R で学ぶデータ解析とシミュレーション

⑤

～ グラフとデータマイニング ～

５時間目のメニュー

- イントロ ←
  - データ「iris」の紹介
  - 1 つの変数を要約する
  - 2 つの変数の関係を見る
- CART の紹介
- データ「iris」でお試し

Graphic by (c)Tomo.Yun (http://www.yunphoto.net)

グラフとデータマイニング ⇒ データ「iris」

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
・・・	・・・	・・・	・・・	・・・

- フィッシャーが判別分析法を紹介するために利用したアヤメの品種分類（Species：setosa，versicolor，virginica）に関するデータ
  ⇒ 以下の４変数を説明変数としてアヤメの種類を判別しようとした
  - アヤメのがくの長さ（Sepal.Length）
  - アヤメのがくの幅　（Sepal.Width）
  - アヤメの花弁の長さ（Petal.Length）
  - アヤメの花弁の幅　（Petal.Width）

グラフとデータマイニング ⇒ データ「iris」

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3.0	1.4	0.1	setosa
・・・	・・・	・・・	・・・	・・・

- データを眺めてもよく分からない… ⇒ データを要約する！

グラフとデータマイニング ⇒ 1 変数の要約

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
・・・	・・・	・・・	・・・	・・・

- アヤメのがくの長さ（Sepal.Length）の特徴をつかむには・・・

1. 数値による要約 ⇒ 要約統計量を求める
2. グラフによる要約 ⇒ ヒストグラムを作成する
3. 層別して要約統計量やヒストグラム

グラフとデータマイニング ⇒ 要約統計量

```
> x <- iris                    # データ iris の読み込み
> summary(x$Sepal.Length)      # 要約統計量を算出する
   Min. 1st Qu.  Median    Mean   3rd Qu.    Max.
  4.300   5.100   5.800   5.843     6.400   7.900
```

一番小さい値
（最小値）

真ん中の値
（中央値，平均値）

一番大きい値
（最大値）

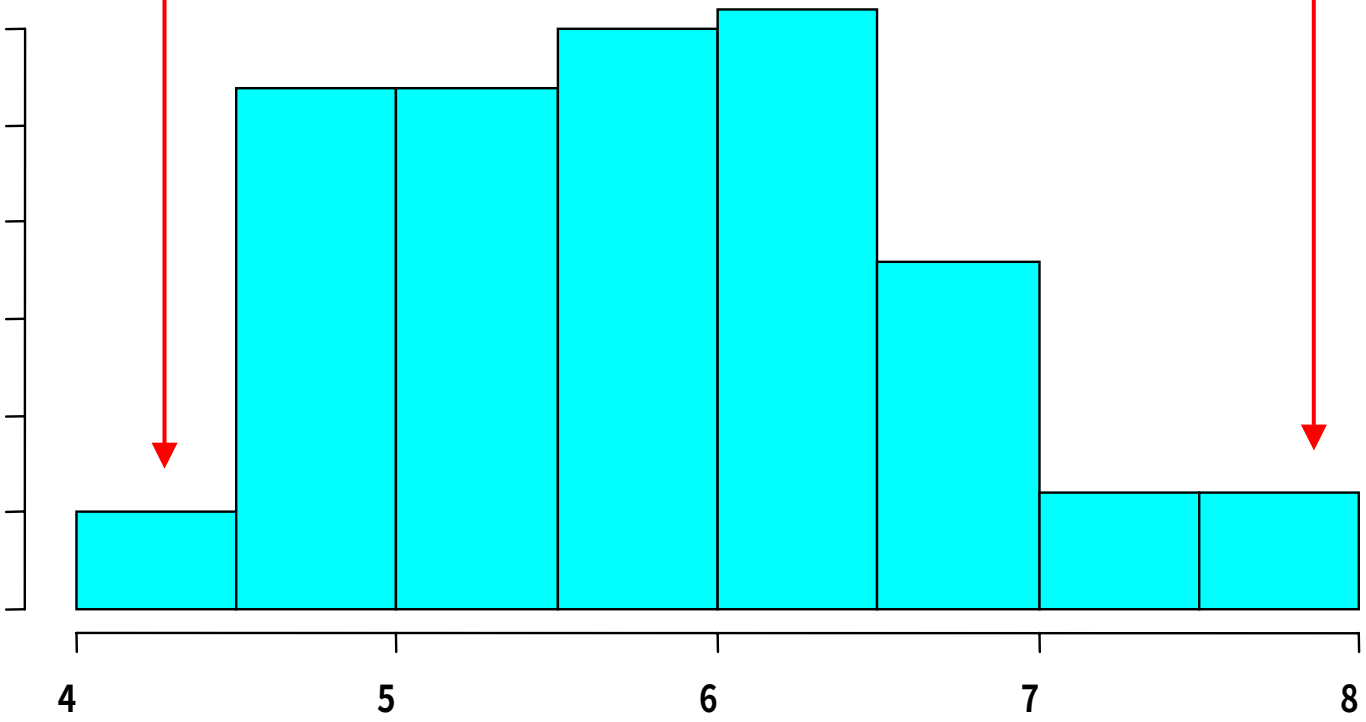
- 要約統計量を眺めてもよく分からない… ⇒ データをグラフにする！

（注）要約統計量を算出するのも大事なお仕事です^^;

グラフとデータマイニング ⇒ ヒストグラム

```
> x <- iris                    # データ iris の読み込み
> hist(x$Sepal.Length)         # ヒストグラムを作成する
   Min. 1st Qu.  Median    Mean   3rd Qu.    Max.
  4.300   5.100   5.800   5.843     6.400   7.900
```

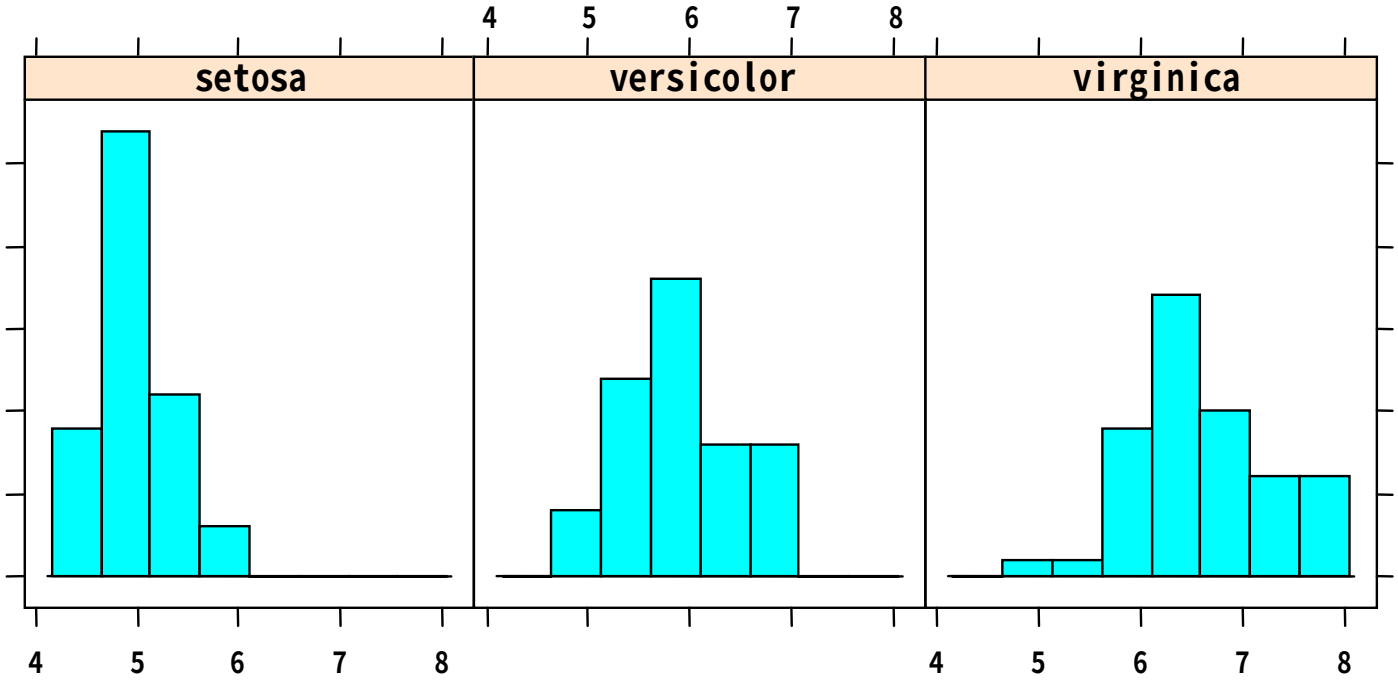
一番小さい値　　真ん中の値　　一番大きい値



一目瞭然！

グラフとデータマイニング ⇒ 層別解析

```
> library(lattice)                                  # おまじない
> histogram(~ Sepal.Length | Species, data=x)       # 層別ヒストグラム
```



- Setosa　：がくが短い
- Verginica：がくが長い

⇒ 層別すると特徴が浮き出る！

一目瞭然！

グラフとデータマイニング ⇒ 2 変数の関係

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
・・・	・・・	・・・	・・・	・・・

- アヤメの花弁の幅（Petal.Width）と花弁の長さ（Petal.Length）の関係を見る場合は・・・

1. 数値による要約 ⇒ 相関係数を求める

2. グラフによる要約 ⇒ 散布図を描く

3. 層別してグラフ（散布図）を描く

グラフとデータマイニング ⇒ 2 変数の関係

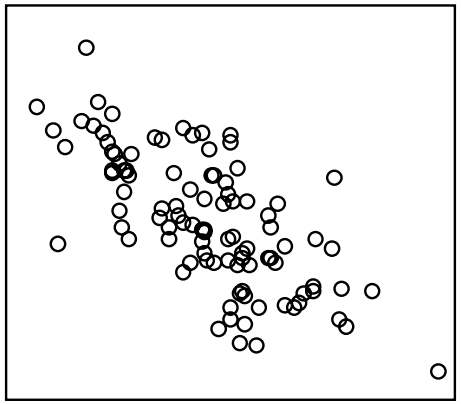
```
> # Petal.Width と Petal.Length の相関係数を算出する
> cor(x$Petal.Width, x$Petal.Length)
[1] 0.9628654
```

- 花弁の幅（Petal.Width）と長さ（Petal.Length）の関係を調べるには
  1. 相関係数を算出する
  2. グラフを描く
  3. 層別にグラフを描く

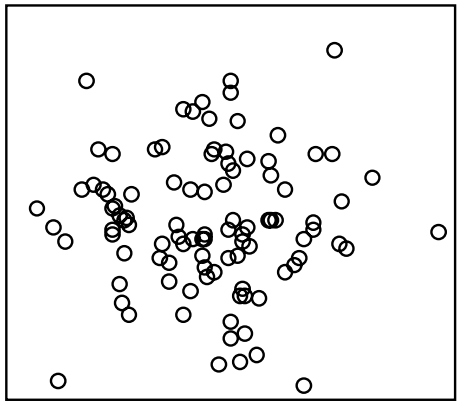
```
> # Petal.Width と Petal.Length の散布図を描く
> library(lattice)
> xyplot(x$Petal.Width ~ x$Petal.Length, ps=20)
> # Petal.Width と Petal.Length の層別散布図を描く
> xyplot(x$Petal.Width ~ x$Petal.Length,
+        col=as.integer(x$Species), pch=as.integer(x$Species), ps=20)
```

