juliacon

Econometrics.jl

José Bayoán Santiago Calderón

Social and Decision Analytics Division, Biocomplexity Institute and Initiative, University of Virginia

ABSTRACT

Econometrics.jl is a package for econometrics analysis. It provides a series of most common routines for applied econometrics such as models for continuous, nominal, and ordinal outcomes, longitudinal estimators, variable absorption, and support for convenience functionality such as weights, rank deficient, and robust variance covariance estimators. This study complements the package through a discussion of the motivation, placing the contribution within the Julia ecosystem and econometrics software in general, and provides insights on current gaps and ways the Julia ecosystem can evolve.

Keywords

Julia, Econometrics, Regression Analysis, Generalized Linear Models, Discrete Choice Models, Instrumental Variables Estimation, Panel Data, Variable Absorption, Software Validation

1. Introduction

This study has four core sections. The first surveys what are some commonly used functionality for econometrics analysis. The second provides an overview of the current state of these within the Julia ecosystem including current tools and gaps. The third provides a comprehensive summary of the functionality provided by the new package, Econometrics.jl, and next steps. Lastly, I provide insights on how the Julia ecosystem can evolve from my experience developing the new tool.

Regression analysis is a core tool used to understand the relation among variables. For example, understanding these relationships can provide insights into causal relations or as a basis to develop predictive models. In the realm of statistical inference from regression analysis, one may be interested in common targets such as confidence intervals of the parameters estimates, joint-significance of features, out-of-sample predictive performance, and others. Moreover, most models make certain assumptions which can be tested through statistical tests or model diagnostics which provide confidence in the estimation and results.

Regression analysis is a broad term that encompasses a wide variety of estimators and models. What follows is a brief summary of some of the many components that may fall within the concept. Regression analysis can be used for both observational and experimental settings and it allows great flexibility for a multitude of applications. The main idea is to find estimates for model parameters to optimize some objective such as the likelihood in maximum likelihood estimation (MLE). Other potential objectives include restricted maximum likelihood (REML) or a (Quasi-)Bayesian approach such as maximum a posteriori probability (MAP). One framework is the generalized linear model (GLM) which use a linear predictor that is mapped through a link function to a distribution modeling the response. Continuous responses might use a Normal distribution, count responses a log link with a Negative Binomial distribution, and probability models might use a categorical distribution with links that map to valid probabilities such as the Logit link. In cases such as probability models where the responses are multidimensional, the generalization is known as vector generalized linear models (VGLM). Other generalizations include relaxing the relation between the linear predictor and the outcome to be the sum of smoothing functions through a generalized additive model (GAM) framework or incorporating random effects through a mixed models approach. Some estimators address challenges such as endogeneity, censored responses, and zero-inflated responses through various solutions such as instrumental variables or censored regression model. Others, exploit aspects of the data to overcome challenges or increase efficiency such as random effects in longitudinal data. In relation to the second moment of the estimator, robust variance covariance estimators or boostrapping may be required for inference.

Out of the many potential tools practitioners may require, what are some of the most common? Not every estimator is as widely accessible or commonly used. Some educated guesses may be well justified such as ordinary least squares being more widely used than spatially-weighted regressions. In order to avoid speculation, I defer to a reasonable assumption that the most common estimators are those usually taught in academic programs and available in widely used software [18]. Most programs teach tools to address the most common response types: continuous, count / rates, nominal, ordinal, and duration outcomes. This suggests some common models may include linear models, Poisson/negative binomial, multinomial logistic regression, and ordinal logistic regression with proportional odds assumption. Topics in time series and panel data are usually offered in most programs. Perhaps, the most common topic is short panels (i.e., many units of observations and relatively small number of repeated observations). Common estimators include pooling, first-difference, fixed effects / within estimator, and one-way random effects. The between estimator is usually masked as an intermediate model for estimating the error component in the random effects model. Lastly, the two big challenges taught in most programs are endogeneity and heteroscedasticity. These challenges are usually countered through instrumental variables (e.g., 2SLS) or robust variance-covariance estimators (e.g., heteroscedasticity consistent estimators).

Previous work have surveyed the functionality of 24 alternatives for common econometrics routines [18]. Throughout the history of econometrics software, alternatives have risen and fallen in following. Some high contenders by market share include Stata [20], R [17], MATLAB, Python [16], IBM SPSS Statistics, SAS software, and EViews. These include both commercial and open-source alternatives. Functionality may be provided by the base/standard libraries in the statistical software environment, as a product such as a toolkit or user contributed such as a module/package that is distributed. Some examples of user-contributed functionality include

the reghdfe Stata module and a series of R packages such as MASS [22], lmtest [24], sandwich [23], plm [8], and mlogit [7].
The Julia language [3] is an upcoming language especially well-suited for scientific computing such as econometrics, data science, machine learning, and other related tasks.

