Hacettepe Journal of Mathematics and Statistics
Volume 38 (2) (2009), 199 – 216

TESTING BINARY PARAMETRIC MODELS AGAINST THEIR SEMI-PARAMETRIC ALTERNATIVES USING COMMANDS WRITTEN IN VERSION 4.8 OF THE XploRe PACKAGE

Özge Akkuş* † and Hüseyin Tatlıdil‡

Received 12 : 11 : 2008 : Accepted 12 : 06 : 2009

Abstract

The aim of this study is to introduce the commands we wrote for testing the parametric logit and probit models against their semiparametric alternatives in the windows based version 4.8 of the XploRe package, and to show their applicability by using an artificial data set. This study extends the study of I. Proença and A. Werwatz (*Comparing Parametric and Semiparametric Binary Response Models*, Sonderforschungsbereich **373** 2000-20, Humboldt Universitaet, Berlin, 1994) in which the code was written in the old MsDOS format of XploRe for the parametric logit model, and only for the model with continuous explanatory variables. Here the parametric probit model and the mixed type of the explanatory variables (continuous-discrete) are also discussed, and the new XploRe commands generated for these types of model. Uniform Confidence Band limits have been used as the testing criteria.

Keywords: Semiparametric model, Density weighted average derivative estimator, Uniform confidence band, Probit model, Logit model, XploRe.

2000 AMS Classification: Primary: 62 J 99 Secondary: 62 J 12.

*Department of Statistics, Muğla University, Muğla, Turkey. E-mail: ozge.akkus@mu.edu.tr
†Corresponding Author
‡Department of Statistics, Hacettepe University, 06800 Beytepe, Ankara, Turkey.
E-mail: tatlidil@hacettepe.edu.tr

1. Introduction

Testing the validity of model assumptions in statistical modeling is one of the most important points to be taken into consideration by researchers. The validity test of the assumptions related to the error term is generally ignored in discrete dependent variable models.

The two most widely used models for binary dependent variables are the parametric probit model based on a normally distributed error term and the parametric logit model that assumes a logistic distribution for the error term. Biased estimates and very misleading results are obtained when the model assumptions are violated.

The use of semiparametric methods may be seen as a solution to this problem. No other assumption is required in these models beyond the linear index restriction on the explanatory variables. The main problem here is the difficulty of application and interpretation, compared with the parametric alternatives. Therefore, the validity of the parametric model assumptions should be tested before the analysis.

In this study, Uniform Confidence Band (UCB) limits were used as a testing criteria. In the event that the parametric model is true, there will be no need to use the semiparametric alternative and take into consideration its complicated structure.

2. The theoretical background

Most research fields of applied Econometrics and Statistics focus on the estimation of the conditional mean function denoted by $E(Y/X = x)$. The dependent variable Y may be continuous or binary. If it is binary, the conditional mean function gives the probability of observations belonging to category "1" coded in the dependent variable. The model is generally defined as:

$$E(Y/X = x) = P\left[Y = 1/X = x\right], \tag{1}$$

where X represents the vector of explanatory variables.

As mentioned above, the two popular approaches to model estimation are the fully parametric approach and the semiparametric approach.

2.1. The parametric approach. In the parametric approach for the model given by Eq. (1), there are a finite number of parameters (finite number of estimates of β) and the linear index restriction ($X^T\beta$) is accepted:

$$\mathrm{E}(Y/X = x) = \mathrm{P}\left[Y = 1/X = x\right] = G(X^T\beta) \tag{2}$$

Here G is a known function that represents the distribution of the error term. The name and the parameters of the distribution are also known. As a result, a probability expression is obtained related to the X values. Because of the linear index assumption ($X^T\beta$), the functional form of the explanatory variables is known and this approach is called the "parametric approach".

The parametric probit model is obtained by assuming a normally distributed error term $[G(\,\cdot\,) = \Phi(\,\cdot\,)]$. The model is defined as,

$$\mathrm{E}(Y/X = x) = \mathrm{P}\left[Y = 1/X = x\right] = \Phi(X^T\beta), \tag{3}$$

where Φ represents the standard cumulative normal distribution function.

The parametric logit model is obtained by assuming the logistic distribution for the error term of the model $[G(\,\cdot\,) = \Lambda(\,\cdot\,)]$. This model is defined as,

$$\mathrm{E}(Y/X = x) = \mathrm{P}\left[Y = 1/X = x\right] = \Lambda(X^T\beta) = \frac{\exp(X^T\beta)}{1 + \exp(X^T\beta)}. \tag{4}$$

The model parameters (β's) are estimated by the Maximum Likelihood Estimation Technique (MLE) in either model [1, 12, 14].

2.2. The semiparametric approach. In the semiparametric approach, G is an unknown function (denoted by g) and must be estimated by the nonparametric regression of Y on the estimated linear index $x^T\hat{\beta}$. Similar to the parametric model, the linear index restriction is still valid here. However, estimation methods for the βs' differ considerably from the parametric alternatives. The model expression is given as follows:

$$\mathrm{E}(Y/X = x) = \mathrm{P}\left[Y = 1/X = x\right] = g(X^T\beta). \tag{5}$$

Various methods have been developed for the estimation of the β's. Ichimura [9] proposed the use of the semiparametric least square estimator of β. Klein and Spady [10] developed a quasi-maximum-likelihood estimator. The main disadvantages of these estimators are the computational difficulty and the requirement of solving nonlinear optimization problems iteratively. Powell, Stock and Stoker [13] developed an estimator based on the Average Derivatives (ADE). The distribution assumption is not required for the dependent variable Y and the resulting estimator is a "Direct Estimator" which is not iterative. Its only disadvantage is that it can only be applied to continuous explanatory variables because it has to satisfy a differentiability condition [9, 10, 13].

2.2.1. *The density weighted average derivative estimator of the index parameters.* Assume that X is a continuously distributed random vector and that G is a differentiable function required for the identifiability of β. Under these assumptions,

$$\frac{\partial \mathrm{E}(Y/x)}{\partial x} = \beta G^{'}(X^T\beta) \tag{6}$$

can be derived. Additionally, for any restricted and continuous function W, we have

$$\mathrm{E}\left[W(X)\frac{\partial \mathrm{E}(Y/X)}{\partial x}\right] = \beta \mathrm{E}\left[W(X)G'(X^T\beta)\right] \tag{7}$$

The left side of Eq. (7) is called the *ADE with weight function* W. Eq. (7) shows that the weighted average derivative of $E\left(Y/x\right)$ is proportional to β. Because of the requirement of scale normalization, β is only defined according to the scale and any weighted average derivative of $\mathrm{E}(Y/x)$ is equal to β. Therefore, only estimating the left side of Eq. (7) is adequate for the estimation of β.

Dividing each component on the left side of Eq. (7) by the first component, the scale normalization of $\beta_1 = 1$ can be achieved in the semiparametric approach. The left side of Eq. (7) can be estimated by replacing the kernel estimator of $\frac{\partial \mathrm{E}(Y/X)}{\partial x}$ and the sample mean for the population expected value $[\mathrm{E}(\,\cdot\,)]$.

