glmnetcr: An R Package for Ordinal Response Prediction in High-Dimensional Data Settings

Kellie J. Archer
The Ohio State University

Abstract

This paper describes an R package, **glmnetcr**, that provides a function for fitting a penalized continuation ratio model when interest lies in predicting an ordinal response. The function, `glmnetcr` uses the coordinate descent fitting algorithm as implemented in the **glmnet** package and described by (Friedman, Hastie, and Tibshirani 2010). Methods for extracting all estimated coefficients, extracting non-zero coefficient estimates, obtaining the predicted class, and obtaining the class-specific fitted probabilities have been implemented. Additionally, generic methods from **glmnet** including `print` and `plot` can be applied to a `glmnetcr` object.

Keywords: ordinal response, penalized models, LASSO, L_1 constraint, R.

1. Introduction

High-throughput genomic experiments are frequently conducted for the purpose of examining whether genes are predictive of or significantly associated with phenotype. In many biomedical settings where histopathological or health status data are collected, phenotypic variables are recorded on an ordinal scale. Nevertheless, most often investigators neglect the ordinality of the phenotypic data and rather dichotomize the ordinal class than apply statistical methods suitable for two-class comparisons and predictions. This tendency to analyze ordinal data using dichotomous class methodologies may be due to the lack of available statistical methods and software for modeling an ordinal response in the presence of a high-dimensional covariate space. The approach of collapsing ordinal categories may neglect important information in the study (Armstrong and Sloan 1989).

A variety of statistical modeling procedures, namely, proportional odds, adjacent category, stereotype logit, and continuation ratio models can be used to predict an ordinal response. In this paper, we focus attention to the continuation ratio model because its likelihood can be easily re-expressed such that existing software can be readily adapted and used for model fitting. Suppose for each observation, $i = 1, \ldots, n$, the response Y_i belongs to one ordinal class $k = 1, \ldots, K$ and $\mathbf{x}_i$ represents a p-length vector of covariates. The backward formulation of the continuation ratio models the logit as

$$\texttt{logit}\left(P(Y = k | Y \leq k, \mathbf{X} = \mathbf{x})\right) = \alpha_k + \boldsymbol{\beta}_k^\top \mathbf{x} \tag{1}$$

whereas the forward formulation models the logit as

$$\texttt{logit}\left(P(Y = k | Y \geq k, \mathbf{X} = \mathbf{x})\right) = \alpha_k + \boldsymbol{\beta}_k^\top \mathbf{x}. \tag{2}$$

Rather than describe both formulations in detail, here we present the backward formulation, which is commonly used when progression through disease states from none, mild, moderate, severe is represented by increasing integer values, and interest lies in estimating the odds of more severe disease compared to less severe disease (Bender and Benner 2000). Therefore for $i = 1, \ldots, n$ we can construct a vector $\mathbf{y}_i$ from Y_i to represent ordinal class membership, such that $\mathbf{y}_i = (y_{i1}, y_{i2}, \ldots, y_{iK})^\top$, where $y_{ik} = 1$ if the response is in category k and 0 otherwise, so that $n_i = \sum_{k=1}^{K} y_{ik} = 1$. Using the logit link, Equation 3 represents the conditional probability for class k

$$\delta_k(\mathbf{x}) = P(Y = k | Y \leq k, \mathbf{X} = \mathbf{x}) = \frac{\exp(\alpha_k + \boldsymbol{\beta}_k^\top \mathbf{x})}{1 + \exp(\alpha_k + \boldsymbol{\beta}_k^\top \mathbf{x})}. \tag{3}$$

