Package 'ChainLadder'

April 17, 2009

Type Package

Title Mack-, Bootstrap and Munich-chain-ladder methods for insurance claims reserving

Version 0.1.2-11

Date 2009-03-27

Author Markus Gesmann

Maintainer Markus Gesmann <markus.gesmann@gmail.com>

Description The package contains the Mack-, Munich-, and Bootstrap chain-ladder methods which are used in insurance claims reserving.

Depends Hmisc, lattice, stats

License GPL (>= 2)

URL http://code.google.com/p/chainladder/

LazyData yes

Repository CRAN

Date/Publication 2009-03-28 10:20:28

R topics documented:

`ChainLadder-package`

Various chain-ladder methods for claims reserving

Description

The ChainLadder-package grew out of presentations the author gave at the Stochastic Reserving Seminar at the Institute of Actuaries in November 2007 and 2008. This package has currently implementations for the Mack-, Munich- and Bootstrap-chain-ladder methods.

The ChainLadder-package comes with an example spreadsheet which demonstrates how to use how to use the `ChainLadder` functions in Excel. The spreadsheet is located in the Excel folder of the package. The R command `searchpaths()[grep('ChainLadder', searchpaths())]` will tell you the exact path to the directory. For the the spreadsheet you will need the RExcel-Addin, see `http://sunsite.univie.ac.at/rcom/` for more details.

More information and various presentations given on the ChainLadder-package are available on the project web site `http://code.google.com/p/chainladder/`

If you are also interested in loss distributions modeling, risk theory (including ruin theory), simulation of compound hierarchical models and credibility theory check out the `actuar` package by C. Dutang, V. Goulet and M. Pigeon.

Details

Package:	ChainLadder
Type:	Package
Version:	0.1.2
Date:	2008-10-12
License:	GPL version 2 or later

Author(s)

Markus Gesmann

Maintainer: Markus Gesmann <markus.gesmann@gmail.com>

References

Thomas Mack. Distribution-free calculation of the standard error of chain ladder reserve estimates. Astin Bulletin. *Vol. 23. No 2. 1993. pp.213:225*

Thomas Mack. The standard error of chain ladder reserve estimates: Recursive calculation and inclusion of a tail factor. Astin Bulletin. *Vol. 29. No 2. 1999. pp.361:366*

Gerhard Quarg and Thomas Mack. Munich Chain Ladder. Blatter DGVFM 26. *Munich. 2004.*

England, PD and Verrall, RJ. Stochastic Claims Reserving in General Insurance (with discussion). British Actuarial Journal 8. *III. 2002*

B. Zehnwirth and G. Barnett. Best Estimates for Reserves. Proceedings of the CAS. *Volume LXXXVII. Number 167.November 2000.*

Examples

```
RAA

MCL=MackChainLadder(RAA)
MCL
plot(MCL)

# BootChainLadder
B <- BootChainLadder(RAA, R=999, process.distr="gamma")
B
plot(B)
# fitdistribution
library(MASS)
# fit a log-normal distribution
fit <-  fitdistr(B$IBNR.Totals, "lognormal")
fit
plot(ecdf(B$IBNR.Totals))
curve(plnorm(x,fit$estimate["meanlog"], fit$estimate["sdlog"]), col="red", add=TRUE)

 # Munich Chain Ladder
 MCLpaid
 MCLincurred

 MCL = MunichChainLadder(MCLpaid, MCLincurred)
 MCL
 plot(MCL)
```

ABC *Run off triangle of accumulated claims data*

Description

Run-off triangle of a worker's compensation portfolio of a large company

Usage

```
data(ABC)
```

Format

A matrix with 11 accident years and 11 development years.

Source

B. Zehnwirth and G. Barnett. Best Estimates for Reserves. Proceedings of the CAS. *Volume LXXXVII. Number 167. November 2000.*

Examples

```
ABC
matplot(t(ABC), type="l")
```

BootChainLadder *Bootstrap-Chain-Ladder Model*

Description

The `BootChainLadder` procedure provides a predictive distribution of reserves or IBNRs for a cumulative claims development triangle.

