Sparsifying LS-SVM Models via L_0-Norm Minimization

Jorge López[1] Kris De Brabanter[2] Johan A.K. Suykens[2]

[1]Dpto. de Ingeniería Informática and Instituto de Ingeniería del Conocimiento
Universidad Autónoma de Madrid

[2]Departement Electrotechniek (ESAT)
Katholieke Universiteit Leuven

Tuesday, September 28th 2010

Contents

LS-SVMs primal

Characteristics

- Simplification of standard SVMs via equality constraints.
- Common formulation for classification ($\overrightarrow{y} \in \{+1, -1\}^N$) and regression ($\overrightarrow{y} \in \mathbb{R}^N$).
- ϕ feature map of given Mercer kernel $k(\overrightarrow{x_i}, \overrightarrow{x_j}) = \phi(\overrightarrow{x_i}) \cdot \phi(\overrightarrow{x_j})$.
- Lagrangian coefficients α_i introduced to dualize.

Primal

$$\begin{aligned} \min_{\overrightarrow{w}, b, \overrightarrow{\xi}} \quad & \tfrac{1}{2}\|\overrightarrow{w}\|^2 + \tfrac{C}{2}\sum_{i=1}^{N} \xi_i^2 \\ s.t. \quad & \overrightarrow{w} \cdot \phi\left(\overrightarrow{x_i}\right) + b = y_i - \xi_i, \quad \forall i = 1, ..., N \end{aligned} \tag{1}$$

Lagrangian

$$\mathcal{L}\left(\overrightarrow{w}, b, \overrightarrow{\xi}, \overrightarrow{\alpha}\right) = \frac{1}{2}\|\overrightarrow{w}\|^2 + \frac{C}{2}\sum_{i=1}^{N} \xi_i^2 - \sum_{i=1}^{N} \alpha_i \left[\overrightarrow{w} \cdot \phi\left(\overrightarrow{x_i}\right) + b - y_i + \xi_i\right]$$

LS-SVMs dual

Characteristics

- Dual obtained by setting Lagrangian's derivatives to 0.
- Reduces to a KKT system of equations with $\tilde{K}_{ij} = k(\vec{x_i}, \vec{x_j}) + \delta_{ij}/C$.

Derivatives

$$\frac{\partial L}{\partial \vec{w}} = \vec{w} - \sum_{i=1}^{N} \alpha_i \phi\left(\vec{x_i}\right) = 0 \quad \Rightarrow \quad \vec{w} = \sum_{i=1}^{N} \alpha_i \phi\left(\vec{x_i}\right),$$

$$\frac{\partial L}{\partial b} = -\sum_{i=1}^{N} \alpha_i = 0 \quad \Rightarrow \quad \sum_{i=1}^{N} \alpha_i = 0,$$

$$\frac{\partial L}{\partial \xi} = C\xi_i - \alpha_i = 0 \quad \Rightarrow \quad \alpha_i = C\xi_i.$$

Dual Formulation

$$\left[\begin{array}{c|c} 0 & \vec{1}^T \\ \hline \vec{1} & \tilde{K} \end{array}\right] \left[\begin{array}{c} b \\ \hline \vec{\alpha} \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline \vec{y} \end{array}\right] \tag{2}$$

Some observations

- Final decision function of test point $\overrightarrow{x}$ given by $f(\overrightarrow{x}) = \text{sign}(\overrightarrow{w} \cdot \phi(\overrightarrow{x}) + b)$ (classification) or $f(\overrightarrow{x}) = \overrightarrow{w} \cdot \phi(\overrightarrow{x}) + b$ (regression).
- $\phi(\cdot)$ usually unknown, so change $\overrightarrow{w} \cdot \phi(\overrightarrow{x})$ by $\sum_i \alpha_i \phi(\overrightarrow{x_i}) \cdot \phi(\overrightarrow{x}) = \sum_i \alpha_i k(\overrightarrow{x_i}, \overrightarrow{x})$.
- Thus, test time proportional to number of points with $\alpha_i \neq 0$ (SVs).
- But $\alpha_i = C\xi_i$, so $\alpha_i = 0$ only when $\xi_i = 0$.
- That means when $\overrightarrow{x_i}$ lies exactly on its support hyperplane (classification) or when y_i is exactly the output estimation for $\overrightarrow{x_i}$ (regression).
- Very unlikely to happen, so in practice all patterns are SVs.
- Is there any way to reduce the number of SVs without degrading the model?

