

カーネル法とサポートベクターマシン

1206625c 三輪健太郎

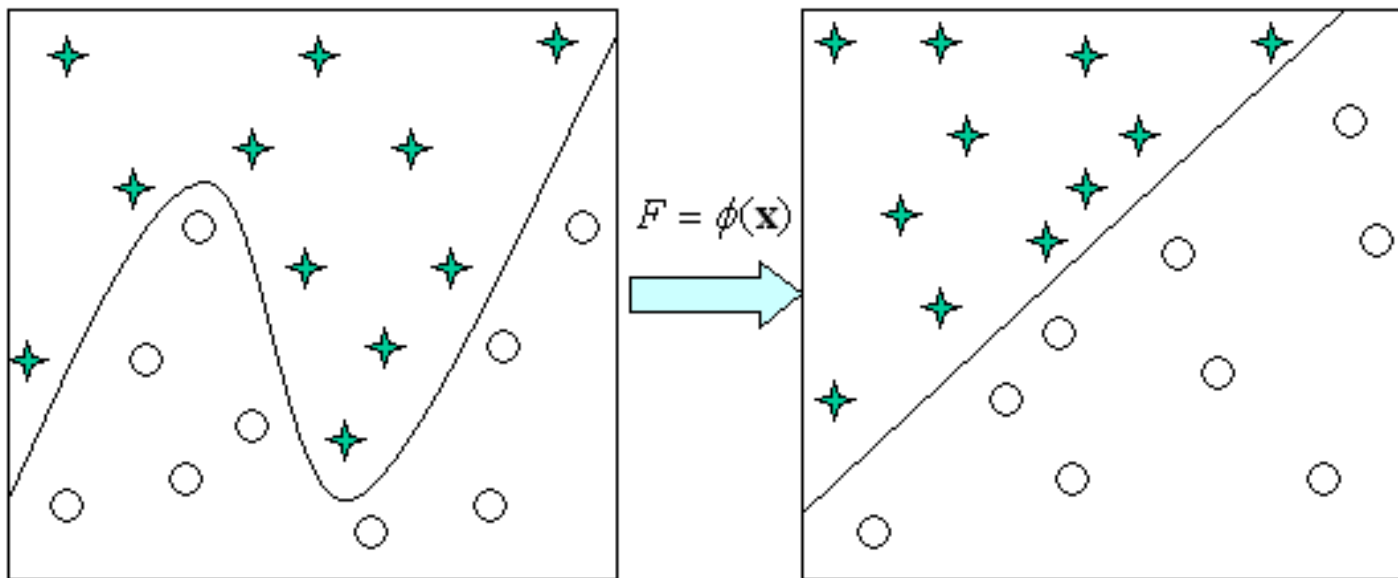
カーネル法とは

- ◉ 非線形的なデータ構造を線形構造に変換すること
- ◉ データを異なる空間に射影することで、非線形構造を線形構造に非線形に変換することが可能
- ◉ カーネル関数を借りてデータを表現しなおす方法で、カーネル関数では表現し直されたデータの連続性と線形性に焦点を当てている。

