Package 'CosmoPhotoz'

August 24, 2014

Type Package

Title Photometric redshift estimation using generalized linear models

Version 0.1

Date 2014-08-20

Author Rafael S. de Souza, Alberto Krone-Martins, Jonathan Elliott, Joseph Hilbe

Maintainer Rafael S. de Souza <rafael.2706@gmail.com>

Description User-friendly interfaces to perform fast and reliable photometric redshift estimation. The code makes use of generalized linear models and can adopt gamma or inverse gaussian families, either from a frequentist or a Bayesian perspective. The code additionally provides a Shiny application providing a simple user interface.

Depends R (>= 3.0)

License GPL (>= 3)

Imports ggplot2, ggthemes, arm, COUNT, gridExtra, pcaPP, mvtnorm,shiny

NeedsCompilation no

Repository CRAN

Date/Publication 2014-08-24 07:22:40

R topics documented:

`CosmoPhotoz-package` *Photometric redshift estimation based on generalized linear models*

Description

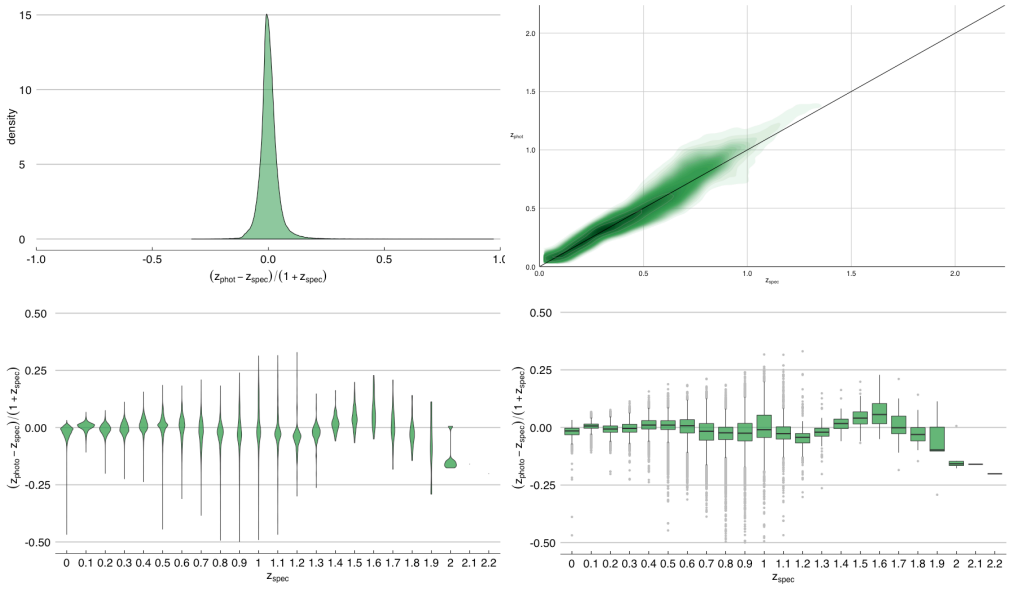
This package provides an user-friendly interfaces to perform fast and reliable photometric redshift estimation. The code makes use of generalized linear models and can adopt gamma or inverse gaussian families, either from a frequentist or a Bayesian perspective. The code additionally provides a Shiny application providing a simple user interface.

Details

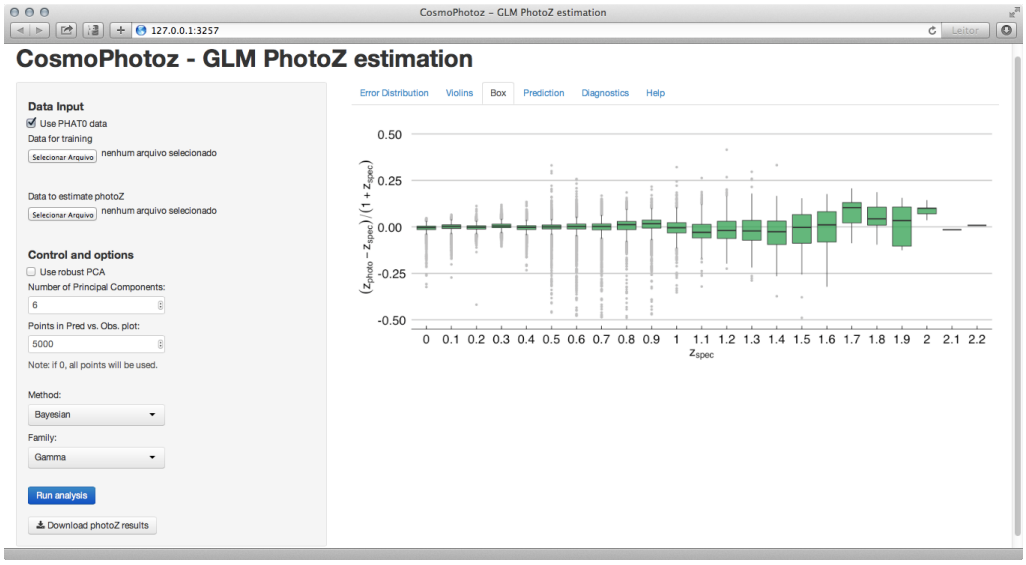
Package:	CosmoPhotoz
Type:	Package
Version:	0.1
Date:	2014-08-22
License:	GPL (>= 3)

