Scientific Research Publishing

Journal of Data Analysis and Information Processing, 2019, 7, 14-27
http://www.scirp.org/journal/jdaip
ISSN Online: 2327-7203
ISSN Print: 2327-7211

Ordinal Outcome Modeling: The Application of the Adaptive Moment Estimation Optimizer to the Elastic Net Penalized Stereotype Logit

André A. A. Williams

Center for Healthcare Delivery Science, Nemours Children's Specialty Care, Jacksonville, USA
Email: wandre719@gmail.com

How to cite this paper: Williams, A.A.A. (2019) Ordinal Outcome Modeling: The Application of the Adaptive Moment Estimation Optimizer to the Elastic Net Penalized Stereotype Logit. *Journal of Data Analysis and Information Processing*, **7**, 14-27.
https://doi.org/10.4236/jdaip.2019.71002

Received: December 17, 2018
Accepted: February 19, 2019
Published: February 22, 2019

Abstract

Penalized ordinal outcome models were developed to model high dimensional data with ordinal outcomes. One option is the penalized stereotype logit, which includes nonlinear combinations of parameter estimates. Optimization algorithms assuming linearity and function convexity were applied to fit this model. In this study the application of the adaptive moment estimation (Adam) optimizer, suited for nonlinear optimization, to the elastic net penalized stereotype logit model is proposed. The proposed model is compared to the L1 penalized ordinalgmifs stereotype model. Both methods were applied to simulated and real data, with non-Hodgkin lymphoma (NHL) cancer subtypes as the outcome, with results presented and discussed.

Keywords

Stereotype Logit, Elastic Net Penalty, Adam Optimizer, Non-Hodgkin Lymphoma

1. Introduction

Many research studies seek to predict related outcomes given a set of independent variables or to quantify the relationship between them. In certain instances, the outcome of interest is ordinal. Ordinal variables are defined as having distinct ordered levels; however, the distance between the levels cannot be ascertained. An example of an ordinal variable is cancer stage. Take, for instance, testicular seminoma, a germ cell tumor in the sperm of the testes [1]. This cancer can be classified according to stage. The stages are:

1) Tumor stage 1, cancer has not spread beyond the testicle.
2) Tumor stage 2, cancer has spread to the blood or lymphatic vessels.

3) Tumor stage 3, cancer has spread beyond the lymphatic and blood vessels nodes to the spermatic cord.

4) Tumor stage 4, cancer has spread beyond previously mentioned areas to other parts of the body.

The ordering of categories is evident. The aim of statistical and machine learning models is to quantify the relationship between covariates and associated outcome so that one can predict the outcome variable and assess the relationship between the two with statistical significance. The range of ordinal outcome models includes cumulative logit, proportional odds model, adjacent-category logit [2], and stereotype logit [3]. These procedures assume there are more observations than independent variables, or covariates. Another assumption is the resulting parameter estimates follow a normal distribution.

In addition, we now live in an era of high dimensional data, and massive amounts of information are being collected [4]. These data are used to better understand and analyze related issues. However, this comes at a cost, and traditional methods are ill-equipped to utilize these datasets. These data may have more variables than observations. The distributions of the parameter estimates may not follow a normal distribution. We may collect genetic data, demographic data, and clinical data, resulting in an analysis data set containing thousands of variables with a few hundred observations when evaluating health conditions with associated ordinal outcomes [5]; the distribution of the parameter estimates may not follow a normal, or other known, distribution.

Penalized ordinal outcome models were developed to analyze high dimensional data with ordinal outcomes. Some of these modeling schemes are glmnetcr [6], ordinalgmifs [7], and penalized stereotype logit models [8]. Apart from the stereotype logit, these models are linear in that the objective cost functions can be represented as a linear combination of parameters estimates; these models also assume the cost function is convex. For the stereotype logit, which includes nonlinear combinations of parameter estimates, optimization algorithms that assumed linearity, and function convexity, were applied [7] [9]. A nonlinear and nonconvex approach to optimize the cost function of the penalized stereotype logit should be explored.

This study investigates the extension of a previously developed elastic net penalized stereotype logit [8]. We add an elastic net penalty [10] to the stereotype logit model [3]. To optimize the penalized function, we use the Adam optimizer [11] which is suited to nonlinear functions. This, in turn, allows us to evaluate the prediction accuracy of the model, which we were not able to do previously. The updated modeling procedure is presented with first order derivatives, optimization procedure, and a bootstrap resampling scheme to assess variable importance. Said modeling procedure is applied to simulated and real-world datasets with reported results. The proposed method is compared with the ordinalgmifs implemented L1 penalized stereotype logit [7], an existing method for analyzing high dimensional data with an ordinal outcome.

