SOFTWARE TOOL ARTICLE

ProSGPV: an R package for variable selection with second-generation p-values [version 1; peer review: 1 approved with reservations]

Yi Zuo , Thomas G. Stewart, Jeffrey D. Blume

Biostatistics, Vanderbilt University, Nashville, USA

V1 First published: 18 Jan 2022, 11:58
<https://doi.org/10.12688/f1000research.74401.1>
Latest published: 18 Jan 2022, 11:58
<https://doi.org/10.12688/f1000research.74401.1>

Abstract

We introduce the ProSGPV R package, which implements a variable selection algorithm based on second-generation p-values (SGPV) instead of traditional p-values. Most variable selection algorithms shrink point estimates to arrive at a sparse solution. In contrast, the ProSGPV algorithm accounts for the estimation uncertainty – via confidence intervals – in the selection process. This additional information leads to better inference and prediction performance in finite sample sizes. ProSGPV maintains good performance even in the high dimensional case where $p > n$, or when explanatory variables are highly correlated. Moreover, ProSGPV is a unifying algorithm that works with continuous, binary, count, and time-to-event outcomes. No cross-validation or iterative processes are needed and thus ProSGPV is very fast to compute. Visualization tools are available in this package for assessing the variable selection process. Here we present simulation studies and a real-world example to demonstrate ProSGPV's inference and prediction performance in relation to the current standards in variable selection procedures.

Keywords

R, variable selection, statistical inference

Open Peer Review

Approval Status ?

	1
version 1	?
18 Jan 2022	view
1. Georg Heinze , Medical University of Vienna, Vienna, Austria	

Any reports and responses or comments on the article can be found at the end of the article.

This article is included in the RPackage gateway.

Corresponding author: Yi Zuo (yi.zuo@vanderbilt.edu)

Author roles: **Zuo Y:** Conceptualization, Data Curation, Formal Analysis, Methodology, Software, Visualization, Writing – Original Draft Preparation; **Stewart TG:** Methodology, Supervision, Validation; **Blume JD:** Conceptualization, Methodology, Project Administration, Supervision, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: The author(s) declared that no grants were involved in supporting this work.

Copyright: © 2022 Zuo Y *et al.* This is an open access article distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

How to cite this article: Zuo Y, Stewart TG and Blume JD. **ProSGPV: an R package for variable selection with second-generation p-values [version 1; peer review: 1 approved with reservations]** F1000Research 2022, 11:58
<https://doi.org/10.12688/f1000research.74401.1>

First published: 18 Jan 2022, 11:58 <https://doi.org/10.12688/f1000research.74401.1>

Introduction

As the sheer volume of data grows at an astronomical rate, variable selection plays an increasingly crucial role in research. This is particularly true in the high dimensional setting where $p > n$ and classical statistical methods exploiting the full feature space no longer work. An ideal variable selection procedure would recover the underlying true support with high probability, yield parameter estimation with low bias, and achieve good prediction performance. While it is hard for a statistical procedure to strike a balance between inference and prediction tasks,^{1,2} the ProSGPV algorithm is remarkably able in this sense.^{3,4}

One natural approach to variable selection is best subset selection (BSS). BSS chooses k out of p total variables for each $k \in \{1, 2, \dots, p\}$ that maximize a chosen loss function. It can be thought of as an ℓ_0 -penalized regression.⁵ showed that the BSS problem is nonconvex and NP-hard. With recent advancements,^{6–8} solving the BSS routine with thousands of features is no longer infeasible. Particularly, an efficient R package called **BeSS**⁸ is scalable to identify the best sub-model in seconds or a few minutes when p is around 10,000. ℓ_1 -penalized likelihood procedures are also used for variable selection. Lasso, for example, produces models with a strong predictive ability.⁹ However, lasso is not always variable selection consistent.^{1,10} To address the inconsistency issue, adaptive lasso was proposed, which introduces weights in the ℓ_1 penalty.¹¹ Despite oracle variable selection properties of the adaptive lasso, it is often hard in practice to find a tuple of tuning parameters that achieve the properties. Both lasso and adaptive lasso can be implemented in the **glmnet** package.^{12,13} Smoothly clipped absolute deviation (SCAD)¹⁴ and minimax concave penalty with penalized linear unbiased selection (MC+)¹⁵ were proposed to bridge the gap between the ℓ_0 and ℓ_1 penalties. The two algorithms are largely distinguished by their piecewise linear thresholding functions. Sure independence screening (SIS),^{16,17} implemented in the **SIS** package,¹⁸ ranks the maximum marginal likelihood estimates and can greatly reduce the dimensionality of feature space by keeping top variables in the ranking, even when $p \gg n$. Iterative SIS (ISIS) can improve its performance in finite sample sizes. Note that all aforementioned algorithms shrink point estimates to derive a sparse solution and there is room for improvement of inference and prediction properties in finite sample sizes.

