

# 2018年度資源管理研修 (初級)

プログラム・概要

<https://github.com/ichimomo/Shigen-kensyu-2018>

**13:30～15:10**

# 水産資源データの解析

中央水産研究所資源研究センター  
資源管理グループ 市野川 桃子

# 概要

## 1. 資源管理・持続的利用のための水産資源解析

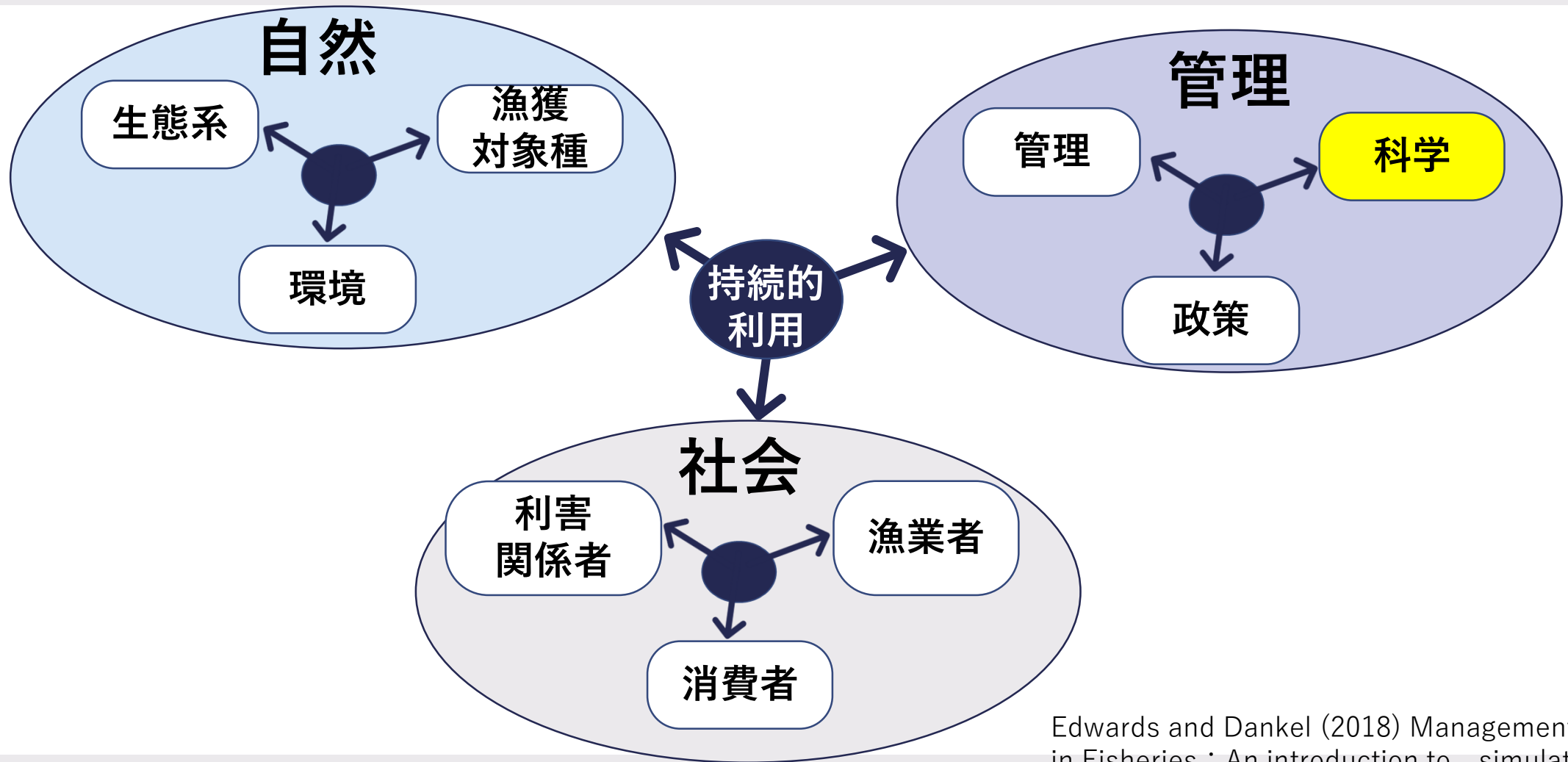
- 資源解析の目的
- 水産政策の改革について
- 水産資源解析フローチャート

## 2. Rと水産資源解析

- Rについて
- いくつかの実例

# 1. 資源管理・持続的利用のための 水産資源解析

# 資源解析の目的：究極目標は「持続的利用」



# 資源研究・解析者の役割：

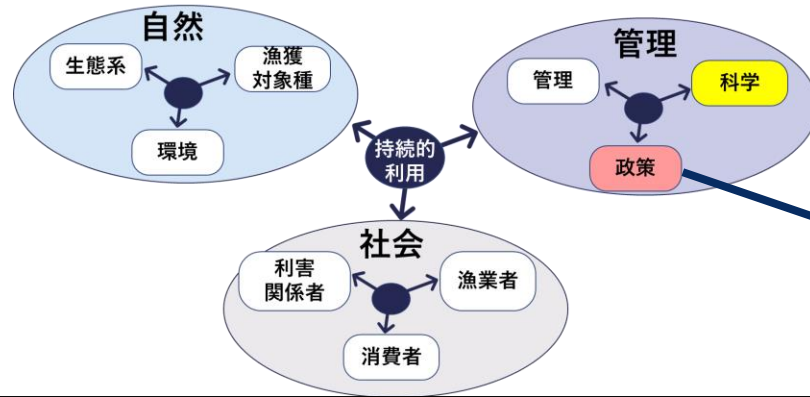
データや経験にもとづいた適切な科学的アドバイス

たとえば...

- 成熟年齢を考慮すると何歳くらいから獲りはじめたほうがよいか？（網目の大きさ）
- 魚の季節回遊や産卵時期を考えると、漁期はいつくらいがよいか？
- 保護区を設定する場合、どこに置くのが効率的か？
- 資源は減っている？増えている？
- 減っているなら努力量を削減したほうがよいか？
- 毎年のTACをどのくらいにしたらよいか？

# 水産政策の改革の中での役割

<http://www.jfa.maff.go.jp/j/kikaku/kaikaku/attach/pdf/suisankaikaku-11.pdf> より



国際的にみて遜色のない科学的・効果的な評価方法・管理

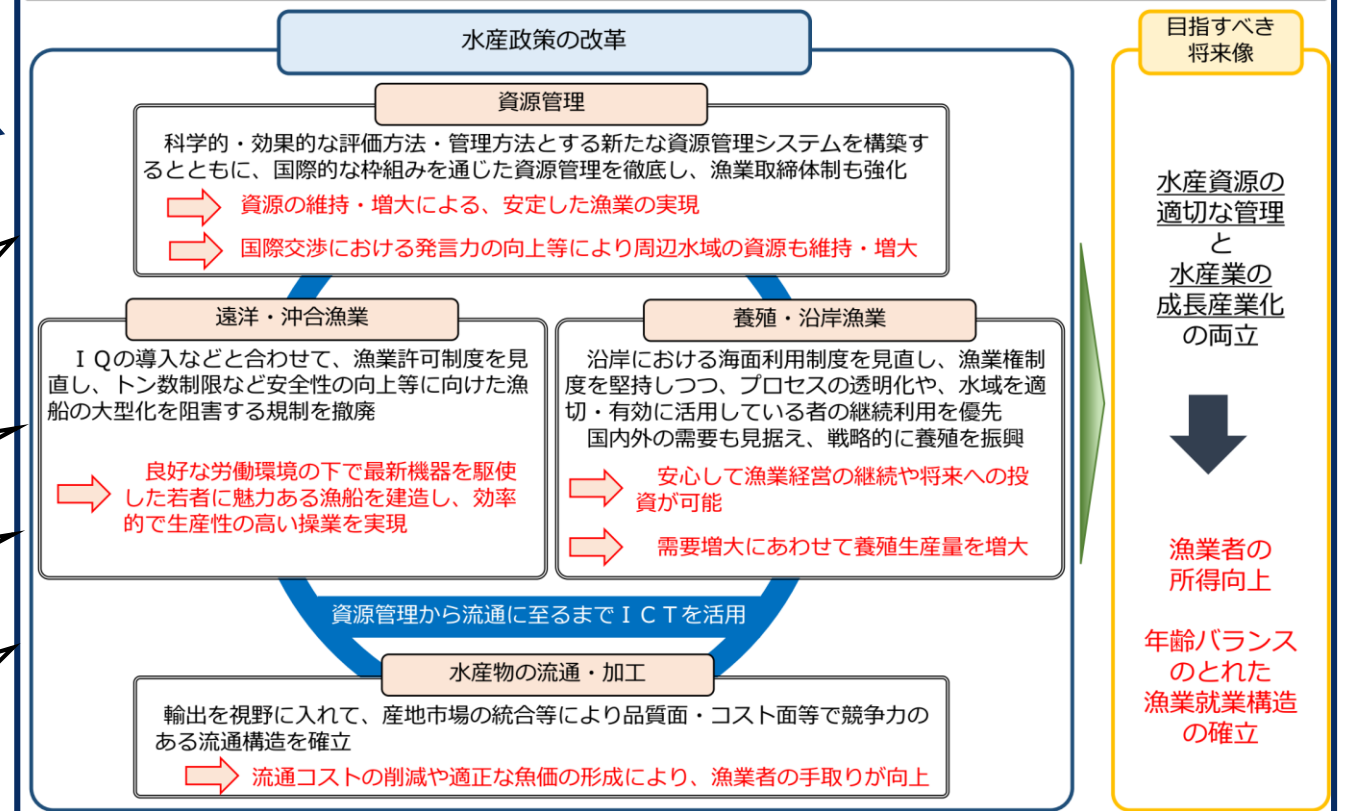
目標管理基準の設定

ビッグデータの活用

資源評価対象魚種は有用資源全体をカバー

## 水産政策の改革の全体像

- 水産資源の適切な管理と水産業の成長産業化を両立させ、漁業者の所得向上と年齢バランスの取れた漁業就業構造を確立することを目指し、水産政策の改革を実施。



# 国際的にみて遜色のない科学的・効果的な評価方法・管理

## 目標管理基準値の設定

- 評価方法：リスク・不確実性の評価
  - 推定値の信頼区間の計算（**点でなく「区間」で見る**）
  - 将来予測における「確率的な」加入変動を考慮し、「リスク」を推定（**不確実性が高い場合には「リスク」をできるだけ回避する**）
- 国際基準に沿った管理目標・管理ルールの設定
  - 国連海洋法条約・持続可能な開発目標（SDGs）→ MSY