2.1. Theorem. *Let $p(\,\cdot\,)$ be the probability density function of X and $W(x) = p(x)$. Then the left side of Eq. (7) can be written as follows.*

$$\mathrm{E}\left[W(x)\frac{\partial \mathrm{E}(Y/X)}{\partial x}\right] = \mathrm{E}\left[p(X)\frac{\partial \mathrm{E}(Y/X)}{\partial x}\right] = \int \frac{\partial \mathrm{E}(Y/x)}{\partial x} p(x)^2\, dx. \tag{8}$$

In this case, δ is defined as $\delta = \mathrm{E}\left[W(X)\frac{\partial \mathrm{E}(Y/X)}{\partial x}\right]$. An efficient estimator of δ can be obtained by replacing p with a nonparametric estimator of it and replacing the expectation operator (E) with the sample mean. The estimator of δ is given as,

$$\delta_n = -\frac{2}{n}\sum_{i=1}^{n} Y_i \frac{\partial p_{ni}(x_i)}{\partial x}, \tag{9}$$

where $\{Y_i, X_i;\ i = 1, \ldots, n\}$ denotes the sample values of the observation "i" and $p_{ni}(x_i)$ is the estimator of the joint probability density function $p(X_i)$. Since, the joint probability

density function of X is used as the weight function, the resulting estimator δ_n is called the "Density Weighted Average Derivative Estimator" (DWADE).

The kernel estimation of the density function of $p_{ni}(x_i)$ is given as,

$$p_{ni}(x) = \frac{1}{n-1}\sum_{\substack{j=1\\ j\neq i}} \left(\frac{1}{h_n}\right)^k K\left(\frac{x-X_j}{h_n}\right), \tag{10}$$

where k denotes the dimension of X, K is a multivariate kernel function with k-dimensional component and $\{h_n\}$ is the series of bandwidth parameters. The formulation of $\frac{\partial p_{ni}(x)}{\partial x}$ is given as follows.

$$\begin{aligned}\frac{\partial p_{ni}(x)}{\partial x} &= \frac{1}{n-1}\sum_{\substack{j=1\\ j\neq i}}^{n} \left(\frac{1}{h_n}\right)^k K'\left(\frac{x-X_j}{h_n}\right)\left(\frac{1}{h_n}\right) \\ &= \frac{1}{n-1}\sum_{\substack{j=1\\ j\neq i}}^{n} \left(\frac{1}{h_n}\right)^{k+1} K'\left(\frac{x-X_j}{h_n}\right).\end{aligned} \tag{11}$$

Here, $K'$is the first order derivative of K (gradient vector). Replacing Eq. (11) in Eq. (9), the DWADE estimator is obtained [5, 13] as follows:

$$\delta_n = -\frac{2}{n\,(n-1)}\sum_{i=1}^{n}\sum_{\substack{j=1\\ j\neq i}}^{n} \left(\frac{1}{h_n}\right)^{k+1} K'\left(\frac{X_i-X_j}{h_n}\right) Y_i \tag{12}$$

2.2.2. *The estimation procedure of β's in the model with mixed explanatory variable.* In this model, discrete and continuous variables are shown by Z and X, respectively. The conditional expectation is given as,

$$\mathrm{E}(Y/X=x, Z=z) = g(X^T\beta + Z^T\alpha), \tag{13}$$

where β and α are vectors of parameters. Ichimura [9], Klein and Spady [10] and Manski [11] proved that at least one continuous explanatory variable had to be included in the model to achieve the identifiability of the parameters β and α. The first component of the vector of the continuous variables is set to "1" for this reason. The parameter β can be estimated using existing methods given in subsection 2.2. "DWADE" is used in this study.

Horowitz and Hardle [7] developed an estimator for the parameter α. The horizontal distance between $g(v+z^{(i)}\alpha)$ and $g(v+z^{(1)}\alpha)$, $(i=2,\ldots,M)$ is used for this estimator. Here, $S_z \equiv \{z^{(i)} : i=1,\ldots,M\}$ define the discrete random variable Z. They assumed that $g(v+z\alpha)$ satisfies a weak monotonicity condition. They also assumed that there are finite numbers v_0, v_1, c_0 and c_1 such that $v_0 < v_1$, $c_0 < c_1$, $g(v+z\alpha) < c_0$ for each $z \in S_z$ if $v < v_0$ and $g(v+z\alpha) > c_1$ for each $z \in S_z$ if $v > v_1$. The complex structure of the estimator is defined clearly in the study of Horowitz and Hardle [7]. Only the determination of the scalars c_0 and c_1 is required in the commands in XploRe. To achieve this, the data is graphed on each level of the discrete variable and the interval where the monotonicity condition is satisfied is determined [6, 7, 9, 10, 11].

2.2.3. *The optimal bandwidth selection problem.* The nonparametric regression method is used in the estimation of the link function g and the bandwidth (h) selection problem arises at this point. A specific method that gives the optimal bandwidth value has not yet been determined. The Least Square Cross-Validation (CV) method given in Eq. (14)

is used here because of its simple mathematical structure. The optimal h is obtained by minimizing the CV function.

$$\text{CV}(h) = \frac{1}{n}\sum_{i=1}^{n}\left[Y_i - \frac{\sum_{j\neq i}^{n} Y_j\, K_h\,(X_i - X_j)}{\sum_{j\neq i}^{n} K_h\,(X_i - X_j)})\right]^2 \tag{14}$$

In Eq. (14), K is a kernel function, Y is the observed dependent variable values and n is the sample size [3, 6].

In this study we firstly wrote the XploRe commands for the estimation of the β's in the semiparametric model estimation on the basis of the DWADE estimator by taking into consideration the advantages discussed in Subsection 2.2.1. Then we extended these commands to the case of both continuous and discrete explanatory variable models.

3. The uniform confidence bands procedure

UCB were used for testing the validity of the parametric logit and probit models. The UCB procedure generally includes the following steps.

- Firstly, the linear index function $X^T\beta$ is estimated using one of the estimators introduced in Subsection 2.2.
- After the estimation of $X^T\beta$, the nonparametric regression of Y on the estimated value $X^T\hat{\beta}$ is applied.
- UCB limits are constructed based on the nonparametric estimates.

