

生物資源解析学演習 Lecture-7&8 Handout

最尤法～回帰分析～プロダクションモデル

海洋生物資源学科北門利英

2019年12月3日 + 17日

目次

0.1	目的	1
0.2	用語	1
0.3	R の関数（復習含む）	1
1	独立かつ同一な確率分布に従う観測値に基づく正規分布の平均と標準偏差の推定	2
1.1	正規分布に関する R の関数	2
1.2	R による確率密度の計算	2
1.3	R による乱数の生成	3
1.4	平均値 μ と標準偏差 $\log \sigma$ に対する (負の) 対数尤度関数の定義	3
1.5	平均値 μ と標準偏差 $\log \sigma$ に対する (負の) 対数尤度関数の最適化	4
2	独立ではあるか同一ではない確率分布に従う観測値に基づく回帰係数と標準偏差の推定—線形回帰—	6
2.1	アロメトリー式とモデルの定義	6
2.2	データの読み込みと図示	7
2.3	回帰パラメータと標準偏差 $\log \sigma$ に対する (負の) 対数尤度関数の定義	7
2.4	(負の) 対数尤度関数の最適化	8
2.5	R の関数 lm を使った応え合わせ	8
3	独立ではあるが時間的構造をもった観測値の基づくパラメータの推定—プロダクションモデル—	9
3.1	プロダクションモデル	9
3.2	統計モデルの定義	9
3.3	データの読み込み	10
3.4	データの図示	10
3.5	個体群動態の定義	11
3.6	尤度関数の定義 ($z=1$; Schaefer model)	12

3.7	パラメータの推定	12
3.8	結果の図示	13
3.9	KOBE plot	14
4	提出課題	15

```
set.seed(2019)
```

0.1 目的

- 1) 独立かつ同一な確率分布に従う観測値に基づく正規分布の平均と標準偏差の推定
- 2) 独立ではあるか同一ではない確率分布に従う観測値に基づく回帰係数と標準偏差の推定—線形回帰—
- 3) 独立ではあるが時間的構造をもった観測値のに基づくパラメータの推定—プロダクションモデル—

0.2 用語

- ニュートン法による最適化
- 正規分布
- 線形回帰
- 非線形回帰
- 成長式
- プロダクションモデル

0.3 R の関数（復習含む）

- optim: 最適化
- dnorm: 正規分布の密度関数
- function: 関数定義

1 独立かつ同一な確率分布に従う観測値に基づく正規分布の平均と標準偏差の推定

1.1 正規分布に関する R の関数

`dnorm`(観測値の値 (y), 平均 (mu), 標準偏差 (sigma), log=T or F)

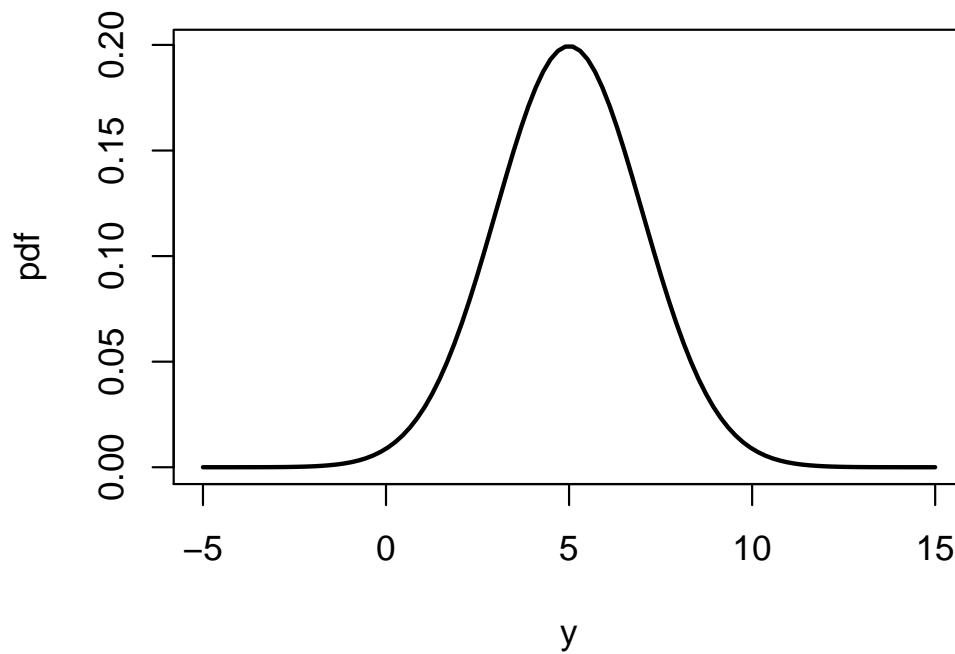
`rnorm`(乱数の数, 平均 (mu), 標準偏差 (sigma))

