

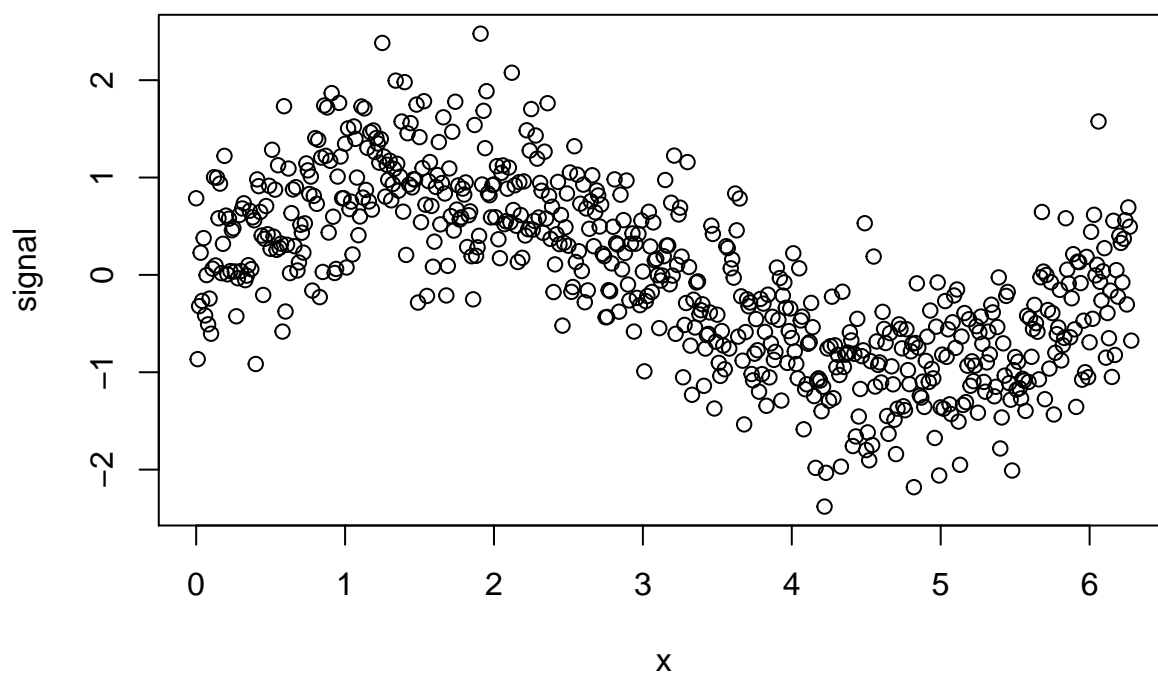
# SVM\_sin\_fun

$Y_e$

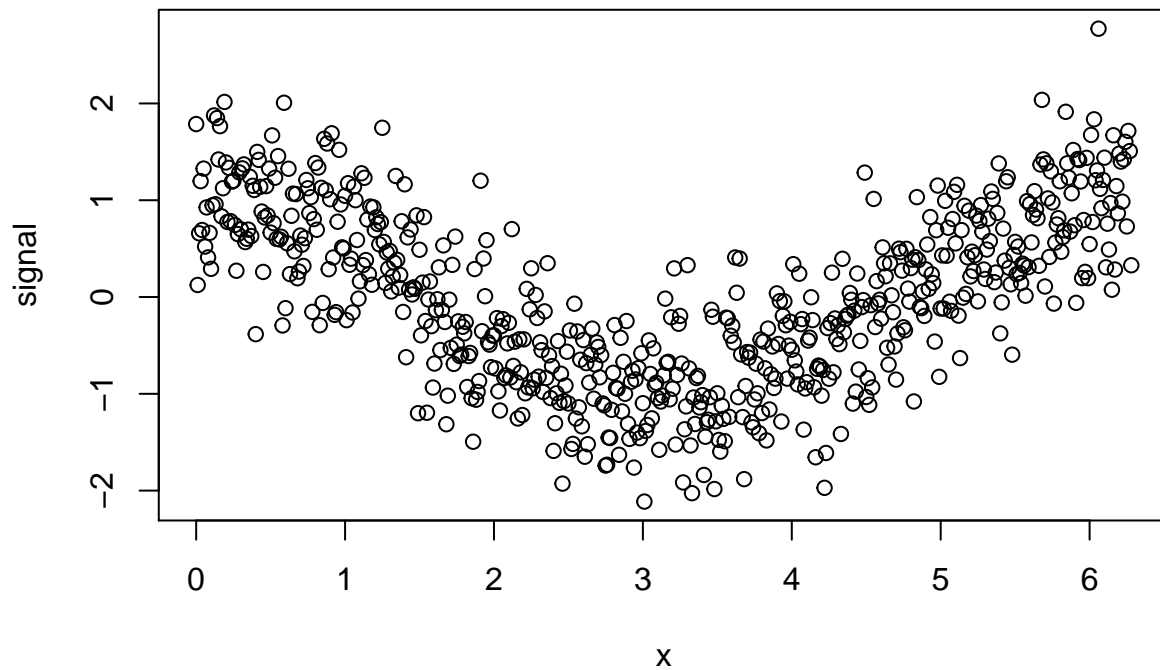
8/11/2017

## Generate Data

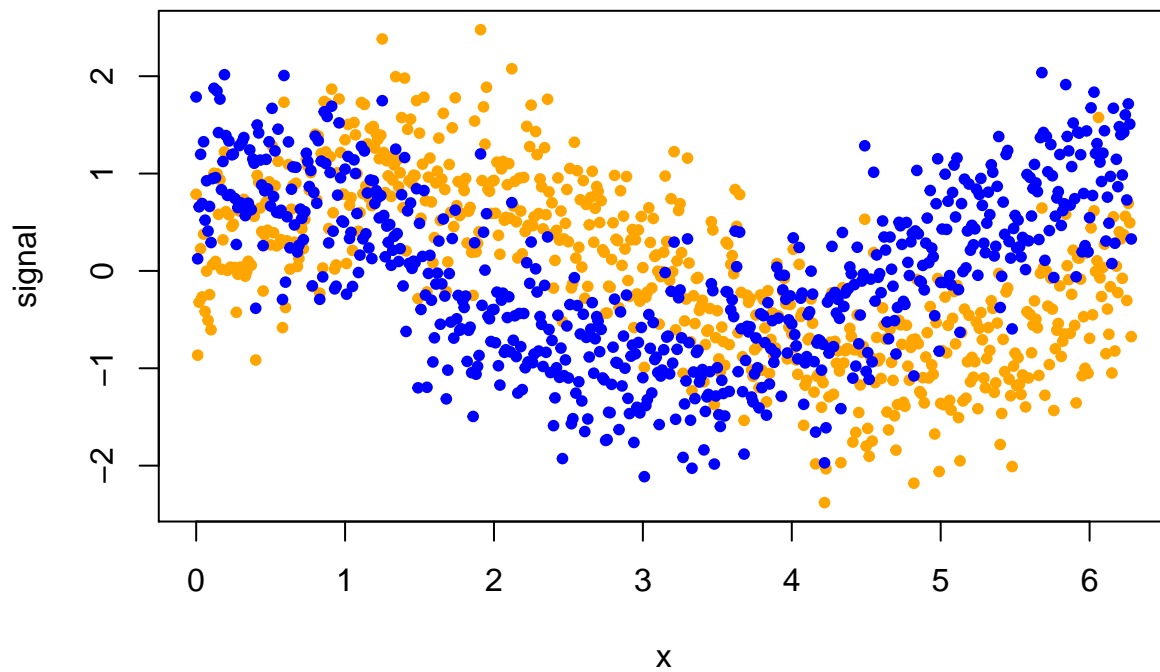
```
suppressWarnings(library(e1071))
set.seed(1180)
x<-seq(0,2*pi,by=.01)
sigmaEps<-.5
eps<-rnorm(length(x),0,sigmaEps)
signal<-sin(x)+eps
data1<-data.frame(x=x,signal=signal)
plot(data1)
```



```
data2<-data.frame(x=x,signal=cos(x)+eps)
plot(data2)
```



```
plot(data1,pch=20,col="orange")
points(data2,pch=20,col="blue")
```