グラフとデータマイニング ⇒ 相関係数

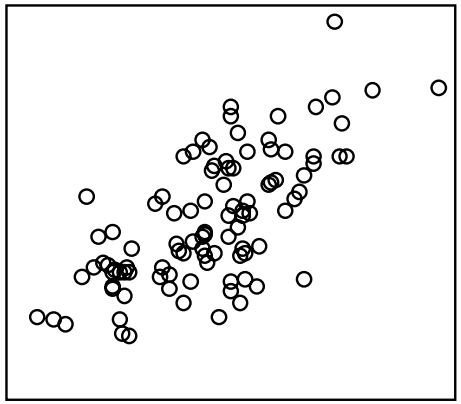
r = - 0.7
（負の相関）



r = 0
（相関なし）

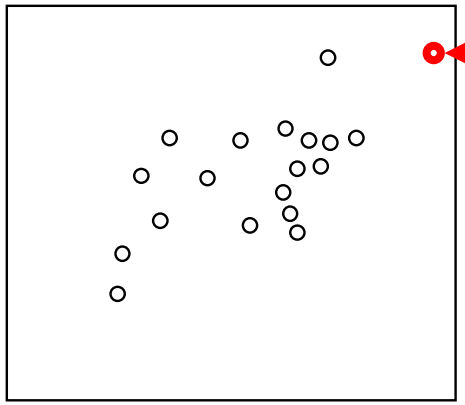


r = 0.7
（正の相関）



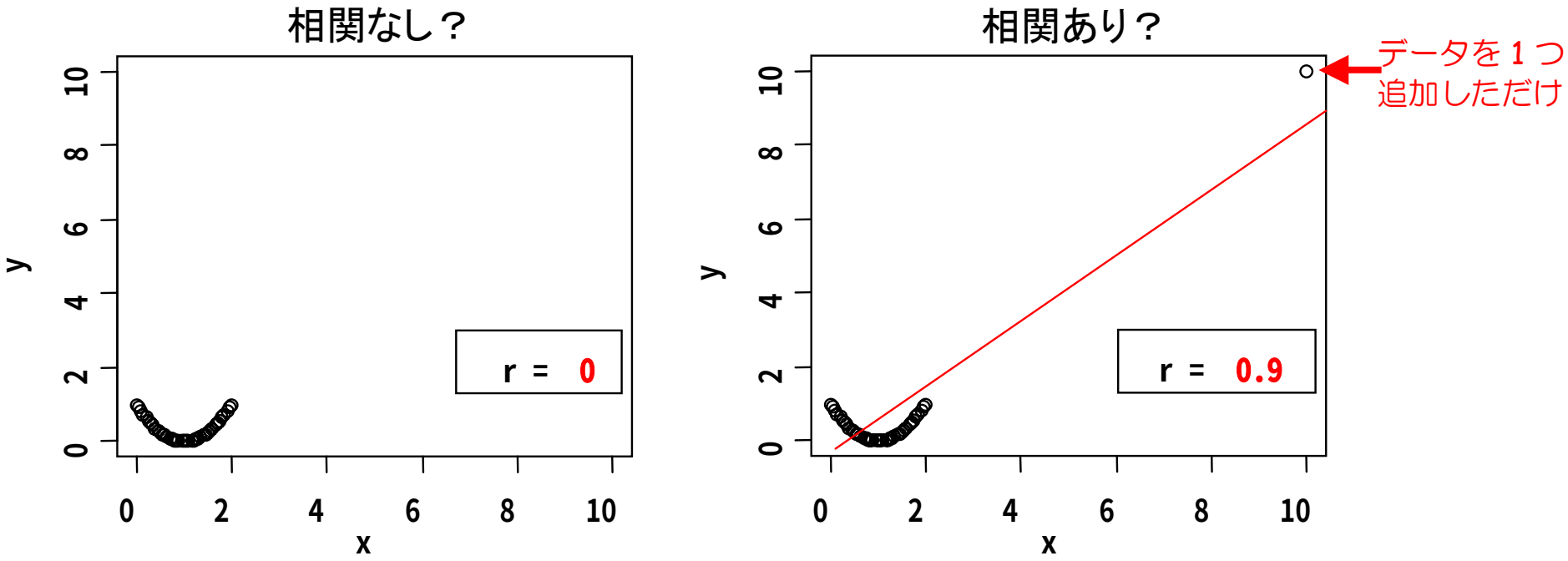
- 相関係数とは？
  - -1 ～ 1 の間の値をとる値
  - 1 に近い：右肩上がり
  - -1 に近い：右肩下がり
  - 中途半端な値のときは解釈が難しい… ⇒

r = 0.7
（相関あり？）



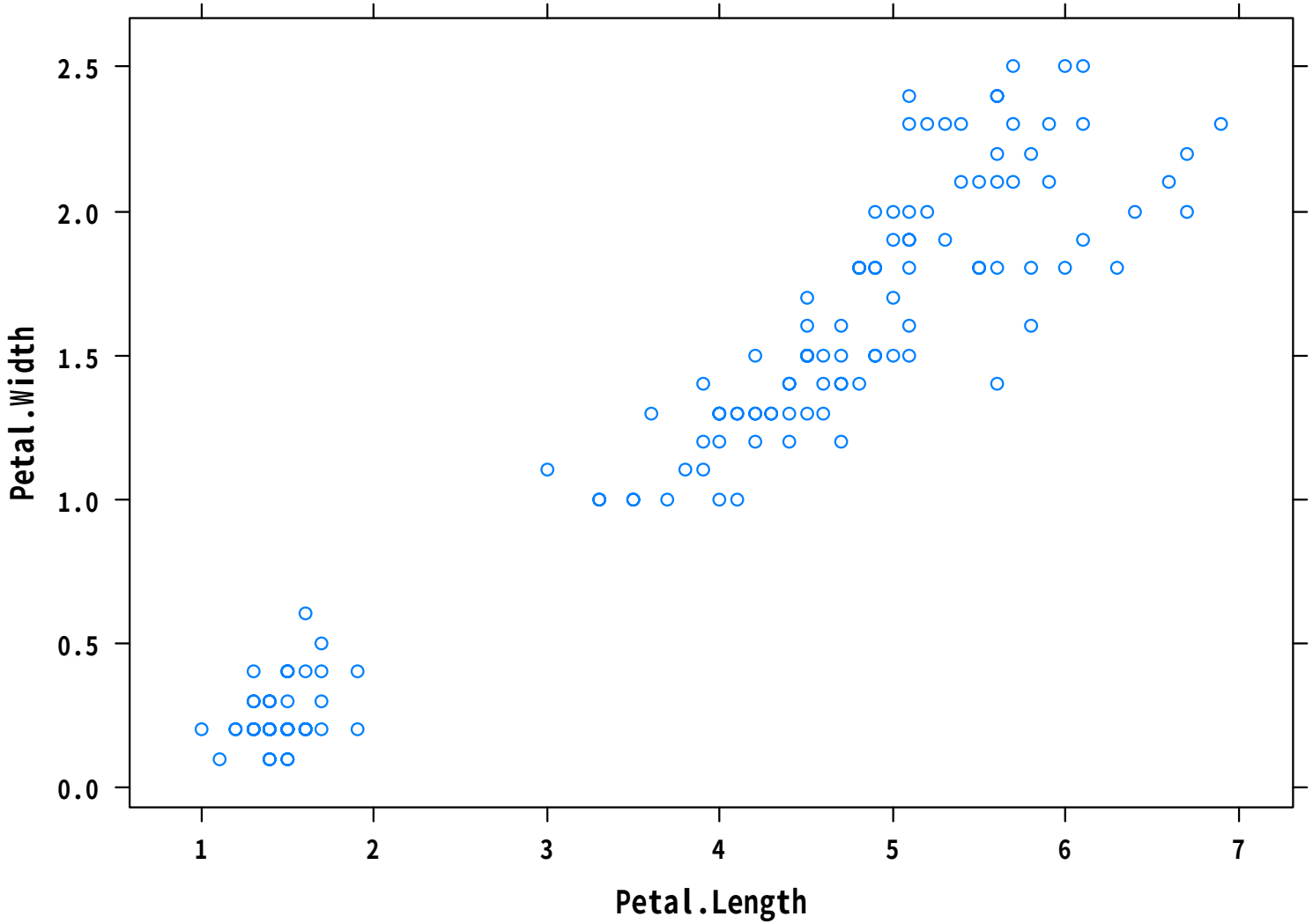
このデータを除いたら相関なし？

グラフとデータマイニング ⇒ 相関係数



- 花弁の幅（Petal.Width）と長さ（Petal.Length）の関係を調べるには
  1. 相関係数を算出する ⇒ よく分からない場合が多い…
  2. グラフを描く ⇒ 分かる
  3. 層別にグラフを描く ⇒ 非常によく分かる！