Usage

```
BootChainLadder(Triangle, R = 999, process.distr=c("gamma", "od.pois"))
```

Arguments

`Triangle`	cumulative claims triangle. A (mxn)-matrix C_{ik} which is filled for $k \leq n+1-i; i = 1, \ldots, m; m \geq n$, see `qpaid` for how to use (mxn)-development triangles with m<n, say higher development period frequency (e.g quarterly) than origin period frequency (e.g accident years).
`R`	the number of bootstrap replicates.
`process.distr`	character string indicating which process distribution to be assumed. One of "gamma" (default), or "od.pois" (overdispersed Poisson), can be abbreviated

Details

The `BootChainLadder` function uses a two-stage bootstrapping/simulation approach. In the first stage an ordinary chain-ladder methods is applied to the cumulative claims triangle. From this we calculate the scaled Pearson residuals which we bootstrap R times to forecast future incremental claims payments via the standard chain-ladder method. In the second stage we simulate the process error with the bootstrap value as the mean and using the process distribution assumed. The set of reserves obtained in this way forms the predictive distribution, from which summary statistics such as mean, prediction error or quantiles can be derived.

Value

BootChainLadder gives a list with the following elements back:

`call`	matched call
`Triangle`	input triangle
`f`	chain-ladder factors
`simClaims`	array of dimension `c(m,n,R)` with the simulated claims
`IBNR.ByOrigin`	array of dimension `c(m,1,R)` with the modeled IBNRs by origin period
`IBNR.Triangles`	array of dimension `c(m,n,R)` with the modeled IBNR development triangles
`IBNR.Totals`	vector of R samples of the total IBNRs
`ChainLadder.Residuals`	adjusted Pearson chain-ladder residuals
`process.distr`	assumed process distribution
`R`	number of bootstraps

Note

The implimentation of `BootChainLadder` follows closely the discussion of the bootstrap model in section 8 and appendix 3 of the paper by England and Verall.

Author(s)

Markus Gesmann, ⟨markus.gesmann@gmail.com⟩

References

England, PD and Verrall, RJ. Stochastic Claims Reserving in General Insurance (with discussion), British Actuarial Journal *8, III. 2002*

Barnett and Zehnwirth. The need for diagnostic assessment of bootstrap predictive models, Insureware technical report. *2007*

See Also

See also `summary.BootChainLadder`, `MackChainLadder`

Examples

```
# See as well the example in section 8 of England & Verrall's paper on page 55.

B <- BootChainLadder(RAA, R=999, process.distr="gamma")
B
plot(B)
# Compare to MackChainLadder
MackChainLadder(RAA)
```

```
quantile(B, c(0.75,0.95,0.99, 0.995))

# fit a distribution to the IBNR
library(MASS)
plot(ecdf(B$IBNR.Totals))
# fit a log-normal distribution
fit <- fitdistr(B$IBNR.Totals[B$IBNR.Totals>0], "lognormal")
fit
curve(plnorm(x,fit$estimate["meanlog"], fit$estimate["sdlog"]), col="red", add=TRUE)

# See as well the ABC example in the Barnett and Zehnwirth paper
A <- BootChainLadder(ABC, R=999, process.distr="gamma")
A
plot(A, log=TRUE)
```

`GenIns` *Run off triangle of accumulated general insurance claims data*

Description

Run off triangle of claims data.

Usage

```
data(GenIns)
```

Format

A matrix with 10 accident years and 10 development years.

Source

TAYLOR, G.C. and ASHE, F.R. (1983) Second Moments of Estimates of Outstanding Claims. Journal of Econometrics ***23***, *37-61.*

References

See table 1 in: *Distribution-free Calculation of the Standard Error of Chain Ladder Reserve Estimates, Thomas Mack, 1993,* ASTIN Bulletin ***23***, *213 - 225*

Examples

```
GenIns
matplot(t(GenIns), type="l")
```

`MackChainLadder` *Mack-Chain-Ladder Model*

Description

The Mack-chain-ladder model forecasts reserves or IBNR (Incurred But Not Reported) claims based on a cumulative claims development triangle and estimates the standard error around it.

Usage

```
MackChainLadder(Triangle, weights = 1/Triangle, est.sigma="log-linear",
tail=FALSE, tail.se=NULL, tail.sigma=NULL)
```

