Journal of Statistical Software

MMMMMM YYYY, Volume VV, Issue II. *doi: 10.18637/jss.v000.i00*

SurrogateRsq: an R package for categorical data goodness-of-fit analysis using the surrogate R^2

Xiaorui Zhu
Towson University

Zewei Lin
University of Cincinnati

Dungang Liu
University of Cincinnati

Brandon Greenwell
84.51° and University of Cincinnati

Abstract

Categorical data are prevalent in almost all research fields and business applications. Their statistical analysis and inference often rely on probit/logistic regression models. For these common models, however, there is no universally adopted measure so as to perform goodness-of-fit analysis. To this end, Liu, Zhu, Greenwell, and Lin (2023) proposed a so-called surrogate R^2 that resembles the ordinary least square (OLS) R^2 for linear regression models. The surrogate R^2 used the notion of surrogacy, namely, generating a continuous response S and using it as a surrogate of the original categorical response Y (Liu and Zhang 2018; Liu, Li, Yu, and Moustaki 2021; Cheng, Wang, and Zhang 2021). In this paper, we develop an R package **SurrogateRsq** to implement the surrogate R^2 method. The package is compatible with existing model fitting functions (e.g., `glm()`, `polr()`, `clm()`, and `vglm()`), and its features are exhibited in a wine rating analysis. Our package can be used jointly with other R packages developed for variable selection and model diagnostics so as to form a complete model development process. This process is summarized and demonstrated in a categorical-data-modeling workflow that practitioners can follow. To exemplify an extended utility of the surrogate-R^2-based goodness-of-fit analysis, we also use this package to illustrate how to compare different empirical models trained from different samples in the wine rating analysis. The result suggest that the package allows us to evaluate comparability across multiple samples/models/studies that address the same or similar scientific or business question.

Keywords: categorical data analysis, goodness-of-fit measure, logistic regression, model comparison, probit model, surrogate method, surrogate residual.

1. Introduction

Categorical data are prevalent in all areas, including economics, marketing, finance, psychology, and clinical studies. To analyze categorical data, the probit or logit models are often used to make inferences. To perform model assessment and comparison, researchers often rely on goodness-of-fit measures, such as R^2 (also known as the coefficient of determination). For example, the ordinary least square (OLS) R^2 is one of the most extensively used goodness-of-fit measures for linear models in continuous data analysis. For categorical data analysis, however, there is no such universally adopted R^2 measure (Hagle and Mitchell 1992; Veall and Zimmermann 1996). There have been continuous efforts in developing sensible R^2 measures for probit/logistic models, and more generally, generalized linear models (McFadden 1973; McKelvey and Zavoina 1975; Efron 1978; Cox and Wermuth 1992; Laitila 1993; Zheng and Agresti 2000; Liu and Agresti 2005; Hu, Shao, and Palta 2006; Liu *et al.* 2023). Among the existing R^2 measures, McKelvey-Zavoina's R^2_{MZ} (McKelvey and Zavoina 1975) and McFadden's R^2 (McFadden 1973) are probably the most well-known and widely used in domain research (Hagle and Mitchell 1992; Veall and Zimmermann 1996). But as demonstrated in Liu *et al.* (2023), Mckelvey-Zavoina's R^2_{MZ} does not hold monotonicity, which means a larger model may have a smaller R^2_{MZ}. This serious defection of R^2_{MZ} may be misleading in practice and misguide the model-building process. On the other hand, McFadden's R^2 relies on the ratio of likelihoods, and it does not preserve the interpretation of explained variance. Neither of these two R^2 measures meets all of the three criteria considered in Liu *et al.* (2023):

(C1) It can approximate the OLS R^2 based on the latent continuous outcome.

(C2) It has the interpretation of the explained proportion of variance.

(C3) It maintains the monotonicity property between nested models, which means that a larger model should have a larger R^2 value.

Liu *et al.* (2023) proposed a so-called surrogate R^2 that satisfies all three criteria for probit models. This surrogate R^2 used the notion of surrogacy, namely, generating a continuous response S and using it as a surrogate for the original categorical response Y (Liu and Zhang 2018; Liu *et al.* 2021; Cheng *et al.* 2021; Greenwell, McCarthy, Boehmke, and Liu 2018; Li, Zhu, Chen, and Liu 2021). In the context of probit analysis, Liu *et al.* (2023) used the truncated distributions induced by the latent variable structure to generate a surrogate response S. This surrogate response S is then regressed on explanatory variables through a linear model. The OLS R^2 of this linear model is used as a surrogate R^2 for the original probit model. This surrogate R^2 meets all three criteria (C1)-(C3).

The goals of this paper are (i) developing an R package to implement Liu *et al.* (2023)'s method; (ii) demonstrating how this new package can be used jointly with other existing R packages for variable selection and model diagnostics in the model building process; and (iii) illustrating how this package can be used to compare different empirical models trained from two different samples (a.k.a. comparability) in real data analysis.

Specifically, we first develop an R package to implement the surrogate R^2 method for probit/logistic regression models. This package contains the R functions for generating point and interval estimates of the surrogate R^2 measure. The point/interval estimates allow researchers and practitioners to evaluate the model's overall goodness of fit and understand its uncertainty. In addition, we develop an R function that calculates the percentage contribution of each variable to the overall surrogate R^2. This percentage reflects each variable's contribution to the model's overall explanatory power. Based on the contribution's relative size, our R function provides a "importance" ranking of all the explanatory variables.

Second, to provide practical guidance for categorical data modeling, we use the developed R package to demonstrate how it can be used jointly with other R packages developed for variable screening/selection and model diagnostics (**leaps** (Lumley and Lumley 2013), `step()` function from the R core, **glmnet** (Friedman, Hastie, and Tibshirani 2010), **ordinalNet** (Wurm, Rathouz, and Hanlon 2021), **ncvreg** (Breheny 2013), **grpreg** (Breheny and Breheny 2014), **SIS** (Saldana and Feng 2018), **sure** (Greenwell *et al.* 2018), **PAsso** (Zhu, Li, Chen, and Liu 2020)). In particular, we recommend a workflow that consists of three steps, including variable screening/selection, model diagnostics, and goodness-of-fit analysis. The workflow is illustrated in the analysis of wine-tasting preference datasets.

Third, the comparability of the surrogate R^2 across different samples and/or models allows us to compare goodness-of-fit analysis from similar studies. The comparison can lead to additional scientific/business insights which may be useful for decision making. To illustrate this, we conduct goodness-of-fit analysis separately for the red wine and white wine samples to demonstrate the comparability of the surrogate R^2. Our analysis result reveals that (i) the same set of explanatory variables has different explanatory power for red wine and white wine (43.8% versus 31.0%), and (ii) the importance ranking of the explanatory variable (in terms of their contribution to the surrogate R^2) is different between red wine and white wine.

Our **SurrogateRsq** package has broad applicability. It is compatible with the following R functions that can fit probit/logistic regression models for a binary or ordinal response: `glm()` in the **R** core, `polr()` in the **MASS** package (Ripley, Venables, Bates, Hornik, Gebhardt, Firth, and Ripley 2013), `clm()` in the **ordinal** package (Christensen 2019), and `vglm()` in the **VGAM** package (Yee *et al.* 2010).