2. Common Estimators

2.1 Weighted Least Squares

Weighted least squares solves

$$\beta = \left(X^\top W X\right)^{-1} X^\top W y \tag{1}$$

with information matrix

$$\Psi = \left(X^\top W X\right)^{-1} \tag{2}$$

where X is the full rank version of a model matrix, W a diagonal matrix with positive values (e.g., frequency), and y the response. The common solution method is to factorize X as either its QR decomposition or Cholesky decomposition. Singular value decomposition may also be used, but it is rare as the computational complexity is significantly higher. In the case of the QR decomposition the solution method comes down to, transforming the model matrix and the response by row-wise multiplying them by the square root of the weights. Afterwards, the factorization is used to solve the system of equations using the appropriate method. In the case of a QR decomposition, R is an upper triangular matrix which enables back substitution to obtain the solution efficiently without matrix inversion. However, a Cholesky decomposition would still be required if the information matrix is desired. The solution method with QR decomposition is delineated in equation 3.[1]
The case for the Cholesky decomposition follows closely and without loss of generality other variants could be used such as Bunch-Kaufman decomposition or the upper triangular form ($U^\top U$). The QR decomposition is more numerically stable, but more expensive than Cholesky.[2] Equation 4 delineates the solution method with Cholesky decomposition. Since the information matrix is an important component, Bunch-Kaufman decomposition, a Cholesky variant, is the preferred method used in Econometrics.jl.

$$\begin{aligned} \tilde{X} &= X. * \sqrt{w} \\ \tilde{y} &= y. * \sqrt{w} \\ QR &= \tilde{X} \\ \beta &= R \setminus \left(Q^\top \tilde{y}\right) \end{aligned} \tag{3}$$

$$\begin{aligned} LL &= \left(X^\top W X\right) \\ \beta &= L \setminus \left(X^\top W y\right) \\ \Psi &= \left(L^{-1}\right)^\top L^{-1} \end{aligned} \tag{4}$$

The remaining estimators will assume a Cholesky decomposition as part of the estimation technique. The QR decomposition will be used in models estimated through iterative reweighted least squares (IRLS) as the factorization may be computed once and recycled.

[1] The (\) refers to multiplication of b by the inverse of A on the left.
[2] $\mathcal{O}\left(n^3\right) > \mathcal{O}\left(2mn^2 - \frac{2}{3}2n^3\right)$ where the matrix has m rows and n columns.

2.2 Within Estimator

The within estimator is an application of the Frisch-Waugh-Lovell theorem [10, 13]. The estimator allows to compute the parameters estimates and information matrix for a subset of predictors without having to include the full set of categorical features. For example, one may include individual fixed effects in a large data set that may increase the dimension of the model matrix to several thousand making the problem unfeasible or inefficient. Moreover, some parameters may not be consistently estimated in certain contexts. For example, individual fixed effects are not consistently estimated when there is a fixed length for the panels (i.e., more observations implies more parameters a type of curse of dimensionality).
Consider the following model,

$$y = X\beta + D\theta + e \tag{5}$$

where y is the response, β the parameters of interest, X the features of the parameters of interest, D a high dimensional representation of categorical features as control, θ the parameters on said covariates, and e the error term. In order to obtain the parameter estimates β and the associated information matrix, we can estimate an alternative specification.

$$\tilde{y} = \tilde{X}\beta + e \tag{6}$$

where $\tilde{X}$ and $\tilde{y}$ are obtained by using projections, such as the annihilator matrix (i.e., $I - X\left(X^\top W X\right)^{-1} X^\top$). There are several methods to obtain a suitable alternative regression and these are not unique. Stata's module reghdfe [6] presents several approaches to solving these problem including specialized methods in certain applications. Some implementations include Stata module reghdfe and the FixedEffects.jl package. The two most common approaches are solving for the residuals through a sparse least-squares problems such as with LSMR [9] or using some variant for the method of alternating projections. The residuals approach tends to be more efficient, but degrades certain aspects of the model (e.g., no longer able to obtain the mean response). The method of alternating projections is able to preserve under certain conditions artifacts of the original regression such as obtaining the same estimate for the intercept even though it is not particularly meaningful. Using the original response and the invariant residuals allows to recover the fitted values of the original model.

2.3 Between Estimator

The between estimator estimates

$$\tilde{y} = \tilde{X}\beta + e \tag{7}$$

where the transformed model components are collapsed through some dimension through the mean function. For example, one approach to obtaining the error component for a random effects model is to use model statistics of the between estimator collapsing by panel. The weighted version of the model uses the observation weights to compute the weighted mean values and may use the weight fractions by the collapsing dimension as weights for the weighted least squares regression on the transformed model.

2.4 Random Effects Model

The random effects model relies in estimating the unobserved error components. Random effects requires a particular schema for the data which has a panel component and a temporal component. There are multiple approaches, but the most common one is the Swamy-Arora approach [21]. This estimator uses the mean squared residuals

estimates (i.e., deviance divided by residual degrees of freedom) of the between and within models using the panel dimension as the collapsing / dimension to absorb. The error components are estimated as

$$\begin{aligned} \theta_g &= 1 - \sqrt{\frac{\sigma_e^2}{T_g * \sigma_u^2 + \sigma_e^2}} \\ \sigma_e^2 &= W \\ \sigma_u^2 &= \max\left\{0, B - \sigma_e^2 * \bar{T}\right\} \end{aligned} \tag{8}$$

where W is the mean squared residuals of the within model, B the mean squared residuals of the between model, and T_g is the length of the panel g, and $\bar{T}$ is the harmonic mean of the panel lengths.
The model terms are then transformed by partial demeaning

$$\begin{aligned} \tilde{y}_{it} &= y_{it} - \theta_g * \bar{y}_{.t} \\ \tilde{X}_{it} &= X_{it} - \theta_g * \bar{X}_{.t} \end{aligned} \tag{9}$$

and these are used in the standard regression setting.

