1. Classification
    1.1. Support Vector Data Description

A linearly separable two-class dataset in n-dimensions can be separated by an (n-1)-dimensional hyperplane of the form

$$\mathrm{y}(\mathbf{x}) = \mathbf{w}^{\mathbf{T}}\mathbf{x} + \mathrm{b}$$

(5.1.1.)

as illustrated in figure 1 for a 2-dimensional dataset with two classes.

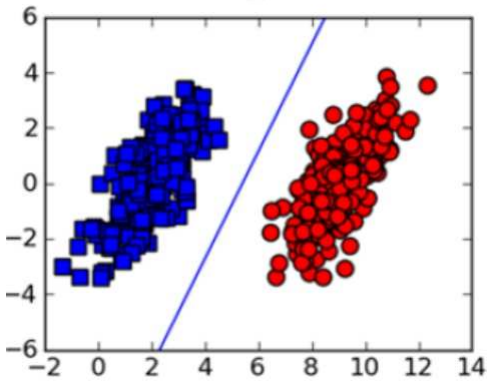

Fig. 1 Linearly separable two class dataset

For a linearly separable dataset with data vectors $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n$ and the corresponding class labels $\boldsymbol{t}_1, \boldsymbol{t}_2, \ldots, \boldsymbol{t}_n$ where $\boldsymbol{t}_n \in \{-1,1\}$, at least one choice of the parameters $\mathbf{w}$ and b can be found such that 5.1.1. satisfies

$$\mathrm{y}(\mathbf{x}) > 0, \forall\, t_n = +1$$

and

$$\mathrm{y}(\mathbf{x}) < 0, \forall\, t_n = -1.$$

A support vector machine tries to find optimal values for the parameters $\mathbf{w}$ and b by maximizing the margin, which is defined as the perpendicular distance between the hyperplane and the points closest to the hyperplane, as shown in figure 2.

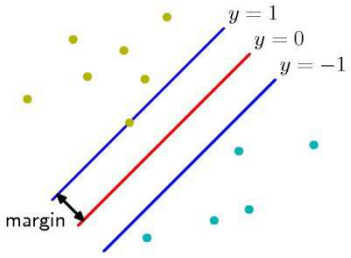

Fig. 2 The concept of margin

The solution for this optimization problem and the subsequent classification of test points solely depend on the points closest to the hyperplane marked by a circle in figure 3, which are known as the support vectors because they define and support the decision boundary.

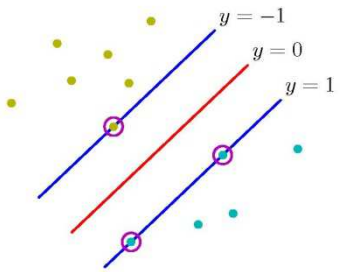

Fig. 3 The Support Vectors

The perpendicular distance of a point to a hyperplane of the form 5.1.1. is given by

$$\frac{|\mathrm{y}(\mathbf{x})|}{\|\mathbf{w}\|}.$$

With $t_n \mathrm{y}(\mathbf{x}_n) > 0$ for all correctly classified points of a training set and $\boldsymbol{t}_n \in \{-1,1\}$, the distance of a training vector to the decision surface is given by

$$\frac{t_n \mathrm{y}(\mathbf{x}_n)}{\|\mathbf{w}\|} = \frac{t_n\left(\mathbf{w}^{\mathbf{T}}\mathbf{x}_n + \mathrm{b}\right)}{\|\mathbf{w}\|}.$$

(5.1.2)

The support vector approach tries to find the closest points to a hyperplane and then adjusts the parameters $\mathbf{w}$ and b such that the distance (5.1.2) between the closest points and the hyperplane is maximized. This maximum margin solution can be found by solving (Bishop C. M., 2009)

$$\underset{w,b}{\operatorname{argmax}}\left\{\frac{1}{\|\mathbf{w}\|}\min_n\left[t_n\left(\mathbf{w}^{\mathbf{T}}\mathbf{x}_n + \mathrm{b}\right)\right]\right\}.$$

(5.1.3)

If (5.1.1) is set to

$$\mathbf{w}^{\mathbf{T}}\mathbf{x} + \mathrm{b} = 1$$

for the points closest to the hyperplane then the constraint

$$t_n\left(\mathbf{w}^{\mathbf{T}}\mathbf{x}_n + \mathrm{b}\right) \geq 1$$

(5.1.4)

holds for all training data points and

$$\min_n\left[t_n\left(\mathbf{w}^{\mathbf{T}}\mathbf{x}_n + \mathrm{b}\right)\right] = 1.$$