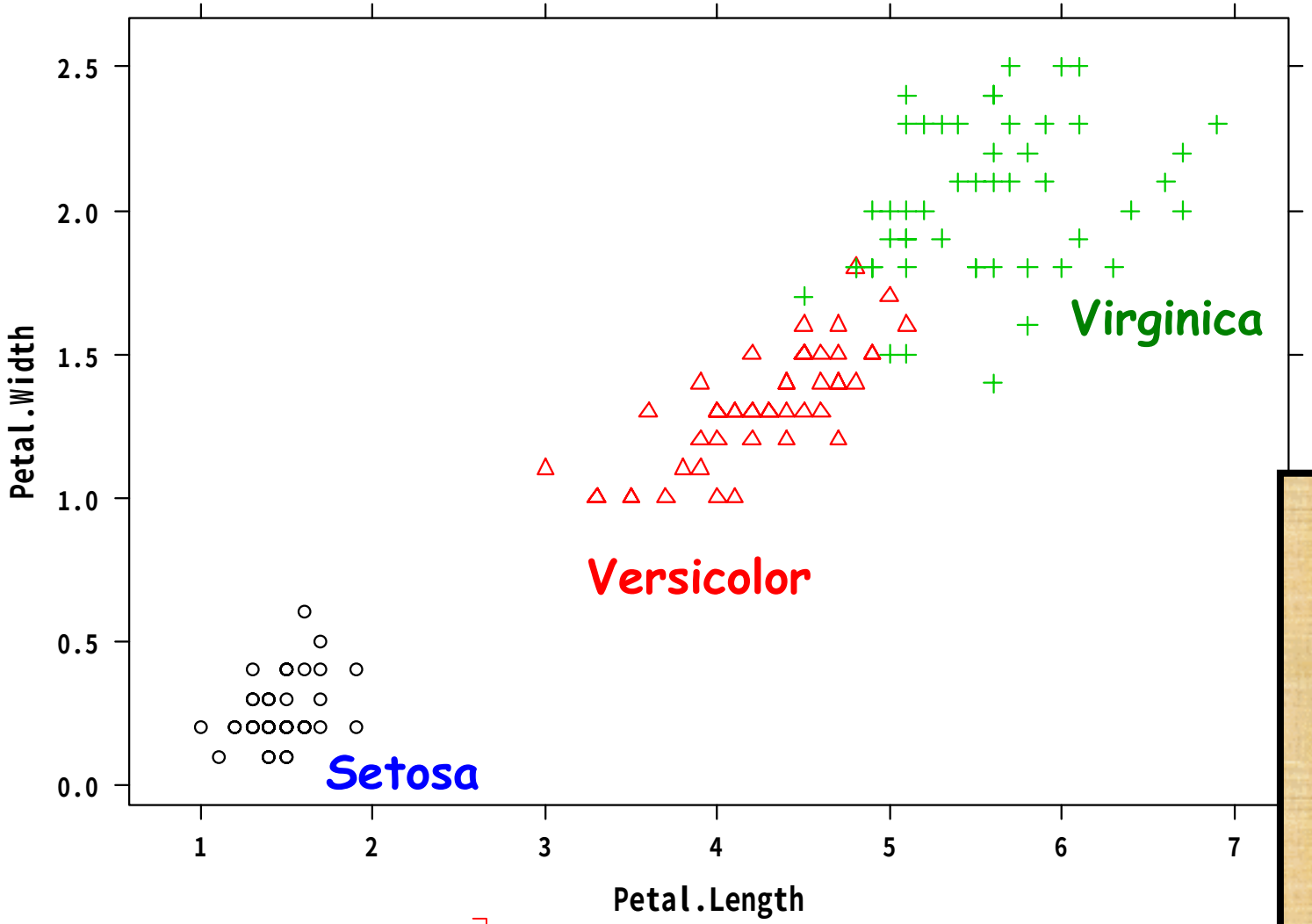
グラフとデータマイニング ⇒ 2 変数の関係



一目瞭然！

- Petal.Width と Petal.Length の関係は右肩上がり ⇒ ひと目で分かる！

グラフとデータマイニング ⇒ 層別解析



一目瞭然！

- Setosa　：左下に分布
- Verginica：右上に分布

⇒ 層別すると特徴が浮き出る！

重要なのは・・・

層別解析！

ここまでのまとめ

■ データ解析を行う手順は

1. データを R Commander に読み込ませる
2. データ解析を実行する
3. 結果を保存する，結果をレポートにまとめる　etc…

■ 具体的なデータ解析例

★ 1 つの変数について要約する場合：

❑ 数値による要約（要約統計量）を計算する前に，まずグラフ！

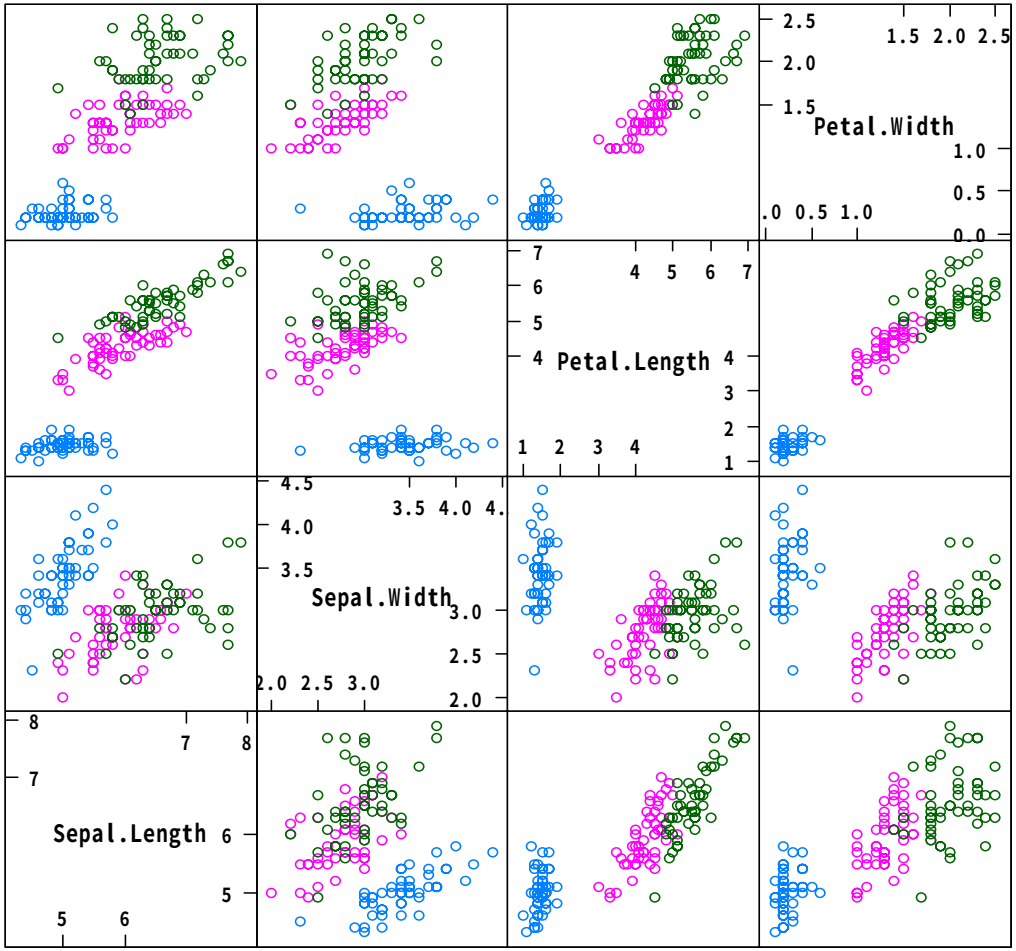
❑ 注目しているカテゴリ（アヤメの種類など）があるなら層別解析！

★ 2 つの変数間の関係を見る場合：

❑ 数値による要約（相関係数）を計算する前に，まずグラフ！

❑ 注目しているカテゴリ（アヤメの種類など）があるなら層別解析！

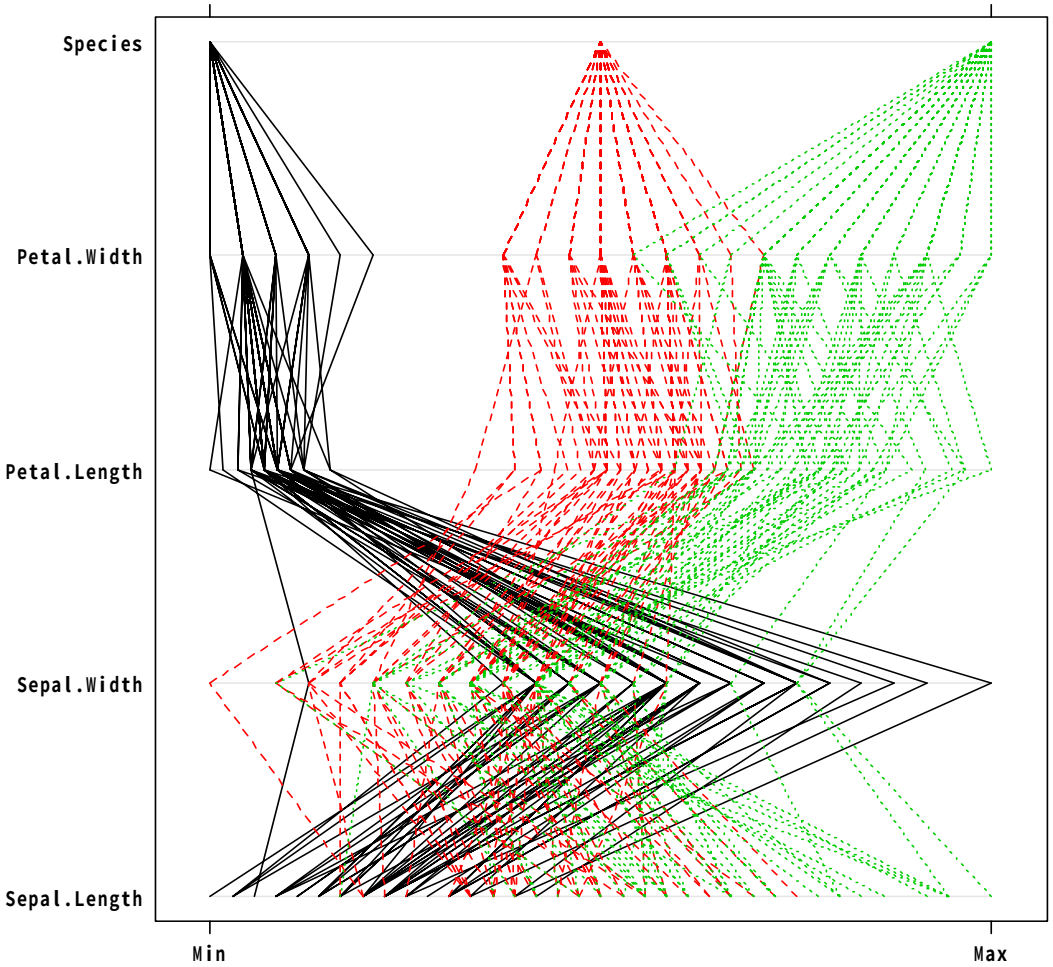
グラフとデータマイニング ⇒ その他



Scatter Plot Matrix

```
> library(lattice)
> splom(~ x[,1:4], groups=Species, data=x)    # group=グループを表す変数
```

グラフとデータマイニング ⇒ その他




```
> library(lattice)
> parallel(x, col=as.integer(x$Species), # parallel(データフレーム名)
+            lty=as.integer(x$Species))
```

グラフとデータマイニング ⇒ その他

```
> x <- iris
> cor(x[,-5], method = "spearman")            # 相関係数
             Sepal.Length Sepal.Width Petal.Length Petal.Width
Sepal.Length    1.0000000  -0.1667777    0.8818981   0.8342888
Sepal.Width    -0.1667777   1.0000000   -0.3096351  -0.2890317
Petal.Length    0.8818981  -0.3096351    1.0000000   0.9376668
Petal.Width     0.8342888  -0.2890317    0.9376668   1.0000000

> symnum(abs(cor(x[-5])), cutpoints=c(0, 0.2, 0.4, 0.6, 0.8, 1),
+        symbols=c(" ", ".", "-", "+", "*")) # 相関係数
             S.L S.W P.L P.W
Sepal.Length *      *   *
Sepal.Width     *   -   .
Petal.Length *   -   *   *
Petal.Width  *   .   *   *
attr(,"legend")
[1] 0 ' ' 0.2 '.' 0.4 '-' 0.6 '+' 0.8 '*' 1
```

５時間目のメニュー

- イントロ
  - データ「iris」の紹介
  - 1 つの変数を要約する
  - 2 つの変数の関係を見る
- **CART の紹介 ←**
  - **CART とは？**
  - **CART の作成方法**
  - **CART の剪定作業**
- データ「iris」でお試し

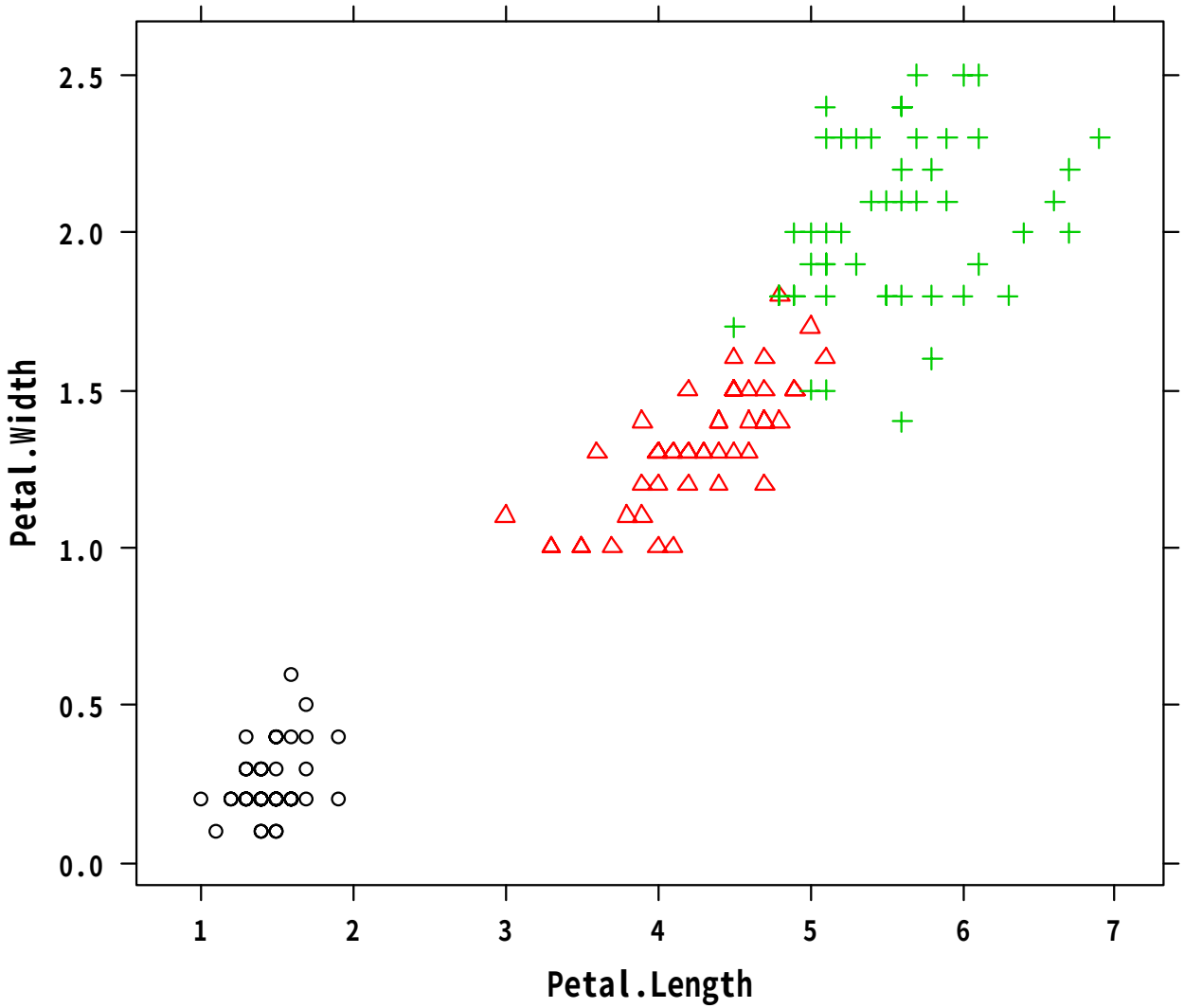
Graphic by (c)Tomo.Yun (http://www.yunphoto.net)