Many R packages have been developed to address certain data types. For example, **clogitL1**¹⁹ performs variable selection with lasso and elastic net penalties in conditional logistic regression; **pogit**²⁰ performs Bayesian variable selection with spike and slab priors in Poisson and Logistic regressions; **penPHcure**²¹ performs variable selection in Cox proportional hazards cure model with time-varying covariates. The ideal R package would have superior or comparable performance as the current algorithms, and work with each of continuous, binary, count, and time-to-event outcomes.

Recently, we^{3,4} developed penalized regression with second generation p-values (ProSGPV) for variable selection in both low-dimensional ($n > p$) and high-dimensional ($p > n$) settings. Unlike traditional algorithms, ProSGPV incorporates estimation uncertainty via confidence intervals in the variable selection process. This addition often leads to better support recovery, parameter estimation, and prediction performance. This paper describes an R package named **ProSGPV**, which implements the ProSGPV algorithm.²⁹ Here, we extend the algorithm to work with data from logistic regression, Poisson regression, and Cox proportional hazards regression. We also provide visualization tools for variable selection process, which was not discussed in the original paper.^{3,4} Simulation studies below compare the inference and prediction performance of **ProSGPV** against that of **glmnet**, **BeSS**, and **ISIS**, in scenarios not discussed in.^{3,4} A real-world example compares the sparsity of solutions and prediction accuracy of all algorithms.

Methods

What is a second-generation p-value?

Second-generation p-values (SGPVs) were proposed to address some of the well-known flaws in traditional p-values.^{22,23} The basic idea is to replace the point null hypothesis with a pre-specified interval null. The SGPV is denoted by p_δ , where δ represents the half-width of the interval null. The interval null represents the set of effect sizes that are scientifically indistinguishable from the point null hypothesis, due to limited precision or practicality.

SGPVs are essentially the fraction of data-supported hypotheses that are also null hypotheses. See Figure 1 for an illustration of how SGPVs work. Their formal definition is as follows: let θ be the parameter of interest, and let $I = [\theta_l, \theta_u]$ be an interval estimate of θ whose length is given by $|I| = \theta_u - \theta_l$. Here, I can be a confidence interval (we will use 95% CIs in this paper) or likelihood support interval or a credible interval. The coverage probability of the interval estimate I will largely drive the frequency properties of the SGPV upon which it is based.

Denote the length of the interval null by $|H_0|$. Then the SGPV, p_δ , is defined as

$$p_\delta = \frac{|I \cap H_0|}{|I|} \times \max \left\{ \frac{|I|}{2|H_0|}, 1 \right\} \quad (1)$$

Figure 1. An illustration of how SGPVs work.

where $I \cap H_0$ is the intersection of two intervals.^{22,23} Notice how the SGPV indicates when the data are compatible with null hypotheses ($p_\delta = 1$), or with alternative hypotheses ($p_\delta = 0$), or when the data are inconclusive ($0 < p_\delta < 1$).

Notice also that the correction term $\max\{|I|/(2|H_0|), 1\}$ resolves any problems that arise when the interval estimate I is too wide to be useful or reliable, in which case the data are effectively deemed inconclusive. It is in this way that SGPVs emphasize effects that are clinically meaningful over effects that are small and near the null hypothesis. Empirical studies have shown how SGPVs can be used to identify feature importance in high dimensional settings.^{3,4,22,23} The null bound in the SGPVs is typically the smallest effect that would be clinically relevant or the effect magnitude that can be distinguished from noise, on average.^{3,4} proposed using a generic null interval for regression coefficients that shrinks to zero and is based on the observed level of noise in the data. This extends the null bound in^{22,23} that remains constant. The interval is easily obtained in the variable selection step and promotes good statistical properties in the selection algorithm.

The ProSGPV algorithm

The ProSGPV algorithm is a two-stage algorithm. In the first stage, the algorithm identifies a candidate set of variables using a broad regularization scheme. In the second stage, the algorithm applies SGPV regularization to the model based on the candidate set identified in the first stage. The successive regularization approach is easy and fast to implement, and quite accurate for screening out false features. The steps of the ProSGPV algorithm are shown in Algorithm 1.

Algorithm 1. The ProSGPV algorithm.