2. Method

For a given observation i (there are a total of n observations), denote the outcome vector $\boldsymbol{y}_i$ as $(y_{i1}, y_{i2}, \cdots, y_{iJ})$ where $y_{ij} = 1$ if for that observation, the outcome is in the j^{th} category, and all other entries are set to 0. There are J possible outcomes. Denote the vector $\boldsymbol{x}_i = (x_{i1} x_{i2} \cdots x_{ip})'$ as the covariate vector consisting of p values. The log of the information entropy, based on a multinomial distribution, is represented as

$$L(\boldsymbol{\theta} \mid \boldsymbol{y}, \boldsymbol{x}) = \frac{1}{n} \sum_{i=1}^{n} \left[\sum_{j=1}^{J-1} y_{ij} \theta_{ij} + \log \pi_J (\boldsymbol{x}_i) \right] \tag{1}$$

where

$$\theta_{ij} = \log \frac{\pi_j (\boldsymbol{x}_i)}{\pi_J (\boldsymbol{x}_i)}, \tag{2}$$

and

$$\pi_j (\boldsymbol{x}_i) = \frac{\mathrm{e}^{\theta_{ij}}}{\sum_{j=1}^{J} \mathrm{e}^{\theta_{ij}}}. \tag{3}$$

The log of the odds ratio, with level J being the reference level, θ_{ij}, is represented as $\alpha_j + \phi_j \{\boldsymbol{x}_i' \boldsymbol{\beta}\}$. Therefore, $\pi_j (\boldsymbol{x}_i)$ are now modeled as

$$\frac{\exp(\alpha_j + \phi_j \boldsymbol{x}_i' \boldsymbol{\beta})}{1 + \sum_{j'=1}^{J-1} \exp(\alpha_{j'} + \phi_{j'} \boldsymbol{x}_i' \boldsymbol{\beta})}. \tag{4}$$

This representation is known as the stereotype logit [3]. For each ordered level, the effect of the independent variables is equal to an overall effect, $\boldsymbol{x}_i' \boldsymbol{\beta}$, multiplied by a value ϕ_j, which is referred to as the intensity parameter. Primarily, as we are concerned with modeling the log of the odds ratios, θ_{ij}, the log of the information entropy can now be written as

$$L(\boldsymbol{\beta}, \boldsymbol{\alpha}, \boldsymbol{\phi} \mid \boldsymbol{y}, \boldsymbol{x}) = \frac{1}{n} \sum_{i=1}^{n} \sum_{i=1}^{J-1} y_{ij} \left(\alpha_j + \phi_j \{\boldsymbol{x}_i' \boldsymbol{\beta}\} \right) - \log 1 + \sum_{j=1}^{J-1} \mathrm{e}^{\alpha_j + \phi_j \{\boldsymbol{x}_i' \boldsymbol{\beta}\}}. \tag{5}$$

2.1. Elastic Net Penalized Stereotype Logit

We take the log of the information entropy, with a stereotype logit parameterization, and add an elastic net penalty [10]. A penalty on the sum of the squared and absolute values of the parameters is enforced [12] [13]. For a set of parameters, represented in a p length vector $\boldsymbol{\beta}$, the elastic net penalty is defined as

$$\iota = \frac{\lambda}{2n} \sum_{k=1}^{p} \left(\varsigma \beta_k^2 + (1 - \varsigma) |\beta_k| \right), \tag{6}$$

where $0 < \lambda < \infty$ can vary and n is the sample size of the dataset. For this study, ς was set to 0.5. The goal of the elastic net is to penalize large values of the parameter estimates, forcing their magnitude to decrease in proportion to their size. During optimization, the system is forced shrink the parameter estimates' size

when finding an optimal solution.

Based on the log of the information entropy, we are concerned with finding estimates for parameters, $\hat{\boldsymbol{\beta}}$, $\hat{\boldsymbol{\alpha}}$, and $\hat{\boldsymbol{\phi}}$ such that

$$\left(\hat{\boldsymbol{\beta}},\hat{\boldsymbol{\alpha}},\hat{\boldsymbol{\phi}}\right)=\underset{\boldsymbol{\beta},\boldsymbol{\alpha},\boldsymbol{\phi}}{\arg\max}\, L\left(\boldsymbol{\beta},\boldsymbol{\alpha},\boldsymbol{\phi}\mid \boldsymbol{y},\boldsymbol{x}\right) \tag{7}$$

where $\hat{\boldsymbol{\alpha}}$ denotes the vector of length $J-1$ containing the intercepts for the $J-1$ logits and $\hat{\boldsymbol{\phi}}$ denotes the vector on length $J-1$ containing the intensity parameters. In addition, minimizing the negative log entropy is equivalent to maximizing the log entropy, and we will work with the negative representation. Therefore, after imposing the elastic net penalty, we are concerned with finding parameter estimates such that:

$$\left(\hat{\boldsymbol{\beta}},\hat{\boldsymbol{\alpha}},\hat{\boldsymbol{\phi}}\right)=\underset{\boldsymbol{\beta},\boldsymbol{\alpha},\boldsymbol{\phi}}{\arg\min}\left\{-L\left(\boldsymbol{\beta},\boldsymbol{\alpha},\boldsymbol{\phi}\mid \boldsymbol{y},\boldsymbol{x}\right)+\frac{\lambda}{2n}\sum_{k=1}^{p}\left(\varsigma\beta_k^2+\left(1-\varsigma\right)\left|\beta_k\right|\right)\right\} \tag{8}$$