2.5 First-Difference

The first-difference estimator is a special case that use time / panel context for feature designs. The most common transformations include contrasts such as treatment coding (dummy coding), sum coding (effects coding) or Helmert coding which apply to categorical variables. Other common feature engineering techniques include log-transform and polynomial terms. However, certain transformations require a context such as a time dimension. Some examples include shift operations (lag, lead) and differentiating (e.g., first-difference). These operations may optionally require a group context such that the operations are performed group wise. Time-context operations have important concepts such as frequency and gaps. The frequency describes the difference between periods/observations and gaps describe observations that are skipped and should be understood as missing.

2.6 Instrumental Variables

Every estimator thus far can be generalized to include endogenous covariates through instrumental variables. The most common method is through two stages least squares (2SLS). The idea is to first apply all the relevant transformations to the model terms and apply the 2SLS standard procedure. In the case of the random effects model, the within and between models are estimated using 2SLS to obtain the error component estimates. After applying the random effects transformation to each model term the 2SLS process is employed in the final regression model [1].
The standard 2SLS estimator uses,

$$\begin{aligned} \hat{z} &= [XZ]\left[\left([XZ]^\top W[XZ]\right)^{-1}[XZ]^\top W z\right] \\ \hat{\beta} &= \left([X\hat{z}]^\top W[X\hat{z}]\right)^{-1}[X\hat{z}]^\top W y \\ \Psi &= \left([X\hat{z}]^\top W[X\hat{z}]\right)^{-1} \\ \hat{y} &= [Xz]\hat{\beta} \end{aligned} \tag{10}$$

for each model where z is endogenous variables and Z the additional instruments.

2.7 Nominal Response Model

Multinomial logistic regression is a probability model for estimating probabilities across multiple categories. It is a vector generalized linear model with softmax link function and the categorical distribution. It is estimated through iterative re-weighted least squares (IRLS) methods such as the QR Newton variant [15]. The data schema for discrete choice models include the response (observed behavior), unit of observation covariates, and outcomes-specific covariates. The initial implementation allows for the base case of no-outcome specific features.

2.8 Ordinal Response Model

Ordinal logistic regression is a probability model for estimating probabilities across multiple ordered categories. Similarly to its nominal counterpart, it has a pool of alternatives, and observed outcome, unit of observation covariates, and outcome-specific covariates. A common assumption is the proportional odds assumption which may be relaxed in other models.
The log-likelihood function has the same form as the general form for computing the cost associated with a categorical distribution and predicted probability for realization. More specific,

$$\ell\ell = \sum_{i=1}^{m}\sum_{k}^{K} \mathbb{1}\left(y_i = k\right)\ln\left[F\left(\alpha_{k+1} - \eta\right) - F\left(\alpha_k - \eta\right)\right] \tag{11}$$

where F is the cumulative distribution function of the logistic distribution with zero location and unit scale, η is the linear projection, and α_k is the threshold for lower threshold [14]. The log-likelihood function and the gradient are passed to the Optim.jl framework [11] using ForwardDiff.jl [19] forward mode automatic differentiation (AD) for the Newtonian solver.

2.9 Count/Rate Model

Count/rate models are generalized linear models and follow a similar description as nominal models. The most common distribution choices are Poisson and Negative Binomial with the log link function. Negative Binomial is a generalization of the Poisson model, which adds an extra parameter for modeling the second moment (i.e., relaxes the mean equal variance assumption in the Poisson model). For the Negative Binomial to be a distribution in the exponential family it needs a restriction parameter which may be optimized through maximum likelihood estimation. An offset may be included to handle rates, a generalization of counts, that account for differences in exposures. Other generalizations include additive or multiplicative errors relations.

2.10 Duration Models

Duration models deal with responses of the type time until an event. One such model is the Cox proportional hazards model which relies on the proportional hazards assumption. Various models of these kind may be re-specified in a generalized linear model framework relating to the previous descriptions.

3. Technical Challenges

One technical challenge that is prevalent through every model is the issue of rank deficient terms. Rank deficient systems of linear equations are not identifiable. One approach is to error out and let the user explore and find a subset of features such that the no multi-collinearity assumption holds. The second approach is to automatically promote the system to a full rank version by excluding linearly dependent features. How much collinearity is too much is not an exact science. Some potential criteria include using the absolute values of the diagonal in the triangular matrix of the factorization (e.g., L in $LL^\top$, R in QR, D in $LDL^\top$, Σ in $U\Sigma V^\top$).