The likelihood for the continuation ratio model is then the product of conditionally independent binomial terms (Cox 1975), which is given by

$$L(\boldsymbol{\beta} | \mathbf{y}, \mathbf{x}) = \prod_{i=1}^{n} \delta_2^{y_{i2}} (1 - \delta_2)^{1 - \sum_{k=2}^{K} y_{ik}} \times \cdots \times \delta_K^{y_{iK}} (1 - \delta_K)^{1 - y_{iK}} \tag{4}$$

where here we have simplified our notation by not explicitly including the dependence of the conditional probability δ_k on $\mathbf{x}$. Further, simplifying our notation to let $\boldsymbol{\beta}$ represent the vector containing both the thresholds $(\alpha_2, \ldots, \alpha_K)$ and the log odds $(\beta_1, \ldots, \beta_p)$ for all $K - 1$ logits, the full parameter vector is

$$\boldsymbol{\beta} = (\alpha_2, \beta_{21}, \beta_{22}, \ldots, \beta_{2p}, \ldots, \alpha_K, \beta_{K,1}, \beta_{K,2}, \ldots, \beta_{K,p})^\top \tag{5}$$

which is of length $(K-1)(p+1)$. As can be seen from Equation 4, the likelihood can be factored into $K - 1$ independent likelihoods, so that maximization of the independent likelihoods will lead to an overall maximum likelihood estimate for all terms in the model (Bender and Benner 2000). A model consisting of $K - 1$ different $\boldsymbol{\beta}$ vectors may be overparameterized so to simplify, one commonly fits a constrained continuation model, which includes the $K - 1$ thresholds $(\alpha_2, \ldots, \alpha_K)$ and one common set of p slope parameters, $(\beta_1, \ldots, \beta_p)$. To fit a constrained continuation ratio model, the original dataset can be restructured by forming $K - 1$ subsets, where for classes $k = 2, \ldots, K$, the subset contains those observations in the original dataset up to class k. Additionally, for the kth subset, the outcome is dichotomized as $y = 1$ if the ordinal class is k and $y = 0$ otherwise. Furthermore, an indicator is constructed for each subset representing subset membership. Thereafter the $K - 1$ subsets are appended to form the restructured dataset, which represents the $K - 1$ conditionally independent datasets in Equation 4. Applying a logistic regression model to this restructured dataset yields an L_1 constrained continuation ratio model.

2. Penalized models

For datasets where the number of covariates p exceeds the sample size n, the backwards stepwise procedure cannot be undertaken. Furthermore, for any problem using a forward selection procedure the discrete variable inclusion process can exhibit high variance. Moreover, for high-dimensional covariate spaces, the best subset procedure is computationally prohibitive. Two penalized methods, ridge and L_1 penalization, places a penalty on a function of the coefficient

estimates, thereby permitting a model fit even for high-dimensional data Tibshirani (1996, 1997). A generalization of these penalized models can be expressed as,

$$\tilde{\beta} = \arg\min_{\beta}(\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}x_{ij}\beta_j)^2 + \lambda\sum_{j=1}^{p}|\beta_j|^q) \tag{6}$$

for $q \geq 0$. When $q = 1$ we have the an L_1 penalized model, when $q = 2$ we have ridge regression. Values of $q \in (1, 2)$ provide a compromise between the L_1 and ridge penalized models. Because when $q > 1$ coefficients are no longer set exactly equal to 0, the elastic net penalty was introduced

$$\lambda\sum_{j=1}^{p}(\alpha\beta_j^2 + (1 - \alpha)|\beta_j|). \tag{7}$$

3. Implementation

The **glmnetcr** package was written in the R programming environment (R Development Core Team 2009) and depends on the **glmnet** package (Park and Hastie 2007). Similar to the **Design** package which includes a function `cr.setup` for restructuring a dataset for fitting a forward continuation ratio model, in this package the model is fit by restructuring the dataset then passing the restructured dataset to a penalized logistic regression fitting function. However, unlike `cr.setup` which produces an object of class `list` from which the response and restructured independent variables are extracted and passed to a model fitting algorithm, in the **glmnetcr** package the restructuring functions are transparent to the user. Specifically, the **glmnetcr** package fits either a forward or backward (default) penalized constrained continuation ratio model by specification of `method="forward"` in the `glmnetcr` call. The `glmnetcr` function restructures the dataset to represent the $K - 1$ conditionally independent likelihoods and then fits the penalized continuation ratio model using the **glmnet** framework. This allows fitting a penalized model for situations where the number of covariates p exceed the sample size n. In addition, methods for extracting the best fitting model from the path using AIC and BIC criteria, obtaining predicted class and fitted class probabilities, and returning coefficient estimates were written in addition to adapting the `print` and `plot` methods from **glmnet** for a `glmnetcr` object.