In machine learning, for any given model there is usually a hyperparameter set, a set of parameters that is not optimized over but whose choice of values affects the final solution. In machine learning, if there are multiple hyperparameters for any optimization procedure, there is still no established method to select these to optimize the function with respect to the parameters [14]. For the purposes of this manuscript, the hyperparameters were set to given values (a small range of values was considered with the optimal set being selected); λ was set to 0.001. The partial first derivatives with respect to α_j, β_k, and ϕ_j are presented.

$$\frac{\partial L}{\partial \alpha_j}=\frac{1}{n}\sum_{i=1}^{n}\left(y_{ij}-\pi_{ij}\right) \tag{9}$$

$$\frac{\partial L}{\partial \beta_k}=\frac{1}{n}\left(\sum_{i=1}^{n}x_{ik}\sum_{j=1}^{J-1}\phi_j\left(y_{ij}-\pi_{ij}\right)-\lambda\left(\varsigma\beta_k+\left(1-\varsigma\right)sign\left(\beta_k\right)/2\right)\right) \tag{10}$$

$$\frac{\partial L}{\partial \phi_j}=\frac{1}{n}\sum_{i=1}^{n}\left(y_{ij}-\pi_{ij}\right)\sum_{k=1}^{p}x_{ik}\beta_k \tag{11}$$

Denote the full parameter set $\boldsymbol{\beta}$, $\boldsymbol{\alpha}$, and $\boldsymbol{\phi}$ as $\boldsymbol{\psi}$. The partial derivatives are vectorized (placed into one vector) and are represented by a derivative vector, denoted $\nabla L\left(\boldsymbol{\psi}\right)$.

2.2. Adam Optimization

The implemented Adam algorithm [11] attempts to find a parameter set that will minimize model Equation (8). The approach uses non-linear programs to find optimal solutions given a hyperparameter set. The Adam optimizer combines the idea of momentum optimization [15] and RMSProp [16]. Adam keeps track of an exponentially decaying average of past gradients from previous iterations (momentum optimization). Adam also tracks the exponentially decaying average of past squared gradients from previous iterations (RMSProp). The applied algorithm is listed below.

1) Initialize $\boldsymbol{m}$ and $\boldsymbol{s}$ to have all zero entries; these vectors are of length

$p+2(J-1)$.

2) Initialize $\boldsymbol{\psi}$ using He Initialization [17].
3) Compute $\nabla L(\boldsymbol{\psi})$.
4) $\boldsymbol{m} \leftarrow \nu_1 \boldsymbol{m} + (1-\nu_1)\nabla L(\boldsymbol{\psi})$.
5) $\boldsymbol{s} \leftarrow \tau_2 \boldsymbol{s} + (1-\tau_2)\nabla L(\boldsymbol{\psi})^2$.
6) $\boldsymbol{\psi} \leftarrow \boldsymbol{\psi} - \eta \boldsymbol{m} \div \sqrt{\boldsymbol{s}+\varepsilon}$.
7) Repeat steps 3 through 6 until $L(\boldsymbol{\psi} \mid \boldsymbol{y}, \boldsymbol{x})^{i+i} - L(\boldsymbol{\psi} \mid \boldsymbol{y}, \boldsymbol{x})^{i} < \zeta$, where i references the iteration number, or until a prespecified number of iterations are reached.

The vectors $\boldsymbol{m}$ and $\boldsymbol{s}$ contain the exponentially decaying averages of $\nabla L(\boldsymbol{\psi})$ and $\nabla L(\boldsymbol{\psi})^2$. For the He initialization [17], the parameter estimate vector $\boldsymbol{\psi}$ is initialized using the random normal function of the form

$$Norm(0,1) * \sqrt{2/p}, \tag{12}$$

where $Norm(0,1)$ are randomly generated values from a normal distribution with a mean 0 and standard deviation 1 and p is the number of covariates in the dataset. The hyperparameter set consists of ν_1, τ_2, η, ε, ζ, and ς. For this study, after considering a small range of candidate values, ν_1 was set to 0.5, τ_2 was set to 0.8, η was set to 0.008, ε was set to 1E-7, ζ was set to 1E-5, λ was set to 0.001 and, ς was set to 0.5. Steps three through six are repeated until a specified number of iterations is reached (800 for this study) or until $L(\boldsymbol{\psi} \mid \boldsymbol{y}, \boldsymbol{x})^{i+i} - L(\boldsymbol{\psi} \mid \boldsymbol{y}, \boldsymbol{x})^{i} < \zeta$. In applying this algorithm, we need to include an adjustment for the elastic net penalty. When taking derivatives with respect to $\boldsymbol{\beta}$, we adjust these functions by subtracting the derivatives of the elastic net penalty. This is not done for $\boldsymbol{\alpha}$ or $\boldsymbol{\phi}$. As a result, when computing the derivatives for the $\boldsymbol{\beta}_k$, where k references the iteration, we subtract from that derivative term $(\lambda/n)(\varsigma\beta_k + (1-\varsigma)sign(\beta_k)/2)$ which leads to the derivatives for each $\boldsymbol{\beta}$ subject to the elastic net penalty. For each iteration, in addition to modifying $\boldsymbol{\beta}$ by subtracting a function of its derivative we also shrink the parameters by a factor of $\lambda(\varsigma\beta_k + (1-\varsigma)sign(\beta_k)/2)$. This method was implemented in the R programming environment [18]. Functions from the MASS [19] and matrixcalc [20] R packages were used to implement the proposed model.