1.2 R による確率密度の計算

```
mu <- 5
sigma <- 2
y <- seq(-5, 15, length=100)
pdf <- dnorm(y, mu, sigma)
round(pdf, digits=3)
```

```
[1] 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
[12] 0.000 0.000 0.000 0.000 0.000 0.001 0.001 0.001 0.002 0.002 0.003
[23] 0.004 0.006 0.007 0.009 0.012 0.015 0.019 0.023 0.029 0.035 0.042
[34] 0.050 0.059 0.068 0.079 0.090 0.102 0.114 0.126 0.138 0.150 0.161
[45] 0.171 0.180 0.187 0.193 0.197 0.199 0.199 0.197 0.193 0.187 0.180
[56] 0.171 0.161 0.150 0.138 0.126 0.114 0.102 0.090 0.079 0.068 0.059
[67] 0.050 0.042 0.035 0.029 0.023 0.019 0.015 0.012 0.009 0.007 0.006
[78] 0.004 0.003 0.002 0.002 0.001 0.001 0.001 0.000 0.000 0.000 0.000
[89] 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
[100] 0.000
```

```
plot(y, pdf, type="l", lwd=2)
```



1.3 R による乱数の生成

10 個の独立同一な観測値を $N(5, 2^2)$ から生成.

```
ns <- 10
mu <- 5
sigma <- 2
yobs <- rnorm(ns, mu, sigma)
round(yobs, digits=3)
```

```
[1] 6.477 3.970 1.720 6.832 2.465 6.476 3.435 6.019 2.020 4.362
```

1.4 平均値 μ と標準偏差 $\log \sigma$ に対する (負の) 対数尤度関数の定義

```
NLL1 <- function(par){
  mu <- par[1]
  sigma <- exp(par[2])
```

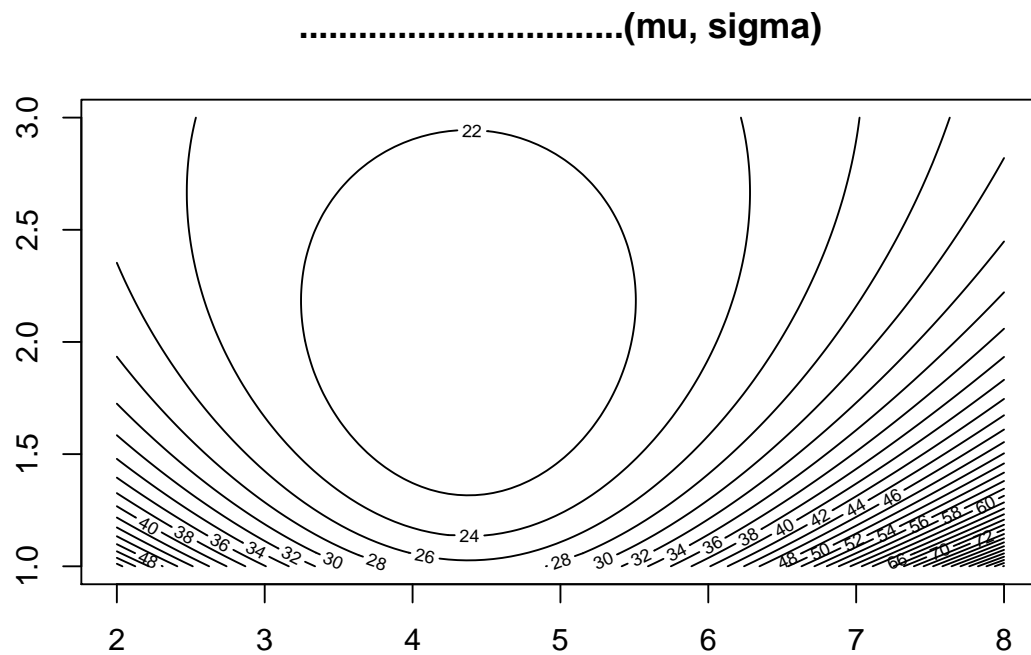
```
pdf <- dnorm(yobs, mu, sigma, log=T)
loglike <- sum(pdf)
negloglike <- -loglike
return(negloglike)
}
```