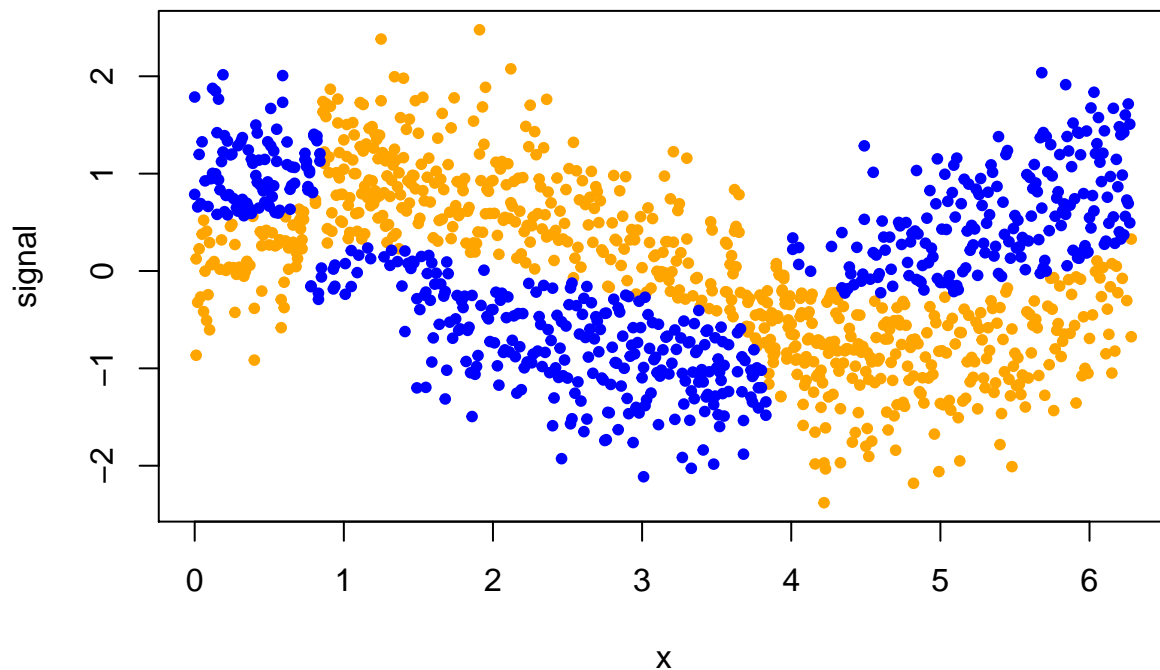
## SVM with radial base function kernel

```
data1$class<-1
data2$class<-2
dataClasses<-rbind(data1,data2)
svmFit.radial <- svm(class~., data = dataClasses)
```

```
summary(svmFit.radial)
```

```
##  
## Call:  
## svm(formula = class ~ ., data = dataClasses)  
##  
##  
## Parameters:  
##   SVM-Type:  eps-regression  
## SVM-Kernel:  radial  
##      cost:   1  
##    gamma: 0.5  
##  epsilon: 0.1  
##  
##  
## Number of Support Vectors: 1025
```

```
plot_class = function(dataClasses, model){  
  plot(dataClasses[model$decision.values<0,1:2],pch=20,col="orange")  
  points(dataClasses[model$decision.values>0,1:2],pch=20,col="blue")  
  ylim = c(-2.5, 2.5)  
}  
plot_class(dataClasses, svmFit.radial)
```