2.3. Applied Bootstrap Resampling Procedure

For the proposed model, the standard errors of our parameter estimates are computed using a bootstrapping pairs design [21]. Denote B as the number of resamples without replacement. For this study, B is set to 200 [21]. For each bootstrap resample, we resample n tuples with replacement, which gives us the dataset $\boldsymbol{X}_b$ and $\boldsymbol{y}_b$, $b = 1, 2, \cdots, B$. The proposed model is then fit to each resampled data set. Once the B models are fit, the corresponding parameter estimates are obtained. Denote the bth bootstrap parameter estimates as $(\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\phi}})_b$. Having these B parameter estimates allows us to estimate their standard errors and construct confidence intervals.

The bootstrap-t confidence interval method is used to construct confidence

intervals. The bootstrap-t confidence intervals are of the form

$$\left[\hat{\beta}_k - \hat{t}^{(1-\alpha)} \times se\left(\hat{\beta}_k\right), \hat{\beta}_k + \hat{t}^{(1-\alpha)} \times se\left(\hat{\beta}_k\right)\right] \tag{13}$$

where

$$se\left(\hat{\beta}_k\right) = \sqrt{V\left(\hat{\beta}_k\right)/B} \tag{14}$$

with $V\left(\hat{\beta}_k\right)$ being defined as

$$V\left(\hat{\beta}_k\right) = \frac{1}{B-1}\sum_{b=1}^{B}\left(\hat{\beta}(.)_k - \hat{\beta}^*_{kb}\right)^2 \tag{15}$$

where $k = 1, 2, \cdots, p$ and

$$\hat{\beta}(.)_k = \frac{1}{B}\sum_{b=1}^{B}\hat{\beta}^*_{k,b}, \tag{16}$$

where $\hat{\beta}^*_{k,b}$ denotes the estimate of $\hat{\beta}_k$ from the b^{th} bootstrapped resamples dataset. In addition, $\hat{t}^{(\alpha)}$ is chosen from the standard normal distribution such that

$$\sum_{b=1}^{B}\left\{Z^*(b) \le \hat{t}^{(\alpha)}\right\}\Big/B = \alpha, \tag{17}$$

where $Z^*(b)$ is defined as

$$Z^*(b) = \frac{\hat{\beta}^*_{k,p} - \hat{\beta}(.)_k}{se\left(\hat{\beta}(.)_k\right)} \tag{18}$$

The R programming environment [18] was used to implement this procedure.

3. Application to Simulated Data

The simulation procedure used is the same as previously presented [8] with a few noted changes. In that study, one dataset was simulated using a compound symmetric correlation structure for the covariates with ρ = 0.01. In addition, we simulated three additional dataset types. The second dataset type was simulated using a first order autoregressive, AR(1), correlation structure with ρ set to 0.1. The third dataset type has a Toeplitz correlation structure with each ρ generated randomly using a uniform distribution U (0, 0.4). The fourth dataset type has an unstructured correlation structure with each ~U (0, 0.4). For all simulated datatypes, 20 covariates (10 are significant) were generated for 1000 observations. Among the 10 significant parameter estimates, 5 were randomly set at 0.5 and 5 at −0.5 for all datasets. Each dataset was centered and scaled before the proposed model was fit. For each datatype, 100 datasets were simulated. The described bootstrap resampling technique was used to provide 95% confidence intervals; B = 200 resamples were used. In the model fitting process, the data were split into training: test data in the ratio 8:2. The model was developed using the training data. The final model was applied to the test dataset (independent data not used to build the model). The main criteria examined were the number of significant covariates that have non-zero parameter estimates, the number of non-significant

coefficients that have estimates close to zero within a threshold, the accuracy of predictions when the model is applied to the test data set, and execution times. Functions from the R packages MASS [19], mvtnorm [22] [23], futility [24], MBESS [25], Matrix [26], and corpcor [27] were employed to implement this procedure.

Results

The proposed methodology, and ordinalgmifs method with the option probability, model = "Stereotype", was applied to the simulated data. The goal was to compare two implementations of penalized stereotype logit models. **Table 1** presents the mean, and standard deviations, for prediction accuracy (determined by the test data) and execution times for both methods. Two-sided, two sample Welch's t tests, with significance level of 0.05, were used to compare mean accuracy and execution times for both methods. For the proposed method, the average prediction accuracy for the compound symmetric simulated data is 96.1%; for AR(1) correlated data, 96.52% of the observations were correctly classified. This rate was 96.24% for the Toeplitz correlated data and 96.49% for the unstructured correlated data. Regarding classification, the proposed method outperformed the ordinalgmifs method on all datasets as determined by the t tests (all p values < 0.001). Regarding execution times, the proposed method executed faster for all datasets considered (all p values < 0.001). The average execution times for the proposed method range from 17.21 to 23.63 seconds, for the ordinalgmifs method the range is 166.98 to 200.06 seconds. The proposed method executed ~10-fold faster on average.

