

# Package ‘EBCHS’

June 1, 2021

**Type** Package

**Title** An Empirical Bayes Method for Chi-Squared Data

**Version** 0.1.0

**Author** Lilun Du [aut, cre],  
Inchi Hu [aut]

**Maintainer** Lilun Du <dulilun@ust.hk>

**Description** We provide the main R functions to compute the posterior interval for the noncentrality parameter of the chi-squared distribution. The skewness estimate of the posterior distribution is also available to improve the coverage rate of posterior intervals. Details can be found in Du and Hu (2020) <[doi:10.1080/01621459.2020.1777137](https://doi.org/10.1080/01621459.2020.1777137)>.

**License** MIT + file LICENSE

**Encoding** UTF-8

**URL** <https://github.com/dulilun/EBCHS>

**RoxygenNote** 7.1.1

**Imports** stats, pracma, splines, fda

**Suggests** testthat

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2021-06-01 07:20:08 UTC

## R topics documented:

density_LS . . . . .	2
density_PLS . . . . .	2
EB_CS . . . . .	3
<b>Index</b>	<b>5</b>

---

 density\_LS

*log-density derivatives-parametric approach*


---

### Description

Assuming the log density of the chi-squared statistics admits a parametric form, this function estimates up to the fourth order log density derivatives.

### Usage

```
density_LS(x)
```

### Arguments

`x` a sequence of chi-squared test statistics

### Value

a list: the first-to-fourth log density derivatives

### Examples

```
p = 1000
k = 7
# the prior distribution for lambda
alpha = 2
beta = 10
# lambda
lambda = rep(0, p)
pi_0 = 0.8
p_0 = floor(p*pi_0)
p_1 = p-p_0
lambda[(p_0+1):p] = stats::rgamma(p_1, shape = alpha, rate=1/beta)
# Generate a Poisson RV
J = sapply(1:p, function(x){rpois(1, lambda[x]/2)})
X = sapply(1:p, function(x){rchisq(1, k+2*J[x])})
out = density_LS(X)
```

---

 density\_PLS

*Penalized least-squares method*


---

### Description

The semiparametric model is employed to estimate the log density derivatives of the chi-squared statistics.

**Usage**

```
density_PLS(x, qq)
```

**Arguments**

```
x          a sequence of chi-squared test statistics
qq         the quantiles used for splines
```

**Value**

a list: the first and second density derivatives

**Examples**

```
p = 1000
k = 7
# the prior distribution for lambda
alpha = 2
beta = 10
# lambda
lambda = rep(0, p)
pi_0 = 0.5
p_0 = floor(p*pi_0)
p_1 = p-p_0
lambda[(p_0+1):p] = stats::rgamma(p_1, shape = alpha, rate=1/beta)
# Generate a Poisson RV
J = sapply(1:p, function(x){rpois(1, lambda[x]/2)})
X = sapply(1:p, function(x){rchisq(1, k+2*J[x])})
qq = c(0.2, 0.4, 0.6, 0.8)
out = density_PLS(X, qq)
```

---

 EB\_CS

---

*Main function used in the paper (Du and Hu, 2020)*


---

**Description**

Give a sequence of chi-squared statistic values, the function computes the posterior mean, variance, and skewness of the noncentrality parameter given the data.

**Usage**

```
EB_CS(
  x,
  df,
  qq = c(0.2, 0.4, 0.6, 0.8),
  method = c("LS", "PLS", "g_model"),
  mixture = FALSE
)
```

**Arguments**

x	a sequence of chi-squared test statistics
df	the degrees of freedom
qq	the quantiles used in spline basis
method	LS: parametric least-squares; PLS: penalized least-squares; g-model: g-modeling
mixture	default is FALSE: there is no point mass at zero.

**Value**

a list: posterior mean, variance, and skewness estimates

**References**

Du and Hu (2020), *An Empirical Bayes Method for Chi-Squared Data*, *Journal of American Statistical Association*, forthcoming.

**Examples**

```
p = 1000
k = 7
# the prior distribution for lambda
alpha = 2
beta = 10
# lambda
lambda = rep(0, p)
pi_0 = 0.8
p_0 = floor(p*pi_0)
p_1 = p-p_0
lambda[(p_0+1):p] = rgamma(p_1, shape = alpha, rate=1/beta)
# Generate a Poisson RV
J = sapply(1:p, function(x){rpois(1, lambda[x]/2)})
X = sapply(1:p, function(x){rchisq(1, k+2*J[x])})
qq_set = seq(0.01, 0.99, 0.01)
out = EB_CS(X, k, qq=qq_set, method='LS', mixture = TRUE)
E = out$E_lambda
V = out$V_lambda
S = out$S_lambda
```

# Index

density\_LS, 2  
density\_PLS, 2

EB\_CS, 3