

# 診断・予後解析研究におけるデータ解析

---

1. 診断の性能指標

2. ROC

3. 参考

## 診断の性能指標

---

## 復習: 診断の性能指標

- ・ 感度 sensitivity / 偽陰性率 false-negative rate
- ・ 特異度 specificity / 偽陽性率 false-positive rate
- ・ 陽性予測値、陽性反応の中率 positive predictive value
- ・ 陰性予測値、陰性反応の中率 negative predictive value

## 復習: 診断の性能指標

### 診断法の適応集団での頻度分布

	検査陽性	検査陰性	合計
疾患あり	$Y_1$	$N_1 - Y_1$	$N_1$
疾患なし	$Y_0$	$N_0 - Y_0$	$N_0$
合計	$S$	$T$	$N$

### 各セルに対応する用語

	検査陽性	検査陰性
疾患あり	真陽性 (True Positive)	偽陰性 (False Negative)
疾患なし	偽陽性 (False Positive)	真陰性 (True Negative)

## 復習: 診断の性能指標

	検査陽性	検査陰性	合計
疾患あり	$Y_1$	$N_1 - Y_1$	$N_1$
疾患なし	$Y_0$	$N_0 - Y_0$	$N_0$
合計	$S$	$T$	$N$

- ・ 正しい診断の割合・確率:

- ・ 感度 (Sensitivity) :  $Se = \frac{Y_1}{N_1}$

- ・ 特異度 (Specificity) :  $Sp = \frac{N_0 - Y_0}{N_0}$

- ・ 誤った診断の割合・確率:

- ・ 偽陰性率 (False Negative Rate) :  $F_n = \frac{N_1 - Y_1}{N_1} = 1 - Se$

- ・ 偽陽性率 (False Positive Rate) :  $F_p = \frac{Y_0}{N_0} = 1 - Sp$

- ・ 診断の有用度に関わる指標:

- ・ 陽性予測値 (Positive Predictive Value) :  $PPV = \frac{Y_1}{S}$

- ・ 陰性予測値 (Negative Predictive Value) :  $NPV = \frac{N_0 - Y_0}{T}$

## epiR パッケージの epi.tests 関数

- ・ 使い方

```
epi.tests(dat, method = "exact", digits = 2, conf.  
          level = 0.95)
```

- ・ dat:  $2 \times 2$  分割表
- ・ method: 信頼区間の算出方法 (exact: 二項分布に基づく方法)
- ・ digits: output の小数点以下の桁数
- ・ conf.level: 信頼区間のレベル
- ・ 表形式で出力される
  - ・ 詳細は `summary(epi.tests(...))` もしくは `epi.tests(...)$detail` で確認可能

## 例: epi.tests

```
install.packages("epiR")  
library(epiR)
```

→ パッケージの読み込み

```
tab <- matrix(c(80, 20, 10, 90),  
              nrow = 2, byrow = TRUE)  
colnames(tab) <- c(" 検査陽性", " 検査陰性")  
rownames(tab) <- c(" 疾患あり", " 疾患なし")  
print(tab)
```

2×2 表

行：疾患有無

列：検査結果

```
epi.tests(t(tab))
```

 → epi.tests 用に転置

**演習：**上記のコードを実行し、各指標を計算してみましょう



## 出力例: epi.tests

```
> epi.tests(t(tab))
```

	Outcome +	Outcome -	Total
Test +	80	10	90
Test -	20	90	110
Total	100	100	200

入力した分割表

Point estimates and 95% CIs:

点推定値 信頼区間

Apparent prevalence *	0.45	(0.38, 0.52)
True prevalence * 有病率	0.50	(0.43, 0.57)
Sensitivity * 感度	0.80	(0.71, 0.87)
Specificity * 特異度	0.90	(0.82, 0.95)
Positive predictive value * 陽性予測値	0.89	(0.81, 0.95)
Negative predictive value * 陰性予測値	0.82	(0.73, 0.89)
Positive likelihood ratio	8.00	(4.41, 14.52)
Negative likelihood ratio	0.22	(0.15, 0.33)
False T+ proportion for true D- *	0.10	(0.05, 0.18)
False T- proportion for true D+ *	0.20	(0.13, 0.29)
False T+ proportion for T+ *	0.11	(0.05, 0.19)
False T- proportion for T- *	0.18	(0.11, 0.27)
Correctly classified proportion *	0.85	(0.79, 0.90)

