Parameter estimation and distribution selection by ExtDist

Haizhen Wu and A. Jonathan R. Godfrey

2014-10-17

Introduction

Parameter estimation and distribution selections are common tasks in statistical analysis. For example, in the context of variables acceptance sampling (see Wu and Govindaraju 2014 etc.), when the underlying distribution model of the quality characteristic is determined, the estimated quality of a product batch, which is measured by the proportiong nonconforming, is computed through the estimated parameter(s) of the underlying distribution based on a sample; on the other hand, if a collection of candidate distributions are considered to be eligible distributions, and when want to know which one can best describe the avaialbe data, then distribution selection functionality becomes necessary .

The **ExtDist** is devoted to provide a consistent and unified framework for these tasks.

```
require(ExtDist)
```

Parameter Estimation

Suppose we have a set of data, which is deemed generated from a Weibull distributed population,

```
head(X)
```

```
## [1] 0.1286348 0.6365761 0.6574366 0.2462515 0.7279375 0.2928906
```

It is possible we write a bunch of code to achieve a MLE estimation to the data. However, it is more convenient to use a single function to achieve this task.

```
(est.par <- eWeibull(X))
```

```
##
## Parameters for the Weibull distribution.
## (found using the  numerical.MLE method.)
##
##  Parameter  Type Estimate      S.E.
##      shape shape 1.767160 0.3255615
##      scale scale 2.867011 0.6099244
```

The $e-$ prefix we introduced in **ExtDist** is a logical extension to the $d-$, $p-$, $q-$, $r-$ prefixes of the distribution-related functions in R base package. Moreover, the output of $e-$ functions is defined as a S3 class object

```r
class(est.par)
```

```
## [1] "eDist"
```

The “eDist” object can be easily pluged into other $d-$, $p-$, $q-$, $r-$ functions in **ExtDist** to get the density, pencitile, quantile and random variables for distribution with estimated paramters.

```r
dWeibull(seq(0,2,0.4), params = est.par)
```

```
## [1] 0.00000000 1.09591880 0.59246913 0.25686509 0.10174247 0.03835279
```

```r
pWeibull(seq(0,2,0.4), params = est.par)
```

```
## [1] 0.0000000 0.3914879 0.7291509 0.8914692 0.9588224 0.9848941
```

```r
qWeibull(seq(0,1,0.2), params = est.par)
```

```
## [1] 0.0000000 0.2339313 0.4077962 0.6164455 0.9362328       Inf
```

```r
rWeibull(10, params = est.par)
```

```
##  [1] 0.1982688 0.2035324 2.4240519 0.4980837 0.5203063 0.1950049 0.3763785
##  [8] 0.2820143 0.6843038 0.2979968
```

To compatible with the convention, these functions also accept the paramters as individual argument, hence the following code are also eligible.

```r
dWeibull(seq(0,2,0.4), shape = est.par$shape, scale = est.par$scale)
pWeibull(seq(0,2,0.4), shape = est.par$shape, scale = est.par$scale)
qWeibull(seq(0,1,0.2), shape = est.par$shape, scale = est.par$scale)
rWeibull(10, shape = est.par$shape, scale = est.par$scale)
```

```
## [1] 0.00000000 1.09591880 0.59246913 0.25686509 0.10174247 0.03835279
## [1] 0.0000000 0.3914879 0.7291509 0.8914692 0.9588224 0.9848941
## [1] 0.0000000 0.2339313 0.4077962 0.6164455 0.9362328       Inf
##  [1] 0.5041854 0.2626802 0.5478566 1.3655725 0.9226692 0.2688135 1.9135509
##  [8] 1.0334979 0.8783582 0.7194805
```

The unified framework in **ExtDist** can help to construct functions/procedures with distributions becoming an argument. For example, if we want to construnct a function which can disply necessary results and plots of the parameter estimation, we can construct the follwoing function,

