Machine Learning: 708.063 (VO)

7. Sparse Kernel Machines

Thomas Pock

Institute of Computer Graphics and Vision

June 3, 2020

Introduction

- In the kernel-based methods, we discussed so far, the kernel function $k(\boldsymbol{x}_n, \boldsymbol{x}_m)$ must be evaluated for all possible pairs $\boldsymbol{x}_n$ and $\boldsymbol{x}_m$.
- This can be computationally infeasible during training and during prediction.
- In this chapter we look at kernel-based methods that have sparse solutions, so that only a subset of the training data is needed for making predictions.
- The most popular method from this class of methods is the support vector machine (SVM) which is a very popular method for classification, regression, etc.
- A striking advantage of the SVM is that it is defined via a convex optimization problem.

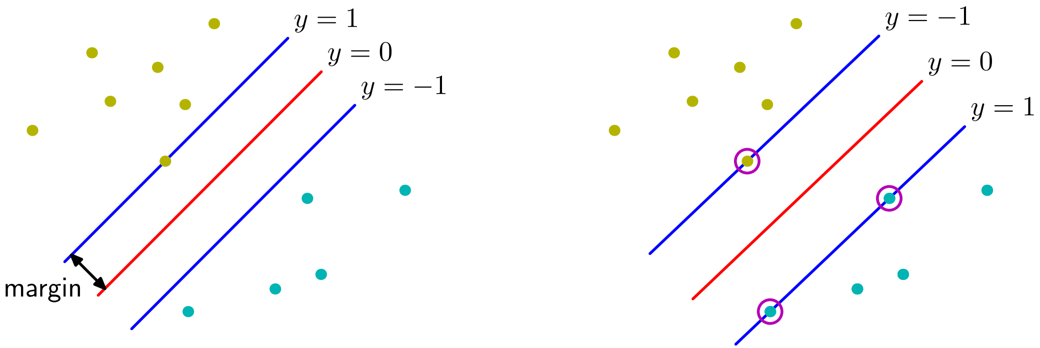
Maximum margin classifiers

- Let us consider the two-class classification problem

$$y(\boldsymbol{x}) = \boldsymbol{w}^T\phi(\boldsymbol{x}) + b.$$

- The training data consists of input vectors $\boldsymbol{x}_1, ..., \boldsymbol{x}_n$ with corresponding target values $t_1, ..., t_N$ with $t_n \in \{-1, +1\}$.
- A new data point is classified according to $\mathrm{sgn}(y(\boldsymbol{x}))$.
- We shall assume that the data set is linearly separable, that is there exists a hyperplane $\boldsymbol{w}, b$ such that $y(\boldsymbol{x}_n) > 0$ for vectors $\boldsymbol{x}_n$ with $t_n = +1$ and $y(\boldsymbol{x}_n) < 0$ for vectors $\boldsymbol{x}_n$ with $t_n = -1$, that is $t_n y(\boldsymbol{x}_n) > 0$ for all points.
- Of course, the choice of the hyperplane $\boldsymbol{w}, b$ is non-unique, but we should try choose the most robust one.
- A fruitful idea is the concept of the margin, which is defined to be the smallest distance between the hyperplane and any of the samples.
- The central idea of the SVM is to choose the hyperplane that maximizes the margin.

The geometry of the margin



- The left figure shows that the margin is defined as the perpendicular distance between the hyperplane and the closest data sample.
- The right figure shows the choice of the hyperplane with the maximum margin.
- Note that only three data points define the configuration of the hyperplane (sparse solution).