\* Exact CIs

## 有病率と PPV/NPV の関係

- ・ 感度・特異度が一定でも、陽性予測値（PPV）と陰性予測値（NPV）は有病率に依存
- ・ 実臨床では、検査前確率としての有病率を考慮することが重要

感度  $Se$ 、特異度  $Sp$ 、有病率  $p$  のとき：

$$PPV = \frac{p \cdot Se}{p \cdot Se + (1 - p) \cdot (1 - Sp)}$$

$$NPV = \frac{(1 - p) \cdot Sp}{p \cdot (1 - Se) + (1 - p) \cdot Sp}$$

## 実習: 有病率の PPV・NPV への影響

感度、特異度ともに 95% の診断の PPV・NPV が有病率によってどの様になるか

```
print_ppv_npv <- function(p, Se=0.95, Sp=0.95){  
  ppv <- (p * Se) / (p * Se + (1 - p) * (1 - Sp))  
  npv <- ((1 - p) * Sp) / (p * (1 - Se) + (1 - p) *  
  Sp)  
  cat(sprintf(" 有 病 率: %.4f の と き PPV: %.3f, NPV:  
  %.3f\n", p, ppv, npv))  
}
```

→ 感度、特異度、有病率から PPV・NPV を算出する関数の作成

```
print_ppv_npv(0.1)  
print_ppv_npv(0.01)  
print_ppv_npv(0.001)  
print_ppv_npv(0.0001)
```

→ 有病率を変えて出力

演習: 上記のコードを実行し、各指標を計算してみましょう

## 出力例: 有病率の PPV・NPV への影響

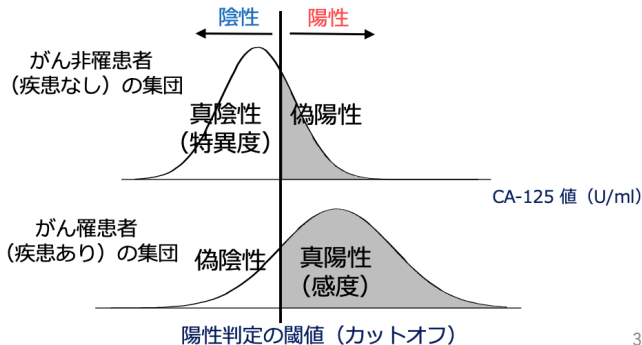
```
> print_ppv_npv(0.1)
有病率: 0.1000 のとき PPV: 0.679, NPV: 0.994
> print_ppv_npv(0.01)
有病率: 0.0100 のとき PPV: 0.161, NPV: 0.999
> print_ppv_npv(0.001)
有病率: 0.0010 のとき PPV: 0.019, NPV: 1.000
> print_ppv_npv(0.0001)
有病率: 0.0001 のとき PPV: 0.002, NPV: 1.000
```

- ・ 有病率は PPV・NPV に影響
- ・ どのような集団での診断かに注意が必要

ROC

---

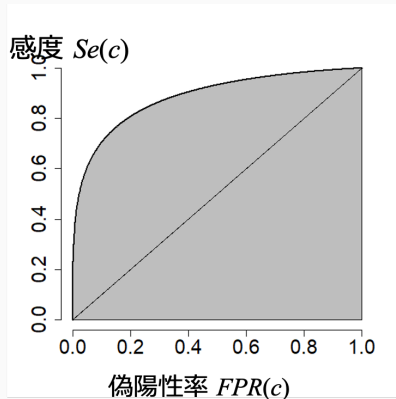
### 再掲：腫瘍マーカー．血清中の糖鎖抗原 CA-125（U/ml）を用いた膀胱がんの検出



30

- ・ 閾値（カットオフ）を動かすと感度、特異度の値は変化
- ・ 一方が上がると他方が下がるトレードオフの関係
- ・ 閾値を変化させたときの指標の推移をみる  $\Rightarrow$  ROC