This reduces the optimization problem in (5.1.3) to the maximization of $\frac{1}{\|\mathbf{w}\|}$, which is equivalent to minimizing

$$\frac{1}{2}\|\mathbf{w}\|^2,$$

(5.1.5)

subject to the constraints given by (5.1.4). With the introduction of one Lagrange Multipliers for each constraint, the optimization problem can be transformed into the Lagrangian

$$L(\mathbf{w}, \mathrm{b}, \mathbf{a}) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{n=1}^{N} a_n\{t_n(\mathbf{w}^{\mathbf{T}}\mathbf{x}_n + \mathrm{b}) - 1\}.$$

(5.1.6)

Setting the partial derivatives of (5.1.6) with respect to $\mathbf{w}$ and $\mathbf{a}$ to zero results in the conditions

$$\mathbf{w} = \sum_{n=1}^{N} a_n t_n \mathbf{x}_n$$

(5.1.7)

and

$$0 = \sum_{n=1}^{N} a_n t_n$$

which can be plugged into (5.1.6) again to give

$$L(\mathbf{a}) = \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n\, a_m t_n t_m \mathbf{x}_n \mathbf{x}_m.$$

(5.1.8)

The last expression is also known as the dual representation of the optimization problem, with a dot product or inner product of the training vectors $\mathbf{x}_n\mathbf{x}_m$. This dot product can now be replaced by a kernel $(\mathbf{x}_n\mathbf{x}_m)$, which can be motivated as a similarity measure of two vectors (Schölkopf & Smola, 2002) and implicitly transforms the training data into a potentially infinite feature space. With the introduction of a kernel, (5.1.8) is transformed to

$$L(\mathbf{a}) = \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n\, a_m t_n t_m \mathrm{k}(\mathbf{x}_n \mathbf{x}_m).$$

(5.1.9)

The use of kernels makes the classifier considerably more flexible by allowing the substitution of a variety of different kernels and extends the application of support vector machines to problems which are non-linear in the original data space, through the implicit feature space transformation.

The maximum margin problem can now be solved by maximizing (5.1.9) with respect to $\mathbf{a}$ and subject to the constraints

$$a_n \geq 0, n = 1, \dots, N,$$

$$0 = \sum_{n=1}^{N} a_n t_n$$

(5.1.10)

If the expression for $\mathbf{w}$ derived in (5.1.7) is plugged into the initial hyperplane formula (5.1.1), new test points $\mathbf{x}$ can now be classified by evaluating the sign of

$$\mathrm{y}(\mathbf{x}) = \sum_{n=1}^{N} a_n t_n \mathrm{k}(\mathbf{x}, \mathbf{x}_n) + b$$

(5.1.11)

where the inequality in the first constraint of (5.1.10) only holds for a_n corresponding to the support vectors $\mathbf{x}_n$ and all other a_n are zero and do not contribute to the solution of (5.1.11).

The optimization problem (5.1.9) with the constraints given by (5.1.10) is based on the assumption, that data is linearly separable in the feature space to which it is implicitly transformed to by the kernel function. In case of overlapping data distribution in feature space, this approach can result in a high error on the training data and in a classifier with poor generalization. This can be prevented by relaxing the hard constraints defined by (5.1.4) through the introduction of slack variables ξ_n, such that each training data point satisfies

$$t_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + \mathrm{b}\right) \geq 1 - \xi_n$$

(5.1.12)

with $\xi_n = 0$ for data points that are on or within the correct margin boundary and $\xi_n = |t_n - \mathrm{y}(\mathbf{x}_n)|$ for any other training data point, as illustrated in figure 4 for the case of 2 dimensional data.

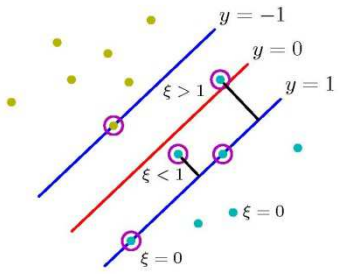

Fig. 4 Slack Variables

The maximum margin can now be found by minimizing

$$C\sum_{n=1}^{N}\xi_n+\frac{1}{2}\|\mathbf{w}\|^2$$

(5.1.13)

Where C regulates the tradeoff between the influence of the slack variables and the margin and thus controls the overall error.