2. Review of the Surrogate R^2

We briefly review the surrogate R^2 measure in the study of Liu *et al.* (2023). For the model setting, we consider a probit/logit model with a set of explanatory variables. The categorical response is either a binary or ordinal variable Y that has J categories $\{1, 2, \ldots, J\}$, with the order $1 < 2 < \cdots < J$,

$$\Pr\{Y \leq j\} = G\{\alpha_j - (\beta_1 X_1 + \cdots + \beta_l X_p)\}, \quad j = 1, \ldots, J, \tag{1}$$

where $-\infty < \alpha_1 < \cdots < \alpha_J < +\infty$. The link function $G(\cdot)$ can be a probit $(G(\cdot) = \Phi(\cdot))$ link or a logit $(G(\eta) = 1/(1 + e^{-\eta})$). Each generic symbol of $\{X_1, \ldots X_p\}$ in Model (1) can represent a single variable of interest, a high-order term (e.g., X^2), or an interaction term between X and another variable. It is well-known that an equivalent way to express Model (1) is through a latent variable. For example, if the link is probit, the latent variable has the following form with a normally distributed ϵ:

$$Z = \alpha_1 + \beta_1 X_1 + \cdots + \beta_p X_p + \epsilon, \quad \epsilon \sim N(0, 1).$$

The categorical response Y can be viewed as generated from censoring the continuous latent variable Z in the following way:

$$Y = \begin{cases} 1 & \text{if } -\infty < Z \leq \alpha_1 + \alpha_1, \\ 2 & \text{if } \alpha_1 + \alpha_1 < Z \leq \alpha_2 + \alpha_1, \\ \cdots & \\ J & \text{if } \alpha_{J-1} + \alpha_1 < Z < +\infty. \end{cases}$$

To construct a goodness-of-fit R^2, Liu *et al.* (2023) adopted the surrogate approach proposed by Liu and Zhang (2018). The idea of the surrogate approach is to simulate a continuous variable and use it as a surrogate for the original categorical variable in the analysis (Liu and Zhang 2018; Liu *et al.* 2021; Cheng *et al.* 2021). In the context of probit models, Liu *et al.* (2023) proposed to generate a surrogate response variable using the following truncated conditional distribution:

$$S \sim \begin{cases} Z \mid -\infty < Z \leq \alpha_1 + \alpha_1 & \text{if } Y = 1, \\ Z \mid \alpha_1 + \alpha_1 < Z \leq \alpha_2 + \alpha_1 & \text{if } Y = 2, \\ \cdots & \\ Z \mid \alpha_{J-1} + \alpha_1 < Z < +\infty & \text{if } Y = J. \end{cases}$$

Liu *et al.* (2023) proposed to regress the surrogate response S on $\{X_1, \ldots, X_p\}$ using a linear model below:

$$S = \alpha_1 + \beta_1 X_1 + \cdots + \beta_p X_p + \epsilon, \quad \epsilon \sim N(0,1). \tag{2}$$

Their approach used the OLS R^2 measure of this linear model as a surrogate R^2 for Model (1):

$$R^2_{(S)}\{X_1, \ldots, X_p\} = \text{the OLS } R^2 \text{ of the linear model (2)}.$$

Liu *et al.* (2023) showed that the surrogate $R^2_{(S)}$ measure has three desirable properties. First, it approximates the OLS R^2 calculated using the latent continuous outcome Z. This property enables us to compare surrogate R^2's and OLS R^2's across different models and samples that address the same scientific question. Second, as it is the OLS R^2 calculated using the continuous surrogate response S, the surrogate $R^2_{(S)}$ has the interpretation of the explained proportion of variance. It measures the explained proportion of the variance of the surrogate response S through the linear model. This explained proportion of variance implies the explanatory power of all the features in the fitted model. Third, the surrogate $R^2_{(S)}$ maintains monotonicity between nested models, which makes it suitable for comparing the relative explanatory power of different models. In contrast, the well-known McFadden's R^2 does not preserve the first two properties of the surrogate $R^2_{(S)}$. McFadden's R^2 relies on the ratio of likelihoods, so it neither approximates the OLS R^2 nor preserves the interpretation of explained variance. On the other hand, Liu *et al.* (2023) showed that McKelvey-Zavoina's R^2_{MZ} did not necessarily maintain monotonicity between nested models. This serious issue may make McKelvey-Zavoina's R^2_{MZ} an unsuitable tool for measuring the goodness of fit.

To make inferences for the surrogate $R^2_{(S)}$, Liu *et al.* (2023) provided procedures to produce point and interval estimates. Since the surrogate response S is obtained through simulation, Liu *et al.* (2023) used a multiple-sampling scheme to "stabilize" the point estimate. They also provided an implementation to produce an interval estimate with a 95% confidence level.

This confidence interval is constructed through a bootstrap-based pseudo algorithm. When the sample size is large (e.g., $n = 2000$), Liu *et al.* (2023)'s numerical studies show that the interval measure of the surrogate $R^2_{(S)}$ can approximate the nominal coverage probability.

It is also worth noting that Liu *et al.* (2023)'s method requires a full model. This paper will illustrate how to use existing tools, such as variable selection and model diagnostics, to initiate a full model. The full model is used to generate a common surrogate response S, which is then used to calculate surrogate $R^2_{(S)}$'s of whatever reduced models. We will demonstrate how to carry it out in a real data analysis presented in Section 5.

3. Main Functions of the SurrogateRsq package

We develop an R package **SurrogateRsq** for goodness-of-fit analysis of probit models. This package contains functions to provide (i) a point estimate of the surrogate R^2; (ii) an interval estimate of the surrogate R^2; (iii) an importance ranking of explanatory variables based on their contributions to the total surrogate R^2 of the full model; and (iv) other existing R^2 measures in the literature. In this section, we explicitly explain the inputs and outputs of these functions. In the next two sections, we will demonstrate the use of these functions through a recommended workflow and real data examples.

1. `surr_rsq`: a function for producing a point estimate of the surrogate $R^2_{(S)}$ for a user-specified model. It requires three inputs: a reduced model, a full model, and a dataset. This function generates an S3 object of the class "`surr_rsq`". Other functions in this package can directly call this S3 object. The details of the three inputs are as follows:

    - `model`: a model to be evaluated for the goodness of fit. Our implementation supports a few popular classes of objects. They are the `probit` model from the `glm` function in the R core `stats` package, the ordered probit model generated from the `plor` function in the `MASS` package, `clm()` in the `ordinal` package, and `vglm()` in the `VGAM` package.
    - `full_model`: a full model initiated by the investigator. Liu *et al.* (2023)'s method requires a full model. In Sections 4 and 5, we discuss in detail how to initiate a full model.
    - `data`: a dataset containing a categorical response and explanatory variables.
    - `avg.num`: an optional input that specifies the numbers of simulations used in multiple sampling. The default value is 30. The surrogate $R^2_{(S)}$ is calculated using the simulated surrogate response S. A multiple-sampling scheme can be used to "stabilize" the point estimate of $R^2_{(S)}$ by using the average of multiple $R^2_{(S)}$'s values.

```
R> surr_rsq(model,
+           full_model,
+           data,
+           avg.num = 30)
```