Arguments

`Triangle`	cumulative claims triangle. A (mxn)-matrix C_{ik} which is filled for $k \leq n+1-i; i = 1, \ldots, m; m \geq n$, see `qpaid` for how to use (mxn)-development triangles with m<n, say higher development period frequency (e.g quarterly) than origin period frequency (e.g accident years).
`weights`	weights. Default: 1/Triangle, to use volume weighted chain ladder factors.
`est.sigma`	defines how to estimate $sigma_{n-1}$, the variability of the individual age-to-age factors at development time $n-1$. Default is "log-linear" for a log-linear regression, "Mack" for Mack's approximation from his 1999 paper. Alternatively the user can provide a numeric value.
`tail`	can be logical or a numeric value. If `tail=FALSE` no tail factor will be applied, if `tail=TRUE` a tail factor will be estimated via a linear extrapolation of $log(chainladder factors - 1)$, if `tail` is a numeric value than this value will be used as tail factor.
`tail.se`	defines how to estimate the standard error of the tail factor. Only needed if a tail factor > 1 is provided. Default is `NULL`. If `tail.se` is `NULL`, `tail.se` is estimated via "log-linear" regression, if `tail.se` is a numeric value than this value will be used instead.
`tail.sigma`	defines how to estimate individual tail variability. Only needed if a tail factor > 1 is provided. Default is `NULL`. If `tail.sigma` is `NULL`, `tail.sigma` is estimated via "log-linear" regression, if `tail.sigma` is a numeric value than this value will be used instead

Details

Following Mack's 1999 paper let C_{ik} denote the cumulative loss amounts of origin period (e.g. accident year) $i = 1, \ldots, m$, with losses known for development period (e.g. development year) $k \leq n+1-i$. In order to forecast the amounts C_{ik} for $k > n+1-i$ the Mack chain-ladder-model assumes:

$$\text{CL1: } E[F_{ik}|C_{i1}, C_{i2}, \ldots, C_{ik}] = f_k \text{ with } F_{ik} = \frac{C_{i,k+1}}{C_{ik}}$$

$$\text{CL2: } Var(\frac{C_{i,k+1}}{C_{ik}}|C_{i1}, C_{i2}, \ldots, C_{ik}) = \frac{\sigma_k^2}{w_{ik}C_{ik}^{\alpha}}$$

$$\text{CL3: } \{C_{i1}, \ldots, C_{in}\}, \{C_{j1}, \ldots, C_{jn}\}, \text{ are independent for origin period } i \neq j$$

with $w_{ik} \in [0; 1]$, $\alpha \in \{0, 1, 2\}$. If these assumptions are hold, the Mack-chain-ladder-model gives an unbiased estimator for IBNR (Incurred But Not Reported) claims.

The Mack-chain-ladder model can be regarded as a weighted linear regression through the origin for each development period: `lm(y ~ x + 0, weights=1/x)`, where `y` is the vector of claims at development period $k + 1$ and `x` is the vector of claims at development period k.

Value

MackChainLadder returns a list with the following elements

`call`	matched call
`Triangle`	input triangle of cumulative claims
`FullTriangle`	forecasted full triangle
`Models`	linear regression models for each development period
`f`	chain-ladder-link-ratios
`f.se`	standard errors of the chain-ladder-link-ratios f (assumption CL1)
`F.se`	standard errors of the true chain-ladder-link-ratios F_{ik} (square root of the variance in assumption CL2)
`sigma`	sigma parameter in CL2
`Mack.ProcessRisk`	variability in the projection of future losses not explained by the variability of the link ratio estimators (unexplained variation)
`Mack.ParameterRisk`	variability in the projection of future losses explained by the variability of the link-ratio estimators alone (explained variation)
`Mack.S.E`	total variability in the projection of future losses by the chain ladder method; the square root of the mean square error of the chain ladder estimate: Mack.S.E.^{2} = Mack.ProcessRisk^2 + Mack.ParameterRisk^2
`Total.Mack.S.E`	total variability of projected loss for all origin years combined
`tail`	tail factor used. If tail was set to TRUE the output will include the linear model used to estimate the tail factor

Note

Additional references for further reading:

England, PD and Verrall, RJ. Stochastic Claims Reserving in General Insurance (with discussion), British Actuarial Journal *8, III. 2002*

Murphy, Daniel M. Unbiased Loss Development Factors. Proceedings of the Casualty Actuarial Society Casualty Actuarial Society - *Arlington, Virginia 1994: LXXXI 154-222.*

Zehnwirth and Barnett. Best estimates for reserves. Proceedings of the CAS, LXXXVI I(167), November 2000.