The support vector machine

- Recall that the unsigned distance of a sample to the hyperplane is given by $|y(\boldsymbol{x})|/\|\boldsymbol{w}\|$, which is equivalent to

$$\frac{t_n y(\boldsymbol{x}_n)}{\|\boldsymbol{w}\|} = \frac{t_n(\boldsymbol{w}^T \phi(\boldsymbol{x}_n) + b)}{\|\boldsymbol{w}\|}$$

- The margin is given by the distance of the closest point, that is

$$\text{margin} = \min_n \frac{t_n(\boldsymbol{w}^T \phi(\boldsymbol{x}_n) + b)}{\|\boldsymbol{w}\|} = \frac{1}{\|\boldsymbol{w}\|} \min_n t_n(\boldsymbol{w}^T \phi(\boldsymbol{x}_n) + b)$$

- The maximum margin solution is obtained by maximizing the margin with respect to the hyperplane parameters $\boldsymbol{w}, b$

$$\max_{\boldsymbol{w}, b} \left\{ \frac{1}{\|\boldsymbol{w}\|} \min_n t_n(\boldsymbol{w}^T \phi(\boldsymbol{x}_n) + b) \right\}$$

- This problem is equivalent to the convex optimization problem

$$\min_{\boldsymbol{w}, b} \frac{1}{2}\|\boldsymbol{w}\|^2 \quad \text{s.t.} \quad t_n(\boldsymbol{w}^T \phi(\boldsymbol{x}_n) + b) \geq 1, \; n = 1, ..., N,$$

  which is a quadratic optimization problem.

Dual SVM

- In the original form, the SVM is not straight-forward to solve.
- The Lagrange dual problem is given by

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

  subject to the constraints

$$0 \leq a_n, \quad n = 1, \ldots, N,$$
$$\sum_{n=1}^{N} a_n t_n = 0.$$

  and as before, the kernel function is given by

$$k(\boldsymbol{x}_n, \boldsymbol{x}_m) = \phi(\boldsymbol{x}_n)^T \phi(\boldsymbol{x}_m)$$

- This problem can be easily solved using a projected gradient version of Nesterov's accelerated gradient method.
- Note, however, that the dual problem is a maximization problem!

Classifying a new example

- In order to classify new data points using the trained model, one needs to evaluate the sign of

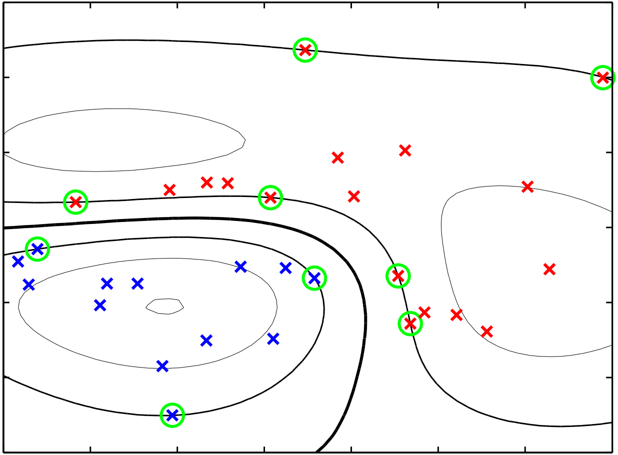
$$y(\boldsymbol{x}) = \sum_{n=1}^{N} a_n t_n k(\boldsymbol{x}, \boldsymbol{x}_n) + b.$$

- For an optimal dual vector $\boldsymbol{a}$ it follows from the KKT conditions that $a_n = 0$ or $t_n y(\boldsymbol{x}_n) = 1$, hence terms for which $a_n = 0$ will disappear from the sum.
- The remaining data points are called the support vectors and they lie on the maximum margin hyperplanes.
- Hence, once a model is trained the classification of new examples is computationally very efficient.
- The bias parameter b can be computed from the support vectors from the equations

$$t_n \left(\sum_{m \in \mathcal{S}} a_m t_m k(\boldsymbol{x}_n, \boldsymbol{x}_m) + b \right) = 1,$$

where $\mathcal{S}$ is the set of indices corresponding to support vectors.