If the parametric link function lies around the nonparametric estimates between the confidence limits, it is concluded that the use of the parametric model is appropriate for the data. The UCB limits for the nonparametric estimate $(m(x))$ at point x is given as,

$$\text{P}\left\{\hat{m}_h(x) - z_{n,\,\alpha}\sqrt{\frac{\hat{\sigma}_h^2\,\|K\|_2^2}{nh\hat{f}_h(x)}} \leq m(x) \leq \hat{m}_h(x) + z_{n,\alpha}\sqrt{\frac{\hat{\sigma}_h^2\,\|K\|_2^2}{nh\hat{f}_h(x)}}\right\} \cong 1-\alpha, \tag{14}$$

where h is the optimal bandwidth parameter required for the nonparametric estimate, $\hat{\sigma}_h^2$ is the estimated variance of $m(x)$ given by Eq. (18) and K is an arbitrary kernel function. Gaussian, Epanechnikov and Quadratic kernels are frequently used in practice. It is a well known fact that the choice of the kernel function does not significantly change the estimation results. Therefore any kernel function can be used in the estimation procedure. K' is the first order derivative of K and $\|K\|_2^2$ is the second order norm of K defined by Eq. (16). Here,

$$\begin{aligned} \|K\|_2^2 &= \int [K\,(s)]^2\; ds; \\ z_{n,\alpha} &= \left\{\frac{-\log\left(-\frac{1}{2}\log(1-\alpha)\right)}{(2\delta\log n)^{1/2}} + d_n\right\}^{1/2} \end{aligned} \tag{15}$$

$$d_n = (2\delta\log n)^{1/2} + (2\,\delta\log n)^{-1/2}\log\left(\frac{1}{2\pi}\frac{\left\|K'\right\|_2}{\|K\|_2}\right)^{1/2} \tag{16}$$

$$\hat{\sigma}_h^2(x) = \frac{\frac{1}{n}\sum_{i=1}^{n} K\left(\frac{x_i - x}{h}\right)\{y_i - \hat{m}_h(x)\}^2}{\sum_{i=1}^{n} K\left(\frac{x_i - x}{h}\right)} \tag{17}$$

Restrictive assumptions are needed for the UCB. These assumptions are listed below.

a) The support of X is $[0, 1]$.
b) $m(\cdot), f_X(\cdot)$ and $\sigma(\cdot)$ are twice differentiable.
c) K is differentiable with support [-1,1] and $K(-1) = K(1) = 0$.

d) $h_n = n^{-\delta};\ \delta \in (1/5,\ 1/2)$.

If the semiparametric link function (g) is not scaled in the same way as the parametric link function (G), the two link functions cannot be shown on the same graph simultaneously. The following process was followed for solving this problem [6, 8, 15].

a) β is estimated using one of the semiparametric methods.
b) Index values are computed using the estimates $\hat{\beta}$. ($\upsilon_i = x_i\hat{\beta};\ i = 1, \ldots, n$).
c) The scale parameter s and constant term c of the parametric model are estimated using y_i and υ_i.
d) A probability estimation for observation i is obtained from $\hat{y}_i = cdfn[(\upsilon_i - c/s)]$ and $\hat{y}_i = (1 + \exp(c - \upsilon_i)/s)^{-1}$ for the probit and logit model, respectively.
e) The $\tilde{y}_i$'s are computed by applying the nonparametric regression of y_i on υ_i, then the link function is estimated and confidence limits are constructed.
f) $\hat{y}_i$, $\tilde{y}_i$ and the confidence limits are graphed against υ_i.

4. XploRe commands for testing the parametric models against their semiparametric alternatives

In this section, the commands we constructed in the windows based version 4.8 of the XploRe package for testing the parametric logit and probit models against their semiparametric alternatives are introduced in the case of continuous and mixed explanatory variable models, separately. The quantlet "dwade" is used for the models with continuous explanatory variables whereas the quantlet "adedis" is used for the estimation of the discrete-continuous explanatory variable models [2, 4, 15].

4.1. Commands for testing the validity of the parametric probit model. In this subsection, explanations of the commands we wrote for testing the validity of the parametric probit model with continuous and mixed explanatory variables,respectively, are given.

4.1.1. *Commands for the model with continuous explanatory variable(s).*

`proc(cb4)=ozge()`
`dat=read("probit1")` ; Reads the data set called "probit1" written in ASCII format.
`y=dat[,3]` ; Describes the column number of the dependent variable (y) in the data set.
`x=dat[,1:2]` ; Describes the column number of the explanatory variables (x) in the data set.
`x=x.-mean(x)` ; Centralizes x values to eliminate high correlation.
`ozdeg=eigsm(cov(x))` ; Calculates the eigenvalues and eigenvectors of the covariance matrix of x.
`w=ozdeg.values` ; Expresses the eigenvalues using matrix "w".
`v=ozdeg.vectors` ; Expresses the eigenvectors using matrix "v".
`mah=v*(sqrt(1./w).*v')` ; Applies the Mahalanobis transformation.
`x=x*mah` ; Weights raw data matrix x by the transformation matrix "mah".
`library("smoother")` ; Calls the "smoother" library for the estimation of β.
`library("metrics")` ; Calls the "metrics" library for the mathematical calculations.
`library("plot")` ; Calls the "plot" library for the graphical representation.
`h=0.2*(max(x).-min(x))'` ; Describes the bandwidth value required for the estimation of β.
`b=dwade(x,y,h)` ; Gives the semiparametric estimation using the "dwade" method.

`b=mah*b` ; Gives the original values of b estimations.
`b=b./abs(b[1,])` ; Normalizes all estimated b 's dividing by the first estimated coefficient. This normalization is required for the comparison of the estimated parameters of the parametric probit model and the semiparametric alternative.
υ_i `= x*b` ; Gives the linear index estimation of observation i.
`x=matrix(rows(x))~`υ_i ; Adds a column matrix with entries "1" to the left side of the matrix.

; **The estimation of the scale s and constant c of the parametric probit model**

`library("glm")` ; Calls the "glm" library for the estimation of the parametric model.
`g=glmest("bipro",x,y)` ; Gives the estimations of the parametric probit model.
`glmout("bipro",x,y,g.b,g.bv,g.stat)` ; Gives the outputs of the parametric probit model.
`c=g.b[1,]` ; Gives the first coefficient of the parametric probit model (b_0).
`s=g.b[2,]` ; Gives the second coefficient of the parametric probit model (b_1).
`yhatpro=cdfn((` υ_i`-c)/s)` ; Calculates the probability of belonging to the category "1" coded in the dependent variable for each observation using c and s values of the probit model.
`z=y~yhatpro` ; Adds the yhatpro column to the right side of y.
`z1=` υ_i`~yhatpro` ; Adds the yhatpro column on the right side of υ_i.
`z1sirali=sort(z1)` ; Sorts the z1 values.

; **Nonparametric regression of y on** υ_i

`data=`υ_i `~y` ; Adds the column matrix y to the right side of υ_i.
`h1=regxbwsel(data)` ; Gives alternative bandwidth selection methods such as Cross-Validation, Shibata's Model Selector, Akaike's Information Criterion, Rice's T etc. The Cross-Validation method is used here.
`{mh,clo,cup}=regxcb(data,h1,0.05,"gau")` ; Calculates mh, the lower confidence band (clo) limit and the upper confidence band (cup) limit at the $\alpha = 0.05$ level and with the "Gaussian" kernel function. This command provides users a chance to change the confidence level (0.10, 0.20 etc.) and the kernel function ("epa", "qua", etc).
`{mh,cli,cui}=regxci(data,h1,0.05,"gau")` ; Calculates mh and the pointwise confidence intervals with level and with the "Gaussian" kernel function.

; **Graphical representation of mh, yhatpro and the confidence bands**

`z1sirali=setmask(z1sirali,"circles","red")` ; Describes the image of "z1 sirali" in the graph.
`mh=setmask(mh,"line","black")` ; Describes the image of "mh" in the graph.
`clo=setmask(clo,"line","blue","thin","dashed")` ; Describes the image of "clo" in the graph.
`cup=setmask(cup,"line","blue","thin","dashed")` ; Describes the image of "cup" in the graph.
`plot(z1sirali,mh,clo,cup)` ; Plots "z1sirali", "mh", "clo" and "cup".
`endp`
`ozge()`

4.1.2. *Commands for the model with mixed explanatory variable(s).*

`proc(cb4)=ozge ()`
`dat=read("probit2")` ; Reads the data set called "probit2" written in ASCII format.