グラフとデータマイニング ⇒ 分類・予測

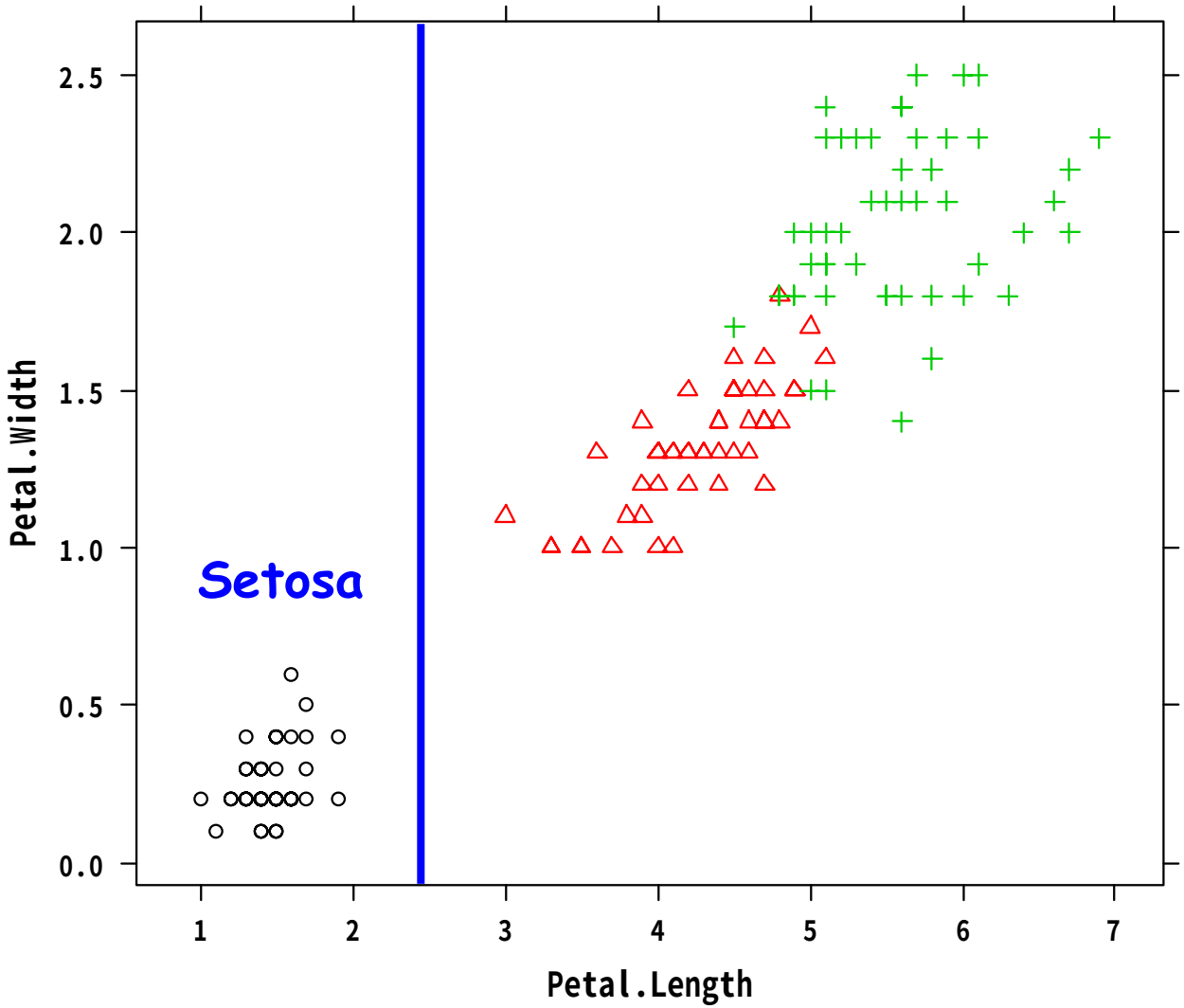
Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
・・・	・・・	・・・	・・・	・・・

- ここまでは「アヤメの種類（Species）で層別するのが重要！」というお話でした
- 今度は逆に「他の変数からアヤメの種類を予測する」ことは出来る？
- 例えば「花弁の長さ（Petal.Length）が●●以下ならば setosa 」のような分類ルールを作ることは出来る？