4. Example

The **glmnetcr** package includes a filtered microarray dataset `diabetes` in which asymptomatic males not previously diagnosed with Type II diabetes were enrolled and subsequently were cross-classified as either normal controls ($N = 8$), having impaired fasting glucose ($N = 7$), or as Type II diabetics ($N = 9$) based on a fasting glucose intolerance test. From the code below we can see that the classification variable is stored as `y` in the first column of the `diabetes data.frame`; all subsequent columns are the 11,066 Illumina probes having no negative expression values. In fitting the model we can extract the covariates into an object `x` and the ordinal outcome into the object `y`. The code for fitting a backward (default) continuation ratio model is given by

```
R> library("glmnetcr")
R> data("diabetes")
R> dim(diabetes)
```

```
[1]    24 11067
```

```
R> names(diabetes)[1:10]
```

```
 [1] "y"            "ILMN_1343291" "ILMN_1651228" "ILMN_1651229"
 [5] "ILMN_1651236" "ILMN_1651254" "ILMN_1651262" "ILMN_1651268"
 [9] "ILMN_1651278" "ILMN_1651286"
```

```
R> summary(diabetes$y)
```

```
          control impaired fasting glucose
                8                        7
       type 2 diabetes
                9
```

```
R> x <- diabetes[, 2:dim(diabetes)[2]]
R> y <- diabetes$y
R> fit <- glmnetcr(x, y)
```

Notice that because the three classes are completely separated after including a few covariates, a warning was issued because larger models could not be fit. Nevertheless, all models before the convergence issue arose can be extracted from the resulting object. As with `glmnet` model objects, methods such as `print` and `plot` can be applied to `glmnetcr` model objects, which are helpful for selecting the step at which to select the final model from the solution path. For example, Figure 1 can be used to identify a more parsimonious model having a BIC close to the minimum BIC. Figure 2 can be used to identify the path of coefficient estimates.

```
R> print(fit)
```

```
Call:  glmnet(x = glmnet.data$x, y = glmnet.data$y, family = "binomial",      weights = gl
```

```
        Df     %Dev Lambda
 [1,]   2 0.006051 0.3914
 [2,]   3 0.053980 0.3736
 [3,]   3 0.098090 0.3566
 [4,]   3 0.138700 0.3404
 [5,]   3 0.176400 0.3249
 [6,]   3 0.211400 0.3102
 [7,]   3 0.244000 0.2961
 [8,]   3 0.274500 0.2826
 [9,]   3 0.303100 0.2698
[10,]   3 0.330000 0.2575
```

```
[11,]  3 0.355400 0.2458
[12,]  3 0.379400 0.2346
[13,]  3 0.402200 0.2240
[14,]  3 0.423700 0.2138
[15,]  3 0.444200 0.2041
[16,]  3 0.463800 0.1948
[17,]  3 0.482400 0.1859
[18,]  3 0.500200 0.1775
[19,]  3 0.517200 0.1694
[20,]  3 0.533500 0.1617
[21,]  4 0.550800 0.1544
[22,]  4 0.568300 0.1474
[23,]  4 0.585100 0.1407
[24,]  4 0.601300 0.1343
[25,]  6 0.617300 0.1282
```
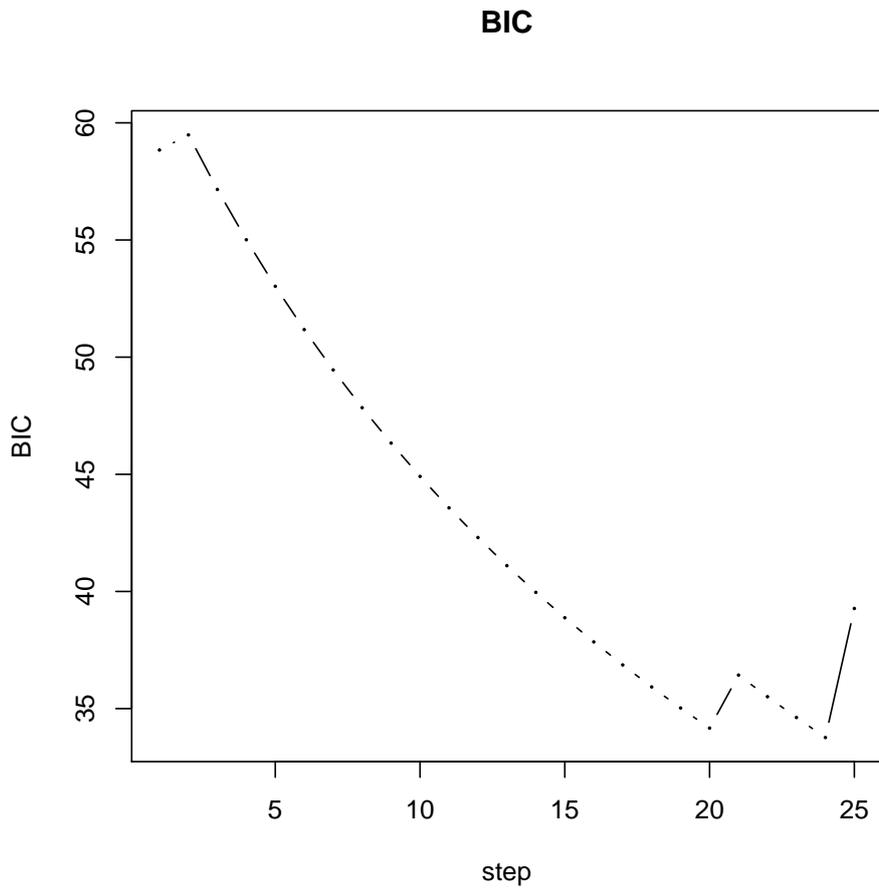
BIC