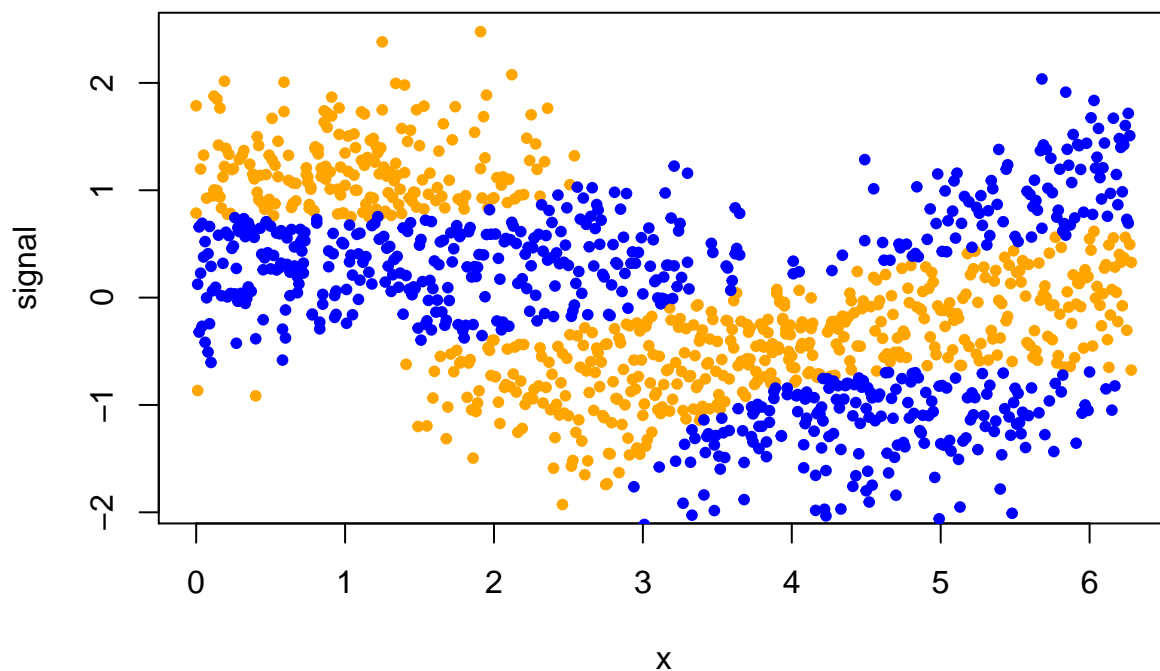


### SVM with sigmoid kernel (tanh function)

```
svmFit.sigmoid <- svm(class~., data = dataClasses,kernel="sigmoid")  
summary(svmFit.sigmoid)
```

```
##  
## Call:
```

```
## svm(formula = class ~ ., data = dataClasses, kernel = "sigmoid")
##
##
## Parameters:
##   SVM-Type:  eps-regression
##   SVM-Kernel: sigmoid
##     cost:    1
##     gamma:   0.5
##   coef.0:    0
##   epsilon:   0.1
##
##
## Number of Support Vectors: 1256
plot_class(dataClasses, svmFit.sigmoid)
```

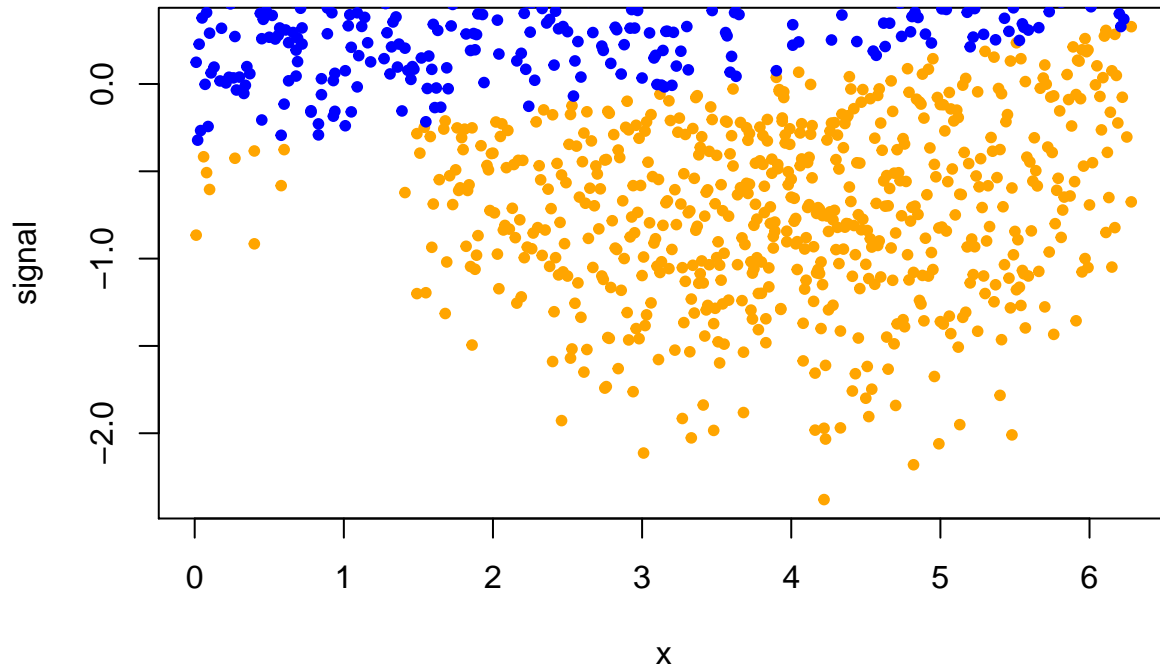


## SVM with linear kerner

```
svmFit.linear <- svm(class~., data = dataClasses,kernel="linear")
summary(svmFit.linear)
```

```
##
## Call:
## svm(formula = class ~ ., data = dataClasses, kernel = "linear")
##
##
## Parameters:
##   SVM-Type:  eps-regression
##   SVM-Kernel: linear
##     cost:    1
##     gamma:   0.5
##   epsilon:   0.1
```