2. `surr_rsq_ci`: a function for generating an interval measure of the surrogate R^2 with the designated confidence level. This interval accounts for and reflects the uncertainty in the R^2 statistic. This function requires three inputs:

- `object`: an object generated from the previous `surr_rsq` function.
- `alpha`: the value of `alpha` determines the confidence level of the interval, namely, $100(1-\alpha)\%$. The default value of `alpha` is 0.05.
- `B`: the number of bootstrap replications. The default value of `B` is 2000. The confidence interval is derived from a bootstrap distribution for $R^2_{(S)}$. See the section of "Inference by Multiple Sampling" in Liu *et al.* (2023).

```
R> surr_rsq_ci(object,
+              alpha = 0.05,
+              B     = 2000)
```

3. `surr_rsq_rank`: a function to give ranks of explanatory variables based on their contributions to the overall surrogate R^2. The rank is based on the variance contribution of each variable. Specifically, it calculates the reduction of the surrogate $R^2_{(S)}$ of the model that removes each variable one at a time. The rank is then determined according to the reduction, which indicates the importance of each variable relevant to others. In addition to the ranks, the output table includes the R^2 reduction and its percentage in reference to the total surrogate R^2 of the full model. The function requires two inputs: `object` and `data`. They are, respectively, a generated object from the `surr_rsq` function and the dataset. The optional `avg.num` argument is the same as the one in the `surr_rsq` function, and the option `var.set` is explained below.

   - `object`: an object generated from the previous `surr_rsq` function.
   - `var.set`: an optional argument that allows users to examine the contribution of a set of variables, as a whole, to the total surrogate R^2. If not specified, the function gives results for individual variables.

```
R> surr_rsq_rank(object,
+                data,
+                var.set,
+                avg.num = 30)
```

4. `rsq`: a function to produce other pseudo R^2 measures in the literature. They include the R^2's proposed by McFadden (1973), McKelvey and Zavoina (1975), Cox and Snell (1989), Nagelkerke (1991), and Tjur (2009). Liu *et al.* (2023) provided a comparison between the surrogate R^2 and those proposed by McFadden and McKelvey-Zavoina. The function `rsq` has an argument `which` for specifying which pseudo R^2 measure to use. The rest of the arguments are the same as those in the `surr_rsq` function.

   - `which`: an argument to specify which pseudo R^2 measure to calculate. The function `rsq` can produce the surrogate R^2, McFadden R^2, McKelvey-Zavoina R^2, CoxSnell R^2, Nagelkerke R^2, and Tjur R^2.

```
R> rsq(model,
+      full_model=NULL,
+      data,
```

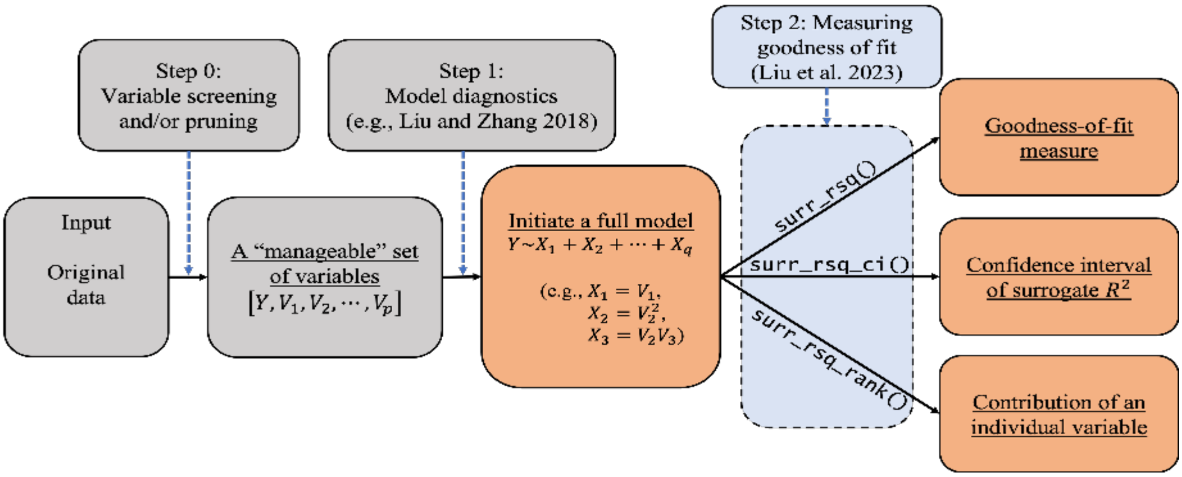

Figure 1: An illustration of the workflow for modeling categorical data.

```
+     which = c("Surrogate", "McFadden", "McKelveyZavoina",
+               "CoxSnell", "Nagelkerke", "Tjur"),
+     avg.num = 30, ...)
```

4. Using R packages for categorical data modeling: a workflow

In empirical studies, goodness-of-fit analysis should be used jointly with other statistical tools, such as variable screening/selection and model diagnostics, in the model-building and refining process. In this section, we discuss how to follow the workflow in Figure 1 to carry out statistical modeling for categorical data. We also discuss how to use the **SurrogateRsq** package with other existing R packages to implement this workflow. As Liu *et al.* (2023)'s method requires a full model, researchers and practitioners can also follow the process in Figure 1 to initiate a full model so as to facilitate goodness-of-fit analysis.

1. In `Step-0`, we can use the AIC/BIC/LASSO or any other variable selection methods deemed appropriate to trim or prune the set of explanatory variables to a "manageable" size (e.g., less than 20). The goal is to eliminate irrelevant variables so that researchers can better investigate the model structure and assessment. The variable selection techniques have been studied extensively in the literature. Specifically, one can implement (i) the best subset selection using the function `regsubsets()` in the **leaps** package; (ii) the forward/backward/stepwise selection using the function `step()`in the R core; (iii) the shrinkage methods including the (adaptive) LASSO in the **glmnet** package; (iv) the regularized ordinal regression model with an elastic net penalty in the **ordinalNet** package; and (v) the penalized regression models with minimax concave penalty (MCP) or smoothly clipped absolute deviation (SCAD) penalty in the **ncvreg** package (Tibshirani 1996; Zou and Hastie 2005; Zou 2006; Simon, Friedman, Hastie, and Tibshirani 2011; Wurm *et al.* 2021). When the dimension is ultrahigh, the sure independence screening method can be applied through the **SIS** package (Fan and Lv 2008). When the variables are grouped, one can apply the group selection methods

including the group lasso, group MCP, and group SCAD through the **grpreg** package (Breheny and Breheny 2014). In some cases, `Step-0` may be skipped if the experiment only involves a (small) set of controlled variables. In these cases, the controlled variables should be modeled regardless of statistical significance or predictive power. We limit our discussion here because our focus is on goodness-of-fit analysis.

2. In `Step-1`, we can use diagnostic tools to inspect the model passed from `Step-0`, adjust its functional form, and add additional elements if needed (e.g., higher-order or interaction terms). For categorical data, we can use the function `autoplot.resid()` in the **sure** package (Liu and Zhang 2018; Greenwell *et al.* 2018) to generate three types of diagnostic plots: residual Q-Q plot, residual-vs-covariate plot, and residual-vs-fitted plots. These plots can be used to visualize the discrepancy between the working model and the "true" model. Similar plots can be produced using the function `diagnostic.plot()` in the **PAsso** package (Zhu *et al.* 2020). These diagnostic plots give practitioners insights on how to refine the model by possibly transforming the regression form or adding higher-order terms. At the end of this diagnosing and refining process, we expect to have a **full model** ($\mathcal{M}_{full}$) for subsequent inferences including goodness-of-fit analysis.