These values are then compared against a chosen tolerance and the column of the term is deemed linearly independent if the values are greater than the tolerance.
Note that Cholesky, QR, and Bunch-Kaufman decomposition allow to identify which columns are independent while singular values only allow to determine the rank. It may be arbitrary to choose among linearly dependent features. An additional level of complexity in probability models is the issue of linearly separability for probability models [12].

4. Julia Ecosystem

The usual pipeline for regression analysis involves (1) accessing data (I/O), (2) obtaining a tabular data representation, (3) data wrangling, and (4) employing regression analysis tools. The Julia ecosystem follows this canonical pipeline. The following sections provides an overview of the pipeline available in Julia.

4.1 Data to Modeling

StatsBase.jl builds on top of Statistics.jl (standard library) to provide additional statistical functionality. One which includes the abstraction for Statistical Models (and Regression models which inherits from the former). It provides a simple and powerful API for the whole Julia ecosystem to use. It allows packages to implement the API and easily support a common functionality users can expect and interact with in a familiar manner. For example, *coef* will extract the parameter estimates from any object that implements the API. The full API include model statistics such as: coefficient of determination (or adjusted), information criteria statistics such as AIC/BIC (and corrected), statistics about the model fitness such as deviance, log-likelihood, and usual queries such as point estimates, variance covariance estimates, standard errors, confidence intervals, degrees of freedom (or residual degrees of freedom), etc. Lastly, several accessors are available for fitted values, response, model matrix, information matrix, leverage values, error components, etc. Lastly, it also provides an abstraction for weights including frequency weights and analytical weights.
Tables.jl provide an interface for tabular data. This API allows users to choose from various solutions the tabular data implementation of their choosing without having to worry that their choice will limit potential functionality. Many tabular implementations such as DataFrames.jl provide robust functionality to many routines such as handling categorical features, dates/time, missing values, reshaping data, split/apply operations, and others. Users need not to worry about any I/O issues as a rich array of options exist for importing and exporting across different file formats such as delimiter-separated values, JSON, Feather, HDF5, MATLAB, Stata, SPSS, SAS, and R.
StatsModels.jl is a package that provides the means to go from data to model terms. It provides the formulae language (e.g., similar to R's formulae syntax). A model is then build using a formula, data, and additional model specific arguments. The process can be summarized as (1) collecting the information in the formula, (2) parsing its meaning by applying a schema based on the data, user-specified contrasts or other arguments, and (3) generating the model terms such as a response, model matrices, etc. Lastly, a package fits said model and implements the API.

4.2 Regression Analysis

The regression analysis ecosystem in Julia has GLM.jl as its flagship. GLM.jl provides the typical functionality for fitting generalized linear models through Fisher scoring. This includes linear models, Poisson/Negative Binomial, Logit/Probit, and other non-canonical link models. CovarianceMatrices.jl provides various variance covariance estimators for GLM.jl models à la R's sandwich package. LinearMixedModels.jl [2] extends GLM.jl for mixed-effects models. FixedEffectModels.jl provides fast estimation of linear models with instrumental variables and high dimensional categorical variables à la reghdfe. Survival.jl provides a series of estimators for duration models. Two major gaps in the ecosystem include estimating nominal and ordinal response models (i.e., discrete choice) with more than two alternatives and support for longitudinal estimators.

5. Econometrics.jl

Econometrics.jl is a package for performing several common econometrics routines in the Julia language. It aims to provide the following functionality for two major gaps in the ecosystem, longitudinal estimators and discrete choice models. Developing the package has resulted in many contributions in the current ecosystem. However, the development of this package serves multiple purposes beyond the immediate effect. As the statistics ecosystem evolves and matures, Econometrics.jl aims to serve as inspiration and an alternative to design decisions, standards, and option for users and developers.

5.1 Fitting Models

This section will showcase some examples of using the package for various estimators. For each estimator a brief description of the data, model, syntax, and output will be provided. Results will be provided for Econometrics.jl and some alternatives such as R or Stata.
For linear models, the examples use a crime dataset [4]. The data set is a balanced longitudinal data set with 90 counties in North Carolina from 1981 to 1987. The outcome variable is the crime rate and the explanatory variables include the probability of conviction, average sentence, and probability of prison sentence. Estimating the pooling estimator or the between estimator can be accomplished as in figure 1 and 2 on the next page. Table 1 on page 8 shows the estimated 95% confidence intervals using Econometrics.jl, Stata, and R's plm package.

```
model = fit(EconometricModel,
            @formula(CRMRTE ~ PrbConv + AvgSen + PrbPris),
            data)
```

```
Continuous Response Model
Number of observations: 630
Null Loglikelihood: 1633.30
Loglikelihood: 1643.30
R-squared: 0.0312
LR Test: 19.99 ~ x²(3) ⟹  Pr > x² = 0.0002
Formula: CRMRTE ~ 1 + PrbConv + AvgSen + PrbPris
Variance Covariance Estimator: OIM
```