With the introduction of ξ_n and the corresponding new constraints $\xi_n \geq 0$ for $1 \leq n \leq N$, the Lagrangian (5.1.6) now expands to

$$L(\mathbf{w},\mathrm{b},\boldsymbol{\xi},\mathbf{a},\boldsymbol{\mu})=\frac{1}{2}\|\mathbf{w}\|^2+C\sum_{n=1}^{N}\xi_n-\sum_{n=1}^{N}a_n\{t_n(\mathbf{w}^{\mathbf{T}}\mathbf{x}_n+\mathrm{b})-1+\xi_n\}-\sum_{n=1}^{N}\mu_n\xi_n\,.$$

(5.1.14)

Setting the partial derivatives with respect to **w**,b and **ξ** to zero and using the results to eliminate **w**,b and **ξ** from (5.1.14), the dual representation of the margin maximization problem again results in

$$L(\mathbf{a})=\sum_{n=1}^{N}a_n-\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}a_n\,a_m t_n t_m \mathrm{k}(\mathbf{x}_n\mathbf{x}_m)$$

Which is identical to the separable case (5.1.9) but with different constraints given by

$$0\leq a_n\leq C, n=1,\dots,N,$$

$$0=\sum_{n=1}^{N}a_n t_n.$$

(5.1.15)

The classification is again applied by evaluating the sign of (5.1.11).

The standard Support Vector Machine as introduced above is a supervised classification algorithm, which is in its basic form able to separate two classes and can be extended to multiclass and regression problems.

An approach to apply Support Vector Machines to semi-supervised or outlier detection problems was introduced by Tax and Duin as *Support Vector Data Description* (SVDD) (Tax & Duin, 2004). Instead of using a hyperplane to separate two or more classes, the SVDD tries to surround the available training data by a minimal hypersphere, defined by its radius R and the center **a**. The data enclosed in that hypersphere is the target class, data outside the boundary is considered outlier data.

Similar to the optimization problem in the training of a supervised Support Vector Machine, the goal is to find an optimal decision boundary which in SVDD is the minimal hypersphere enclosing all or most of the data. The optimal solution can be found by minimizing

$$R^2$$

subject to the constraints

$$\|\boldsymbol{x}_n - \boldsymbol{a}\|^2 \leq R^2, 1 \leq n \leq N$$

(5.1.16)

Where $\boldsymbol{a}$ and R are the center and the radius of the hyperspere and $\boldsymbol{x}_n$ is the n-th training data vector, with the assumption that the training data does not contain outliers.

To allow the occurrence of outlier data in the dataset, the constraints in (5.1.16) have to be softened to a certain degree by the introduction of slack variable, such that the new constraints are given by

$$\|\boldsymbol{x}_n - \boldsymbol{a}\|^2 \leq R^2 - \xi_n, 1 \leq n \leq N$$

(5.1.17)

and the optimization problem transforms to

$$C \sum_{n=1}^{N} \xi_n + R^2$$

(5.1.18)

where $\xi_n \geq 0$. Similar to (5.1.13), C controls the tradeoff between a tight decision boundary and the influence of outliers.

A compact formulation of the optimization problem is again obtained by combining the objective function (5.1.18) and the corresponding constraints (5.1.16) to a Lagrangian, which is given by

$$L(\mathrm{R}, \mathrm{b}, \boldsymbol{\xi}, \mathbf{a}, \boldsymbol{\mu}) = R^2 + C \sum_{n=1}^{N} \xi_n - \sum_{n=1}^{N} a_n \{R^2 + \xi_n - (\|\mathbf{x}_n\|^2 - 2\boldsymbol{x}_n \boldsymbol{a} + \|\boldsymbol{a}\|^2)\} - \sum_{n=1}^{N} \mu_n \xi_n .$$

(5.1.19)

To minimize the Lagrangian, its partial derivatives with respect to $\mathbf{w}$, b and ξ_n must be set to zero, which results in the new set of constraints

$$\sum_{n=1}^{N} a_n = 1$$

$$\mathbf{a} = \frac{\sum_{n=1}^{N} a_n \boldsymbol{x}_n}{\sum_{n=1}^{N} a_n} = \sum_{n=1}^{N} a_n \boldsymbol{x}_n$$

