樹木モデル

1216557c
城　麻莉乃

■樹木モデル(tree-based model)とは

説明変数の値を何らかの基準をもとに分岐させ、判別・予測のモデルを構築する。

回帰問題　→回帰木(regression tree)
分類問題　→分類木(classification tree)
　　決定木(decision tree)

非線形回帰分析・非線形判別分析の1つの方法

■樹木モデルの基礎

アルゴリズム	概要	分岐基準
CHAID	Hartigan（1975） AID(Automatic Interaction Detection)の発展	カイ2乗統計量 F統計量
C4.5/C5.0/ See5	J.Ross Quinlan（1986） 機械学習のアプローチで発表した ID3(Iterative Dichotomiser 3)の改良版 ２進木になるとは限らない	利得比(gain ratio)
CART	R.A.Olshen, C.J.Stone,J.H.Friedman(1970s~1980） ・説明変数を２進木に分岐させる ・データと対話しながら樹木の剪定を行う	ジニ係数 ジニ多様性指標 ＋ 情報利得 (information gain)

■ケーススタディ～分類木～

ｉ.木の生成

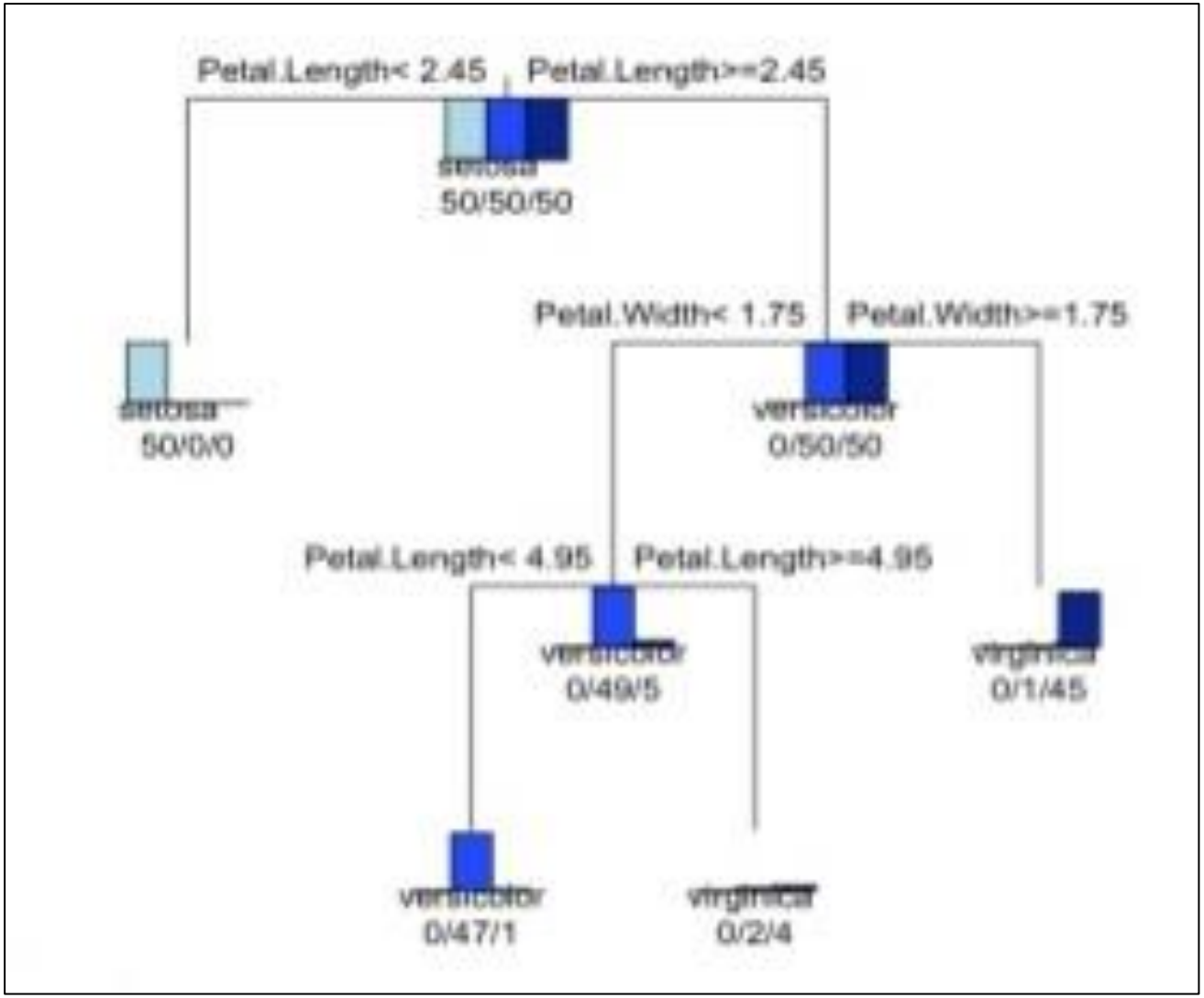
同じ結果を出すため

```
> library(mvpart)
> set.seed(20)
> iris.rp<-rpart(Species~.,data=iris)
> print(iris.rp,digit=1)
n= 150

node), split, n, loss, yval, (yprob)
      * denotes terminal node

 1) root 150 100 setosa (0.33 0.33 0.33)
   2) Petal.Length< 2 50   0 setosa (1.00 0.00 0.00) *
   3) Petal.Length>=2 100  50 versicolor (0.00 0.50 0.50)
     6) Petal.Width< 2 54   5 versicolor (0.00 0.91 0.09)
      12) Petal.Length< 5 48   1 versicolor (0.00 0.98 0.02)
      13) Petal.Length>=5 6   2 virginica (0.00 0.33 0.67) *
     7) Petal.Width>=2 46   1 virginica (0.00 0.02 0.98) *
> plot(iris.rp,uniform=T)
> text(iris.rp,use.n=T,all=T)
```