1.5 平均値 μ と標準偏差 $\log \sigma$ に対する (負の) 対数尤度関数の最適化

```
Len <- 100
Out <- array(0, c(Len,Len))
mu.vec <- seq(mu-3, mu+3, length.out=Len)
sigma.vec <- seq(1, 3, length.out=Len)

for(i in 1:Len){
  for(j in 1:Len){
    par <- c(mu.vec[i], log(sigma.vec[j]))
    Out[i,j] <- NLL1(par)
  }
}

contour(mu.vec, sigma.vec, Out, nlevels=50, main=" 負の対数尤度の等高線図 (mu, sigma)")
```



```
Res1 <- optim(c(0,0), NLL1, method="BFGS")
```

```
Res1$par
```

```
[1] 4.3775945 0.6244561
```

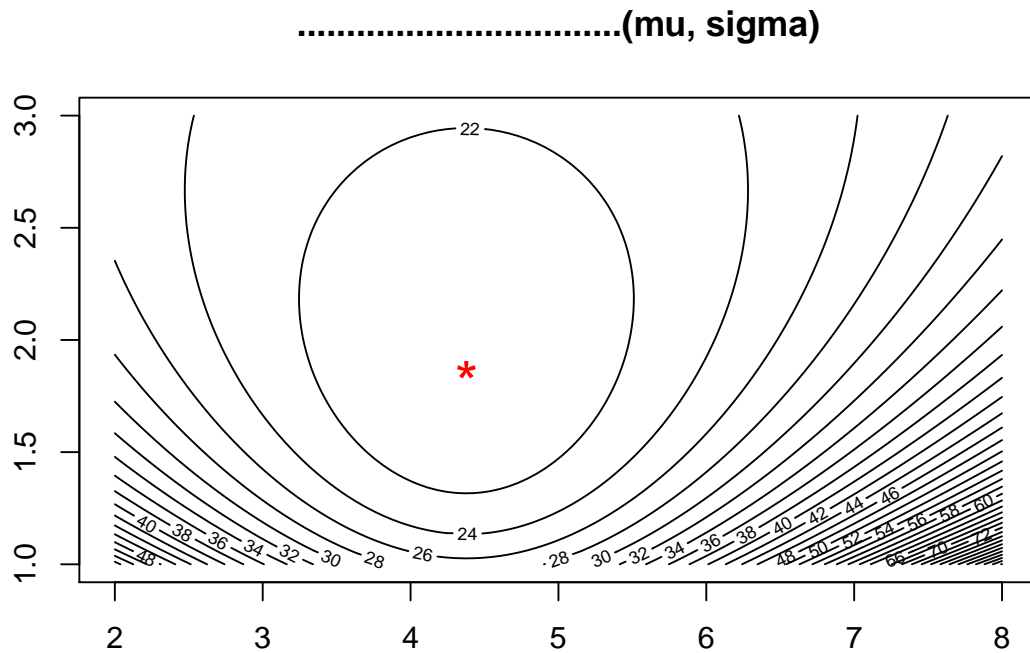
```
data.frame(mu.est=Res1$par[1], sigma.est=exp(Res1$par[2]))
```

```
      mu.est sigma.est
1 4.377594  1.86723
```

```
contour(mu.vec, sigma.vec, Out, nlevels=50,
```

```
      main=" 負の対数尤度の等高線図 (mu, sigma)")
```

```
points(Res1$par[1], exp(Res1$par[2]), pch="*", col="red", cex=2)
```



