

FULANO DE TAL



Mestrando em Fulanologia, Instituto de Assuntos Aleatórios, USP.

O título de minha tese é: “O encantamento possível: sub-semântica do viés enquanto multiplicidade de olhares cruzados”, orientado pelo Roberto Carlos (cantor e enredo de escola de samba, não o jogador).

Meus Exercícios

Linque para a página com os meus exercícios resolvidos [exec](#)

Proposta de Trabalho Final

Principal

Uma função que faça um gráfico dos valores contra os quantis teóricos da normal, como na função `qqnorm`, mas que também simule amostras de mesmo tamanho para criar um intervalo de confiança empírico para a linha do esperado.

A idéia é que se simulamos várias amostras e ordenamos os valores, podemos achar com a função `quantile` quais os intervalos que contêm uma certa fração das observações, para a primeira, segunda, terceira, n -ésima observação. Estes limites podem ser usados para delimitar uma espécie de “intervalo de confiança” simulado, que serviria para avaliar o grau de aderência da amostra a uma distribuição normal.

Plano B

Portar o [mussum grapher](#) para uma função do R ¹⁾.

Página de Ajuda

qqnorm.ci

package:nenhum

R Documentation

Gráfico dos quantis da normal com intervalo de confiança empírico

Description:

Produz um gráfico dos valores de um vetor em função dos quantis esperados pela distribuição normal. Adiciona ao gráfico uma linha de inclinação = 1, e também os limites do intervalo de confiança simulado.

Usage:

```
qqnorm.ci(x, alfa=0.9, nrep=1000, qqline=TRUE, ci=TRUE, ...)
```

Arguments:

x: Vetor numérico. Valores de uma amostra.
alfa: numérico. Probabilidade associada ao intervalo de confiança (nível de confiança).
nrep: número de randomizações para o cálculo do intervalo de confiança simulado (ver detalhes)
qqline: lógico. Acrescentar linha do esperado?
ci: lógico. Calcular o intervalo de confiança simulado?
...: qualquer argumento a repassar para a função plot.

Details:

Os valores da amostra (x) são ordenados e plotados contra os quantis esperados para uma distribuição normal com média e desvio padrão iguais aos da amostra. A linha esperada tem inclinação um e intercepto zero, e difere da obtida com a função 'qqline', que passa pelos segundo e terceiro quartis.
Para o intervalo de confiança, nrep amostras de mesmo tamanho de x são simuladas de uma distribuição normal de mesma média e variância de x. Os valores em cada amostra simulada são ordenados e o intervalo de confiança é aquele que inclui a proporção dos valores simulados definida por alfa. Estes intervalos são obtidos aplicando-se a função 'quantile' aos nrep valores simulados do primeiro, segundo, terceiro, enésimo valor do vetor ordenado.

Value:

Um gráfico é gerado. Um data frame é retornado mas não é exibido na tela, com os

seguintes vetores:

Obs : valores da amostra em ordem crescente

Percentis: os percentis correspondentes a cada elemento de x

Quantis: os quantis esperados pela distribuição normal
Lower, Upper: se o argumento `ci` é verdadeiro, os limites inferior e superior do intervalo

de confiança simulado.

Warning:

O intervalo simulado não é uma solução analítica e serve apenas para fins exploratórios.

Author(s):

Fulano de Tal

fulano@fulanonet.com.br

References:

Cleveland, W.S. (1994) The Elements of Graphing Data, Hobart Press ISBN 0-9634884-1-4

Wilk, M.B., Gnanadesikan, R. (1968) Probability plotting methods for the analysis of data.

Normal probability plot.

http://en.wikipedia.org/wiki/Normal_probability_plot

See Also:

'`qqnorm`' e '`qqline`' do pacote base, para o gráfico de quantil padrão do R, e a função

'`ppoints`', com a qual os percentis são calculados.

Para uma função similar a esta mas bem mais completa, veja '`qq.plot`', do pacote `car`.

Examples:

```
qqnorm.ci(rnorm(100)) ## Uma amostra de uma normal padronizada.
```

```
qqnorm.ci(rexp(100, rate=10)) ## Amostra de uma distribuição exponencial com média =10
```

```
valores <- qqnorm(rpois(30,5)) ## Cria um objeto com os valores do gráfico
```

Código da Função

```
qqnorm.ci <- function(x, alfa=0.90, nrep=1000, qqline=TRUE, ci=TRUE,...){  
  if(is.vector(x)==F | is.numeric(x)==F){  
    cat("\n x deve ser um vetor numérico \n")  
  }  
  media <- mean(x)  
  dp <- sd(x)  
  percentis <- ppoints(length(x))  
  quantis <- qnorm(percentis,mean=media,sd=dp)  
  plot(sort(x)~quantis,  
        xlab="Quantis Teóricos",ylab="Observado", ...)  
  results <- data.frame(Obs=sort(x),Percentile=percentis, Quantis= quantis)  
  if(qqline==TRUE){  
    abline(0,1, col="blue")  
  }  
  if(ci==TRUE){  
    rmat <- matrix(rnorm(nrep*length(x),mean=media,sd=dp),nrow=length(x))  
    rmat <- apply(rmat,2,sort)  
    lower <- apply(rmat,1,quantile,prob=(1-alfa)/2)  
    upper <- apply(rmat,1,quantile,prob=(1+alfa)/2)  
    lines(lower~quantis,col="red",lty=2)  
    lines(upper~quantis,col="red",lty=2)  
    results$Lower.CI <- lower  
    results$Upper.CI <- upper  
  }  
  invisible(results)  
}
```

Arquivo da Função

[qqnorm.ci](#)

1)

NOTA: alunos e professores da Disciplina agradecem ao Prof. Rodrigo S. Pereira pela indicação do utilitário Mussum Grapher.

From:
<http://ecor.ib.usp.br/> - **ecoR**

Permanent link:
http://ecor.ib.usp.br/doku.php?id=01_curso_atual:alunos:trabalho_final:fulano:start



Last update: **2020/08/20 16:10**