

# Package: persDx (via r-universe)

May 27, 2026

**Type** Package

**Title** Personalized Diagnostics Rules for Subgroup Identification and  
Personalized Biomarker Discovery

**Version** 0.5.0

**Date** 2023-08-15

**Description** Tailoring the optimal biomarker(s) for disease screening  
or diagnosis based on subjects' individual characteristics.

**License** GPL (>= 2)

**Depends** R (>= 4.2.0), pROC, survivalROC, stats, graphics

**NeedsCompilation** no

**Author** Yunro Chung [aut, cre]  
(<https://orcid.org/0000-0001-9125-9277>)

**Maintainer** Yunro Chung <yunro.chung@asu.edu>

**Repository** <https://zea23i651096531.r-universe.dev>

**Date/Publication** 2023-08-16 07:30:49 UTC

**RemoteUrl** <https://github.com/cran/persDx>

**RemoteRef** HEAD

**RemoteSha** 7bc664b7329865073866926a8f058641694e6869

## Contents

persDx-package . . . . .	2
np_lpd . . . . .	2
np_lpd_survival . . . . .	4
<b>Index</b>	<b>8</b>

---

persDx-package

*Estimating Personalized Diagnostics Rules*

---

### Description

Personalized Recommendation of biomarkers or screening/diagnostic tests based on patients' individual profile.

### Details

Package: persDx  
Type: Package  
Version: 0.5.0  
Date: 2023-08-15  
License: GPL (>= 2)

### Author(s)

Yunro Chung [aut,cre] Maintainer: Yunro Chung <yunro.chung@asu.edu>

### References

Yaliang Zhang and Yunro Chung, Nonparametric estimation of linear personalized diagnostics rules via efficient grid algorithm (under revision)

---

np\_lpd

*Nonparametric estimation of linear personalized diagnostic rules.*

---

### Description

Nonparametric estimation of the personalized diagnostics rule to find subgroup-specific biomarkers according to linear combination of predictors.

### Usage

np\_lpd(D, YA, YB, X, dirA, dirB, eps, plot, A, B, c, d)

**Arguments**

D	Binary outcome with D=1 for disease (or case) and D=0 for non-diseased (or control) (n X 1 vector).
YA	Biomarker A, measured on a continuous scale (n X 1 vector).
YB	Biomarker B, measured on a continuous scale (n X 1 vector).
X	Predictors (n x p matrix).
dirA	Direction of YA to D, where dirA="<" (or dirA=">") indicates higher (or lower) YA is associated with Pr(D=1). Default is dirA="<".
dirB	Direction of YB to D, where dirB="<" (or dirB=">") indicates higher (or lower) YB is associated with Pr(D=1). Default is dirB="<".
eps	Tuning parameter for predictor selections. Default is eps=0.01.
plot	plot=TRUE (or FALSE) shows (or does not show) the receiver operating characteristics (ROC) curve.
A	Grid search parameter (Discrete). Default is A=0
B	Grid search parameter (Discrete). Default is B=0
c	Grid search parameter. Default is c=2
d	Grid search parameter. Default is d=2

**Details**

The np.lin.persDx function estimates the personalized diagnostics rule  $\tau(X)$ , where  $\tau(X)=A$  recommends  $YA$  if  $\theta_1 X_1 + \dots + \theta_p X_p > \theta_0$  or  $\tau(X)=B$  recommends  $YB$  otherwise by maximizing (empirical) area under the ROC curve (AUC). Here, the AUC is computed based on  $YC$  with the direction of "<", i.e. higher  $YC$  is associated with Pr(D=1), where  $YC = YA$  if  $\tau(X)=A$  and dirA="<", or  $YC = YB$  if  $\tau(X)=B$  and dirB="<". If dirA=">" (or dirB=">"), negative YA (or YB) is used.