(5.1.20)

$$C - a_n - \mu_n = 0$$

(5.1.21)

Where $a_n \geq 0$, $\mu_n \geq 0$ is required for all Lagrange multipliers and $C \geq 0$ because of 5.1.18, and thus (5.1.21) implies

$$0 \leq a_n \leq C$$

(5.1.22)

The substitution of the relations (5.1.20) and (5.1.21) into the Lagrangian (5.1.19) results in the dual representation of the SVDD problem, which is given by

$$L(\mathbf{a}) = \sum_{n=1}^{N} a_n \, \mathbf{x}_n \mathbf{x}_n - \sum_{n=1}^{N} \sum_{m=1}^{N} a_n \, a_m \mathbf{x}_n \mathbf{x}_m,$$

subject to the constraints (5.1.21), where the inner products can again be replaced by kernels and the $\mathbf{x}_n$ corresponding to the Lagrange multipliers for which $a_n > 0$ are the Support Vectors.

Substituting (5.1.20) for **a** in the initial hypersphere constraint (5.1.16), a test point **x** is accepted if it satisfies

$$\mathbf{x}\mathbf{x} - 2 \sum_{n=1}^{N} a_n \mathbf{x} \boldsymbol{x}_n + \sum_{n=1}^{N} \sum_{m=1}^{N} a_n \, a_m \mathbf{x}_n \mathbf{x}_m \leq \mathbf{x}_{SV} \mathbf{x}_{SV} - 2 \sum_{n=1}^{N} a_n \boldsymbol{x}_n \mathbf{x}_{SV} + \sum_{n=1}^{N} \sum_{m=1}^{N} a_n \, a_m \mathbf{x}_n \mathbf{x}_m$$

(5.1.23)

And rejected otherwise. The left term in (5.1.23) corresponds to the left term of (5.1.16) and the right term is the radius R^2 defined by the support vectors $\mathbf{x}_{SV}$.

Figures 5 to 6 illustrate SVDD boundaries with a Radial Basis Function Kernel and different values for sigma, fitted to the same banana shaped dataset.

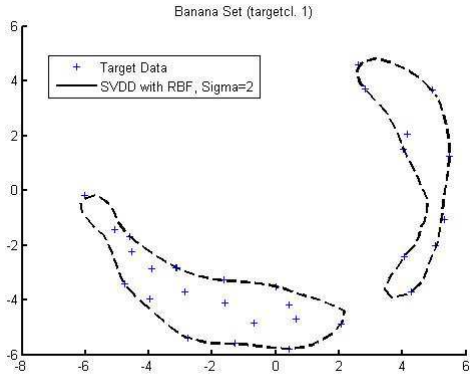

Fig. 5 SVDD with RBF, sigma=2

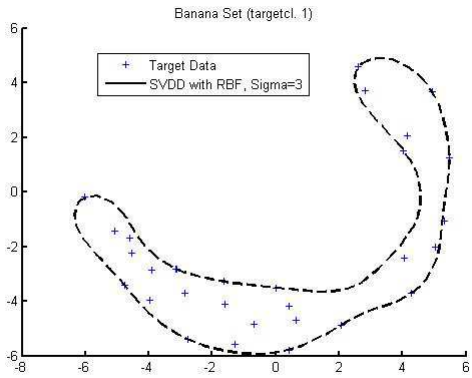

Fig. 6, SVDD with RBF, sigma=3

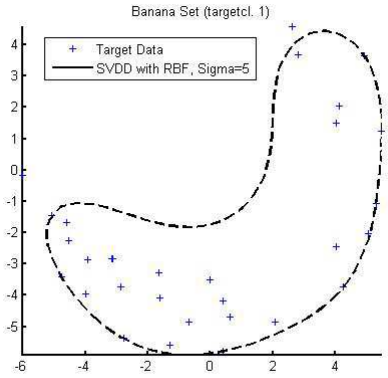

Fig. 7, SVDD with RBF, sigma=5

References

Bishop, C. M. (2009). *Pattern Recognition and Machine Learning.* Springer.

Schölkopf, B., & Smola, A. J. (2002). *Learning with Kernels.* London: MIT Press.

Tax, D. M., & Duin, R. P. (January 2004). Support Vector Data Description. *Machine Learning, Volume 54, Issue 1*, S. 54-66.