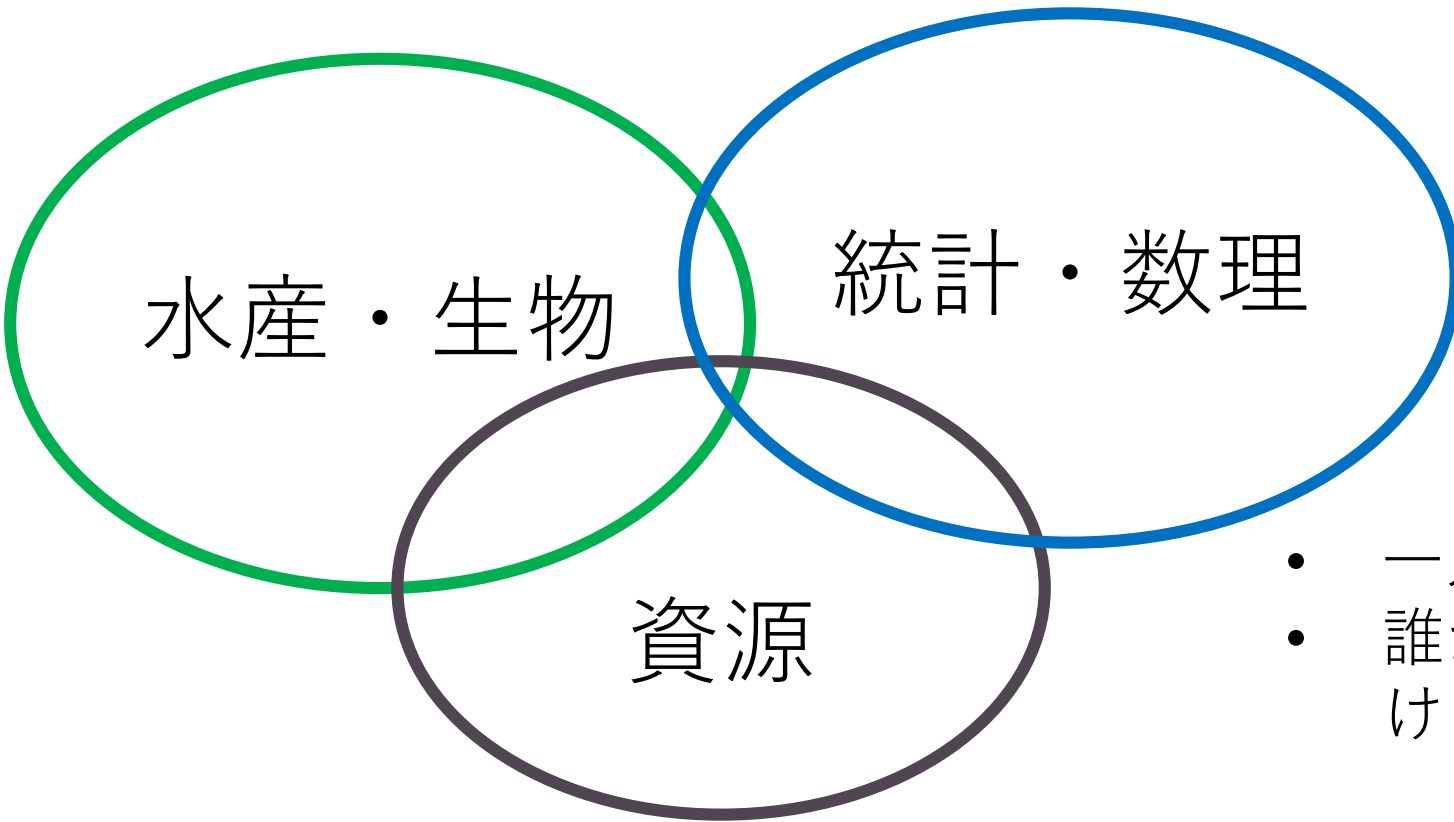


# ビッグデータの活用

資源評価対象魚種は有用資源全体をカバー

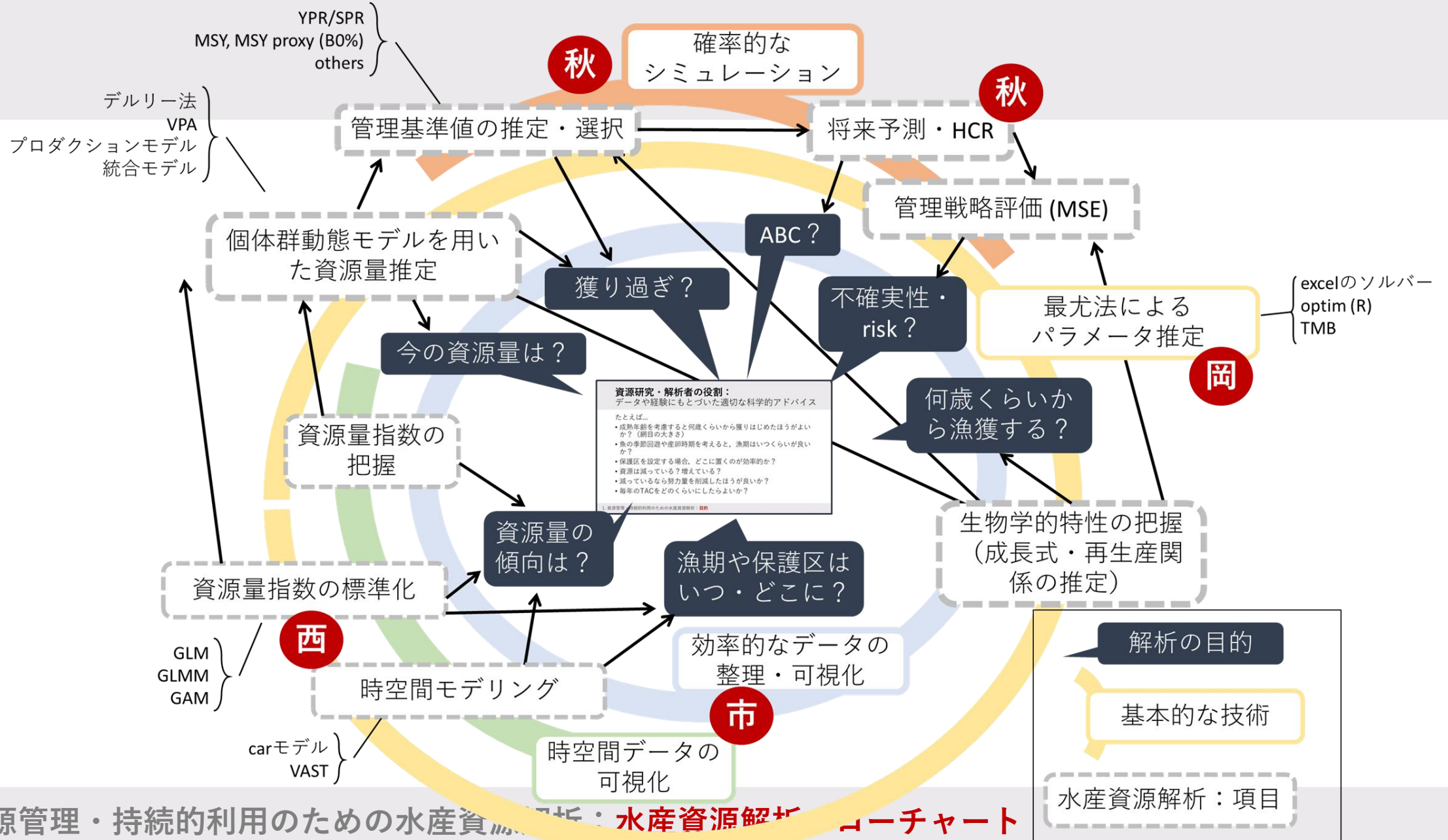
- 巨大なデータの整理・利活用の知識
  - データベースの利用、専門ソフトウェアによる統計解析
  - 機械学習など新たな解析手法の導入
- データや知見が不十分な条件下での資源評価

# 資源研究・解析者の役割は ますます大きく、複雑・高度に（複合科学）



- 一人が全部を知る必要はない
- 誰がどの分野をやっていて、誰に聞けばいいのか、把握するのが大事

# 水産資源解析フローチャート



# データの整理・可視化

市野川

- Rを使うと便利にできる
- 大規模データの整理
- for ループを使った複数枚のグラフの自動描画
- ビッグデータの活用

個体群動態モデルを用いた資源量推定

資源量指数の把握

資源量指数の標準化

時空間モデリング

生物学的特性の把握  
(成長式・再生産関係の推定)

効率的なデータの整理・可視化

管理基準値の推定・選択

管理戦略評価 (MSE)