2 独立ではあるか同一ではない確率分布に従う観測値に基づく回帰係数と標準偏差の推定—線形回帰—

2.1 アロメトリー式とモデルの定義

下記のアロメトリー式の推定を考える.

$$W = aL^b$$

両辺対数をとると線形化できる.

$$\log W = \log a + b \log L$$

この関係式に従うとされる n 組のデータ (L_i, W_i) があるとし, これが以下の正規分布に従うとする.

$$\log W_i = \log a + b \log L_i + \varepsilon_i, \quad \varepsilon_i \sim N(0, \sigma^2) \quad (i = 1, \dots, n)$$

上式は以下のようにも表現できる.

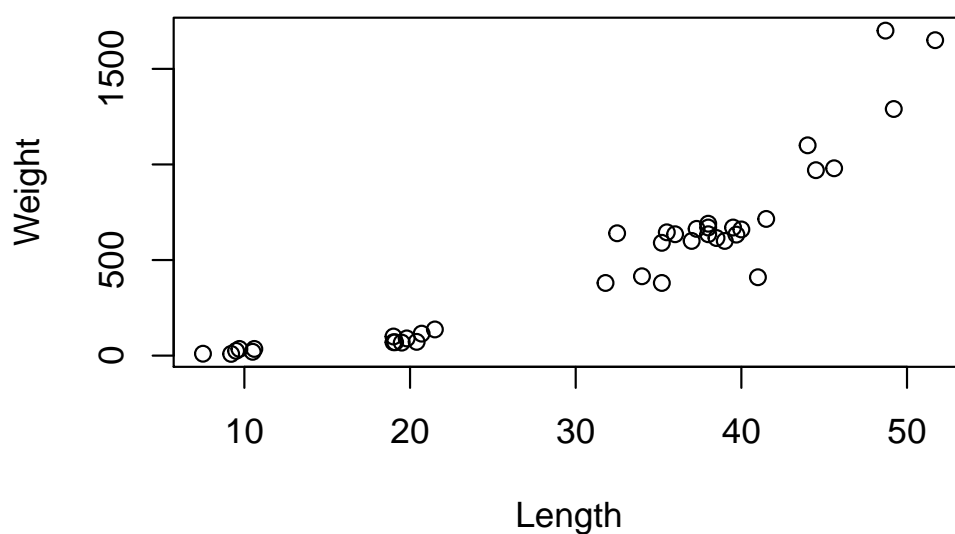
$$\log W_i \sim N(\log a + b \log L_i, \sigma^2) \quad (i = 1, \dots, n)$$

2.2 データの読み込みと図示

```
Data <- read.csv("Rainbowtrout.csv", header=T) # データの読み込み
names(Data) # データのヘッダーの確認
```

```
[1] "Length" "Weight"
```

```
Length <- Data$Length
Weight <- Data$Weight
n <- length(Length)
plot(Length, Weight)
```



2.3 回帰パラメータと標準偏差 $\log \sigma$ に対する (負の) 対数尤度関数の定義

```
NLL2 <- function(par){
  a <- exp(par[1])
  b <- exp(par[2])
  sigma <- exp(par[3])
```



```

yobs <- log(Weight)
mu <- log(a) + b*log(Length)

pdf <- dnorm(yobs, mu, sigma, log=T)
loglike <- sum(pdf)
negloglike <- -loglike
return(negloglike)
}

