Citation: Archer, K.J.; Seffernick, A.E.; Sun, S.; Zhang, Y. ordinalbayes: Fitting Ordinal Bayesian Regression Models to High-Dimensional Data Using R. Stats 2022, 5, 371–384. https://doi.org/10.3390/stats5020021

Abstract: The stage of cancer is a discrete ordinal response that indicates the aggressiveness of disease and is often used by physicians to determine the type and intensity of treatment to be administered. For example, the FIGO stage in cervical cancer is based on the size and depth of the tumor as well as the level of spread. It may be of clinical relevance to identify molecular features from high-throughput genomic assays that are associated with the stage of cervical cancer to elucidate pathways related to tumor aggressiveness, identify improved molecular features that may be useful for staging, and identify therapeutic targets. High-throughput RNA-Seq data and corresponding clinical data (including stage) for cervical cancer patients have been made available through The Cancer Genome Atlas Project (TCGA). We recently described penalized Bayesian ordinal response models that can be used for variable selection for over-parameterized datasets, such as the TCGA-CESC dataset. Herein, we describe our ordinalbayes R package, available from the Comprehensive R Archive Network (CRAN), which enhances the runjags R package by enabling users to easily fit cumulative logit models when the outcome is ordinal and the number of predictors exceeds the sample size, such as for TCGA and other high-throughput genomic data. We demonstrate the use of this package by applying it to the TCGA cervical cancer dataset. Our ordinalbayes package can be used to fit models to high-dimensional datasets, and it effectively performs variable selection.

Keywords: cumulative logit; penalized models; LASSO; variable inclusion indicators; spike-and-slab

1. Introduction

Despite the advent of HPV vaccinations and effective screening programs, globally, cervical cancer is the fourth most common cancer among women [1]. The estimated number of new cases in 2020 is 604,127 with 341,831 deaths [2]. The stage of cervical cancer, as outlined in the International Federation of Gynecology and Obstetrics (FIGO) guidelines, is based on physical examinations, endoscopic procedures, and imaging. Specifically, the FIGO stage is based on the size and depth of the tumor as well as the level of spread [3]. It is important that the stage, a discrete ordinal response, be correct as it is used to guide treatment planning, counsel patients with respect to prognosis, and to determine whether the patient meets eligibility criteria for available clinical trials or other research studies [4,5]. Unfortunately, there is still debate as to whether surgical or non-invasive radiological modalities for identifying parametrial and lymph node involvement is preferred when staging a patient [4]. Thus, it is clinically relevant to identify molecular features from high-throughput genomic assays that are associated with the stage of cervical cancer to elucidate pathways related to tumor aggressiveness, identify improved molecular features that may be useful for staging, and identify therapeutic targets.

Penalized frequentist models have been widely applied when analyzing high-dimensional data. Such models were initially described for linear [6] and logistic [7] regression and subsequently for ordinal response models [8–10]. However, when applying penalized frequentist
models, the penalty parameter, or vector of parameters in the case of elastic net, must be selected by the analyst. As a result, the coefficient estimates from the resulting model are conditional on that penalty parameter. For that reason, penalized Bayesian models were developed for the linear [11–14] and logistic [15–17] regression settings. We also recently described penalized Bayesian models for the ordinal response setting [18] and demonstrated that our penalized Bayesian cumulative logit model has improved variable selection performance when compared to penalized frequentist cumulative logit models [19].

Herein, we describe our ordinalbayes R package, which enhances the runjags R package [20] by enabling users to easily fit penalized Bayesian cumulative logit models. The ordinalbayes function can be used to fit LASSO, normal spike-and-slab, double exponential spike-and-slab, and regression-based variable inclusion indicator Bayesian models. Variable selection can be performed using the Bayes factor or using the posterior distributions of the variable inclusion indicators directly. In the following sections, we describe our implementation and describe the syntax required for each of our Bayesian models. We then illustrate the functions in the ordinalbayes R package using two examples where we were interested in identifying transcripts important to predicting the FIGO stage in cervical cancer patients using high-throughput gene expression data. A small example is provided in Appendix A.

2. Materials and Methods
2.1. Ordinal Bayesian Models and R Syntax

We previously described four penalized cumulative logit Bayesian models that can be fit when the covariate space is high-dimensional [18]. This includes a regression-based variable inclusion indicator ordinal model, a LASSO ordinal model, a normal spike-and-slab ordinal model, and a double exponential spike-and-slab ordinal model. To introduce our penalized cumulative logit Bayesian models, we let \( Y_1, \ldots, Y_n \) represent the ordinal responses for \( n \) subjects, which can take on one of 1, \( \ldots, K \) ordinal response levels, with \( K \) representing the number of ordinal levels. Let \( x_i = (x_{i1}, x_{i2}, \ldots, x_{iP})' \) represent the vector of covariates for subject \( i \), where \( P \) represents the number of predictors. When assuming proportional odds, the effect of each covariate is constant across all ordinal response levels such that the slope for the ordinal responses are parallel. For each ordinal level \( k = 1, 2, \ldots, K-1 \), let \( \beta = (\beta_1, \beta_2, \ldots, \beta_P)' \) denote a vector of unknown regression coefficients. The cumulative logit model is

\[
\log \left[ \frac{Pr(Y_i \leq k | x_i)}{Pr(Y_i > k | x_i)} \right] = a_k - \beta'x_i, \quad k = 1, 2, \ldots, K-1,
\]

where \( Pr(Y_i \leq k | x_i) \) is the cumulative probability of the event \( Y_i \leq k \) given \( x_i \). The thresholds differentiate between the \( K \) ordinal levels and must satisfy the constraint \( -\infty = a_0 < a_1 < a_2 < \cdots < a_{K-1} < a_K = \infty \).

Herein, we describe our ordinalbayes package that enhances the functionality of the runjags package by providing functions specific to fitting these four penalized ordinal Bayesian models and extracting results of interest. We also provide an overview of each model. Tables summarizing the package functions and syntax appears in Appendix C.