Author(s)

Markus Gesmann ⟨markus.gesmann@gmail.com⟩

References

Thomas Mack. Distribution-free calculation of the standard error of chain ladder reserve estimates. Astin Bulletin. *Vol. 23. No 2. 1993. pp.213:225*

Thomas Mack. The standard error of chain ladder reserve estimates: Recursive calculation and inclusion of a tail factor. Astin Bulletin. *Vol. 29. No 2. 1999. pp.361:366*

See Also

See also `qpaid`, `summary.MackChainLadder`, `plot.MackChainLadder`, `residuals.MackChainLadder`, `MunichChainLadder`, `BootChainLadder`,

Examples

```
# See the Taylor/Ashe example in Mack's 1993 paper
GenIns
GNI <- MackChainLadder(GenIns, est.sigma="Mack")
GNI$f
GNI$sigma^2
GNI # compare to table 2 and 3 in Mack's 1993 paper
plot(GNI)

# See the example in Mack's 1999 paper
Mortgage
MRT <- MackChainLadder(Mortgage, tail=1.05, tail.sigma=71, tail.se=0.02, est.sigma="Mack")
MRT
# Table 1 in the above paper
MRT$f
MRT$f.se
MRT$F.se[3,]
MRT$sigma
plot(MRT) # We observe trends along calendar years.

# Table 2 in the above paper
MRT$FullTriangle[,9]/1000 ## C_{i9}
MRT$FullTriangle[,10]/1000 ## C_{i,ult}
MRT$Mack.S.E[,9]/1000 ## s.e.(C_{i9})

# Access process risk error
MRT$Mack.ProcessRisk

# Access parameter risk error
MRT$Mack.ParameterRisk

# Total risk
MRT$Mack.S.E
```

```
op <- par(mfrow=c(2,1))
plot(with(summary(MRT)$ByOrigin, Mack.S.E/Ultimate),t="l",
ylab="CV(Ultimate)", xlab="origin period")
plot(summary(MRT)$ByOrigin[["CV(IBNR)"]], t="l", ylab="CV(IBNR)",
xlab="origin period")
par(op)


# This data set is discussed many papers, e.g. England and Verall 2000
# paper, see Table 1 just there
RAA
R <- MackChainLadder(RAA)
R
plot(R)
# Table 12 in England and Verall 2000 paper
R$f
R$sigma^2
# Table 13 in England and Verall 2000 paper
# Please note the different indexing of sigma
MackChainLadder(RAA, est.sigma=R$sigma[7])
# Table 14 in England and Verall 2000 paper
MackChainLadder(RAA, est.sigma=R$sigma[8])

# Use lattice to plot the individual origin years
longRAA <- expand.grid(origin=as.numeric(dimnames(RAA)$origin), dev=as.numeric(dimnames(RAA)
longRAA$value <- as.vector(R$FullTriangle)
longRAA$valuePlusMack.S.E <-  longRAA$value + as.vector(R$Mack.S.E)
longRAA$valueMinusMack.S.E <- longRAA$value - as.vector(R$Mack.S.E)
library(lattice)
xyplot(valuePlusMack.S.E + valueMinusMack.S.E + value ~ dev |
factor(origin), data=longRAA, t="l", col=c("red","red","black"), as.table=TRUE)

# Different weights
# Using 1/Triangle^2 as weight will use mean chain ladder ratios
MackChainLadder(RAA, weights=1/RAA^2)$f
apply(RAA[,-1]/RAA[,-10],2, mean, na.rm=TRUE)

# Let's investigate the Mack model more detail
R[["Models"]][[1]]   # Model for first development period
summary( R[["Models"]][[1]]) # Look at the model stats
op <- par(mfrow=c(2,2)) # plot residuals
  plot( R[["Models"]][[1]])
par(op)

# Let's include an intercept in our model
newModel <- update(R[["Models"]][[1]], y ~ x+1,
             weights=1/R[["Triangle"]][1:9,1],
             data=data.frame(x=R[["Triangle"]][1:9,1],
                             y=R[["Triangle"]][1:9,2])
              )

# View the new model
summary(newModel)
```

```
op <- par(mfrow=c(2,2))
  plot( newModel )
par(op)

# Change the model for dev. period one to the newModel
R2 <- R
R2[["Models"]][[1]] <- newModel
predict(R2) # predict the full triangle with the new model
#(only the last origin year will be affected)

R2[["FullTriangle"]] <-  predict(R2)
R2[["FullTriangle"]]
R2   # Std. Errors have not been re-estimated!
# Plot the result

plot(R2, title="Changed R Model")

##
```

`MCLpaid` *Run off triangles of accumulated paid and incurred claims data.*

Description

Run-off triangles based on a fire portfolio

Usage

```
data(MCLpaid)
data(MCLincurred)
```

Format

A matrix with 7 origin years and 7 development years.

Source

Gerhard Quarg and Thomas Mack. Munich Chain Ladder. Blatter DGVFM. *26, Munich, 2004.*

Examples

```
MCLpaid
MCLincurred
op=par(mfrow=c(2,1))
matplot(t(MCLpaid), type="l")
matplot(t(MCLincurred), type="l")
par(op)
```

Mortgage *Run off triangle of accumulated claims data*

Description

Development triangle of a mortgage guarantee business

Usage

```
data(Mortgage)
```

Format

A matrix with 9 accident years and 9 development years.