`y=dat[,4]` ; Describes the column number of the dependent variable (y) in the data set.

`x=dat[,1:2]` ; Describes the column number of the continuous explanatory variable(s) (x) in the data set.

`z=dat[,3]` ; Describes the column number of the discrete explanatory variable(s) (z) in the data set.

`x=x.-mean(x)` ; Centralizes x values to eliminate high correlation.

`ozdeg=eigsm(cov(x))` ; Calculates the eigenvalues and eigenvectors of the covariance matrix of x.

`w=ozdeg.values` ; Expresses the eigenvalues using a matrix "w".

`v=ozdeg.vectors` ; Expresses the eigenvectors using a matrix "v".

`mah=v*(sqrt(1./w).*v')` ; Applies the Mahalanobis transformation.

`x=x*mah` ; Weights the raw data matrix x by the transformation matrix "mah".

`library("smoother")` ; Calls the "smoother" library for the estimation of β.

`library("metrics")` ; Calls the "metrics" library for the mathematical calculations.

`library("plot")` ; Calls the "plot" library for the graphical representation.

`h=0.2*(max(x).-min(x))'` ; Describes the bandwidth value required for the estimation of β.

`{delt,alphahat,lim,hd,text}=adedis(z,x,y,h,1.5,0.2,0.8)` ; Executes the "adedis" command for the estimation of the β's for the discrete and continuous explanatory variables, separately. "delt" contains the β estimations of the continuous variable(s) whereas "alphahat" contains the β estimations of the discrete one(s). Using the methods in Subsection 2.2.3, hfac = 1.5; c0 = 0.2 and c1 = 0.8 are determined.

`b=mah*delt` ; Shows the transformations to the original values of the estimations of the continuous explanatory variables.

`b=b./abs(b[1,])` ; Normalizes all estimated b 's by dividing by the first estimated coefficient. This normalization is required for the comparison of the estimated parameters of the parametric probit model and the semiparametric alternative.

υ_i`= x*b+z*alphahat` ; Gives the linear index estimation of observation i.

`x=matrix(rows(x))`$\sim \upsilon_i$; Adds a column matrix of elements "1" to the left side of the matrix υ_i.

; **The estimation of the scale s and constant c of the parametric probit model**

`library("glm")` ; Calls the "glm" library for the estimation of the parametric model.

`g=glmest("bipro",x,y)` ; Gives the estimations of the parametric probit model.

`glmout("bipro",x,y,g.b,g.bv,g.stat)` ; Gives the outputs of the parametric probit model.

`c=g.b[1,]` ; Gives the first coefficient of the parametric probit model (b_0).

`s=g.b[2,]` ; Gives the second coefficient of the parametric probit model (b_1).

`yhatpro=cdfn((`υ_i`-c)/s)` ; Calculates the probability of belonging to the category "1" coded in the dependent variable for each observation using the c and s values of the probit model.

`z=yyhatpro` ; Adds the yhatpro column to the right side of y.

`z1=`υ_i $\sim$`yhatpro` ; Adds the yhatpro column to the right side of υ_i.

`z1sirali=sort(z1)` ; Sorts the z1 values.

; **Nonparametric regression of y on** υ_i

`data=`υ_i $\sim$`y` ; Adds the y column matrix to the right side of υ_i.

`h1=regxbwsel(data)` ; Gives alternative bandwidth selection methods such as Cross-Validation, Shibata's Model Selector, Akaike's Information Criterion, Rice's T etc. The Cross-Validation method is used here.

`{mh,clo,cup}=regxcb(data,h1,0.05,"gau")` ; Calculates mh, the lower confidence band (clo) limit and the upper confidence band (cup) limit at the $\alpha = 0.05$ confidence level and with the "Gaussian" kernel function. This command provides the user the chance to change the confidence level (0.10, 0.20 etc.) and the kernel function ("epa", "qua" etc).

`{mh,cli,cui}=regxci(data,h1,0.05,"gau")` ; Calculates mh and pointwise confidence intervals at the $\alpha = 0.05$ level and with the "Gaussian" kernel function.

Graphical representation of mh, yhatpro and the confidence bands

```
z1sirali=setmask(z1sirali,"circles","red") ; Describes the image of "z1 sirali" in the graph.
mh=setmask(mh,"line","black") ; Describes the image of "mh" in the graph.
clo=setmask(clo,"line","blue","thin","dashed") ; Describes the image of "clo" in the graph.
cup=setmask(cup,"line","blue","thin","dashed") ; Describes the image of "cup" in the graph.
plot(z1sirali,mh,clo,cup) ; Plots "z1sirali", "mh", "clo" and "cup".
endp
ozge()
```

4.2. Commands for testing the validity of the parametric logit model. In this subsection, explanations of the commands written for testing the validity of the parametric logit model with continuous and mixed explanatory variables are given.

4.2.1. *Commands for the model with continuous explanatory variable(s).*

```
proc(cb4)=ozge ()
dat=read("logit1") ; Reads the data set called "logit1" written in ASCII format.
y=dat[,3] ; Describes the column number of the dependent variable (y) in the data set.
x=dat[,1:2] ; Describes the column number of the explanatory variables (x) in the data set.
x=x.-mean(x) ; Centralizes x values to eliminate high correlation.
ozdeg=eigsm(cov(x)) ; Calculates the eigenvalues and eigenvectors of the covariance matrix of x.
w=ozdeg.values ; Expresses the eigenvalues using a matrix "w".
v=ozdeg.vectors ; Expresses the eigenvectors using a matrix "v".
mah=v*(sqrt(1./w).*v') ; Applies the Mahalanobis transformation.
x=x*mah ; Weights raw data matrix x by the transformation matrix "mah".
library("smoother") ; Calls the "smoother" library for the estimation of β.
library("metrics") ; Calls the "metrics" library for the mathematical calculations.
library("plot") ; Calls the "plot" library for the graphical representation.
h=0.2*(max(x).-min(x))' ; Describes the bandwidth value required for the estimation of β.
b=dwade(x,y,h) ; Gives the semiparametric estimation of β using the "dwade" method.
b=mah*b ; Gives the original values of the b estimations.
```

`b=b./abs(b[1,])` ; Normalizes all estimated b 's by dividing by the first estimated coefficient. This normalization is required for the comparison of the estimated parameters of the parametric logit model and the semiparametric alternative.

υ_i`= x*b` ; Gives the linear index estimation of observation i.

`x=matrix(rows(x))`$\sim \upsilon_i$; Adds a column matrix with elements "1" to the left side of the matrix υ_i.

; **The estimations of the scale s and constant c of the parametric logit model**

`library("glm")` ; Calls the "glm" library for the estimation of the parametric model.

`g=glmest("bilo",x,y)` ; Gives the estimations of the parametric logit model.

`glmout("bilo",x,y,g.b,g.bv,g.stat)` ; Gives the outputs of the parametric logit model.