Previous approaches (1)

Pruning after training, then retraining

- Patterns with smallest $|\alpha_i|$ [1](Performance can drop quickly).
- Correctly classified patterns and furthest from boundary [2](Performance can drop quickly).
- Basing on the SMO algorithm while solving dual [3](Only homogeneous LS-SVM).
- Single pattern that introduces smallest error when omitted [4](Very costly computationally).

---

[1] J.A.K. Suykens, L. Lukas, J. Vandewalle. Sparse Approximation using Least Squares Support Vector Machines. Proceedings of the IEEE International Symposium on Circuits and Systems (ISCASS'2000), pp. 757–760, 2000.

[2] Y. Li, C. Lin, W. Zhang. Improved Sparse Least–Squares Support Vector Machine Classiffiers, Neurocomputing 69 (13–15), pp. 1655–1658, 2006.

[3] X. Zeng, X.W. Chen. SMO-based Pruning Methods for Sparse Least Squares Support Vector Machines. IEEE Transactions on Neural Networks 16 (6), pp. 1541–1546, 2005.

[4] B.J. De Kruif, T.J.A. De Vries. Pruning Error Minimization in Least–Squares Support Vector Machines. IEEE Transactions on Neural Networks 14 (3), pp. 696–702, 2003.

Previous approaches (2)

Enforcing before training

- Hierarchical model with L_1-norm minimization [5](Resulting problem difficult to solve).
- Searching for linearly independent subset of patterns [6](Not always good results).
- Fixing the size of the final model [7](Feature map only approximated).

[5] K. Pelckmans, J.A.K. Suykens, B. De Moor. Building Sparse Representations and Structure Determination on LS–SVM Substrates. Neurocomputing 64, pp. 137–159, 2005.

[6] J. Valyon, G. Horvath. A Sparse Least Squares Support Vector Machine Classiffier. Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN'04), pp. 543–548, 2004.

[7] K. De Brabanter, J. De Brabanter, J.A.K. Suykens, B. De Moor. Optimal Fixed–Size Kernel Models for Large Data Sets. Computational Statistics and Data Analysis 54(6), pp. 1484–1504, 2010.

L_0-norm

Basic facts

- Defined as limit of p-norms when $p \to 0$:

$$\|\overrightarrow{v}\|_0 = \lim_{p \to 0} \left(\sum_i |v_i|^p \right)^{\frac{1}{p}}$$

- Alternatively, $\|\overrightarrow{v}\|_0 = |\{v_i : v_i \neq 0\}|$.
- It counts number of non-zero elements, so minimizing it implies sparseness.
- In our context, we can think of minimizing $\|\overrightarrow{w}\|_0$ or $\|\overrightarrow{\alpha}\|_0$.
- The former corresponds to using minimal number of features for prediction, the latter to using minimal number of patterns.
- Problems: nonconvex, NP-hard to optimize [8].
- Solutions: approximations [9], iterative procedures.

---

[8] E. Amaldi, V. Kann. On the Approximability of Minimizing Nonzero Variables or Unsatisfied Relations in Linear Systems. Theoretical Computer Science 209 (1–2), pp. 237–260, 1998.

[9] J. Weston, A. Elisseeff, B. Schölkopf, M. Tipping. Use of the Zero Norm with Linear Models and Kernel Methods. Journal of Machine Learning Research 3, pp. 1439–1461, 2003.