Source

Competition Presented at a London Market Actuaries Dinner, D.E.A. Sanders, 1990

References

See table 4 in: *Distribution-free Calculation of the Standard Error of Chain Ladder Reserve Estimates, Thomas Mack, 1993*, ASTIN Bulletin ***23***, *213 - 225*

Examples

```
Mortgage
Mortgage
matplot(t(Mortgage), type="l")
```

MunichChainLadder *Munich-Chain-Ladder Model*

Description

The Munich-Chain-Ladder model forecasts IBNR claims based on a cumulative paid and incurred claims triangle. The Munich-Chain-Ladder model assumes that the Mack-model is applicable to the paid and incurred claims triangle, see MackChainLadder.

Usage

```
MunichChainLadder(Paid, Incurred,
                  est.sigmaP = "log-linear", est.sigmaI = "log-linear",
                  tailP=FALSE, tailI=FALSE)
```

Arguments

`Paid`	cumulative paid claims triangle. A (mxn)-matrix P_{ik} which is filled for $k \leq n+1-i; i = 1, \ldots, m; m \geq n$
`Incurred`	cumulative incurred claims triangle. A (mxn)-matrix I_{ik} which is filled for $k \leq n+1-i; i = 1, \ldots, m, m \geq n$
`est.sigmaP`	defines how $sigma_{n-1}$ for the Paid triangle is estimated, see `est.sigma` in `MackChainLadder` for more details, as `est.sigmaP` gets passed on to `MackChainLadder`
`est.sigmaI`	defines how $sigma_{n-1}$ for the Incurred triangle is estimated, see `est.sigma` in `MackChainLadder` for more details, as `est.sigmaI` is passed on to `MackChainLadder`
`tailP`	defines how the tail of the `Paid` triangle is estimated and is passed on to `MackChainLadder`, see `tail` just there.
`tailI`	defines how the tail of the `Incurred` triangle is estimated and is passed on to `MackChainLadder`, see `tail` just there.

Value

MunichChainLadder returns a list with the following elements

`call`	matched call
`Paid`	input paid triangle
`Incurred`	input incurred triangle
`MCLPaid`	Munich-chain-ladder forecasted full triangle on paid data
`MCLIncurred`	Munich-chain-ladder forecasted full triangle on incurred data
`MackPaid`	Mack-chain-ladder output of the paid triangle
`MackIncurred`	Mack-chain-ladder output of the incurred triangle
`PaidResiduals`	paid residuals
`IncurredResiduals`	incurred residuals
`QResiduals`	paid/incurred residuals
`QinverseResiduals`	incurred/paid residuals
`lambdaP`	dependency coefficient between paid chain ladder ratios and incurred/paid ratios
`lambdaI`	dependency coefficient between incurred chain ladder ratios and paid/incurred ratios
`qinverse.f`	chain-ladder-link-ratio of the incurred/paid triangle
`rhoP.sigma`	estimation of the conditional deviation around the paid/incurred ratios
`q.f`	chain-ladder-link-ratio of the paid/incurred triangle
`rhoI.sigma`	estimation of the conditional deviation around the incurred/paid ratios

Author(s)

Markus Gesmann ⟨markus.gesmann@gmail.com⟩

References

Gerhard Quarg and Thomas Mack. Munich Chain Ladder. Blatter DGVFM *26, Munich, 2004.*

See Also

See also `MackChainLadder`, `summary.MunichChainLadder`, `plot.MunichChainLadder`

Examples

```
MCLpaid
MCLincurred

M <- MunichChainLadder(MCLpaid, MCLincurred)
M
plot(M)
# You can access the standard chain ladder (Mack) output via
M$MackPaid
M$MackIncurred

# Following the example output in Quarg's paper:
MCL <- MunichChainLadder(MCLpaid, MCLincurred, est.sigmaP=0.1, est.sigmaI=0.1)
# Input triangles section 3.3.1
MCL$Paid
MCL$Incurred
# Parameters from section 3.3.2
# Standard chain ladder factors
MCL$MackPaid$f
MCL$MackIncurred$f
MCL$MackPaid$sigma
MCL$MackIncurred$sigma
# Check Mack's assumptions graphically
plot(MCL$MackPaid)
plot(MCL$MackIncurred)

MCL$q.f
MCL$rhoP.sigma
MCL$rhoI.sigma

MCL$PaidResiduals
MCL$IncurredResiduals

MCL$QinverseResiduals
MCL$QResiduals

MCL$lambdaP
MCL$lambdaI
# Section 3.3.3 Results
```

```
MCL$MCLPaid
MCL$MCLIncurred
```

`plot.BootChainLadder`

Plot method for a BootChainLadder object

Description

`plot.BootChainLadder`, a method to plot the output of `BootChainLadder`. It is designed to give a quick look at a `BootChainLadder` object, to check the result and the model assumptions.