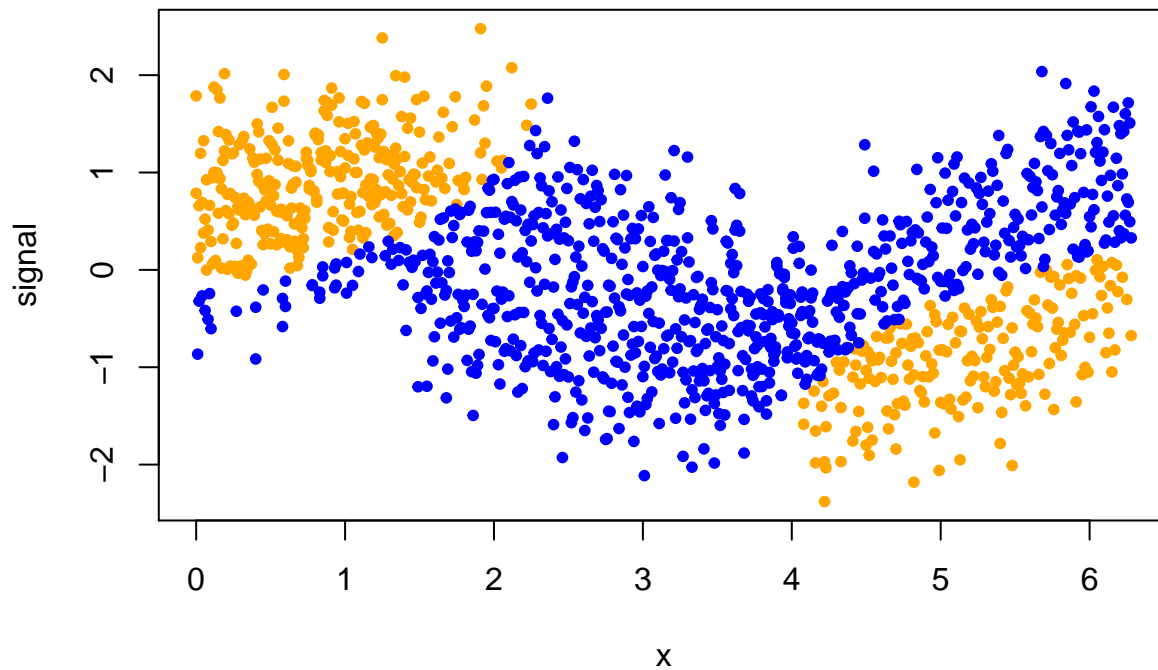
```
##
##
## Number of Support Vectors: 1258
plot_class(dataClasses, svmFit.linear)
```



## SVM with polynomial kerner

```
svmFit.polynomial2 <- svm(class~., data = dataClasses, kernel="polynomial",degree=2)
summary(svmFit.linear)
```

```
##
## Call:
## svm(formula = class ~ ., data = dataClasses, kernel = "linear")
##
##
## Parameters:
##   SVM-Type:  eps-regression
##   SVM-Kernel: linear
##     cost:    1
##    gamma:   0.5
##   epsilon:  0.1
##
##
## Number of Support Vectors: 1258
plot_class(dataClasses, svmFit.polynomial2)
```



```
svmFit.polynomial3 <- svm(class~., data = dataClasses, kernel="polynomial",degree=3)
summary(svmFit.linear)
```

```
##
## Call:
## svm(formula = class ~ ., data = dataClasses, kernel = "linear")
##
##
## Parameters:
##   SVM-Type:  eps-regression
## SVM-Kernel:  linear
##      cost:   1
##      gamma:  0.5
##      epsilon: 0.1
##
##
## Number of Support Vectors: 1258
plot_class(dataClasses, svmFit.polynomial3)
```