`c=g.b[1,]` ; Gives the first coefficient of the parametric logit model (b_0).

`s=g.b[2,]` ; Gives the second coefficient of the parametric logit model (b_1).

`yhat=(1+exp(c-vi)/s)^-1` ; Calculates the probability of belonging to the category "1" coded in the dependent variable for each observation using the c and s values of the logit model.

`z=y`$\sim$`yhat` ; Adds the yhat column to the right side of y.

`z1=`υ_i $\sim$`yhat` ; Adds the yhat column to the right side of υ_i.

`z1sirali=sort(z1)` ; Sorts the z1 values.

; **Nonparametric regression of y on** υ_i

`data=`υ_i $\sim$`y` ; Adds the y column matrix to the right side of υ_i.

`h1=regxbwsel(data)` ; Gives alternative bandwidth selection methods such as Cross-Validation, Shibata's Model Selector, Akaike's Information Criterion, Rice's T etc. The Cross-Validation method is used here.

`{mh,clo,cup}=regxcb(data,h1,0.05,"gau")` ; Calculates mh, the lower confidence band (clo) limit and upper confidence band (cup) limit at the $\alpha = 0.05$ level and with the "Gaussian" kernel function. This command provide users the chance to change the confidence level (0.10, 0.20 etc.) and the kernel function ("epa", "qua" etc).

`{mh,cli,cui}=regxci(data,h1,0.05,"gau")` ; Calculates mh and the pointwise confidence intervals at the $\alpha = 0.05$ level and with the "Gaussian" kernel function.

; **Graphical representation of mh, yhat and the confidence bands**

`z1sirali=setmask(z1sirali,"circles","red")` ; Describes the image of "z1 sirali" in the graph.

`mh=setmask(mh,"line","black")` ; Describes the image of "mh" in the graph.

`clo=setmask(clo,"line","blue","thin","dashed")` ; Describes the image of "clo" in the graph.

`cup=setmask(cup,"line","blue","thin","dashed")` ; Describes the image of "cup" in the graph.

`plot(z1sirali,mh,clo,cup)` ; Plots "z1sirali", "mh", "clo" and "cup".

```
endp
ozge()
```

4.2.2. *Commands for the model with mixed explanatory variable(s).*

```
proc(cb4)=ozge ()
```

`dat=read("logit2")` ; Reads the data set called "logit2" written in ASCII format.

`y=dat[,4]` ; Describes the column number of the dependent variable (y) in the data set.

`x=dat[,1:2]` ; Describes the column number of the continuous explanatory variable(s) (x) in the data set.

`z=dat[,3]` ; Describes the location in the data set of the discrete explanatory variable(s) (z).

`x=x.-mean(x)` ; Centralizes the x values to eliminate high correlation.

`ozdeg=eigsm(cov(x))` ; Calculates the eigenvalues and eigenvectors of the covariance matrix of x.

`w=ozdeg.values` ; Expresses the eigenvalues using a matrix "w".

`v=ozdeg.vectors` ; Expresses the eigenvectors using a matrix "v".

`mah=v*(sqrt(1./w).*v')` ; Applies the Mahalanobis transformation.

`x=x*mah` ; Weights the raw data matrix x by the transformation matrix "mah".

`library("smoother")` ; Calls the "smoother" library for the estimation of β.

`library("metrics")` ; Calls the "metrics" library for the mathematical calculations.

`library("plot")` ; Calls the "plot" library for the graphical representation.

`h=0.2*(max(x).-min(x))'` ; Describes the bandwidth value required for the estimation of β.

`{delt,alphahat,lim,hd,text}=adedis(z,x,y,h,1.5,0.2,0.8)` ; Executes the "adedis" command for the estimation of the β's for the discrete and continuous explanatory variables, separately. "delt" contains the β estimations of the continuous variable(s) whereas "alphahat" contains the β estimations of the discrete one(s). Using the methods in Subsection 2.2.3, hfac = 1.5; c0 = 0.2 and c1 = 0.8 are determined.

`b=mah*delt` ; Shows the transformations to the original values of the estimations of the continuous explanatory variables.

`b=b./abs(b[1,])` ; Normalizes all estimated b's by dividing by the first estimated coefficient. This normalization is required for the comparison of the estimated parameters of the parametric logit model and the semiparametric alternative.

υ_i`= x*b+z*alphahat` ; Gives the linear index estimation of observation i.

`x=matrix(rows(x))~` υ_i ; Adds a column matrix with elements "1" to the left side of the matrix υ_i.

; **The estimations of the scale s and constant c of the parametric logit model**

`library("glm")` ; Calls the "glm" library for the estimation of the parametric model.

`g=glmest("bilo",x,y)` ; Gives the estimations of the parametric logit model.

`glmout("bilo",x,y,g.b,g.bv,g.stat)` ; Gives the outputs of the parametric logit model.

`c=g.b[1,]` ; Gives the first coefficient of the parametric logit model (b_0).

`s=g.b[2,]` ; Gives the second coefficient of the parametric logit model (b_1).

`yhat=(1+exp(c-vi)/s)^ -1` ; Calculates the probability of belonging to the category "1" coded in the dependent variable for each observation using the c and s values of the logit model.

`z=y~yhat` ; Adds the yhat column to the right side of y.

`z1=`υ_i `~yhat` ; Adds the yhat column to the right side of υ_i.

`z1sirali=sort(z1)` ; Sorts the z1 values.

; **Nonparametric regression of y on** υ_i

`data=`υ_i `~y` ; Adds the y column matrix to the right side of υ_i.

`h1=regxbwsel(data)` ; Gives alternative bandwidth selection methods such as Cross-Validation, Shibata's Model Selector, Akaike's Information Criterion, Rice's T etc. The Cross-Validation method is used here.

{mh,clo,cup}=regxcb(data,h1,0.05,"gau") ; Calculates mh, the lower confidence band (clo) limit and upper confidence band (cup) limit at the $\alpha = 0.05$ level and with the "Gaussian" kernel function. This command provide users to with the chance to change the confidence level (0.10, 0.20 etc.) and the kernel function ("epa", "qua" etc).

{mh,cli,cui}=regxci(data,h1,0.05,"gau") ; Calculates mh and the pointwise confidence intervals at the $\alpha = 0.05$ level and with the "Gaussian" kernel function.

; **Graphical representation of mh, yhatpro and the confidence bands**

```
z1sirali=setmask(z1sirali,"circles","red") ; Describes the image of "z1 sirali" in the graph.
mh=setmask(mh,"line","black") ; Describes the image of "mh" in the graph.
clo=setmask(clo,"line","blue","thin","dashed") ; Describes the image of "clo" in the graph.
cup=setmask(cup,"line","blue","thin","dashed") ; Describes the image of "cup" in the graph.
plot(z1sirali,mh,clo,cup) ; Plots "z1sirali", "mh", "clo" and "cup".
endp
ozge()
```

5. An application

In this section, the applicability of all XploRe commands was shown using an artificial data. In the simulated data, Y is a binary variable coded as 0 and 1. X is a $n \times 2$ matrix denoting the observed continuous variables. Z is a $n \times 1$ matrix representing the observed discrete explanatory variable. The sample size is 80. The commands given in Section 4 were run to test the validity of the parametric probit and logit models. When the procedures were run, an optional bandwidth selection method for the estimation of mh (such as Cross-Validation, AIC etc.) was displayed. The UCB confidence limits are calculated and graphed after selecting one of them.