Digitは返す値の小数点以下の桁数を指定する引数

パッケージmvpartがインストールされていないと図のスタイルが若干異なる

関数rpartによるirisの分類木



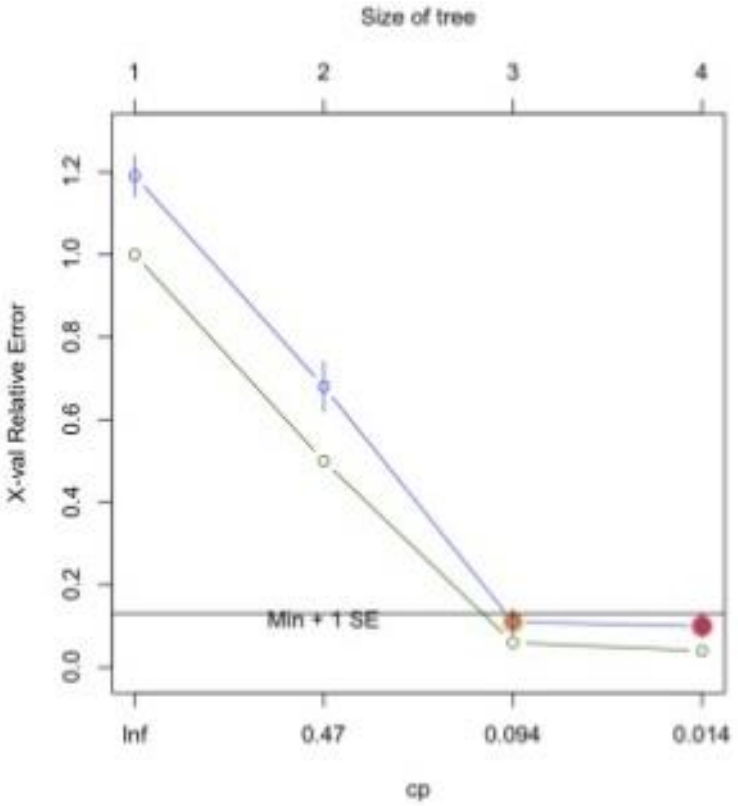
Plotとtextを用いてグラフを作成

ii .木の剪定

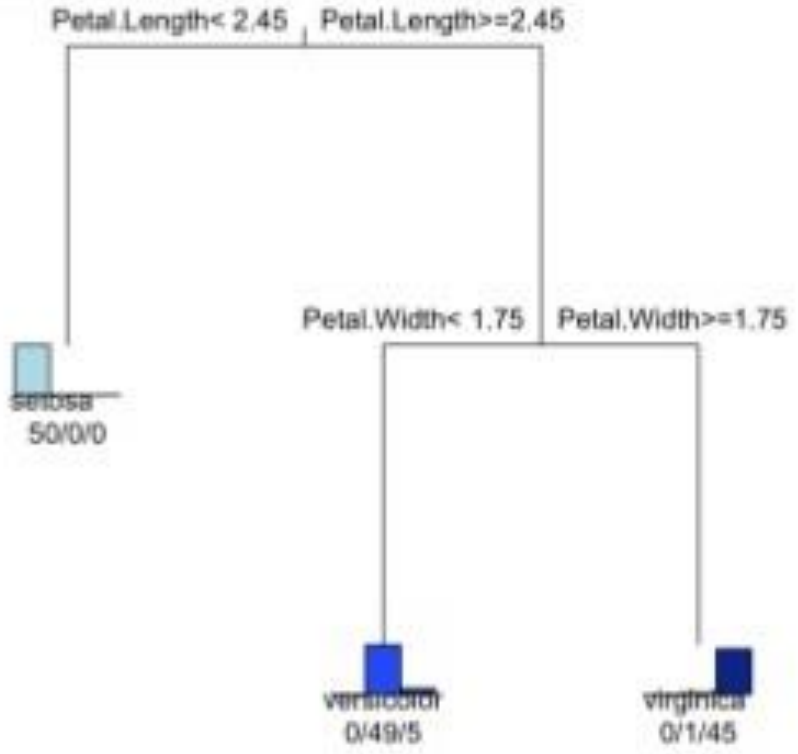
```
> printcp(iris.rp)

Classification tree:
rpart(formula = Species ~ ., data = iris)

Variables actually used in tree construction:
[1] Petal.Length Petal.Width

Root node error: 100/150 = 0.66667

n= 150

    CP nsplit rel error xerror     xstd
1 0.50      0      1.00   1.19 0.049592
2 0.44      1      0.50   0.68 0.060970
3 0.02      2      0.06   0.11 0.031927
4 0.01      3      0.04   0.10 0.030551
```