個体群動態モデルを用いた資源量推定

- Rを使うと便利にできる
- 「尤度」を最小化する
- 高速パラメータ推定→TMB

資源量指数の標準化

時空間モデリング

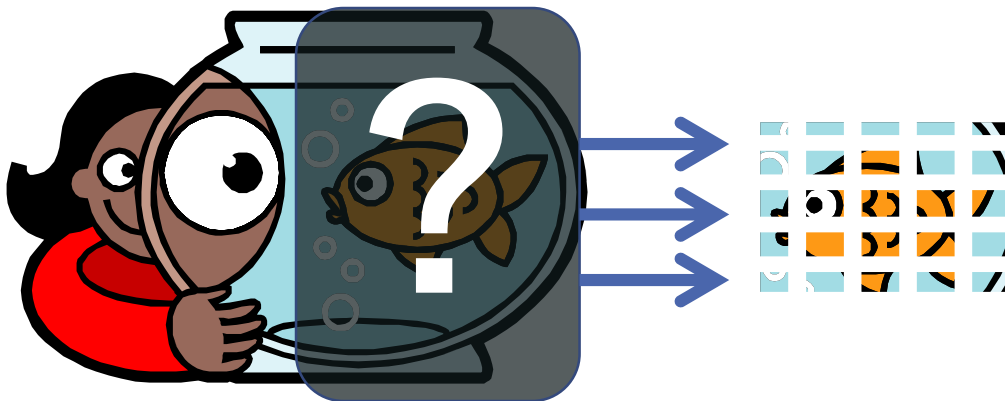
最小二乗法やAIC等を用いたパラメータ推定

## 最尤法による パラメータ推定

生物学的特性の把握  
(成長式・再生産関係の推定)

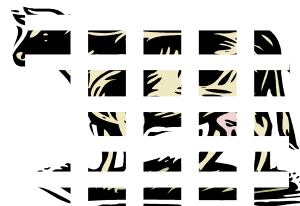
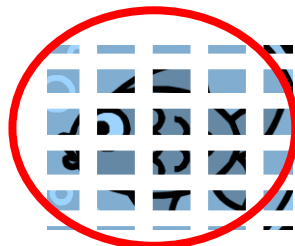
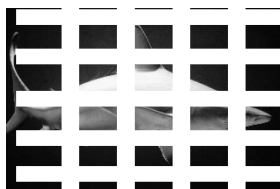
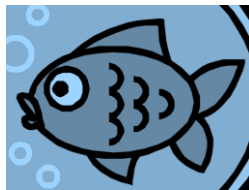
岡村

# パラメータ推定 の 概念



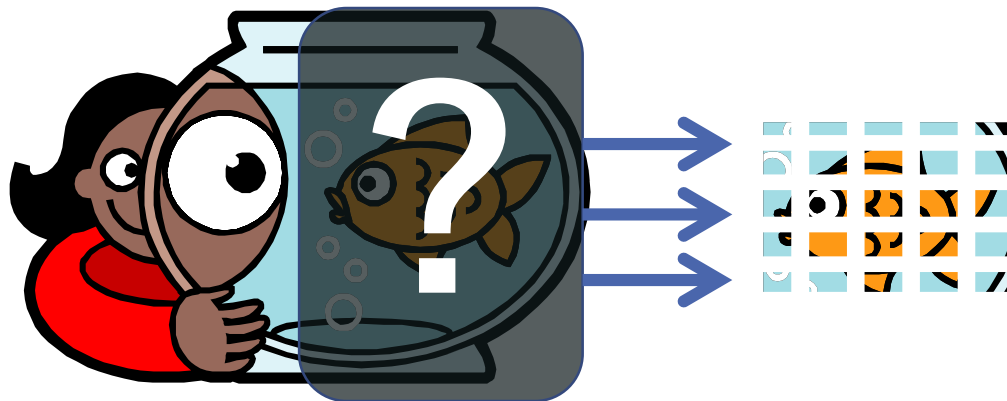
観察対象を直接観察できず、データだけが与えられる

もとの形を想像するモデルを作る



じっさいのデータとモデルとの違い（目的関数）が最小となるモデル・パラメータを選ぶ

# パラメータ推定 の 概念



単回帰・重回帰モデル  
一般化線形（混合・加法）モデル  
資源評価モデル・成長式・など

最小二乗法  
(残差の2乗和)

最尤法  
(負の尤度)

ベイズ推定

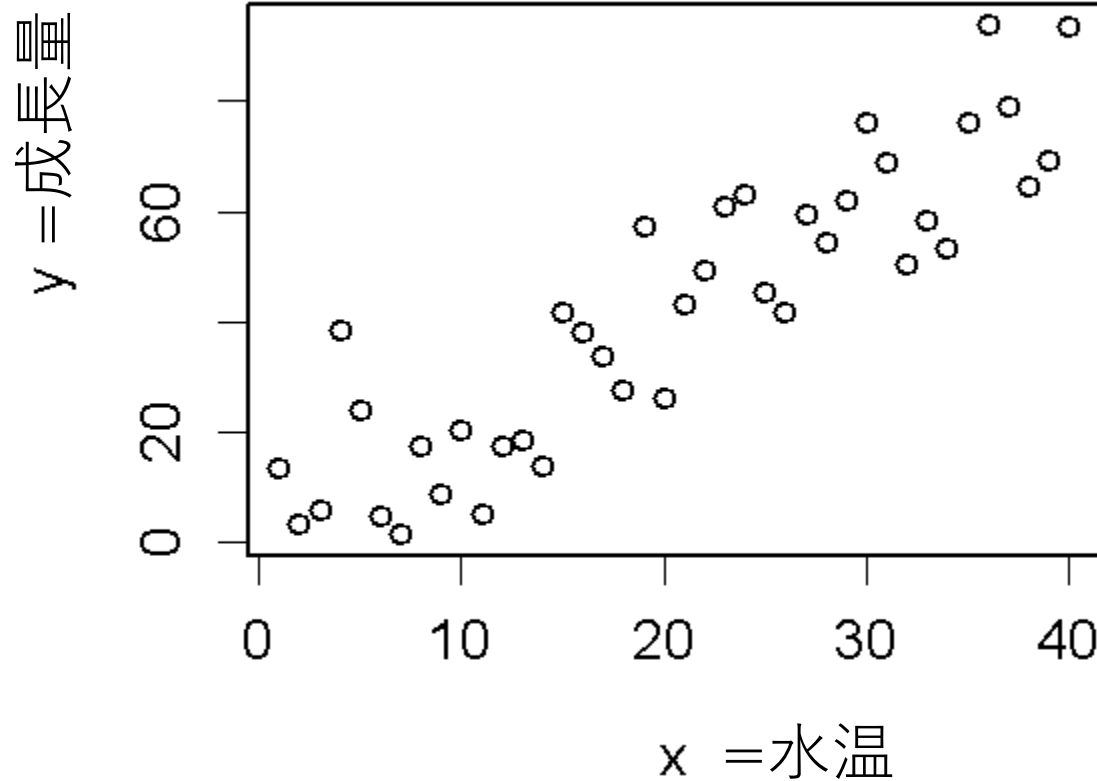
異なるモデル間の選択(AICなど)