	PE	SE	t-value	Pr > |t|	2.50%	97.50%
(Intercept)	0.0186413	0.00431066	4.32447	<1e-4	0.0101762	0.0271064
PrbConv	-0.00116473	0.000422062	-2.75961	0.0060	-0.00199356	-0.0003359
AvgSen	0.000236185	0.000268216	0.88058	0.3789	-0.000290526	0.000762897
PrbPris	0.0273394	0.00817642	3.34369	0.0009	0.0112829	0.0433959

Fig. 1: Estimation of the pooling estimator

The fixed effects model or within estimator can be estimated as in figure 3 on the next page which a two-ways fixed effects model (i.e., fixed effects for panel and time dimensions). Table 2 on page 8 shows the estimated 95% confidence intervals using Econometrics.jl, Stata, and R's plm package.

```
model = fit(BetweenEstimator,
            @formula(CRMRTE ~ PrbConv + AvgSen + PrbPris),
            data,
            panel = :County)
```

```
Between Estimator
County with 630 groups
Balanced groups with size 7
Number of observations: 90
Null Loglikelihood: 239.56
Loglikelihood: 244.40
R-squared: 0.1029
Wald: 3.29 ~ F(3, 86) ⟹ Pr > F = 0.0246
Formula: CRMRTE ~ 1 + PrbConv + AvgSen + PrbPris
Variance Covariance Estimator: OIM
```

	PE	SE	t-value	Pr > |t|	2.50%	97.50%
(Intercept)	-0.00464339	0.0186571	-0.24888	0.8040	-0.0417325	0.0324457
PrbConv	-0.00354113	0.0018904	-1.87322	0.0644	-0.00729911	0.00021685
AvgSen	0.000848049	0.00116477	0.728086	0.4685	-0.00146743	0.00316353
PrbPris	0.0730299	0.0332057	2.19932	0.0305	0.0070191	0.139041

Fig. 2: Estimation of the between panel estimator

```
model = fit(EconometricModel,
            @formula(CRMRTE ~ PrbConv + AvgSen + PrbPris + absorb(County + Year)),
            data)
```

```
Continuous Response Model
Number of observations: 630
Null Loglikelihood: 1633.30
Loglikelihood: 2290.25
R-squared: 0.8775
Wald: 0.23 ~ F(3, 531) ⟹ Pr > F = 0.8767
Formula: CRMRTE ~ 1 + PrbConv + AvgSen + PrbPris + absorb(County + Year)
Variance Covariance Estimator: OIM
```

	PE	SE	t-value	Pr > |t|	2.50%	97.50%
(Intercept)	0.0319651	0.00206378	15.4886	<1e-44	0.027911	0.0360193
PrbConv	7.90362e-5	0.000196089	0.403062	0.6871	-0.00030617	0.000464242
AvgSen	-9.4788e-5	0.000134924	-0.702531	0.4827	-0.000359838	0.000170262
PrbPris	0.000979533	0.0040432	0.242267	0.8087	-0.00696309	0.00892216

Fig. 3: Estimation of the within estimator with multiple fixed effects

The random effects model can be estimated as in figure 4 which shows estimating a random effects model with instrumental variables. Table 3 on page 8 shows the estimated 95% confidence intervals using Econometrics.jl, Stata's reghdfe module, and R's plm package.

```
model = fit(RandomEffectsEstimator,
            @formula(CRMRTE ~ PrbConv + (AvgSen ~ PrbPris)),
            data,
            panel = :County,
            time = :Year)
```

```
One-way Random Effect Model
Longitudinal dataset: County, Year
Balanced dataset with 90 panels of length 7
individual error component: 0.0413
idiosyncratic error component: 0.0074
ρ: 0.9691
Number of observations: 630
Null Loglikelihood: 2268.01
Loglikelihood: 2248.34
R-squared: NaN
Wald: 0.03 ~ F(2, 626) ⟹ Pr > F = 0.9671
Formula: CRMRTE ~ PrbConv + (AvgSen ~ PrbPris)
Variance Covariance Estimator: OIM
```

	PE	SE	t-value	Pr > |t|	2.50%	97.50%
(Intercept)	0.037747	0.0241684	1.56183	0.1188	-0.00971405	0.085208
PrbConv	1.39581e-5	0.000200244	0.0697054	0.9445	-0.000379273	0.000407189
AvgSen	-0.000688924	0.00266514	-0.258494	0.7961	-0.00592262	0.00454478

Fig. 4: Estimation of the random effects model

The sysdsn1 Stata example health insurance data set is used to illustrate the multinominal logistic regression when the response is nominal as seen in figure 5. A comparison with the estimates for the 95% confidence intervals between Econometrics.jl and Stata is shown in table 4 on page 9.

```
model = fit(EconometricModel,
            @formula(insure ~ age + male + nonwhite + site),
            data,
            contrasts = Dict(:insure => DummyCoding(base = "Uninsure")))
```

```
Probability Model for Nominal Response
Categories: Uninsure, Indemnity, Prepaid
Number of observations: 615
Null Loglikelihood: -555.85
Loglikelihood: -534.36
R-squared: 0.0387
LR Test: 42.99 ~ x²(10) ⟹ Pr > x² = 0.0000
Formula: insure ~ 1 + age + male + nonwhite + site
```