3. In `Step-2`, we can use the functions developed in our **SurrogateRsq** package to examine the goodness of fit of the full model $\mathcal{M}_{full}$ and various reduced models of interest. Specifically, we can produce the point and interval estimates of the surrogate R^2 by using the functions `surr_rsq()` and `surr_rsq_ci()`. In addition, we can quantify the contribution of each individual variable to the overall surrogate R^2 by using the function `surr_rsq_rank()`. Based on the percentage contribution, the function `surr_rsq_rank()` also provides ranks of the explanatory variables to show their relative importance. In the following section, we will show in a case study how our package can help us understand the relative importance of explanatory variables and compare the results across different samples. The "comparability" across different samples and/or models is an appealing feature of the surrogate R^2, which will be discussed in detail along with the R implementation.

5. Analysis of the wine rating data: a demonstration

In this section, we demonstrate how to use our `SurrogateRsq` package, coupled with R packages for model selection and diagnostics, to carry out statistical analysis of the wine rating data. A critical problem in wine analysis is to understand how physicochemical properties of wines may influence humane tasting preferences (Cortez, Cerdeira, Almeida, Matos, and Reis 2009). For this purpose, Cortez *et al.* (2009) collected a data set that contains wine ratings for 1599 red wine samples and 4898 white wine samples. The response variable, wine ratings, is measured on an ordinal scale ranging from 0 (very bad) to 10 (excellent). The explanatory variables are 11 physicochemical features, including alcohol, sulphates, acidity, dioxide, pH, and others.

Our analysis of the wine rating data follows the workflow discussed in Section 4. Specifically, in Section 5.1, we initiate a full model using several R packages for variable selection and model diagnostics. In Section 5.2, we use our `SurrogateRsq` package to evaluate (i) the goodness-of-fit of the full model and several reduced models; (ii) the contribution of each individual variable to the overall R^2; and (iii) the difference between the red wine and white wine in terms of how physicochemical features may influence human tasting differently.

5.1. Initiating a full model using variable selection and model diagnostics

To start, we use the function `polr()` to fit a probit model to the red wine sample using all the 11 explanatory variables. This "naive" model has identified three explanatory variables are that insignificant; they are `fixed.acidity`, `citric.acid`, and `residual.sugar`.

```
R> library(SurrogateRsq)
R> library(MASS)
R> data("RedWine")
R> ### We remove an outlier where total.sulfur.dioxide>200.
R> RedWine2 <- subset(RedWine, total.sulfur.dioxide <= 200)
R> naive_formula <-
+   as.formula(quality ~ fixed.acidity + volatile.acidity + citric.acid +
+                        residual.sugar + chlorides + free.sulfur.dioxide +
+                        total.sulfur.dioxide + density + pH + sulphates +
+                        alcohol)
R> naive_model <- polr(formula = naive_formula,
+                      data    = RedWine2,
+                      method  = "probit")
R> summary(naive_model)
Call:
polr(formula = full_formula, data = RedWine2, method = "probit")

Coefficients:
                         Value Std. Error  t value
fixed.acidity         0.026476   0.028154   0.9404
volatile.acidity     -1.867959   0.213445  -8.7515
citric.acid          -0.336632   0.256198  -1.3140
residual.sugar        0.011032   0.020944   0.5267
chlorides            -3.234491   0.733213  -4.4114
free.sulfur.dioxide   0.010063   0.003829   2.6278
total.sulfur.dioxide -0.007198   0.001343  -5.3597
density              -6.678993   0.538393 -12.4054
pH                   -0.754044   0.277469  -2.7176
sulphates             1.589296   0.194509   8.1708
alcohol               0.480603   0.031945  15.0447

Intercepts:
    Value    Std. Error t value
3|4  -7.4023   0.5513   -13.4280
4|5  -6.5749   0.5483   -11.9915
5|6  -4.5379   0.5480    -8.2802
6|7  -2.9068   0.5530    -5.2563
7|8  -1.3617   0.5624    -2.4214
```

```
Residual Deviance: 3079.007
AIC: 3111.007
```

Variable selection

As the number of explanatory variables is small, we use the exhaustive search method to select variables.

```
R> model_exhau <- leaps::regsubsets(x     = quality ~ .,
+                                  data  = RedWine2,
+                                  nbest = 2,
+                                  nvmax = 11)
R> # We plot the exhaustive search selection results in Figure 2
R> plot(model_exhau)
```

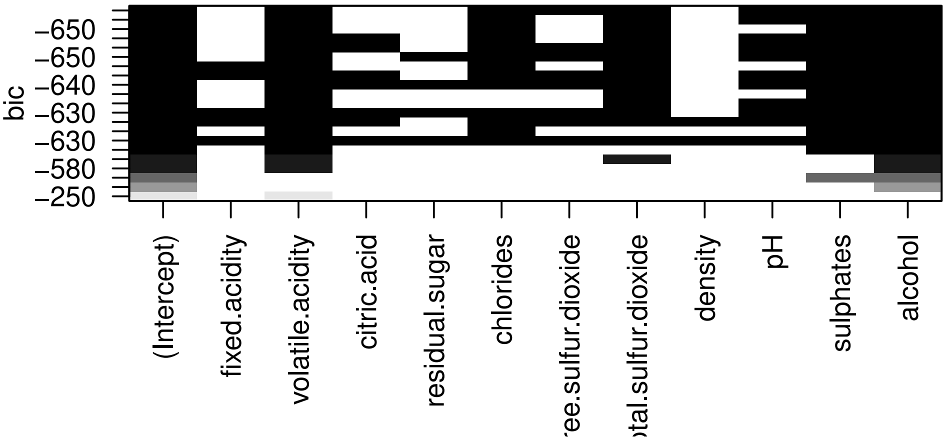

Figure 2: The selection results of exhaustive search method for the red wine analysis.

Figure 2 plots the exhaustive search selection results based on the BIC. Each row in the plot represents a model that has been trained with the variables highlighted in the black color. The top row is the selected model with the smallest BIC value. This model does not select `fixed.acidity`, `citric.acid`, `residual.sugar`, and `density`. Note that the first three are not significant. We will perform diagnostics on this model in the subsection that follows.

```
R> select_model <-
+   update(naive_model,
+          formula. =
+            ". ~ . - fixed.acidity - citric.acid - residual.sugar - density")
```

We remark that if the number of explanatory variables is (moderately) large, we can use the step-wise selection method or regularization methods (e.g., with an L1, elastic net, minimax concave, or SCAD penalty). Example code is attached in the supplementary materials.

