SVM Multi-Class Classification

• A SVM is a binary classifier, that is, the class labels can only take two values: ±1.

• Many real-world problems, however, have more than two classes (e.g. optical character recognition).

One Versus the Rest: To get $M$-class classifiers, construct set of binary classifiers $f^1, f^2, \ldots, f^M$, each trained to separate one class from rest.

Combine them to get a multi-class classification according to the maximal output before applying the sgn function.

$$\arg\max_{j=1 \ldots M} g^j(x), \text{ where } g^j(x) = \sum_{i=1}^{m} y_i \alpha_i^j k(x, x_i) + b^j.$$
SVM Multi-Class Classification (cont.)

• Recall: $g^j(x)$ returns a signed real-valued value which can be interpreted as the distance from the separation (hyper)plane to the point $x$.

• Value can also be interpreted as a confidence value. The larger the value the more confident one is that the point $x$ belong to the positive class.

• Hence, assign point $x$ to the class whose confidence value is largest for this point.
SVM Pairwise Classification

- Train a classifier for each possible pair of classes.
- For $M$ classes, this results in $\binom{M}{2} = \frac{(M-1)M}{2}$ binary classifiers.
- Classify an unknown point $x$ by applying each of the $\binom{M}{2}$ binary classifiers and count how many times point $x$ was assigned to that class label.
- Class label with highest count is then the considered label for the unknown point $x$. 

- p. 251
One-Class SVM for Novelty Detection

Idea: enclose data with a hypersphere and classify new data as *normal* if it falls within the hypersphere and otherwise as anomalous data.
Minimum Enclosing Hypersphere

Given normal data \( \mathcal{X} = \{x_1, x_2, \ldots, x_m\} \in \mathbb{R}^d \) and let \( r \) be the radius of the hypersphere and \( c \in \mathbb{R}^d \) the center. To find the minimum enclosing hypersphere we have to solve the following optimization problem:

\[
\begin{align*}
\text{minimize} & \quad r^2 \\
\text{subject to} & \quad \|\Phi(x_i) - c\|^2 \leq r^2, \quad i = 1, \ldots, m.
\end{align*}
\]

Lagrangian multiplier \( \alpha_i \geq 0 \) for each constraint

\[
L(c, r, \alpha) = r^2 + \sum_{i=1}^{m} \alpha_i \left\{ \|\Phi(x_i) - c\|^2 - r^2 \right\}
\]
Minimum Enclosing Hypersphere (cont.)

Setting the derivatives with respect to $c$ and $r$ to zero

\[
\frac{\partial L(c, r, \alpha)}{\partial c} = 2 \sum_{i=1}^{m} \alpha_i (\Phi(x_i) - c) = 0
\]

\[
\frac{\partial L(c, r, \alpha)}{\partial r} = 2r \left(1 - \sum_{i=1}^{m} \alpha_i\right) = 0
\]

one obtains the following equations

\[
\sum_{i=1}^{m} \alpha_i = 1 \text{ and } c = \sum_{i=1}^{m} \alpha_i \Phi(x_i).
\] (1)
Minimum Enclosing Hypersphere (cont.)

Inserting relation (1) into

\[ L(c, r, \alpha) = r^2 + \sum_{i=1}^{m} \alpha_i \left\{ \| \Phi(x_i) - c \|^2 - r^2 \right\} \]

\[ = \sum_{i=1}^{m} \alpha_i \| \Phi(x_i) - c \|^2 \]

\[ = \sum_{i=1}^{m} \alpha_i k(x_i, x_i) - \sum_{i,j=1}^{m} \alpha_i \alpha_j k(x_i, x_j) \]

gives the dual form.
Minimum Enclosing Hypersphere (cont.)

To find $\alpha$ in dual form, solve optimization problem:

maximize $W(\alpha) = \sum_{i=1}^{m} \alpha_i k(x_i, x_i) - \sum_{i,j=1}^{m} \alpha_i \alpha_j k(x_i, x_j)$

subject to $\sum_{i=1}^{m} \alpha_i = 1$, and $\alpha_i \geq 0$, $i = 1, \ldots, m$.