Tables 2-5 present the parameter estimates for the 10 significant parameters based on the proposed method. For all simulated datasets, the 10 non-significant parameters (not shown in the tables) had a maximum absolute value of 0.04; these values were close to 0. For the ordinalgmifs method, all non-significant parameters had estimates of 0. The proposed method selects the significant parameters that are truly related to the outcome while setting estimates of the non-significant parameters close to 0. In comparison, the ordinalgmifs methods set these values to 0. The confidence intervals are somewhat narrow (~0.5) for parameter estimates of significant covariates.

Figure 1 presents the percent of observations correctly classified per iteration for the training data of the simulated datasets. The method performance never decreases for all simulated datasets. Once the method maximizes the proportion that it can correctly estimate, it oscillates around that value. This could be due in part to the use of the Adam optimization algorithm [11]. The results indicate that the proposed model framework is adept at variable selection and classification capabilities when applied to independent datasets. The proposed model outperforms the ordinalgmifs implementation with regards to prediction accuracy and execution times. Computationally, the proposed model executes faster than the ordinalgmifs implementation by ~10-fold. The analysis was performed in the R programming environment [18].

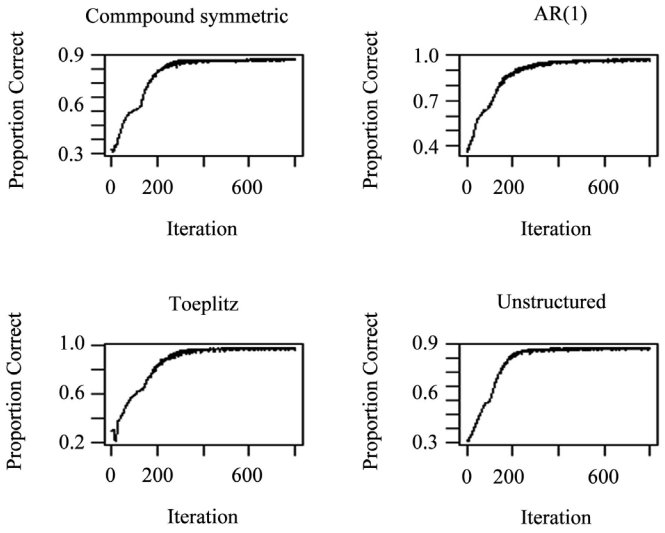

Figure 1. Plots presenting the percent correctly classified per iteration for the four simulated datasets. "The black line demonstrates the progress (% correctly classified) for the training data".

Table 1. Average accuracy (% correctly classified) and execution times for proposed and ordinalgmifs methods, along with standard deviations.

	Proposed Method		Ordinalgmifs	
Dataset (Correlation Type)	Accuracy (%)	Execution Time (Seconds)	Accuracy (%)	Execution Time (Seconds)
Compound Symmetric	96.1 (1.37)*	23.63 (17.50)*	93.51 (2.1)*	187.38 (50.28)*
First Order Autoregressive	96.52 (1.46)*	17.90 (3.23)*	94.13 (2.19)*	183.59 (39.30)*
Toeplitz	96.24 (1.33)*	18.21 (3.89)*	93.01 (2.46)*	200.06 (55.93)*
Unstructured	96.49 (1.32)*	17.21 (0.13)*	93.95 (2.1)*	166.98 (41.73)*

For the two methods, accuracy and executions times were compared using a two-sided Welch's two sample t test, with significance level of 0.05. "*" indicates a statistically significant difference.

Table 2. Parameter estimates and 95% confidence intervals for truly important variables included in the final model of the compound symmetric correlated data.

Truly Important Variable	Parameter Estimate	95% Confidence Interval
V1	−1.548	(−1.725, −1.372)
V2	−1.574	(−1.754, −1.394)
V3	−1.62	(−1.805, −1.434)
V4	−1.629	(−1.816, −1443)
V5	−1.55	(−1.727, −1.373)
V6	1.625	(1.44, 1.81)
V7	1.618	(1.433, 1.803)
V8	1.639	(1.452, 1.826)
V9	1.701	(1.506, 1.896)
V10	1.691	(1.497, 1.884)

Table 3. Parameter estimates and 95% confidence intervals for truly important variables included in the final model of the AR(1) correlated data.

Truly Important Variable	Parameter Estimate	95% Confidence Interval
V1	1.497	(1.317, 1.676)
V2	1.466	(1.29, 1.642)
V3	1.416	(1.246, 1.585)
V4	1.531	(1.347, 1.715)
V5	1.446	(1.273, 1.62)
V6	−1.525	(−1.709, −1.341)
V7	−1.533	(−1.716, −1.349)
V8	−1.537	(−1.723, −1.352)
V9	−1.587	(−1.778, −1.397)
V10	−1.533	(−1.717, −1.35)

Table 4. Parameter estimates and 95% confidence intervals for truly important variables included in the final model of the Toeplitz correlated data.

Truly Important Variable	Parameter Estimate	95% Confidence Interval
V1	1.466	(1.296, 1.636)
V2	1.522	(1.347, 1.698)
V3	1.485	(1.313, 1.657)
V4	1.521	(1.346, 1.697)
V5	1.507	(1.334, 1.681)
V6	−1.579	(−1.761, −1.396)
V7	−1.547	(−1.726, −1.368)
V8	−1.569	(−1.75, −1.388)
V9	−1.582	(−1.765, −1.399)
V10	−1.588	(−1.771, −1.405)

Table 5. Parameter estimates and 95% confidence intervals for truly important variables included in the final model of the unstructured correlated data.