Min＋1 SE＝
0.1+0.030551=0.130551

```
> iris.rp1<-prune(iris.rp,cp=0.02)
> plot(iris.rp1,uniform=T,margin=0.05)
> text(iris.rp1,use.n=T)
> plotcp(iris.rp)
```



cp=0.02で剪定したirisの分類木



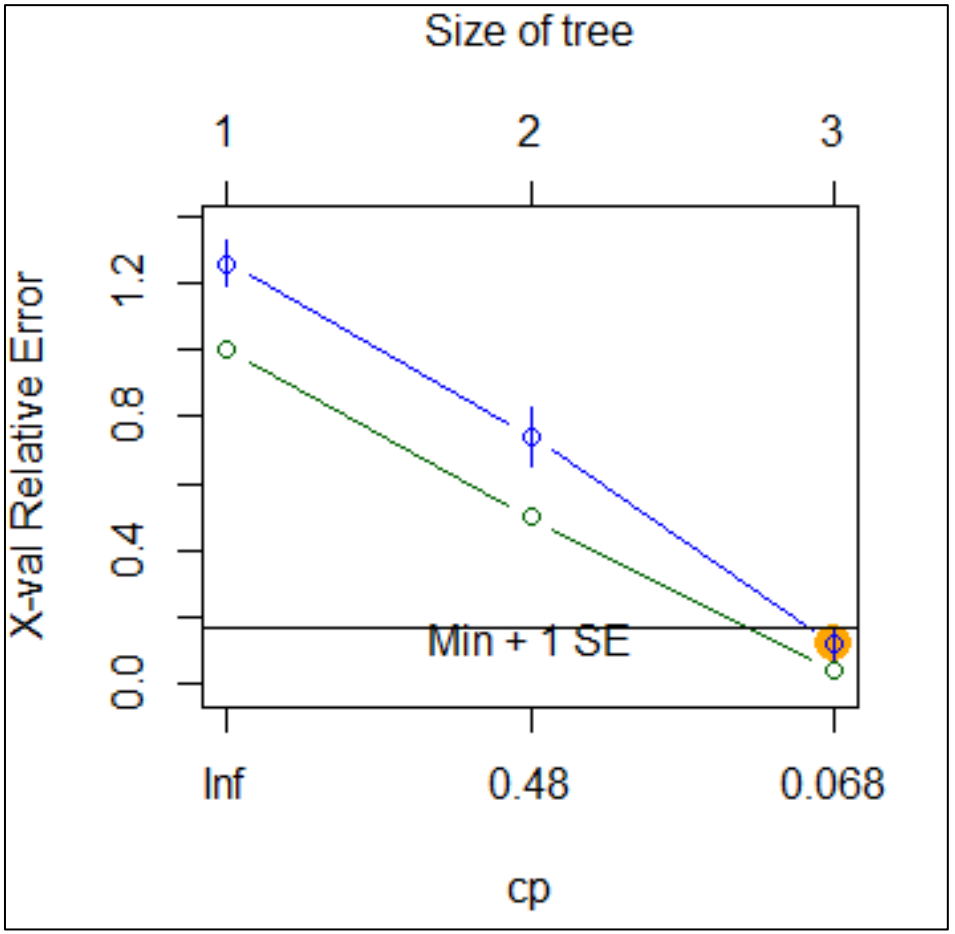
Iris.rpのpLotcpのプロット

iii .判別

奇数行と偶数行に分ける

```
> even.n<-2*(1:75)-1
> iris.train<-iris[even.n,]
> iris.test<-iris[-even.n,]
> set.seed(20)
> iris.rp2<-rpart(Species~.,iris.train)
> plotcp(iris.rp2)
```