## 復習: ROC 曲線



- ・ 閾値を変化させたときの、偽陽性率（1 - 特異度）と感度の関係をプロット
- ・ 曲線の下面積（AUC）が大きいほど、分類性能が高い
  - ・ AUC= 1: 完璧な診断
  - ・ AUC= 0.5: ランダムな診断
- ・ カットオフ選択にも使える

## 各関数の基本的な使い方

- 関数 `roc`: ROC オブジェクトの作成

`roc(response=アウトカムのベクトル, predictor=連続的な予測値のベクトル)`

- `response`: 2 値の真の疾患の状態 (例: 0 = 肺炎, 1 = 肺がん)
  - `predictor`: 連続的な予測値 (例: マーカーの値)
  - `direction`: 「陽性」を大きい値で表すなら "<" (自動推定されることが多い)
  - 出力の class は "roc": ROC オブジェクト
- 関数 `auc`: AUC (曲線下面積) の計算

`auc(ROCオブジェクト)`

- `roc()` で作成された ROC オブジェクトを入力



## 各関数の基本的な使い方

- 関数 `plot`: ROC 曲線の描画

```
plot(ROCオブジェクト, legacy.axes = TRUE)
```

- ・ 厳密には自動で `plot.roc` 関数が呼び出されている
  - ・ `legacy.axes = TRUE` により「1 - 特異度」vs「感度」の軸
  - ・ 様々なオプションで調整が可能
- 関数 `ggroc`: `ggplot2` 形式で ROC 曲線を描画

```
ggroc(roclist, legacy.axes = TRUE)+...
```

- ・ `roclist`: ROC オブジェクトのリスト（重ね書きが容易）
- ・ `legacy.axes = TRUE` により「1 - 特異度」vs「感度」の軸

## 実習: ROC 曲線の描画

データはここからダウンロード

```
install.packages("pROC", dep=TRUE)
library(pROC)
df <- read.csv("tumor_marker.csv")
head(df)
```

パッケージ読み込み  
データ読み込み

```
roc1 <- roc(df$cancer, df$CA19.9)
auc(roc1)
plot(roc1, legacy.axes=TRUE)
```

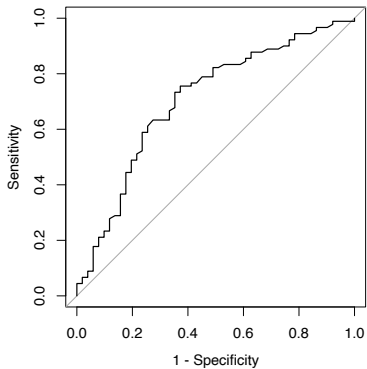
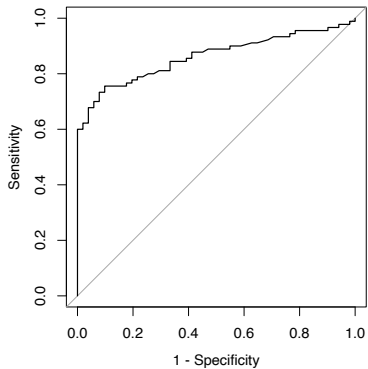
→ CA19-9 による ROC

```
roc2 <- roc(df$cancer, df$CA125)
auc(roc2)
plot(roc2, legacy.axes=TRUE)
```