5.1. Results for the parametric probit model with continuous explanatory variables. Figure 1 shows the optimal bandwidth parameter value ($h1 = 1.06988$) obtained by the cross-validation method. The optimal range of h was (0.168495-2.69523).

Figure 1. Optimal bandwidth value for the nonparametric regression of Y on X with continuous explanatory variables

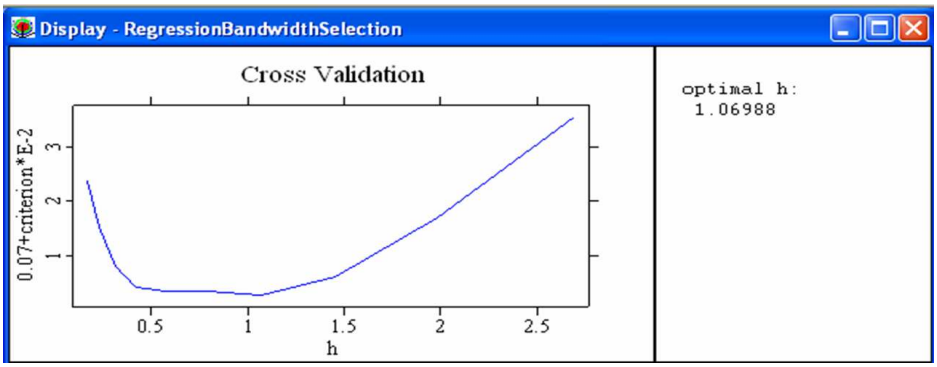
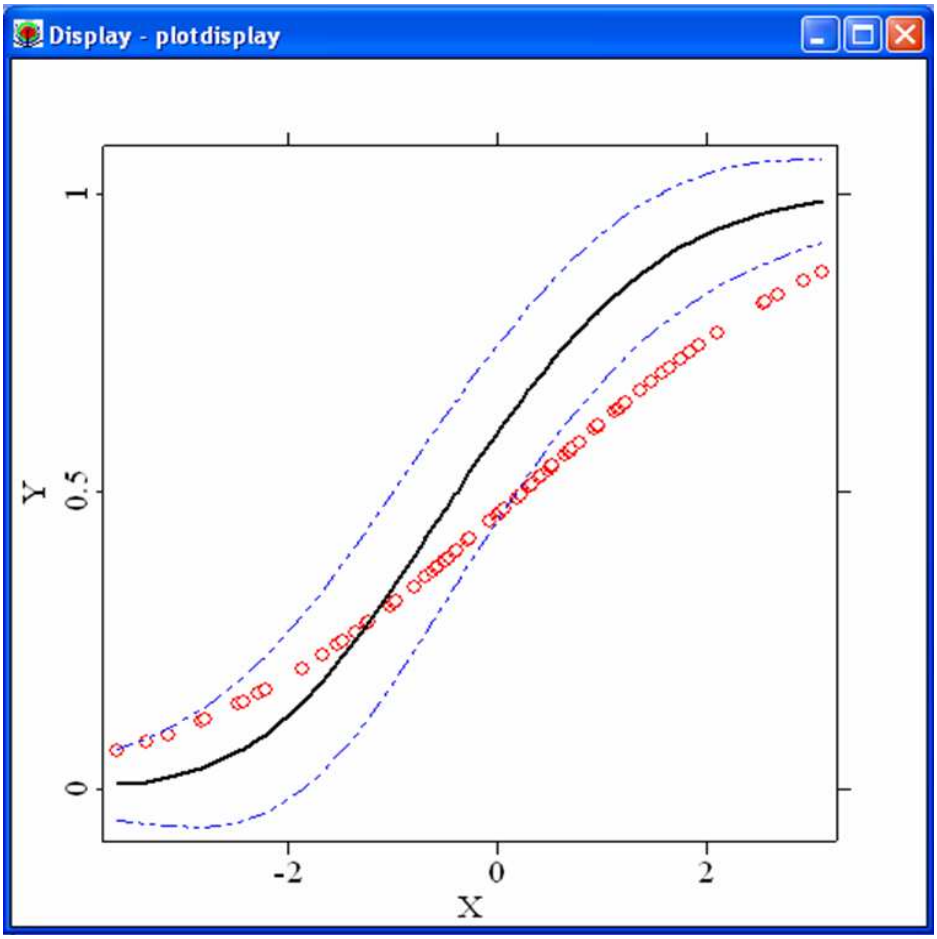


Figure 2 shows the graph of the estimated parametric curve, nonparametric curve and UCB limits for the $\alpha = 0.05$ level and the Gaussian kernel.

Figure 2. Estimated parametric curve, nonparametric curve and UCB limits with a $1 - \alpha = 0.95$ confidence level

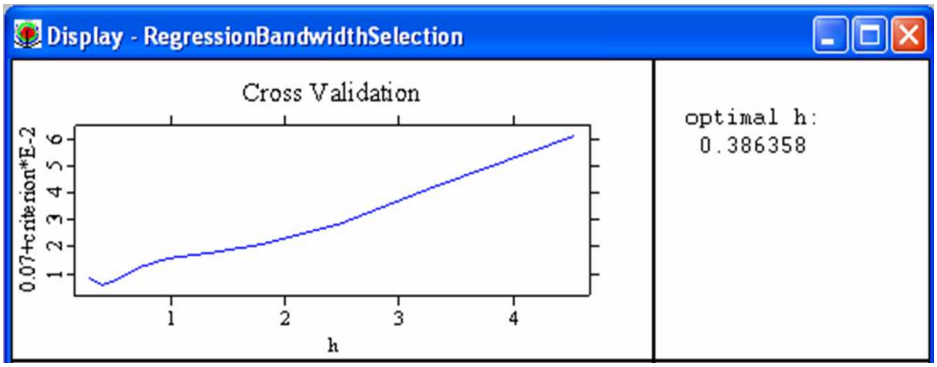


In Figure 2, red circles represent the parametric link function, the black line represents the estimated nonparametric curve and the broken blue line represents the lower and upper UCB limits.

Because some part of the red circles lie outside the UCB limits, it is concluded that the use of the parametric probit model is not appropriate for modeling the data and the use of the semiparametric approach is proposed.