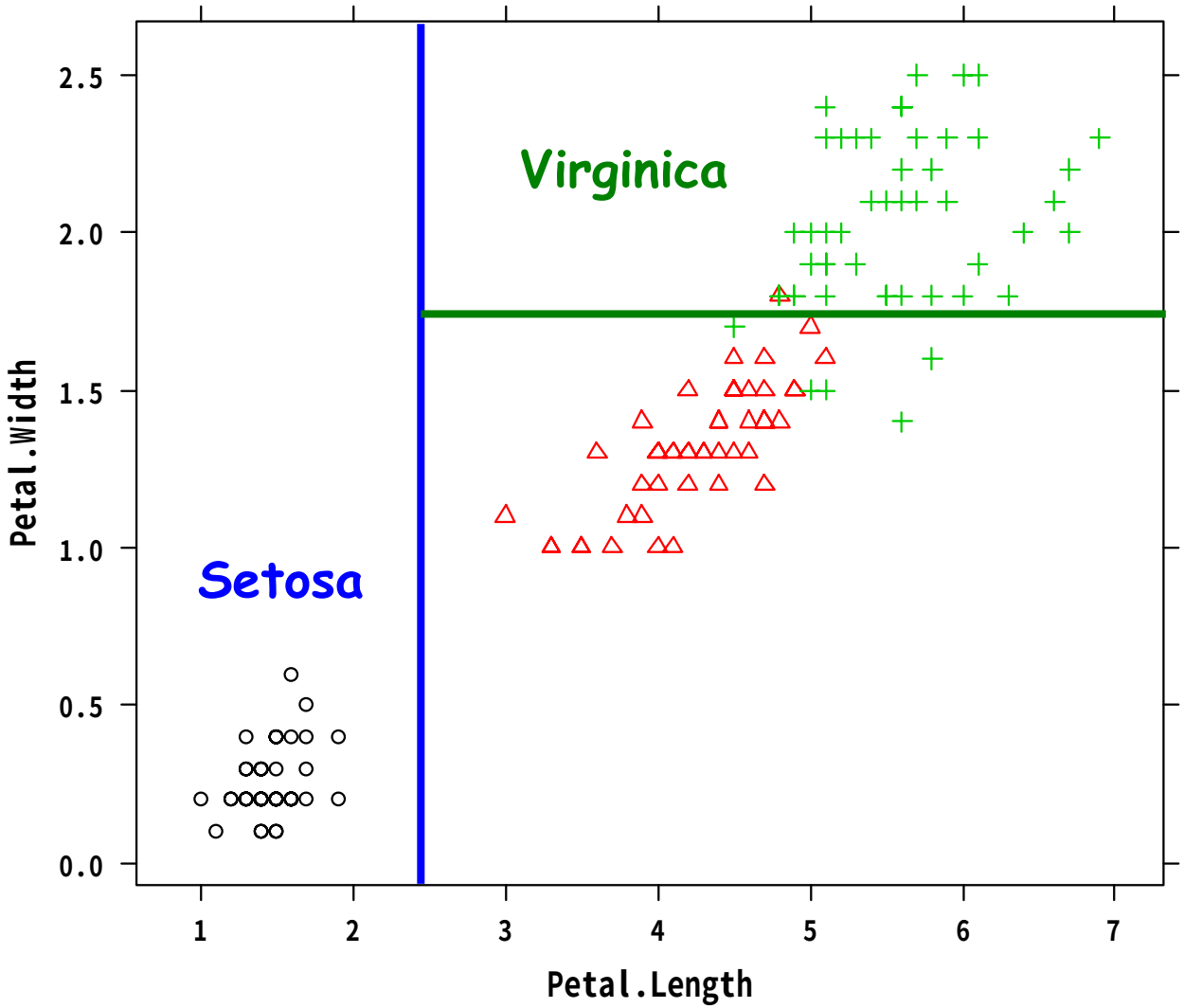
グラフとデータマイニング ⇒ 分類・予測



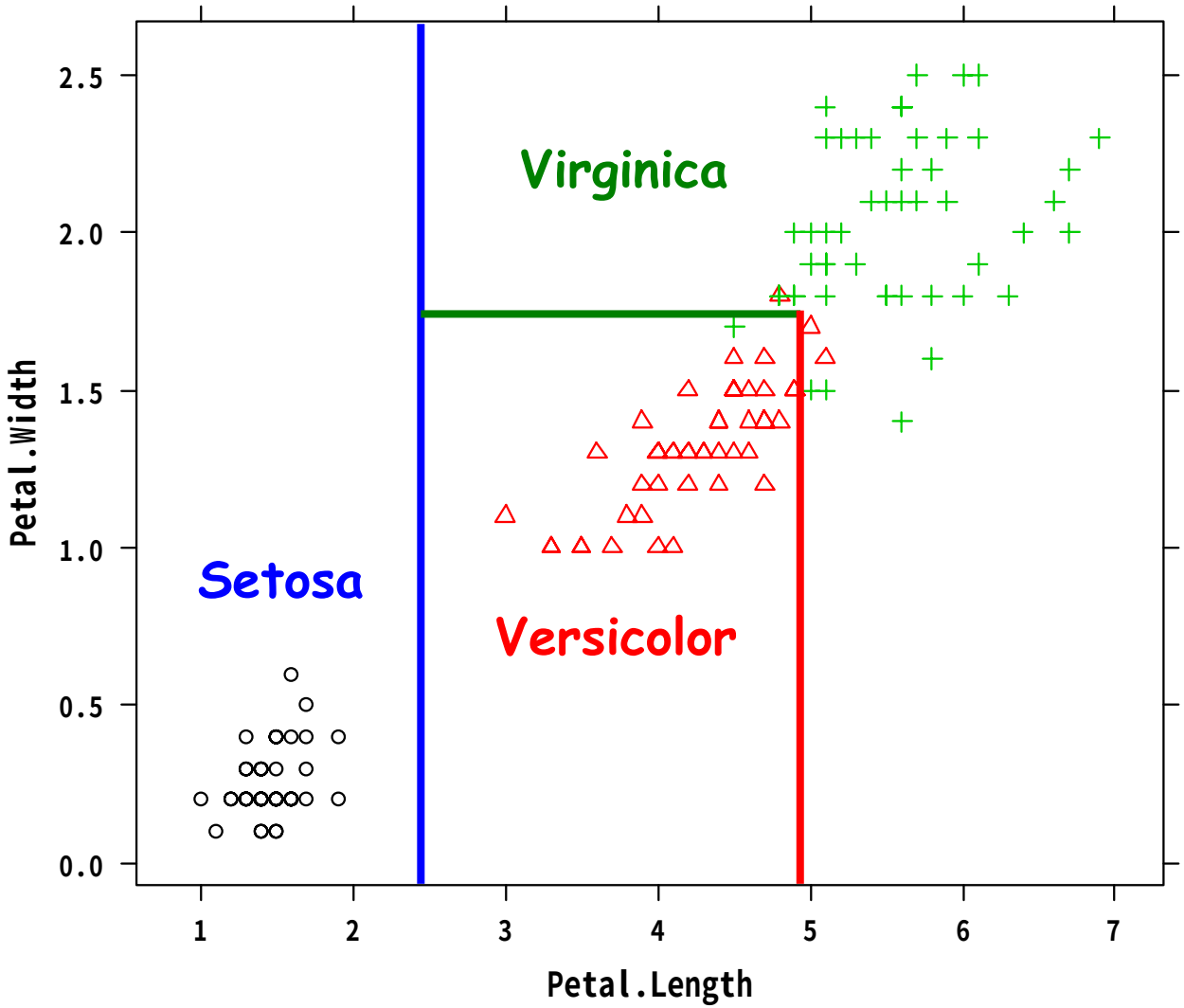
グラフとデータマイニング ⇒ 分類・予測



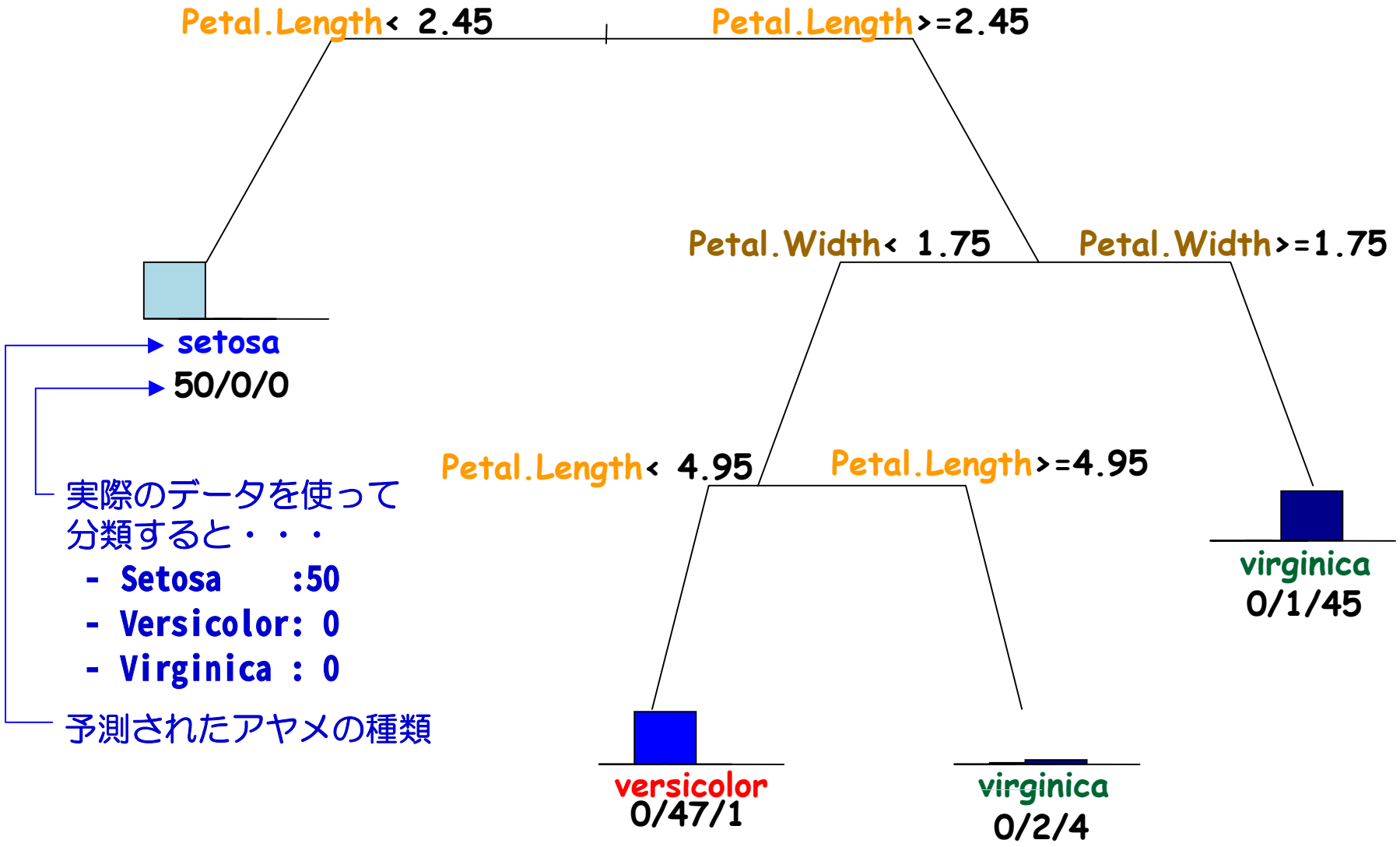
グラフとデータマイニング ⇒ 分類・予測



グラフとデータマイニング ⇒ 分類・予測



前頁のスライドをルール化 ⇒ 分類木（CART）



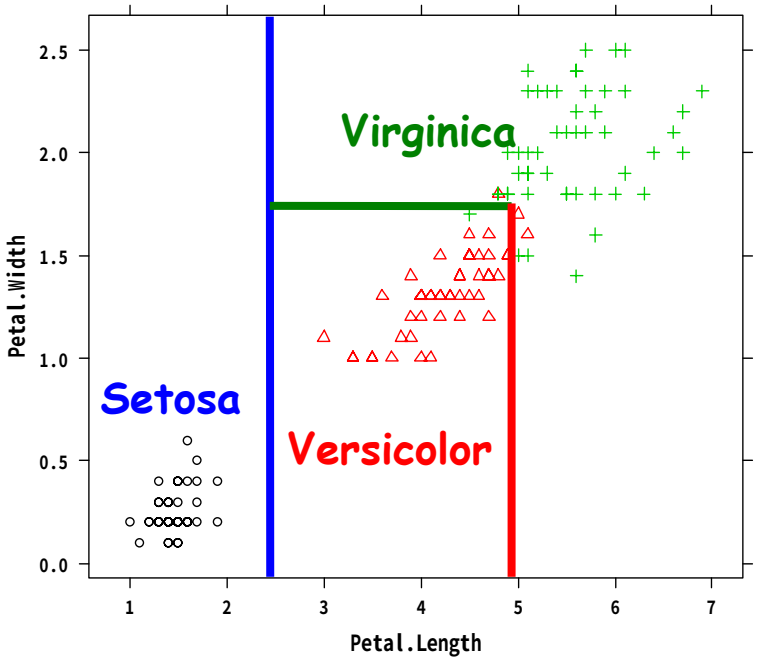
実際のデータを使って
分類すると・・・

- Setosa :50
- Versicolor: 0
- Virginica : 0

予測されたアヤメの種類

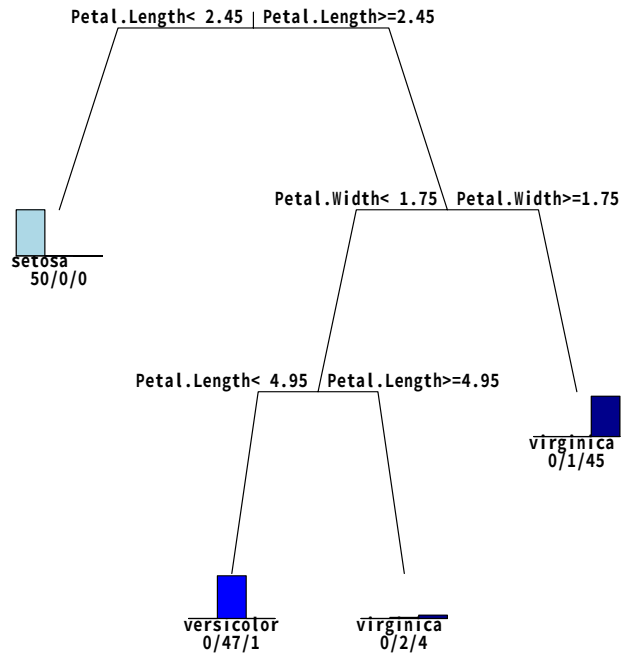
CART（Classification and Regression Trees）とは？

- あるルールに従ってデータを分け，分類や予測を行う
- 目的変数がカテゴリ：分類木 ← この iris の例！
- 目的変数が連続変数：回帰木 ← 次のスライド



ルール化！

（分類・予測）



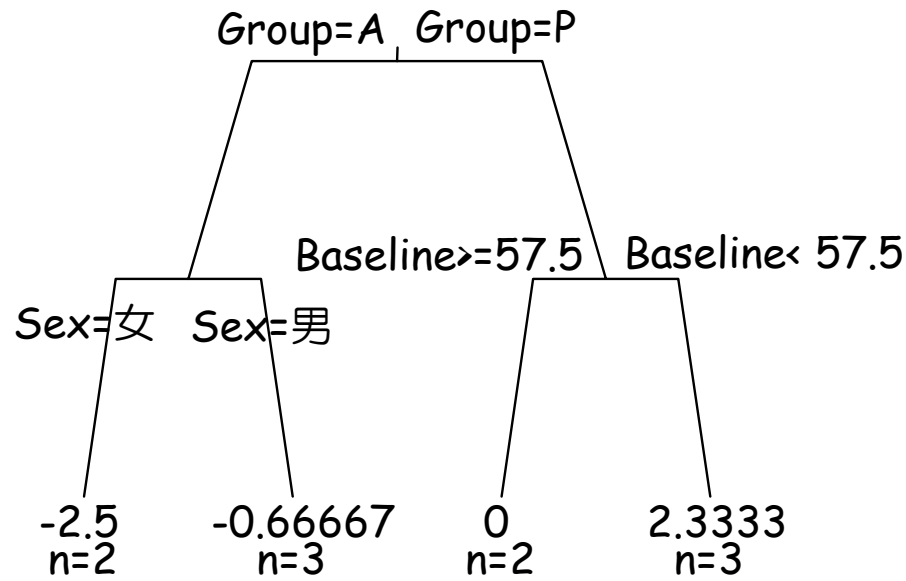

```
> library(rpart)        # library(mvpart)
> result <- rpart(Species ~ ., data=x)
> plot.new()            # デバイスを開く
```

```
> par(xpd=T) # 文字が切れないための命令
> plot(result, uniform=T, branch=0.7)
> text(result, use.n=T, all.leaves=F)
```

回帰木の例

- 男女 10 人にやせ薬（A）と偽薬（P）を飲んでもらう
- 飲みはじめの体重を測り，1 ヶ月飲み続けた後，再度測定
- 体重の変化量（kg）を目的変数として回帰木を作成

Difference (体重の変化)	Group (薬剤)	Sex (性別)	Baseline (前値)
1	A	Man	55
-1	A	Man	65
-2	A	Man	70
-2	A	Woman	45
-3	A	Woman	50
3	P	Woman	50
2	P	Woman	55
2	P	Woman	55
1	P	Man	60
-1	P	Man	65