Model diagnostics

We conduct diagnostics of the model with variables selected in the preview step. For this purpose, we use surrogate residuals (Liu and Zhang 2018), which can be implemented by the function `autoplot.resid()` in the package **sure** (Greenwell *et al.* 2018) or the function `diagnostic.plot()` in the package **PAsso** (Zhu *et al.* 2020). The code below produces residual-vs-covariate plots for the object `select_model` by specifying the `output = "covariate"`.

```
R> library(PAsso)
R> p_sulphates <-
+   diagnostic.plot(object     = select_model,
+                   output     = "covariate",
+                   x          = RedWine2$sulphates,
+                   xlab       = "sulphates")
```

Among all the residual-vs-covariate plots, we find that the residual-vs-`sulphates` plot in Figure 3(a) shows an inverted U-shape pattern, which suggests a missing quadratic term of `sulphates`. We update the model by adding a squared term $I($`sulphates^`$2)$ to the object `select_model` and run model diagnostics again using the code below. Figure 3(b) shows that the plot for `sulphates` still exhibits a nonlinear pattern. We therefore add a cubit term $I($`sulphates^`$3)$ to the model. The LOESS curve in the updated plot in Figure 3(c) turns out to be flat. We use this model as our **full model** ($\mathcal{M}_{full}$).

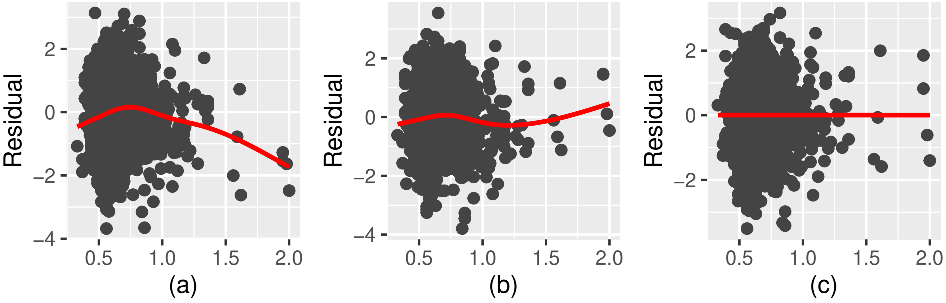

Figure 3: Plots of surrogate residual versus `sulphates` for (a) the model with a linear term of `sulphates`; (b) the model with a quadratic term of `sulphates`; and (c) the model with a cubit term of `sulphates`. The solid red curves are LOESS curves.

```
R> mod_add_square <-
+   update(select_model, formula. = ". ~ . + I(sulphates^2)")
R> p_sulphates2 <-
+   diagnostic.plot(object     = mod_add_square,
+                   output     = "covariate",
+                   x          = RedWine2$sulphates,
+                   xlab       = "sulphates")

R> mod_full <-
+   update(mod_add_square, formula. = ". ~ . + I(sulphates^3)")
R> p_sulphates3 <-
```

```
+   diagnostic.plot(object    = mod_full,
+                   output    = "covariate",
+                   x         = RedWine2$sulphates,
+                   xlab      = "sulphates")
R> grid.arrange(p_sulphates, p_sulphates2, p_sulphates3, ncol = 2)
```

Table 1: Model development for the red wine by variable selection and model diagnostics

	Dependent variable: quality			
Model	Naive	Selected	+ sulphates^2	+ sulphates^3 **full model** $\mathcal{M}_{full}$
fixed.acidity	0.026			
	(0.028)			
volatile.acidity	−1.868***	−1.722***	−1.534***	−1.491***
	(0.213)	(0.180)	(0.183)	(0.183)
citric.acid	−0.337			
	(0.256)			
residual.sugar	0.011			
	(0.021)			
chlorides	−3.234***	−3.488***	−2.965***	−2.604***
	(0.733)	(0.699)	(0.707)	(0.715)
free.sulfur.dioxide	0.010***	0.011***	0.010**	0.010**
	(0.004)	(0.004)	(0.004)	(0.004)
total.sulfur.dioxide	−0.007***	−0.008***	−0.007***	−0.007***
	(0.001)	(0.001)	(0.001)	(0.001)
density	−6.679***			
	(0.538)			
pH	−0.754***	−0.780***	−0.969***	−1.028***
	(0.277)	(0.205)	(0.208)	(0.209)
sulphates	1.589***	1.570***	5.937***	15.147***
	(0.195)	(0.193)	(0.678)	(2.591)
sulphates^2			−2.515***	−12.397***
			(0.374)	(2.707)
sulphates^3				3.092***
				(0.839)
alcohol	0.481***	0.479***	0.475***	0.472***
	(0.032)	(0.031)	(0.031)	(0.031)
Note:				*p<0.1; **p<0.05; ***p<0.01

Table 1 summarizes the model fitting results for the naive model and models progressively trained in the procedures of variable selection and model diagnostics. Compared to the naive model, the "Selected" column basically removes `density` and three non-significant variables, which results in a lower BIC value. The last two columns of Table 1 confirm the statistical significance of both the squared and cubit terms of `sulphates`, which are identified and added in the model diagnostics procedure. The model presented in the last column will be used as the **full model** $\mathcal{M}_{full}$ in our goodness-of-fit assessment in the next subsection.

5.2. Goodness-of-fit analysis and its extended utility

In this subsection, we use the our developed **SurrogateRsq** package to illustrate how to use the surrogate R^2 to (i) assess goodness-of-fit of the full model and reduced models; (ii) rank exploratory variables based on their contributions to R^2; and (iii) compare goodness of fit across multiple samples and/or models.

Surrogate R^2 for the full model

First of all, we use the function `surr_rsq` to calculate the surrogate R^2 of the full model $\mathcal{M}_{full}$ identified in the previous subsection. To do so, in the code below we set the arguments `model` and `full_model` to be the same as $\mathcal{M}_{full}$. We use 30 as the number of simulations for multiple sampling. The purpose of performing multiple sampling is to "stabilize" the point estimate of R^2 (Liu *et al.* 2023).

```
R> library(SurrogateRsq)
R> surr_obj_mod_full <-
+   surr_rsq(model      = mod_full,
+            full_model = mod_full,
+            data       = RedWine2,
+            avg.num    = 30)
R> print(surr_obj_mod_full$surr_rsq,
+       digits = 3)
[1] 0.439
```

This function provides a point estimate of the surrogate R^2 of the full model. The value 0.439 implies 43.9% of the variance of the surrogate response S can be explained by the seven explanatory variables and two nonlinear terms of `sulphates`.

Surrogate R^2 for a reduced model

We can also use the same function `surr_rsq` to calculate the surrogate R^2 of a reduce model. For example, to evaluate the goodness of fit of the model without high-order terms of `sulphates`, we simply need to change the `model` argument to be the reduced model `select_model` as shown in the code below. The specification of the full model is still required in the code, and such a full model should be common to all the reduced models to be compared. This is a way to eliminate the non-monotonicity issue as seen in Mckelvey-Zavoina's R^2_{MZ} (Liu *et al.* 2023).

```
R> surr_obj_lm <- surr_rsq(model      = select_model,
+                          full_model = mod_full,
+                          data       = RedWine2,
+                          avg.num    = 30)
R>  print(surr_obj_lm$surr_rsq,
+        digit = 3)
[1] 0.41
```

The result shows that the surrogate R^2 has been reduced to 0.41 if the squared and cubit terms of `sulphates` are removed from the model. This means that the high-order terms of `sulphates` constitutes 6.60% of the total surrogate R^2.

