

R入門

限界状態設計, モンテカルロシミュ
レーション

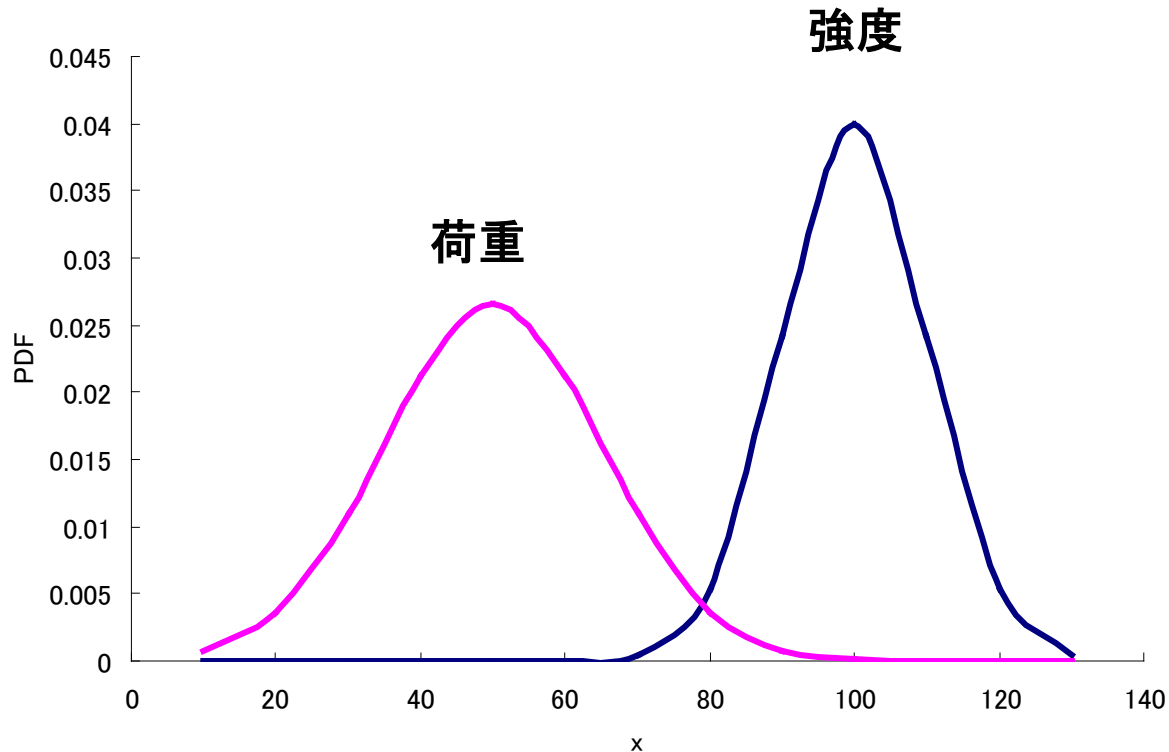
東京大学大学院

酒井信介

荷重と強度が正規分布するとき

課題:

荷重が $N(\mu_S, \sigma_S^2)$ 、強度が $N(\mu_R, \sigma_R^2)$ のとき破損確率を求めよ



正規分布関連コマンド

Rには種々の分布形表現があるが、分布形の名称がxxxであるとき、下記の約束事がある

dxxx() 確率密度関数、pxxx() 累積分布関数 qxxx() 確率点 rxxx() 乱数発生

xxxの代表的なものとしては、下記がある

norm 正規分布 lnorm 対数正規分布 weibull ワイブル分布

各コマンドのマニュアルを見たいときには下記のように?を使う

>?コマンド

例えば>?dnormと入力すると下記に対する説明が出ることを確認する事

```
dnorm(x, mean = 0, sd = 1, log = FALSE)
```

```
pnorm(q, mean = 0, sd = 1, lower.tail = TRUE, log.p = FALSE)
```

```
qnorm(p, mean = 0, sd = 1, lower.tail = TRUE, log.p = FALSE)
```

```
rnorm(n, mean = 0, sd = 1)
```

```
>dnorm(0) #標準正規確率密度関数のx=0→0.3989423
```

```
>pnorm(0) #標準正規累積分布関数のx=0→0.5
```

```
>qnorm(0.5) #標準正規分布でquantileが0.5→ x=0
```

```
>rnorm(10) #標準正規分布の乱数を10個発生する
```

分布名	コード名	パラメータ
ベータ分布	beta	shape1, shape2, ncp
二項分布	binom	size, prob
コーシー分布	cauchy	location, scale
カイ二乗分布	chisq	df, ncp
指数分布	exp	rate
F分布	f	df1, df2, ncp
ガンマ分布	gamma	shape, scale
幾何分布	geom	prob
超幾何分布	hyper	n, m, k
対数正規分布	lnorm	meanlog, sdlog
ロジスティック分布	logis	location, scale
多項分布	multinom	n, size, prob
負の二項分布	nbinom	size, prob
正規分布	norm	mean, sd
ポアソン分布	pois	lambda
ウィルコクソンの符号付順位和統計量の分布	signrank	m, n
t分布	t	df, ncp
一様分布	unif	min, max
スチューデント化された分布	tukey	nmeans, df
ワイブル分布	weibull	shape, scale
ウィルコクソンの順位和統計量の分布	wilcox	m, n