データの取得

モデル構造  
の決定

目的関数を  
最小化して  
パラメータ推定・  
モデル選択

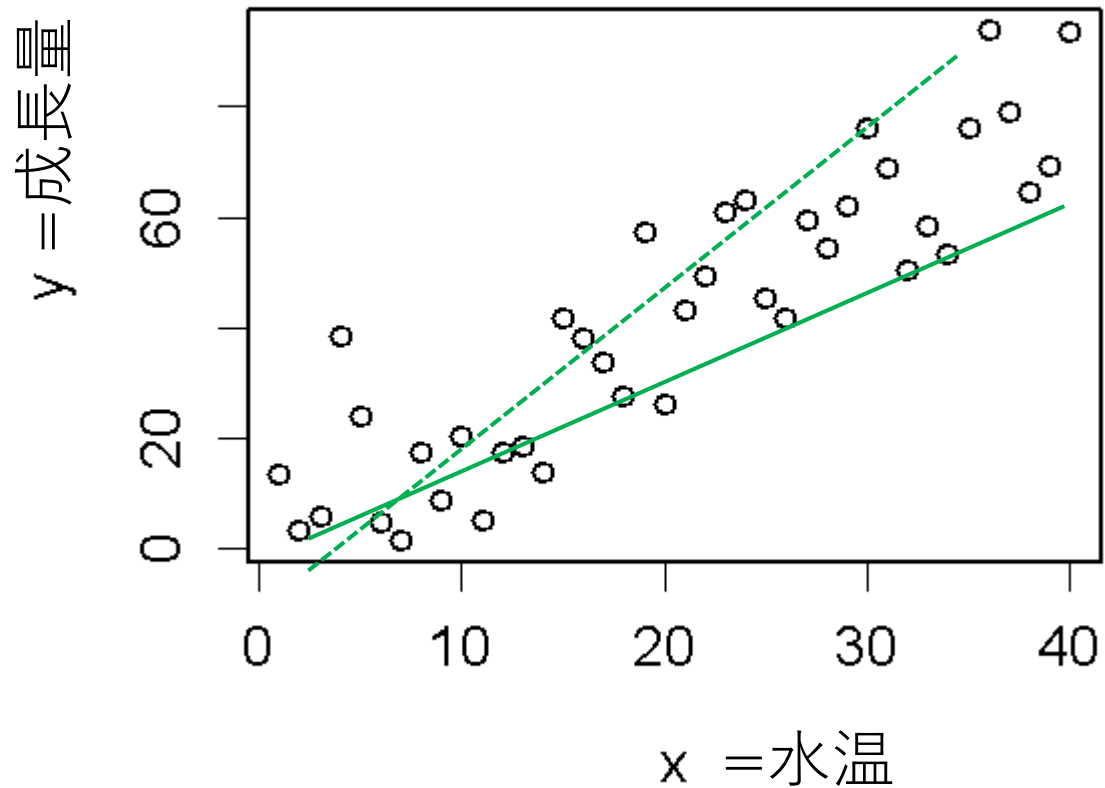
# 単回帰分析でのパラメータ推定（最小二乗法）



データ  
Y = 成長量  
X = 水温



# 単回帰分析でのパラメータ推定 (最小二乗法)

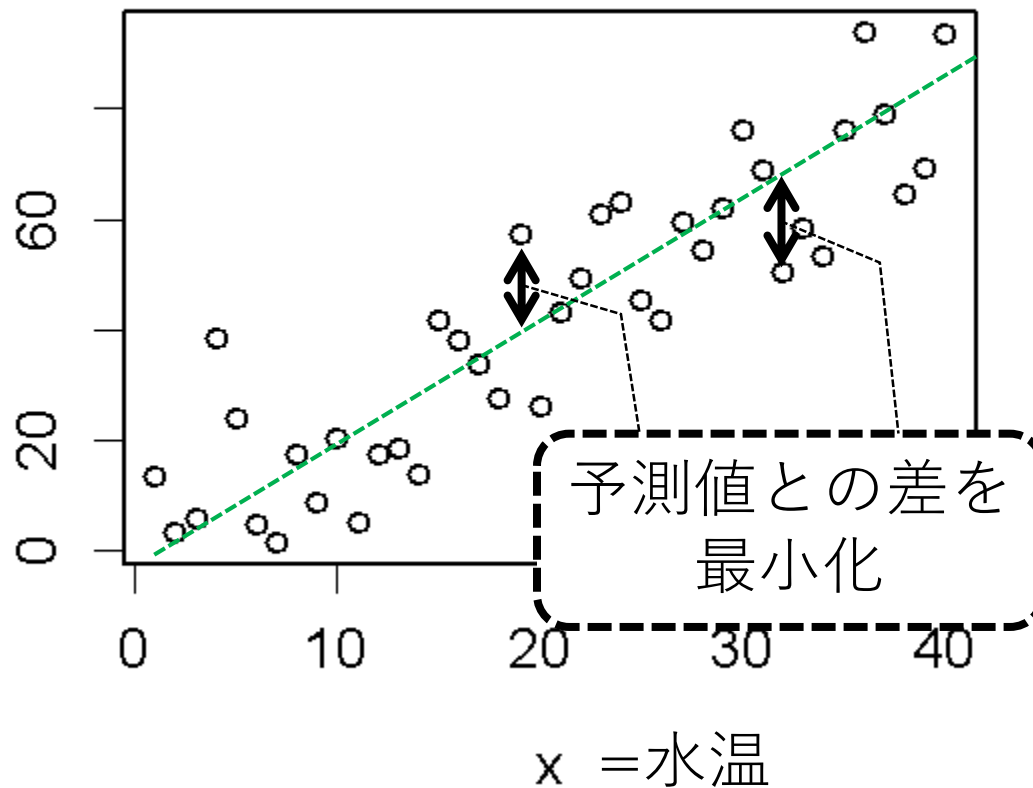


データ  
 $Y = \text{成長量}$   
 $X = \text{水温}$

モデル  
 $Y = a + bX$   
(yに対する単回帰式)

# 単回帰分析でのパラメータ推定（最小二乗法）

y = 成長量



目的関数

$$\mu_i = a + by_i$$

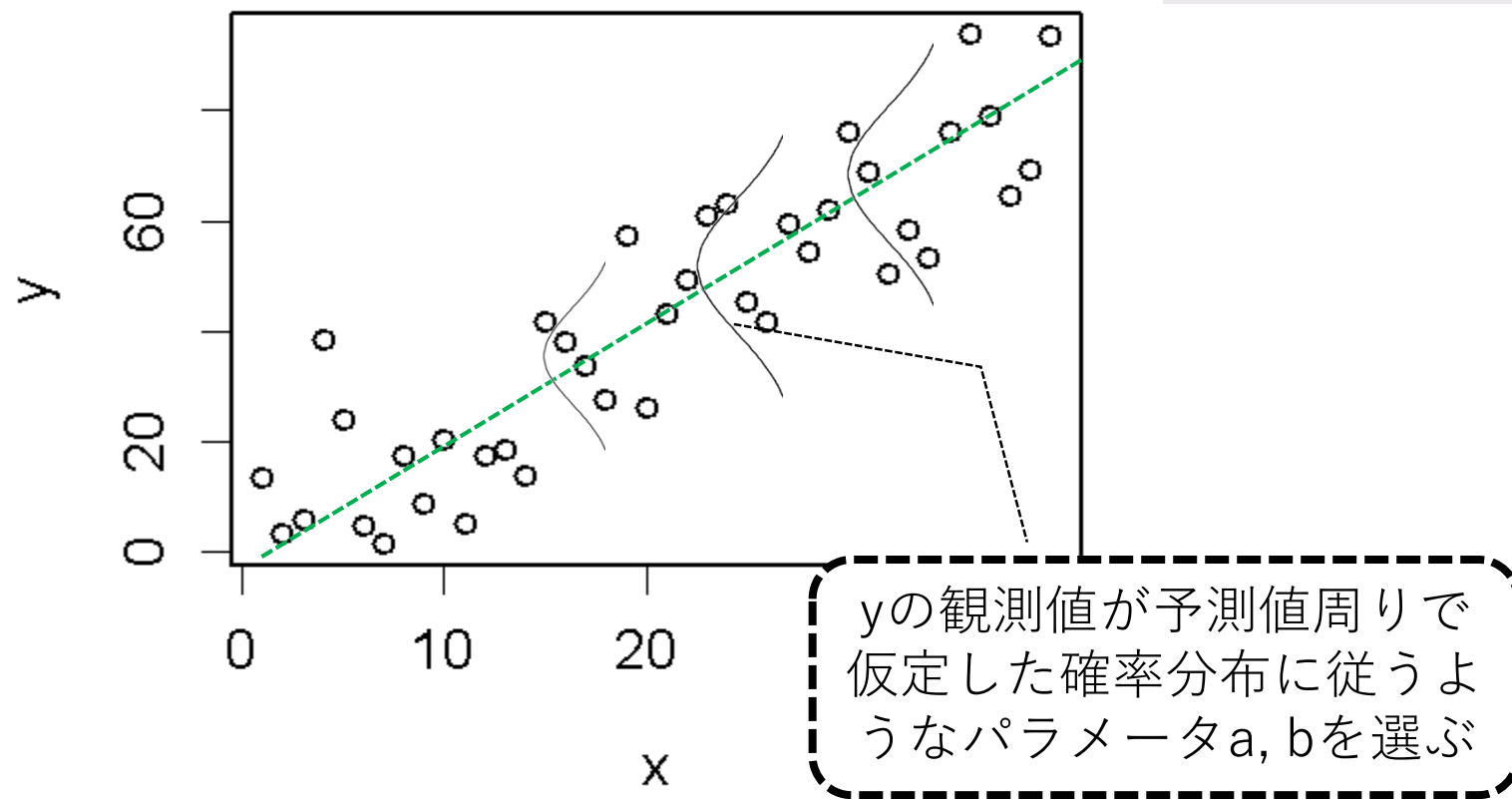
$$\min \sum_{i=1}^{i=N} (y_i - \mu_i)^2$$

データ  
Y = 成長量  
X = 水温

モデル  
 $Y = a + bX$   
(yに対する単回帰式)

目的関数 = 残差平方和  
を最小化して  
パラメータ推定

# 最尤法で同じ事をする？



$$L(\mu, \sigma^2 | y) = \prod_{i=1}^{i=N} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_i - \mu)^2}{2\sigma^2}\right)$$

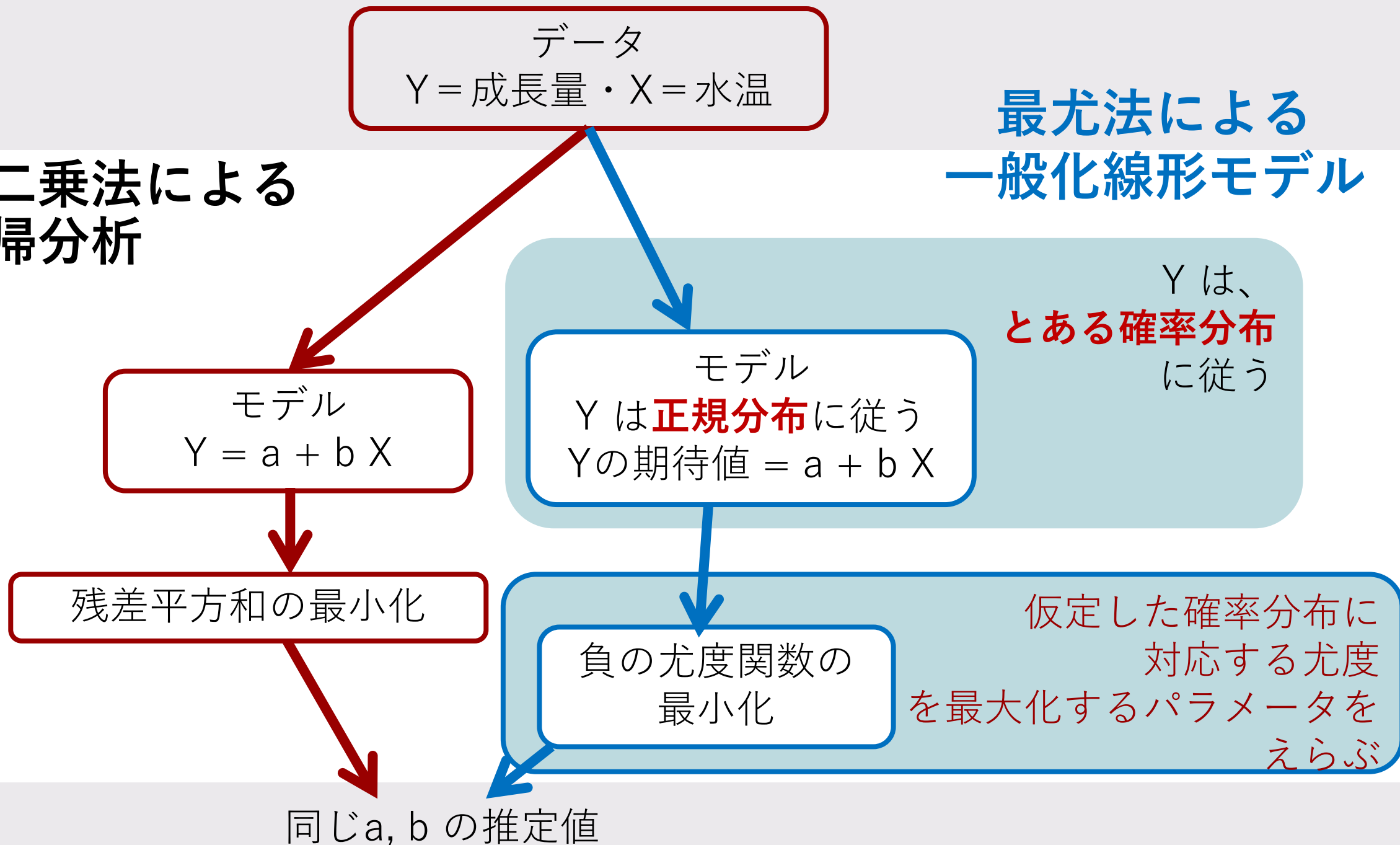
データ  
 $Y = \text{成長量}$ 、 $X = \text{水温}$