→ CA125 による ROC

**演習**：上記のコードを実行し、ROC を描画してみましょう

# 出力例



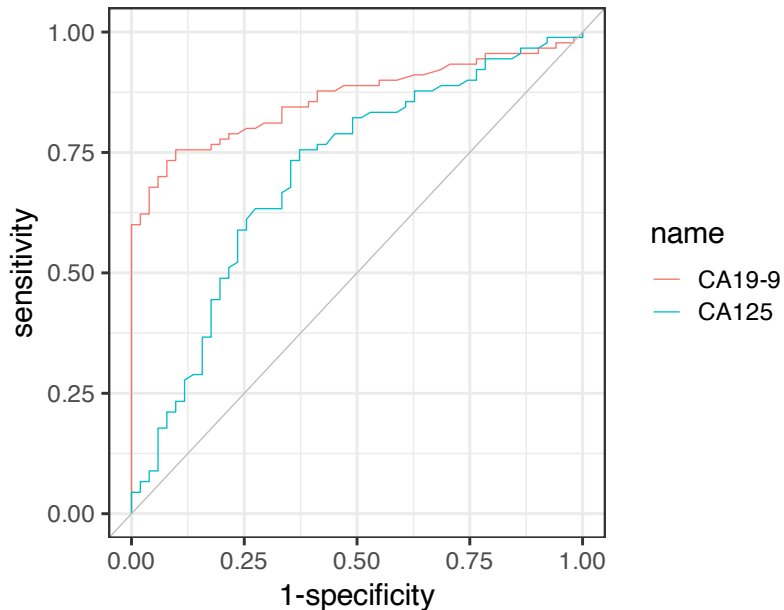
## 実習: ROC 曲線の描画

```
library(ggplot2)
roclist <- list(roc1,roc2)
names(roclist) <- c("CA19-9","CA125")
ggroc(roclist,legacy.axes = TRUE)+
geom_abline(intercept=0, slope=1,col="grey")+
theme_bw(base_size = 15)
```

→ リストと ggplot で重ねて表示

**演習**：上記のコードを実行し、ROC を描画してみましょう

## 出力例



## 参考

---

pROC パッケージの `roc.test` 関数を使うと、2 つの ROC 曲線を統計的に比較可能。

- ・ 使い方

```
roc.test(roc1, roc2, method = "delong")
```

- ・ `roc1, roc2`: 比較したい 2 つの ROC オブジェクト
- ・ `method`: デフォルトは "delong"
- ・ その他: "bootstrap", "venkatraman" など選択可
- ・ 出力される内容
  - ・ AUC の差の有意性検定 (帰無仮説: AUC に差はない)
  - ・ 信頼区間や p 値

## 参考: AUC と Concordance の関係

- ・ AUC は、疾患あり群（陽性群）のマーカ一値が疾患なし群（陰性群）より高い確率として解釈できる

$$\text{AUC} = \Pr(X_1 > X_0) + \frac{1}{2} \Pr(X_1 = X_0)$$

- ・  $X_1$ : 疾患ありのマーカ一値
- ・  $X_0$ : 疾患なしのマーカ一値
- ・ 有限標本内では、全ペアを取り出し、正しい順序 ( $X_1 > X_0$ ) になっている割合

$$\text{AUC} = \frac{1}{n_0 n_1} \sum_{i=1}^{n_1} \sum_{j=1}^{n_0} I(x_1^{(i)} > x_0^{(j)}) + \frac{1}{2} I(x_1^{(i)} = x_0^{(j)})$$

- ・  $n_1, n_0$  はそれぞれ疾患あり群, 疾患なし群のサンプルサイズ
- ・  $x_1^{(i)}, x_0^{(j)}$  はそれぞれ疾患あり群の  $i$  番目のサンプルのマーカ一値 ( $i = 1, \dots, n_1$ ), 疾患なし群の  $j$  番目のサンプルのマーカ一値 ( $j = 1, \dots, n_0$ ).
- ・ Mann-Whitney U statistic と関連:  $\text{AUC} = \frac{U}{n_0 n_1}$



**演習：** 腫瘍マーカーのデータセットの CA19-9 に対する AUC が疾患あり群（陽性群）のマーカー値が疾患なし群（陰性群）より高い確率（疾患あり群と疾患なし群の全ペアの中で、疾患あり群（陽性群）のマーカー値が疾患なし群（陰性群）より高い割合）と一致することを確認してください

```
library(pROC)
roc_obj <- roc(df$cancer, df$CA19.9)
auc(roc_obj)

allcomb <- combn(1:nrow(df),2)
concordance <- rep(NA,ncol(allcomb))
for(i in 1:ncol(allcomb)){
  index <- allcomb[,i]
  dfi <- df[index,]
  if(length(unique(dfi$cancer))!=1){
    diff <- dfi$CA19.9[dfi$cancer==1] - dfi$CA19.9[dfi$cancer==0]
    conc <- ifelse(diff>0,1,ifelse(diff==0,0.5,0))
  }else{
    conc <- NA
  }
  concordance[i] <- conc
}

mean(concordance,na.rm=TRUE)
auc(roc_obj)
```