An iterative approach for LS-SVMs (1)

Primal problem

$$\begin{aligned} \min_{\overrightarrow{v}^t, b^t, \overrightarrow{\xi}^t} \quad & \frac{1}{2}\sum_{i=1}^{N} \lambda_i^t (v_i^t)^2 + \frac{C}{2}\sum_{i=1}^{N} (\xi_i^t)^2 \\ s.t. \quad & \sum_j v_j^t k\left(\overrightarrow{x_i}, \overrightarrow{x_j}\right) + b^t = y_i - \xi_i^t, \quad \forall i = 1, ..., N. \end{aligned} \tag{3}$$

Observations

- Implicit vector $\overrightarrow{w}^t = \sum_j v_j^t \phi(\overrightarrow{x_j})$ still underlying in constraints.
- Regularization not on $\|\overrightarrow{w}^t\|^2$, but on the weighted L_2-norm of coefficients v_i^t.
- These v_i^t are no longer Lagrangian coefficients.
- Weights of regularization given by λ_i^t.
- Solution of (3) gives $\overrightarrow{v}^{t+1}, b^{t+1}$.
- Under probabilistic framework, can be shown that $\lim_{t\to\infty} \overrightarrow{v}^t = \overrightarrow{v}^*$ and that $\lim_{t\to\infty} \sum_i \lambda_i^t (v_i^t)^2 = \|\overrightarrow{v}^*\|_0$, provided we update $\lambda_i^{t+1} = \frac{1}{(v_i^{t+1})^2}$ [10].

[10] K. Huang, D. Zheng, J. Sun, Y. Hotta, K. Fujimoto, S. Naoi. Sparse Learning for Support Vector Classiffication. Pattern Recognition Letters 31 (13), pp. 1944–1951, 2010.

An iterative approach for LS-SVMs (2)

Algorithm (IS-LSSVM)

1. Compute the LS-SVM solution $\overrightarrow{\alpha}$ for given sample, kernel and C using (2).
2. Set $t \leftarrow 0$, $\overrightarrow{\lambda}^0 \leftarrow \alpha$ and $\overrightarrow{v}^0 \leftarrow \overrightarrow{\infty}$.
3. Solve problem (3) to give $\overrightarrow{v}^{t+1}$ and b^{t+1}.
4. Update $\lambda_i^{t+1} \leftarrow \frac{1}{(v_i^{t+1})^2}, i = 1, \ldots, N$.
5. Set $t \leftarrow t + 1$ and go back to 3 until convergence.
6. Return model (v^t, b^t).

Remarks

- (3) can be solved either in primal or in dual (more on this later).
- Convergence criterion based on similarity between $\overrightarrow{v}^t$ and $\overrightarrow{v}^{t+1}$, when $\frac{1}{N} \| \overrightarrow{v}^t - \overrightarrow{v}^{t+1} \|^2 \leq \epsilon$.
- Final result dependent on choice of $\overrightarrow{\lambda}^0$: only local optimum attained.

Solution of the problem

In the primal

- Writing (3) in matrix notation and substituting ξ^t yields

$$\min_{\vec{v}^t, b^t} \quad \frac{1}{2}(\vec{v}^t)^T \operatorname{diag}(\vec{\lambda}^t)\vec{v}^t + \frac{C}{2}(\vec{y} - K\vec{v}^t - b^t\vec{1})^T(\vec{y} - K\vec{v}^t - b^t\vec{1})$$

- Differentiating w.r.t. $\vec{v}^t, b^t$ and equalling to 0 produces system

$$\left[\begin{array}{c|c} N & \vec{1}^T K \\ \hline K^T\vec{1} & K^T K + \frac{\operatorname{diag}(\vec{\lambda}^t)}{C} \end{array}\right] \left[\begin{array}{c} b^t \\ \hline \vec{v}^t \end{array}\right] = \left[\begin{array}{c} \vec{y}^T\vec{1} \\ \hline K^T\vec{y} \end{array}\right] \tag{4}$$

In the dual

- Differentiating Lagrangian and equalling to 0 produces another system

$$\left[\begin{array}{c|c} 0 & \vec{1}^T \\ \hline \vec{1} & K^T \operatorname{diag}(\vec{\lambda}^t)^{-1} K + \frac{I}{C} \end{array}\right] \left[\begin{array}{c} b^t \\ \hline \vec{\beta}^t \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline \vec{y} \end{array}\right] \tag{5}$$

- Same as (2), but with kernel switched to

$$\widehat{k}(X_i, X_j) = \sum_m \frac{k(X_i, X_m)\, k(X_j, X_m)}{\lambda_m} + \frac{\delta_{ij}}{C}$$

- Primal variable recovered with $\vec{v}^t = \operatorname{diag}(\vec{\lambda}^t)^{-1} K\beta^t$.