The primary function for model fitting in the ordinalbayes package is ordinalbayes.

function (formula, data, x = NULL, subset, center = TRUE, scale = TRUE, a = 0.1, b = 0.1, model = "regressvi", gamma.ind = "fixed", pi.fixed = 0.05, c.gamma = NULL, d.gamma = NULL, alpha.var = 10, sigma2.0 = NULL, sigma2.1 = NULL, coerce.var=10, lambda0 = NULL, adaptSteps = 5000, burnInSteps = 5000, nChains = 3, numSavedSteps = 9999, thinSteps = 3, parallel = TRUE, seed = NULL, quiet = FALSE)
of the equation. Unpenalized predictors are variables such as age that we include in the model without applying any shrinkage of their corresponding parameter estimates. When unpenalized predictors are included as covariates in the model, the user can specify the variance associated with the corresponding model parameters (default \( \text{coerce.var} = 10 \)). If no unpenalized predictor variables are included, the model formula should be \( y \sim 1 \) (representing the intercept). The user can subset the data.frame prior to model fitting, for example, \( \text{subset} = \{ \text{race} == \text{"white"} \} \). To specify the penalized covariates in the model, the user should pass the data.frame to the \( x \) parameter, indicating the relevant columns of covariates. By default, the penalized covariates are centered (\( \text{center} = \text{TRUE} \)) and scaled (\( \text{scale} = \text{TRUE} \)).

The selected parameters are initialized prior to updating through MCMC. For one chain, the \( k - 1 \) ordinal thresholds, \( \alpha_k \), are initialized to the logit of the cumulative response probabilities, which is equivalent to the estimated \( k - 1 \) thresholds in an intercept-only model

\[
\alpha_k = \log \left( \frac{\sum_{i=1}^{n} \sum_{m=1}^{k} Y_{ik} / n}{1 - \sum_{i=1}^{n} \sum_{m=1}^{k} Y_{ik} / n} \right).
\]

For multiple chains, initial values for the \( \alpha_k \) terms for chains beyond the first chain are sampled from a Normal(0, 0.5) distribution and then sorted to impose the \( \alpha_1 < \cdots < \alpha_{k-1} \) order restriction. Within the MCMC, the \( \alpha_k \) terms are sampled from a Normal \( (0, \sigma_k^2) \), and users can adjust the variance by specifying \( \alpha\text{.var} \) (default 10 such that the precision is 0.10). All penalized coefficients (\( \beta_j \) for \( j = 1, \ldots, P \)) are initialized to zero.

Other relevant parameters common to all model types include: \( \text{nChains} \), the number of parallel chains for the model (default 3); \( \text{adaptSteps} \), the number of iterations for adaptation (default 5000); \( \text{burnInSteps} \), the number of iterations of the Markov chain to run (default 5000); \( \text{numSavedSteps} \), the number of saved steps per chain (default 9999); and \( \text{thinSteps} \), the thinning interval for monitors (default 3). Provided the user will be running the model on a machine with multiple processors, the computational speed can be improved by running the chains in parallel by specifying \( \text{parallel} = \text{TRUE} \). When \( \text{parallel} = \text{TRUE} \), \text{runjags} executes the MCMC sampling using \( \text{nChains} \) parallel processors. To ensure the user can obtain reproducible results, \( \text{seed} \) accepts an integer that is used to set the random seed. The output from JAGS can be suppressed by specifying \( \text{quiet} = \text{TRUE} \). The user can fit one of four available Bayesian models. A list of the parameters the user can set for all four models is provided in Table A1. Following Section 2.1.1, which describes applying \( \text{ordinalbayes} \) to Bioconductor objects, each of the four models is described along with the relevant arguments that must be specified by the user. A list of the parameters the user needs to set for each specific model is provided in Table A2.

2.1.1. Use with Bioconductor Objects: \text{SummarizedExperiment} and \text{ExpressionSet}

When analyzing data processed using the DESeq2 Bioconductor package, the genomic feature object is of class \text{DESeqTransform}, which is a \text{SummarizedExperiment}, and therefore, the phenotypic data are accessed using the \text{colData} extractor function. When analyzing data processed using packages that structure the genomic feature object as a Biobase \text{ExpressionSet}, the phenotypic data are accessed using the \text{pData} extractor function. Therefore, in the \text{ordinalbayes} call, data should be either a \text{colData()} or \text{pData()} call to the genomic feature object. Again, the \text{ordinalbayes} function accepts a model formula that specifies the ordinal response on the left-hand side of the equation and any unpenalized predictor variable(s) from the phenotypic dataset on the right-hand side of the equation. If no unpenalized predictor variables are included, the model formula should be \( y \sim 1 \) (representing the intercept).

When specifying the penalized covariates in the model, the user should pass to the \( x \) parameter the appropriate call for extracting the genomic feature data from the object. For \text{SummarizedExperiment} objects, the genomic features to be penalized are accessed using the \text{assay()} extractor function. For \text{ExpressionSet} objects, the genomic features to be penalized are accessed using the \text{exprs()} extractor function. The user can also pass...
a matrix to $x$; however, the user needs to carefully verify that the observations in the $x$ matrix are appropriately aligned to the phenotypic data. Note that the number of rows in both data and $x$ should be the same, such that the transpose of assay or exprs should be supplied to $x$.

2.2. Regression-Based Variable Inclusion Indicator Ordinal Model

By default, the model that is fit is the regression-based variable inclusion indicator Bayesian model, specified by model = "regressvi". This model takes the form

$$
\log \left[ \frac{Pr(Y_i \leq k|x_i)}{Pr(Y_i > k|x_i)} \right] = a_k - \sum_{j=1}^{p} \gamma_j \beta_j x_{ij}, \quad \text{for } k = 1, 2, \ldots, K - 1
$$

$$
\beta_j | \lambda \sim \text{DE}(0, 1/\lambda), \quad \text{for } j = 1, \ldots, p
$$

$$
\lambda \sim \text{Gamma}(a, b)
$$

$$
a_k \sim \text{Normal}(0, \sigma_{a_k}^2), \quad a_1 < a_2 < \ldots < a_{K-1}, \quad \text{for } k = 1, 2, \ldots, K - 1
$$

$$
\gamma_j \sim \text{Bernoulli}(\pi_j), \quad \text{for } j = 1, \ldots, p
$$

$$
\pi_j = t \text{ or } \pi_j \sim \text{Beta}(c, d), \quad \text{for } j = 1, \ldots, p
$$