Recall: Lagrange multiplier can be non-zero only if the corresponding inequality constraint is an equality at the solution.
The KKT complementarity conditions are satisfied by the optimal solutions $\alpha, (c, r)$

$$\alpha_i \left\{ \|\Phi(x_i) - c\|^2 - r^2 \right\}, \quad i = 1, \ldots, m.$$ 

This implies that only training examples $x_i$ that lie on the surface of the optimal hypersphere have their corresponding $\alpha_i > 0$. 
Decision Function

\[ f(x) = \text{sgn}(r^2 - \|\Phi(x) - c\|^2) \]

\[ = \text{sgn} \left( r^2 - \left\{ (\Phi(x) \cdot \Phi(x)) - 2 \sum_{i=1}^{m} \alpha_i (\Phi(x) \cdot \Phi(x_i)) \right\} \right. \]

\[ + \sum_{i,j=1}^{m} \alpha_i \alpha_j (\Phi(x_i) \cdot \Phi(x_j)) \left\} \right) \]

\[ = \text{sgn} \left( r^2 - \left\{ k(x, x) - 2 \sum_{i=1}^{m} \alpha_i k(x, x_i) \right\} \right. \]

\[ + \sum_{i,j=1}^{m} \alpha_i \alpha_j k(x_i, x_j) \right\} \) \]
Soft Enclosing Hypersphere

If we have some noise in our training set the “hard” enclosing hypersphere approach may force a larger radius than should really be needed. In other words, the solution would not be robust.

Aim: Find minimum enclosing hypersphere that contains (almost) all training examples, but not some small portion of extreme training examples.
Introduce slack variables $\xi, \xi_i \geq 0, i = 1, \ldots, m$

minimize $r^2 + C \sum_{i=1}^{m} \xi_i$

subject to $\| \Phi(x_i) - c \|^2 \leq r^2 + \xi_i, \quad \xi_i \geq 0, i = 1, \ldots, m$.

Lagrangian multiplier $\alpha_i, \beta_i \geq 0$ for each constraint

$L(c, r, \alpha, \beta) = r^2 + C \sum_{i=1}^{m} \xi_i$

$$+ \sum_{i=1}^{m} \alpha_i \left\{ \| \Phi(x_i) - c \|^2 - r^2 - \xi_i \right\} - \sum_{i=1}^{m} \beta_i \xi_i$$
Soft Enclosing Hypersphere (cont.)

Setting partial derivatives to 0 gives

\[ \sum_{i=1}^{m} \alpha_i = 1, \quad c = \sum_{i=1}^{m} \alpha_i \Phi(x_i) \]

This leads to the dual form

\[
\text{minimize} \quad \sum_{i,j=1}^{m} \alpha_i \alpha_j k(x_i, x_j) - \sum_{i=1}^{m} \alpha_i k(x_i, x_i) \\
\text{subject to} \quad 0 \leq \alpha_i \leq C, \quad \sum_{i=1}^{m} \alpha_i = 1
\]
Hyperplane One-Class SVM

Idea: Separate in high-dimensional feature space $\mathcal{F}$, the points from the origin (circled point) with a maximum distance, and allow $\nu \cdot m$ many “outliers” which lie between the origin and the hyperplane, i.e. the $-1$ side.
Hyperplane One-Class SVM (cont.)

Normal vector of the hyperplane is determined by solving the primal quadratic optimization problem

\[
\begin{align*}
\text{minimize} & \quad \frac{1}{2} \| w \|^2 + \frac{1}{\nu m} \sum_i \xi_i - \rho \\
\text{subject to} & \quad (w \cdot \Phi(x_i)) \geq \rho - \xi_i, \; \xi_i > 0, \; i = 1, \ldots, m. \quad (2) \\
\end{align*}
\]

Lagrangian multiplier \( \alpha_i, \beta_i \geq 0 \) for each constraint

\[
L(w, \xi, \rho, \alpha, \beta) = \frac{1}{2} \| w \|^2 + \frac{1}{\nu m} \sum_i \xi_i - \rho \\
- \sum_{i=1}^{m} \alpha_i((w \cdot \Phi(x_i)) - \rho + \xi_i) - \sum_{i=1}^{m} \beta_i \xi_i
\]

Reformulating (2) and (3) to a dual optimization problem in terms of a kernel function \( k(\cdot, \cdot) \), one obtains

– p. 263
maximize \[
\frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j k(x_i, x_j)
\] (4)
subject to \[0 \leq \alpha_i \leq \frac{1}{\nu m}, i = 1, \ldots, m\] and \[\sum_{i=1}^{m} \alpha_i = 1.\] (5)

Differentiating the primal with respect to \(w\), one gets
\[w = \sum_{i=1}^{m} \alpha_i \Phi(x_i).\]