Experimental framework

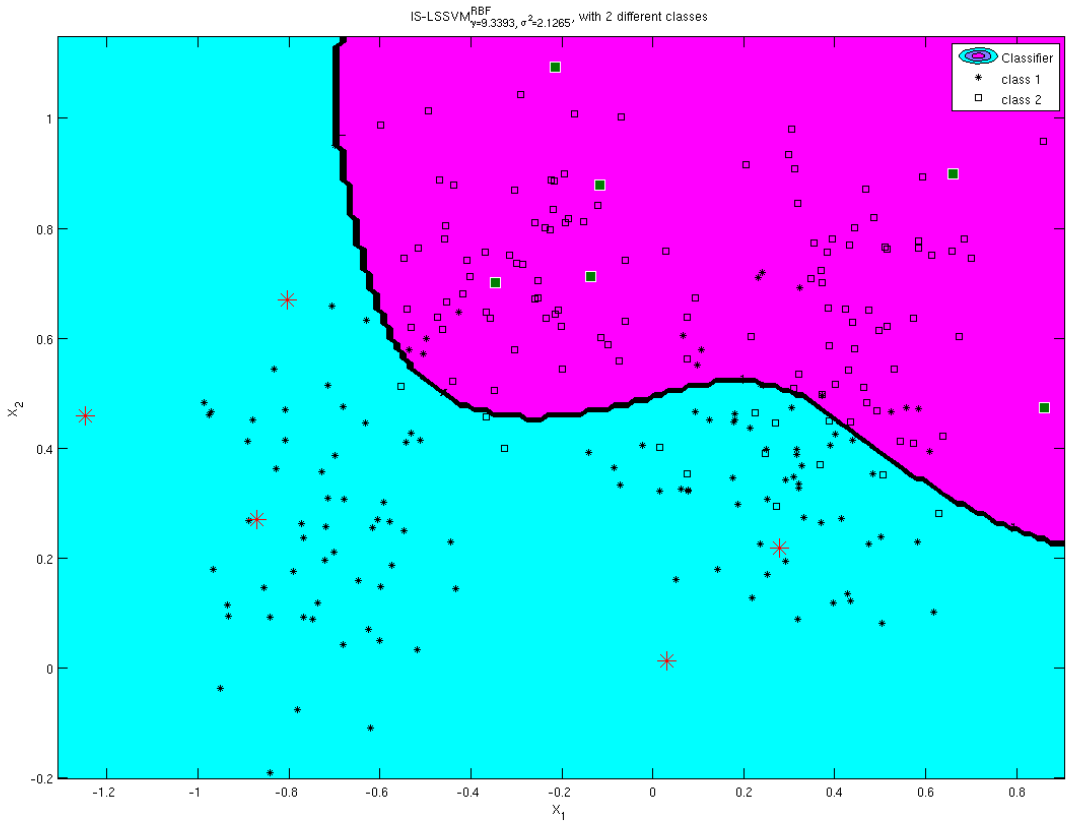
Datasets

- Well-known datasets for classification (*Ripley*, *Fourclass*) and regression (*Motorcycle*, *Fossil*).
- All bidimensional for plotting purposes.