	PE	SE	t-value	Pr > |t|	2.50%	97.50%
insure: Indemnity ~ (Intercept)	1.28694	0.59232	2.17271	0.0302	0.123689	2.4502
insure: Indemnity ~ age	0.00779612	0.0114418	0.681372	0.4959	-0.0146743	0.0302666
insure: Indemnity ~ male	-0.451848	0.367486	-1.22957	0.2193	-1.17355	0.269855
insure: Indemnity ~ nonwhite	-0.217059	0.425636	-0.509965	0.6103	-1.05296	0.618843
insure: Indemnity ~ site: 2	1.21152	0.470506	2.57493	0.0103	0.287497	2.13554
insure: Indemnity ~ site: 3	0.207813	0.366293	0.56734	0.5707	-0.511547	0.927172
insure: Prepaid ~ (Intercept)	1.55666	0.596327	2.61041	0.0093	0.385533	2.72778
insure: Prepaid ~ age	-0.00394887	0.0115993	-0.340439	0.7336	-0.0267287	0.018831
insure: Prepaid ~ male	0.109846	0.365187	0.300793	0.7637	-0.607343	0.827035
insure: Prepaid ~ nonwhite	0.757718	0.419575	1.80592	0.0714	-0.0662835	1.58172
insure: Prepaid ~ site: 2	1.32456	0.469789	2.81947	0.0050	0.401941	2.24717
insure: Prepaid ~ site: 3	-0.380175	0.372819	-1.01973	0.3083	-1.11235	0.352001

Fig. 5: Estimation of the multinomial logistic regression

The fullauto Stata example automobile models data set is used to illustrate the proportional ordinal logistic regression when the response is ordinal as seen in figure 6. A comparison with the estimates for the 95% confidence intervals between Econometrics.jl, Stata, and R's MASS is shown in table 5 on page 9.

```
model = fit(EconometricModel,
            @formula(RecParks ~ Age + Sex + Schooling),
            data)
```

```
Probability Model for Ordinal Response
Categories: 1 < 2 < 3 < 4 < 5
Number of observations: 1827
Null Loglikelihood: -2677.60
Loglikelihood: -2657.40
R-squared: 0.0075
LR Test: 40.42 ~ x²(3) ⟹ Pr > x² = 0.0000
Formula: RecParks ~ Age + Sex + Schooling
```

	PE	SE	t-value	Pr > |t|	2.50%	97.50%
Age	0.00943647	0.00254031	3.71469	0.0002	0.00445424	0.0144187
Sex: male	-0.0151659	0.0846365	-0.179188	0.8578	-0.181161	0.150829
Schooling	-0.103902	0.0248742	-4.17711	<1e-4	-0.152687	-0.0551174
(Intercept): 1 | 2	-2.92405	0.191927	-15.2352	<1e-48	-3.30047	-2.54763
(Intercept): 2 | 3	-1.54922	0.171444	-9.03632	<1e-18	-1.88547	-1.21297
(Intercept): 3 | 4	-0.298938	0.166904	-1.79108	0.0734	-0.626281	0.0284051
(Intercept): 4 | 5	0.669835	0.167622	3.9961	<1e-4	0.341083	0.998587

Fig. 6: Estimation of the proportional odds logistic regression

5.2 Design Decisions

Statistical software developers play a very powerful role in shaping culture and norms. For example, whether to default to maximum likelihood estimation (MLE) or restricted maximum likelihood estimation (REML), can shape not only choices by practitioners, but by stakeholders, regulatory agencies, and expected components for reports. These changes may be good or bad depending on the case. For example, advances in econometrics are rarely widely adopted without buy-in from software developers. The following discussions will survey some of the decisions relevant to Econometrics.jl.

Should software be dummy-proof? Many times software developers have to choose between exposing users to make mistakes on their own volition or put safeguard against potential misuses by restricting behavior that may be correct under rare scenarios. For example, a basic tool might allow users to mix and match link and distributions in a GLM settings even if the combinations are nonsensical. A safer approach would be to restrict combinations to those

"safe" combinations such as distributions with canonical links. The trade-off occurs when users may encounter a specification that while uncommon it may be the correct one for that particular model. Currently, Econometrics.jl takes a conservative approach that provides "dummy-proof" experience as well as ease of use. For example, rather than requiring users to specify the model, the estimator is inferred based on the type of the response and information provided as long as it is unambiguous. Other examples of this approach include auto-promoting model statistics such as the coefficient of determination to a pseudo-version for non-linear models (a generalization of the linear case), but providing a not a number (NaN) value for instrumental variable models or those that do not include an intercept. Similarly, the software will promote terms to full rank as required. The formula term and coefficient table provide all the information needed to figure which if any feature was suppressed.