Recall KKT theorem: For \(\alpha_i > 0\) the corresponding pattern \(x_i\) satisfies
\[\rho = (w \cdot \Phi(x_i)) = \sum_{j=1}^{m} \alpha_j k(x_j, x_i)\]
The decision function (left/right side of the hyperplane):

\[ f(x) = \text{sgn}((w \cdot \Phi(x_i)) - \rho) \]

\[ = \text{sgn} \left( \sum_{i=1}^{m} \alpha_i k(x_i, x) - \rho \right) \]

\(\nu\)-Property:

- \(\nu\) is an upper bound on the fraction of outliers.
- \(\nu\) is a lower bound on the fraction of Support Vectors.
Hyperplane One-Class SVM Example

\[ \nu = 0.05 \]

\[ \nu = 0.5 \]
Support Vector Regression

Basic idea: map the data $x$ into a high-dimensional feature space $\mathcal{F}$ via a nonlinear mapping $\Phi$, and do linear regression in this space.

$$f(x) = (w \cdot \Phi(x)) + b \text{ with } \Phi : \mathbb{R}^d \to \mathcal{F}, w \in \mathcal{F}.$$  

Linear regression in a high dimensional feature space corresponds to nonlinear regression in the low dimensional space $\mathbb{R}^d$.

Vapnik’s $\epsilon$-insensitive loss function:

$$|y - f(x)|_\epsilon := \max\{0, |y - f(x)| - \epsilon\}$$

Find function $f(x)$ that has at most $\epsilon$ deviation from all the targets $y_i$
Support Vector Regression (cont.)

Estimate linear regression $f(x) = (w \cdot \Phi(x)) + b$ leads to the problem of minimizing the term

$$\frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} |y_i - f(x_i)|_\epsilon$$

In the soft margin case one needs two types of slack variables $(\xi, \xi^*)$ for the two cases $f(x_i) - y_i > \epsilon$ and $y_i - f(x_i) > \epsilon$.

Figure is taken from Schölkopf’s and Smola’s book (Learning with Kernels) – p. 268
Support Vector Regression (cont.)

Optimization problem is given by:

\[
\begin{align*}
\text{minimize} & \quad \frac{1}{2}\|w\|^2 + C \cdot \sum_{i=1}^{n} (\xi_i + \xi_i^*) \\
\text{subject to} & \quad f(x_i) - y_i \leq \epsilon + \xi_i \\
& \quad y_i - f(x_i) \leq \epsilon + \xi_i^* \\
& \quad \xi_i, \xi_i^* \geq 0 \quad \text{for all } i = 1, \ldots, n
\end{align*}
\]
Support Vector Regression (cont.)

Introducing Lagrange multipliers $\alpha, \alpha^*$ (dual form):

maximize \[-\epsilon \sum_{i=1}^{n} (\alpha_i^* + \alpha_i) + \sum_{i=1}^{n} (\alpha_i^* - \alpha_i) y_i\]

\[-\frac{1}{2} \sum_{i,j}^{n} (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) k(x_i, x_j)\]

subject to \[0 \leq \alpha_i, \alpha_i^* \leq C \text{ for all } i = 1, \ldots, n \text{ and } \]

\[\sum_{i=1}^{n} (\alpha_i^* - \alpha_i) = 0\]

Regression estimate takes the form

\[f(x) = \sum_{i=1}^{n} (\alpha_i^* - \alpha_i) k(x_i, x) + b\]
Offset $b$ can be computed by exploiting Karush-Kuhn-Tucker conditions: $f(x_i) - y_i \leq \epsilon + \xi_i$ becomes an equality with $\xi_i = 0$ if $0 < \alpha_i < C$ and $y_i - f(x_i) \leq \epsilon + \xi_i^*$ becomes an equality with $\xi_i^* = 0$ if $0 < \alpha_i^* < C$ that is:

$$\alpha_i (\epsilon + \xi_i - y_i + (w \cdot \Phi(x_i)) + b) = 0$$

$$\alpha_i^* (\epsilon + \xi_i^* + y_i - (w \cdot \Phi(x_i)) - b) = 0$$

and leads to solution

$$b = y_i - (w \cdot \Phi(x_i)) - \epsilon \quad \text{for } \alpha_i \in (0, C')$$

$$b = y_i - (w \cdot \Phi(x_i)) + \epsilon \quad \text{for } \alpha_i^* \in (0, C')$$
library(kernlab);
x <- seq(-20,20,0.1);
y <- sin(x)/x + rnorm(401,sd=0.03);
# train SVM
reg.svm <- ksvm(x,y,epsilon=0.01,kpar=list(sigma=16),cross=3);
plot(x,y,type="l",lwd=3);
lines(x,predict(reg.svm,x),col="red",lwd=3);