Setting

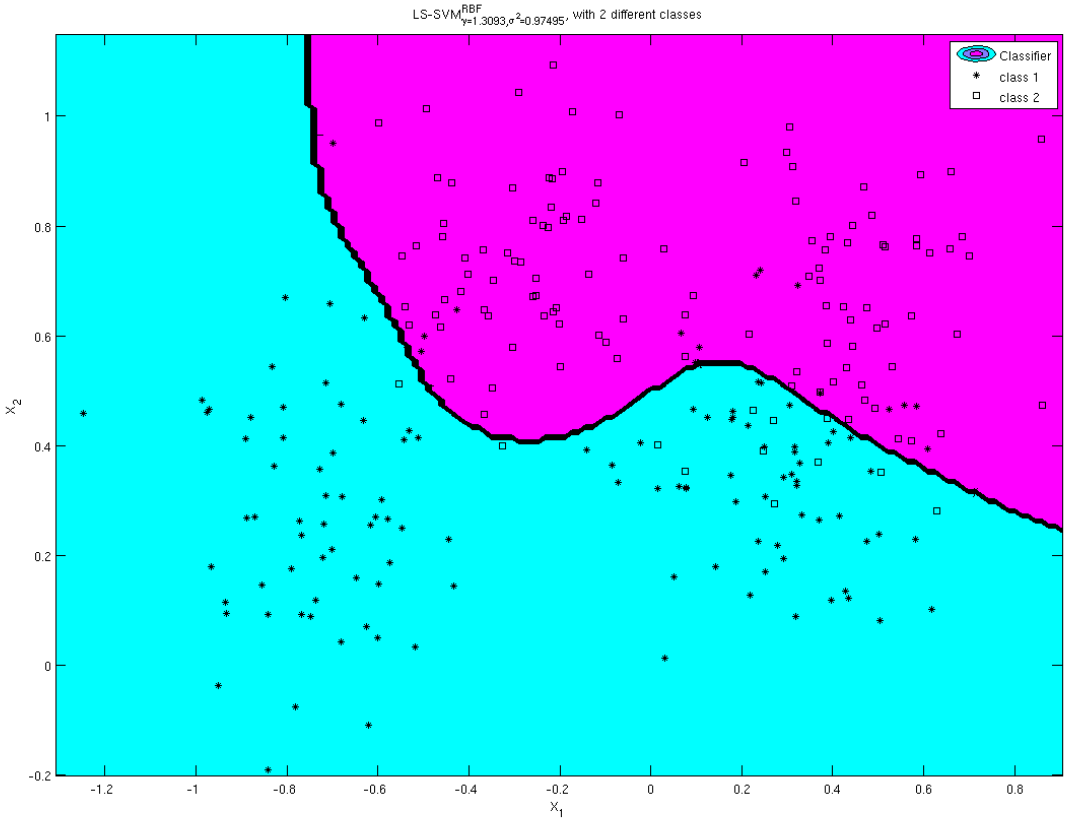
- RBF kernel used for all experiments.
- Stopping criterion with $\epsilon = 10^{-4}$.
- Hyperparameters C and σ for both ISLS-SVM and LS-SVM tuned with CSA (broad search) + simplex (fine search).
- CSA uses 10-fold CV as score function.
- Implemented in LS-SVM Matlab Toolbox (future version 2.0).

Ripley dataset: ISLS-SVM



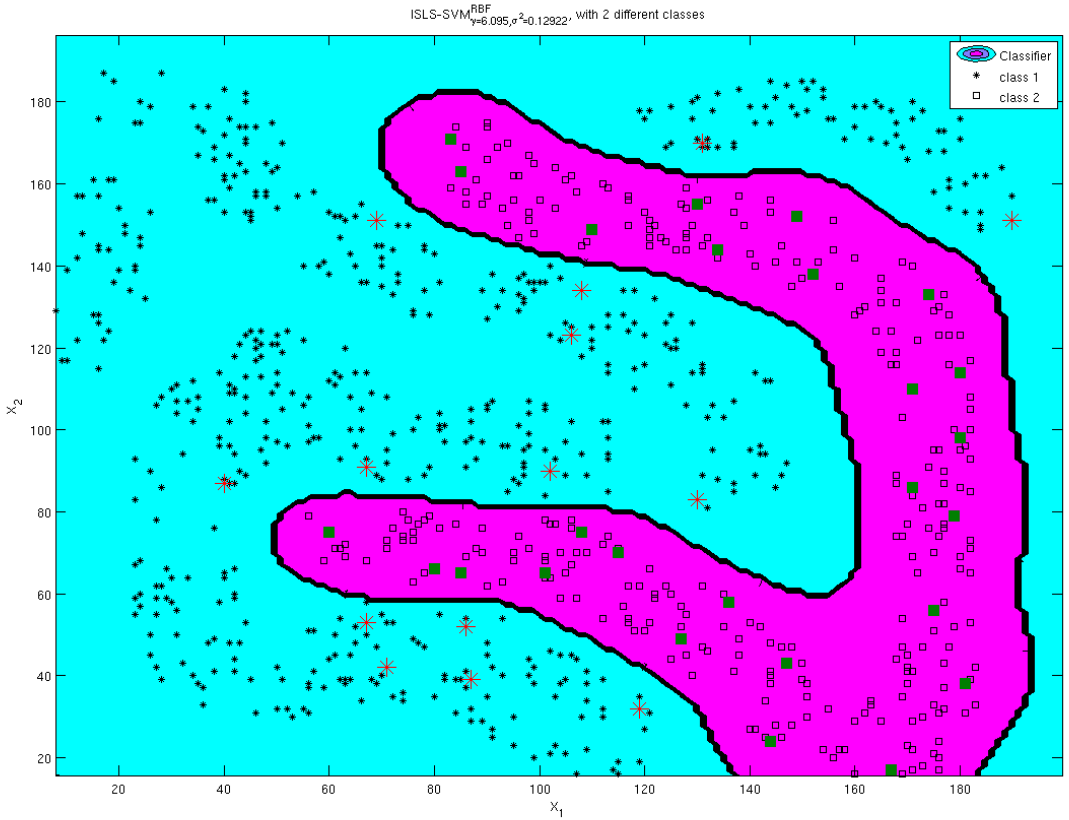
- 11 SVs out of 250 patterns.
- Smooth decision border.