and assumes the penalized coefficients are from a Laplace (or double exponential) distribution with parameter $\lambda$ and that $\lambda$ is from a Gamma distribution with parameters $a$ and $b$. Based on our extensive simulations [19], model performance is not affected by choices of $a$ and $b$, so we provide defaults of 0.1 for both. The variable inclusion indicator $\gamma_j$ is assumed to follow a Bernoulli distribution with parameter $\pi_j$. The user can use either a fixed constant prior (default) or a random prior. When using a fixed constant prior, the user must specify both gamma.ind="fixed" and set pi.fixed to some constant in the (0, 1) interval (default is 0.05). Alternatively, a random prior for $\pi_j$ is achieved by specifying both gamma.ind="random" and parameter values (c.gamma and d.gamma) for the Beta distribution. Values of c.gamma and d.gamma should be selected such that the mean of the Beta distribution for the variable inclusion indicators corresponds to the anticipated proportion of covariates truly associated with the ordinal response, given by $c/(c+d)$, while considering that the variance is given by

$$
\frac{cd}{(c+d)^2(c+d+1)}.
$$

If unpenalized coefficients are included in the model, their coefficients are $\zeta \sim \text{Normal}(0, \sigma_{\text{coerce}}^2)$.

2.3. Lasso Ordinal Model

The LASSO Bayesian ordinal model can be fit by specifying model="lasso". This model assumes the penalized coefficients $\beta_j$ for $j = 1, \ldots, P$ are from independent Laplace (or double exponential) distributions with parameter $\lambda$ and that $\lambda$ is from a Gamma distribution with parameters $a$ and $b$.

$$
\log \left[ \frac{Pr(Y_i \leq k|x_i)}{Pr(Y_i > k|x_i)} \right] = a_k - \sum_{j=1}^{p} \beta_j x_{ij}, \quad \text{for } k = 1, 2, \ldots, K - 1
$$

$$
\beta_j | \lambda \sim \text{DE}(0, 1/\lambda), \quad \text{for } j = 1, \ldots, p
$$

$$
\lambda \sim \text{Gamma}(a, b)
$$

$$
a_k \sim \text{Normal}(0, \sigma_{a_k}^2), \quad a_1 < a_2 < \ldots < a_{K-1}, \quad \text{for } k = 1, 2, \ldots, K - 1
$$

As previously mentioned, model performance is not affected by choices of $a$ and $b$, so we provide defaults of 0.1 for both. If unpenalized coefficients are included in the model, their coefficients are $\zeta \sim \text{Normal}(0, \sigma_{\text{coerce}}^2)$. 


2.4. Normal Spike-and-Slab Ordinal Model

The normal spike-and-slab Bayesian ordinal model can be fit by specifying model="normalss". This model is given by

\[
\log \left( \frac{\Pr(Y_i \leq k|x_i)}{\Pr(Y_i > k|x_i)} \right) = a_k - \sum_{j=1}^{p} \beta_j x_{ij}, \quad \text{for } k = 1, 2, \ldots, K - 1
\]

\[
\beta_j | \gamma_j \sim (1 - \gamma_j) \times \text{Normal}(0, \sigma_0^2) + \gamma_j \times \text{Normal}(0, \sigma_1^2), \quad \text{for } j = 1, \ldots, p
\]

\[
a_k \sim \text{Normal}(0, \sigma_0^2), \quad a_1 < a_2 < \ldots < a_{K-1}, \quad \text{for } k = 1, 2, \ldots, K - 1
\]

\[
\gamma_j \sim \text{Bernoulli}(\pi_j), \quad \text{for } j = 1, \ldots, p
\]

\[
\pi_j = t \text{ or } \pi_j \sim \text{Beta}(c, d), \quad \text{for } j = 1, \ldots, p
\]

When fitting this model, the user is required to specify the variance for the slab \((\sigma_0^2)\) by setting \texttt{sigma2.0} to a small positive value (e.g., 0.01) and variance for the slab \((\sigma_1^2)\) by setting \texttt{sigma2.1} to a large positive value (e.g., 20). As with the regression-based variable inclusion indicator Bayesian model, the variable inclusion indicator \(\gamma_j\) is assumed to follow a Bernoulli distribution with parameter \(\pi_j\). The user can use either a fixed constant prior (default) or a random prior. When using a fixed constant prior, the user must specify both \texttt{gamma.ind="fixed"} and set \texttt{pi.fixed} to some constant in the \((0, 1)\) interval (default is 0.05). Alternatively, a random prior for \(\pi_j\) is achieved by specifying both \texttt{gamma.ind="random"} and parameter values \((c, \texttt{gamma} and d, \texttt{gamma})\) for the Beta distribution. If unpenalized coefficients are included in the model, their coefficients are \(\zeta \sim \text{Normal}(0, \sigma^2_{\text{coeff}})\).

2.5. Double Exponential Spike-and-Slab Ordinal Model

The double exponential spike-and-slab ordinal model can be fit by specifying model="dess" and is given by

\[
\log \left( \frac{\Pr(Y_i \leq k|x_i)}{\Pr(Y_i > k|x_i)} \right) = a_k - \sum_{j=1}^{p} \beta_j x_{ij}, \quad \text{for } k = 1, 2, \ldots, K - 1
\]

\[
\beta_j | \lambda, \gamma_j \sim (1 - \gamma_j) \times \text{DE}(0, 1/\lambda_0) + \gamma_j \times \text{DE}(0, 1/\lambda), \quad \text{for } j = 1, \ldots, p
\]

\[
\lambda \sim \text{Gamma}(a, b)
\]

\[
a_k \sim \text{Normal}(0, \sigma_0^2), \quad a_1 < a_2 < \ldots < a_{K-1}, \quad \text{for } k = 1, 2, \ldots, K - 1
\]

\[
\gamma_j \sim \text{Bernoulli}(\pi_j), \quad \text{for } j = 1, \ldots, p
\]

\[
\pi_j = t \text{ or } \pi_j \sim \text{Beta}(c, d), \quad \text{for } j = 1, \ldots, p
\]

When fitting this model the user is required to specify the parameter for the spike \((\lambda_0)\) using \texttt{lambda0}, which should be a large positive value (e.g., 20), while the slab is taken to be a double exponential distribution with parameter \(\lambda\) where that \(\lambda\) is from a Gamma distribution with parameters \(a\) and \(b\). As with the regression-based variable inclusion indicator and Normal spike-and-slab models, the variable inclusion indicator \(\gamma_j\) is assumed to follow a Bernoulli distribution with parameter \(\pi_j\). The user can use either a fixed constant prior (default) or a random prior. When using a fixed constant prior, the user must specify both \texttt{gamma.ind="fixed"} and set \texttt{pi.fixed} to some constant in the \((0, 1)\) interval (default is 0.05). Alternatively, a random prior for \(\pi_j\) is achieved by specifying both \texttt{gamma.ind="random"} and parameter values \((c, \texttt{gamma} and d, \texttt{gamma})\) for the Beta distribution. If unpenalized coefficients are included in the model, their coefficients are \(\zeta \sim \text{Normal}(0, \sigma^2_{\text{coeff}})\).

