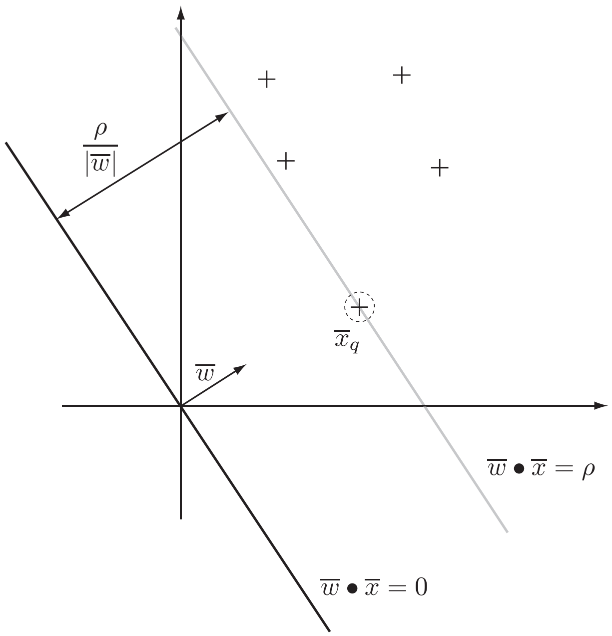
Novelty Dectection

Novelty/outlier detection – find points in an *unlabeled* data set that do not conform to the general distribution pattern of the overall data set.

The central idea in novelty detection with maximum margin machines is to construct a hyperplane through the origin of the input space whose margin separates the unlabeled training points from the origin in some optimal way.



Maximum Margin Machines

Assume we have an unlabeled training set,

$$D = \{\overline{x}_1, \overline{x}_2, \ldots, \overline{x}_l\} \subset \mathbb{R}^n.$$

whose elements are located only in the first hyperoctant (the components of all vectors are positive) and can be linearly separated from the origin.

In this case maximizing the margin of a hyperplane going through the origin gives rise to the following convex optimization problem,

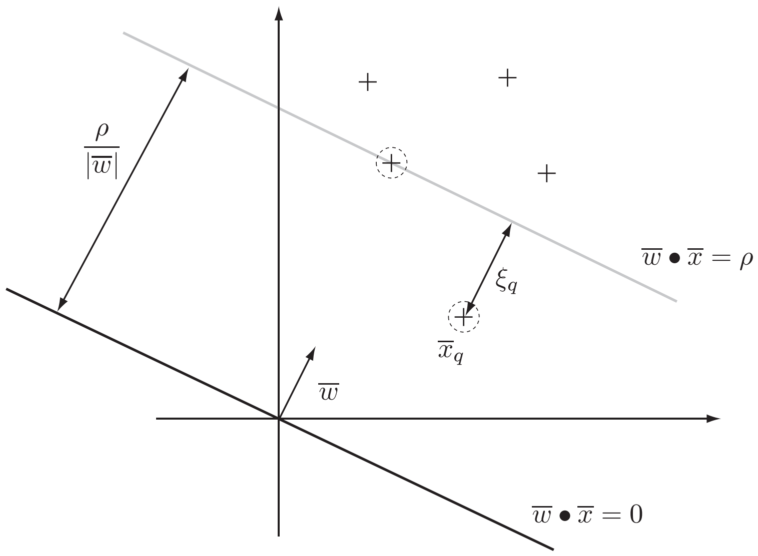
$$\min \phi(\overline{w}, \rho) = \min_{\overline{w}, \rho} \frac{1}{2} \overline{w} \bullet \overline{w} - \rho,$$

subject to the constraints,

$$\overline{w} \bullet \overline{x}_i \geq \rho,$$

where $\overline{x}_i \in D$.

Outliers are Margin Errors!



$$\min \phi(\overline{w}, \rho) = \min_{\overline{w}, \rho} \frac{1}{2} \overline{w} \bullet \overline{w} - \rho + \frac{1}{\nu\, l} \sum_{i=1}^{l} \xi_i,$$

subject to the constraints,

$$\overline{w} \bullet \overline{x}_i \geq \rho - \xi_i,$$
$$\xi_i \geq 0,$$

for $i = 1, \ldots, l$ and $0 < \nu \leq 1$.

Classifying Outliers

We can perform a simple trick that allows us to construct a decision surface that will classify any point not in the training data as a novelty or not:

> *We translate the hyperplane that runs through the origin in such a way that it coincides with the supporting hyperplane of the margin and consider it a decision surface that separates outliers from the rest of the data.*

More formally,

$$\hat{f}(\overline{x}) = \overline{w}^* \bullet \overline{x} - \rho^*,$$

where $\overline{w}^*$ and ρ^* are solutions to the Lagrangian dual.

Now, given a point $\overline{z} \in \mathbb{R}^n$, if that point lies below the decision surface, $\hat{f}(\overline{z}) < 0$, then we consider the point an outlier.