Software analysis should always include diagnostics and tools to make it easier for dissemination. Many tests and diagnostics are applicable to a wide set of implementations. The best manner to make these available and for these to "play nicely" with one another is to have an effective API. Sadly, the Julia ecosystem has yet to experience wide adoption of this pattern. For example, packages might need to access components for computing a test or providing some estimates such as variance covariance estimates. The test might include components such as the residual degrees of freedom, the information matrix, residuals, and score. Currently, some packages rely on accessing internal components specific to those implementations rather than an implementation agnostic interface. A common public API would also allow to catch context specific instances of improper behavior.

Various decisions are software specific with asymptotic justification, but significant finite-sample consequences. Software may differ on whether to report statistics using finite-sample statistics (t-distribution, F-distribution) or asymptotic equivalent counterparts (Normal, Chi squared). These tend to have negligible effect in most applications, but other decisions such as degrees of freedom may have larger consequences. For example, software may differs on how it computes the degrees of freedom for instrumental variables or absorbed variables depending on the context (e.g., main regression or auxiliary regression for estimating error components). Refinements and robustness checks can also contribute to a better analysis such as verifying gaps for time variant operations such as in first-difference or purging singletons and other degree of freedom adjustments [5].

5.3 Best Practices

Econometrics.jl adopts the best practices standards for open-source statistical software. These include adhering to semantic versioning (semver) for descriptive versioning, continuous integration for development, software validation through a comprehensive code coverage and test suite, and lastly online hosted documentation for the public API.

6. Conclusion

Econometrics.jl is a new addition to the Julia ecosystem that brings highly demanded functionality concerning longitudinal estimators and discrete choice models. This study serves as a complement to the software documentation providing context to the development, design considerations, and roadmap of the project. A philosophical motivation for the project is to make econometrics accessible to practitioners not only through functionality, but transparency in the code readability, replicability, and correctness. For example, transparent well-written code is easier to maintain, inspect / audit, and can be useful for learning and teaching.

Community contributions and feedback are highly encouraged in order to best continue developing the project. Some features I would like the project to support in the future include more advanced estimators such as: (1) choice-variant categorical response models, (2) count/rate models such as zero-inflated, and (3) censored/truncated response models. Econometrics.jl is ISC licensed and available at the GitHub repository.

7. Acknowledgement

This research benefited from support from the National Science Foundation and the AEA Mentoring Program: NSF Awards 1357478 & 1730651.

8. References

[1] Pietro Balestra and Jayalakshmi Varadharajan-Krishnakumar. Full information estimations of a system of simultaneous equations with error component structure. *Econometric Theory*, 3(2):223–246, 1987.

[2] Douglas Bates, José Bayoán Santiago Calderón, Dave Kleinschmidt, Tony Kelman, Simon Babayan, Patrick Kofod Mogensen, Morten Piibeleht, Milan Bouchet-Valat, Michael Hatherly, Elliot Saba, Antoine Baldassari, and Andreas Noack. Mixedmodels.jl, 2019.

[3] Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B. Shah. Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1):65–98, 2017.

[4] Christopher Cornwell and William N. Trumbull. Estimating the economic model of crime with panel data. *The Review of Economics and Statistics*, 76(2):360–366, 1994.

[5] Sergio Correia. Singletons, cluster-robust standard errors and fixed effects: A bad mix, 2015.

[6] Sergio Correia. Linear models with high-dimensional fixed effects: An efficient and feasible estimator. Technical report, Duke University, 2017. Working Paper.

[7] Yves Croissant. *mlogit: Multinomial Logit Models*, 2018. R package version 0.3-0.

[8] Yves Croissant and Giovanni Millo. Panel data econometrics in *R* : The plm package. *Journal of Statistical Software*, 27(2), 2008.

[9] David Chin-Lung Fong and Michael Saunders. LSMR: An iterative algorithm for sparse least-squares problems. *SIAM Journal on Scientific Computing*, 33(5):2950–2971, 2011.

[10] Ragnar Frisch and Frederick V. Waugh. Partial time regressions as compared with individual trends. *Econometrica*, 1(4):387, 1933.

[11] Patrick K Mogensen and Asbjørn N Riseth. Optim: A mathematical optimization package for julia. *Journal of Open Source Software*, 3(24):615, 2018.

[12] Kjell Konis. *Linear Programming Algorithms for Detecting Separated Data in Binary Logistic Regression Models*. PhD thesis, Worcester College, University of Oxford, 2007.

[13] Michael C. Lovell. A simple proof of the FWL theorem. *The Journal of Economic Education*, 39(1):88–91, 2008.

[14] Richard D. McKelvey and William Zavoina. A statistical model for the analysis of ordinal level dependent variables. *The Journal of Mathematical Sociology*, 4(1):103–120, 1975.

[15] Dianne P. O'Leary. Robust regression computation using iteratively reweighted least squares. *SIAM Journal on Matrix Analysis and Applications*, 11(3):466–480, 1990.

[16] Python Software Foundation. *Python Software*, 2018.

[17] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2018.

[18] Charles G. Renfro. *The Practice of Econometric Theory*, volume 44 of *Advanced Studies in Theoretical and Applied Econometrics*. Springer Berlin Heidelberg, 2009.

[19] Jarrett Revels, Miles Lubin, and Theodore Papamarkou. Forward-mode automatic differentiation in julia. *CoRR*, abs/1607.07892, 2016.