```

2.4 (負の) 対数尤度関数の最適化

```

Res2 <- optim(c(0,0,0), NLL2, method="BFGS")
Res2$par

```

```
[1] -3.048818  0.956397 -1.278024
```

```
data.frame(a.est=exp(Res2$par[1]), b.est=exp(Res2$par[2]), sigma.est=exp(Res2$par[3]))
```

```

      a.est      b.est sigma.est
1 0.04741493 2.602303 0.2785872

```

2.5 R の関数 lm を使った応え合わせ

```

res <- lm(log(Weight)~log(Length)) # 回帰分析の実行
summary(res)

```

Call:

```
lm(formula = log(Weight) ~ log(Length))
```

Residuals:

```

      Min       1Q   Median       3Q      Max
-0.59889 -0.08977 -0.00832  0.14493  0.69146

```

Coefficients:

```
Estimate Std. Error t value Pr(>|t|)
```

```
(Intercept) -3.04898    0.28442   -10.72  6.67e-13 ***
log(Length)  2.60235    0.08474    30.71  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.286 on 37 degrees of freedom
Multiple R-squared:  0.9622,    Adjusted R-squared:  0.9612
F-statistic: 943.1 on 1 and 37 DF,  p-value: < 2.2e-16
```

(注) lm の標準偏差の推定は自由度を補正しているのて最尤推定値とは少し異なる.

```
sigma.ML <- exp(Res2$par[3])
sigma.Unbiased <- sqrt(sigma.ML^2*(n)/(n-2))
data.frame(sigma.ML, sigma.Unbiased)
```

```
      sigma.ML sigma.Unbiased
1 0.2785872    0.2860175
```

3 独立ではあるが時間的構造をもった観測値の基づくパラメータの推定—プロダクションモデル—

3.1 プロダクションモデル

いま, B_t を t 年初めの資源量とし, 漁獲を始めた年初の資源量を $B_0 = K$ (環境収容力) する. t 年の漁獲量を C_t で表すとき, 翌 $t+1$ 年初めの資源量は以下のロジスティック余剰生産モデルに従うとする.

$$B_{t+1} = B_t + rB_t \left\{ 1 - \left(\frac{B_t}{K} \right)^z \right\} - C_t$$

3.2 統計モデルの定義

CPUE が資源量に比例すると仮定する.

$$CPUE = qB$$

ここで, q は漁具能率という比例係数である. いま, CPUE の時系列データに対して次のような統計モデルを考える.

$$\log CPUE_t = \log q + \log B_t + \varepsilon_t, \quad \varepsilon_t \sim N(0, \sigma^2)$$

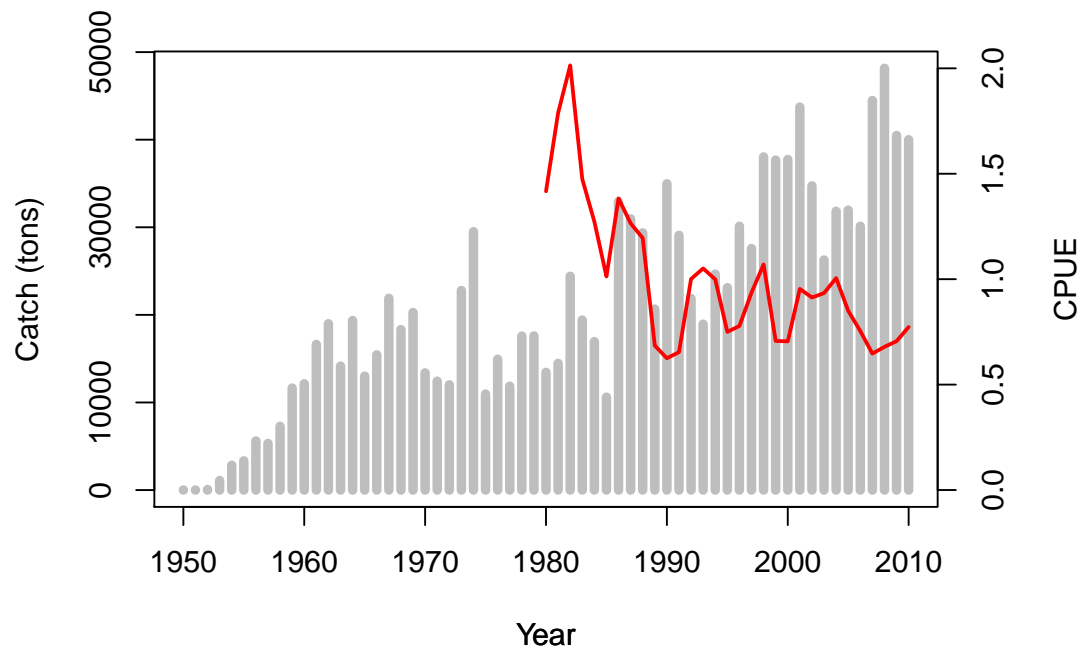
3.3 データの読み込み

```
# Data for Indian Ocean albacore
Data <- read.csv("Albacore.csv", header=T)
Year <- Data$Year
Catch <- Data$Catch
CPUE <- Data$CPUE
CV <- Data$CV

TT <- length(Year)
start.year <- min(Year)
final.year <- max(Year)
```