Example



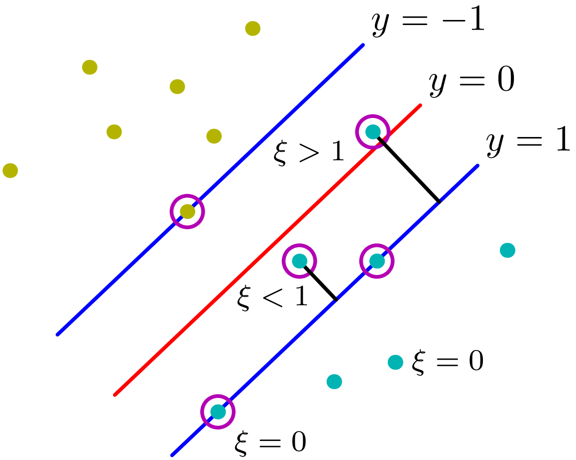
- SVM with Gaussian kernel function for a 2D problem.
- The bold line is the decision boundary
- The green circles mark the support vectors on both margins.

Overlapping class distributions

- So far we have assumed that the training data is linearly separable in the feature space $\phi(\boldsymbol{x})$.
- In practice, however, the class-conditional distributions might overlap, so that the assumption does no longer hold.
- Hence, we need to modify the SVM such that it allows for misclassified examples.
- The modified SVM should penalize examples which lie on the "wrong side" of the margin boundary.
- The modified SVM is known under the name soft-margin SVM.
- The main idea is to introduce slack variables $\xi_n \geq 0$ that account for feasibility errors in the constraints of the original SVM:

$$t_n(\boldsymbol{w}^T \phi(\boldsymbol{x}_n) + b) \geq 1 - \xi_n, \;\; n = 1, \ldots, N,$$

Geometric interpretation



- Observe that $\xi_n = 0$ means that the example is correctly classified and outside the margin, $0 < \xi_n \leq 1$ means that the example is still correctly classified but inside the margin and $\xi_n > 1$ means that the example is wrongly classified.

Soft-margin SVM

- In order to minimize the number of wrongly classified examples as well as examples inside the margins, the soft-margin SVM minimizes

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{n=1}^{N}\xi_n \quad \text{s.t. } t_n(\boldsymbol{w}^T\phi(\boldsymbol{x}_n)+b) \geq 1-\xi_n,\ \xi_n \geq 0,\ n=1,\dots,N,$$

  where the parameter $C > 0$ controls the trade-off between the slack variables penalty and the margin.
- Observe that for $C \to \infty$, one recovers the standard SVM.
- Since $\xi_n > 1$ for any misclassified feature $\phi(\boldsymbol{x}_n)$, the term $\sum_{n=1}^{N}\xi_n$ is also ab upper bound to the number of misclassified examples.
- The soft-margin SVM also poses a quadratic optimization problem which can be solved more easily in its dual formulation.

Dual soft-margin SVM

- The Lagrange dual problem of the soft-margin SVM is given by

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

  subject to the constraints

$$0 \leq a_n \leq C, \quad n = 1, ..., N,$$
$$\sum_{n=1}^{N} a_n t_n = 0.$$

  and as before, the kernel function is given by

$$k(\boldsymbol{x}_n, \boldsymbol{x}_m) = \phi(\boldsymbol{x}_n)^T \phi(\boldsymbol{x}_m)$$

- Note the "tiny" difference to the standard SVM, which is that the dual variables a_n are now upper bounded by the parameter C.

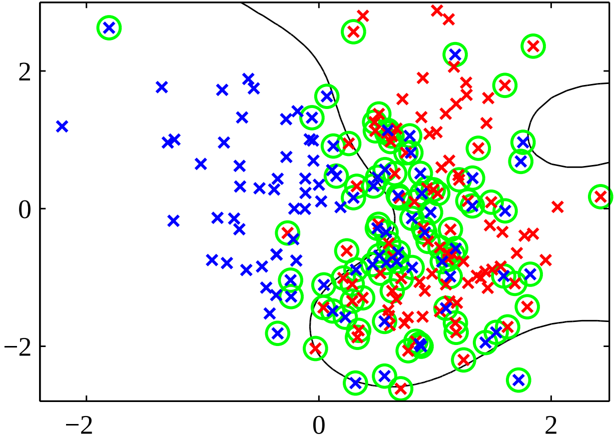
Classifying new examples

- As before, only vectors for which $a_n > 0$ will be of interest because otherwise they do not contribute to the predictive function

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

- Hence the classification of new examples is again very efficient.
- If $a_n = C$, it means that the support vectors are inside the margin or wrongly classified.
- The bias term b is computed - as for the standard SVM - from the support vectors that satisfy $0 < a_n < C$, $\xi_n = 0$ and hence $t_n y(\boldsymbol{x}_n) = 1$.