[20] StataCorp. *Stata Statistical Software: Release 15*, 2017.

[21] P. A. V. B. Swamy and S. S. Arora. The exact finite sample properties of the estimators of coefficients in the error components regression models. *Econometrica*, 40(2):261, 1972.

[22] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*. Springer, New York, fourth edition, 2002. ISBN 0-387-95457-0.

[23] Achim Zeileis. Econometric computing with HC and HAC covariance matrix estimators. *Journal of Statistical Software*, 11(10):1–17, 2004.

[24] Achim Zeileis and Torsten Hothorn. Diagnostic checking in regression relationships. *R News*, 2(3):7–10, 2002.

APPENDIX

Table 1. : Pooling and Between Estimators

Model	Parameter	Econometrics.jl		Stata		R (plm)	
Pooling	Intercept	0.0102	0.0271	0.0102	0.0271	0.0102	0.0271
	PrbConv	-0.0020	-0.0003	-0.0020	-0.0003	-0.0020	-0.0003
	AvgSen	-0.0003	0.0008	-0.0003	0.0008	-0.0003	0.0008
	PrbPris	0.0113	0.0434	0.0113	0.0434	0.0113	0.0434
Between	Intercept	-0.0417	0.0324	-0.0417	0.0324	-0.0412	0.0319
	PrbConv	-0.0073	0.0002	-0.0073	0.0002	-0.0072	0.0002
	AvgSen	-0.0015	0.0032	-0.0015	0.0032	-0.0014	0.0031
	PrbPris	0.0070	0.1390	0.0070	0.1390	0.0079	0.1381

Table 2. : Absorbing Panel or Panel and Temporal Indicators

Model	Parameter	Econometrics.jl		Stata (reghdfe)		R (plm)	
Within PID	Intercept	0.0274	0.0355	0.0274	0.0355		
	PrbConv	-0.0004	0.0004	-0.0004	0.0004	-0.0004	0.0004
	AvgSen	-0.0002	0.0003	-0.0002	0.0003	-0.0002	0.0003
	PrbPris	-0.0093	0.0066	-0.0093	0.0066	-0.0093	0.0066
Within PTID	Intercept	0.0279	0.0360	0.0279	0.0360		
	PrbConv	-0.0003	0.0005	-0.0003	0.0005	-0.0003	0.0005
	AvgSen	-0.0004	0.0002	-0.0004	0.0002	-0.0004	0.0002
	PrbPris	-0.0070	0.0089	-0.0070	0.0089	-0.0069	0.0089

Table 3. : Random Effects and Instrumental Variables

Model	Parameter	Econometrics.jl		Stata		R (plm)	
Random	Intercept	0.0257	0.0362	0.0257	0.0362		
	PrbConv	-0.0004	0.0004	-0.0004	0.0003	-0.0004	0.0004
	AvgSen	-0.0002	0.0003	-0.0002	0.0003	-0.0002	0.0003
	PrbPris	-0.0081	0.0078	-0.0080	0.0078	-0.0093	0.0066
IV Random	Intercept	-0.0097	0.0852	-0.0096	0.0851	-0.0096	0.0851
	PrbConv	-0.0004	0.0004	-0.0004	0.0004	-0.0004	0.0004
	AvgSen	-0.0059	0.0045	-0.0059	0.0045	-0.0059	0.0045

Table 4. : Multinomial Logistic Regression

Response	Parameter	Econometrics.jl		Stata	
Indemnity	(Intercept)	-0.3753	0.9148	-0.3740	0.9134
	Age	-0.0239	0.0004	-0.0239	0.0004
	Gender: Male	0.1635	0.9599	0.1643	0.9591
	Nonwhite	0.5107	1.4389	0.5116	1.4380
	Site: 2	-0.2998	0.5258	-0.2989	0.5250
	Site: 3	-1.0356	-0.1404	-1.0347	-0.1412
Prepaid	(Intercept)	-2.4502	-0.1237	-2.4479	-0.1260
	Age	-0.0303	0.0147	-0.0302	0.0146
	Gender: Male	-0.2698	1.1736	-0.2684	1.1721
	Nonwhite	-0.6188	1.0530	-0.6172	1.0513
	Site: 2	-2.1356	-0.2875	-2.1338	-0.2894
	Site: 3	-0.9272	0.5115	-0.9257	0.5101

Table 5. : Parallel Ordinal Logistic Regression

Parameter	Econometrics.jl		Stata		R's MASS	
Foreign	1.3168	4.4768	1.3472	4.4464	1.4111	4.5293
Length	0.0374	0.1282	0.0383	0.1274	0.0395	0.1292
MPG	0.0900	0.3716	0.0927	0.3689	0.0986	0.3781
(Intercept): Poor | Fair	6.8343	29.0206	7.0473	28.8076		
(Intercept): Fair | Average	8.6814	31.0487	8.8962	30.8340		
(Intercept): Average | Good	10.6949	33.5117	10.9140	33.2926		
(Intercept): Good | Excellent	12.9204	36.4639	13.1465	36.2378		