```
library(mlbench)
data(PimaIndiansDiabetes2)
pid<-PimaIndiansDiabetes2
pid<-na.omit(pid)
grp <- as.factor(pid[,9])
x <- pid[,1:8]
set.seed(100)
train <- sample(1:nrow(pid),300)

library(e1071)
resSVM <- svm(x[train,],grp[train],kernel="radial")
predSVM <- predict(resSVM,newdata=x[-train,])
TAB1 <- table(predSVM,pid[-train,9])
mkrSVM <- 1-sum(diag(TAB1))/sum(TAB1)
mkrSVM


tuneSVM <- tune.svm(x[train,],grp[train],gamma=2^(-10:0),
        cost=2^(-4:2),kernel="radial")

tuneSVM
plot(tuneSVM)

resSVM <- svm(x[train,],grp[train],kernel="radial",gamma=2^-9,cost=2^2)
predSVM <- predict(resSVM,newdata=x[-train,])
TAB1 <- table(predSVM,pid[-train,9])
mkrSVM <- 1-sum(diag(TAB1))/sum(TAB1)
mkrSVM

####
# bank data
#http://archive.ics.uci.edu/ml/datasets/Bank+Marketing
d <- read.csv2("bank.csv")

library(e1071)
set.seed(123)
train <- sample(nrow(d), 3000)
resSVM <- svm(y ~ ., data=d, subset=train)
predSVM <- predict(resSVM,newdata=d[-train,])
TAB1 <- table(d$y[-train],predSVM)
#> TAB1
#     predSVM
#        no  yes
#  no  1336    7
#  yes  164   14

mkrSVM <- 1-sum(diag(TAB1))/sum(TAB1)
mkrSVM
#[1] 0.112426
# aber: "yes"-Kunden falsch:
164/(164+14)
#[1] 0.9213483

tuneSVM <- tune.svm(y ~ ., data=d[train,],gamma=2^(-5:-2),
        cost=2^(-1:2),kernel="radial")
tuneSVM
plot(tuneSVM)

resSVM <- svm(y ~ ., data=d, subset=train,kernel="radial",gamma=2^-4,cost=2^2)
```

```
predSVM <- predict(resSVM,newdata=d[-train,])
TAB1 <- table(d$y[-train],predSVM)
#> TAB1
#     predSVM
#        no  yes
#  no  1314   29
#  yes  123   55
mkrSVM <- 1-sum(diag(TAB1))/sum(TAB1)
mkrSVM
# [1] 0.09993425
# aber: "yes"-Kunden falsch:
123/(123+55)
#[1] 0.6910112


# regelmaessige Verteilung von y:
d1 <- read.csv2("bank-full.csv")
table(d1$y)
#   no   yes
#39922  5289

# sample from "no":
sel <- c(1:nrow(d1))
selno <- sample(sel[d1$y=="no"],sum(d1$y=="yes"))
d1n <- rbind(d1[d1$y=="yes",],d1[selno,])
table(d1n$y)
#  no  yes
#5289 5289

set.seed(123)
samp <- sample(nrow(d1n),8000)
res1SVM <- svm(y~.,data=d1n,subset=samp,kernel="radial",gamma=2^-4,cost=2^2)
pred1SVM <- predict(res1SVM,newdata=d1n[-samp,])
TAB1n <- table(d1n$y[-samp],pred1SVM)
#> TAB1n
#     pred1SVM
#        no  yes
#  no  1048  242
#  yes  142 1146

mkr1SVM <- 1-sum(diag(TAB1n))/sum(TAB1n)
mkr1SVM
#[1] 0.1489527

# jetzt: "yes"-Kunden falsch:
142/(142+1146)
#[1] 0.1102484

# Random Forest result: #[1] 0.08773292


#####################################################################################
# SVM for regression:

library("UsingR")
data(fat)
attach(fat)
fat$body.fat[fat$body.fat==0]<-NA
fat<-fat[,-cbind(1,3,4,9)]
fat<-fat[-42,]
```

```
fat[,4]<-fat[,4]*2.54
fat <- na.omit(fat)

set.seed(100)
train=sample(1:nrow(fat),150)

tuneSVM <- tune.svm(fat[train,-1],fat[train,1],gamma=2^(-8:0),
        cost=2^(-4:3),kernel="radial")

tuneSVM
plot(tuneSVM)

resSVM <- svm(body.fat~.,data=fat,subset=train,kernel="radial",gamma=2^-5,cost=2^2)
predSVM <- predict(resSVM,newdata=fat[-train,])
RMSEtest <- sqrt(mean((fat$body.fat[-train]-predSVM)^2))
RMSEtest


plot(fat$body.fat[-train],predSVM,xlab="Body fat measured",ylab="Body fat predicted")
abline(c(0,1))
```