Package ‘wSVM’

February 20, 2015

Title  Weighted SVM with boosting algorithm for improving accuracy
Version 0.1-7
Date 2010-10-03
Author SungHwan Kim and Soo-Heang Eo
Maintainer SungHwan Kim <swiss747@korea.ac.kr>
Description We propose weighted SVM methods with penalization form. By adding weights to loss term, we can build up weighted SVM easily and examine classification algorithm properties under weighted SVM. Through comparing each of test error rates, we conclude that our Weighted SVM with boosting has predominant properties than the standard SVM have, as a whole.
License GPL-2
Depends R (>= 2.10.1), MASS, quadprog
LazyData yes
LazyLoad yes
Repository CRAN
Date/Publication 2012-10-29 08:59:59
NeedsCompilation no

R topics documented:

wSVM-package .................................................. 2
mixture.example ............................................. 2
wsvm .......................................................... 3
wsvm.boost .................................................. 4
wsvm.predict .................................................. 6

Index  8
Description

We propose weighted SVM methods with penalization form. By adding weights to loss term, we can build up weighted SVM easily and examine classification algorithm properties under weighted SVM. Through comparing each of test error rates, we conclude that our Weighted SVM with boosting has predominant properties than the standard SVM have, as a whole.

Details

Package: wSVM
Type: Package
Version: 0.1-7
Date: 2010-10-03
License: GPL-2
LazyLoad: yes

Author(s)

SungHwan Kim <swiss747@korea.ac.kr>
Soo-heang Eo <hanansh@korera.ac.kr>

See Also

wsvm, wsvm.predict, wsvm.boost

Description

example of mixture data

Source

**Description**

Compute Weighted SVM with boosting algorithm

**Usage**

```r
wsvm(X, Y, c.n, kernel = list(type = 'linear', par = NULL), C = 1, eps = 1e-10)
```

**Arguments**

- **X**: input variable matrix. Data type must be a matrix format.
- **Y**: output variable vector which will be declared as a matrix in SVM. Data type must be a matrix format.
- **c.n**: weighted term.
- **kernel**: set attributes of kernel using list(). `kernel$type` means a type of kernel, including 'linear', 'poly', and 'rbf'. `kernel$par` means a parameter of kernel. For example, `par = degree` for 'poly' and `par = scale` for 'rbf'.
- **C**: regularization parameter.
- **eps**: epsilon value.

**Details**

Weighted SVM with boosting algorithm for improving accuracy.

**Value**

A function `wsvm` generate a list consists of `fit`, `alpha`, `bias` and `sv`.
- `model$fit` = predicted values (n by 1)
- `model$alpha` = estimated coefficients (n by 1)
- `model$bias` = bias term
- `model$sv` = index of support vectors

**Author(s)**

SungWhan Kim `<swiss747@korea.ac.kr>`
Soo-Heang Eo `<hanansh@korea.ac.kr>`

**See Also**

`wsvm.predict`, `wsvm.boost`
Examples

```r
# generate a simulation data set using mixture example (page 17, Friedman et al. 2000)
svm.data <- simul.svm(set.seeds = 123)
X <- svm.data$X
Y <- svm.data$Y
new.X <- svm.data$new.X
new.Y <- svm.data$new.Y

# run weighted K-means clustering SVM with boosting algorithm
model <- wsvm(X, Y, c.n = rep(1/ length(Y), length(Y)))

# predict the model and compute an error rate.
pred <- wsvm.predict(X, Y, new.X, new.Y, model)
Error.rate(pred$predicted.Y, Y)

# add boost algorithm
boo <- wsvm.boost(X, Y, new.X, new.Y, c.n = rep(1/ length(Y), length(Y)),
                 B = 50, kernel.type = list(type = "rbf", par= 0.5), C = 4,
                 epub = 1e-10, plotting = TRUE)
boo
```

---

**wsvm.boost**

*Weighted SVM using boosting algorithm*

**Description**

Improve accuracy for learning algorithm to bond with a lot of weak classifiers to construct the only one strong classifier.