Figure 1: Plot of Bayesian Information Criteria across the regularization path for the fitted `glmnetcr` object using the `diabetes` data.
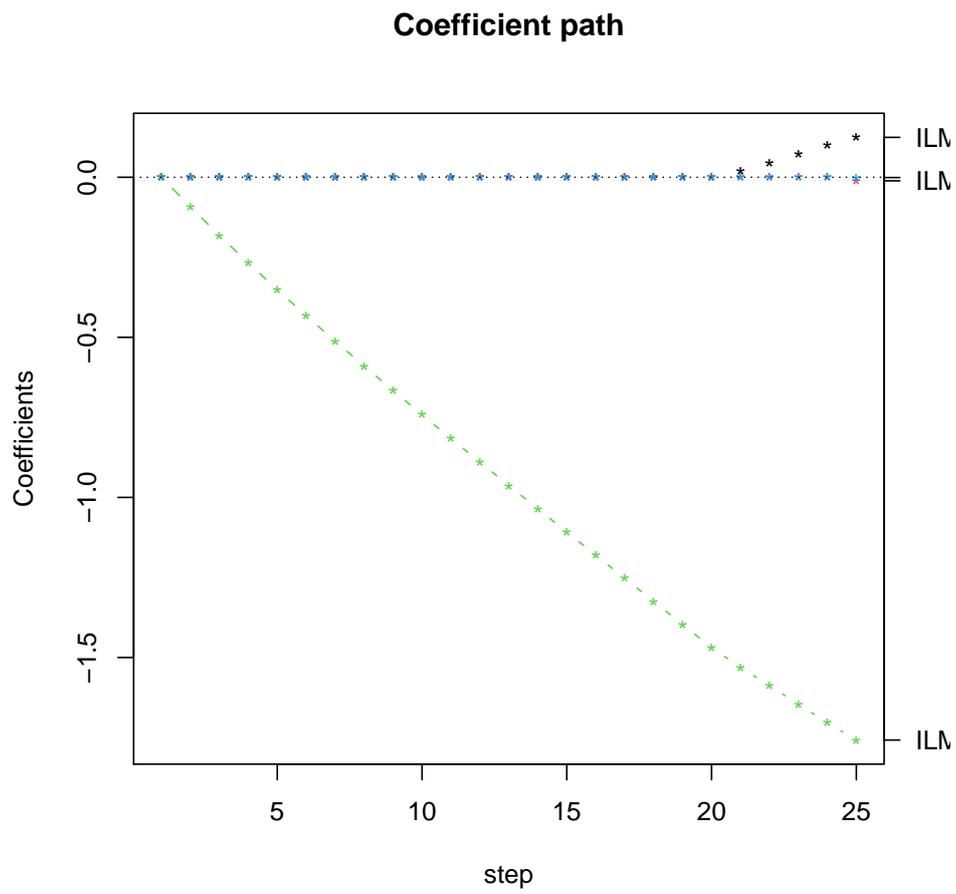
Coefficient path

