Relevance Vector Machines

Erik Kuefler

CS152, Fall 2007

26 November 2007

Support Vector Machine Summary

- Used for classification or regression.
- Finds separating hyperplanes after using kernels function to transform the input into a higher-dimensional space.
- Minimizes errors while maximizing margin.
- $y(x) = \sum_{n=1}^{N} w_n K(x, x_n) + w_0$
- Where w_n is the vector of sample weights, x_n is the vector of input samples, and K is the kernel function.

Advantages

- Successfully applied to many real-world problems.
- Applicable to linearly inseparable data.
- Avoids overfitting.
- Sparse representation.
- No local minima.

Disadvantages

- Predictions are not probabilistic.
- Number of support vectors grows with training set size.
- Must specify C and ϵ.
- Difficult to extend to more than two classes.
- Kernel function must satisfy Mercer's condition.

Overview of Relevance Vector Machines

- Proposed by Michael Tipping in 2000.
- Uses a Bayesian approach to learning to generate probabilistic output.
- Determines relevance vectors instead of support vectors.
- Leads to much sparser representations.
- Expressed more naturally in terms of regression.
- Maintains the advantages of SVM's while avoiding their main limitations.

Mathematical Background

- Model gives a conditional distribution of the form $p(t|x, w, \beta) = \aleph(t|y(x), \beta^{-1})$ with mean
$$y(x) = \sum_{n=1}^{N} w_n K(x, x_n) + b.$$
- $\beta = \sigma^{-2}$ (inverse noise variance).
- Likelihood of target values given by
$$p(t|X, w, \beta) = \prod_{n=1}^{N} p(t_n|x_n, w, \beta^{-1}).$$
- Prior over weights given by $p(w|\alpha) = \prod_{i=1}^{M} \aleph(w_i|0, \alpha_i^{-1})$
- α is the set of hyperparameters, one for each w, which denote the precision of the corresponding parameters.

- This gives $p(w|t, X, \alpha, \beta) = \aleph(W|\mu, \Sigma)$, where $\mu = \beta\Sigma K^T t$, and $\Sigma = (A + \beta K^T K)^{-1}$ for $A = diag(\alpha_i)$.
- Learning proceeds by re-estimating α and β and the mean and covariance, where $\alpha_i^{new} = \frac{\gamma_i}{\mu_i^2}$ and $(\beta^{new})^{-1} = \frac{||t-KM||^2}{N-\Sigma_i\gamma_i}$.
- γ_i measures how well w_i is determined by the data, and is computed as $\gamma_i = 1 - \alpha_i\Sigma_{ii}$.
- Can also use the expectation-maximization algorithm (see Bishop 2006).
- Ultimately, most α are driven toward ∞, so their corresponding weights have zero mean and variance. The remaining weights are the relevance vectors.

Synthetic Data Example (Tipping 2000)