モデルiris.rpを生成する際には同じ結果を出すため、乱数の種set.seedを使う



Irisの奇数行のprintcpプロット

作成したモデルを用いた予測・
判別はpredict

```
> iris.rp3<-predict(iris.rp2,iris.test[,-5],type="class")
> table(iris.test[,5],iris.rp3)
            iris.rp3
             setosa versicolor virginica
  setosa         25          0         0
  versicolor      0         24         1
  virginica       0          3        22
```

iv.コントロール

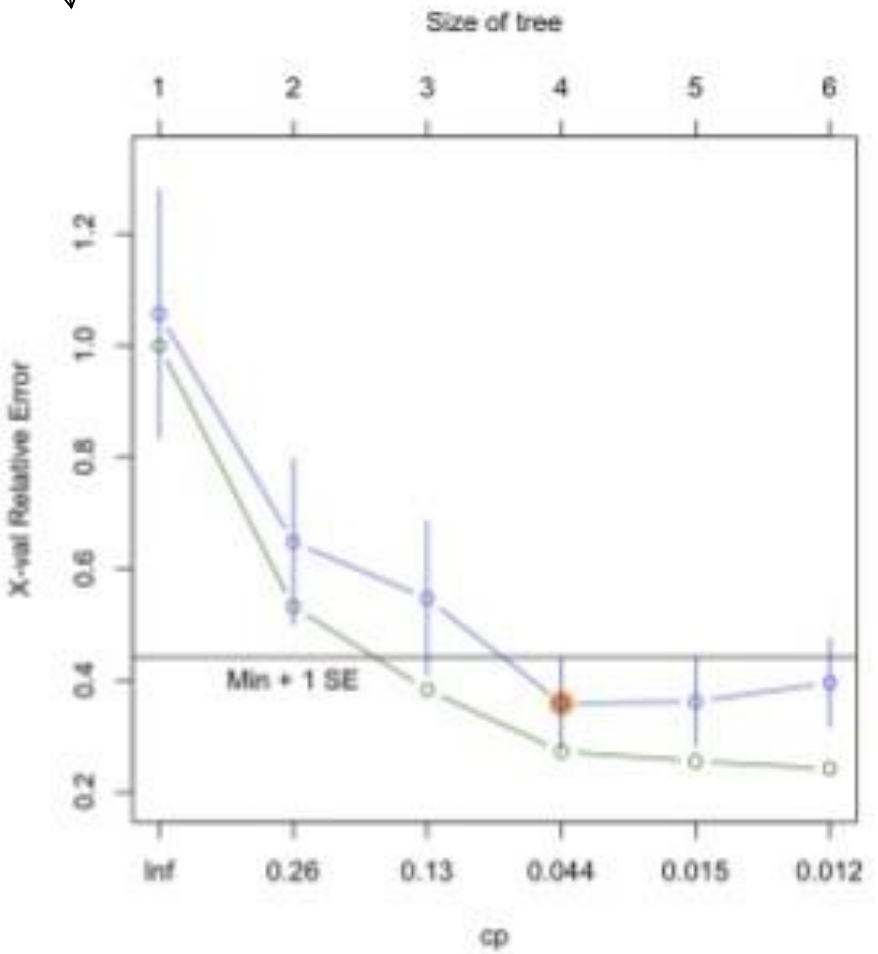
- Help(rpart.control)にて確認可能

■ケーススタディ～回帰木～

rpartによるデータcarsの回帰木の生成
（自動車の速度とブレーキから停止までにかかる距離のデータ）

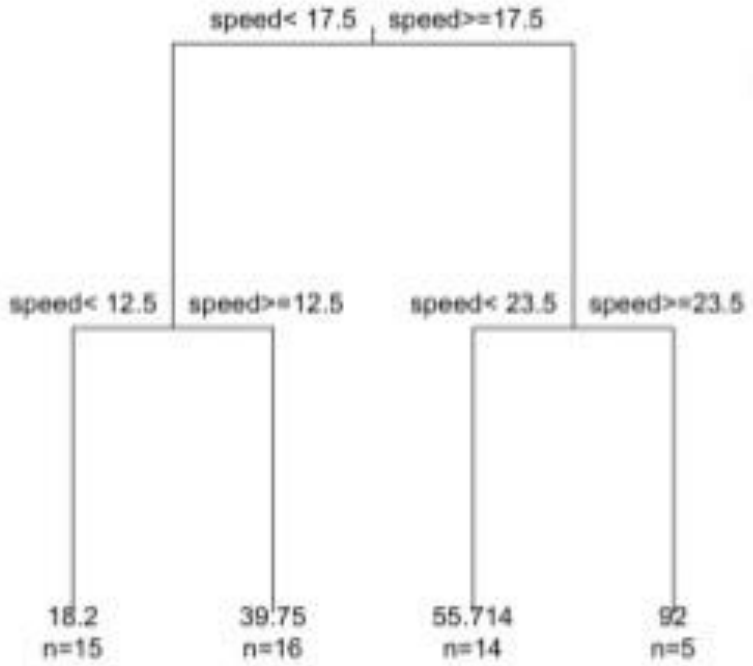
```
> (car.rp<-rpart(dist~speed,data=cars))
n= 50

node), split, n, deviance, yval
      * denotes terminal node

 1) root 50 32538.9800 42.98000
   2) speed< 17.5 31  8306.7740 29.32258
     4) speed< 12.5 15  1176.4000 18.20000
       8) speed< 9.5 6   277.3333 10.66667 *
       9) speed>=9.5 9   331.5556 23.22222 *
     5) speed>=12.5 16  3535.0000 39.75000 *
   3) speed>=17.5 19  9015.6840 65.26316
     6) speed< 23.5 14  2846.8570 55.71429
      12) speed>=18.5 10  1323.6000 52.20000 *
      13) speed< 18.5 4  1091.0000 64.50000 *
     7) speed>=23.5 5  1318.0000 92.00000 *
```

```
> plotcp(car.rp)
> car.rp1<-prune(car.rp, cp=0.044)
> plot(car.rp1,uniform=T,margin=0.05)
> text(car.rp1,use.n=T)
```



データcarsのplotcpプロット

４つの葉に剪定



データcarsの回帰木

■ケーススタディ～多変量回帰木～

- 多変量回帰(multivariate regression)