Usage

```
## S3 method for class 'BootChainLadder':
plot(x, mfrow=c(2,2), title=NULL, log=FALSE, ...)
```

Arguments

`x`	output from `BootChainLadder`
`mfrow`	see `par`
`title`	see `title`
`log`	logical. If `TRUE` the y-axes of the 'latest incremental actual vs. simulated' plot will be on a log-scale
`...`	optional arguments to `plot` method

Details

`plot.BootChainLadder` shows four graphs, staring with a histogram of the total simulated IBNRs over all origin periods, including a rug plot; a plot of the empirical cumulative distribution of the total IBNRs over all origin periods; a box-whisker plot of simulated ultimate claims costs against origin periods; and a box-whisker plot of simulated incremental claims cost for the latest available calendar period against actual incremental claims in the same period. In the last plot the simulated data should follow the same trend as the actual, otherwise the original data might have some intrinsic trends which are not reflected in the model.

Note

The box-whisker plot of latest actual incremental claims against simulated claims follows ideas of Barnett and Zehnwirth in: *Barnett and Zehnwirth. The need for diagnostic assessment of bootstrap predictive models,* Insureware technical report. *2007*

Author(s)

Markus Gesmann

See Also

See Also BootChainLadder

Examples

```
B <- BootChainLadder(RAA)
plot(B)
```

`plot.MackChainLadder` *Plot method for a MackChainLadder object*

Description

`plot.MackChainLadder`, a method to plot the output of `MackChainLadder`. It is designed to give a quick look at a `MackChainLadder` object, to check the result and Mack's assumptions.

Usage

```
## S3 method for class 'MackChainLadder':
plot(x, mfrow=c(3,2), title=NULL, ...)
```

Arguments

`x`	output from `MackChainLadder`
`mfrow`	see `par`
`title`	see `title`
`...`	optional arguments to `plot` method

Details

`plot.MackChainLadder` shows six graphs, starting from the top left with a stacked bar-chart of the latest claims position plus IBNR and Mack's standard error by origin period; next to it on the right is a plot of the forecasted development patterns of all origin periods (numbered, starting with 1 for the oldest origin period), and 4 residual plots. The residual plots show the standardised residuals against fitted values, origin period, calendar period and development period. All residual plot should show no pattern or direction for Mack's method to be applicable. Pattern in any direction can be the result of trends. See *Zehnwirth and Barnett. Best estimates for reserves. Proceedings of the CAS, LXXXVI I(167), November 2000.* for more details on trends.

Author(s)

Markus Gesmann

See Also

See Also `MackChainLadder`, `residuals.MackChainLadder`

Examples

```
plot(MackChainLadder(RAA))
```

---

plot.MunichChainLadder *Plot method for a MunichChainLadder object*

Description

plot.MunichChainLadder, a method to plot the output of MunichChainLadder. It is designed to give a quick look at a MunichChainLadder object, to check the result and the correlation between the paid and incurred residuals.

Usage

```
## S3 method for class 'MunichChainLadder':
plot(x,  mfrow=c(2,2), title=NULL, ...)
```

Arguments

x	output from MunichChainLadder
mfrow	see par
title	see title
...	optional arguments to plot method

Details

plot.MunichChainLadder shows four plots, starting from the top left with a barchart of forecasted ultimate claims costs by Munich-chain-ladder (MCL) on paid and incurred data by origin period; the barchart next to it compares the ratio of forecasted ultimate claims cost on paid and incurred data based on the Mack-chain-ladder and Munich-chain-ladder methods; the two residual plots at the bottom show the correlation of (incurred/paid)-chain-ladder factors against the paid-chain-ladder factors and the correlation of (paid/incurred)-chain-ladder factors against the incurred-chain-ladder factors.

Note

The design of the plots follows those of Quarg's paper: *Gerhard Quarg and Thomas Mack. Munich Chain Ladder. Blatter DGVFM 26, Munich, 2004.*

Author(s)

Markus Gesmann

See Also

See Also as MunichChainLadder

Examples

```
M <- MunichChainLadder(MCLpaid, MCLincurred)
plot(M)
```

`predict.TriangleModel`
Prediction of a claims triangle

Description

The function is internally used by MackChainLadder to forecast future claims.

