model is fit to the data. To build each type of model, the cv.glmnet R function was utilized. The cv function performs 10-fold cross-validation to determine the lambda value to use for model building. The glmnet function then built the model via the x and y indicated and the family=Gaussian to avoid problems with the correlated nature of the covariates. As done by Ogutu *et al.* (2012), we then analyzed the mean root mean square error (RMSE) of each set of models produced by the different alpha values, computed using the 'MLmetrics' R package (Yan, 2016). Each of the representative RMSE values were then visualized via a scatter plot using the 'ggplot2' R package (Wickham, 2016).

Similar methods were used to build and test a non-parametric linear model using multivariate adaptive regression splines (MARS). The MARS models were produced using the 'earth' R package (Milborrow; Hastie; Tibshirani, 2014). Similar to the previously performed methods replication, we split the dataset into 5 training / testing sets. A separate MARS model was built from each data partition and analyzed to determine the RMSE of the produced models. The mean RMSE value of the five MARS models was then plotted with the RMSE values from the previous analysis using visualization methods mentioned previously.

In parallel to the two linear model production methods, multinomial logistic (ML) regression models were built using very similar methods. The ML regression models were computed using the 'nnet' R package (Venables; Ripley, 2002). The data set was then manipulated to make the Y (test) variable (the patient condition in the case of the obesity dataset) categorical instead of numeric similar to the linear models. Just as in the previously performed methods, we split the dataset into 5 training / testing sets. A separate ML model was built from each data partition and analyzed to determine the RMSE of the produced models. The mean RMSE value of the five ML models was then plotted with the RMSE values from the previous analysis using visualization methods mentioned previously.

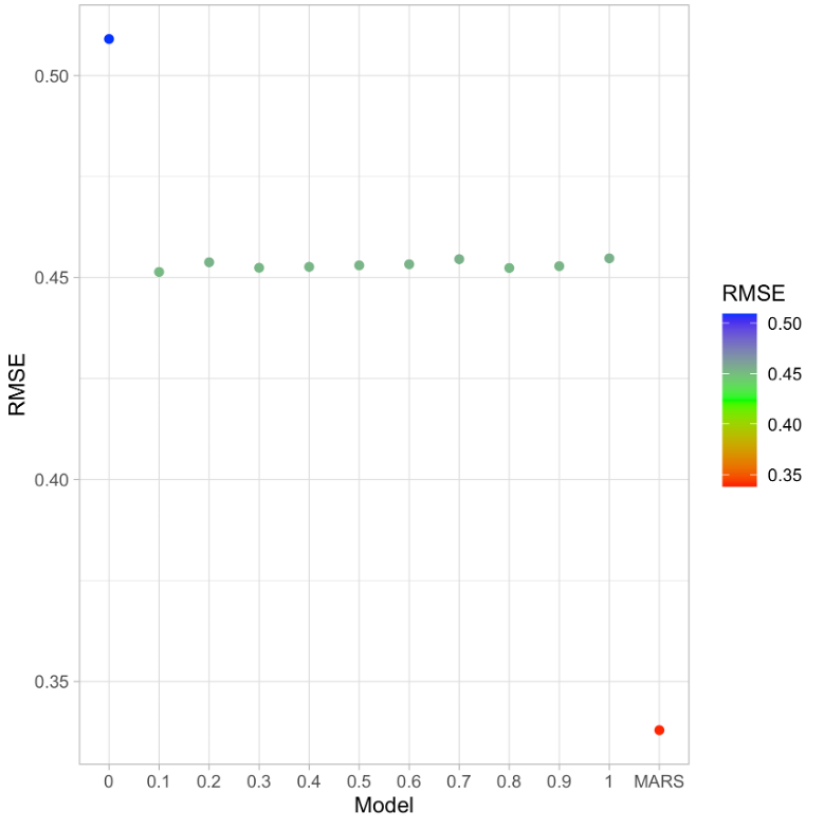
Results and Discussion

When assessing the average RMSE values for the Ridge, Lasso, Elastic Net regression models, the lowest RMSE value of 0.451 was found using an alpha value of 0.1. The highest RMSE value observed was 0.509 at an alpha value of 0 (RR). For every alpha value greater than 0, the RMSE values only varied at the thousandth decimal place. Because there was low RMSE variation between the different alpha values, it is difficult to determine an optimal alpha for model building. Rather, we determined that the alpha value used is not particularly determinant of model RMSE and, therefore, overall model efficacy.

When assessing the average RMSE value from the MARS models, the value observed was 0.338. Importantly, the RMSE value of the MARS model was substantially lower than the other regression models tested, suggesting that the MARS model was the best at modelling our obesity dataset. The RMSE values obtained from the different regression models are shown below in **Figure 1**.