（各カテゴリの平均値と例数が表示される）

分類木・回帰木の分岐基準

- 各ノードの不純度（impurity）を $I(A)$、 p_A をノードA 内の確率分布（ the class distribution ）とすると
  - Entropy： $I(A) = i(p_A) = -\Sigma_j p_{Aj} \log p_{Aj}$
  - Gini index： $I(A) = i(p_A) = 1 - \Sigma_j P_{Aj}^2$
- 平方和：$SS_* =$ 平方和 $= \Sigma_j (y_j - y)^2$
- ノード A をノード A_L とノード A_R に分割する場合は、以下の ΔI を最大化するような分割ルールを選択する
  - 分類木：$\Delta I = P(A)I(A) - P(A_L)I(A_L) - P(A_R)I(A_R)$
  - 回帰木：$\Delta I = \{SS_T - (SS_L + SS_R)\}/N$

分類木について

Improve (体重改善?)	Group (薬剤)	Sex (性別)
No	A	Man
Yes	A	Man
Yes	A	Man
Yes	A	Woman
Yes	A	Woman
No	P	Woman
No	P	Woman
No	P	Woman
No	P	Man
Yes	P	Man

- 分岐なし
  - I(A) = 1-((5/10)^2+(5/10)^2)=0.5
- 薬剤で分岐
  - $I(A_A)$= 1-((4/5)^2+(1/5)^2) = 0.32
  - $I(A_P)$= 1-((1/5)^2+(4/5)^2) = 0.32
  - ⊿I = 0.5-5/10*0.32-5/10*0.32 = 0.18
- 性別で分岐
  - $I(A_{男})$= 1-((3/5)^2+(2/5)^2) = 0.48
  - $I(A_{女})$= 1-((2/5)^2+(3/5)^2) = 0.48
  - ⊿I = 0.5-5/10*0.48-5/10*0.48 = 0.02

☆ 0.18 > 0.02 ⇒「薬剤」で分岐させる方が良い

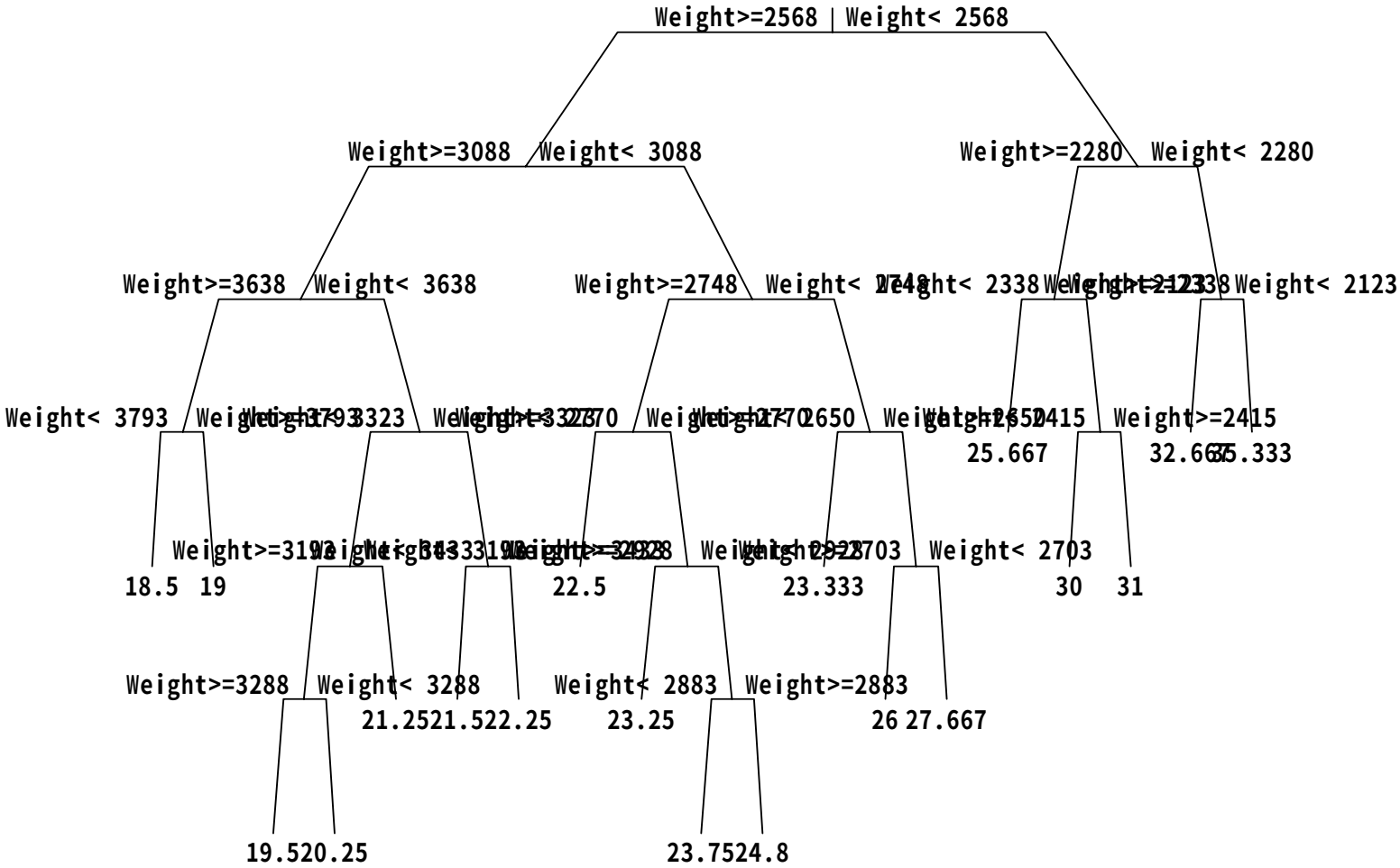
回帰木について

Difference (体重の変化)	Group (薬剤)	Sex (性別)
1	A	Man
-1	A	Man
-2	A	Man
-2	A	Woman
-3	A	Woman
3	P	Woman
2	P	Woman
2	P	Woman
1	P	Man
-1	P	Man

- 全体の平方和
  - $SS_T = \sum_j (y_j - \bar{y})^2 = 38$
- 薬剤で分岐した場合
  - $SS_A = \sum_j (y_{Aj} - \bar{y}_A)^2 = 9.2$
  - $SS_P = \sum_j (y_{Pj} - \bar{y}_P)^2 = 9.2$
  - $\Delta I = SS_T - (SS_L + SS_R) = \underline{19.6}$
- 性別で分岐した場合
  - $SS_{男} = \sum_j (y_{男j} - \bar{y}_{男})^2 = 7.2$
  - $SS_{女} = \sum_j (y_{女j} - \bar{y}_{女})^2 = 29.2$
  - $\Delta I = SS_T - (SS_L + SS_R) = \underline{1.6}$

☆ 19.2 > 1.6 ⇒「薬剤」で分岐させる方が良い

CART は放っておくとどこまでも枝分かれしていく・・・



分類木・回帰木の剪定

- 分類木・回帰木は放っておくとどこまでも枝分かれしていく
  ⇒ 剪定（Pruning）が必要！
- 剪定（Pruning）：リスクと複雑度を評価する（AICみたいな）
  - cp：複雑度パラメータ（complexity parameter）
  - クロスバリデーションで各「木」の リスク R(・) を評価する
    （分岐が無い木の誤判別率 = 1 となるように調整）
    - 分類木の場合：誤判別率（relative risk）を用いる
    - 回帰木の場合：SS_* を用いる
    - 「木」T の良さを以下の値（Cost-complexity）で評価する
      $R_\alpha(T) = R(T) + cp \times T$ のノード数
      （↑を最小にするような T の部分木を選択する）
    - 最適な cp の値はクロスバリデーションで評価する

回帰木の場合

	複雑度 CP	分岐数 nsplit	リスクR(・) rel error	CVで求めたリスク xerror	CVで求めたリスクのSD xstd
1	0.5157895	0	1.0000000	1.234568	0.3005857
2	0.1719298	1	0.4842105	0.756579	0.2794117
3	0.1061404	2	0.3122807	0.756579	0.2794117
4	0.0100000	3	0.2061404	0.756579	0.2794117

■ 最適な木の探索手順

① 複雑度 cp を 0.01 にした上で木を構築（↑が作成した木のリスト）
⇒ 初期値は 0.01，全ての木を出力する場合は 0 にすればよい

② 各木について，クロスバリデーションでリスクと SE を計算する

③ 上記リストの中でリスク（xerror）が最小となるものを探す（↑では **2** ）

④ ③の木の xerror＋1SE を算出する（↑では 0.756579+0.2794117）

⑤ リストを上からなぞり，④の値を初めて下回った木が最適な木（↑では **2** ）

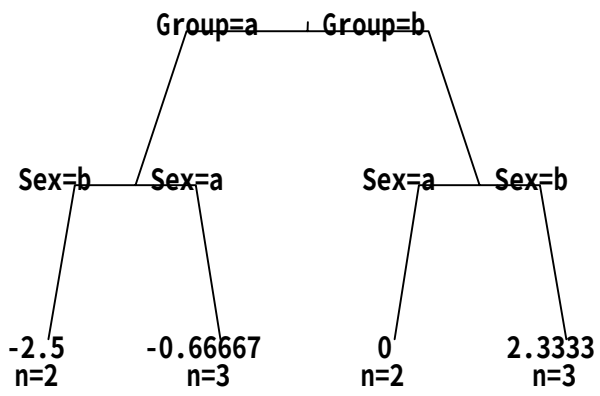
❑ ⑤の意味：木の葉（ノード）の数が少なくても，木の葉の数が大きい場合のリスクとそれほど変わらない，という意味

回帰木の場合

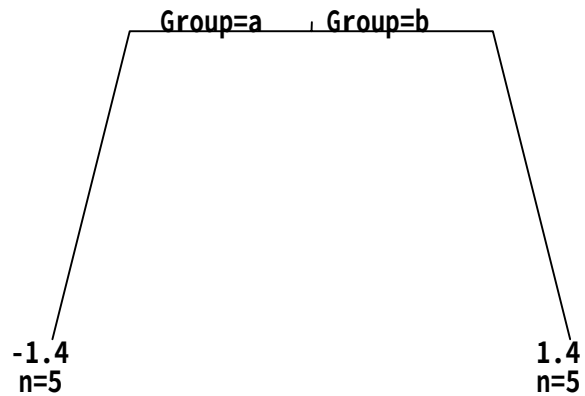
	複雑度 CP	分岐数 nsplit	リスクR(・) rel error	CVで求めたリスク xerror	CVで求めたリスクのSD xstd
1	0.5157895	0	1.0000000	1.234568	0.3005857
2	0.1719298	1	0.4842105	0.756579	0.2794117
3	0.1061404	2	0.3122807	0.756579	0.2794117
4	0.0100000	3	0.2061404	0.756579	0.2794117