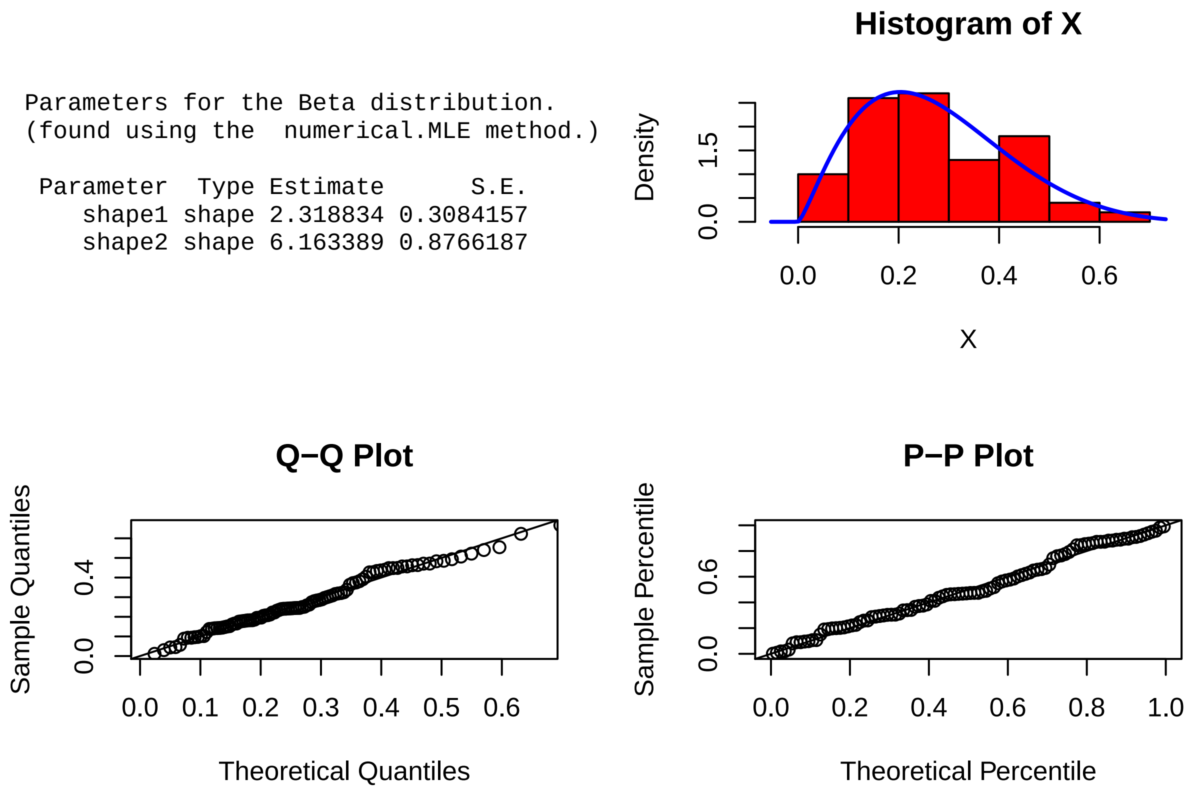
```r
fit_Dist <- function(X, Dist){
  l <- min(X); u <- max(X); d <- u-l; n <- length(X)

  est.par <- get(paste0("e",Dist))(X)
  dDist <- function(X) get(paste0("d",Dist))(X,param = est.par)
  pDist <- function(X) get(paste0("p",Dist))(X,param = est.par)
```

```
  qDist <- function(X) get(paste0("q",Dist))(X,param = est.par)

  op <- par(mfrow=c(2,2))
  PerformanceAnalytics::textplot(capture.output(print(est.par)), valign = "top")

  hist(X, col="red", probability=TRUE, xlim=c(l-0.1*d,u+0.1*d))
  curve(dDist, add=TRUE, col="blue", lwd=2)

  plot(qDist((1:n-0.5)/n), sort(X), main="Q-Q Plot", xlim = c(l,u), ylim = c(l,u),
       xlab="Theoretical Quantiles", ylab="Sample Quantiles")
  abline(0,1)

  plot((1:n-0.5)/n, pDist(sort(X)), main="P-P Plot", xlim = c(0,1), ylim = c(0,1),
       xlab="Theoretical Percentile", ylab="Sample Percentile")
  abline(0,1)

  par(op)
}
```

which can be used for aribitory data and distributions.

```
X <- rBeta(100,2,5)
fit_Dist(X, "Beta")
```

```
##
## Package PerformanceAnalytics (1.4.3541) loaded.
## Copyright (c) 2004-2014 Peter Carl and Brian G. Peterson, GPL-2 | GPL-3
## http://r-forge.r-project.org/projects/returnanalytics/
```



Distribution selection

As a S3 class object, several S3 methods have been developed in **ExtDist** to extract the distribution selection criteria and other relavent information.

```r
logLik(est.par) # log likihood
```

```
## [1] -21.69997
```

```r
AIC(est.par) # Akaike information criterion
```

```
## [1] 47.39994
```

```r
AICc(est.par) # corrected Akaike information criterion
```

```
## [1] 47.65526
```

```r
BIC(est.par) # Bayesian Information Criterion.
```

```
## [1] 51.22399
```

```r
MDL(est.par) # minimum description length
```

```
## [1] 24.0096
```

```r
vcov(est.par) # variance-covariance matrix of the parameters of the fitted distribution
```

```
##           shape     scale
## shape 0.1059903 0.1719569
## scale 0.1719569 0.3720078
```