  -目的変数が複数である回帰分析

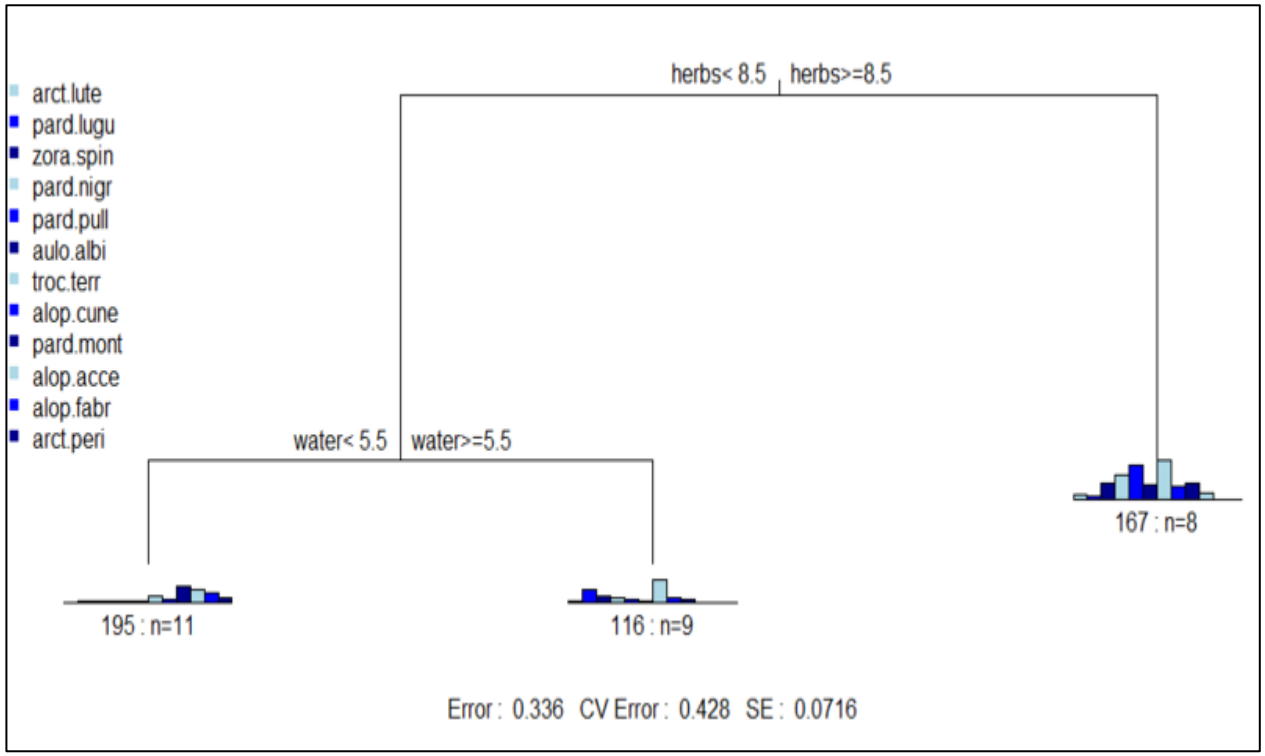
データspider

28行18列のデータフレーム

左12列　異なる蜘蛛の種類の分布　　→目的変数

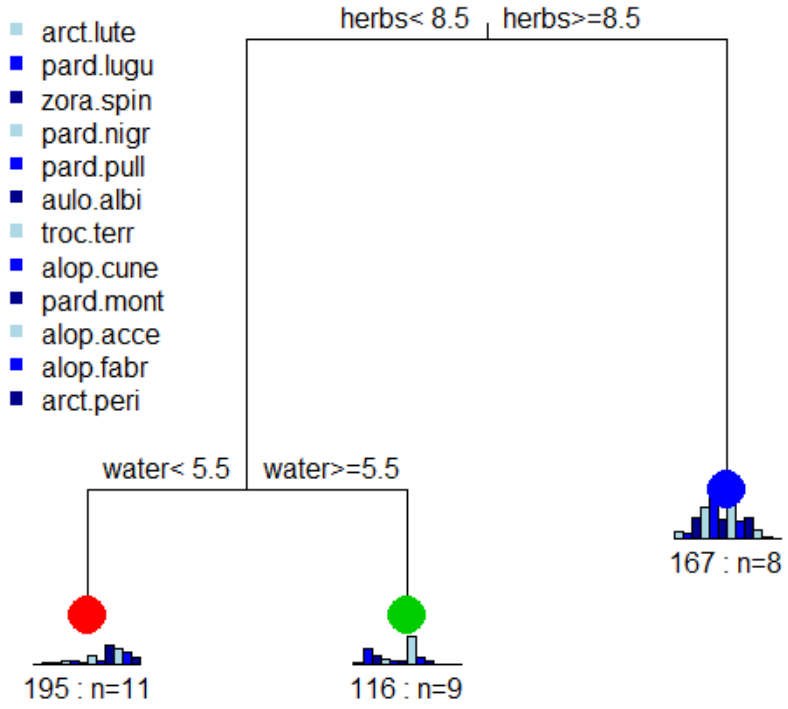
右6列　　雲狩りの環境に関するデータ　→説明変数

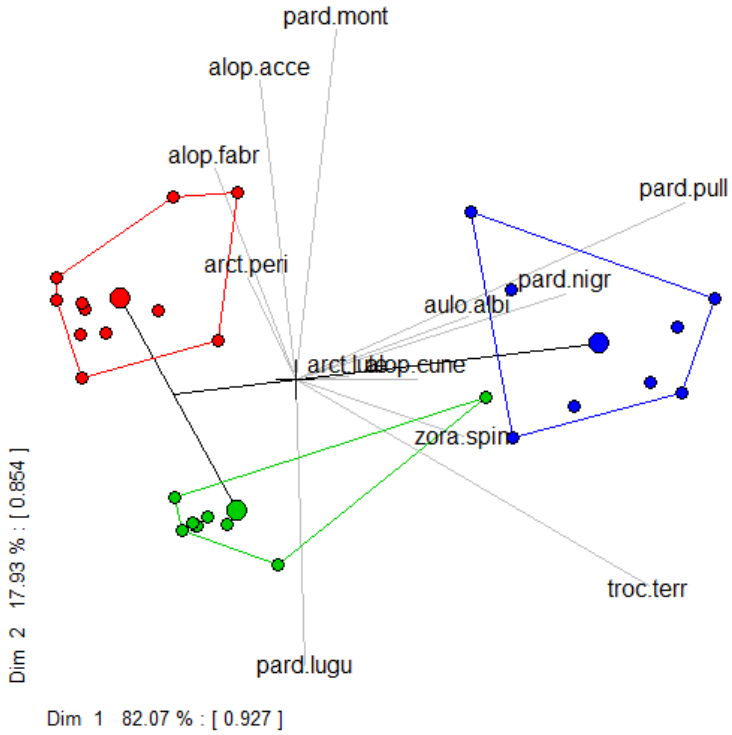
```
>spider[1,13:18]
  water sand moss reft twigs herbs
1     9    0     1    1     9     5
> spider.mv<-mvpart(as.matrix(spider[,1:12])~water+sand+moss+reft+twigs+herbs,data=spider)
```



```
> mvpart(as.matrix(spider[,1:12])~water+sand+moss+reft+twigs+herbs,spider,pca=T)
```