2.6. Other Package Functions

The \texttt{ordinalbayes} function yields an object of class \texttt{ordinalbayes}. Generic functions have been specifically tailored to extract meaningful results from the resulting MCMC chain. The \texttt{print} function returns several summaries from the MCMC output for each parameter monitored, including: the 95th lower confidence limit for the highest posterior density
(HPD) credible interval (Lower95), the median value (Median), the 95th upper confidence limit for the HPD credible interval (Upper95), the mean value (Mean), the sample standard deviation (SD), the mode of the variable (Mode), the Monte Carlo standard error (MCerr), percent of SD due to MCMC (MC%ofSD), effective sample size (SSeff), autocorrelation at a lag of 30 (AC.30), and the potential scale reduction factor (psrf). The plot function provides a trace of the sampled output and optionally the density estimate for each variable in the chain. This function additionally adds the appropriate beta and gamma labels for each penalized variable name.

When identifying important covariates, the regression-based variable inclusion indicator, normal spike-and-slab, and double exponential spike-and-slab Bayesian ordinal models all incorporate a variable inclusion indicator, $\gamma_j$, in the model. Variable selection can be based on whether the posterior mean of $\gamma_j$ exceeds a pre-specified threshold. Alternatively, we can use the Bayes factor to test the hypotheses $H_{0j}: \gamma_j = 0$ versus $H_{aj}: \gamma_j > 0$, where the null hypothesis is rejected for feature $j$ if the Bayes factor exceeds a pre-specified threshold.

For the LASSO, normal spike-and-slab, and double exponential spike-and-slab Bayesian ordinal models, the Bayes factor can be used to test an interval null hypothesis $H_{0j}: |\beta_j| \leq \epsilon$ versus $H_{aj}: |\beta_j| > \epsilon$, where $\epsilon$ is a small positive value that is close to 0. For the regression-based variable inclusion indicator Bayesian ordinal model, the Bayes factor can be used to test $H_{0j}: |\gamma_j\beta_j| \leq \epsilon$ versus $H_{aj}: |\gamma_j\beta_j| > \epsilon$. Note that for the Bayesian LASSO, no variable inclusion indicators are incorporated, so variable selection can only be performed using the Bayes factor for $\beta$. The summary function requires an ordinalbayes object, and the user can specify epsilon (default 0.1) for testing the null hypothesis that $H_{0j}: |\beta_j| \leq \epsilon$. The output from summary is a list containing the following components: alphamatrix, the MCMC output for the threshold parameters; betamatrix, the MCMC output for the penalized parameters; zetamatrix, The MCMC output for the unpenalized parameters (if included); gammamatrix, the MCMC output for the variable inclusion parameters (not available when model = "lasso"); gammamean, the posterior mean of the variable inclusion indicators (not available when model = "lasso"); gamma.BayesFactor, Bayes factor for the variable inclusion indicators (not available when model = "lasso"); Beta.BayesFactor, Bayes factor for the penalized parameters; and lambdamatrix, the MCMC output for the penalty parameter (not available when model="normalss"). The coef function also accepts an ordinalbayes object and returns a function (default is method=mean) of the posterior distribution of the penalized parameter estimates and variable inclusion indicators.

The predict function accepts an ordinalbayes object and optionally allows the user to specify new data for unpenalized predictors and the penalized predictors by invoking neww = and newx =, respectively. If neww and newx are not supplied, the original data are used for prediction. The model.select parameter allows the user to obtain model predictions through one of three different methods. When model.select = "average" (default), the mean coefficient values over the MCMC chain are used to estimate fitted probabilities; the predicted response is attaining the maximum fitted probability. When model.select = "median", the median coefficient values over the MCMC chain are used to estimate fitted probabilities; the predicted ordinal response is attaining the maximum fitted probability. When model.select = "max.predicted.class", each step in the chain is used to calculate fitted probabilities and the ordinal response, then the final predicted ordinal response is taken as that ordinal response level that is most frequently predicted.

The function fitted is synonymous with predict.

2.7. Analysis of Cervical Cancer Dataset

We downloaded the transcript-level HTSeq count data for the 309 subjects from the The Cancer Genome Atlas Cervical Squamous Cell Carcinoma and Endocervical Adenocarcinoma (TCGA-CESC) project [21] having transcriptome profiling performed using the TCGAbiolinks Bioconductor package [22]. We then restricted attention to the 253 cervical cancer subjects with a primary diagnosis of squamous cell carcinoma. Subsequently, we re-
moved one subject whose sample was FFPE preserved, one subject with metastatic disease, two subjects who contributed only solid normal tissue, and seven subjects lacking FIGO stage. This left 242 subjects in Stage I ($N = 124$), II ($N = 61$), and III-IV ($N = 57$). Using the DESeq2 Bioconductor package [23], we performed differential expression analysis using the stage as the independent predictor in the negative binomial model. We then applied the regularized log transformation to robustly transform the count data to a log2 scale to stabilize the variance and then filtered the resulting dataset to retain transcripts that had a mean expression $> 0.5$ and $FDR < 0.10$ from the stage I versus stages III/IV contrast.

We fit a regression-based variable inclusion indicator Bayesian ordinal model using a Beta(0.01, 0.19) hyperprior for the $\pi_j$ using the runjags package to run three parallel chains with 5000 burn-in, 5000 tuning steps, and thinned to keep every third step in the sampling process to reduce auto-correlation in our posterior samples, and we kept 9999 saved steps per chain. Convergence was assessed using Gelman and Rubin’s potential scale reduction factor (PSRF).