Truly Important Variable	Parameter Estimate	95% Confidence Interval
V1	−1.479	(−1.653, −1.305)
V2	−1.602	(−1.774 −1.43)
V3	−1.63	(−1.808, −1.453)
V4	−1.708	(−1.895, −1521)
V5	−1.7	(−1.887, −1.514)
V6	1.729	(1.539, 1.919)
V7	1.703	(1.516, 1.889)
V8	1.654	(1.468, 1.84)
V9	1.806	(1.607, 2.005)
V10	1.623	(1.442, 1.804)

4. Application to NHL Data

The data came from a study titled "Subclass Mapping: Identifying Common Subtypes in Independent Disease Data Sets" [28]. The primary aim of this manuscript was to find gene expression profiles that could predict, with some degree of error, molecular subtypes of diseases. The cancer types evaluated by this manuscript were lymphoma and breast cancer. Lymphoma is defined as a cancer of the lymphatic system. The following review is taken from Cancer Stat Facts [29]. NHL make up approximately 90% of all malignant lymphomas, with the Hodgkin lymphomas accounting for the remaining 10%. NHL "is a heterogeneous disease resulting from the malignant transformation of lymphocytes and includes multiple subtypes each with specific molecular and clinical characteristics" [29]. NHL can either start in the B-lymphocytes or the T-lymphocytes. Among B-cell lymphomas, diffuse large B-cell lymphomas are the most common. T-cell lymphomas account for 15% of NHL in the United States. NHL account for 4.3% of all new cancer cases. There were 72,240 estimated cases for 2017 and 20,140 estimated deaths. The median age of diagnosis was 67, with the highest proportion of new cases occurring in the 65 - 74 age group. The estimated 5-year survival rate was 71.0%. The issue of stage prediction with NHL, using a set of covariates, provides an opportunity to evaluate the performance of the proposed model framework.

The raw data, DLBCL-A: data set and DLBCL-A: class labels, were downloaded from http://portals.broadinstitute.org/cgi-bin/cancer/datasets.cgi [28]. The data were generated using one-channel oligonucleotide microarrays. The data have three subtypes, designated as oxidative phosphorylation (OxPhos), B-cell response (BCR), and host response (HR) [28]. The independent variables are gene expression values. The R package CePa [30] was used to read in the datasets. All variables were input into the model. The gene expression values were standardized (centered and scaled) prior to model fitting. There were 661 genes in the dataset. Among the 141 samples, 49 were OxPhos, 50 were BCR, and 42 were HR. The proposed model, along with the ordinalgmifs implementation of the stereotype logit, was applied to the gene expression dataset, with associated outcome vectors, to select genes associated with NHL subtypes. The described bootstrap resampling procedure was applied, yielding estimates of standard errors that were used to compute 95% confidence intervals. B = 200 resamples were used. Due to the small sample size, leave-one-out cross validation was used to estimate the predictive capabilities of the model. The analysis was performed in the R programming environment [18].

Results

Table 6 shows selected genes along with parameter estimates and confidence intervals. The displayed genes are those with the largest absolute value of the coefficient of variation, with standard deviations provided by the bootstrap resampling scheme. Only the top 20 were displayed. The corresponding gene

names are also presented. The names of the genes are provided by the HUGO Gene Nomenclature Committee (https://www.genenames.org/) [31]. As with the results in the simulation section, the 95% confidence intervals are narrow, and prediction accuracy was 73%. When applying the stereotype logit-based ordinalgmifs function to the NHL data, parameter estimates could not be obtained due to optimization issues with that implementation. The following error was reported by the R programming environment "Error in optim(c(alpha, phi), fn.stereo, w = w, x = x, beta = beta, y = y: L-BFGS-B needs finite values of 'fn'". The above error relates to the optimization function being passed infinite values during the optimization process. All covariate values were centered and scaled prior to analysis. To correct the error, multiple values of the hyperparameters were passed to the ordinalgmifs function in R with no success. The data was checked for missing values; there were none. In addition, all the data were numeric. As a result, no comparison could be made with the ordinalgmifs implementation of the L1 penalized stereotype logit.

Table 6. Variable importance based on the application of the proposed model to the NHL dataset. The topmost 20 genes, in terms of variable importance, are presented. The model achieved a prediction accuracy of 73%.