Usage

```
## S3 method for class 'TriangleModel':
predict(object,...)
```

Arguments

`object`	a list with two items: `Models`, `Triangle`
	`Models` list of linear models for each development period
	`Triangle` input triangle to forecast
`...`	not in use

Value

`FullTriangle`	forecasted claims triangle

Author(s)

Markus Gesmann

See Also

See Also MackChainLadder

Examples

```
RAA
MCL=MackChainLadder(RAA)
MCL
plot(MCL)
MCL[["Models"]][[1]]   # Model for first development period
summary( MCL[["Models"]][[1]]) # Look at the model stats
op=par(mfrow=c(2,2)) # plot residuals
  plot( MCL[["Models"]][[1]])
par(op)

# Let's include an intercept in our model
newModel <- update(MCL[["Models"]][[1]], y ~ x+1,
             weights=1/MCL[["Triangle"]][1:9,1],
             data=data.frame(x=MCL[["Triangle"]][1:9,1],
                             y=MCL[["Triangle"]][1:9,2])
              )

# view the new model
summary(newModel)
op=par(mfrow=c(2,2))
  plot( newModel )
par(op)

# change the model for dev. period one to the newModel
MCL2=MCL
MCL2[["Models"]][[1]] = newModel
predict(MCL2) # predict the full triangle with the new model
#(only the last origin year will be affected)

MCL2[["FullTriangle"]] <-  predict(MCL2)
MCL2[["FullTriangle"]]
MCL2   # Std. Errors have not been re-estimated!
# plot the res

plot(MCL2, title="Change MCL Model")
```

`qpaid` *Quarterly run off triangle of accumulated claims data*

Description

Sample data to demonstrate how to work with triangles with a higher development period frequency than origin period frequency

Usage

```
data(qpaid); data(qincurred)
```

Format

A matrix with 12 accident years and 45 development quarters of claims costs.

Source

Made up data for testing purpose

Examples

```
dim(qpaid)
dim(qincurred)
op=par(mfrow=c(1,2))
ymax <- max(c(qpaid,qincurred),na.rm=TRUE)*1.05
matplot(t(qpaid), type="l", main="Paid development",
                  xlab="Dev. quarter", ylab="$", ylim=c(0,ymax))
matplot(t(qincurred), type="l", main="Incurred development",
                      xlab="Dev. quarter", ylab="$", ylim=c(0,ymax))
par(op)
## MackChainLadder expects a quadratic matrix so let's expand
## the triangle to a quarterly origin period.
n <- ncol(qpaid)
Paid <- matrix(NA, n, n)
Paid[seq(1,n,4),] <- qpaid
M <- MackChainLadder(Paid)
plot(M)

# We expand the incurred triangle in the same way
Incurred <- matrix(NA, n, n)
Incurred[seq(1,n,4),] <- qincurred

# With the expanded triangles we can apply MunichChainLadder
MunichChainLadder(Paid, Incurred)

# In the same way we can apply BootChainLadder
# We reduce the resample size R from the default of 999 to 99 in this example purely to redu
BootChainLadder(Paid, R=99)
```

RAA *Run off triangle of accumulated claims data*

Description

Run-off triangle of Automatic Factultative business in General Liability

Usage

```
data(RAA)
```

Format

A matrix with 10 accident years and 10 development years.

Source

Historical Loss Development, Reinsurance Association of Ammerica (RAA), ***1991***, *p.96*

References

See Also: *Which Stochastic Model is Underlying the Chain Ladder Method?*, *Thomas Mack*, Insurance Mathematics and Economics, ***15, 2/3***, *pp133-138*, *1994*

P.D.England and R.J.Verrall, *Stochastic Claims Reserving in General Insurance*, British Actuarial Journal, ***Vol. 8***, *pp443-544*, *2002*

Examples

```
RAA
matplot(t(RAA), type="l")
```

---

residuals.MackChainLadder

Extract residuals of a MackChainLadder model

Description

Extract residuals of a `MackChainLadder` model by origin period, calendar period and development period.

Usage

```
## S3 method for class 'MackChainLadder':
residuals(object, ...)
```

Arguments

`object`	output of `MackChainLadder`
`...`	not in use

Value

The function returns a `data.frame` of residuals and standardised residuals by origin-, calendar- and development period.