Confidence interval for the surrogate R^2

The package **SurrogateRsq** allows us to produce a confidence interval for the surrogate R^2 using the function `surr_rsq_ci`. This function can directly use the object `surr_obj_mod_full` created earlier as the input of the `object` argument. In the code below, we set the significance level `alpha = 0.05` to produce a 95% confidence interval and the number of bootstrap repetitions to be 2000. The output is a table with the lower and upper bounds of the confidence interval. For the full model $\mathcal{M}_{full}$, the 95% confidence interval of the surrogate R^2 is $[0.435, 0.441]$. The tightness of this interval implies that the uncertainty of the R^2 inference is low.

```
R> full_mod_rsq_ci <-
+   surr_rsq_ci(object = surr_obj_mod_full,
+               alpha  = 0.05,
+               B      = 2000)
R> full_mod_rsq_ci
                    Lower  Upper
Percentile          2.50% 97.50%
Confidence Interval 0.435  0.441
```

Importance ranking of explanatory variables

We apply the function `surr_rsq_rank()` to examine the contribution of each individual variable to the overall surrogate R^2, which in turn produces a table of importance ranking. In the code below, we set the `object` argument as the object `surr_obj_mod_full` created earlier to examine the relative contribution of the variables in the full model. The output table shows (i) the surrogate R^2 for the model that removes an explanatory variable one at a time; (ii) the reduction of the R^2 after removing such a variable; (iii) the percentage contribution of this variable to the total surrogate R^2; and (iv) the rank of the variable by its percentage contribution. In the table below, we observe that the variable `alcohol` is ranked at the top as it explains 25.80% of the total surrogate R^2. It is followed by `volatile.acidity` (7.12%), `total.sulfur.dioxide` (3.52%), and `sulphates` (3.13%). The rest of the explanatory variables contribute less than 3% to the total surrogate R^2.

```
R> Rank_table_mod_full <-
+   surr_rsq_rank(object  = surr_obj_mod_full,
+                 data    = RedWine2,
+                 avg.num = 30)
R> print(Rank_table_mod_full, digits = 3)
     Removed Variable SurrogateRsq Reduction Contribution Ranking
```

```
           alcohol       0.325     0.113       25.80%       1
  volatile.acidity       0.407     0.031        7.12%       2
total.sulfur.dioxide     0.423     0.015        3.52%       3
         sulphates       0.425     0.014        3.13%       4
                pH       0.426     0.012        2.78%       5
    I(sulphates^2)       0.429     0.009        2.11%       6
         chlorides       0.433     0.005        1.21%       7
    I(sulphates^3)       0.433     0.005        1.17%       8
 free.sulfur.dioxide     0.434     0.004        0.96%       9
-----------------------------------------------------------------------
The total surrogate R-squared of the full model is:
[1] 0.439
```

In the ranking table above, the contributions of `sulphates` and its higher order terms `sulphates`^{2} and `sulphates`^{3} to the surrogate R^2 are evaluated separately. This is the default setting of the function `surr_rsq_rank()` if the optional argument `var_set` is not specified. If it is of interest to evaluate the factor sulphates as a whole, the function `surr_rsq_rank()` allows us to group `sulphates`, `sulphates`^{2}, and `sulphates`^{3} by using the optional argument `var_set`. For example, in the code below we create a list of two groups: one group contains all terms of `sulphates` and the second group only contains higher order terms of `sulphates`.

```
R> var_set <- list(c("sulphates", "I(sulphates^2)", "I(sulphates^3)"),
+                  c("I(sulphates^2)", "I(sulphates^3)"))
R> Rank_table2 <-
+   surr_rsq_rank(object  = surr_obj_mod_full,
+                 data    = RedWine2,
+                 var.set = var_set,
+                 avg.num = 30)
R> print(Rank_table2, digits = 2)
                       Removed Variable SurrRsq Redu... Cont... Ranking
 sulphates+I(sulphates^2)+I(sulphates^3)   0.378   0.061   13.82%    1
           I(sulphates^2)+I(sulphates^3)   0.411   0.027    6.19%    2
-----------------------------------------------------------------------
The total surrogate R-squared of the full model is:
[1] 0.439
```

The output table above shows that the factor `sulphates` in fact contributes 13.82% to the total surrogate R^2 if its linear, squared, and cubic terms are considered altogether. This percentage contribution is much higher than that when only the linear term of `sulphates` was evaluated (3.13%). By this result, `sulphates` is lifted to the second place in terms of its relative contribution to the total surrogate R^2. The output table also shows that if we only consider the higher order terms of `sulphates`, the percentage contribution is 6.19%, which is higher than any other individual variables except `volatile.acidity` (7.12%). This is another piece of evidence that can supports the inclusion of the squared and cubit terms of `sulphates` in the full model.

Comparability of the surrogate R^2 across different samples and models

One of the motives of Liu *et al.* (2023) is to find an R^2 measure so that we can compare goodness of fit across different models (e.g., linear, binary, or ordinal regression models) and/or samples that address the same or similar scientific/business question. We use the wine data in Cortez *et al.* (2009) to demonstrate that the surrogate R^2 enables this comparability, which may lead to new insights into decision-making. Cortez *et al.* (2009)'s data include 1599 red wine samples and 4898 white wine samples. Although the same rating scale (i.e., from 0 to 10) were offered to wine experts, in the red wine sample only 6 rating categories (3 to 8) were observed whereas in the white wine sample 7 rating categories (3 to 9) were observed. As a result, the ordered probit models fitted to red and white wine samples have different number of intercept parameters. In addition, after conducting the same analysis but to the white wine sample (using similar code as presented before), we find out that the set of selected variables is not the same. The 7 selected variables are `alcohol`, `volatile.acidity`, `residual.sugar`, `free.sulfur.dioxide`, `sulphates`, `fixed.acidity`, and `pH`. As a result, the ordered probit models fitted to red and white wine samples have different number of slope parameters as well. Given the differences between the samples and models, the surrogate R^2, nevertheless, enables us to compare goodness-of-fit measures across the board. Table 2 summarizes the result obtained using our developed package **SurrogateRsq**.

Table 2: Percentage contributions and ranks of the physicochemical variables in the analysis of the red wine and white wine samples.

	Red wine data		*White wine data*	
	Surrogate R^2=0.439		Surrogate R^2=0.307	
Variable	Contribution	Ranking	Contribution	Ranking
alcohol	25.80%	1	77.16%	1
sulphates (& higher-order terms)	13.82%	2	0.51%	5
volatile.acidity	7.12%	3	20.39%	2
total.sulfur.dioxide	3.52%	4		
pH	2.78%	5	0.06%	7
chlorides	1.21%	6		
free.sulfur.dioxide	0.96%	7	1.42%	4
residual.sugar			5.34%	3
fixed.acidity			0.32%	6
sulphates^2 & sulphates^3	6.19%			

By comparing the result in the two panels (red versus white wine) of Table 2, we can make the following conclusions: (i) the same set of measured physicochemical features in the experiment of Cortez *et al.* (2009) has greater explanatory power for red wine (43.9% versus 30.7%); (ii) the ranking of explanatory variables is different for the two types of wine with only one exception which is `alcohol` (top for both); and (iii) the percentage contributions of each variable differ significantly in magnitude for red versus white wine (e.g., `alcohol`, 25.80% versus 77.16%; `sulphates`, 13.82% versus 0.51%; `volatile.acidity`, 7.12% versus 20.39%). These insights drawn from our goodness-of-fit analysis may be useful to help us understand how physicochemical features influence wine ratings and how the influence may be different depending on the type of wine. The percentage contributions and ranking of physicochemical features may be used to guild or even devise the wine making process.