A forward grid rotation algorithm (FGR) is used to estimate  $\theta_0, \theta_1, \dots, \theta_p$  by sequentially adding each of the predictors to  $\tau(X)$  that increases the AUC the most. The stopping criteria is AUC increasement is less than or equal to eps. The eps controls the model complexity. The cross-validation techniques can be used to find the optimal eps.

The FGR results in a suboptimal solution. The accuracy is improved by setting higher A, B, c, d, but it increases increase computational costs, or vice versa. We thus recond this function when p is small or around 10.

**Value**

A list of class np.lin.persDx:

df	Data frame with D, YA, YB, X, tau, YC, where tau=A or B for recommending YA or YB, respectively.
AUCA	AUC for YA.
AUCB	AUC for YB.
AUC	AUC for YC.

tpfp	Data frame with cutoff, tp, fp, where tp and fp are true and false positive positives at the cutoff values of YC.
theta	Estimated regression parameters.
theta0	Estimated threshold parameter.
PLOT	TRUE or FALSE to show ROC curves.

**Author(s)**

Yunro Chung [aut, cre]

**References**

Yaliang Zhang and Yunro Chung, Nonparametric estimation of linear personalized diagnostics rules via efficient grid algorithm (submitted)

**Examples**

```
#simulate data
set.seed(1)
n=100
D=c(rep(1,n/2),rep(0,n/2))

X1=runif(n,0,1)
X2=runif(n,0,1)
X3=runif(n,0,1)
X=data.frame(X1,X2,X3)

tau=rep("B",n)
tau[X1+X2>=1]="A"

YA=D*(rnorm(n,2,1)*(tau=="A")+rnorm(n,0,1)*(tau=="B"))+
  (1-D)*rnorm(n,0,1)
YB=D*(rnorm(n,1,1)*(tau=="B")+rnorm(n,0,1)*(tau=="A"))+
  (1-D)*rnorm(n,0,1)

#run
fit=np_lpd(D, YA, YB, X)
fit
```

---

np\_lpd\_survival

*Nonparametric estimation of linear personalized diagnostic rules with right-censored survival outcome.*

---

**Description**

Nonparametric estimation of personalized diagnostics rule to find subgroup-specific biomarkers according to linear combination of predictors.

**Usage**

```
np_lpd_survival(Stime,D,YA,YB,X,dirA,dirB,predict.time,span,eps,plot,A,B,c,d)
```

**Arguments**

Stime	Event time or censoring time for subjects (n x 1 vector).
D	Indicator of status, where D=1 if death or event, and D=0 otherwise (n X 1 vector).
YA	Biomarker A, measured on a continuous or ordinal scale (n X 1 vector).
YB	Biomarker B, measured on a continuous or ordinal scale (n X 1 vector).
X	Predictors (n x p matrix).
predict.time	Time point to evaluate YA and YB.
span	Span for the nearest neighbor estimation (NNE).
dirA	Direction of YA to D, where dirA="<" (or dirA=">") indicates higher (or lower) YA is associated with Pr(D=1)). Default is dirA="<".
dirB	Direction of YB to D, where dirB="<" (or dirB=">") indicates higher (or lower) YB is associated with Pr(D=1)). Default is dirB="<".
eps	Tuning parameter for predictor selections. Default is eps=0.01.
plot	plot=TRUE (or FALSE) shows (or does not show) the receiver operating characteristics (ROC) curve.
A	Grid search parameter (Discrete). Default is A=0
B	Grid search parameter (Discrete). Default is B=0
c	Grid search parameter. Default is c=2
d	Grid search parameter. Default is d=2

**Details**

The `np.lin.survival.persDx` function estimates the personalized diagnostics rule  $\tau(X)$ , where  $\tau(X)=A$  recommends  $YA$  if  $\theta_1 X_1 + \dots + \theta_p X_p > \theta_0$  or  $\tau(X)=B$  recommends  $YB$  otherwise by maximizing (empirical) survival area under the ROC curve (AUC) at the `predict.time` using the Nearest Neighbor Estimation. Here, the survival AUC is computed based on  $YC$  with the direction of "<", i.e. higher  $YC$  is associated with  $\Pr(D=1)$ , where  $YC = YA$  if  $\tau(X)=A$  and  $\text{dirA}="<"$ , or  $YC = YB$  if  $\tau(X)=B$  and  $\text{dirB}="<"$ . If  $\text{dirA}=">"$  (or  $\text{dirB}=">"$ ), negative  $YA$  (or  $YB$ ) is used.