Author(s)

Markus Gesmann

See Also

See Also MackChainLadder

Examples

```
RAA
MCL=MackChainLadder(RAA)
MCL

residuals(MCL)
```

---

summary.BootChainLadder

Methods for BootChainLadder objects

Description

summary, print, mean, and quantile methods for BootChainLadder objects

Usage

```
## S3 method for class 'BootChainLadder':
summary(object, probs=c(0.75,0.95), ...)

## S3 method for class 'BootChainLadder':
print(x, probs=c(0.75,0.95), ...)

## S3 method for class 'BootChainLadder':
quantile(x, probs=c(0.75, 0.95), na.rm = FALSE,
             names = TRUE, type = 7,...)

## S3 method for class 'BootChainLadder':
mean(x, ...)

## S3 method for class 'BootChainLadder':
residuals(object, ...)
```

Arguments

x, object	output from BootChainLadder
probs	numeric vector of probabilities with values in [0,1], see quantile for more help
na.rm	logical; if true, any NA and NaN's are removed from 'x' before the quantiles are computed, see quantile for more help

names	logical; if true, the result has a names attribute. Set to FALSE for speedup with many 'probs', see quantile for more help
type	an integer between 1 and 9 selecting one of the nine quantile algorithms detailed below to be used, see quantile
...	further arguments passed to or from other methods

Details

print.BootChainLadder calls summary.BootChainLadder and prints a formatted version of the summary. residuals.BootChainLadder gives the residual triangle of the expected chain-ladder minus the actual triangle back.

Value

summary.BootChainLadder, mean.BootChainLadder, and quantile.BootChainLadder, give a list with two elements back:

ByOrigin	data frame with summary/mean/quantile statistics by origin period
Totals	data frame with total summary/mean/quantile statistics for all origin period

Author(s)

Markus Gesmann

See Also

See Also BootChainLadder

Examples

```
B <- BootChainLadder(RAA, R=999, process.distr="gamma")
B
summary(B)
mean(B)
quantile(B, c(0.75,0.95,0.99, 0.995))
```

---

summary.MackChainLadder

Summary and print function for Mack-chain-ladder

Description

summary and print methods for a MackChainLadder object

Usage

```
## S3 method for class 'MackChainLadder':
summary(object, ...)

## S3 method for class 'MackChainLadder':
print(x, ...)
```

Arguments

x, object	object of class "MackChainLadder"
...	optional arguments to print or summary methods

Details

print.MackChainLadder calls summary.MackChainLadder and prints a formatted version of the summary.

Value

summary.MackChainLadder gives a list of two elements back

ByOrigin	data frame with Latest (latest actual claims costs), Dev.To.Date (chain ladder development to date), Ultimate (estimate ultimate claims cost), IBNR (estimated IBNR), Mack.S.E (Mack's estimation of the standard error of the IBNR), and CV(IBNR) (Coefficient of Variance, Mack.S.E/IBNR)
Totals	data frame of totals over all origin periods. The items follow the same naming convention as in ByOrigin above

Author(s)

Markus Gesmann

See Also

See Also MackChainLadder, plot.MackChainLadder

Examples

```
R <- MackChainLadder(RAA)
R
summary(R)
summary(R)$ByOrigin$Ultimate
```

summary.MunichChainLadder *Summary and print function for Munich-chain-ladder*

Description

summary and print methods for a MunichChainLadder object

Usage

```
## S3 method for class 'MunichChainLadder':
summary(object, ...)

## S3 method for class 'MunichChainLadder':
print(x, ...)
```

Arguments

x, object	object of class "MunichChainLadder"
...	optional arguments to print or summary methods

Details

print.MunichChainLadder calls summary.MunichChainLadder and prints a formatted version of the summary.

Value

summary.MunichChainLadder gives a list of two elements back

ByOrigin	data frame with *Latest Paid* (latest actual paid claims costs), *Latest Incurred* (latest actual incurred claims position), *Latest P/I Ratio* (ratio of latest paid/incurred claims), *Ult. Paid* (estimate ultimate claims cost based on the paid triangle), *Ult. Incurred* (estimate ultimate claims cost based on the incurred triangle),*Ult. P/I Ratio* (ratio of ultimate paid forecast / ultimate incurred forecast)
Totals	data frame of totals over all origin periods. The items follow the same naming convention as in ByOrigin above

Author(s)

Markus Gesmann

See Also

See Also MunichChainLadder, plot.MunichChainLadder

Examples

```
M <- MunichChainLadder(MCLpaid, MCLincurred)
M
summary(M)
summary(M)$ByOrigin
```

Index