**Usage**

`wsvm.boost(X, Y, new.X, new.Y, c.n, B = 50, kernel.type = list(type = "rbf", par= 0.5), C = 4, eps = `)

**Arguments**

- **X**
  - input variable matrix to generate kernel. Data type must be a matrix format.
- **Y**
  - output variable vector which will be declared as a matrix in SVM. Data type must be a matrix format.
- **new.X**
  - test predictors.
- **new.Y**
  - test response.
- **c.n**
  - weighted term.
- **B**
  - the number of iterations.
kernel.type  set an attributes of kernel using list(). kernel$type means a type of kernel, including 'linear', 'poly', and 'rbf'. kernel$par means a parameter of kernel. For example, par = degree for 'poly' and par = scale for 'rbf'.

C  regularization parameter.
eps  epsilon value.
plotting  logical values. If TRUE, plot the result.

Value
A function wsvm.boost generates a list consists of error.rate and predicted.model.

error.rate  misclassification error rate
predicted.model  predicted model

Author(s)
SungWhan Kim <swiss747@korea.ac.kr>
Soo-Heang Eo <hanansh@korea.ac.kr>

References

See Also
wsvm, wsvm.predict

Examples

# generate a simulation data set using mixture example(page 17, Friedman et al. 2000)

svm.data <- simul.wsvm(set.seeds = 123)
X <- svm.data$X
Y <- svm.data$Y
new.X <- svm.data$new.X
new.Y <- svm.data$new.Y

# run Weighted K-means clustering SVM with boosting algorithm
model <- wsvm(X, Y, c.n = rep(1/ length(Y), length(Y)))

# predict the model and compute an error rate.
pred <- wsvm.predict(X,Y, new.X, new.Y, model)
Error.rate(pred$predicted.Y, Y)

# add boost algorithm
boo <- wsvm.boost(X, Y, new.X, new.Y, c.n = rep(1 / length(Y), length(Y)),
                   B = 50, kernel.type = list(type = "rbf", par = 0.5), C = 4,
                   eps = 1e-10, plotting = TRUE)
boo

---

wsvm.predict

Predict new test set using wsvm function and compute error rate

Description

Predict a weighted svm fit and compute error rate.

Usage

wsvm.predict(X, Y, new.X, new.Y, model, comp.error.rate = FALSE)

Arguments

X input variable matrix to generate kernel. Data type must be a matrix format.
Y output variable vector which will be declared as a matrix in SVM. Data type
   must be a matrix format.
new.X test predictors.
new.Y test response.
model predicted model including alpha and bias terms. The alpha means estimated
   coefficients(nrow(X) by 1) and bias means bias term.
comp.error.rate logical value. If true, calculate error rate.

Details

Predict a weighted svm fit.

Value

A function wsvm.predict generates a list consists of values, g, and error.rate.

predicted.values fitted value at new.X
g signs of predicted values
error.rate misclassification error rate

Author(s)

SungWhan Kim <swiss747@korea.ac.kr>
Soo-Heang Eo <hanansh@korea.ac.kr>
See Also

wsvm, wsvm.boost

Examples

# generate a simulation data set using mixture example(page 17, Friedman et al. 2000)

svm.data <- simul.wsvm(set.seeds = 123)
X <- svm.data$x
Y <- svm.data$y
new.X <- svm.data$new.X
new.Y <- svm.data$new.Y

# run Weighted K-means clustering SVM with boosting algorithm
model <- wsvm(X, Y, c.n = rep(1/ length(Y),length(Y)))

# predict the model and compute an error rate.
pred <- wsvm.predict(X,Y, new.X, new.Y, model)
Error.rate(pred$predicted.Y, Y)

# add boost algorithm

boo <- wsvm.boost(X, Y, new.X, new.Y, c.n = rep(1/ length(Y),length(Y)),
                  B = 50, kernel.type = list(type = "rbf", par= 0.5), C = 4,
                  eps = 1e-10, plotting = TRUE)

boo
Index

*Topic datasets
  mixture.example, 2
*Topic package
  wSVM-package, 2
*Topic svm
  wsvm, 3
  wsvm.boost, 4
  wsvm.predict, 6
mixture.example, 2
wSVM (wSVM-package), 2
wsvm, 2, 3, 5, 7
wSVM-package, 2
wsvm.boost, 2, 3, 4, 7
wsvm.predict, 2, 3, 5, 6