- 1: procedure ProSGPV(**X**, **Y**)
- 2: **Stage one:** Find a candidate set
- 3: Fit a lasso and evaluate it at λ_{gic}
- 4: Fit OLS/GLM/Cox models on the lasso active set
- 5: **Stage two:** SGPV screening
- 6: Extract the confidence intervals of all variables from the previous step
- 7: Calculate the mean coefficient standard error $\overline{SE}$
- 8: Calculate the SGPV for each variable where $I_j = \hat{\beta}_j \pm 1.96 \times SE_j$ and $H_0 = [-\overline{SE}, \overline{SE}]$
- 9: Keep variables with SGPV of zero
- 10: Refit the OLS/GLM/Cox with selected variables
- 11: end procedure

By default, data are scaled and centered in linear regression but are not transformed as such in GLM and Cox regression. No notable difference is observed when data are standardized in GLM and Cox regression. In the first stage, lasso is used to reduce the feature space to a candidate set that is very likely to contain true signals. This pre-screening is crucial to high dimensional data ($n < p$) and improves the support recovery and parameter estimation in low dimensional data.³ The lasso is evaluated at λ_{gic} , but the algorithm is robust with respect to the choice of λ .³ In the second stage, the null bound is set to be the mean standard error of all coefficient estimates. However, the algorithm is insensitive to any reasonable scale change in the null bound.^{3,4} When data are highly correlated, a generalized variance inflation factor (GVIF)²⁴ adjusted null bound can be used to improve the inference and prediction performance.⁴

In essence, ProSGPV is a hard thresholding algorithm that shrinks small effect to zero and reserve the large effect to obtain an unbiased estimate when the true support is successfully recovered. Notation-wise, the solution to the ProSGPV algorithm $\hat{\beta}^{\text{prosgpv}}$ is

$$\hat{\beta}^{\text{prosgpv}} = \hat{\beta}_S^m \in \mathbb{R}^p, \text{ where} \quad (2)$$

$$S = \left\{ k \in C : |\hat{\beta}_k^m| > \lambda_k \right\}, C = \left\{ j \in \{1, 2, \dots, p\} : |\hat{\beta}_j^{\text{lasso}}| > 0 \right\}$$

where $\hat{\beta}_S^m$ is a vector of length p with non-zero elements being the OLS/GLM/Cox coefficient estimates from the model with variables only in the set S . S is the final selection set and C is the candidate set from the first-stage screening. Note that the cutoff $\lambda_k = 1.96 * SE_k + SE$.

When λ_{gic} in Algorithm 1 is replaced with zero in the first stage, ProSGPV reduces to a one-stage algorithm. That amounts to calculating SGPVs for each variable in the full model and select variables whose effects are above the threshold. However, the support recovery and parameter estimation performance of the one-stage algorithm is slightly worse than that of the two-stage algorithm.³ Moreover, the one-stage algorithm is not applicable when $p > n$, i.e. when the full OLS/GLM/Cox model is not identifiable.

Implementation

The ProSGPV package is publicly available from the [Comprehensive R Archive Network \(CRAN\)](#), a development version is available on [GitHub](#), and is archived with [Zenodo](#).²⁹ To install from CRAN, please run

```
install.packages("ProSGPV")
```

To install a development version, please run

```
library(devtools)
devtools::install_github("zuoyi93/ProSGPV")
```

The main function `pro.sgpv` implements the default two-stage and optional one-stage ProSGPV algorithm. User-friendly `print`, `coef`, `summary`, `predict`, and `plot` functions are equipped with `pro.sgpv` for both one- and two-stage algorithms. Jeffreys prior penalized logistic regression²⁵ is used when the outcome is binary to stabilize coefficient estimates in the case of complete/quasi-complete separation. In the next section, we demonstrate how `pro.sgpv` works with simulated continuous outcome data.

Operation

The ProSGPV package works across different platforms (Windows, mac OS, and Linux). The R version number should be greater than 3.5.0. Once installed, the workflow is described as below. We first present an example by applying the ProSGPV algorithm to a simulated dataset by use of `gen.sim.data` function. With sample size $n = 100$, number of variables $p = 20$, number of true signals $s = 4$, smallest effect size $\beta_{\min} = 1$, largest effect size $\beta_{\max} = 5$, autoregressive correlation $\rho = 0.2$ and variance $\sigma^2 = 1$, signal-to-noise ratio (SNR) defined in²⁶ $v = 2$, we generate outcomes Y following Gaussian distribution. `gen.sim.data` outputs X , Y , indices of true signals, and a vector of true coefficients.

```

> library (ProSGPV)
> set.seed(1)
> sim.data <- gen.sim.data(n = 100, p = 20, s = 4, family = "gaussian",
                           beta.min = 1, beta.max = 5, rho = 0.2, nu = 2)
> x <- sim.data [[1]]
> y <- sim.data [[2]]
> (true.index <- sim.data [[3]])

[1] 2 4 5 7

> true.beta <- sim.data [[4]]