Example



- Example of the soft-margin SVM applied to non-separable data. The support vectors (also wrongly classified) are indicated by circles.

The hinge loss

- In the soft-margin SVM, we need to solve for fixed $t_n y_n$ the pointwise problem

$$h(t_n y_n) = \min_{\xi_n} \xi_n \quad \text{s.t.} \quad \xi_n \geq 0, \ \xi_n \geq 1 - t_n y_n.$$

- This can be equivalently be written as the hinge error function:

$$h(t_n y_n) = \max\{0, 1 - t_n y_n\}.$$

- Hence the soft-margin SVM can also be written as (setting $\lambda = (C)^{-1}$)

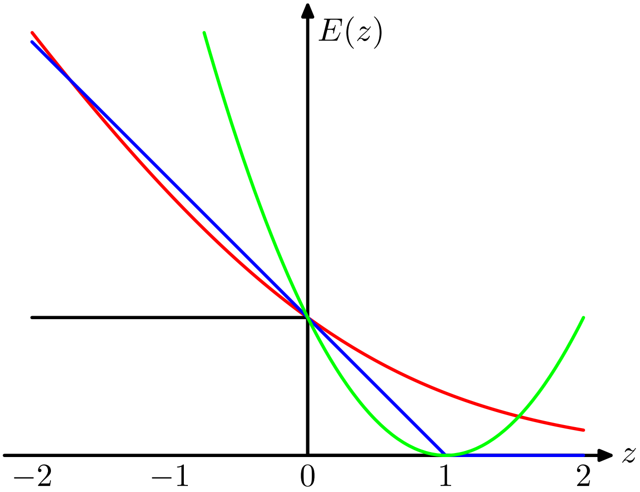
$$\min_{\boldsymbol{w}, b} \frac{\lambda}{2} \|\boldsymbol{w}\|^2 + \sum_{n=1}^{N} h(t_n y_n).$$

- On the other hand, the regularized (Bayesian) logistic regression model can be written as

$$\min_{\boldsymbol{w}, b} \frac{\lambda}{2} \|\boldsymbol{w}\|^2 + \sum_{n=1}^{N} l(t_n y(\boldsymbol{x}_n)),$$

with $l(s) = \log(1 + \exp(-s))$.

SVM vs. logistic regression vs. least squares classification



- The error function of the soft-margin SVM is shown in blue.
- The error function of logistic regression (rescaled) is shown in red.
- The error function of least squares regression is shown in green.
- Note the similarity between the SVM and logistic regression.
- The quadratic error function is very different which explains their sensitivity to outliers.

Multiclass SVMs

- In principle, the SVM is a 2-class classifier but in practice we often want to solve problems with $K > 2$.
- It turns out the the extension of the SVM to multiple classes is not as straight-forward as it was for logistic regression.

- A common approach is to train K SVMs in an one-vs-rest approach, but this can lead to inconsistent results.
- Another approach is to train $K(K-1)/2$ SVMs in a one-vs-one approach, but this requires more computations for training and testing.
- There are also single objective function approaches that solve the multi-class SVM problem in a unified framework but it also requires significantly more computations in training and testing.