Figure 2: Plot of estimated coefficients across the regularization path for the fitted `glmnetcr` object using the `diabetes` data.

```
R> plot(fit, xvar = "step", type = "bic")
R> plot(fit, xvar = "step", type = "coefficients")
```

Note that when plotting, the horizontal axis can be `"norm"`, `"lambda"`, or `"step"`, however extractor functions for `glmnetcr` generally require the step to be selected, so we have selected `xvar = "step"` in this example. The vertical axis can be `"coefficients"`, `"aic"`, or `"bic"`. As one can see, there is a multitude of models fit from one call to `glmnetcr`. To faciliate extraction of best fitting models using commonly used criterion, the `select.glmnetcr` function can be used. The `select.glmnetcr` function extracts the best fitting model from the solution path, where the `which` parameter allows one to select either AIC or by default, BIC.

```
R> BIC.step <- select.glmnetcr(fit)
R> BIC.step

s23
 24

R> AIC.step <- select.glmnetcr(fit, which = "AIC")
R> AIC.step

s23
 24
```

In this example, Step 23 (`s23`) corresponds to the model attaining the minimum BIC. The `coef` function returns all estimated coefficients for a `glmnetcr` fitted model, including the intercept which is returned as `a0` as well as the estimated slope and threshold estimates stored in `beta`. The coefficient estimates are returned for a specific step of the regularization path by specifying the step number, `s`, to extract. The `nonzero.glmnetcr` function returns only those non-zero coefficient estimates for a selected model. This latter function is useful when the number of predictor variables is large.

```
R> coefficients<-coef(fit, s = BIC.step)
R> coefficients$a0

     s23
3.192533

R> sum(coefficients$beta != 0)

[1] 4

R> nonzero.glmnetcr(fit, s = BIC.step)

$a0
     s23
3.192533

$beta
 ILMN_1705116  ILMN_1759232            cp1           cp2
 6.614969e-03 -2.035163e-02  1.894492e+00  3.032420e-15
```

A forward continuation ratio model can be fit using the syntax

```
R> fit <- glmnetcr(x, y, method = "forward")
```

Again, the `select.glmnetcr` function extracts the best fitting model from the solution path, where the `which` parameter allows one to select either AIC or by default, BIC.

```
R> BIC.step <- select.glmnetcr(fit)
R> BIC.step

s15
 16
```

As before, the parameter estimates corresponding to the model attaining the minimum BIC can be extracted using the following code.

```
R> coefficients<-coef(fit, s = BIC.step)
R> coefficients$a0

      s15
-2.588566

R> sum(coefficients$beta != 0)

[1] 3

R> nonzero.glmnetcr(fit, s = BIC.step)

$a0
      s15
-2.588566

$beta
 ILMN_1759232              cp1             cp2
 1.435081e-02 -1.354540e+00  4.062327e-15
```

Note that the `glmnetcr` function fits a penalized constrained continuation ratio model; therefore for K classes, there will be $K - 1$ intercepts representing the cutpoints between adjacent classes. In this package, the nomenclature for these cutpoints is to use "cp$k$" where $k = 1, \ldots, K - 1$. In this dataset, $K = 3$ so the intercepts are `cp1` and `cp2` with `a0` being an offset. When using the BIC to select the final forward continuation ratio model, the only probe having a non-zero coefficient estimate `ILMN_1759232` which corresponds to the insulin receptor substrate 1 (IRS1) gene which is biologically meaningful.