3. Results

There were 1137 transcripts that were differentially expressed at a Benjamini–Hochberg $FDR < 0.05$ and 2009 transcripts that were differentially expressed at a Benjamini-Hochberg $FDR < 0.10$ when examining the contrast between stage I and stages III/IV. These 2009 transcripts were retained for Bayesian modeling. Forty transcripts had a Bayes factor $> 4$ when testing $H_{0j} : |\gamma_j \beta_j| \leq 0.1$ versus $H_{aj} : |\gamma_j \beta_j| > 0.1$. Forty-one transcripts had a Bayes factor $> 4$ when testing $H_{0j} : \gamma_j = 0$ versus $H_{aj} : \gamma_j = 1$ (Table 1). Notably, the features were the same with the exception that Bayes factor testing $\gamma_j = 0$ additionally identified ENSG00000115548 (Gene symbol KDM3A).

Table 1. Transcripts significant from the regression-based variable inclusion indicator Bayesian ordinal model when testing $H_{0j} : \gamma_j = 0$ versus $H_{aj} : \gamma_j = 1$ using the Bayes factor and a threshold of 4. Annotation information obtained on 28 February 2022 from https://www.ncbi.nlm.nih.gov/gene, https://www.genecards.org, and https://lncipedia.org.

<table>
<thead>
<tr>
<th>Ensemble ID</th>
<th>Gene Symbol</th>
<th>Chr</th>
<th>$\hat{\gamma}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENSG000000076344</td>
<td>RGS11</td>
<td>16</td>
<td>0.179</td>
</tr>
<tr>
<td>ENSG00000077274</td>
<td>CAPN6</td>
<td>X</td>
<td>0.264</td>
</tr>
<tr>
<td>ENSG00000101888</td>
<td>NXT2</td>
<td>X</td>
<td>0.194</td>
</tr>
<tr>
<td>ENSG00000115548</td>
<td>KDM3A</td>
<td>2</td>
<td>0.174</td>
</tr>
<tr>
<td>ENSG00000122884</td>
<td>P4HA1</td>
<td>10</td>
<td>0.186</td>
</tr>
<tr>
<td>ENSG00000125430</td>
<td>HS3ST3B1</td>
<td>17</td>
<td>0.286</td>
</tr>
<tr>
<td>ENSG00000131370</td>
<td>SH3BP5</td>
<td>3</td>
<td>0.175</td>
</tr>
<tr>
<td>ENSG00000135443</td>
<td>KRT85</td>
<td>12</td>
<td>0.334</td>
</tr>
<tr>
<td>ENSG00000136457</td>
<td>CHAD</td>
<td>17</td>
<td>0.179</td>
</tr>
<tr>
<td>ENSG00000138398</td>
<td>PP1G</td>
<td>2</td>
<td>0.240</td>
</tr>
<tr>
<td>ENSG00000150636</td>
<td>CCDC102B</td>
<td>18</td>
<td>0.281</td>
</tr>
<tr>
<td>ENSG00000161277</td>
<td>THAP8</td>
<td>19</td>
<td>0.283</td>
</tr>
<tr>
<td>ENSG00000163510</td>
<td>CWC22</td>
<td>2</td>
<td>0.301</td>
</tr>
<tr>
<td>ENSG00000164485</td>
<td>IL22RA2</td>
<td>6</td>
<td>0.196</td>
</tr>
<tr>
<td>ENSG00000164651</td>
<td>SP8</td>
<td>7</td>
<td>0.231</td>
</tr>
<tr>
<td>ENSG00000166091</td>
<td>CMTM5</td>
<td>14</td>
<td>0.215</td>
</tr>
<tr>
<td>ENSG00000166342</td>
<td>NETO1</td>
<td>18</td>
<td>0.197</td>
</tr>
<tr>
<td>ENSG00000171121</td>
<td>KCNMNB3</td>
<td>3</td>
<td>0.186</td>
</tr>
</tbody>
</table>
Table 1. Cont.

<table>
<thead>
<tr>
<th>Ensemble ID</th>
<th>Gene Symbol</th>
<th>Chr</th>
<th>( \gamma )</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENSG00000177173</td>
<td>Pseudogene, parent NAP1L4P1</td>
<td>1</td>
<td>0.258</td>
</tr>
<tr>
<td>ENSG00000180229</td>
<td>HERC2P3</td>
<td>15</td>
<td>0.196</td>
</tr>
<tr>
<td>ENSG00000188817</td>
<td>SNTN</td>
<td>3</td>
<td>0.236</td>
</tr>
<tr>
<td>ENSG00000197360</td>
<td>ZNF98</td>
<td>19</td>
<td>0.214</td>
</tr>
<tr>
<td>ENSG00000203601</td>
<td>LINC00970</td>
<td>1</td>
<td>0.316</td>
</tr>
<tr>
<td>ENSG00000225449</td>
<td>RAB6C-AS1</td>
<td>2</td>
<td>0.235</td>
</tr>
<tr>
<td>ENSG00000230201</td>
<td>Pseudogene, parent ATP6V0CP1</td>
<td>17</td>
<td>0.286</td>
</tr>
<tr>
<td>ENSG000002333996</td>
<td>Pseudogene, parent KDM3AP1</td>
<td>2</td>
<td>0.248</td>
</tr>
<tr>
<td>ENSG00000236138</td>
<td>DUX4L26</td>
<td>3</td>
<td>0.247</td>
</tr>
<tr>
<td>ENSG00000236819</td>
<td>LINC01563</td>
<td>17</td>
<td>0.311</td>
</tr>
<tr>
<td>ENSG00000250602</td>
<td>Inc-ALDH7A1-1</td>
<td>5</td>
<td>0.246</td>
</tr>
<tr>
<td>ENSG00000253923</td>
<td>Pseudogene, parent HSPE1</td>
<td>8</td>
<td>0.302</td>
</tr>
<tr>
<td>ENSG00000256980</td>
<td>KDHC1L</td>
<td>6</td>
<td>0.207</td>
</tr>
<tr>
<td>ENSG00000259083</td>
<td>Inc-TRAPPC6B-1</td>
<td>14</td>
<td>0.263</td>
</tr>
<tr>
<td>ENSG00000259134</td>
<td>LINC00924</td>
<td>15</td>
<td>0.352</td>
</tr>
<tr>
<td>ENSG00000260484</td>
<td>Inc-OPRK1-2</td>
<td>8</td>
<td>0.263</td>
</tr>
<tr>
<td>ENSG00000263612</td>
<td>Inc-ZNF517-4</td>
<td>8</td>
<td>0.228</td>
</tr>
<tr>
<td>ENSG00000264049</td>
<td>MIR4737</td>
<td>17</td>
<td>0.266</td>
</tr>
<tr>
<td>ENSG00000264954</td>
<td>PRR29-AS1</td>
<td>17</td>
<td>0.221</td>
</tr>
<tr>
<td>ENSG00000265579</td>
<td>Inc-CBLN2-1</td>
<td>18</td>
<td>0.227</td>
</tr>
<tr>
<td>ENSG00000271711</td>
<td>Pseudogene, parent SAP30</td>
<td>3</td>
<td>0.264</td>
</tr>
<tr>
<td>ENSG00000272071</td>
<td>Inc-PAPD7-2</td>
<td>5</td>
<td>0.279</td>
</tr>
<tr>
<td>ENSG00000276517</td>
<td>Lnc-TTC27-9</td>
<td>2</td>
<td>0.221</td>
</tr>
</tbody>
</table>