The CosmoPhotoz package aims to provide an user-friendly interface to enable the estimation of photometric redshifts. The present version employs generalized linear models and the user can adopt either gamma or inverse gaussian families from either a frequentist or a Bayesian perspective at the fitting step.

The package includes a plotting function to enable the production of diagnostic plots. Four examples of the implemented visual tests can be seen in the figures bellow:



Additionally, the code is also accompanied by a **Shiny** application that can be hosted locally or deployed by the user at a webserver. This application allows the user to run the photometric redshift estimation, to configure many parameters of the code visually and experiment with the results. It also allows the user to either use the PHAT0 data, or to upload its own data files (the expected format can be found at the application's help tab). A screenshot of this application can be seen in the next figure.



To run the graphical interface locally, it is sufficient to call:

```
runApp(paste(find.package("CosmoPhotoz"),"/glmPhotoZ-2/",sep=""))
```

Finally, a short tutorial on how to use the package can be found at:

———————— http://rafaelsdesouza.github.io/CosmoPhotoz/ ————————

Author(s)

Rafael S. de Souza, Alberto Krone-Martins, Jonathan Elliott, Joseph Hilbe

Maintainer: Rafael S. de Souza <rafael.2706@gmail.com>

See Also

CosmoPhotoZestimator, glmTrainPhotoZ, glmPredictPhotoZ, glm, bayesglm

Examples

```
## Not run:
# Load the data
data(PHAT0train)
data(PHAT0test)
# Run the analysis
photoZest <- CosmoPhotoZestimator(PHAT0train, PHAT0test, 6)
# This is considerably faster, but the results are not so good
# photoZest <- CosmoPhotoZestimator(PHAT0train, PHAT0test, 6, robust=FALSE)
```

```
# Create a boxplot showing the results
plotDiagPhotoZ(photoz = photoZest, specz = PHAT0test$redshift, type = "box")

## End(Not run)
```

---

computeCombPCA *Combined PCA for training and test sample*

Description

computeCombPCA computes combined PCA projections of the training and test samples.

Usage

```
computeCombPCA(x, y, robust)
```

Arguments

x	a matrix or a data.frame
y	a matrix or a data.frame
robust	a boolean indicating if robust PCA should be used or not

Details

The program is a simple alteration of PCAgrid() that computes a desired number of robust principal components using the grid search algorithm in the plane.

Value

PCA projections for each matrix

Author(s)

Rafael S. de Souza, Alberto Krone-Martins

Examples

```
#Multivariate data with outliers
library(mvtnorm)
x <- rbind(rmvnorm(100, rep(0, 6), diag(c(5, rep(1,5)))),
          rmvnorm( 15, c(0, rep(20, 5)), diag(rep(1, 6))))
y <- rbind(rmvnorm(100, rep(0, 6), diag(c(5, rep(1,5)))),
          rmvnorm( 15, c(0, rep(20, 5)), diag(rep(1, 6))))
#Here we calculate the principal components
pc <- computeCombPCA(x, y)
```

computeDiagPhotoZ *Simple diagnostics for the photometric redshift results*

Description

computeDiagPhotoZ computes a list of simple summary statistics for the photometric redshift estimation.