The Dual

For the dual we construct the Lagrangian,

$$L(\overline{\alpha}, \overline{\beta}, \overline{w}, \rho, \overline{\xi}) = \frac{1}{2}\overline{w} \bullet \overline{w} - \rho + \frac{1}{\nu\, l}\sum_{i=1}^{l} \xi_i$$

$$- \sum_{i=1}^{l} \alpha_i \left(\overline{w} \bullet \overline{x}_i - \rho + \xi_i\right)$$

$$- \sum_{i=1}^{l} \beta_i \xi_i .$$

As usual we formulate this as the Lagrangian optimization,

$$\max_{\overline{\alpha}, \overline{\beta}} \min_{\overline{w}, \rho, \overline{\xi}} L(\overline{\alpha}, \overline{\beta}, \overline{w}, \rho, \overline{\xi}),$$

subject to the constraints,

$$\alpha_i \geq 0,$$
$$\beta_i \geq 0,$$

for $i = 1, \ldots, l$.

The Dual

It is now straightforward to state the KKT conditions and from a solution that has to satisfy the KTT conditions it is easy to derive the dual,

$$\max \phi'(\overline{\alpha}) = \max_{\overline{\alpha}} \left(-\frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \alpha_j \overline{x}_i \bullet \overline{x}_j \right),$$

subject to the constraints,

$$\frac{1}{\nu\, l} \geq \alpha_i \geq 0,$$

$$\sum_{i=1}^{l} \alpha_i = 1,$$

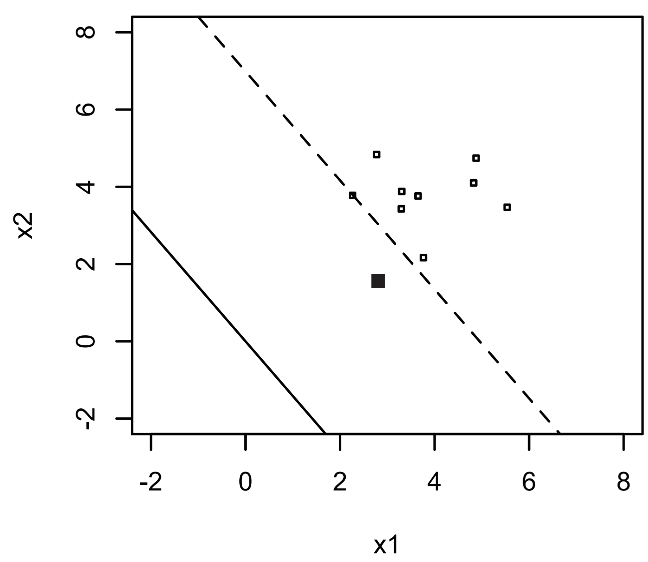
for $i = 1, \ldots, l$.

Novelty Detection in R

```
library(e1071)
# setup our output device
quartz(height=4,width=4,pointsize=8)

# create a 2D data set and plot as squares
x1 <- rnorm(10,mean=4)
x2 <- rnorm(10,mean=4)
x <- data.frame(x1,x2)
plot(x,pch=22,cex=.5,xlim=c(-2,8),ylim=c(-2,8))

# build the novelty detection model
model <- svm(x,
             type="one-classification",
             kernel="linear",
             nu=0.1,
             scale=FALSE)

# plot the support vector outliers as filled squares
ix <- model$index[model$coefs == 1.0]
x1 <- x$x1[ix]
x2 <- x$x2[ix]
sv <- data.frame(x1,x2)
points(sv,type="p",pch=22,cex=.5,bg="red",col=2)
```

Novelty Detection in R

```
# plot the hyperplane together with the
# margin that constitutes the novelty decision surface
w1 <- sum(x$x1[model$index]*model$coefs)
w2 <- sum(x$x2[model$index]*model$coefs)
slope <- -(w1/w2)
offset <- (model$rho/w2)
abline(a=offset,b=slope,lty=2)
abline(a=0,b=slope)
```



Novelty Detection in R

```
library(e1071)
# setup our output device
quartz(height=4,width=4,pointsize=8)

# create a 2D data set and plot as squares
x1 <- rnorm(100)
x2 <- rnorm(100)
x <- data.frame(x1,x2)
plot(x,pch=22,cex=.5,xlim=c(-4,4),ylim=c(-4,4))

# build the novelty detection model
model <- svm(x,
             type="one-classification",
             kernel="radial",
             gamma=0.1,
             nu=0.05)

# plot the support vectors as filled squares
ix <- model$index[model$coefs == 1.0]
x1 <- x$x1[ix]
x2 <- x$x2[ix]
sv <- data.frame(x1,x2)
points(sv,type="p",pch=22,cex=.5,bg="red",col=2)
```

Novelty Detection in R