Continuation ratio models predicts conditional probabilities so a new method to extract the fitted probabilities and predicted class was created. The `predict` returns the AIC, BIC, predicted class, and the fitted probabilities for the K classes for all steps along the regularization path. By default the training data is used to obtain model predictions, though predicted class

and fitted probabilities can be obtained for a test dataset by specifying a different dataset using the `newx` parameter. The `fitted` function extracts the AIC, BIC, predicted class, and the fitted probabilities for the K classes for a specific step of the regularization path by specifying the parameter `s`.

```
R> hat<-fitted(fit, s = BIC.step)
R> names(hat)

[1] "BIC"   "AIC"   "class" "probs"

R> table(hat$class, y)
```

```
                          y
                           control impaired fasting glucose
  control                        8                         1
  impaired fasting glucose       0                         6
  type 2 diabetes                0                         0
                          y
                           type 2 diabetes
  control                                0
  impaired fasting glucose               0
  type 2 diabetes                        9
```

5. Summary

Herein we have described the **glmnetcr** package which works in conjunction with the **glmnet** package in the R programming environment. The package provides methods for fitting either a forward or backward penalized continuation ratio model. Moreover, the likelihood-based penalized constrained continuation ratios models have been demonstrated to have good performance in simulation studies and when applied to microarray gene expression datasets (Archer and Williams 2012), as well as in comparison to penalized Bayesian continuation ratio models using their encoded sparsity priors (Kiiveri 2008). A similar package, **glmpathcr**, which uses the **glmpath** fitting algorithm for fitting a penalized constrained continuation ratio model has also been developed and is available for download from the Comprehensive R Archive Network. Functions for extracting coefficients, extracting non-zero coefficients, and obtaining fitted probabilities and predicted class in the **glmpathcr** package follow those in **glmnetcr** and both packages have similar performance (Archer and Williams 2012). Therefore either the **glmpathcr** or **glmnetcr** package should be helpful when predicting an ordinal response for datasets where the number of covariates exceeds the number of available samples.

6. Acknowledgments

References

Archer KJ, Williams AA (2012). "L_1 Penalized Continuation Ratio Models for Ordinal Response Prediction Using High-Dimensional Datasets." *Statistics in Medicine*, **31**, 1464–1474.

Armstrong B, Sloan M (1989). "Ordinal Regression Models for Epidemiologic Data." *American Journal of Epidemiology*, **129**, 191–204.

Bender R, Benner A (2000). "Calculating Ordinal Regression Models in SAS and S-PLUS." *Biometrical Journal*, **42**, 677–699.

Cox D (1975). "Partial Likelihood." *Biometrika*, **62**, 269–276.

Friedman J, Hastie T, Tibshirani R (2010). "Regularization Paths for Generalized Linear Models Via Coordinate Descent." *Journal of Statistical Software*, **33**, 1–22.

Kiiveri HT (2008). "A General Approach to Simultaneous Model Fitting and Variable Elimination in Response Models for Biological Data with Many More Variables than Observations." *BMC Bioinformatics*, **9**, 195.

Park MY, Hastie T (2007). ***glmpath**: L1 Regularization Path for Generalized Linear Models and Cox Proportional Hazards Model.* R package version 0.94, URL http://CRAN.R-project.org/package=glmpath.

R Development Core Team (2009). *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org.

Tibshirani R (1996). "Regression Shrinkage and Selection Via the Lasso." *Journal of the Royal Statistical Society B*, **58**, 267–288.

Tibshirani R (1997). "The Lasso Method for Variable Selection in the Cox Model." *Statistics in Medicine*, **16**, 385–395.

Affiliation:

Kellie J. Archer
Division of Biostatistics
College of Public Health
The Ohio State University
1841 Neil Ave
Columbus, OH 43210
E-mail: archer.43@osu.edu
URL: https://cph.osu.edu/people/karcher