Based on these criteria, for any sample, the best fitting distribution can be obtained from a list of candidate distributions.

```r
set.seed(1234)
X <- rBeta(50, shape1 = 2, shape2 = 10 )
bestDist(X, candDist = c("Beta_ab","Laplace","Normal"), criterion = "AIC")
```

```
## [1] "Beta_ab"
## attr(,"best.dist.par")
##
## Parameters for the Beta_ab distribution.
## (found using the  numerical.MLE method.)
##
##  Parameter     Type     Estimate         S.E.
##     shape1    shape 2.304770e+00 7.748262e-01
##     shape2    shape 5.719641e+03 1.273120e+05
##          a boundary 4.117003e-03 9.905024e-03
##          b boundary 2.737058e+02 6.083552e+03
##
##
## attr(,"criterion.value")
##   Beta_ab   Laplace    Normal
## -128.4569 -122.7168 -111.5958
```

When some time multiple crietia results are of interest for a list of candition distribution, a summary table can be output by using function DistSelCriteriaValues.

```r
set.seed(1234)
X <- rBeta(50, shape1 = 2, shape2 = 10 )
DistSelCriteriaValues(X, candDist = c("Beta_ab","Laplace","Normal"),
                         criteria = c("logLik","AIC","AICc","BIC","MDL"))
```

```
##        Beta_ab   Laplace   Normal
## logLik 68.22847  63.35842  57.79788
## AIC    -128.4569 -122.7168 -111.5958
## AICc   -127.5681 -122.4615 -111.3404
## BIC    -120.8089 -118.8928 -107.7717
## MDL    -77.76456 -53.04336 -48.38949
```

Weighted sample

Another notable feature of the **ExtDist** is that it can deal with weighted sample. In tranditional statisitical analysis, the sample are usually unweighted and the parameter estimation and distribution selection of tranditional functions do not have capibility of dealing with these problem under weighted sample situation.

The weighted sample, however, appear in many contexts, e.g.: in non-parametric and semi-parametric deconvolution (see e.g. Hazelton and Turlach 2010 etc.) the deconvoluted distribution can be represented as a pair (Y, w) where w is a vector of weights with same length as Y; in size-biased (unequal probability) sampling, the true population is more appropriately described by the weighted (with reciprocal of the inclusion probability as weights) observations rather than the observations themselves; in Bayesian inferences, the posterior distribution can be regarded as a weighted version of the prior distribution; the weighted distributions can also play an interesting role in stochastic population dynamics.

In **ExtDist**, the parameter estiamtion was conducted by maximum weighted likelihood estimation, with the estimate $\hat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta}$ being defined by

$$\hat{\boldsymbol{\theta}}^w = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} w_i \ln f(Y_i; \boldsymbol{\theta}), \tag{1}$$

where f is the density function of the ditstribution to be fitted.

For example, for a weighted sample with

```
Y <- c(0.1703, 0.4307, 0.6085, 0.0503, 0.4625, 0.479, 0.2695, 0.2744, 0.2713, 0.2177,
       0.2865, 0.2009, 0.2359, 0.3877, 0.5799, 0.3537, 0.2805, 0.2144, 0.2261, 0.4016)
w <- c(0.85, 1.11, 0.88, 1.34, 1.01, 0.96, 0.86, 1.34, 0.87, 1.34, 0.84, 0.84, 0.83, 1.09,
       0.95, 0.77, 0.96, 1.24, 0.78, 1.12)
```

the parameter estiamtion and distribution selection for weighted samples can be achieved by including the extra argument w:

```
eBeta(Y,w)
```

```
##
## Parameters for the Beta distribution.
## (found using the  numerical.MLE method.)
##
##  Parameter  Type Estimate      S.E.
##     shape1 shape 2.962998 0.8929558
##     shape2 shape 6.491242 2.0481425
```

```
bestDist(Y, w, candDist = c("Beta_ab","Laplace","Normal"), criterion = "AIC")
```

```
## [1] "Normal"
## attr(,"best.dist.par")
##
## Parameters for the Normal distribution.
## (found using the  numerical.MLE method.)
##
##  Parameter     Type  Estimate       S.E.
##       mean location 0.3149269 0.03112527
```

```
##         sd    scale 0.1391965 0.02200889
##
##
## attr(,"criterion.value")
##   Beta_ab   Laplace    Normal
## -14.76974 -17.67491 -18.11722
```

```
DistSelCriteriaValues(Y, w, candDist = c("Beta_ab","Laplace","Normal"),
                         criteria = c("logLik","AIC","AICc","BIC","MDL"))
```

```
##         Beta_ab   Laplace   Normal
## logLik 11.38487  10.83745  11.05861
## AIC    -14.76974 -17.67491 -18.11722
## AICc   -12.10308 -16.96903 -17.41134
## BIC    -10.78682 -15.68344 -16.12575
## MDL    -7.256094 -4.074517 -3.772566
```

References

Hazelton, Martin L., and Berwin A. Turlach. 2010. "Semiparametric Density Deconvolution." *Scandinavian Journal of Statistics* 37 (1) (March): 91–108.

Wu, Haizhen, and Kondaswamy Govindaraju. 2014. "Computer-Aided Variables Sampling Inspection Plans for Compositional Proportions and Measurement Error Adjustment." *Computers & Industrial Engineering* 72 (June): 239–246.