モデル  
 $Y$ は期待値が  $a + bX$   
である正規分布に従う

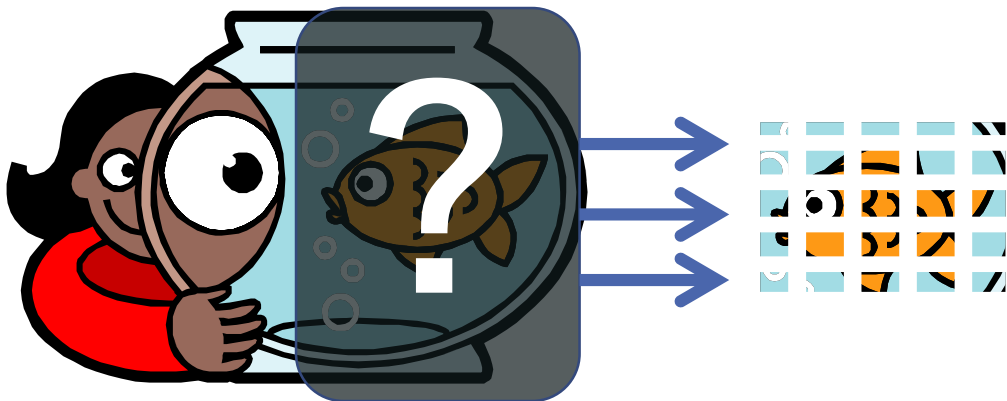
尤度（尤もらしさ）が最大  
になるように $a, b$ を決める

## 最小二乗法による 単回帰分析

## 最尤法による 一般化線形モデル

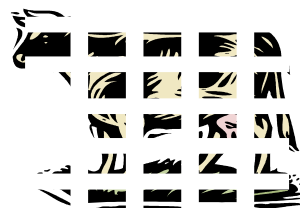
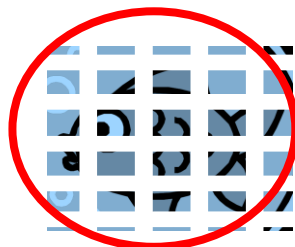
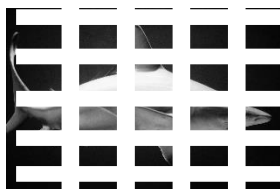
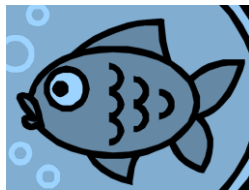


# データ & モデル & 目的関数が何か？ を意識することが大事



観察対象を直接観察できず、データだけが与えられる

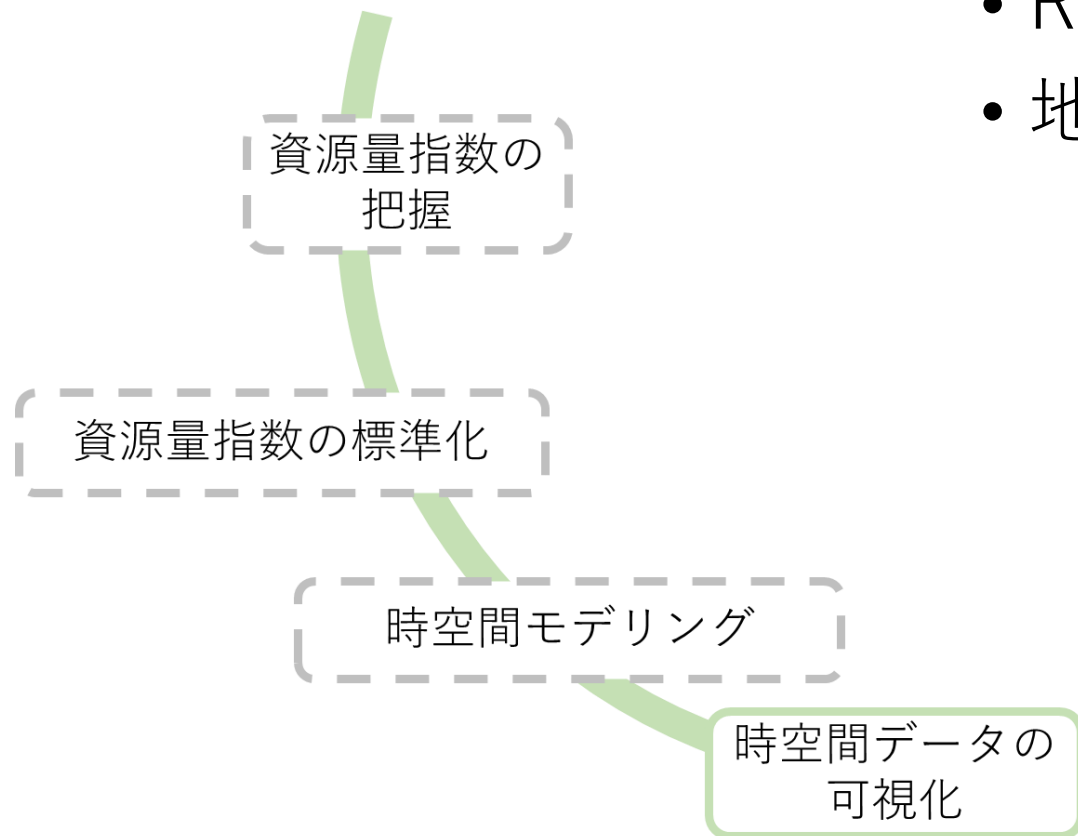
もとの形を想像するモデルを作る



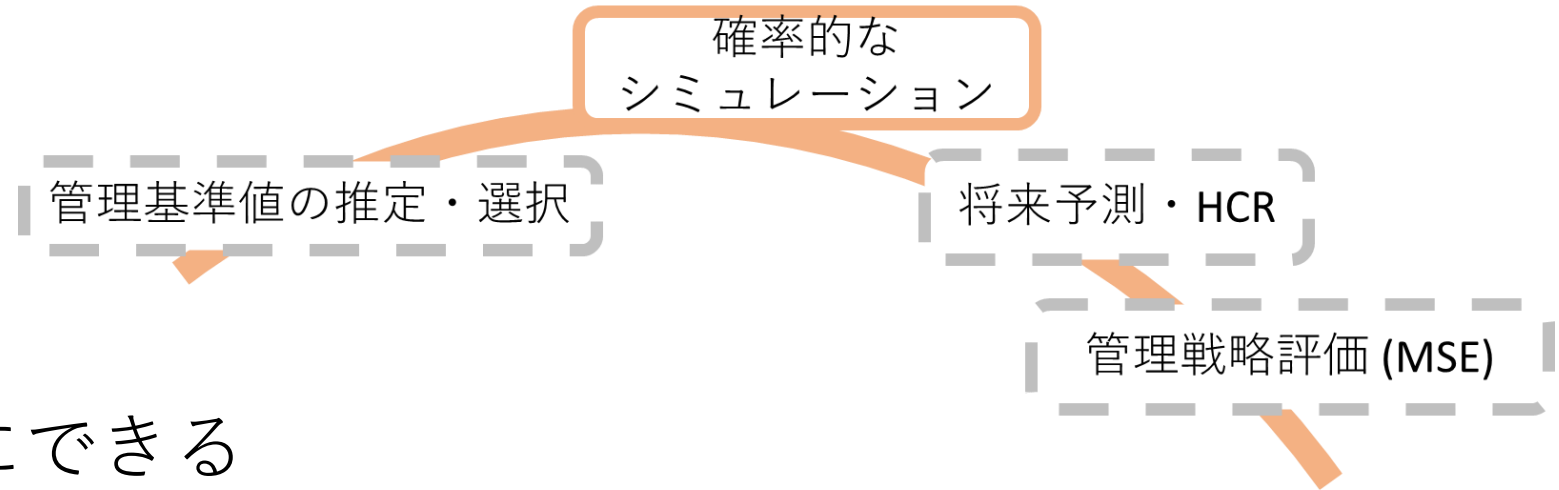
じっさいのデータとモデルとの  
違い（目的関数）が最小となる  
モデル・パラメータを選ぶ