主成分分析





データspiderの多変量回帰木と主成分の散布図との結合

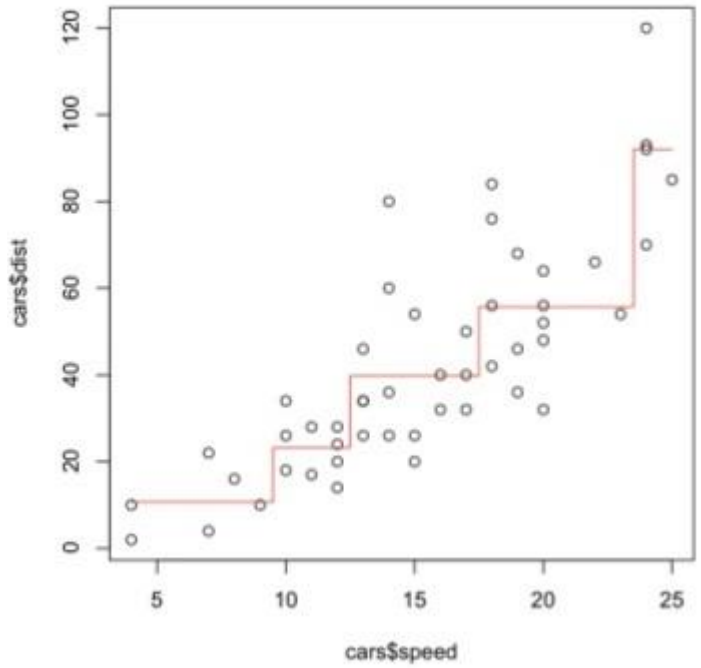
■パッケージ

- tree
  - 木を生成する関数tree
  - Parition.treeは折れ線回帰図や分割分類図が作成可能

- Rweka
  - データマイニングパッケージWekaのRバージョン
  - C4.5のアルゴリズムによる関数J４８がある

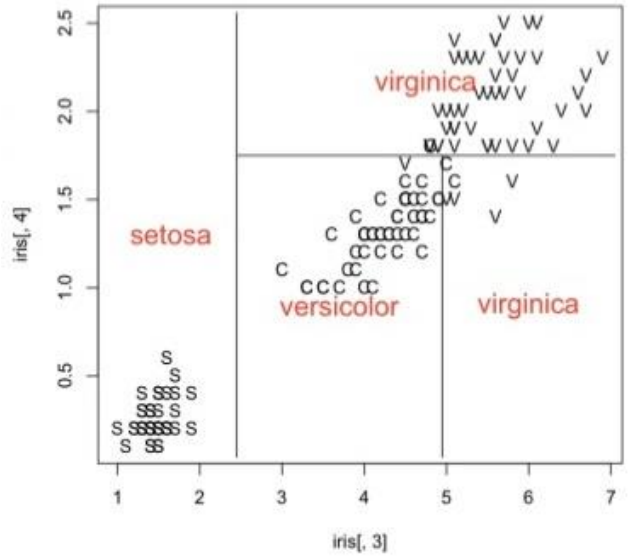
i.tree

```
> library(tree)
> cars.tr<-tree(dist~speed,data=cars)
> plot(cars$speed,cars$dist)
> partition.tree(cars.tr,add=T,col=2)
```



折れ線回帰

```
> iris.tr<-tree(Species~.,data=iris)
> iris.tr1<-snip.tree(iris.tr,nodes=c(12,7))
> iris.label<-c("S","C","V")[iris[,5]]
> plot(iris[,3],iris[,4],type="n")
> text(iris[,3],iris[,4],labels=iris.label)
> partition.tree(iris.tr1,add=T,col=2,cex=1.5)
```