3.4 データの図示

```
par(mai = c(1, 1.1, 1, 1.1))
plot(Year, Catch, type = "h", lwd=5, col="gray", ylab="Catch (tons)")
par(new=T)
plot(Year, CPUE, type="l", lwd=2, col="red", axes=FALSE, ylab="", ylim=c(0,2))
axis(4); mtext("CPUE", side=4, line=3)
```



3.5 個体群動態の定義

```
PDM.PT<-function(r, K, z, B1, TT, Catch)
{
  TT <- TT
  B<-rep(NA,TT)
  B[1]<-B1
  for(t in 1:(TT-1))
  {
    tmp<-B[t]+r*B[t]*(1-(B[t]/K)^z)-Catch[t]
    B[t+1]<-max(tmp,0.001)
  }
  B
}
```

3.6 尤度関数の定義 (z=1; Schaefer model)

```
NLL.PT <- function(par,z)
{
  r<-exp(par[1])
  K<-exp(par[2])
  q<-exp(par[3])
  sigma <-exp(par[4])
  B<-PDM.PT(r, K, z, K, TT, Catch)

  yobs <- log(CPUE)
  mu <- log(q) + log(B)

  pdf <- dnorm(yobs, mu, sigma, log=T)
  loglike <- sum(pdf, na.rm=T)
  negloglike <- -loglike
  return(negloglike)
}
```

3.7 パラメータの推定

```
init.MSY <- 2*mean(Catch)
init.r <- 0.4
init.K <- 4*init.MSY/init.r

init<-log(c(init.r, init.K, 10^(-6), 1))
NLL.PT(init, z=1)
```

```
[1] 46.552
```

```
NLL.PT(2*init, z=1)
```

```
[1] 74.13025
```

```
res.SC<-optim(init, NLL.PT, z=1, method="BFGS", control=list(reltol=10^(-8)))
res <- res.SC; res
```

```
$par
[1] -0.9788076 12.7297969 -12.4127161 -1.4274852
```

```
$value
[1] -0.2648306
```

```
$counts
function gradient
      172      51
```

```
$convergence
[1] 0
```

```
$message
NULL
```

```
r.est<-exp(res$par[1])
K.est<-exp(res$par[2])
q.est<-exp(res$par[3])
sigma.est<-exp(res$par[4])
```

```
MSY.est <- r.est*K.est/4
MSYL.est <- K.est/2
```

```
data.frame(r.est, K.est, q.est, sigma.est)
```

```
      r.est      K.est      q.est sigma.est
1 0.3757589 337660.7 4.066548e-06 0.2399115
```

```
Predicted <- PDM.PT(r.est, K.est, z=1, K.est, TT, Catch)
```

