

クレジット:

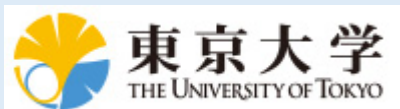
UTokyo Online Education 数理手法Ⅶ 2019 北川源四郎

ライセンス:

利用者は、本講義資料を、教育的な目的に限ってページ単位で利用することができます。特に記載のない限り、本講義資料はページ単位でクリエイティブ・コモンズ 表示-非営利-改変禁止 ライセンスの下に提供されています。

<http://creativecommons.org/licenses/by-nc-nd/4.0/>

本講義資料内には、東京大学が第三者より許諾を得て利用している画像等や、各種ライセンスによって提供されている画像等が含まれています。個々の画像等を本講義資料から切り離して利用することはできません。個々の画像等の利用については、それぞれの権利者の定めるところに従ってください。



R_code(13)_2019.txt

```
#####  
# 第15章  
#####  
#####  
# 粒子フィルタの1サイクル #  
#####  
m <- 100  
tau <- 0.1  
sig <- 1.0  
par(mar=c(2,2,1,1)+0.1)  
par(mfrow=c(5,1))  
alpha0 <- rep(1/m,length=m)  
#  
# Initial distribution  
xf <- rnorm(m,mean=0,sd=1)  
plot(xf,alpha0,type="h",xlim=c(-3,4))  
  
# Prediction step  
v <- rnorm(m,mean=0,sd=tau)  
xp <- xf + v  
plot(v,alpha0,type="h",xlim=c(-3,4))  
plot(xp,alpha0,type="h",xlim=c(-3,4))  
  
# Bayes factor (weights of particles)  
y <- 0.3  
alpha <- dnorm(y-xp,mean=0,sd=sig,log=FALSE)  
alpha <- alpha/sum(alpha)  
# predictive, filter distributions  
plot(xp,alpha,type="h",xlim=c(-3,4))  
  
# re-sampling  
xf1 <- sample(xp,prob=alpha,replace=T)  
plot(xf1,alpha0,type="h",xlim=c(-3,4))  
  
#####  
# TEST data  
y <- as.ts(read.csv("Chap83B_new2.csv"))  
par(mar=c(2,2,1,1)+0.1)  
plot(y,ylim=c(-4,4))
```

R_code(13)_2019.txt

```
#####  
#  
# Particle Filter (平滑化は行っていない)  
#  
  
model <- 1  
m <- 10000  
  
# Gauss model  
if (model == 1){  
tau2 <- 0.018  
sig2 <- 1.045  
tau <- sqrt(tau2)  
sig <- sqrt(sig2)  
}  
# Cauchy model  
if (model == "2"){  
tau22 <- 0.0000355  
sig22 <- 1.006  
scal2 <- sqrt(tau22)  
sig <- sqrt(sig22)  
}  
  
# Initial distribution  
xf <- rnorm(m,mean=0,sd=2)  
n <- length(y)  
# trend <- rep(0,length=n)  
trend <- matrix(0,nrow=n,ncol=3)  
  
xs <- matrix( ncol=m, nrow=lag+1 )  
  
for (ii in 1:n) {  
# Prediction step  
if ( model == 1 ) {v <- rnorm(m,mean=0,sd=tau) }  
if ( model == 2 ) {v <- rcauchy(m,location=0,scale=tau) }  
xp <- xf + v  
  
# Bayes factor (weights of particles)  
alpha <- dnorm(y[ii]-xp,mean=0,sd=sig,log=FALSE)  
alpha <- alpha/sum(alpha)  
  
# re-sampling
```

R_code(13)_2019.txt

```
xf <- sample(xp,prob=alpha,replace=T)

#qxf <- quantile(xf)
#trend[ii] <- mean(qxf)
trend[ii,1] <- quantile(xf,probs=0.1)
trend[ii,2] <- quantile(xf,probs=0.5)
trend[ii,3] <- quantile(xf,probs=0.9)
}

plot(y,type="l",ylim=c(-3,3))
par(new=T)
plot(trend[,1],type="l",col="blue",lwd=1,ylim=c(-3,3))
par(new=T)
plot(trend[,2],type="l",col="red",lwd=2,ylim=c(-3,3))
par(new=T)
plot(trend[,3],type="l",col="blue",lwd=1,ylim=c(-3,3))

#####
# TEST data
y <- as.ts(read.csv("Chap83B_new2.csv"))
par(mar=c(2,2,1,1)+0.1)
plot(y,ylim=c(-4,4))

#####
#
# Particle smoother
#
model <- 1
m <- 10000
lag <- 20

# Gauss model
if (model == 1){
tau2 <- 0.018
sig2 <- 1.045
tau <- sqrt(tau2)
sig <- sqrt(sig2)
}
# Cauchy model
if (model == "2"){
```

R_code(13)_2019.txt

```
tau22 <- 0.0000355
sig22 <- 1.006
scal2 <- sqrt(tau22)
sig <- sqrt(sig22)
}

# Initial distribution
xf <- rnorm(m,mean=0,sd=1)
n <- length(y)
trend <- matrix( nrow=n, ncol=7 )
xs <- matrix( nrow=m, ncol=lag+1 )
#strend <- rep(0,length=n)
strend <- matrix( nrow=n, ncol=7 )

for (ii in 1:n) {
#
# Prediction step
if ( model == 1 ) {v <- rnorm(m,mean=0,sd=tau)}
if ( model == 2 ) {v <- rcauchy(m,location=0,scale=scal2)}

xp <- xf + v

# Bayes factor (weights of particles)
alpha <- dnorm(y[ii]-xp,mean=0,sd=sig,log=FALSE)
alpha <- alpha/sum(alpha)

# re-sampling
# xf <- sample(xp,prob=alpha,replace=T)
ind <- 1:m
jnd <- sample(ind,prob=alpha,replace=T)
for (i in 1:m){
  xf[i] <- xp[jnd[i]]
  for (j in 1:lag) {
    xs[i,lag+2-j] <- xs[jnd[i],lag+1-j]
  }
  xs[i,1] <- xf[i]
}

# trend[ii] <- mean(xf)
# trend[ii,4] <- mean(xs[,lag+1])
trend[ii,3] <- quantile(xs[,lag+1], prob=0.1, na.rm=TRUE)
```

R_code(13)_2019.txt

```
trend[ii,4] <- quantile(xs[,lag+1], prob=0.5, na.rm=TRUE)
trend[ii,5] <- quantile(xs[,lag+1], prob=0.9, na.rm=TRUE)
}

for (ii in lag+1:n) {
  strend[ii-lag,4] <- trend[ii,4]
  strend[ii-lag,3] <- trend[ii,3]
  strend[ii-lag,5] <- trend[ii,5]
}
for (ii in 1:lag) {
#strend[n+1-ii,4] <- mean(xs[,ii])
  strend[n+1-ii,3] <- quantile(xs[,ii], prob=0.1, na.rm=TRUE)
  strend[n+1-ii,4] <- quantile(xs[,ii], prob=0.5, na.rm=TRUE)
  strend[n+1-ii,5] <- quantile(xs[,ii], prob=0.9, na.rm=TRUE)
}

#plot(y,type="l",ylim=c(-3,3))
#par(new=T)
plot(strend[,3],type="l",col="blue",lwd=1,ylim=c(-3,3))
par(new=T)
plot(strend[,4],type="l",col="red",lwd=1,ylim=c(-3,3))
par(new=T)
plot(strend[,5],type="l",col="blue",lwd=1,ylim=c(-3,3))
```