```

By default, the two-stage algorithm is used in ProSGPV. `pro.sgpv` function takes inputs of explanatory variables `x`, outcome `y`, outcome type `family` (default is "gaussian"), stage indicator (default is 2), and a GVIF indicator (default is FALSE). A print method is available to show labels of the variables selected by ProSGPV.

```

> sgpv.out.2 <- pro.sgpv(x,y)
> sgpv.out.2

Selected variables are V2 V4 V5 V7

```

The variable selection process can be visualized by using the plot function. Figure 2 shows the fully relaxed lasso path along a range of λ_s . The shaded area is the null zone and any effect whose 95% confidence interval overlaps with the null region will be considered irrelevant or insignificant. ProSGPV is evaluated at λ_{gic} . The `lpv` argument can be used to choose to display one line per variable (the confidence bound that is closer to the null region) or three lines per variable (an point estimate and confidence bounds, default). The `lambda.max` argument control the limit of the x-axis in the plot.

summary function outputs the regression summary of the selected model. When the outcome is continuous, an OLS is used.

```

> summary (sgpv.out.2)

Call:
lm (formula = Response ~ ., data = data.d)

Residuals:

```

Min	1Q	Median	3Q	Max
-9.8066	-3.5912	0.0817	3.1530	9.5042

Coefficients:

	Estimate	Std.	Error	t value	Pr(> t)
(Intercept)	-0.02359		0.48792	-0.048	0.96154
V2	-4.19922		0.46775	-8.978	2.53e-14 ***
V4	1.59168		0.52876	3.010	0.00334 **
V5	2.66004		0.47441	5.607	2.01e-07 ***
V7	-3.31497		0.46207	-7.174	1.58e-10 ***

```

---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.834 on 95 degrees of freedom
Multiple R-squared: 0.6367, Adjusted R-squared: 0.6214
F-statistic: 41.63 on 4 and 95 DF, p-value: < 2.2e-16

```

`coef` function can be used to extract the coefficient estimates of length p . When signals are sparse, some of estimates are zero. A comparison shows that the estimates are close to the true effect sizes in the simulation.

```
> beta.hat <- coef (sgpv.out.2)
> rbind (beta.hat, true.beta)
```

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]
beta.hat	0	-4.199224	0	1.591675	2.660038	0	-3.314971	0	0	0
true.beta	0	-5.000000	0	1.000000	2.333333	0	-3.666667	0	0	0

`predict` function can be used to predict outcomes using the selected model. In-sample prediction can be made by calling `predict (sgpv.out.2)` and out-of-sample prediction can be made by feeding new data set into the `newdata` argument.

In addition to the two-stage algorithm, one-stage algorithm can also be used to select variables when $n > p$. The computation time is shorter for the one-stage algorithm at the expense of slightly reduced support recovery rate in the limit, as shown in.³ Figure 3 shows the variable selection result of the one-stage algorithm on the same data. The one-stage algorithm missed V4 and only selected three variables. The lower confidence bound of the estimate for V4 barely excludes the null region, and was dropped from the final model because of that. The one-stage algorithm illustrates how estimation uncertainty via confidence intervals (horizontal segments) can be incorporated in variable selection.

Solution to the two-stage algorithm with three lines per variable

Figure 2. Solution path of the two-stage ProSGPV algorithm.

Figure 3. Visualization of variable selection results of the one-stage ProSGPV algorithm.

```
> sgpv.out.1 <- pro.sgpv(x,y,stage=1)
> sgpv.out.1
Selected variables are V2 V5 V7

> plot (sgpv.out.1)
```

Examples of binary, count, and time-to-event data can be found in the package vignettes.

Simulation

Simulation design is adapted from^{3,4} and we will present below scenarios not discussed in those two papers. The setup can be found in the [Table 1](#) below. The scale and shape parameters are used to generate time-to-event from a Weibull distribution. ProSGPV is compared against lasso, BeSS, and ISIS. Results are aggregated over 1000 simulations. Evaluation metrics include support recovery rate, parameter estimation mean absolute error (MAE), prediction root mean square error (RMSE) and area under the curve in a separate test set. Support recovery is defined as capturing the exact true support, not containing. An estimate of MAE is $1/p \sum_{j=1}^p \|\hat{\beta}_j - \beta_{0,j}\|$, where $\beta_{0,j}$ is the j th true coefficient. Simulation results are summarized in [Figure 4](#).