Usage

```
computeDiagPhotoZ(photoz, specz)
```

Arguments

photoz	vector
specz	vector

Value

a list containing the summary statistics

Author(s)

Rafael S. de Souza, Alberto Krone-Martins

Examples

```
# First, generate some mock data
ppo <- runif(1000, min=0.1, max=2)
ppo_ph <- rnorm(length(ppo), mean=ppo, sd=0.05)

# Now, compute the summary stats
computeDiagPhotoZ(ppo_ph, ppo)
```

CosmoPhotoZestimator *Photometric redshift estimation from a training dataset and a test dataset*

Description

CosmoPhotoZestimator returns photometric redshift estimated from photometric data and a training dataset with photometry and spectroscopy. The estimation is based on generalized linear models (see glmTrainPhotoZ and glmPredictPhotoZ).

Usage

```
CosmoPhotoZestimator(trainData, testData, numberOfPcs, method, family, robust)
```

Arguments

trainData	vector containing spectroscopic redshift data and photometry (at least one column shall be called redshift)
testData	vector containing spectroscopic redshift data and photometry (at least one column shall be called redshift)
numberOfPcs	an integer indicating the number of principal components to consider
method	a string containing the chosen GLM method. Two options are available: Frequentist will use the function glm from the package stats; Bayesian will use the function bayesglm from the package arm
family	a string containing gamma or inverse.gaussian (a description of the error distribution and link function to be used in the model)
robust	a boolean indicating if robust PCA should be used or not

Value

a vector with the estimated photometric redshifts

Author(s)

Alberto Krone-Martins, Rafael S. de Souza

Examples

```
## Not run:
# Load the data
data(PHAT0train)
data(PHAT0test)

# Run the analysis
photoZest <- CosmoPhotoZestimator(PHAT0train, PHAT0test, 6)

# Create a boxplot showing the results
plotDiagPhotoZ(photoz = photoZest, specz = PHAT0test$redshift, type = "box")

## End(Not run)
```

---

glmPredictPhotoZ *Predict photometric redshifts using a given glm fit object*

Description

glmPredictPhotoZ computes a list of simple summary statistics for the photometric redshift estimation.

Usage

```
glmPredictPhotoZ(data, train)
```

Arguments

data	a data.frame containing the data one wished to compute the redshift
train	a trained glm object containing the fit of the model

Value

list containing the results of the redshift estimation

Author(s)

Rafael S. de Souza, Alberto Krone-Martins

Examples

```
## Not run:
# Load the data
data(PHAT0train)
data(PHAT0test)

# Combine the training and test data and calculate the principal components
PC_comb <- computeCombPCA(subset(PHAT0train, select=c(-redshift)),
           subset(PHAT0test, select=c(-redshift)),
           robust=FALSE) # robust is false here just to make it faster
Trainpc <- cbind(PC_comb$x, redshift=PHAT0train$redshift)
Testpc <- PC_comb$y

# Fitting
Fit <- glmTrainPhotoZ(Trainpc, formula=redshift~poly(Comp.1,2)*
           poly(Comp.2,2)*Comp.3*Comp.4*Comp.5*Comp.6,
           method="Bayesian", family="gamma")

# Perform the photo-z estimation using the glmPredictPhotoZ function
photoz <- glmPredictPhotoZ(data=Testpc, train=Fit$glmfit)
specz <- PHAT0test$redshift

# Show a plot with the results
plotDiagPhotoZ(photoz$photoz, specz, "box")

## End(Not run)
```

---

glmTrainPhotoZ *Fit a glm for photometric redshift estimation*

Description

glmTrainPhotoZ trains a generalized linear model for photometric redshift estimation.

Usage

```
glmTrainPhotoZ(x, formula, method, family)
```

Arguments

x	a data.frame containing the data to train the model
formula	an object of class "formula" to be adopted
method	a string containing the chosen glm method. Two options are available: Frequentist will use the function glm from the package stats; Bayesian will use the function bayesglm from the package arm.
family	a string containing gamma or inverse.gaussian (a string with a description of the error distribution and link function to be used in the model).

Details

The program is a simple alteration of glm() and bayesglm(). The new arguments here are: Frequentist, Bayesian.