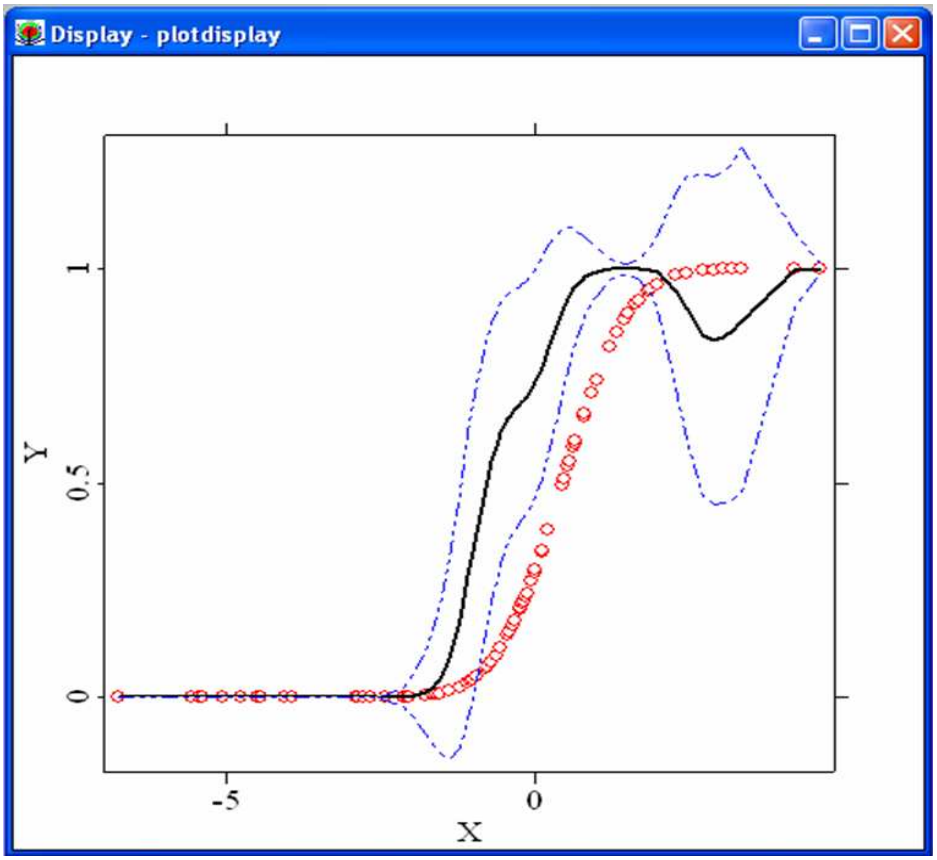
5.2. Results of the parametric probit model with mixed explanatory variables. As seen in Figure 3, the optimal bandwidth parameter value obtained by the cross-validation method is ($h1 = 0.386358$) in this case. The optimal range of h is (0.283922-4.54275).

Figure 3. Optimal bandwidth value for the nonparametric regression of Y on X with mixed explanatory variables



In Figure 4, the use of the parametric probit model is rejected again.

Figure 4. Estimated parametric curve, nonparametric curve and UCB limits with a $1-\alpha=0.95$ confidence level



5.3. Results of the parametric logit model with continuous explanatory variables. The optimal bandwidth parameter value obtained by the cross-validation method is ($h1 = 1.06988$). The optimal range of h is (0.168495-2.69523).

Figure 5. Optimal bandwidth value for the nonparametric regression of Y on X with continuous explanatory variables

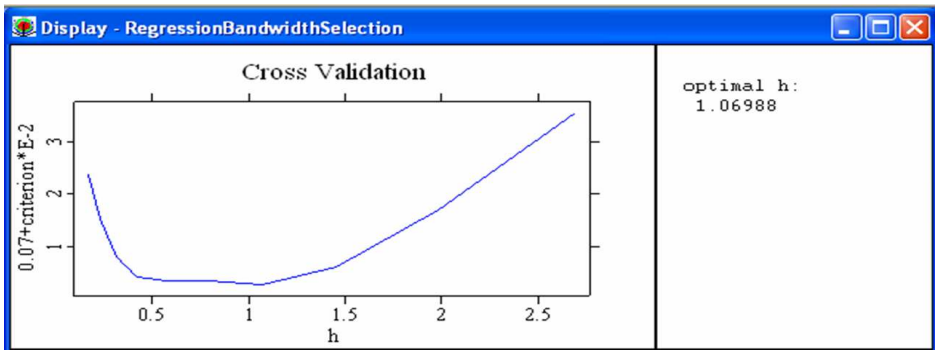
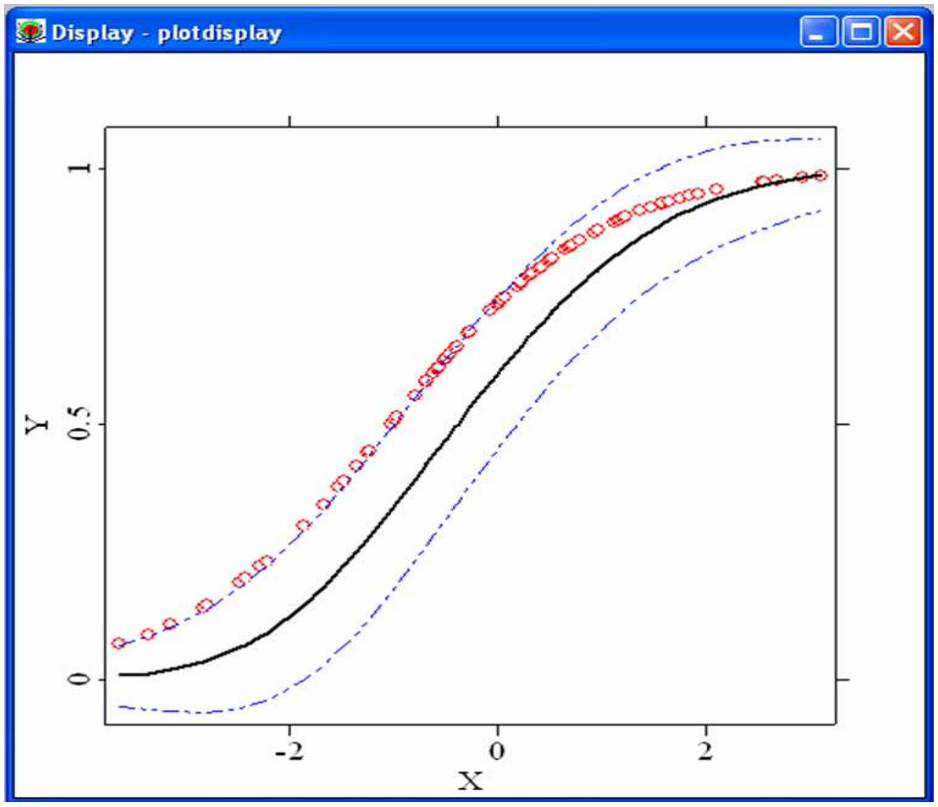


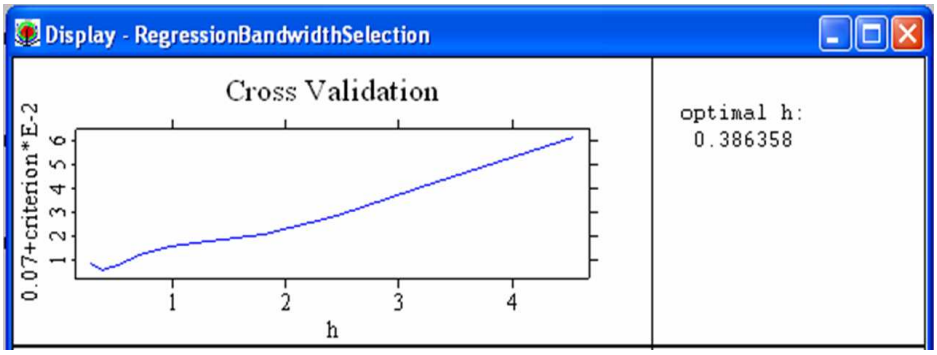
Figure 6 suggests the use of the semiparametric approach instead of the parametric logit model for modeling the data as in the probit model case.