3.8 結果の図示

```
## Graphical output ##
par(mfrow=c(2,2))
plot(log(CPUE), log(q.est*Predicted), xlim=c(-1,1), ylim=c(-1,1)); abline(0,1)
```

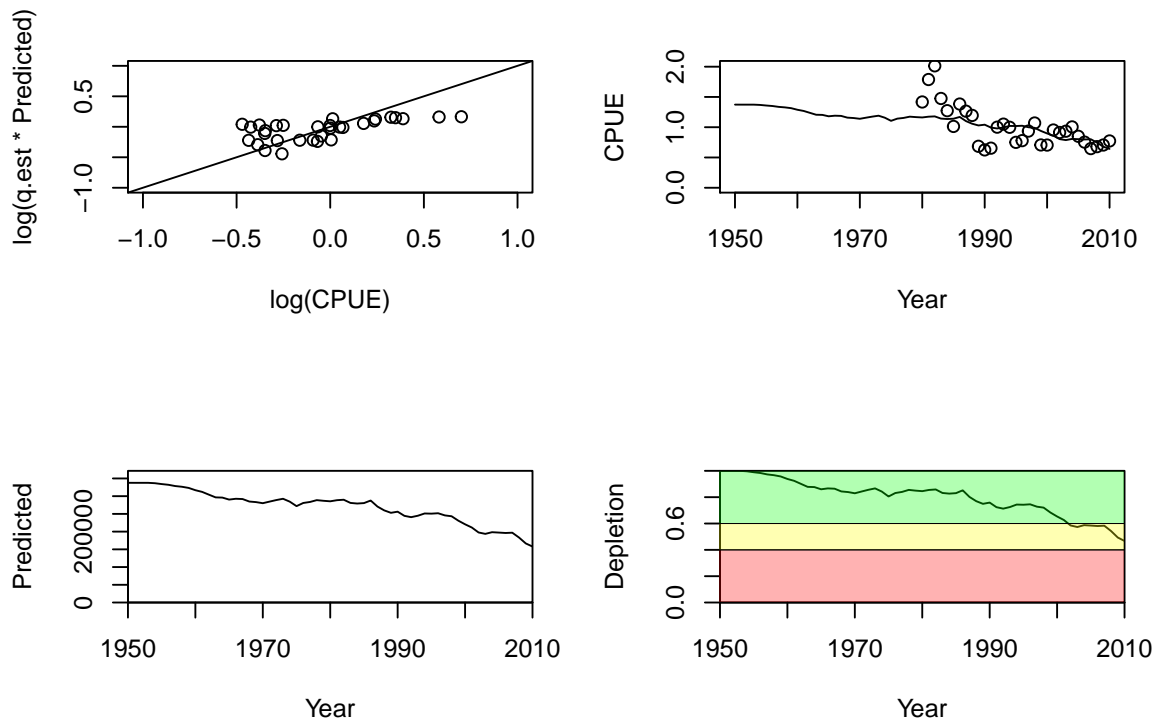
```

plot(Year, CPUE, ylim=c(0, max(CPUE, na.rm=T)))
points(Year, q.est*Predicted, type="l")

plot(Year, Predicted, type="l", ylim=c(0, 1.1*K.est), xaxs="i", yaxs="i")

plot(Year, Predicted/K.est, type="l", ylim=c(0,1), xaxs="i", yaxs="i", ylab="Depletion")
rect(xleft=-1, ybottom=0, xright=2010, ytop=0.4, lwd=0, col=rgb(1, 0, 0, alpha=0.3) )
rect(xleft=-1, ybottom=0.4, xright=2010, ytop=0.6, lwd=0, col=rgb(1, 1, 0, alpha=0.3) )
rect(xleft=-1, ybottom=0.6, xright=2010, ytop=1, lwd=0, col=rgb(0, 1, 0, alpha=0.3) )