6. Summary

In this paper, we have developed the R package **SurrogateRsq** for categorical data goodness-of-fit analysis using the surrogate R^2. The package applies to probit/logistic regression models, and it is compatible with commonly used R packages for binary and ordinal data analysis. With **SurrogateRsq**, we are able to obtain point estimate and the interval estimates of the surrogate R^2. An importance ranking table for all explanatory variables can be produced as well. These new features can be used in conjunction with other R packages developed for variable selection and model diagnostics. This "whole-analysis" is summarized in a workflow diagram, which can be followed in practice for categorical data analysis. To examine the utility of this package in real data analysis, we have used a wine rating dataset as an example and provided sample code. In addition, we have used the package **SurrogateRsq** to demonstate that the surrogate R^2 allows us to compare different models trained from the red wine sample and white wine sample. The comparison has led to new findings and insights that deepen our understanding of how physicochemical features influence the wine quality. The result suggests that our package can be used in a similar way to analyze multiple studies (and/or models) that address the same or similar scientific or business question.

References

Breheny P (2013). "ncvreg: Regularization paths for scad-and mcp-penalized regression models." *R package version*, **2**, 6–0. URL `https://pbreheny.github.io/ncvreg/`.

Breheny P, Breheny MP (2014). "Package 'grpreg'." URL `https://pbreheny.github.io/grpreg/`.

Cheng C, Wang R, Zhang H (2021). "Surrogate Residuals for Discrete Choice Models." *Journal of Computational and Graphical Statistics*, **30**(1), 67–77. `doi:https://doi.org/10.1080/10618600.2020.1775618`.

Christensen RHB (2019). "ordinal—Regression Models for Ordinal Data." R package version 2019.12-10. https://CRAN.R-project.org/package=ordinal, URL `http://www2.uaem.mx/r-mirror/web/packages/ordinal/`.

Cortez P, Cerdeira A, Almeida F, Matos T, Reis J (2009). "Modeling Wine Preferences by Data Mining from Physicochemical Properties." *Decision Support Systems*, **47**(4), 547–553. `doi:10.1016/j.dss.2009.05.016`.

Cox D, Snell E (1989). *Analysis of Binary Data*, volume 32. `doi:https://doi.org/10.1201/9781315137391`.

Cox DR, Wermuth N (1992). "A Comment on the Coefficient of Determination for Binary Responses." *The American Statistician*, **46**(1), 1–4. `doi:10.1080/00031305.1992.10475836`.

Efron B (1978). "Regression and ANOVA with Zero-one Data: Measures of Residual Variation." *Journal of the American Statistical Association*, **73**(361), 113–121. doi:10.1080/01621459.1978.10480013.

Fan J, Lv J (2008). "Sure Independence Screening for Ultrahigh Dimensional Feature Space." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **70**(5), 849–911. doi:10.1111/j.1467-9868.2008.00674.x@10.1111/(ISSN)1467-9868.TOP_SERIES_B_RESEARCH.

Friedman J, Hastie T, Tibshirani R (2010). "Regularization paths for generalized linear models via coordinate descent." *Journal of Statistical Software*, **33**(1), 1. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2929880/.

Greenwell BM, McCarthy AJ, Boehmke BC, Liu D (2018). "Residuals and Diagnostics for Binary and Ordinal Regression Models: An Introduction to the sure Package." *The R Journal*, **10**(1), 381–394. doi:10.32614/RJ-2018-004.

Hagle TM, Mitchell GE (1992). "Goodness-of-Fit Measures for Probit and Logit." *American Journal of Political Science*, **36**(3), 762–784. doi:10.2307/2111590.

Hu B, Shao J, Palta M (2006). "Pseudo-R^2 in Logistic Regression Model." *Statistica Sinica*, **16**(3), 847–860. URL https://www.jstor.org/stable/24307577.

Laitila T (1993). "A Pseudo-R^2 Measure for Limited and Qualitative Dependent Variable Models." *Journal of Econometrics*, **56**(3), 341–356. URL https://doi.org/10.1016/0304-4076(93)90125-O.

Li S, Zhu X, Chen Y, Liu D (2021). "PAsso: an R Package for Assessing Partial Association between Ordinal Variables." *The R Journal*, **13**(2), 135. doi:10.32614/RJ-2021-088.

Liu D, Li S, Yu Y, Moustaki I (2021). "Assessing Partial Association Between Ordinal Variables: Quantification, Visualization, and Hypothesis Testing." *Journal of the American Statistical Association*, **116**(534), 955–968. doi:10.1080/01621459.2020.1796394.

Liu D, Zhang H (2018). "Residuals and Diagnostics for Ordinal Regression Models: A Surrogate Approach." *Journal of the American Statistical Association*, **113**(522), 845–854. doi:10.1080/01621459.2017.1292915.

Liu D, Zhu X, Greenwell B, Lin Z (2023). "A new goodness-of-fit measure for probit models: Surrogate R2." *British Journal of Mathematical and Statistical Psychology*, **76**(1), 192–210. URL https://doi.org/10.1111/bmsp.12289.

Liu I, Agresti A (2005). "The Analysis of Ordered Categorical Data: An Overview and a Survey of Recent Developments (with discussion)." *Test*, **14**(1), 1–73. doi:https://doi.org/10.1007/BF02595397.

Lumley T, Lumley MT (2013). "Package 'leaps'." *Regression subset selection. Thomas Lumley Based on Fortran Code by Alan Miller. Available online: http://CRAN. R-project. org/package= leaps (Accessed on 18 March 2018).* URL https://cran.r-project.org/web/packages/leaps/index.html.

McFadden D (1973). "Conditional Logit Analysis of Qualitative Choice Behavior." In P Zarembka (ed.), *Frontiers in Econometrics*, pp. 105–142. URL https://eml.berkeley.edu/reprints/mcfadden/zarembka.pdf.

McKelvey RD, Zavoina W (1975). "A Statistical Model for the Analysis of Ordinal Level Dependent Variables." *Journal of Mathematical Sociology*, **4**(1), 103–120. URL https://doi.org/10.1080/0022250X.1975.9989847.

Nagelkerke NJ (1991). "A Note on a General Definition of the Coefficient of Determination." *Biometrika*, **78**(3), 691–692. doi:https://doi.org/10.1093/biomet/78.3.691.

Ripley B, Venables B, Bates DM, Hornik K, Gebhardt A, Firth D, Ripley MB (2013). "Package 'mass'." *CRAN R*, **538**, 113–120. URL http://www.stats.ox.ac.uk/pub/MASS4/.

Saldana DF, Feng Y (2018). "SIS: An R package for sure independence screening in ultrahigh-dimensional statistical models." *Journal of Statistical Software*, **83**, 1–25. doi:10.18637/jss.v083.i02.

Simon N, Friedman J, Hastie T, Tibshirani R (2011). "Regularization paths for Cox's proportional hazards model via coordinate descent." *Journal of Statistical Software*, **39**(5), 1. doi:10.18637/jss.v039.i05.