剪定

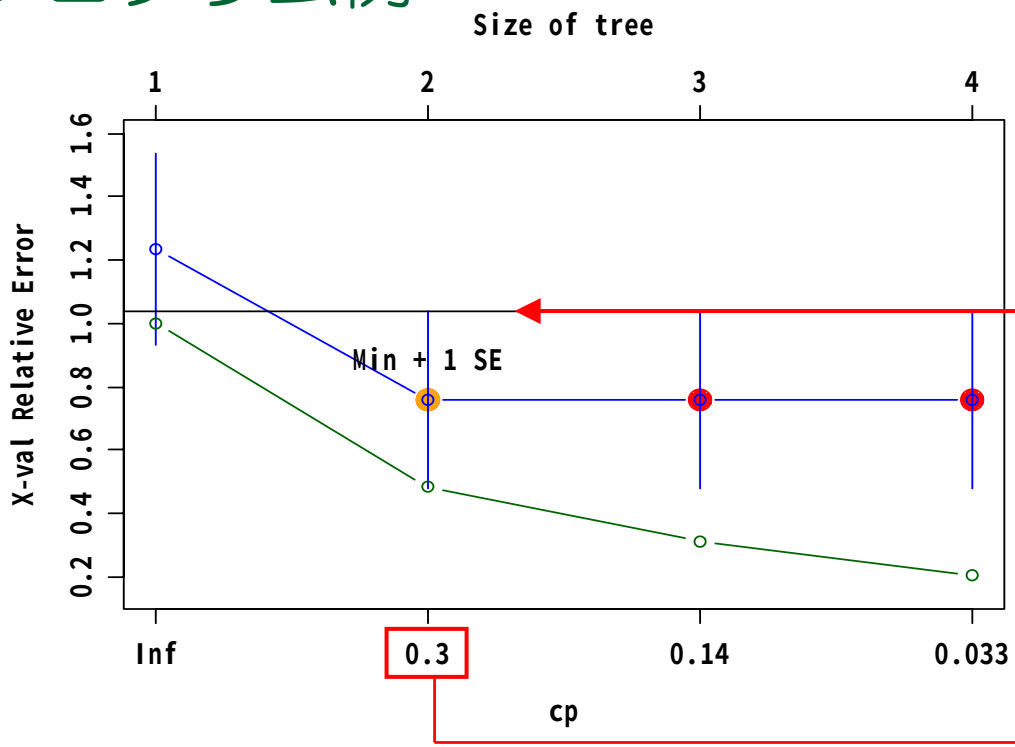
	CP	nsplit	rel error	xerror	xstd
1	0.5157895	0	1.0000000	1.234568	0.3005857
2	0.3000000	1	0.4842105	0.756579	0.2794117



剪定



R でのプログラム例



リスクの最小値＋SE
（0.75…+0.27… = 1.02）

```
y <- read.delim("clipboard")    # データ
library(rpart)          # library(mvpart)
result <- rpart(Difference ~ ., data=y)
plot.new(); par(xpd=T)
plot(result, uniform=T, branch=0.7,
     margin=0.05)
text(result, use.n=T, all.leaves=F)
```

```
summary(result) # 分類ルールを見る
plotcp(result)  # 最適な葉の数を調べる
result2 <- prune(result, cp=0.3)
plot(result2, uniform=T, branch=0.7,
     margin=0.05)
text(result2, use.n=T, all.leaves=F)
summary(result2)
```

リスク ＝ 相対誤判別率 R(・) について

Impr.	Diff.	Group (薬剤)	Sex (性別)
No	1	A	Man
Yes	-1	A	Man
Yes	-2	A	Man
Yes	-2	A	Woman
Yes	-3	A	Woman
No	3	P	Woman
No	2	P	Woman
No	2	P	Woman
No	1	P	Man
Yes	-1	P	Man

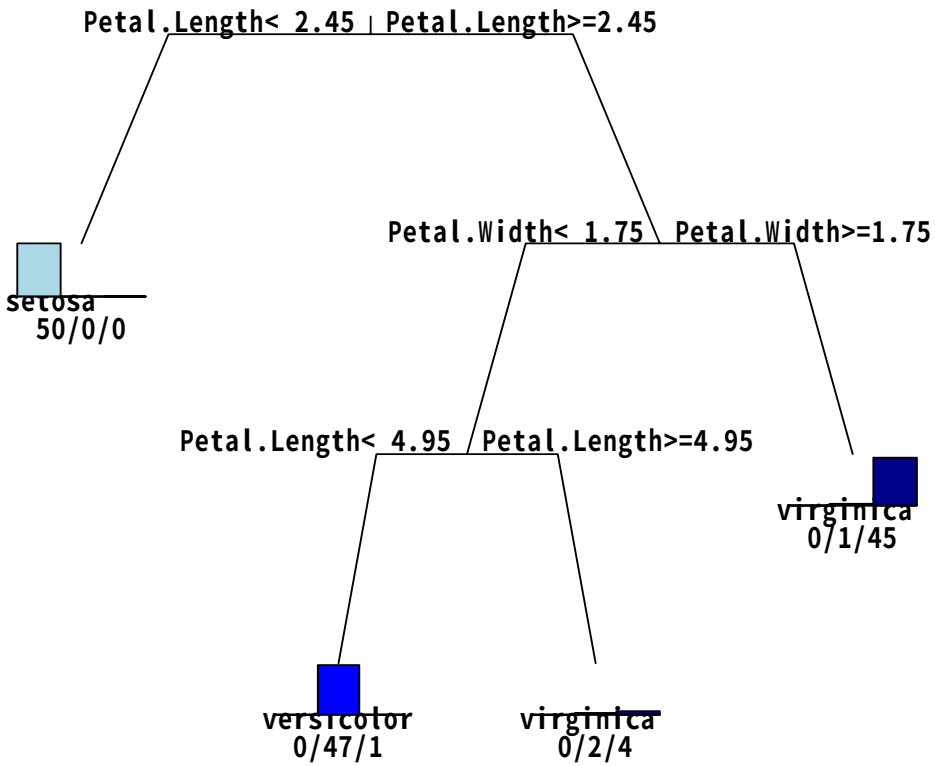
- 分類木の場合：
  - 分岐なしの木 T の誤判別率 = 5/10 = 0.5
  - Group で分岐した木の誤判別率 = 2/10 = 0.2
    ⇒ 分岐なしの木の R(t) = 1.0 ← 固定
    ⇒ Group で分岐した木の R(t) = 0.2/0.5 = 0.4
- 回帰木の場合：
  - 分岐なしの木 T の SS_T = 39.2
  - Group で分岐した木の $SS_L + SS_R$ = 7.2+29.2
    ⇒ 分岐なしの木の R(t) = 1.0 ← 固定
    ⇒ Group で分岐した木の R(t)
    = (7.2+29.2)/39.2 = 0.958
  - Group と Sex で分岐した木の
    $(SS_{ManL} + SS_{ManL} + SS_{Woman} + SS_{Woman})$
    = 4.67 + 2 + 0.67 + 0.5 = 7.84
    (各 Group の中で $SS_L + SS_R$ を算出する)
    ⇒ Group と Sex で分岐した木の R(t)
    = 7.84 / 39.2 = 0.2

５時間目のメニュー

- イントロ
  - データ「iris」の紹介
  - 1 つの変数を要約する
  - 2 つの変数の関係を見る
- CART の紹介
  - CART とは？
  - CART の作成方法
  - CART の剪定作業
- データ「iris」でお試し ←
  - お試し
  - 余談＋演習

Graphic by (c)Tomo.Yun (http://www.yunphoto.net)

データ「iris」でお試し




```
x <- iris                    # データの読み込み
library(rpart)               # library(mvpart)
result <- rpart(Species ~ ., data=x)
plot.new(); par(xpd=T)
plot(result, uniform=T, branch=0.7,
     margin=0.05)
text(result, use.n=T, all.leaves=F)
```

```
summary(result)
plotcp(result)
result2 <- prune(result, cp=???)
plot(result2, uniform=T, branch=0.7,
     margin=0.05)
text(result2, use.n=T, all.leaves=F)
summary(result2)
```

データ「iris」でお試し

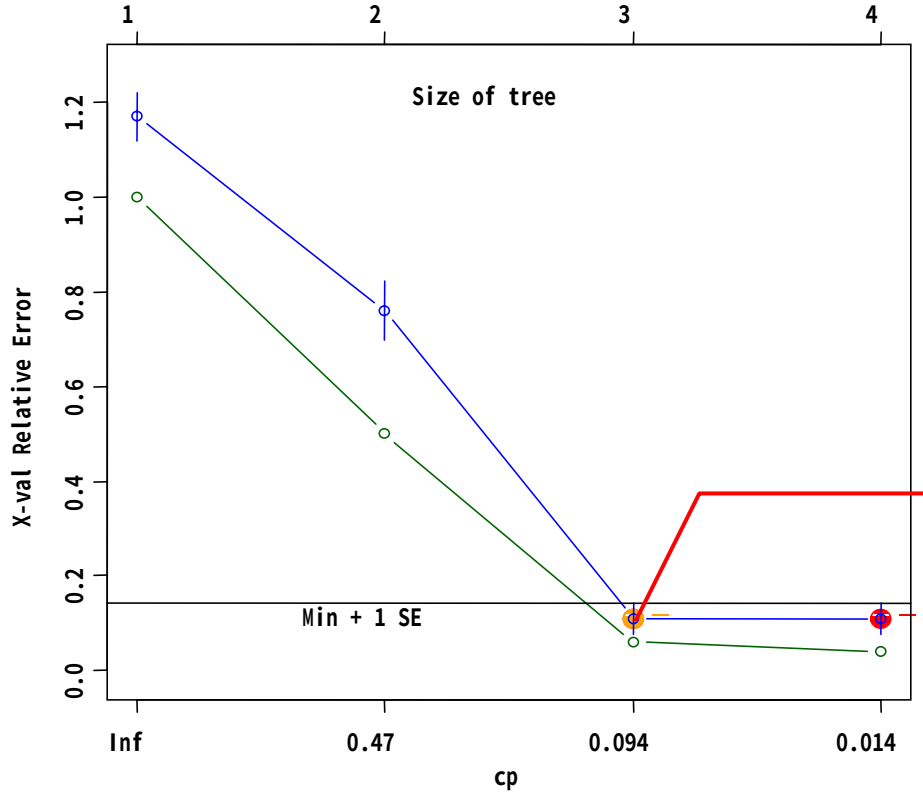
```
n= 150

    CP nsplit rel error xerror      xstd
1 0.50      0      1.00   1.17 0.0507346
2 0.44      1      0.50   0.76 0.0612318
3 0.02      2      0.06   0.11 0.0319270
4 0.01      3      0.04   0.11 0.0319270

Node number 1: 150 observations,    complexity param=0.5
  predicted class=setosa      expected loss=0.6666667
    class counts:    50    50    50
   probabilities: 0.333 0.333 0.333
  left son=2 (50 obs) right son=3 (100 obs)
  Primary splits:
      Petal.Length < 2.45 to the left,  improve=50.00000, (0 missing)
      Petal.Width  < 0.8  to the left,  improve=50.00000, (0 missing)
      Sepal.Length < 5.45 to the left,  improve=34.16405, (0 missing)
      Sepal.Width  < 3.35 to the right, improve=19.03851, (0 missing)
Node number 2: 50 observations
  predicted class=setosa      expected loss=0
    class counts:    50     0     0
   probabilities: 1.000 0.000 0.000
```

```
Node number 3: 100 observations,    complexity param=0.44
  predicted class=versicolor  expected loss=0.5
    class counts:     0    50    50
   probabilities: 0.000 0.500 0.500
  left son=6 (54 obs) right son=7 (46 obs)
  Primary splits:
      Petal.Width  < 1.75 to the left,  improve=38.969400, (0 missing)
      Petal.Length < 4.75 to the left,  improve=37.353540, (0 missing)
      Sepal.Length < 6.15 to the left,  improve=10.686870, (0 missing)
      Sepal.Width  < 2.45 to the left,  improve= 3.555556, (0 missing)
Node number 6: 54 observations,    complexity param=0.02
  predicted class=versicolor  expected loss=0.09259259
    class counts:     0    49     5
   probabilities: 0.000 0.907 0.093
  left son=12 (48 obs) right son=13 (6 obs)
  Primary splits:
      Petal.Length < 4.95 to the left,  improve=4.4490740, (0 missing)
      Petal.Width  < 1.35 to the left,  improve=0.9971510, (0 missing)
      Sepal.Length < 4.95 to the right, improve=0.6894587, (0 missing)
      Sepal.Width  < 2.65 to the right, improve=0.2500139, (0 missing)
Node number 7: 46 observations
  predicted class=virginica   expected loss=0.02173913
    class counts:     0     1    45
   probabilities: 0.000 0.022 0.978
Node number 12: 48 observations
  predicted class=versicolor  expected loss=0.02083333
    class counts:     0    47     1
   probabilities: 0.000 0.979 0.021
Node number 13: 6 observations
  predicted class=virginica   expected loss=0.3333333
    class counts:     0     2     4
   probabilities: 0.000 0.333 0.667
```

```
x <- iris
library(rpart)
result <- rpart(Species ~ ., data=x)
plot.new(); par(xpd=T)
plot(result, uniform=T, branch=0.7,
     margin=0.05)
text(result, use.n=T, all.leaves=F)
```

```
summary(result) # 分類ルールを見る
plotcp(result)
result2 <- prune(result, cp=???)
plot(result2, uniform=T, branch=0.7,
     margin=0.05)
text(result2, use.n=T, all.leaves=F)
summary(result2)
```