# 時空間データの可視化

- Rを使うと便利にできる
- 地図上へのデータのマッピング  
→モデル化



# 確率的なシミュレーション



- Rを使うと便利にできる
- どう管理すればよいか？
- どのような目標を選ぶべきか？

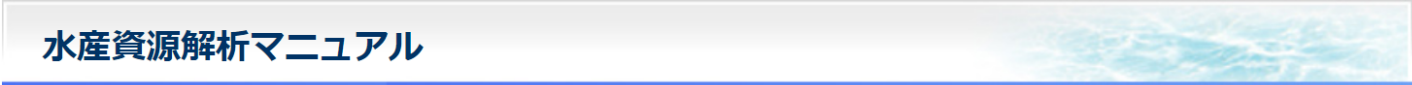
# 本研修でカバーしていない範囲

個体群動態モデルを  
使った資源量推定

- 概要 (2017年度資源管理研修)
  - <http://cse.fra.affrc.go.jp/ichimomo/fish/kensyu2017/ichinokawa2.pdf>
- デルリー法 (2017年度資源管理研修)
  - <http://cse.fra.affrc.go.jp/ichimomo/fish/kensyu2017/nishijima.pdf>
- プロダクションモデル (2013年度資源管理研修)
  - [http://cse.fra.affrc.go.jp/ichimomo/fish/ichinokawa\\_R.pdf](http://cse.fra.affrc.go.jp/ichimomo/fish/ichinokawa_R.pdf)
- VPA (2015年度資源管理研修)
  - <http://cse.fra.affrc.go.jp/ichimomo/fish/kensyu2015/kensyu2015.html>
- 統合モデル (統合モデル検討会)
  - <http://cse.fra.affrc.go.jp/ichimomo/ss-kento/ss-kentos.html>



# 本研修でカバーしていない範囲



2014.4掲載

水産資源解析を学ぶ方のために、標本調査、魚の成長、生残率と死亡率、資源量推定、加入当たり漁獲量と加入当たり産卵親魚量、再生産曲線などの概要と計算のためのエクセル・ファイルのワークシートの使用方法などを紹介しています。

はじめに

目次	エクセルファイル
1. 水産資源解析とは – 水産資源解析の入口から出口 –	
2. 生活史モデル – 資源の回遊を想定する –	
3. 標本調査 – 何尾の魚を測定すれば良いか –	(3-sampling.xls)
4. 魚の成長 – 体長組成と成長曲線 –	(4-growth.xls)
5. 生残率と死亡率 – 魚の生き死に –	(5-survival.xls)
6. 資源量推定 – 魚の量を知る –	(6-vpa.xls)
7. 加入当たり漁獲量と加入当たり産卵親魚量 – 魚を上手に利用する –	(7-ypr_spr_2.xls)
8. 再生産関係 – 親子の関係を知る –	(8-r_bh.xls)
9. 種苗放流と漁獲制限 – 放流と獲り控え –	(9-iafse.xls)
10. プロダクションモデル – 漁獲量と努力量から –	(10-sf_pm.xls)

利用者の方から7章の7.2.3 加入当たり産卵親魚量による資源管理手法および管理基準値(7-ypr\_spr\_2.xls - Sheet 7.2.3)のエクセル上の間違いを

[https://www.fra.affrc.go.jp/kseika/guide\\_and\\_manual/afr/index.html](https://www.fra.affrc.go.jp/kseika/guide_and_manual/afr/index.html)

## 2. Rと水産資源解析

# Rについて

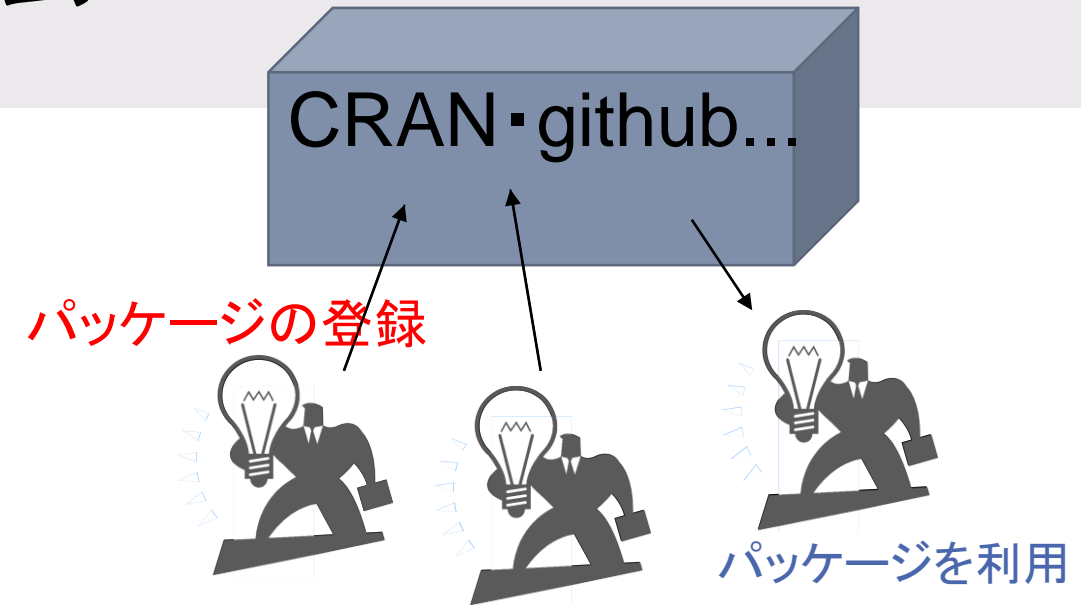
- 統計言語であるSの思想に基づいて開発されたフリーのソフトウェア
- ウェブから誰でも無料で入手できる (<http://www.r-project.org/>)
- 多様なプラットフォームに対応
  - Unix系OS、Mac OS X, Windows
- Rプログラムの作成支援：Rstudio (<https://www.rstudio.com/>)

# Rの特徴

- 大量のデータの取扱い
- 描画機能
- forループを用いた単純作業の繰り返し
- 豊富なパッケージ群を用いた高度な解析の実施
  - パッケージ：共通の目的を達成するための関数群

# Rの「パッケージ」システム

- 一定のフォーマットを満たすような自作のパッケージが多くの人によって開発され、様々な媒体を通じて配布
- パッケージは誰でも使える
- 生態学・漁業資源学の論文で用いられている統計解析手法は、多くの場合、Rによりパッケージ化されている



# 資源解析のためのパッケージ

- FLR project (a framework for fisheries modelling and management strategy simulation in R)
- TropFishR - Fish stock assessment methods and fisheries models based on the FAO Manual “Introduction to tropical fish stock assessment” by P. Sparre and S.C. Venema. Focus is the analysis of length-frequency data and data-poor fisheries.
- DLMTool - Implementation of management procedures for data-limited fisheries
- LBSPR - Functions to run the Length-Based Spawning Potential Ratio (LBSPR) method. The LBSPR package can be used in two ways: 1) simulating the expected length composition, growth curve, and SPR and yield curves using the LBSPR model and 2) fitting to empirical length data to provide an estimate of the spawning potential ratio (SPR).

# Rの実例

- 大量のデータを読み込み，簡単に集計・整形
- "for" ループを使って簡単に複数のグラフを出力

# 実例: Rを使ったデータの整理・可視化

## 事前準備

- 配布したフォルダの中で1-ichinokawa内にある.RData(またはファイル名がないRのアイコン)をクリックするとRが立ち上がります →  
フォルダ「1-ichinokawa」内でRの作業をする準備が整いました
- Rstudioをインストールしている人：Rのコードが書いてある kensyu\_ichinokawa.Rファイルを右クリックし、Rstudioで開いてもOKです



## データの読み込み

- 2-okamuraフォルダ内にある dat1.csv を読み込んでみましょう
- 関数(引数=設定, 引数=設定) という構造
- read.csv(ファイル名)

```
# '../'は一個上のフォルダ、という意味です  
cpue.data <- read.csv("../2-okamura/dat1.csv")
```

# データの中身の確認

```
# オブジェクトの先頭の数行だけを表示する関数（オブジェクト名）  
head(cpue.data)
```

	count	year	site	plant
1	1	0	0	tree
2	1	0	1	tree
3	5	0	2	tree
4	6	0	3	tree
5	3	0	4	tree
6	5	0	5	tree

```
# どのくらいの大きさの行列か確認する  
dim(cpue.data)
```

```
[1] 100  4
```

# データの表示

```
cpue.data[, 1] # 一列目のデータだけを表示する
```

```
[1] 1 1 5 6 3 5 7 6 7 9 4 4 5 6 6 9 8 10 10  
[24] 5 5 4 3 5 6 2 2 1 2 4 3 6 5 6 5 4 0 0  
[47] 0 0 0 0 0 2 3 2 3 4 5 6 6 4 0 2 4 3 6  
[70] 4 0 2 3 4 2  
[ reached getOption("max.print") -- omitted 25 entries ]
```

```
cpue.data[1, 1] # 1行・1列目のデータだけを表示する
```

```
[1] 1
```

```
cpue.data$count # '$' + 列の名前という指定のしかたもできる
```

```
[1] 1 1 5 6 3 5 7 6 7 9 4 4 5 6 6 9 8 10 10  
[24] 5 5 4 3 5 6 2 2 1 2 4 3 6 5 6 5 4 0 0  
[47] 0 0 0 0 0 2 3 2 3 4 5 6 6 4 0 2 4 3 6  
[70] 4 0 2 3 4 2  
[ reached getOption("max.print") -- omitted 25 entries ]
```