Gene Name	Definition	Parameter Estimate	95% Confidence Interval
TCF7	transcription factor 7	−2.666	(−2.845, −2.486)
GSTM2	glutathione S-transferase mu 2	−2.629	(−2.813, −2.444)
ITGB7	integrin subunit beta 7	−2.598	(−2.756, −2.439)
EEF1A1	eukaryotic translation elongation factor 1 alpha 1	−2.59	(−2.839, −2.342)
LOC220594	NA	−2.572	(−2.736, −2.408)
DEK	DEK proto-oncogene	2.57	(2.446, 2.694)
ITGAL	integrin subunit alpha L	−2.561	(−2.717, −2.405)
BIN1	bridging integrator 1	−2.556	(−2.704, −2.409)
RPL21	ribosomal protein L21	−2.503	(−2.671, −2.335)
RBPSUH	recombination signal binding protein for immunoglobulin kappa J region	−2.503	(−2.65, −2.356)
NCOA1	nuclear receptor coactivator 1	−2.494	(−2.648, −2.34)
MYCBP2	MYC binding protein 2, E3 ubiquitin protein ligase	−2.47	(−2.614, −2.326)
A2M	alpha-2-macroglobulin	−2.416	(−2.595, −2.238)
IL10RA	interleukin 10 receptor subunit alpha	−2.413	(−2.564, −2.262)
SLC25A5	solute carrier family 25 member 5	−2.383	(−2.539, −2.227)
CCL21	C–C motif chemokine ligand 21	−2.378	(−2.53, −2.227)
KPNB1	karyopherin subunit beta 1	−2.377	(−2.536, −2.217)
COL9A2	collagen type IX alpha 2 chain	−2.374	(−2.552, −2.196)
RPS21	ribosomal protein S21	−2.363	(−2.524, −2.202)
ACP1	acid phosphatase 1	−2.361	(−2.501, −2.221)

5. Discussion

There are multiple hyperparameters in the elastic net constrained stereotype logit, optimal values for these must be explored as this has the potential to increase variable selection and classification capabilities. This is usually accomplished with a grid search and is an open problem in machine learning for multiple hyperparameters [14]. For this study, a small selection of values was considered for each hyperparameter. The hyperparameter of interest is λ but the choice for the remaining hyperparameters is also very important in determining the optimal solution.

A bootstrap resampling procedure was used to estimate the 95% confidence intervals. The main drawback is the computational time required to produce the confidence intervals with 200 additional models being fit. It may be advisable to perform a closed form estimate of the parameter variance matrix [2].

Although the stereotype logit is considered by many a generalized linear model, it is not. As such, an optimal solution may not exist, or there may be inflexion points. As a result, different starting values may yield different solutions. In this study, applying the method to a given dataset does not exhibit a great deal of variation in results, and the results of the applied bootstrap procedure confirm this. To address this, we applied a variable initialization scheme proposed by He [17]. In addition, the Adam optimization function is well suited to dealing with non-convex functions [11]. The combination of these two factors addresses this issue.

6. Conclusion

A proposed model for the elastic net penalized stereotype logit model, with optimization provided by the Adam optimizer, to analyze ordinal outcome data was presented. The proposed method was applied to simulated and NHL data with reported results. For the simulated data, variable selection was perfect, and only significant variables had parameter estimates not close to 0. The classifications ranged from 96.1% to 96.52% on the test datasets. For the NHL data, 73% were correctly classified. The 20 topmost genes in terms of absolute value of the coefficient of variation were presented. Our evaluation study shows that the proposed method outperforms the ordinalgmifs penalized stereotype logit model; no comparison could be made with the NHL data analysis as the ordinalgmifs implemented stereotype logit was not able to produce parameter estimates. This manuscript is an extension of previous work [8]. In the previous study, variable selection was adequate, but the classification capabilities were lacking. This work improves the prediction accuracy when applied to simulated and NHL data (ranging from 73% to 96.52%). In addition, the variable importance also improved with only the significant parameters having non-zero estimates. This study demonstrates, with success, the application of the Adam optimizer to the elastic net penalized stereotype logit model to analyze ordinal outcome data with promising results, as demonstrated on the simulated and NHL datasets.

Acknowledgements

First and foremost, I would like to thank God from whom all blessings flow. I would also like to express special thanks to Timothy Wysocki, the Co-director of the Center for Healthcare Delivery Science, at Nemours Children's Specialty Care for allowing me to work on this project.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

[1] Williams, M. and Schnellhammer, P. (2016) Testicular Seminoma. http://emedicine.medscape.com/article/437966-overview

[2] Agresti, A. (2014) Categorical Data Analysis. John Wiley & Sons, New York.

[3] Anderson, J.A. (1984) Regression and Ordered Categorical Variables. *Journal of the Royal Statistical Society*: *Series B* (*Methodological*), **46**, 1-30. http://www.jstor.org/stable/2345457 https://doi.org/10.1111/j.2517-6161.1984.tb01270.x

[4] Bellazzi, R. (2014) Big Data and Biomedical Informatics: A Challenging Opportunity. *Yearbook of Medical Informatics*, **9**, 8-13. https://doi.org/10.15265/IY-2014-0024

[5] Murdoch, T.B. and Detsky, A.S. (2013) The Inevitable Application of Big Data to Health Care. *JAMA*, **309**, 1351-1352. https://doi.org/10.1001/jama.2013.393

[6] Archer, K.J. and Williams, A.A.A. (2012) L1 Penalized Continuation Ratio Models for Ordinal Response Prediction Using High-Dimensional Datasets. *Statistics in Medicine*, **31**, 1464-1474. https://doi.org/10.1002/sim.4484