例題1

荷重 S が $N(\mu_S, \sigma_S^2)$ 、強度 R が $N(\mu_R, \sigma_R^2)$ に従うとき
 $R-S$ は $N(\mu_R - \mu_S, \sigma_R^2 + \sigma_S^2)$ に従う

$$P_f = F_{R-S}(0) = \Phi\left(-\frac{\mu_{R-S}}{\sigma_{R-S}}\right) = 1 - \Phi\left(\frac{\mu_{R-S}}{\sigma_{R-S}}\right)$$

$$\mu_{R-S} = \mu_R - \mu_S \quad \sigma_{R-S} = \sqrt{\sigma_R^2 + \sigma_S^2}$$

μ_R	σ_R	μ_S	σ_S
200	10	100	20

この条件での破損確率を評価するプログラムを書け

解答

まず、新しいプロジェクトExampleを作る

File -> New Fileにより、新しいプログラム用のタブが作成される。ここに下記プログラムを記入のこと。

```
#example 1
Ms<- 100; Ss=20
Mr<- 200; Sr=10
Mrs <- Mr-Ms
Srs <- sqrt(Sr*Sr+Ss*Ss)
Pf <- 1-pnorm(Mrs/Srs)
Pf
```

何気なく書いているこのプログラム。
変数宣言していない!何故このような
ことが可能なのか?実は、これはオブ
ジェクト指向の恩恵。

「<-」には重要な意味がある。初学者
はそのことを特に意識しなくともよい

これを実行するためには「Source」ボタンを押す。下記の値が出力されることを確認する事。

3.872108e-06

これが、この条件における破損確率

この値が確認されたら、プログラムを保存する。ファイル名はexample1.rとする。

例題2

$P_f=10^{-6}$ では過剰設計なので、 $P_f=10^{-3}$ 程度となるような強度材料を選択したい。強度の標準偏差は同一として、平均値がどの程度の材料を使えばよいか？

以下の関数を作ることにより、合理化できることを体験せよ

```
#example 2
PfCalc <- function(Mr){
  Ms<- 100; Ss=20
  Sr=10
  Mrs <- Mr-Ms
  Srs <- sqrt(Sr*Sr+Ss*Ss)
  Pf <- 1-pnorm(Mrs/Srs)
  Pf
}
```

PfCalc()を使って $P_f=10^{-3}$ となる μ_R を探す

関数の記述法

fnameという名前の関数定義

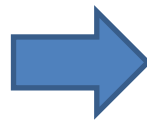
```
fname <- function(arg1,arg2,.....){  
  ....  
  ....  
}
```

利用法

```
fname(arg1,arg2,....)
```

使用例

```
Hollow <- function(rout,rin)  
a1 <- pi * rout * rout  
a2 <- pi * rin * rin  
return(a1-a2)  
}
```



外径100内径80の中空断面
の面積はHollow(100,80)

望ましい記述法

```
Hollow <- function(outer=200,inner=150){  
  ....  
}
```

呼び出し法

Hollow (inner=20)

Hollow(inner=30,outer=150)など

例題3

Pf= 10^{-3} となる μ_R を探索するプロセスをNewton Raphson法で合理化せよ

```
f1 <- function(x){  
  PfCalc(x)-1e-3  
}
```

uniroot(f1,c(100,200)) #100～200の範囲で、f1()が0となるxの解を探索する

「source」ボタンの実行により、先に、求めた解が一瞬で出ることを確認せよ

このように目標値(Pf= 10^{-3})を与えて、これを満足する設計パラメータを決める考え方を信頼性設計と呼ぶ

例題4

例題3で得られた結果についてモンテカルロシミュレーションにより確認せよ。サンプルサイズは100000とする。

```
#example 4
n <- 100000
R <- rnorm(n,mean=169.0997,sd=10)
S <- rnorm(n,mean=100,sd=20)
failure <- 0
for(i in 1:n){
  if(R[i] < S[i])failure <- failure + 1
}
Pf <- failure/n
Pf
```