# 値の代入

- "オブジェクト名" <- "別のオブジェクト"
- 数字や文字などを入れておく「箱」のようなものをRでは「オブジェクト」と呼びます。

```
cpue1.1 <- cpue.data[1, 1] # CPUEデータの一行目のデータをcpue1.1  
cpue1.1
```

```
[1] 1
```

```
cpue.N <- cpue.data$count # CPUEデータのN(尾数)データに別の名前を  
cpue.N
```

```
[1] 1 1 5 6 3 5 7 6 7 9 4 4 5 6 6 9 8 10 10  
[24] 5 5 4 3 5 6 2 2 1 2 4 3 6 5 6 5 4 0 0  
[47] 0 0 0 0 0 2 3 2 3 4 5 6 6 4 0 2 4 3 6  
[70] 4 0 2 3 4 2  
[ reached getOption("max.print") -- omitted 25 entries ]
```

## 年で集計する

- excelのピボットテーブル的な機能に対応します
- `tapply(集計する列, 注目する列, 関数)` といったかたちで使います

```
cpue.year.mean <- tapply(cpue.data$count, cpue.data$year, mean)  
cpue.year.mean
```

0	1	2	3	4	5	6	7	8	9
5.0	7.1	3.5	3.8	0.0	3.5	4.0	3.8	5.7	0.0

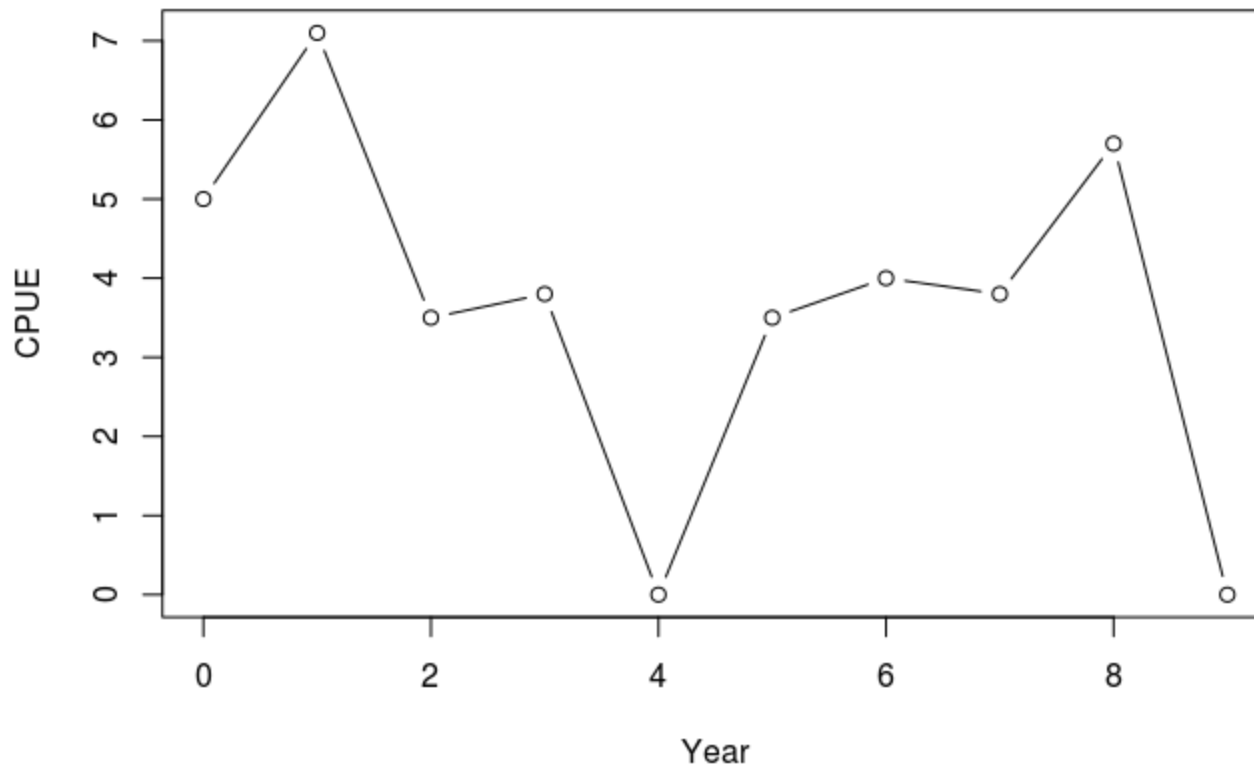
## データを抽出する

- subset(対象とするデータ, 条件)

```
cpue.site1 <- subset(cpue.data, site == 1)
cpue.site2 <- subset(cpue.data, site == 2)
cpue.year.mean.site1 <- tapply(cpue.site1$count, cpue.site1$year, mean)
cpue.year.mean.site2 <- tapply(cpue.site2$count, cpue.site2$year, mean)
```

# データをプロットする

```
# ylim=y軸の範囲, type=どんなやりかたでプロットするか ('b':線と点、  
# 線のみ、'p': 点のみ)  
plot(names(cpue.year.mean), cpue.year.mean, xlab = "Year", ylab =  
      max(cpue.year.mean)), type = "b", col = 1)
```



## 複数のグラフを並べる1

- `par` : グラフのいろいろな設定をする関数です
- `par(mfrow=c(1,5))` # 1x5列で図を並べる
- `par(ps=20)` # フォントの大きさを決める
- `par(bg="wheat")` # 背景色を決める

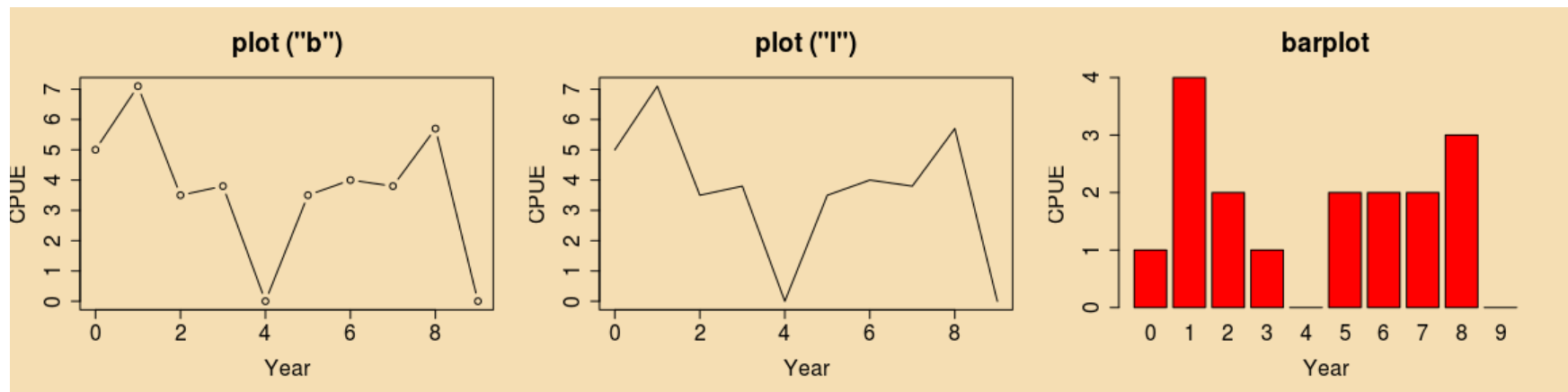


## 複数のグラフを並べる2

```
par(mfrow = c(1, 3), ps = 18, bg = "wheat")
plot(names(cpue.year.mean), cpue.year.mean, xlab = "Year", ylab =
      max(cpue.year.mean)), type = "b", col = 1)
title("plot (\"b\")")

plot(names(cpue.year.mean), cpue.year.mean, xlab = "Year", ylab =
      max(cpue.year.mean)), type = "l", col = 1)
title("plot (\"l\")")

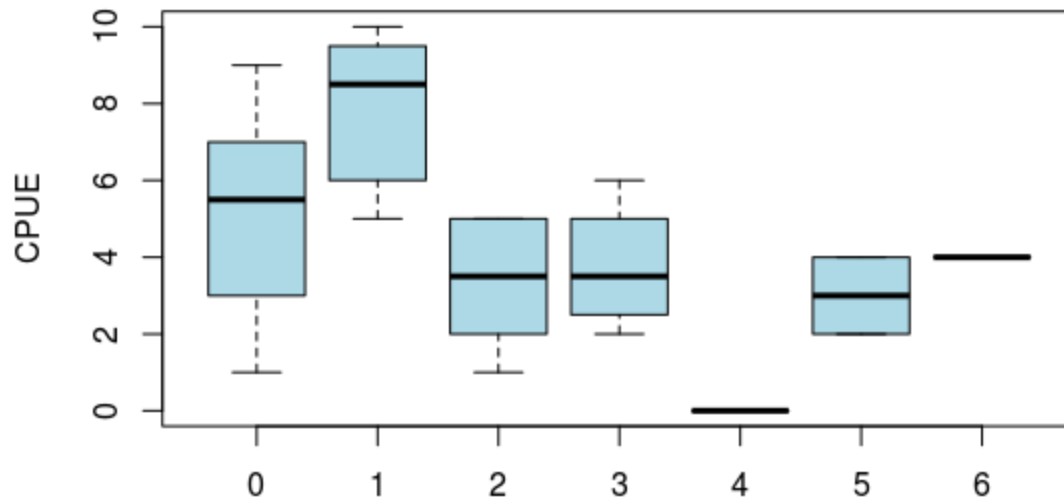
barplot(cpue.year.mean.site1, xlab = "Year", ylab = "CPUE", col =
      "red", las = 2)
title("barplot")
```



# 繰り返し処理1

- たとえばplant/shrubごとに年のトレンドを比較するには??

```
# treeのデータだけを取り出す
cpue.tree <- subset(cpue.data, plant == "tree")
# cpue.treeについて、年のトレンドを見る
boxplot(cpue.tree$count ~ cpue.tree$year, ylab = "CPUE", col =
```