Figure 6. Estimated parametric curve, nonparametric curve and UCB limits with a $1 - \alpha = 0.95$ confidence level



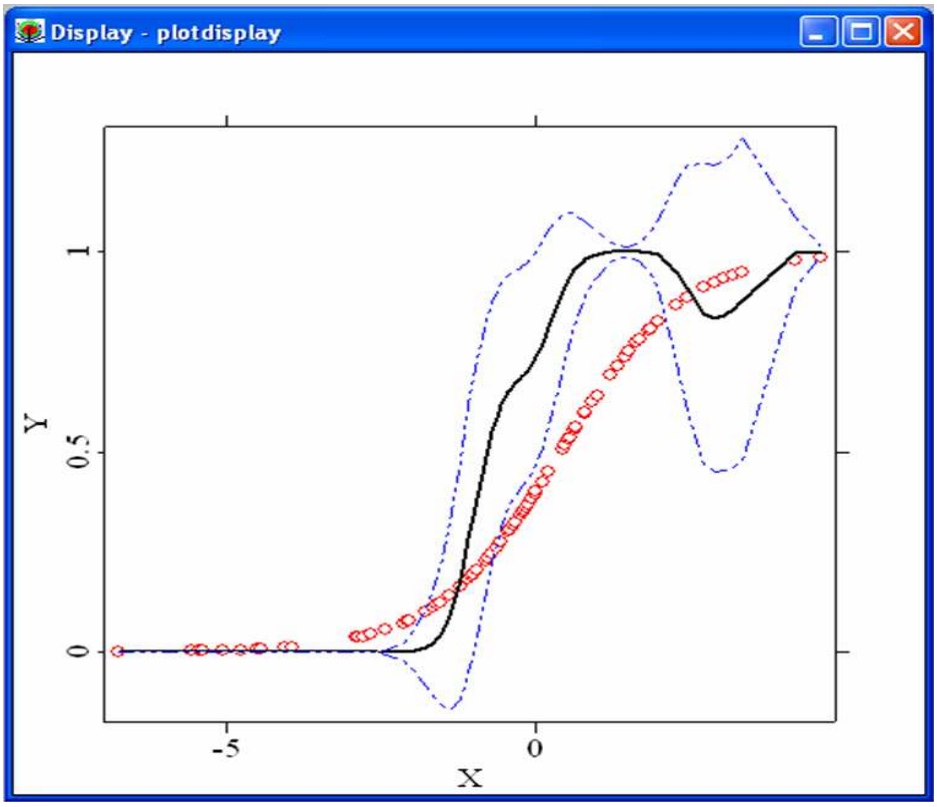
5.4. Results of the parametric logit model with mixed explanatory variables. The optimal bandwidth parameter value obtained by the cross-validation method is ($h1 = 0.386358$) in this case. The optimal range of h is (0.283922-4.54275).

Figure 7. Optimal bandwidth value for the nonparametric regression of Y on X with mixed explanatory variables



In Figure 8, the use of the parametric logit model with mixed explanatory variables is also rejected.

Figure 8. Estimated parametric curve, nonparametric curve and UCB limits with a $1 - \alpha = 0.95$ confidence level



6. Conclusion

Parametric modeling is widely used in most studies because of its simplicity in interpretation and application for binary responses. However, the validity of these types of models is all based on the assumptions related to the error term. The parametric probit model assumes a normally distributed error term whereas a logistic distribution is required for the parametric logit model. The main problem here is to test the validity of these assumptions. At this point, a statistical testing criterion is needed to determine the validity of the parametric models for the data before the analysis part.

In this study, Uniform Confidence Band Limits (UCB) were used as testing criteria. We wrote the commands for both logit and probit models and for continuous and discrete explanatory variable cases in the Windows based version 4.8 of the XploRe package, which is new for the statistical literature. This study extends the study of Proença and Werwatz [15] in which the code was written for the logit model and only for continuous explanatory variables in the old MsDOS format. The explanation of all commands was given in Section 4. Artificial data was used with two continuous and one discrete explanatory variable with a binary dependent variable. The XploRe commands were executed to test the validity of the parametric probit and logit models for this data. In conclusion, the parametric models were rejected against the semiparametric alternatives in all situations.

Due to the fact that they enable a test of the validity of the parametric probit and logit models before the analysis part, we hope that the updated and extended version of the commands in XploRe will be a guide to practitioners studying in this area. Additionally, the applications given in Section 5 will help applied researchers to see the use of and the applicability of the commands in practice.

References

[1] Aldrich, J. H. and Nelson, F. D. *Linear Probability, Logit and Probit Models* (Sage Publications, London, 1984).

[2] Hardle, W., Klinke, S. and Turlach, B. A. *XploRe: An Interactive Statistical Computing Environment: Statistics and Computing* (Springer-Verlag, New York, 2007).

[3] Hardle, W., Müller, M., Sperlich, S. and Werwatz, A. *Nonparametric and Semiparametric Models* (Springer-Verlag, New York, 2004).

[4] Hardle, W., Hlavka, Z. and Klinke, S. *XploRe Application Guide, e-book* (MD Tech, Springer-Verlag, New York, 2003).

[5] Hardle, W. and Stoker, T. M. *Investigating smooth multiple regression by the method of average derivatives*, Journal of the American Statistical Association **84**, 986–995, 1989.

[6] Horowitz, J. L. *Semiparametric Methods in Econometrics* (Springer-Verlag, New York, 1998).

[7] Horowitz, J. L. and Hardle, W. *Direct semiparametric estimation of single-index models with discrete covariates*, Journal of the American Statistical Association **91**, 1632–1640, 1996.

[8] Horowitz, J. L. and Hardle, W. *Testing a parametric model against a semiparametric alternative*, Econometric Theory **10**, 821–848, 1994.

[9] Ichimura, H. *Semiparametric least squares (sls) and weighted sls estimation of single-index models*, Journal of Econometrics **58**, 71–120, 1993.

[10] Klein, W. and Spady, R. H. *An efficient semiparametric estimator for binary response models*, Econometrica, **61**, 387–421, 1993.

[11] Manski, C. F. *Identification of binary response models*, Journal of the American Statistical Association **83**, 729–738, 1988.

[12] McCullagh, P. and Nelder, J. A. *Generalized Linear Models* (Monographs on Statistics and Applied Probability **37**, Chapman and Hall, London, 1989).

[13] Powell, J. L., Stock, J. H. and Stoker, T. M. *Semiparametric estimation of index coefficients*, Econometrica **57** (6), 1403–1430, 1989.

[14] Powers, D. A. and Xie, Y. *Statistical Methods for Categorical Data Analysis* (Academic Press, 2000).
[15] Proença, I. and Werwatz, A. *Comparing Parametric and Semiparametric Binary Response Models*, Sonderforschungsbereich **373** 2000-20 (Humboldt Universitaet, Berlin, 1994).