Many genes listed in Table 1 are relevant to cervical cancer, related cancers of the female reproductive system, or cancer in general. For example, in a tissue-based study, CAPN6 was not detected in normal cervical squamous epithelium, but its expression was observed in low-grade and increased further in high-grade squamous cervical intraepithelial lesions [24]. KDM3A is an epigenetic regulator that has been found to be highly expressed in cervical cancer tissues and involved in cervical cancer progression [25]. P4HA1 was included in a five-gene signature to predict cervical cancer prognosis [26]. A previous study suggested that CMTM5 is a tumor suppressor that is frequently methylated and thus loses function in cancer [27], including cervical cancer [28]. RAB6C has been shown to be aberrantly methylated in cervical cancer compared to normal tissues [29]. ALDH7A1 was among 30 genes that demonstrated a dose–response pattern with NNK, a tobacco carcinogen, in cervical cancer samples [30], implicating tobacco may be a causative factor in cervical cancer development in addition to HPV infection.

Other genes, while not yet described in cervical cancer, have been found to be prognostic in ovarian cancer (RGS11 [31], CHAD and CBLN2 [32], NETO1 [33], HSPE1 [34], and BIRC6, which Lnc-TTC27-9 is intronic to [35]). The expression of SH3BP5 is reduced in ovarian cancer samples compared to normal tissue and that silencing of Sab protein expression may lead to chemo-resistance [36]. The expression of SNTN has high discriminatory power to differentiate between normal tissue, serous borderline ovarian tumors, and serous ovarian carcinoma [37]. IL22RA2 is highly expressed in various tissues, including those in the female reproductive system [38]. With respect to genes associated with other cancers, NXT2 was among 12 genes used to define prognostic risk groups in melanoma [39]. A review article described that the aberrant expression of HS3ST3B1 is observed in many cancers, and the authors posited that HS3ST3B1 may act as a tumor-promoting enzyme [40]. The expression of KRT85 was found to be associated with overall survival in subjects with colon cancer [41].

When using the fitted model using the 2009 transcripts, only 16.9% of subjects were misclassified, with all misclassifications in Stage II. However, when fitting a parsimonious model including only the 41 transcripts in Table 1, the misclassification rate decreased to 11.6%. For evaluating the effectiveness of this multi-category classification, we evaluated
the hypervolume under ROC manifold [42,43], which was 0.865 (95% CI: 0.800, 0.914) for
the 41 transcript model, indicating good discrimination among the three stages.

4. Discussion

The ordinalbayes package is based on runjags and enables the user to easily fit penal-
ized ordinal Bayesian cumulative logit models to high-dimensional datasets. The package
includes methods for monitoring the mixing of chains (plot) and convergence (print).
It also includes a summary function that permits the user to estimate the Bayes factor for
testing an interval null hypothesis for \( \beta_j \) and for testing the null that \( \gamma_j = 0 \) to assist the
user with variable selection. The coef function uses the posterior distribution to return
summary estimates of the penalized \( \beta_j \) and the \( \gamma_j \) indicators. The predict (or equivalently,
fitted) function can be used to obtain the estimated ordinal response probabilities as well
as the predicted ordinal response level for each observation.

When applied to The Cancer Genome Atlas cervical cancer dataset, predictive perfor-
ance was excellent. When restricting attention to only the 41 transcripts with a Bayes
factor > 4, predictive performance yielded an overall misclassification error of 11.6%,
though the misclassification error increased from 0% for Stage I and III/VI in the full model
to 3.2% and 14.0%, respectively, in the reduced model. Interestingly, transcripts that were
identified have known associations with cervical cancer, cancers of the female reproductive
system, and other cancer in general. The syntax we used to analyze this dataset appears in
the Appendix B.

Supplementary Materials: Externally hosted supplementary file 1, can be downloaded at: https:
//github.com/kelliejarcher/ordinalbayes, Description: ordinalbayes R package on GitHub.

Author Contributions: Conceptualization, K.J.A.; methodology, Y.Z. and A.E.S.; software, K.J.A.,
Y.Z. and A.E.S.; validation, A.E.S., Y.Z. and S.S.; formal analysis, K.J.A.; writing—original draft
preparation, K.J.A.; writing—review and editing, A.E.S., Y.Z. and S.S.; funding acquisition, K.J.A. All
authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Cancer Institute of the National Institutes of
Health under Award Number R03CA245771. The content is solely the responsibility of the authors
and does not necessarily represent the official views of the National Institutes of Health.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The TCGA-CESC high-throughput gene expression data can be down-
loaded using the TCGAbiolinks Bioconductor package (Supplementary file 1).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