SVMs for regression

- We now show that we can also use the sparseness property of SVMs for regression.
- In quadratic regression, we used the following regularized least squares approach:

$$\min_{\boldsymbol{w},b} \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \frac{1}{2}\sum_{n=1}^{N}(y_n - t_n)^2$$

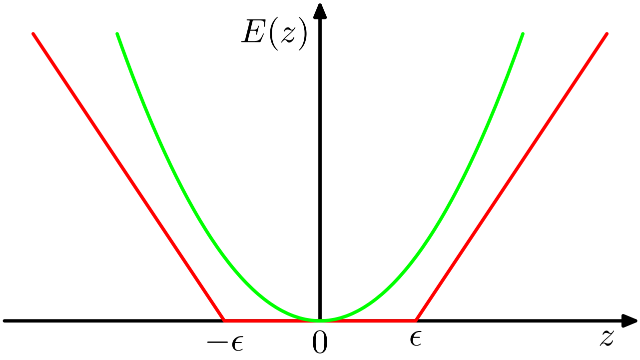
- The idea of SVM regression is now to replace the quadratic function $\frac{1}{2}(z)^2$ in the data fitting term by the function

$$\iota_\varepsilon(z) = \max(0, |z| - \varepsilon),$$

  which is called the ε-insensitive function.
- The objective function of the SVM regression problem is therefore

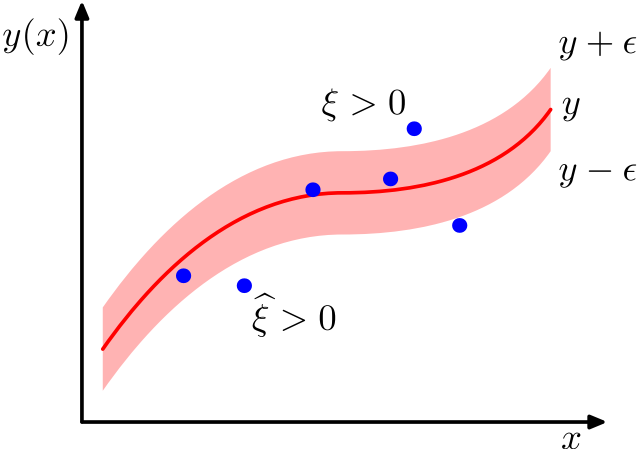
$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{n=1}^{N}\max(0, |y_n - t_n| - \varepsilon)$$

The ε-insensitive function



- The quadratic error function of quadratic regression is shown in green.
- The ε-insensitive function of SVM regression is shown in red.
- Note that the ε-insensitive function gives complete freedom if the error is between $\pm\varepsilon$ and only penalizes errors outside this range with a linear penalty.
- Hence, it is more robust to outliers compared to quadratic regression.

Visualization of SVM regression



- Points inside the red tube are not penalized by the $\iota_\varepsilon(\cdot)$ function.
- Points lying exactly on the boundary of the tube are the support vectors.
- Points outside the tube are considered as outliers.

The dual SVM regression

- We can also derive a dual problem, which is of the form

$$\begin{aligned} D(\boldsymbol{a}) &= -\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n a_m k(\boldsymbol{x}_n, \boldsymbol{x}_m) - \varepsilon \sum_{n=1}^{N} |a_n| + \sum_{n=1}^{N} t_n a_n \\ \text{s.t.} \quad & \sum_{n=1}^{N} a_n = 0, \quad -C \leq a_n \leq C, \quad n = 1, \ldots, N. \end{aligned}$$

- Finally, the regression function $y(\boldsymbol{x})$ is computed as

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

- In order to compute b we consider only points on the boundary of the tube, that is $|y_n - t_n| = \varepsilon \implies -C < a_n < C$, and to solve:

$$\left| \sum_{m=1}^{N} a_m k(\boldsymbol{x}_n, \boldsymbol{x}_m) + b - t_n \right| = \varepsilon,$$

- Numerically it is more stable to average over such estimates.

SVMs for big data sets

- We have argued that solving the SVM problems in the dual formulation is advantageous.
- This is only true if
  - The primal problem has very high dimension and hence cannot be solved.
  - The size of the training data set is still small enough such that storing (or at least computing) the kernel matrix $K \in \mathbb{R}^{N \times N}$ is feasible.
- By the recently seen popularity of "big data", it turns out that solving the dual problem might also be infeasible.
- Therefore, algorithms that can approximately solve the SVM or related problems for example based on stochastic (sub)gradient descent is a very active field of research.