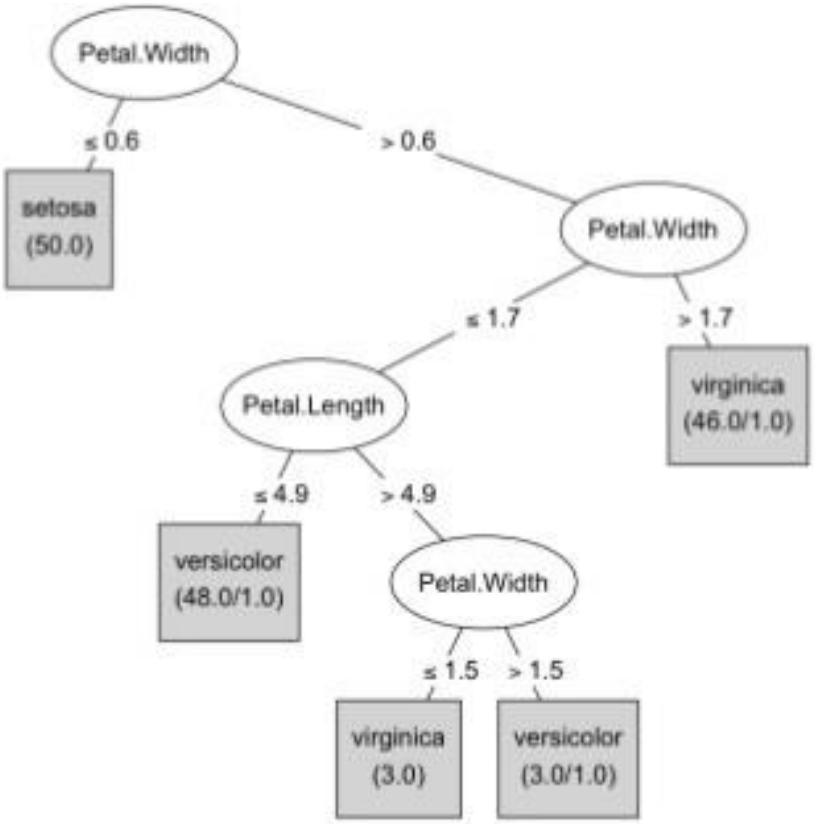
散布図上の分類木の結果

ii .Rweka

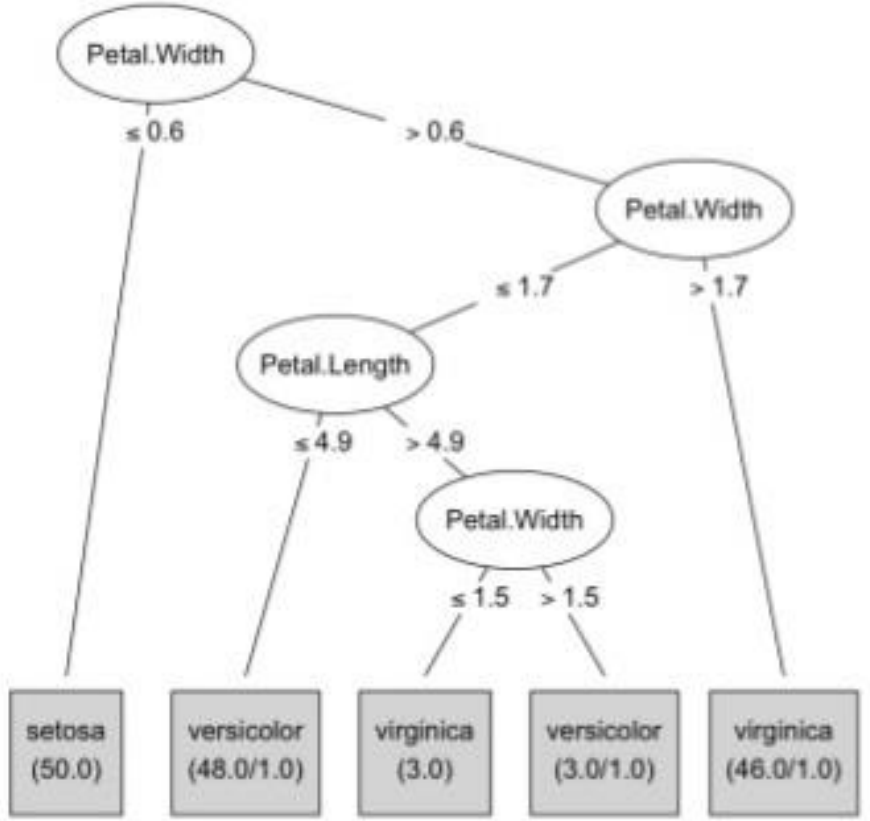
```
> library(RWeka)
> iris.j48 <- J48(Species~., data=iris)
> iris.j48
J48 pruned tree
------------------

Petal.Width <= 0.6: setosa (50.0)
Petal.Width > 0.6
|   Petal.Width <= 1.7
|   |   Petal.Length <= 4.9: versicolor (48.0/1.0)
|   |   Petal.Length > 4.9
|   |   |   Petal.Width <= 1.5: virginica (3.0)
|   |   |   Petal.Width > 1.5: versicolor (3.0/1.0)
|   Petal.Width > 1.7: virginica (46.0/1.0)

Number of Leaves  :     5

Size of the tree :  9
```

> install.packages("party" ,dependencies =TRUE); library(party)

>plot(iris.j48)



>plot(iris.j48,type = c("extended"))



- パッケージpartyの作図関数

2分木を制御

```
> cars.ctr<-ctree(dist~.,data=cars)
> plot(cars.ctr)
> t.style<-node_hist(cars.ctr,ymax=0.06,xscale=c(0,150),col="red",fill=hsv(0.6,0.5,1))
> plot(cars.ctr, terminal_panel=t.style)
```