Figure 1. Linear Model RMSE Values

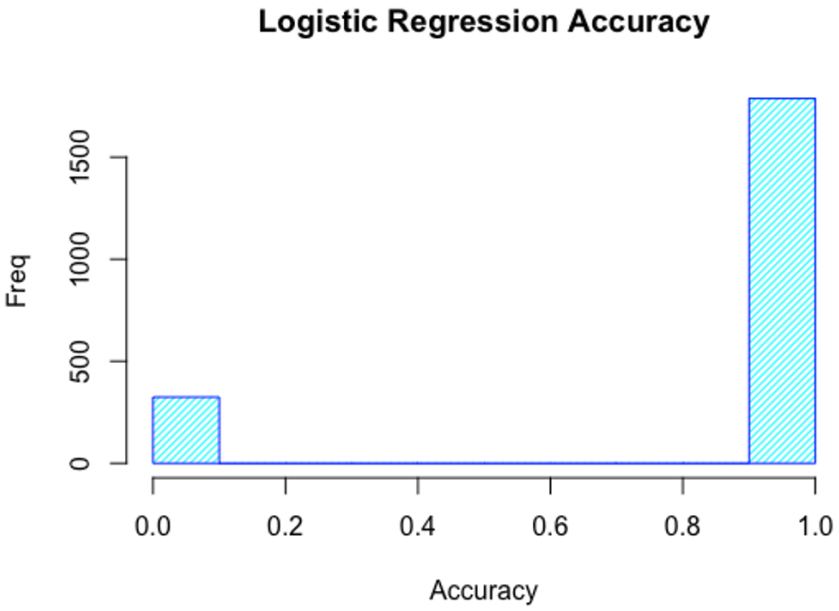
Model (α value)	RMSE
0	0. 5090602
0.1	0. 4514038
0.2	0. 4537953
0.3	0. 4524179
0.4	0. 4526462
0.5	0. 4530231
0.6	0. 4532990
0.7	0. 4545571
0.8	0. 4523694
0.9	0. 4528245
1	0. 4547641
MARS	0. 3379438



The logistic model we trained and tested on the obesity data set performed with 86% accuracy. Interestingly, the logistic models we generated performed very well (>95% accuracy) on all data partitions other than data partition 3. The model trained and tested on partition 3 was only 43% which suggests some problem with the data or model. Further work into the exact problem with the data or model is necessary to explain the observed discrepancy. The results illustrating the accuracy of the logistic models can be seen below in **Figure 2**.

Figure 2. Logistic Model Accuracy

Data Partition	Accuracy
1	0.9431280
2	0.9597156
3	0.4336493
4	0.9526066
5	0.9432624



As illustrated in the raw data and figures, the MARS model is the most accurate linear model for the obesity dataset we utilized for training and testing. The MARS model is likely better suited for the obesity data than the other linear models because it is a nonparametric linear regression model. Because the obesity dataset we used is an obesity focused dataset, the data is unlikely normally distributed due to it being skewed toward heavier individuals. It makes sense then for the MARS model to have the lowest RMSE value. It should also be noted that the number of covariates in the obesity dataset was 16. In the paper by Ogutu *et al.* (2012), Ridge, Lasso, and Elastic Net methods were tested on genomic data, which have many more covariates than our obesity data set. This could also be another reason why MARS provided a more accurate regression model.

The logistic model trained and tested on the obesity data was shown to accurately categorize patients during testing for the most part (accuracy of 86%). The obesity data could be utilized for numerous different research aims. We applied a number of linear regression models as well as a logistic model to the obesity data set to show that both type of regression model can be trained to accurately predict different outcomes. In different situations, different models are helpful. For instance, the obesity data set could be used to train a model that will be implemented to determine patient eligibility for a high cholesterol drug clinical trial. Only patients that are classified as 'Obesity_Type_I' are wanted for the trial. The ideal model to identify individuals for this clinical trial would be a multinomial logistic regression model due to the classification centered goal. Conversely, if another group of scientists are trying to determine what lifestyle factors are most predictive of an individual being classified higher on obesity classification spectrum, a linear model would be more informative due to the more continuous nature of the studied phenomenon. This paper showed that MARS linear regression modeling and multinomial logistic regression modeling are viable options for these two aims when working with the obesity data set.

Conclusion

The results of implementing this dataset to these linear regression models reveal that the Ridge, Lasso, and Elastic Net regression models do not perform particularly well in comparison to the MARS nonparametric regression model. The differences observed are likely due to a non-normally distributed data set used to train the models. Additionally, the application of Ridge, Lasso, and Elastic Net regression models are best at modeling high variable, high intercorrelation data sets, such as genomic data. The obesity dataset we used had intercorrelation, but it only had 16 variables for 2111 subjects. Thus, the obesity dataset may have not be the perfect for the Ridge, Lasso, and Elastic Net regression models to make accurate predictions. After successfully applying the methods of Ogutu *et al.* (2012), we found that while effective for genomic data sets, for datasets with a relatively small number of non-normal, intercorrelated variables, the Ridge, Lasso, and Elastic Net regression models did not perform as well as the MARS nonparametric model. Further analysis could be done to assess whether the MARS linear model is the best linear regression model for the obesity data set analyzed. Additionally, we showed that a multinomial regression model could be trained and tested on the obesity set as well.

Sources:

Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, *33*(1), 1.

Milborrow, S., Hastie, T., & Tibshirani, R. (2014). Earth: multivariate adaptive regression spline models. *R package version*, *3*(7).

Ogutu, J. O., Schulz-Streeck, T., & Piepho, H. P. (2012). Genomic selection using regularized linear regression models: ridge regression, lasso, elastic net and their extensions. In *BMC proceedings* (Vol. 6, No. 2, pp. 1-6). BioMed Central.

Venables WN, Ripley BD (2002). *Modern Applied Statistics with S*, Fourth edition. Springer, New York.

Wickham H (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York.

Yan, Y. (2016). MLmetrics: Machine learning evaluation metrics. *R package version*, *1*(1).

Data:

https://archive.ics.uci.edu/ml/datasets/Estimation+of+obesity+levels+based+on+eating+habits+and+physical+condition+#

Contributions:

Daniel – Paper selection, Ridge Regression, Lasso, Elastic Net, and MARS coding, final write-up editing

Eli – Data selection/manipulation, figure production, multinomial regression coding, methods/results/conclusion write up sections, code organization, clean-up, and testing

Genevieve – Write up introduction, editing, and testing code