変換のイメージ図

$\Phi(x)$ を用いて個体の特徴・属性ベクトルについて変換

変換されたF...カーネル法では特徴空間と呼ぶ

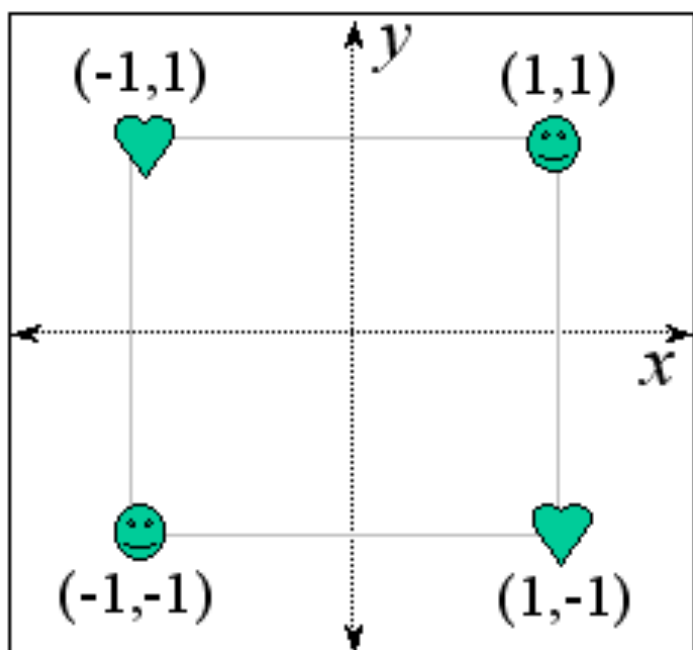


△非線形構造のグラフ

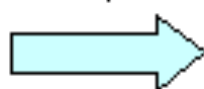
△線形構造のグラフ

データ写像の図

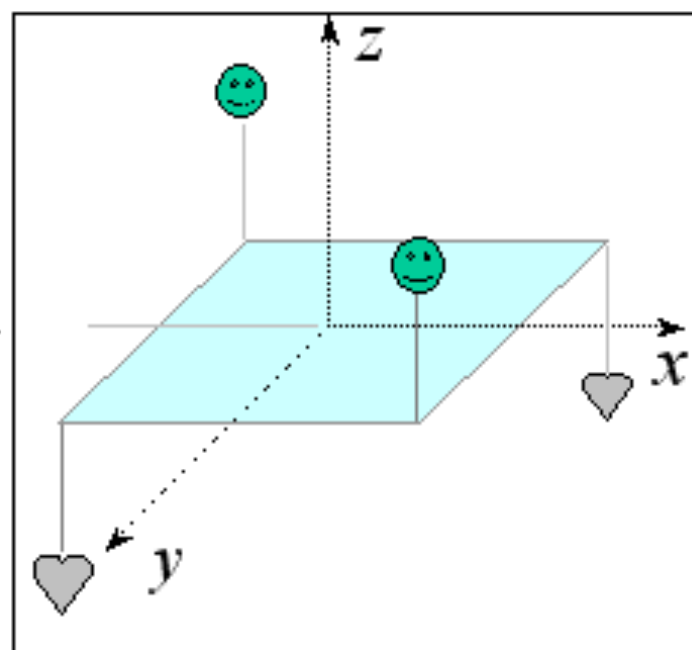
(a)



$$F = \phi(\mathbf{x})$$



(b)



カーネル主成分分析

- 非線形主成分分析とも呼ばれる
- 大まかな流れは、
 1. カーネル関数 $K(x, z)$ を決める
 2. データから写像行列 $K_{n \times n}$ を求める
 3. $K_{n \times n}$ の固有値と固有ベクトルを求める
 4. 固有値と固有ベクトルを正規化する

パッケージと関数

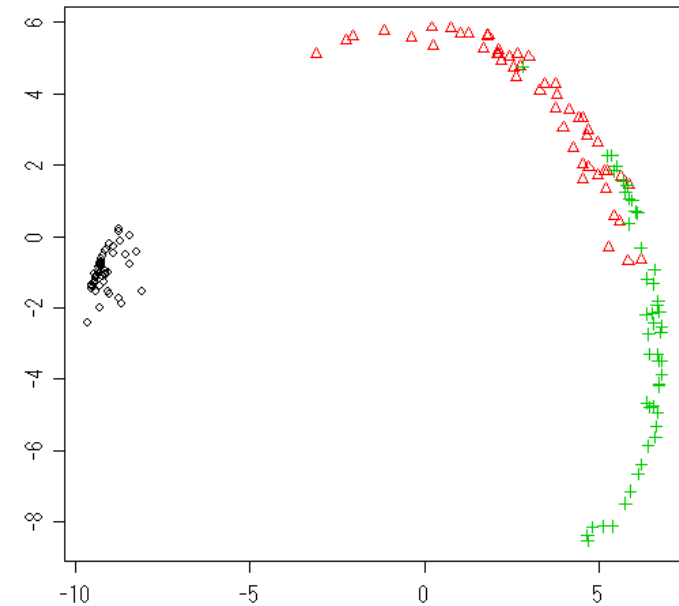
- パッケージkernlabにはカーネル主成分分析の関数kpcaがある。

```
kpca(x, kernel="rbfdot", features=0, kpar=list(sigma=0, 1), ...)
```