A forward grid rotation algorithm (FGR) is used to estimate  $\theta_0, \theta_1, \dots, \theta_p$  by sequentially adding each of the predictors to  $\tau(X)$  that increases the AUC the most. The stopping criteria is AUC increase is less than or equal to `eps`. The `eps` controls the model complexity. The cross-validation techniques can be used to find the optimal `eps`.

The FGR results in a suboptimal solution. The accuracy is improved by setting higher `A, B, c, d`, but it increases computational costs. We thus recond this function when `p` is small or around 10.

**Value**

A list of class `np.lin.persDx`:

<code>df</code>	Data frame with <code>Stime</code> , <code>D</code> , <code>YA</code> , <code>YB</code> , <code>X</code> , <code>tau</code> , <code>YC</code> , where <code>tau=A</code> or <code>B</code> for recommending <code>YA</code> or <code>YB</code> , respectively.
<code>AUCA</code>	Survival AUC for <code>YA</code> at <code>predict.time</code> .
<code>AUCB</code>	Survival AUC for <code>YB</code> at <code>predict.time</code> .
<code>AUC</code>	Survival AUC for <code>YC</code> .
<code>tpfp</code>	Data frame with <code>cutoff</code> , <code>tp</code> , <code>fp</code> , where <code>tp</code> and <code>fp</code> are true and false positive positives at the cutoff values of <code>YC</code> .
<code>theta</code>	Estimated regression parameters.
<code>theta0</code>	Estimated threshold parameter.
<code>PLOT</code>	TRUE or FALSE to show survival ROC curves.

**Author(s)**

Yunro Chung [aut, cre]

**References**

Yaliang Zhang and Yunro Chung, Nonparametric estimation of linear personalized diagnostics rules via efficient grid algorithm (submitted)

**Examples**

```
#simulate data
set.seed(1)
n=100
X=abs(rnorm(n,1,1))
C=abs(rnorm(n,1,1))
Stime=pmin(X,C)
D=as.numeric(X<=C)

X1=runif(n,0,1)
X2=runif(n,0,1)
X3=runif(n,0,1)
X=data.frame(X1,X2,X3)

tau=rep("B",n)
tau[X1+X2>=1]="A"

D2=rep(0,n) #event by time 2
D2[which(Stime<=3 & D==1)]=1

YA=D2*(rnorm(n,2,1)*(tau=="A")+rnorm(n,0,1)*(tau=="B"))+
  (1-D2)*rnorm(n,0,1)
YB=D2*(rnorm(n,1,1)*(tau=="B")+rnorm(n,0,1)*(tau=="A"))+
  (1-D2)*rnorm(n,0,1)
```

```
#run
span=0.1
fit=np_lpd_survival(Stime, D, YA, YB, X, predict.time=1, span=span)
fit
```

# Index

- \* **Area under the ROC curve**
    - [persDx-package, 2](#)
  - \* **Personalized medicine**
    - [persDx-package, 2](#)
  - \* **Precision medicine**
    - [persDx-package, 2](#)
  - \* **Receiver operating characteristics curve**
    - [persDx-package, 2](#)
  - \* **Subgroup identification**
    - [persDx-package, 2](#)
  - \* **Survival analysis**
    - [persDx-package, 2](#)
- [np\\_lpd, 2](#)
- [np\\_lpd\\_survival, 4](#)
- [persDx-package, 2](#)