- MDPI: Multidisciplinary Digital Publishing Institute
- FIGO: International Federation of Gynecology and Obstetrics
- TCGA: The Cancer Genome Atlas Project
- CRAN: Comprehensive R Archive Network
- HPV: Human Papilloma Virus
- LASSO: Least Absolute Shrinkage and Selection Operator
- MCMC: Markov Chain Monte Carlo
- DE: Double exponential
- PSRF: potential scale reduction factor
- HTSeq: High-throughput sequencing
- FFPE: Formalin-Fixed Paraffin-Embedded
- FDR: False discovery rate
Appendix A. Example

The ordinalbayes R package is available from CRAN and github (https://github.com/kelliejarcher/ordinalbayes), where the latter includes installation instructions and the example code in this Appendix. For a toy example, subset data of the cervical cancer data are stored in the data.frame named cesc. This data.frame includes age_at_index, cigarettes_per_day, race, Stage, and expression of 41 transcripts. The regression-based variable inclusion model with random prior to $\pi$ can be fit after loading the ordinalbayes R package using the syntax:

```r
library("ordinalbayes")
data("cesc")
fit <- ordinalbayes(Stage~1, data = cesc,
  x = cesc[,5:45], model = "regressvi",
  gamma.ind = "random", c.gamma = 0.01, d.gamma = 0.19,
  seed = 26, adaptSteps=2000, burnInSteps=2000, numSavedSteps = 3000)
```

This took 2.93 min on a 13 inch MacBook Pro with four cores and 16GB RAM. To include age_at_index and cigarettes_per_day as unpenalized covariates while including the 41 gene expression covariates as penalized covariates, the syntax is

```r
fit.unpenalized <- ordinalbayes(Stage~age_at_index + cigarettes_per_day, 
data = cesc, x = cesc[,5:45], model = "regressvi",
  gamma.ind = "random", c.gamma = 0.01, d.gamma = 0.19,
  seed = 26, adaptSteps=2000, burnInSteps=2000, numSavedSteps = 3000)
```

You can evaluate various aspects of the MCMC results of the ordinalbayes object by issuing the print command.

```r
print(fit)
```

including the psrf to assess model convergence. Please note that to foster reproducibility of our output, we set the random seed. Subsequent runs using different seeds will produce different results due to the random nature of the MCMC sampling. To summarize the fitted model object,

```r
summary.fit<-summary(fit)
```

To identify which transcripts had a Bayes factor $> 4$ when testing $H_0 : |\gamma_j \beta_j| \leq 0.1$ versus $H_1 : |\gamma_j \beta_j| > 0.1$,

```r
names(which(summary.fit$BayesFactor>4))
```

Similarly, to identify which transcripts had a Bayes factor $> 4$ when testing $H_0 : \gamma_j = 0$ versus $H_1 : \gamma_j = 1$,

```r
names(which(summary.fit$gamma.BayesFactor>4))
```

To obtain the $\gamma$ estimates we used the following code:

```r
coefficients<-coeff(fit)
coefficients$gamma[which(summary.fit$gamma.BayesFactor>4)]
```

To obtain model predictions,

```r
phat<-predict(fit)
table(phat$class, cesc$Stage)
```

```
 1 2 3
1 120 15 2
2 4 32 13
3 0 14 42
```
Appendix B. Reproducing CESC Results Using Bioconductor Objects

The data used in this example are stored in the finalSet object. Because this object was derived using the DESeq2 BioConductor package, we load it first. Please note that due to the use of the default parameters for the number of saved steps per chain (9999) and the large size of this dataset, the model took 3.2 days to run on a 13 inch MacBook Pro with four cores and 16GB RAM. For those interested in running examples using this package, a smaller version of these data, reducedSet, which includes the 41 transcripts, may be used instead. Alternatively, parameters related to the number of steps can be reduced.

The regression-based variable inclusion model with random prior to $\pi$ was fit after loading the ordinalbayes R using the syntax:

```r
library("DESeq2")
library("ordinalbayes")
data(finalSet)
fitted.regressvi.random<-ordinalbayes(Stage~1, data=colData(finalSet),
  x=t(assay(finalSet)), model="regressvi",
gamma.ind="random", c.gamma=0.01, d.gamma=0.19, seed=26)
```

Again note that to foster reproducibility of our output, we set the random seed. Subsequent runs using different seeds will produce different results due to the random nature of the MCMC sampling.

We then summarize the fitted model object and to identify which transcripts had a Bayes factor $>4$ when testing $H_0: |\gamma_j \beta_j| \leq 0.1$ versus $H_a: |\gamma_j \beta_j| > 0.1$:

```r
summary.model.fit<-summary(fitted.regressvi.random)
names(which(summary.model.fit$Beta.BayesFactor>4))
```

Similarly, to identify which transcripts had a Bayes factor $>4$ when testing $H_0: \gamma_j = 0$ versus $H_a: \gamma_j = 1$,

```r
names(which(summary.model.fit$gamma.BayesFactor>4))
```

To obtain the $\gamma$ estimates, we used the following code:

```r
coefficients<-coef(fitted.regressvi.random)
coefficients$gamma[which(summary.model.fit$gamma.BayesFactor>4)]
```

To obtain model predictions,

```r
phat<-predict(fitted.regressvi.random)
table(phat$class, colData(finalSet)$Stage)
```

1  2  3  
1 124 28 0  
2 0 20 0  
3 0 13 57  

To determine the adequacy of a more parsimonious model, we then restricted attention to 41 transcripts having gamma.BayesFactor$>4$. The reducedSet object is provided in the ordinalbayes package; however, due to the random nature of the MCMC sampling, the number of transcripts with a Bayes factor for $\gamma$ could differ, so we demonstrate how we derived our object.

```r
reducedSet<-finalSet[which(summary.model.fit$gamma.BayesFactor>4),]
fitted.regressvi.reduced<-ordinalbayes(Stage~1, data=colData(reducedSet),
  x=t(assay(reducedSet)), model="regressvi",
gamma.ind="random", c.gamma=100, d.gamma=1, seed=26)
```
Because we were using `gamma.ind="random"`, we changed the parameter values for the variable inclusion indicator hyperprior to `c.gamma=100, d.gamma=1` to ensure virtually all transcripts would be included in each model. If fitting a model using `gamma.ind="fixed"`, the hyperprior `pi.fixed=0.99` would accomplish the same thing. This smaller model only took 9.1 min to complete.

```r
phat.reduced<-predict(fitted.regressvi.reduced)
table(phat.reduced$class, colData(reducedSet)$Stage)
```

```
1  2  3
1 120 9 1
2  4 45 7
3  0  7 49
```