Value

a trained glm object containing the fit of the model

Author(s)

Rafael S. de Souza, Alberto Krone-Martins

Examples

```
## Not run:
# Load the data
data(PHAT0train)
data(PHAT0test)

# Combine the training and test data and calculate the principal components
PC_comb <- computeCombPCA(subset(PHAT0train, select=c(-redshift)),
           subset(PHAT0test, select=c(-redshift)),
           robust=FALSE) # robust is false here just to make it faster
Trainpc <- cbind(PC_comb$x, redshift=PHAT0train$redshift)
Testpc <- PC_comb$y

# Fitting
Fit <- glmTrainPhotoZ(Trainpc, formula=redshift~poly(Comp.1,2)*
           poly(Comp.2,2)*Comp.3*Comp.4*Comp.5*Comp.6,
           method="Bayesian", family="gamma")

# Perform the photometric redshift estimation
photoz <- predict(Fit$glmfit, newdata=Testpc, type="response")
specz <- PHAT0test$redshift

# Show a plot with the results
```

```
plotDiagPhotoZ(photoz, specz, "box")

## End(Not run)
```

PHAT0test *PHAT0 test dataset*

Description

The CosmoPhotoz package includes the dataset PHAT0test, containing photometric data in the astronomical photometric bands u, g, r, i, z, Y, J, H, K, IRAC_1, IRAC_2 and the redshift, for 161042 galaxies. The original PHAT0 dataset, containing 169520 was split in two (PHAT0train and PHAT0test).

Format

A dataframe with 161042 observations on the following on 12 variables, no NAs. Columns are named as follows:

redshift The redshift of the galaxy.

u The magnitude of the galaxy in the u band.

g The magnitude of the galaxy in the g band.

r The magnitude of the galaxy in the r band.

i The magnitude of the galaxy in the i band.

z The magnitude of the galaxy in the z band.

Y The magnitude of the galaxy in the Y band.

J The magnitude of the galaxy in the J band.

H The magnitude of the galaxy in the H band.

K The magnitude of the galaxy in the K band.

IRAC_1 The magnitude of the galaxy in the IRAC_1 band.

IRAC_2 The magnitude of the galaxy in the IRAC_2 band.

Source

Hildebrandt H., Arnouts S., Capak P., Moustakas L. A., Wolf C., Abdalla F. B., Assef R. J., Banerji M., et. al (2010), Astronomy&Astrophysics, v.523, p.A31

PHAT0train *PHAT0 train dataset*

Description

The CosmoPhotoz package includes the dataset PHAT0train, containing photometric data in the astronomical photometric bands u, g, r, i, z, Y, J, H, K, IRAC_1, IRAC_2 and the redshift, for 8478 galaxies. The original PHAT0 dataset, containing 169520 was split in two (PHAT0train and PHAT0test).

Format

A dataframe with 8478 observations on the following on 12 variables, no NAs. Columns are named as follows:

redshift The redshift of the galaxy.

u The magnitude of the galaxy in the u band.

g The magnitude of the galaxy in the g band.

r The magnitude of the galaxy in the r band.

i The magnitude of the galaxy in the i band.

z The magnitude of the galaxy in the z band.

Y The magnitude of the galaxy in the Y band.

J The magnitude of the galaxy in the J band.

H The magnitude of the galaxy in the H band.

K The magnitude of the galaxy in the K band.

IRAC_1 The magnitude of the galaxy in the IRAC_1 band.

IRAC_2 The magnitude of the galaxy in the IRAC_2 band.

Source

Hildebrandt H., Arnouts S., Capak P., Moustakas L. A., Wolf C., Abdalla F. B., Assef R. J., Banerji M., et. al (2010), Astronomy&Astrophysics, v.523, p.A31

plotDiagPhotoZ *Plot diagnostics for photometric redshift estimations*

Description

plotDiagPhotoZ returns diagnostic plots from the results of photometric redshifts. Different types of plots are available: a density plot of the error distribution (errordist), a predicted versus observed contour plot (predobs), violin plot showing the error distribution at different redshift bins (errorviolins) and a box plot showing the errors at each different redshift bin (box). The produced plots are returned as ggplot2 objects.

Usage

```
plotDiagPhotoZ(photoz, specz, type, npoints)
```

Arguments

photoz	vector containing photoz data
specz	vector containing spectroscopic redshift data
type	a string with one of the following values: errordist, predobs, errorviolins or box
npoints	an integer indicating how many points should be used to create the predobs plot (if 0, all points will be used)

Value

ggplot object

Author(s)

Rafael S. de Souza, Alberto Krone-Martins

Examples

```
# First, generate some mock data
ppo <- runif(1000, min=0.1, max=2)
ppo_ph <- rnorm(length(ppo), mean=ppo, sd=0.05)

# Then generate the plots
plotDiagPhotoZ(ppo_ph, ppo, type="errordist")
#plotDiagPhotoZ(ppo_ph, ppo, type="predobs")
#plotDiagPhotoZ(ppo_ph, ppo, type="errorviolins")
#plotDiagPhotoZ(ppo_ph, ppo, type="box")
```

Index