データ「iris」でお試し

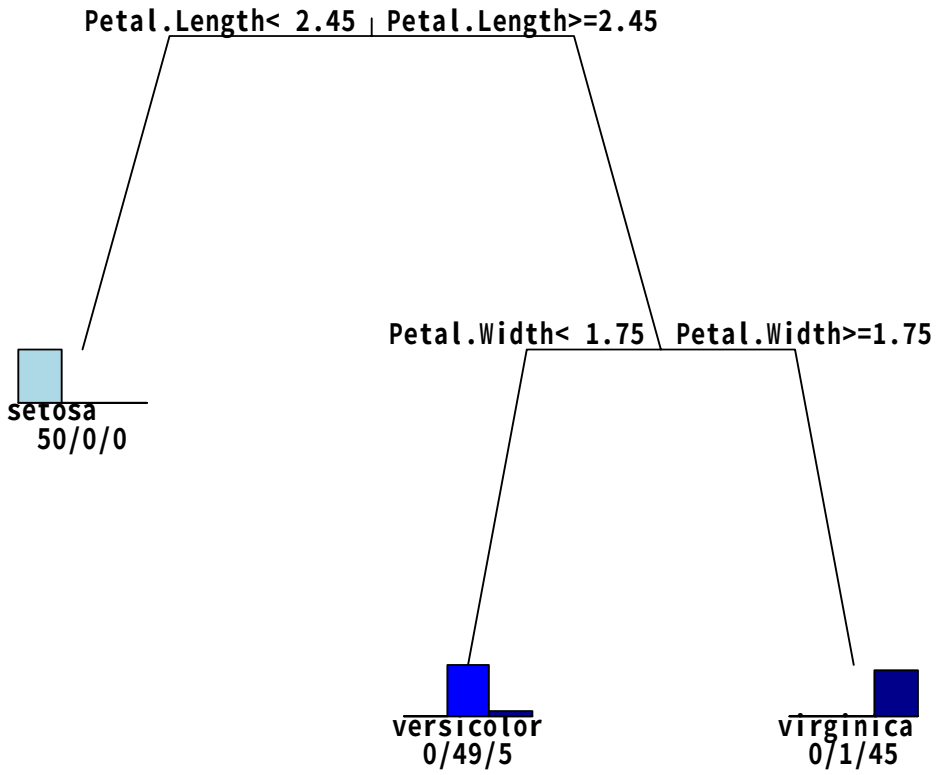


分岐数=2
size=3
が最適！
cp=0.094

```
x <- iris
library(rpart)
result <- rpart(Species ~ ., data=x)
plot.new(); par(xpd=T)
plot(result, uniform=T, branch=0.7,
     margin=0.05)
text(result, use.n=T, all.leaves=F)
```

```
summary(result)
plotcp(result)  # 最適な葉の数を調べる
result2 <- prune(result, cp=0.094)
plot(result2, uniform=T, branch=0.7,
     margin=0.05)
text(result2, use.n=T, all.leaves=F)
summary(result2)
```

データ「iris」でお試し（分岐×2）

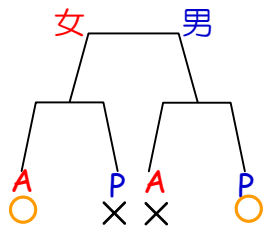



```
x <- iris
library(rpart)
result <- rpart(Species ~ ., data=x)
plot.new(); par(xpd=T)
plot(result, uniform=T, branch=0.7,
     margin=0.05)
text(result, use.n=T, all.leaves=F)
```

```
summary(result)
plotcp(result)
result2 <- prune(result, cp=0.094)
plot(result2, uniform=T, branch=0.7,
     margin=0.05)
text(result2, use.n=T, all.leaves=F)
summary(result2)   # 木を描きなおす
```

【余談】臨床試験に適用出来る？

- 層別解析しているのと同じ
- 外れ値に対しては頑健
- 例数が少ないと不安定（n=1000 以上が望ましい？？）
- 注目している「群」がうまく出てくれたらいいけど。。。
  ⇒「2分木」なので多群の試験に適用するとさらに困難となる
- 作成した「木」の再現性があると言えない。。。
- 「交互作用を見つけるのに適した木」が出来にくい →



- 例えば「男女」で交互作用があるかどうかを検討する場合
  - 体重の変化量の平均値：-5kg（女性・群併合），-5kg（男性・群併合）
  - 女性の体重の変化量の平均値：-10kg（Placebo群），±0kg（Active群）
  - 男性の体重の変化量の平均値：±0kg（Placebo群），-10kg（Active群）
  - 「女性に絞って群別の平均値」「男性に絞って群別の平均値」を算出してはじめて交互作用が浮き出る
    ⇒ CART の性質上、↑のような分岐は出てくれない？

【余談】 CART 以外の分析・分類・予測手法

- 回帰分析（共分散分析・ロジスティック回帰）
  お気楽に，ざっくりと分析する場合・・・
- ニューラル・ネットワーク
  非線形的なデータ構造にも対応，分類ルール不明
- サポートベクターマシン
  CART のような直線的な分類だけではなく，曲線的な分類が出来る
  「判別」が目的なので，説明変数が目的変数に対してどのような影響を与えているかは不明
- 集団学習（バギング，ランダムフォレストなど）
  - バギング：多数の CART を生成して多数決（分類の場合）or 平均（回帰の場合）
  - ランダムフォレスト：与えられたデータからいくつかの変数をランダム抽出し（ブートストラップサンプル）CART を作成
    ⇒ 多数回くり返して多数決（分類の場合）or 平均（回帰の場合）
- アソシエーション・ルール（相関分析）：ビールと紙おむつ

【演習】

1. 変数 x にデータ「iris」を代入してください
2. 1. で作成したデータ x の中身を表示してください
3. 1. で作成したデータ x の中の変数 Sepal.Length について
   - 要約統計量を算出してください
   - ヒストグラムを描いてください
   - アヤメの種類（Species）ごとにヒストグラムを描いてください
   - 変数 Sepal.Length の長さが 5 かどうかの 1 標本 t 検定を実行してください ⇒ 関数 t.test() を使ってください

【演習】

4. 1. で作成したデータ x の中の変数 Petal.Width と Petal.Length について
   - 相関係数を算出してください（余裕のある方は Spearman でも）
   - 散布図を描いてください
   - アヤメの種類（Species）ごとに散布図を描いてください
5. データ x の中の変数 Species を目的変数，それ以外の変数を説明変数として CART を作成してください

【参考】検定関数一覧

関数	実行する検定手法
binom.test()	二項検定
chisq.test()	Pearson の χ2検定
cor.test()	相関係数が0かどうかの検定
fisher.test()	Fisher の正確検定
friedman.test()	Friedman の順位和検定
kruskal.test()	Kruskal-Wallis の順位和検定
ks.test()	Kolmogorov-Smirnov 検定
mantelhaen.test()	Cochran-Mantel-Haenszelの χ2検定
mcnemar.test()	McNemar 検定
oneway.test()	一元配置分散分析
prop.test()	比率の同一性検定
prop.trend.test()	比率の傾向性検定
shapiro.test()	Shapiro-Wilk 検定
t.test()	t 検定（例：t.test(x$Sepal.Length, mu=5)）
var.test()	分散の同一性検定（F 検定）
wilcox.test()	Wilcoxon の順位和検定

参考文献

- 「よくわかる多変量解析の基本と仕組み」
  山口和範, 高橋淳一, 竹内光悦（秀和システム, 2004）

- 「CART による応用 2 進木解析法」
  大滝 厚, 堀江 宥治, Dan Steinberg（日科技連出版社, 1998）

- 「S-PLUS による統計解析」
  W.N. Venables, B.D. Ripley（シュプリンガー, 2001）

- 「An Introduction to Recursive Partitioning Using the RPART Routines」
  Terry M.. Therneau, Elizabeth J.. Atkinson
  （Mayo Foundation, 1997）

★演習（45枚目のスライド分）の回答例

```
> ( x <- iris )                                  # データ iris の読み込み
> summary(x$Sepal.Length)                        # 要約統計量を算出する
> hist(x$Sepal.Length)                           # ヒストグラムを作成する

> library(lattice)                               # おまじない
> histogram(~ Sepal.Length | Species, data=x)    # 層別ヒストグラム
> t.test(x$Sepal.Length, mu=5)                   # 1 標本 t 検定
> cor(x$Petal.Width, x$Petal.Length)             # 相関係数
> cor(x$Petal.Width, x$Petal.Length,
+     method="spearman")                         # Spearman の相関係数
> xyplot(x$Petal.Width ~ x$Petal.Length, ps=20) # 散布図
> xyplot(x$Petal.Width ~ x$Petal.Length,         # 層別散布図
+        col=as.integer(x$Species),
+        pch=as.integer(x$Species), ps=20)
```

★演習（45枚目のスライド分）の回答例

```
> library(rpart)                                    # library(mvpart)
> result <- rpart(Species ~ ., data=x)
> plot.new()                                        # CART
> par(xpd=T)
> plot(result, uniform=T, branch=0.7, margin=0.05)
> text(result, use.n=T, all.leaves=F)
> summary(result)                                   # 分類ルールを見る

> plotcp(result)                                    # 最適な葉の数を調べる
> result2 <- prune(result, cp=0.066)
> plot(result2, uniform=T, branch=0.7, margin=0.05)
> text(result2, use.n=T, all.leaves=F)              # 木を描きなおす
> summary(result2)                                  # 分類ルールを見る
```

この講義の後に読んでいただきたい本

- 本講義の復習
  - R で学ぶデータマイニング I，II（熊谷 悦生，舟尾 暢男；九天社）
  - データ解析環境「R」（舟尾 暢男，高浪 洋平；工学社）
- R と統計を並行して勉強されたい方向け
  - R によるやさしい統計学（山田 剛史 他；オーム社）
  - R による保健医療データ解析演習（中澤 港；ﾋﾟｱｿﾝｴﾃﾞｭｹｰｼｮﾝ）
  - 工学のためのデータサイエンス入門（間瀬 茂 他；数理工学社）
- 本格的に R をはじめる方向け
  - R プログラミングマニュアル（間瀬 茂；数理工学社）
  - R の基礎とプログラミング技法（石田 基広 訳；ｼｭﾌﾟﾘﾝｶﾞｰ・ｼﾞｬﾊﾟﾝ）
- 各論
  - R による医療統計学（岡田 昌史 監訳；丸善）
  - R によるデータサイエンス（金 明哲；森北出版）
  - R と S-PLUS による多変量解析（石田 基広 訳；ｼｭﾌﾟﾘﾝｶﾞｰ・ｼﾞｬﾊﾟﾝ）
- リファレンス
  - RjpWiki（筑波大学・岡田昌史先生 の Wiki）
    http://www.okada.jp.org/RWiki/
  - The R Tips（舟尾 暢男；九天社 ⇒ 書籍とホームページ両方あり）
    http://cse.naro.affrc.go.jp/takezawa/r-tips/r2.html

５時間目にやったこと

- イントロ
  - データ「iris」の紹介
  - 1 つの変数を要約する
  - 2 つの変数の関係を見る
- CART の紹介
  - CART とは？
  - CART の作成方法
  - CART の剪定作業
- データ「iris」でお試し
  - お試し
  - 余談＋演習

終