From [Figure 4](#), we observe similar results as in^{3,4} ProSGPV often has the highest capture rates of the exact true model, has lowest parameter estimation bias, has one of the lowest prediction error, and is the fastest to compute. Note that GVIF-adjusted null bound is used in the Logistic regression because of the high correlation in the design matrix.

Table 1. Summary of parameters in simulation studies.

	Linear regression	Logistic regression	Poisson regression	Cox regression
n	100	32:320	40	80:800
p	100:1000	16	40:400	40
s	10	6	4	20
β_l	1	0.4	0.2	0.3
β_u	2	1.2	0.5	1
ρ	0.3	0.6	0.3	0.3
σ	2	2	2	2
v	2			
Intercept t		0	2	0
Scale				2
Shape				1
Rate of censoring				0.2

Figure 4. Simulation results over 1000 iterations. A) Average support recovery rate, means surrounded by 95% Wald confidence intervals. B) Average mean absolute error, medians surrounded by 1st and 3rd quartiles. C) Prediction accuracy in a separate test set (root mean square error for linear and Poisson regressions, and area under the curve for Logistic regression), medians surrounded by 1st and 3rd quartiles. D) Average running time in seconds, medians surrounded by 1st and 3rd quartiles. Values are censored at 0.8 seconds for aesthetic reasons.

Figure 5. Sparsity of solutions (A) and prediction performance (B) of all algorithms in the real-world example. Medians as well as 1st and 3rd quartiles from 1000 repetitions are compared.

Results

Real-world data example

In this section, we compare the performance of ProSGPV with lasso, BeSS, and ISIS algorithms using a real-world financial data.²⁸ The close price of Dow Jones Industrial Average (DJIA) was documented from Jan 1, 2010 to November 15, 2017 and eight groups of primitive, technical indicators, big U.S. companies, commodities, exchange rate of currencies, future contracts and worlds stock indices, and other sources of information²⁷ were collected to predict the DJIA close price. In the analyzed data with complete records, there are 1114 observations and 82 predictors. We randomly sampled 614 observations as a fixed test set to evaluate the prediction performance of models built on the training set. We allowed the training set size n to vary from 40 to 300. At each n , we recorded the distribution of the training model size for each algorithm as well as the distribution of the prediction RMSE over 1000 repetitions. Results are summarized in Figure 5. ProSGPV and lasso had sparser models than BeSS and ISIS did, while ProSGPV had much better prediction performance in the test set. BeSS and ISIS had better prediction performance at the cost of including much more variables in the final model. The tradeoff between the sparsity of solutions and prediction accuracy is well illustrated in this example. Variables frequently selected by ProSGPV include 5-, 10-, and 15-day rate of change, and 10-day exponential moving average of (DJIA). Technical indicators seem more predictive than other world indices, commodity, exchange rate, futures, etc.

Conclusions

We introduced an R package to implement the ProSGPV algorithm, a variable selection algorithm that incorporates estimation uncertainty via confidence intervals. This novel addition often leads to better support recovery, parameter estimation, and prediction performance. The package is user-friendly, has nice visualization tools for the variable selection process, facilitates subsequent inference and prediction, and very fast computationally. The efficacy of our package is demonstrated on both simulation and real-world data sets.

Data availability

Source data

The real-world data are from a Kaggle data challenge, which is available at <https://www.kaggle.com/ehoseinz/cnnpred-stock-market-prediction>. Detailed description of data elements can be found in.²⁷

Underlying data

Zenodo: zuoyi93/r-code-prosgpv-r: v1.0.0. <https://doi.org/10.5281/zenodo.5655772>.²⁸

- Processed_DJI.csv (real-world data)

Data are available under the terms of the [Creative Commons Zero “No rights reserved” data waiver](#) (CC0 1.0 Public domain dedication).

Extended data

Analysis code available from: <https://github.com/zuoyi93/r-code-prosgpv-r/tree/v1.0.0#readme>

Archived analysis code as at time of publication: <https://doi.org/10.5281/zenodo.5655772>.²⁸

License: [Creative Commons Zero](#) “No rights reserved” data waiver (CC0 1.0 Public domain dedication).

Software availability

Software available from: <https://cran.r-project.org/web/packages/ProSGPV/index.html>

Source code available from: <https://github.com/zuoyi93/ProSGPV>

Archived source code at time of publication: <https://doi.org/10.5281/zenodo.5655795>.²⁹

License: [3-Clause BSD License](#).