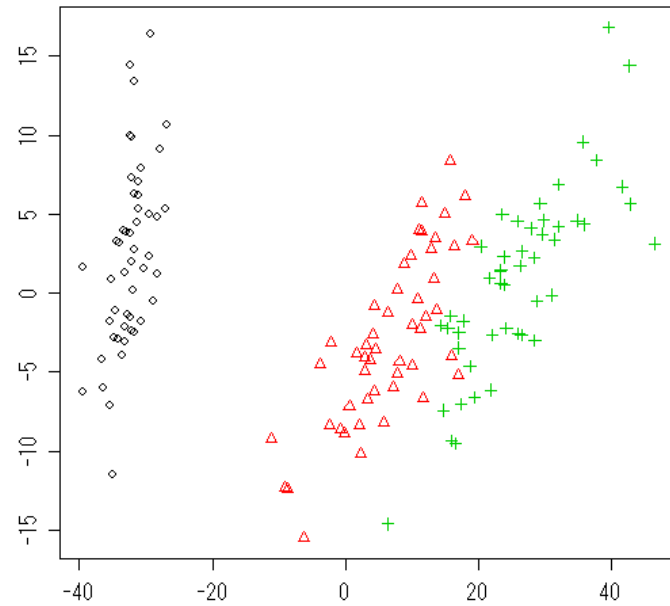
カーネル主成分分析のケーススタディ

```
> library(kernlab)
> x<-as.matrix(iris[,1:4])
> iris.kpc1<-kpca(x, kernel="rbfdot", features=2, kpar=list(sigma=0.1))
> plot(pcv(iris.kpc1), col=as.integer(iris[,5]))
```

ケーススタディのグラフ



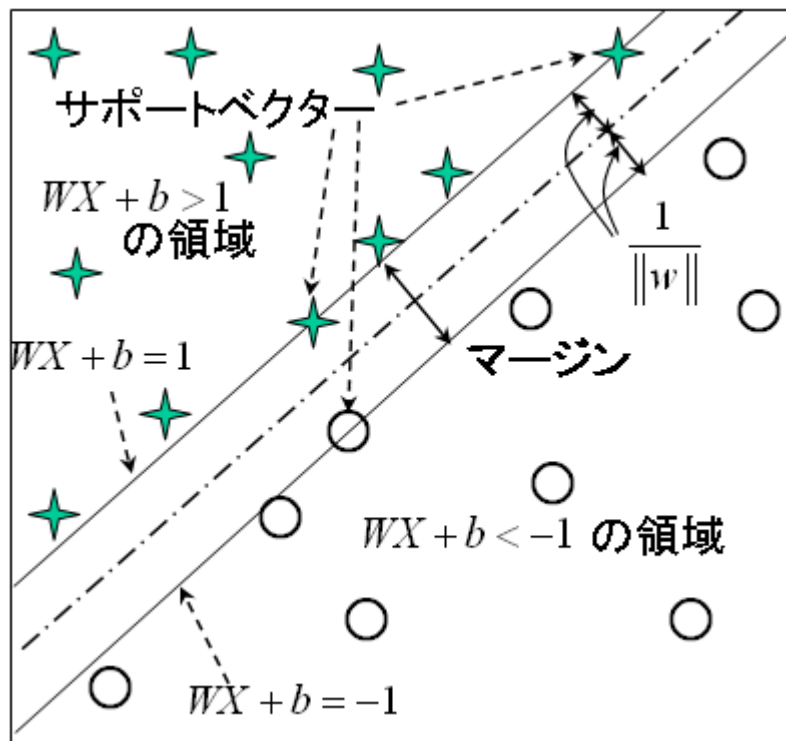
kernel="rbfdot", list(sigma=0.5)
の場合



Kernel="polydot", list(degree=5)
の場合

サポートベクターマシン

- サポートベクターマシン (SVM)とは
分類と回帰問題を主としたデータ解析方法
線形分離が可能な高次元の仮説空間



パッケージと関数

```
Ksvm(formula,data,kernel="rbfdot",kpar=list(sigma=0.1),type=,  
cross=0,)
```

判別分析

```
> library(kernlab)
> data(spam);dim(spam)
[1] 4601  58
> table(spam[,58])
nonspam  spam
  2788   1813
```

```
> set.seed(50)
> tr.num<-sample(4601,2500)
> spam.train<-spam[tr.num,]
> spam.test<-spam[-tr.num,]
```

判別分析のケーススタディ (1)

```
> spam.svm <- ksvm(type~.,data=spam.train,  
kernel="rbfdot",kpar=list(sigma=0.01))  
> spam.pre <- predict(spam.svm, spam.test[,-58])  
> (spam.tab<-table(spam.test[,58], spam.pre))
```

	spam.pre	
	nonspam	spam
nonspam	1226	56
spam	96	723

```
> 1-sum(diag(spam.tab))/sum(spam.tab)  
[1] 0.0723465
```