[7] Archer, K.J., Hou, J., Zhou, Q., Ferber, K., Layne, J.G. and Gentry, A.E. (2014) Ordinalgmifs: An R Package for Ordinal Regression in High-Dimensional Data Settings. *Cancer Informatics*, **13**, CIN.S20806. https://doi.org/10.4137/CIN.S20806

[8] Williams, A.A. and Archer, K.J. (2015) Elastic Net Constrained Stereotype Logit Model for Ordered Categorical Data. *Biometrics & Biostatistics International Journal*, **2**, Article ID: 00049. https://doi.org/10.15406/bbij.2015.02.00049

[9] Hastie, T., Taylor, J., Tibshirani, R., Walther, G., Boyd, S., Friedman, J., *et al.* (2007) Forward Stagewise Regression and the Monotone Lasso. *Electronic Journal of Statistics*, **1**, 1-29. https://doi.org/10.1214/07-EJS004

[10] Zou, H. and Hastie, T. (2005) Regularization and Variable Selection via the Elastic Net. *Journal of the Royal Statistical Society*: *Series B* (*Methodological*), **67**, 301-320. https://web.stanford.edu/~hastie/Papers/B67.2%20%282005%29%20301-320%20Zou%20&%20Hastie.pdf https://doi.org/10.1111/j.1467-9868.2005.00503.x

[11] Kingma, D.P. and Ba, J. (2014) Adam: A Method for Stochastic Optimization. 1-15.

[12] Hoerl, A.E. and Kennard, R.W. (1970) Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, **12**, 55-67. https://doi.org/10.1080/00401706.1970.10488634

[13] Tibshirani, R. (1996) Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society*: *Series B* (*Methodological*), **58**, 267-288. http://www.jstor.org/stable/2346178

https://doi.org/10.1111/j.2517-6161.1996.tb02080.x

[14] Kaiser, L., Gomez, A.N., Shazeer, N., Vaswani, A., Parmar, N., Jones, L. and Uszkoreit, J. (2017) One Model To Learn Them All. https://arxiv.org/pdf/1706.05137.pdf

[15] Polyak, B.T. (1964). Some Methods of Speeding up the Convergence of Iteration Methods. *USSR Computational Mathematics and Mathematical Physics*, **4**, 1-17. http://www.sciencedirect.com/science/article/pii/0041555364901375 https://doi.org/10.1016/0041-5553(64)90137-5

[16] Hinton, G., Srivastava, N. and Swersky, K. (n.d.) Neural Networks for Machine Learning. http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf

[17] He, K., Zhang, X., Ren, S. and Sun, J. (2014) Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. 1026-1034.

[18] R Core Team (2017) R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna. https://www.r-project.org/

[19] Venables, W.N. and Ripley, B.D. (2002) Modern Applied Statistics with S. Fourth Edition, Springer, New York. https://doi.org/10.1007/978-0-387-21706-2

[20] Novomestky, F. (2012) Matrixcalc: Collection of Functions for Matrix Calculations. https://cran.r-project.org/package=matrixcalc

[21] Efron, B. and Tibshirani, R. (1986) Bootstrap Methods for Standard Errors, Confidence Intervals, and Other Methods of Statistical Accuracy. *Statistical Science*, **1**, 54-75. https://doi.org/10.1214/ss/1177013815

[22] Genz, A., Bretz, F., Miwa, T., Mi, X., Leisch, F., Scheipl, F. and Hothorn, T. (2017) Mvtnorm: Multivariate Normal and t Distributions. http://cran.r-project.org/package=mvtnorm

[23] Genz, A. and Bretz, F. (2009) Computation of Multivariate Normal and t Probabilities. Lecture Notes in Statistics Vol. 195, Springer-Verlag, Heidelberg. https://doi.org/10.1007/978-3-642-01689-9

[24] Zhuang, Y., Juraska, M., Grove, D., Gilbert, P. and Luedtke, A. (2017) Futility: Interim Analysis of Operational Futility in Randomized Trials with Time-to-Event Endpoints and Fixed Follow-Up. https://cran.r-project.org/package=futility

[25] Kelley, K. (2017) MBESS: The MBESS R Package. https://cran.r-project.org/package=MBESS

[26] Bates, D. and Maechler, M. (2017) Matrix: Sparse and Dense Matrix Classes and Methods. https://cran.r-project.org/package=Matrix

[27] Schafer, J., Opgen-Rhein, R., Zuber, V., Ahdesmaki, M., Silva, A.P.D. and Strimmer, K. (2017) Corpcor: Efficient Estimation of Covariance and (Partial) Correlation. https://cran.r-project.org/package=corpcor

[28] Hoshida, Y., Brunet, J.P., Tamayo, P., Golub, T.R. and Mesirov, J.P. (2007) Subclass Mapping: Identifying Common Subtypes in Independent Disease Data Sets. *PLoS ONE*, **2**, e1195. https://doi.org/10.1371/journal.pone.0001195

[29] National Cancer Institute (2017) Cancer Stat Fact: Non-Hodgkin Lymphoma. https://seer.cancer.gov/statfacts/html/nhl.html

[30] Gu, Z. (2012) CePa: Centrality-Based Pathway Enrichment. https://cran.r-project.org/package=CePa

[31] HUGO Gene Nomenclature Committee (n.d.) HGNC Database of Human Genes. https://www.genenames.org/