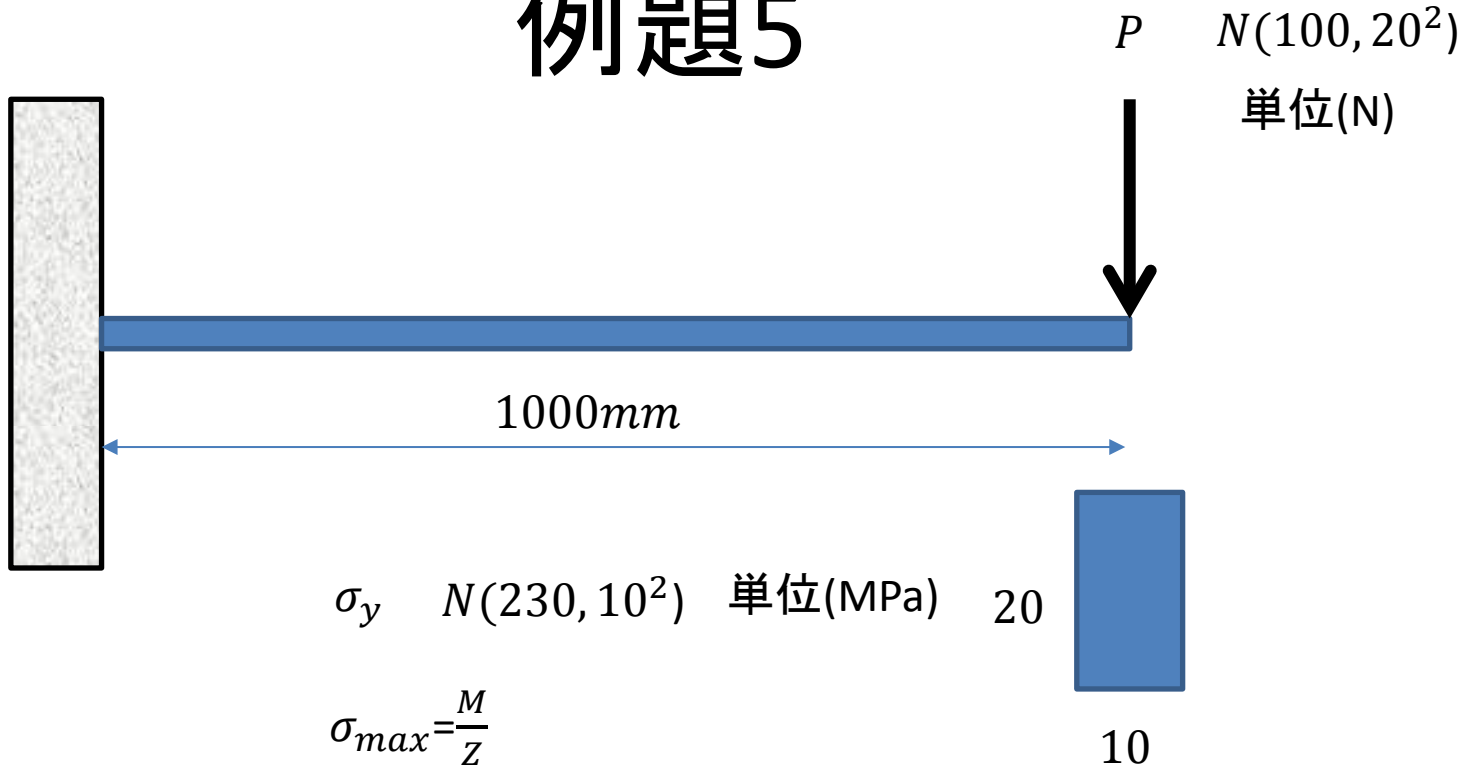
配列宣言も不要!!

$Pf=10^{-3}$ となることを確認せよ

乱数のチェック法

```
library(rgl)
dd<-runif(3*2000)
dd<-matrix(dd,ncol=3)
x<-dd[,1]; y<-dd[,2]; z<-dd[,3]
plot3d(x,y,z)
```

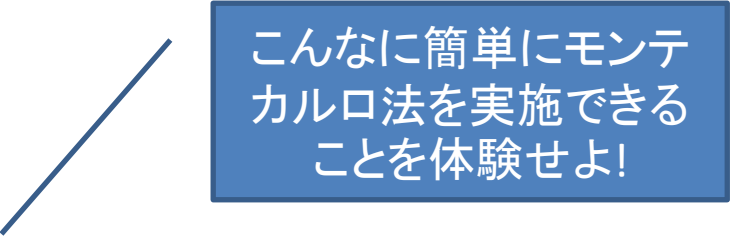
例題5



- ・この梁の、支持部に発生する応力分布のヒストグラムを表示せよ
- ・降伏する確率をモンテカルロ法で求めよ

解答例

```
# example 5
b<- 10; h<- 20
Z <- b*h*h / 6
Smax <- function(x){
  x*1000/Z
}
n <- 100000
P <- rnorm(n,mean=100,sd=20)
S <- Smax(P)
hist(S,main="example 5",xlab="S(MPa)")
R <- rnorm(n,mean=230,sd=10)
failure <- 0
for(i in 1:n){
  if(S[i]> R[i])failure <- failure+1
}
Pf <- failure/n
Pf
```



こんなに簡単にモンテ
カルロ法を実施できる
ことを体験せよ!

信頼性工学サポートパッケージ

```
>library(devtools)
>install_github("ShinsukeSakai0321/LimitState")
>library(LimitState)
```

```
>aa<-reliability(g = "R-S", var = c("R", "S"), dist =
c("normal", "normal"), muX = c(200, 100),
sigmaX = c(10, 20))
>aa<$GetPoF()
3.872108e-06
```