This more parsimonious model that included 41 transcripts had a misclassification rate of 11.6%. The class-specific misclassification rates [Stage I (3.2%), Stage II (26.2%), Stage III/IV (14.0%)] may indicate that smaller classes are more difficult to predict. To obtain the hypervolume under the ROC manifold, we load the mcca R package [43] and use the `hum` function and estimate the 95% confidence interval using the bootstrap method.

```r
library("mcca")
hum.fit <- hum(colData(reducedSet)$Stage, phat.reduced$predicted,
    method = "prob")
hum.fit
hum.ci <- ests(y = colData(reducedSet)$Stage, d = phat.reduced$predicted,
    acc = "hum", level = 0.95, method = "prob")
hum.ci
```

```
$value
[1] 0.8645291
$se
[1] 0.02955225
$interval
[1] 0.7996765 0.9144449
```

The mcca R package additionally includes a `plot` function that can be used to explore the plot of the three-dimensional ROC surface.

### Appendix C. User-Defined Parameters in OrdinalBayes Function

**Table A1.** ordinallybayes parameters available for all models.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description and Default Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>alpha.var</td>
<td>Variance for $\alpha_k$ in the MCMC chain (default 10)</td>
</tr>
<tr>
<td>coerce.var</td>
<td>Variance associated with any unpenalized predictors in the MCMC chain (default 10)</td>
</tr>
<tr>
<td>adaptSteps</td>
<td>Number of iterations for adaptation (default 5000)</td>
</tr>
<tr>
<td>burnInSteps</td>
<td>Number of iterations of the Markov chain to run (default 5000)</td>
</tr>
<tr>
<td>nChains</td>
<td>Number of parallel chains to run (default 3)</td>
</tr>
<tr>
<td>numSavedSteps</td>
<td>Number of saved steps for each chain (default 9999)</td>
</tr>
<tr>
<td>thinSteps</td>
<td>The thinning interval for monitors (default 3)</td>
</tr>
<tr>
<td>parallel</td>
<td>Run the MCMC on multiple processors (default TRUE)</td>
</tr>
<tr>
<td>model</td>
<td>Specify which penalized ordinal model to fit (default regressvi)</td>
</tr>
<tr>
<td>center</td>
<td>If TRUE (default), center the variables to be penalized in the model</td>
</tr>
<tr>
<td>scale</td>
<td>If TRUE (default), scale the variables to be penalized in the model</td>
</tr>
<tr>
<td>seed</td>
<td>An integer value for the random seed to ensure reproducibility</td>
</tr>
<tr>
<td>quiet</td>
<td>If TRUE, suppress output of JAGS (or rjags) when updating models (default FALSE)</td>
</tr>
</tbody>
</table>
### Table A2. ordinalbayes parameters for each penalized ordinal Bayesian model.

<table>
<thead>
<tr>
<th>Model</th>
<th>Parameters in Ordinalbayes Call to Specify</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>lasso</td>
<td>a, b</td>
<td>The penalty parameter $\lambda \sim \text{Gamma}(a, b)$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(default $a = 0.1$, $b = 0.1$)</td>
</tr>
<tr>
<td>normalss</td>
<td>sigma2.0</td>
<td>The variance for the spike (set to some small positive value, e.g., 0.01)</td>
</tr>
<tr>
<td></td>
<td>sigma2.1</td>
<td>The variance for the slab (set to some large positive value, e.g., 10)</td>
</tr>
<tr>
<td></td>
<td>gamma.ind=&quot;fixed&quot;, pi.fixed</td>
<td>Use a constant prior for $\pi_j$ of pi.fixed (default 0.05)</td>
</tr>
<tr>
<td></td>
<td>gamma.ind=&quot;random&quot;, c.gamma, d.gamma</td>
<td>Use a random prior for $\pi_j \sim \text{Beta}(c, d)$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>for example, $c, d = 0.01, 0.19$</td>
</tr>
<tr>
<td>dess</td>
<td>a, b</td>
<td>The penalty parameter $\lambda \sim \text{Gamma}(a, b)$</td>
</tr>
<tr>
<td></td>
<td>lambda0</td>
<td>The parameter value for the spike, e.g., $\lambda = 20$</td>
</tr>
<tr>
<td></td>
<td>gamma.ind=&quot;fixed&quot;, pi.fixed</td>
<td>Use a constant prior for $\pi_j$ of pi.fixed (default 0.05)</td>
</tr>
<tr>
<td></td>
<td>gamma.ind=&quot;random&quot;, c.gamma, d.gamma</td>
<td>Use a random prior for $\pi_j \sim \text{Beta}(c, d)$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>for example, $c, d = 0.01, 0.19$</td>
</tr>
<tr>
<td>regressvi</td>
<td>a, b</td>
<td>The penalty parameter $\lambda \sim \text{Gamma}(a, b)$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(default $a = 0.1$, $b = 0.1$)</td>
</tr>
<tr>
<td></td>
<td>gamma.ind=&quot;fixed&quot;, pi.fixed</td>
<td>Use a constant prior for $\pi_j$ of pi.fixed (default 0.05)</td>
</tr>
<tr>
<td></td>
<td>gamma.ind=&quot;random&quot;, c.gamma, d.gamma</td>
<td>Use a random prior for $\pi_j \sim \text{Beta}(c, d)$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>for example, $c, d = 0.01, 0.19$</td>
</tr>
</tbody>
</table>

### References


27. Shao, L.; Cui, Y.; Li, H.; Liu, Y.; Zhao, H.; Wang, Y.; Zhang, Y.; Ng, K.M.; Han, W.; Ma, D.; et al. CMTM5 exhibits tumor suppressor activities and is frequently silenced by methylation in carcinoma cell lines. *Clin. Cancer Res.* 2007, 13, 5756–5762. [CrossRef]


32. Belotti, Y.; Lim, E.H.; Lim, C.T. The Role of the Extracellular Matrix and Tumor-Infiltrating Immune Cells in the Prognostication of High-Grade Serous Ovarian Cancer. *Cancers* 2022, 14, 404. [CrossRef] [PubMed]