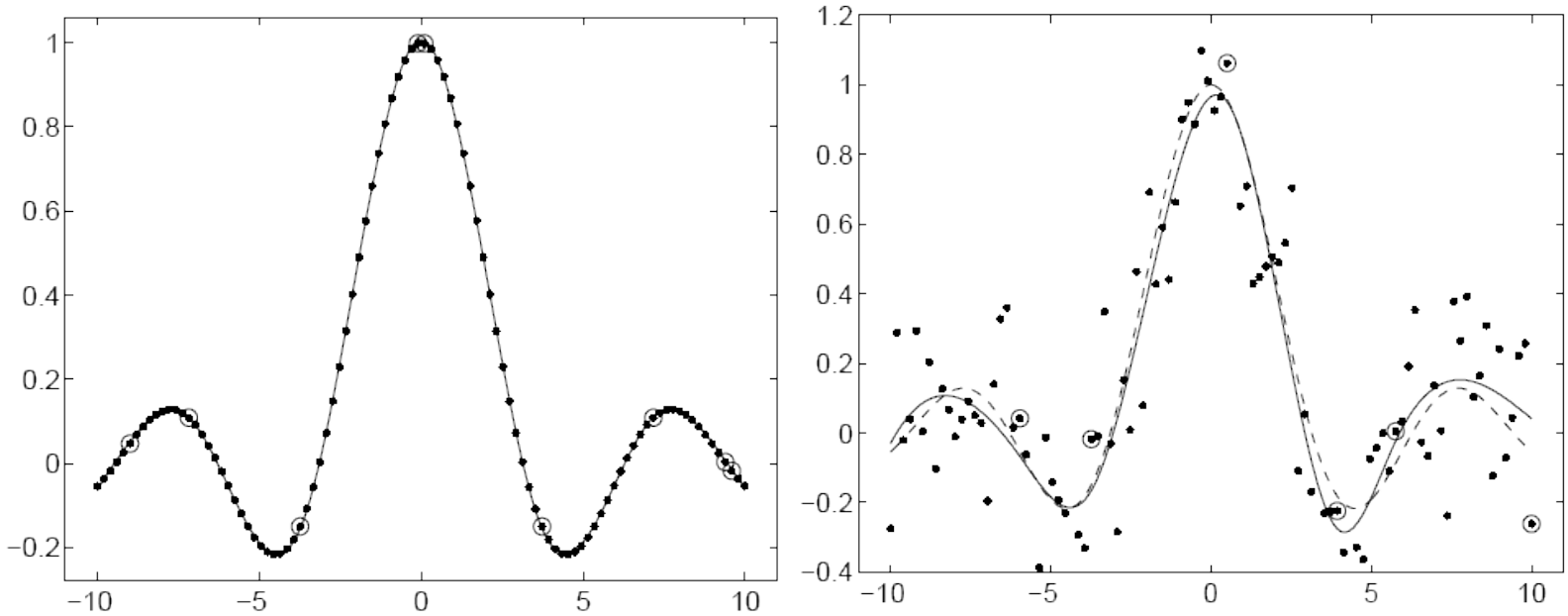

Figure 1: Relevance vector approximation to $\mathrm{sinc}(x)$: noise-free data (left), and with added Gaussian noise of $\sigma = 0.2$ (right). The estimated functions are drawn as solid lines with relevance vectors shown circled, and in the added-noise case (right) the true function is shown dashed.

RVM uses 9 relevance vectors; SVM uses 39 support vectors.

Benchmarks (Tipping 2000)

	errors		*kernels*	
Dataset	**SVR**	**RVR**	**SVR**	**RVR**
Friedman #1	2.92	2.80	116.6	59.4
Friedman #2	4140	3505	110.3	6.9
Friedman #3	0.0202	0.0164	106.5	11.5
Boston Housing	8.04	7.46	142.8	39.0

Support vector regression compared to relevance vector regression on some popular benchmarks.

Synthetic Data for Classification (Tipping 2000)

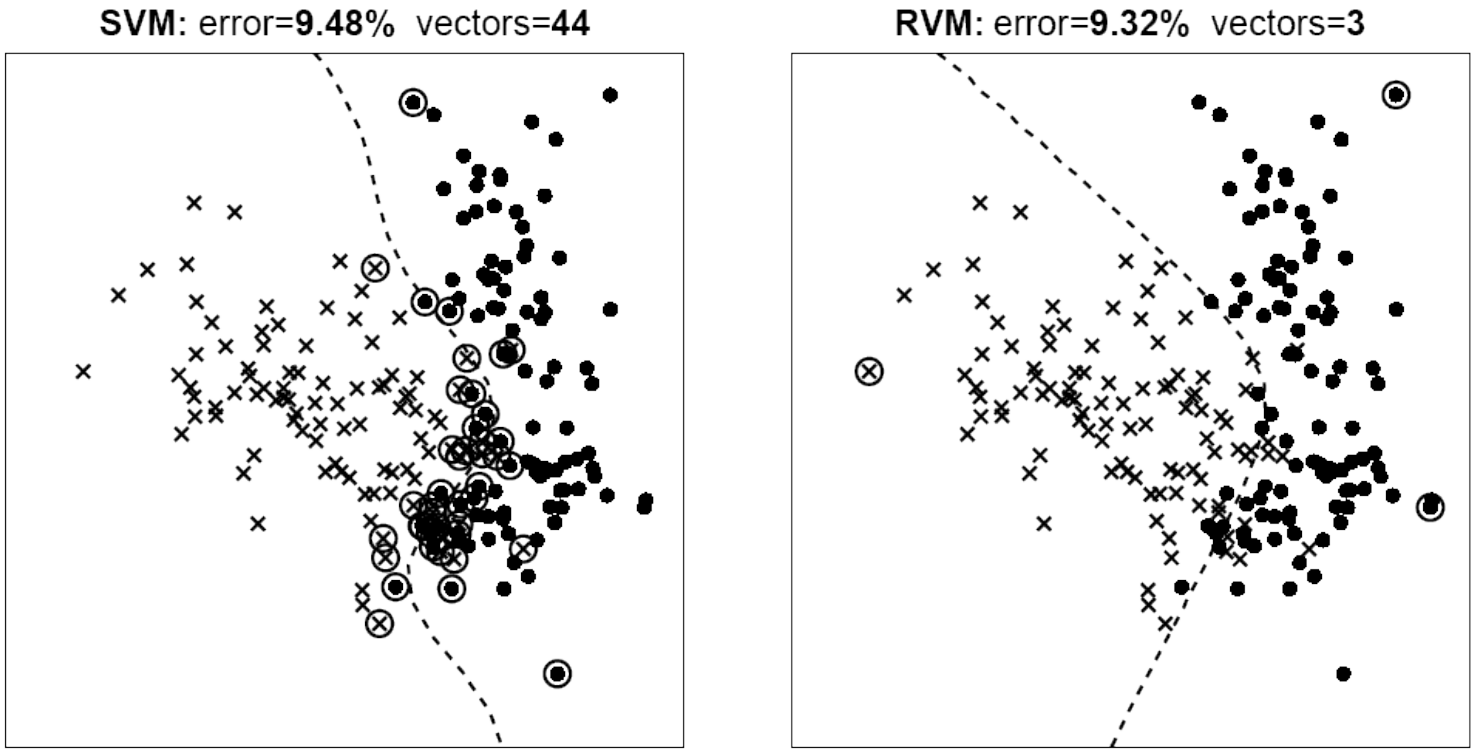

Figure 2: Results of training functionally identical SVM (left) and RVM (right) classifiers on a typical synthetic dataset. The decision boundary is shown dashed, and relevance/support vectors are shown circled to emphasise the dramatic reduction in complexity of the RVM model.