Tibshirani R (1996). "Regression shrinkage and selection via the lasso." *Journal of the Royal Statistical Society: Series B (Methodological)*, **58**(1), 267–288. doi:https://doi.org/10.1111/j.2517-6161.1996.tb02080.x.

Tjur T (2009). "Coefficients of Determination in Logistic Regression Models—A New Proposal: The Coefficient of Discrimination." *The American Statistician*, **63**(4), 366–372. URL https://doi.org/10.1198/tast.2009.08210.

Veall MR, Zimmermann KF (1996). "Pseudo-R^2 Measures for Some Common Limited Dependent Variable Models." *Journal of Economic Surveys*, **10**(3), 241–259. doi:10.1111/j.1467-6419.1996.tb00013.x.

Wurm MJ, Rathouz PJ, Hanlon BM (2021). "Regularized Ordinal Regression and the ordinalNet R Package." *Journal of Statistical Software*, **99**, 1–42. ISSN 1548-7660. doi:10.18637/jss.v099.i06. URL https://doi.org/10.18637/jss.v099.i06.

Yee TW, *et al.* (2010). "The VGAM Package for Categorical Data Analysis." *Journal of Statistical Software*, **32**(10), 1–34. doi:10.18637/jss.v032.i10.

Zheng B, Agresti A (2000). "Summarizing the Predictive Power of a Generalized Linear Model." *Statistics in Medicine*, **19**(13), 1771–1781. URL https://doi.org/10.1002/1097-0258(20000715)19:13<1771::AID-SIM485>3.0.CO;2-P.

Zhu X, Li S, Chen Y, Liu D (2020). "PAsso: an R Package for Assessing Partial Association between Ordinal Variables." *R package Version 0.1.9*. URL https://xiaoruizhu.github.io/PAsso/.

Zou H (2006). "The Adaptive Lasso and Its Oracle Properties." *Journal of the American Statistical Association*, **101**(476), 1418–1429. doi:10.1198/016214506000000735.

Zou H, Hastie T (2005). "Regularization and Variable Selection via the Elastic Net." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **67**(2), 301–320. `doi:10.1111/j.1467-9868.2005.00503.x`.

7. Supplementary materials

7.1. Supplementary code for Section 5.1

In this section, we provide sample code for variable selection using the step-wise selection method and the regularization method with an elastic net penalty.

The step-wise selection methods starts with a null model (`null_model`) with an intercept only. The largest model we specify is the "naive model" with all explanatory variables. The result below shows that this method selects the same variables as the exhaustive search method.

```
R> null_model <- polr(quality ~ 1, data = RedWine2, method = "probit")
R> model_stepwise <- step(object    = null_model,
+                         scope     = list(lower = null_model, upper = naive_model),
+                         direction = 'both',
+                         trace     = 0)
R> results <- coef(model_stepwise)
R> # Print out the excluded covariates:
R> names(RedWine2[,-1])[! names(RedWine2[,-1]) %in% names(results)]

[1] "fixed.acidity" "citric.acid" "residual.sugar"  "density"
```

We also use the function `ordinalNet()` in the R package **ordinalNet** to fit a cumulative probit model with an elastic net penalty. The result below shows it only excludes a single variable which is `density`.

```
R> library(ordinalNet)
R> x <- as.matrix(RedWine2[ , !names(RedWine2) %in% c("quality")])
R> model_Net <- ordinalNet(x       = x,
+                          y       = RedWine2$quality,
+                          family  = "cumulative",
+                          link.   = "probit",
+                          nLambda = 20)
R> results <- coef(model_Net, matrix=TRUE)[-1,1]
R> # Print out the excluded covariates:
R> names(results[results == 0])
[1] "density"
```

7.2. Supplementary Figure for Section 5.1

The figure below contains diagnotic plots for the full model developed in Section 5.1 after performing variable selection and model diagnostics.

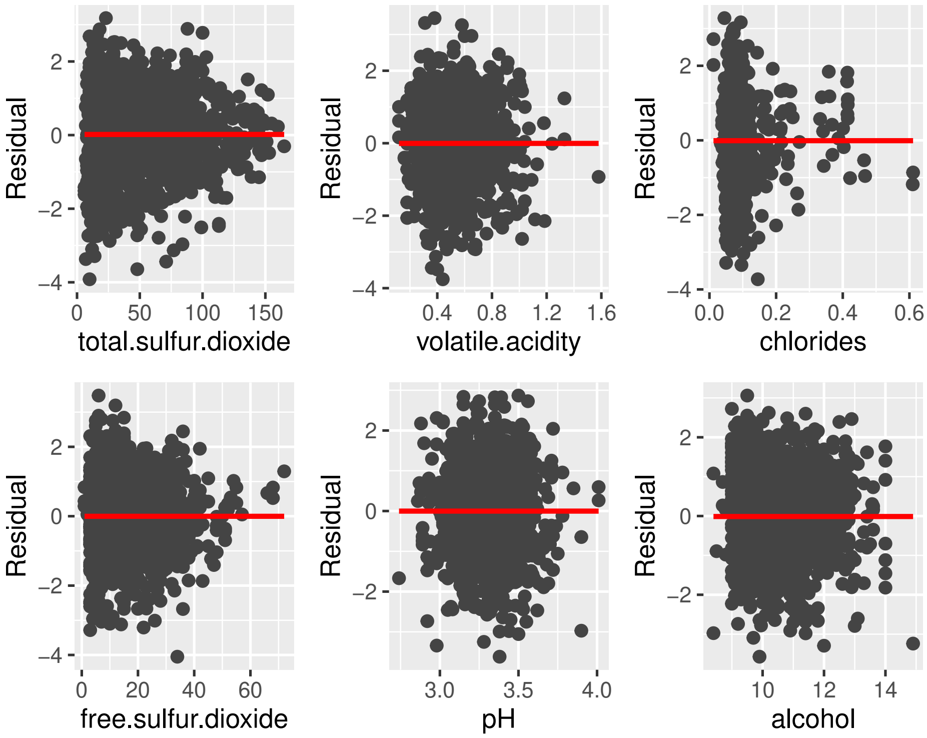

Figure 4: Plots of surrogate residuals versus each of the explanatory variables for the full model after adding the squared and cubic terms of sulphates.

Affiliation:

Xiaorui Zhu
Towson University
Department of Business Analytics & Technology Management
College of Business & Economics
Stephens Hall 301 M
Towson University
E-mail: xzhu@towson.edu
URL: https://homepages.uc.edu/~zhuxr/

Zewei Lin
University of Cincinnati
Department of Operation, Business Analytics, and Information Systems
Lindner College of Business
University of Cincinnati
E-mail: linzw@mail.uc.edu

Dungang Liu
University of Cincinnati
Department of Operation, Business Analytics, and Information Systems
Lindner College of Business
University of Cincinnati
E-mail: liudg@ucmail.uc.edu

Brandon Greenwell
84.51° and University of Cincinnati
E-mail: greenwell.brandon@gmail.com

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
MMMMMM YYYY, Volume VV, Issue II *Submitted:* yyyy-mm-dd
doi:10.18637/jss.v000.i00 *Accepted:* yyyy-mm-dd