References

- Meinshausen N, Bühlmann P: **High-dimensional graphs and variable selection with the lasso**. *Ann. Stat.* 2006; **34**(3): 1436–1462.
[Publisher Full Text](#)
- Shmueli G: **To explain or to predict?** *Stat. Sci.* 2010; **25**(3): 289–310.
[Publisher Full Text](#)
- Zuo Y, Stewart TG, Blume JD: **Variable selection with second-generation p-values**. *Am. Stat.* 2021; 1–11.
[Publisher Full Text](#)
- Zuo Y, Stewart TG, Blume JD: **Variable selection in glm and cox models with second-generation p-values**. *arXiv preprint arXiv: 2109.09851*. 2021.
- Natarajan BK: **Sparse approximate solutions to linear systems**. *SIAM J. Comput.* 1995; **24**(2): 227–234.
[Publisher Full Text](#)
- Hazimeh H, Mazumder R: **Fast best subset selection: Coordinate descent and local combinatorial optimization algorithms**. *Oper. Res.* 2020; **68**(5): 1517–1537.
[Publisher Full Text](#)
- Di Gangi L, Lapucci M, Schoen F, et al.: **An efficient optimization approach for best subset selection in linear regression, with application to model selection and fitting in autoregressive time-series**. *Comput. Optim. Appl.* 2019; **74**(3): 919–948.
[Publisher Full Text](#)
- Weng H, Feng Y, Qiao X: **Regularization after retention in ultrahigh dimensional linear regression models**. *Stat. Sin.* 2019; **29**(1): 387–407.
[Publisher Full Text](#)
- Tibshirani R: **Regression shrinkage and selection via the lasso**. *Journal of the Royal Statistical Society: Series B (Methodological)*. 1996; **58**(1): 267–288.
[Publisher Full Text](#)
- Leng C, Lin Y, Wahba G: **A note on the lasso and related procedures in model selection**. *Stat. Sin.* 2006; 1273–1284.
- Zou H: **The adaptive lasso and its oracle properties**. *J. Am. Stat. Assoc.* 2006; **101**(476): 1418–1429.
[Publisher Full Text](#)
- Friedman J, Hastie T, Tibshirani R: **Regularization paths for generalized linear models via coordinate descent**. *J. Stat. Softw.* 2010; **33**(1): 1–22.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Simon N, Friedman J, Hastie T, et al.: **Regularization paths for cox's proportional hazards model via coordinate descent**. *J. Stat. Softw.* 2011; **39**(5): 1–13.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Fan J, Li R: **Variable selection via nonconcave penalized likelihood and its oracle properties**. *J. Am. Stat. Assoc.* 2001; **96**(456): 1348–1360.
[Publisher Full Text](#)
- Zhang C-H: **Nearly unbiased variable selection under minimax concave penalty**. *Ann. Stat.* 2010; **38**(2): 894–942.
[Publisher Full Text](#)
- Fan J, Lv J: **Sure independence screening for ultrahigh dimensional feature space**. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*. 2008; **70**(5): 849–911.
[Publisher Full Text](#)
- Fan J, Song R: **Sure independence screening in generalized linear models with np-dimensionality**. *Ann. Stat.* 2010; **38**(6): 3567–3604.
[Publisher Full Text](#)
- Saldana DF, Feng Y: **Sis: An r package for sure independence screening in ultrahigh-dimensional statistical models**. *J. Stat. Softw.* 2018; **83**(1): 1–25.
- Reid S, Tibshirani R: **Regularization paths for conditional logistic regression: the clogit1 package**. *J. Stat. Softw.* 2014; **58**(12):
[Publisher Full Text](#)
- Dvorzak M, Wagner H: **Sparse bayesian modelling of underreported count data**. *Stat. Model.* 2016; **16**(1): 24–46.
[Publisher Full Text](#)
- Beretta A, Heuchenne C: **penphcure: Variable selection in proportional hazards cure model with time-varying covariates**. *R Journal.* 2021; **13**(1).
[Publisher Full Text](#)
- Blume JD, D'Agostino McGowan L, Dupont WD, et al.: **Second-generation p-values: Improved rigor, reproducibility, & transparency in statistical analyses**. *PLoS One.* 2018; **13**(3): e0188299.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Blume JD, Greevy RA, Welty VF, et al.: **An introduction to second-generation p-values**. *Am. Stat.* 2019; **73**(sup1): 157–167.
[Publisher Full Text](#)
- Fox J, Monette G: **Generalized collinearity diagnostics**. *J. Am. Stat. Assoc.* 1992; **87**(417): 178–183.
[Publisher Full Text](#)
- Kosmidis I, Firth D: **Jeffreys-prior penalty, finiteness and shrinkage in binomial-response generalized linear models**. *Biometrika.* 2021; **108**(1): 71–82.
[Publisher Full Text](#)
- Hastie T, Tibshirani R, Tibshirani R: **Best subset, forward stepwise or lasso? analysis and recommendations based on extensive comparisons**. *Stat. Sci.* 2020; **35**(4): 579–592.
[Publisher Full Text](#)
- Hoseinzade E, Haratizadeh S: **Cnnpred: Cnn-based stock market prediction using a diverse set of variables**. *Expert Syst. Appl.* 2019; **129**: 273–285.
[Publisher Full Text](#)
- Zuo Y: **zuoyi93/r-code-prosgpv-r: v1.0.0 (v1.0.0)**. *Zenodo*. 2021.
[Publisher Full Text](#)
- Zuo Y: **ProSGPV R package v1.0.0**. *Zenodo*. 2021.
[Publisher Full Text](#)