```



3.9 KOBE plot

```

# Definition of B-ratio and F-ratio
Bratio <- Predicted/MSYL.est
Fratio <- Catch/(MSY.est) #actually Hratio

plot(Bratio,Fratio,type="n",xlim=c(0,2),ylim=c(0,2),

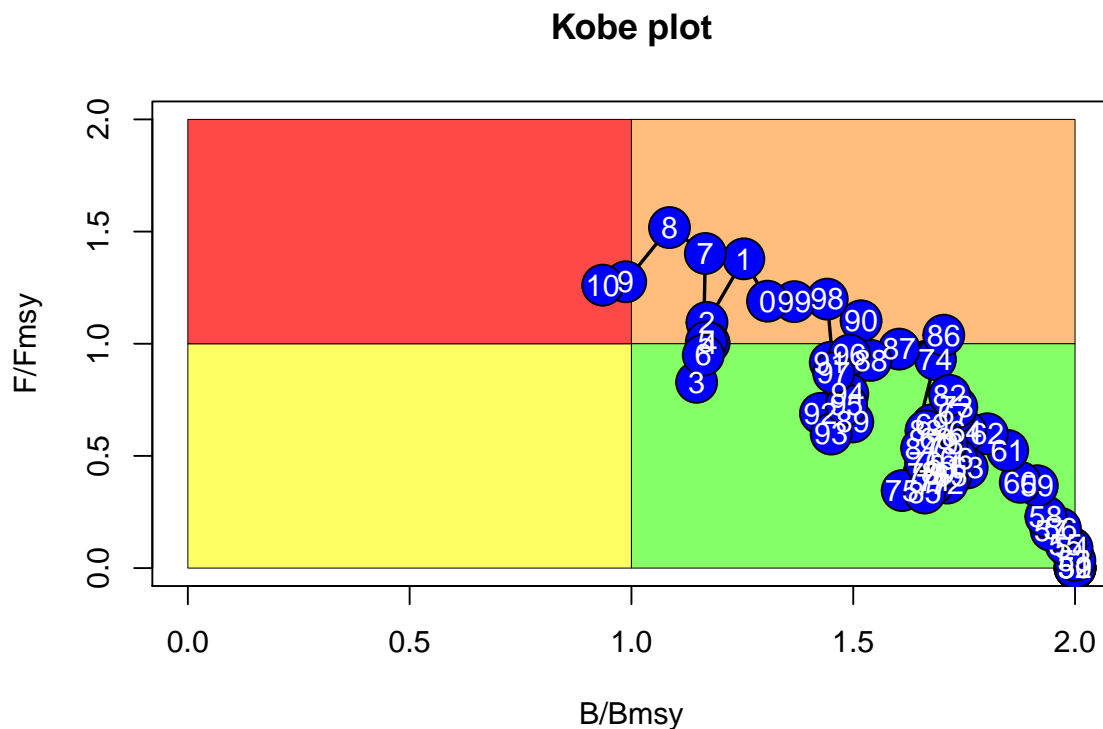
```

```

    xlab="B/Bmsy",ylab="F/Fmsy", main="Kobe plot")
rect(xleft=0,ybottom=0,xright=1,ytop=1,lwd=0,col="#FEFF63FF")
rect(xleft=0,ybottom=1,xright=1,ytop=2,lwd=0,col="#FF4949FF")
rect(xleft=1,ybottom=1,xright=2,ytop=2,lwd=0,col="#FFBE7DFF")
rect(xleft=1,ybottom=0,xright=2,ytop=1,lwd=0,col="#85FF68FF")

points(Bratio,Fratio,type="l",lwd=1.8)
points(Bratio[1:(T-1)],Fratio[1:(T-1)],lwd=1,cex=3,pch=21,bg="skyblue")
points(Bratio[T],Fratio[T],lwd=1,cex=3,pch=21,bg="blue")
num <- c(seq((start.year-1900),99,1),seq(0,(final.year-2000),1))
text(Bratio,Fratio,labels=num,adj=0.5,cex=1, col="white")

```



4 提出課題

3) のプロダクションモデルの結果をワードに貼り付けて提出。