Real-World Data for Classification (Tipping 2000)

	errors			*kernels*		
Dataset	**SVM**	**GP**	**RVM**	**SVM**	**GP**	**RVM**
Pima Indians	67	68	65	109	200	4
U.S.P.S.	4.4%	–	5.1%	2540	–	316

RVM's compared to SVM's and Gaussian Processes. The USPS dataset consists of handwritten digit recognition. RVM is outperformed by SVM of the USPS data, but uses vastly fewer kernels.

More Synthetic Data (Bishop 2006)

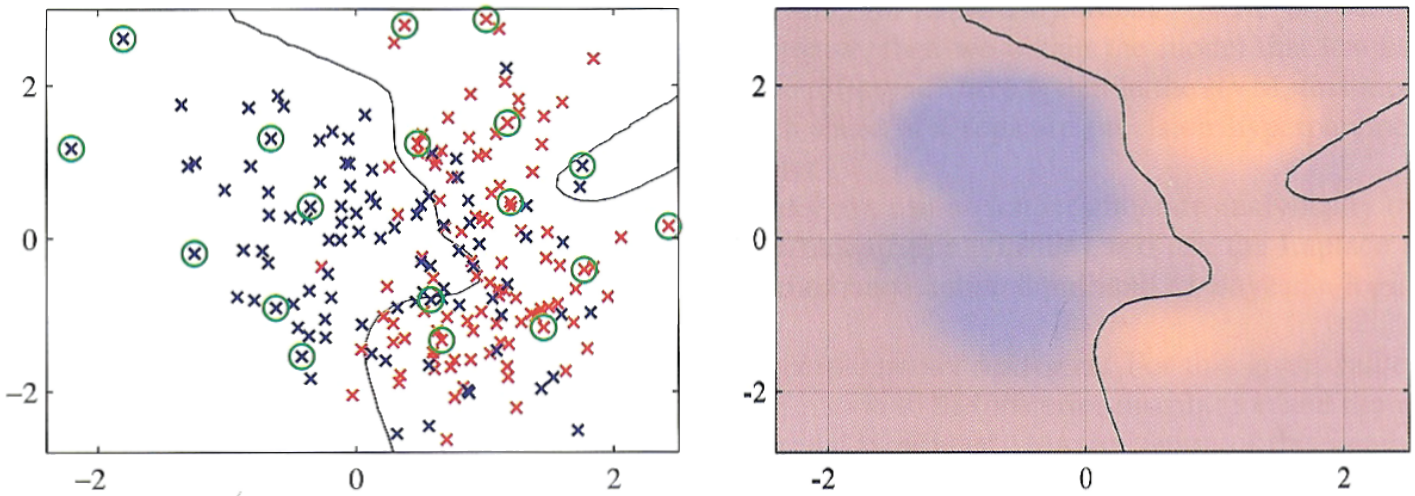

Figure 7.12 Example of the relevance vector machine applied to a synthetic data set, in which the left-hand plot shows the decision boundary and the data points, with the relevance vectors indicated by circles. Comparison with the results shown in Figure 7.4 for the corresponding support vector machine shows that the RVM gives a much sparser model. The right-hand plot shows the posterior probability given by the RVM output in which the proportion of red (blue) ink indicates the probability of that point belonging to the red (blue) class.

The relevance vector machine's ability to generate probabilistic output is a significant advantage.

Advantages

- Gives probabilistic predictions.
- Very sparse kernel representation.
- No parameters to be specified by the user.
- Extension to the multi-class case is relatively straightforward.
- Kernel not restricted to satisfying Mercer's condition.
- Slightly more accurate than SVMs for regression.
- Sparser representation leads to faster computation on test sets.

Disadvantages

- Training requires optimizing a non-convex function.
- Training a model with M basis functions requires inverting an $M \times M$ matrix, which is $O(M^3)$.
- The lack of cross-validation can offset this increase in training time in practice.
- Different training procedures may also improves training speed.

Conclusion

- Relevance vector machines are powerful in their ability to overcome many of the SVM's limitations while maintaining its primary advantages.
- The main disadvantage of RVM's is in their increased training time, but this can be mitigated in many situations.
- Ultimately, RVM's are able to obtain similar accuracies to SVM's using an order of magnitude fewer kernels.

References

- C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer Science+Business Media, LLC, Singapore, 2003.
- M. E. Tipping. The Relevance Vector Machine. *Advances in Neural Information Processing Systems*, 2000.