Open Peer Review

Current Peer Review Status: ?

Version 1

Reviewer Report 26 January 2022

<https://doi.org/10.5256/f1000research.78152.r120267>

© 2022 Heinze G. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

? **Georg Heinze**

Section for Clinical Biometrics, Center for Medical Statistics, Informatics and Intelligent Systems, Medical University of Vienna, Vienna, Austria

The paper describes an R-package with which a recently proposed method of variable selection, ProSGPV, can be applied to data sets.

I am not fully convinced of the method. It would need a neutral comparison study (one not conducted by the authors who invented the method) to prove if it's useful. But to conduct such a study, an efficient implementation of the method is needed, which is introduced in this report. So the paper has some relevance.

Introduction: 'classical statistical methods exploiting the full feature space no longer work': but any variable selection algorithm would have to exploit the full feature space when searching for predictive variables?

One may be interested in getting a sparse solution because one assumes that only a small subset of features is really predictive, or if a smaller prediction model has other advantages, e.g. costs or practicability. But there are many modeling techniques that do not sparsify the feature space which works very well with high-dimensional data such as ridge regression. So more convincing arguments for the need for data-driven variable selection should be given.

Figure 1 may need some more explanations in the caption.

At step 9 of the algorithm, only variables with a SGPV of zero are kept. What if two variables have a high correlation and both of their SGPV are greater than zero, but omitting them both from the model would reduce the information significantly. Isn't this a drawback of this procedure? Why not sequentially (one-by-one) eliminating variables and refitting the model?

In the definition of the null interval, the authors speak of 'minimum clinical relevance' (so having medical applications in mind), but the example is from the financial world. Could the authors give some idea how to define the null interval for other areas of application?

It seems that the SGPV depends on the confidence levels because with lower confidence levels, the intervals are shorter and the probability to get SGPV of 0 gets larger.

The empirical null interval definition seems a bit arbitrary (mean of standardized standard errors). Why not the root of the mean of standardized variances? Why take the mean at all? Does this imply any assumptions on an equal a-priori importance of the variables? What about correlated

variables?

In the simulation study, the authors confounded the type of regression with varying n or p . This simulation can only be seen as proof that the package works, but I am not so convinced yet that it also proves the advantages of the new method. So the results still have to be interpreted with caution. We don't know if the simulation scenarios were pre-specified or selected after the fact, and if the simulated joint covariates-outcome distributions are representative of any real data sets one encounter in practice.

Is the rationale for developing the new software tool clearly explained?

Yes

Is the description of the software tool technically sound?

Yes

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?

Partly

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?

Partly

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?

Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: biostatistics, clinical epidemiology, prognosis research

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Author Response 07 Feb 2022

Yi Zuo

Dear Dr. Heinze,

Thank you for taking the time to review our paper. We appreciate your thoughtful comments and we think these concerns can be readily addressed. Please see our responses below.

Comment 1

Introduction: 'classical statistical methods exploiting the full feature space no longer work': but any variable selection algorithm would have to exploit the full feature space when searching for

predictive variables? One may be interested in getting a sparse solution because one assumes that only a small subset of features is really predictive, or if a smaller prediction model has other advantages, e.g. costs or practicability. But there are many modeling techniques that do not sparsify the feature space which works very well with high-dimensional data such as ridge regression. So more convincing arguments for the need for data-driven variable selection should be given.

Response: The issue is not that the methods exploit the full feature space, but rather how they do it. Our comment was in the context of $p > n$, i.e. when there are more features than samples. In this setting, many of the classical approaches, like stepwise regression, are unable to fit all of the candidate models due to lack of data. While a ridge regression would be useful for prediction, it does not eliminate any features and is therefore not considered a “variable selection” algorithm. The sparsity assumption is commonly accepted in many applications. Also, small here is relative to the size of the entire feature space (including all interactions and non-linear terms) which is often quite big. So sparseness is not all that constraining in practice.