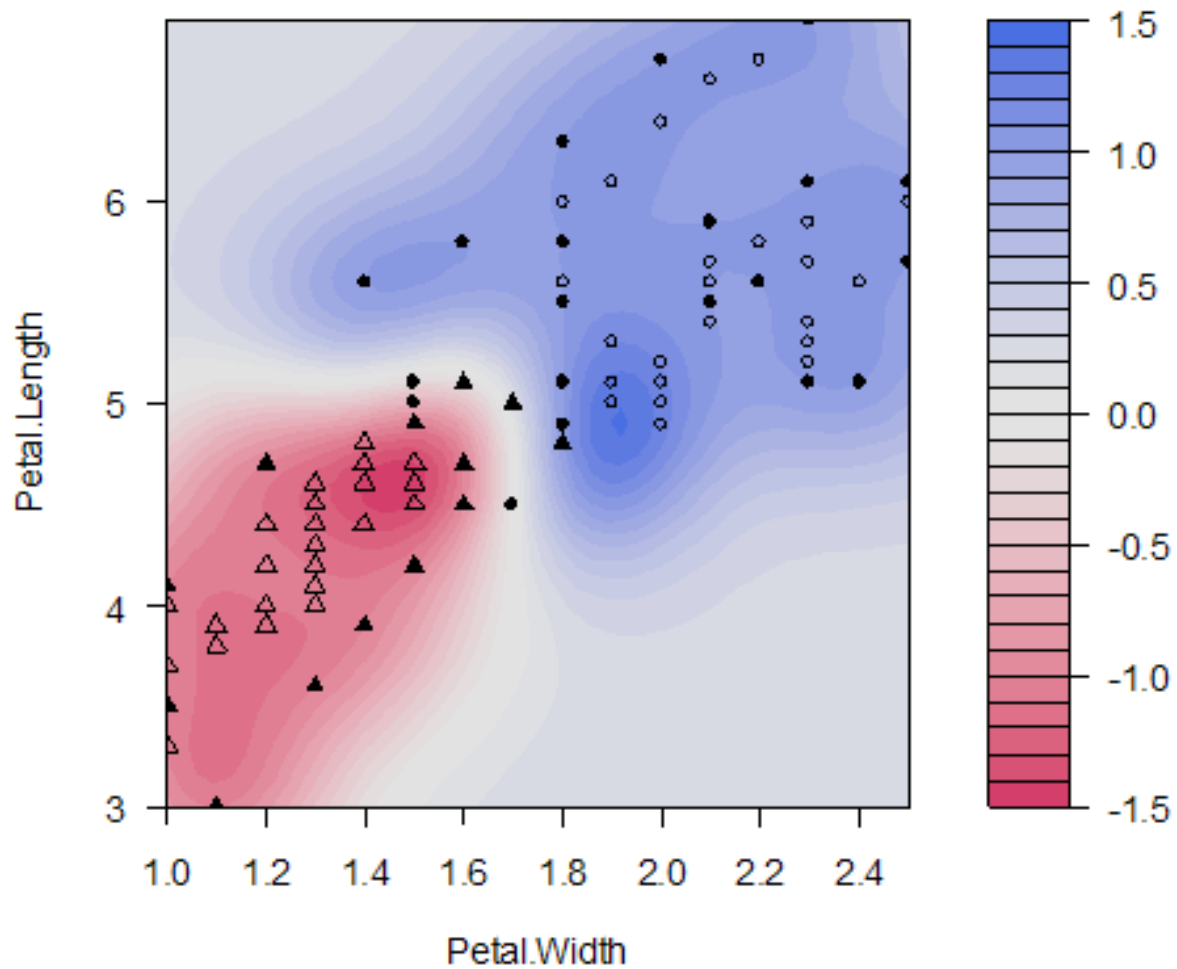
```
> (train.cro <-  
ksvm(type~.,data=spam.train,kernel="rbfdot",kpar=list(sigma=0.05),  
Support Vector Machine object of class "ksvm"  
SV type: C-svc (classification)  
parameter : cost C = 5  
Gaussian Radial Basis kernel function.  
Hyperparameter : sigma = 0.05  
Number of Support Vectors : 967  
Training error : 0.0164  
Cross validation error : 0.082  
C=5,cross=10))
```