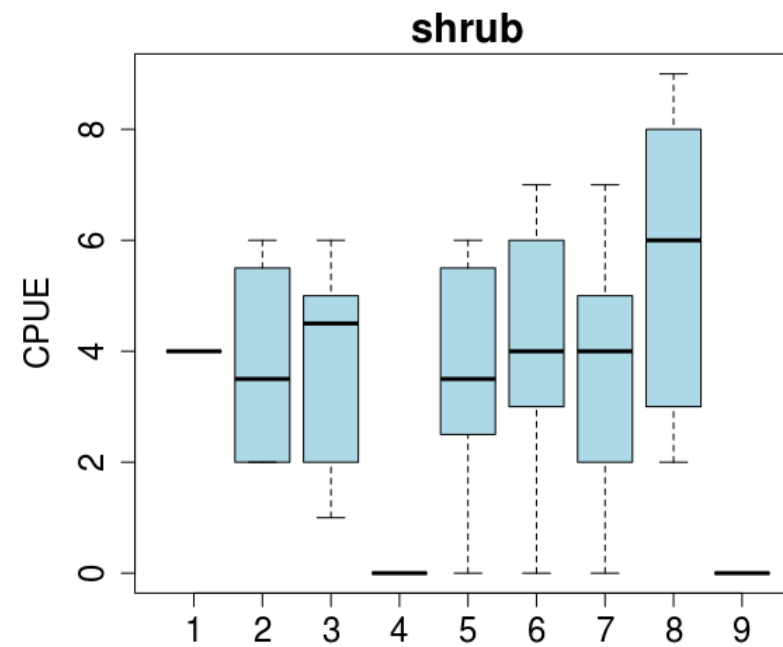
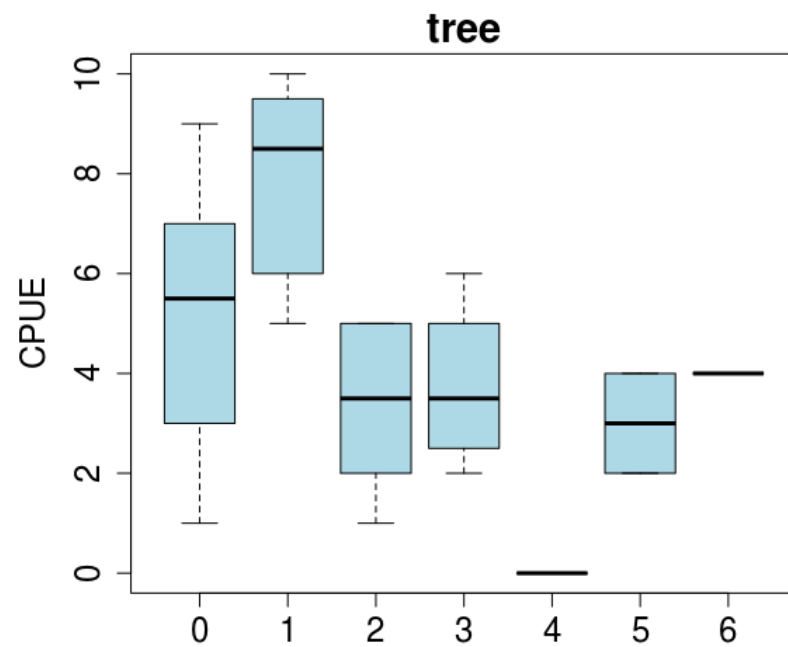
## 繰り返し処理2

- これを2回繰り返せば良い

```
par(mfrow = c(1, 2), mar = c(3, 4.3, 2, 1), ps = 18)
# treeのデータだけを取り出す
cpue.tree <- subset(cpue.data, plant == "tree")
boxplot(cpue.tree$count ~ cpue.tree$year, ylab = "CPUE", col =
title("tree"))

# treeのデータだけを取り出す
cpue.shrub <- subset(cpue.data, plant == "shrub")
boxplot(cpue.shrub$count ~ cpue.shrub$year, ylab = "CPUE", col
title("shrub"))
```

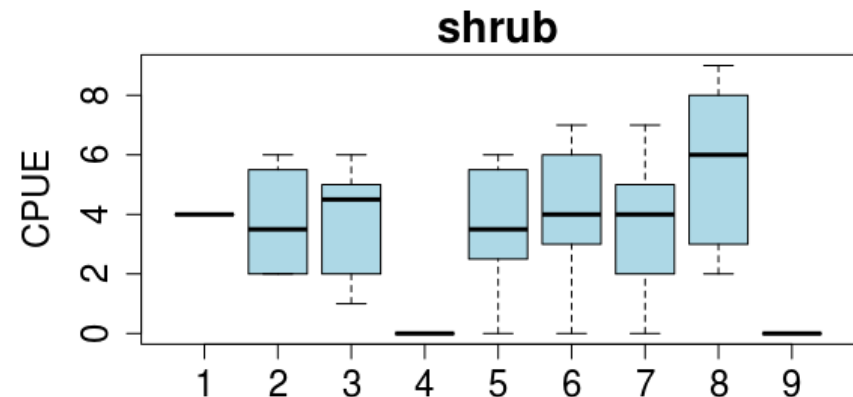
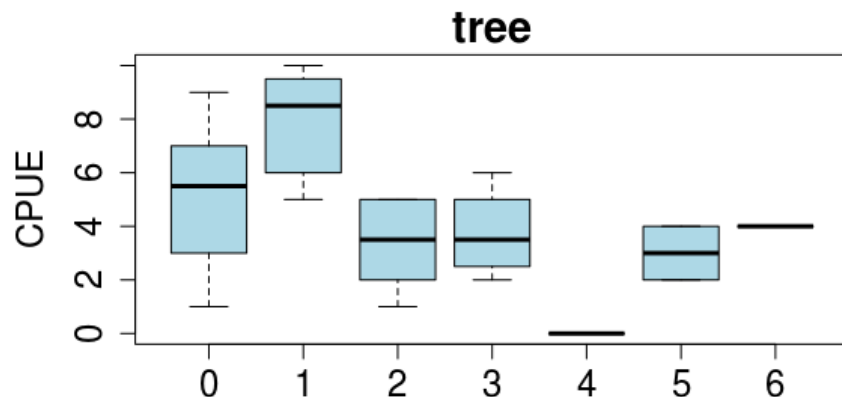
- 結果のプロット



## 繰り返し処理3

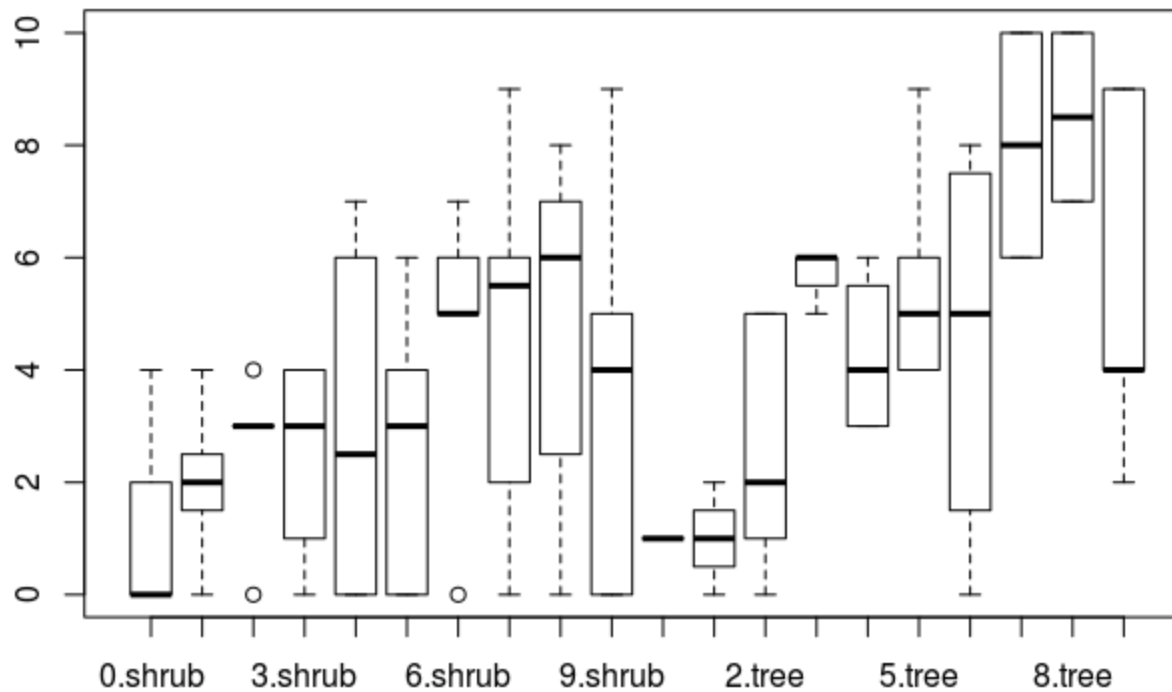
- forループを使うと？ => 複数の図を10枚でも100枚でも一気にかけます

```
par(mfrow = c(1, 2), mar = c(3, 4.3, 2, 1), ps = 18)
type <- c("tree", "shrub")
for (i in 1:2) {
  # iを1から2まで順番に変える
  cpue.site <- subset(cpue.data, plant == type[i])
  boxplot(cpue.site$count ~ cpue.site$year, ylab = "CPUE", col = "lightblue",
    title(type[i]))
}
```



# 別のやり方1

- 1行でそれっぽいのもかけますが、



## # 別のやり方2: ggplot2の利用

- ggplot2とは：きれいなグラフを簡単に書くためのパッケージ

```
library(ggplot2)
ggplot(data = cpue.data, aes(x = factor(year), y = count, fill =
  facet_grid(~. + plant) + theme_classic(base_size = 20, base
```