Comment 2a

Figure 1 may need some more explanations in the caption. At step 9 of the algorithm, only variables with a SGPV of zero are kept. What if two variables have a high correlation and both of their SGPV are greater than zero, but omitting them both from the model would reduce the information significantly. Isn't this a drawback of this procedure? Why not sequentially (one-by-one) eliminating variables and refitting the model?

Response: When data are highly correlated, successful support recovery happens less often (it happens less often for all approaches). Lasso screening handles highly correlated feature sets well, which is why we use it in the first stage. In addition, using the generalized variance inflation factor (GVIF) proposed in improves the performance of the ProSGPV algorithm. Details can be found in . But we do not see this as a drawback of the procedure, as it continues to outperform competitors.

Regarding sequential elimination, what you propose is similar to backward/stepwise regression only with SGPVs. When we first explored our algorithm, we did explore this procedure and found that it leads to poor variable selection performance, especially when there are minor changes to the data. Its performance did not warrant us mentioning it in the paper. In addition, sequentially dropping variables adds increases the computation burden when the number of candidate variables is large. By contrast, our proposed method is much more efficient.

Comment 2b

In the definition of the null interval, the authors speak of 'minimum clinical relevance' (so having medical applications in mind), but the example is from the financial world. Could the authors give some idea how to define the null interval for other areas of application? It seems that the SGPV depends on the confidence levels because with lower confidence levels, the intervals are shorter and the probability to get SGPV of 0 gets larger.

This is a good question. We addressed this directly with sensitivity analysis in

Supplementary Figure 1 in < DOI: 10.1080/00031305.2021.1946150 >. The major difference with this SPGV implementation is that the null zone, because it is the average standard error of the standardized coefficients, shrinks to zero as the sample size grows. Hence our approach is less sensitive to the specification of the zone. However, this is not necessarily surprising, as argued that in order to achieve optimal properties of variable selection, the amount of lasso shrinkage must be proportional to the standard error of the maximum likelihood estimates of coefficients. As for the dependence on the confidence interval coefficient, this is unavoidable. The SGPV framework chooses to make this connection explicit so that it is easy for the user to know what the upper bound on the Type I error rate is (it is one minus the confidence interval coefficient). This clearly affects the frequency properties of the overall procedure in the same way that setting the Type I error cutoff for stepwise regression does.

Comment 3

The empirical null interval definition seems a bit arbitrary (mean of standardized standard errors). Why not the root of the mean of standardized variances? Why take the mean at all? Does this imply any assumptions on an equal a-priori importance of the variables? What about correlated variables?

Response: As we mentioned above, the choice of the null bound was inspired by . In linear regression, we do standardize all explanatory variables so that the standard errors are comparable, and, in this setting, it makes sense to average them. In GLM/Cox regression, we observed in the simulation studies that standardization does not affect the support recovery, so we abandoned it to reduce computation steps. When data are highly correlated, GVIF is used to adjust the null bound which improves the support recovery performance.

Comment 4

In the simulation study, the authors confounded the type of regression with varying n or p. This simulation can only be seen as proof that the package works, but I am not so convinced yet that it also proves the advantages of the new method. So the results still have to be interpreted with caution. We don't know if the simulation scenarios were pre-specified or selected after the fact, and if the simulated joint covariates-outcome distributions are representative of any real data sets one encounter in practice.

Response: We find this argument specious. It is certainly possible that some simulation settings yield excellent results while others, close in the parameter space, do not. While this is not the case here – we found no evidence of highly erratic behavior – no amount of simulating can address this concern. We prepared extensive simulations by varying n, p, s (sparsity of the solution), effect size, correlation structure, and signal-to-noise ratio. There is simply no way to report on them all; however, the general patterns of behavior are consistent, and this is what we report. In all cases considered, our proposed algorithm had one of the best support recovery performances, nearly optimal parameter estimation properties, and acceptable prediction performance. Note that when the effect size is small or the signal-to-noise ratio is low, none of the algorithms we considered yielded good results. That said, our code is accessible and easy to use, and we encourage double-checking and reproduction of our results.

Competing Interests: None

The benefits of publishing with F1000Research:

- Your article is published within days, with no editorial bias
- You can publish traditional articles, null/negative results, case reports, data notes and more
- The peer review process is transparent and collaborative
- Your article is indexed in PubMed after passing peer review
- Dedicated customer support at every stage

For pre-submission enquiries, contact research@f1000.com

F1000Research