ケーススタディ (2)

```
> set.seed(10)
> y<-as.matrix(iris[51:150,5])
> iris1<-data.frame(iris[51:150,3:4],y)
> ir.ksvm<-ksvm(y~,data=iris1)
> plot(ir.ksvm,data=iris1[,1:2])
> table(iris1$y,predict(ir.ksvm,iris1[,1:2]))
```

	versicolor	virginica
versicolor	48	2
virginica	3	47

SVM classification plot



回帰分析のケーススタディ

```
> x1=seq(-10,10,0.1);set.seed(10)
> y1=50*sin(x1)+x1^2+10*rnorm(length(x1),0,1)
```

```
> xy.svm<-ksvm(x1,y1,epsilon=0.01,kpar=list(sigma=16))
> sy.pre<-predict(xy.svm,x1)
> plot(x1,y1,type="l")
> lines(x1,sy.pre,col="red",lty=2)
> legend(locator(1),c("実測値","予測値"),lty=c(1,2),col=c(1,2))
```

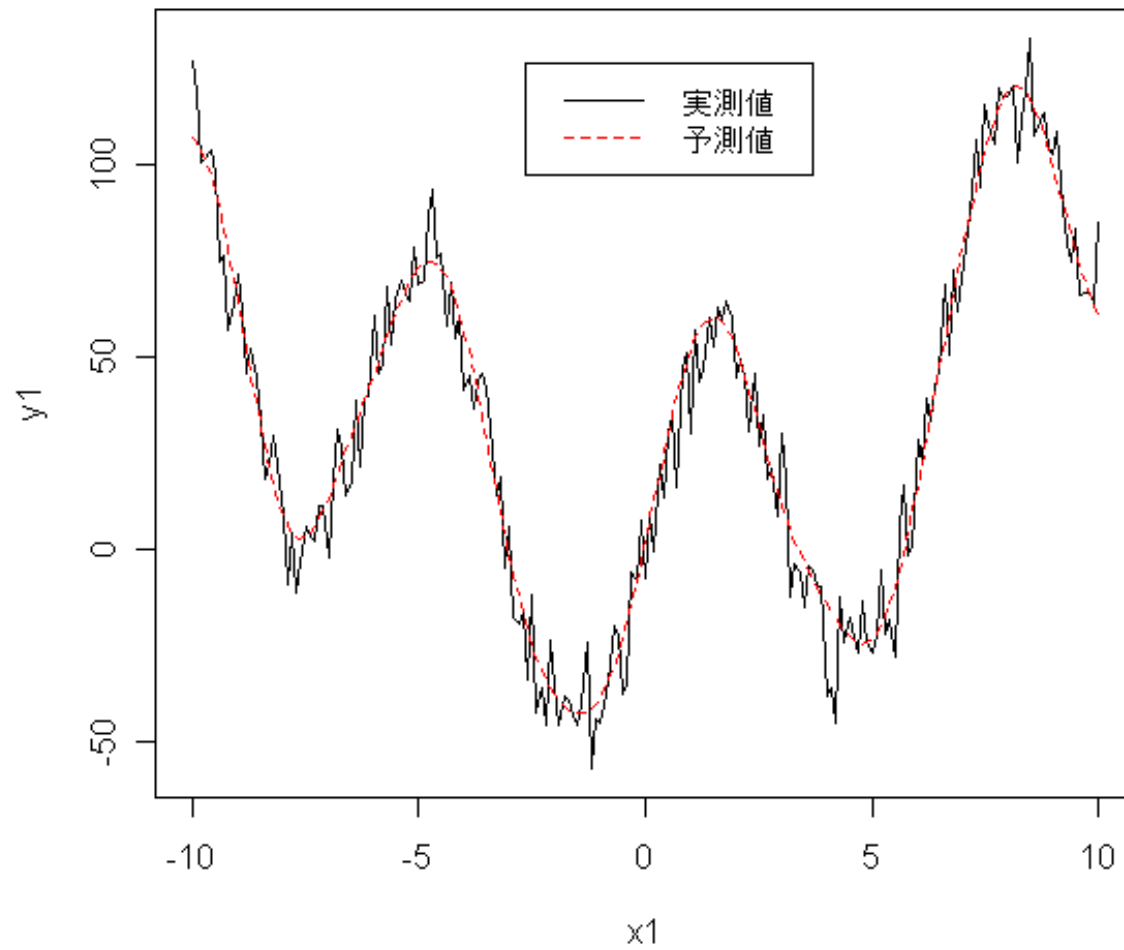


図6 回帰問題における実測値と関数ksvmの予測値