Ripley dataset: LS-SVM



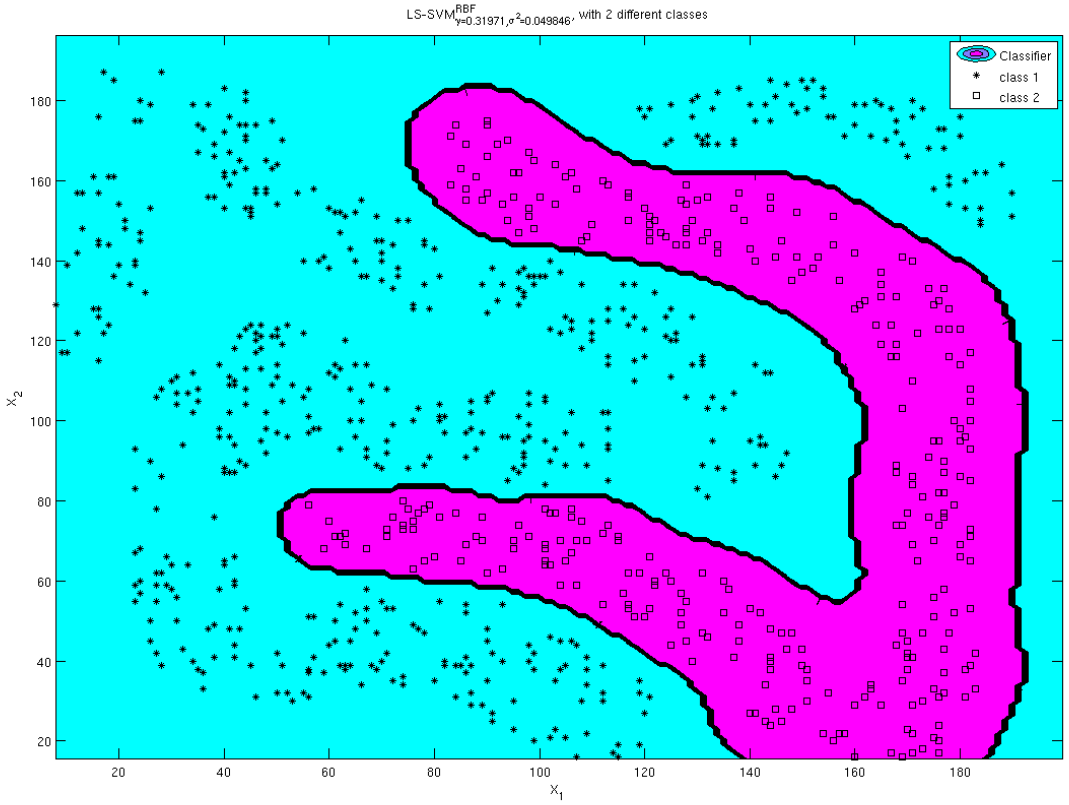
- All 250 patterns are SVs.
- Similar decision function, a bit fitter to the data.

Fourclass dataset: ISLS-SVM



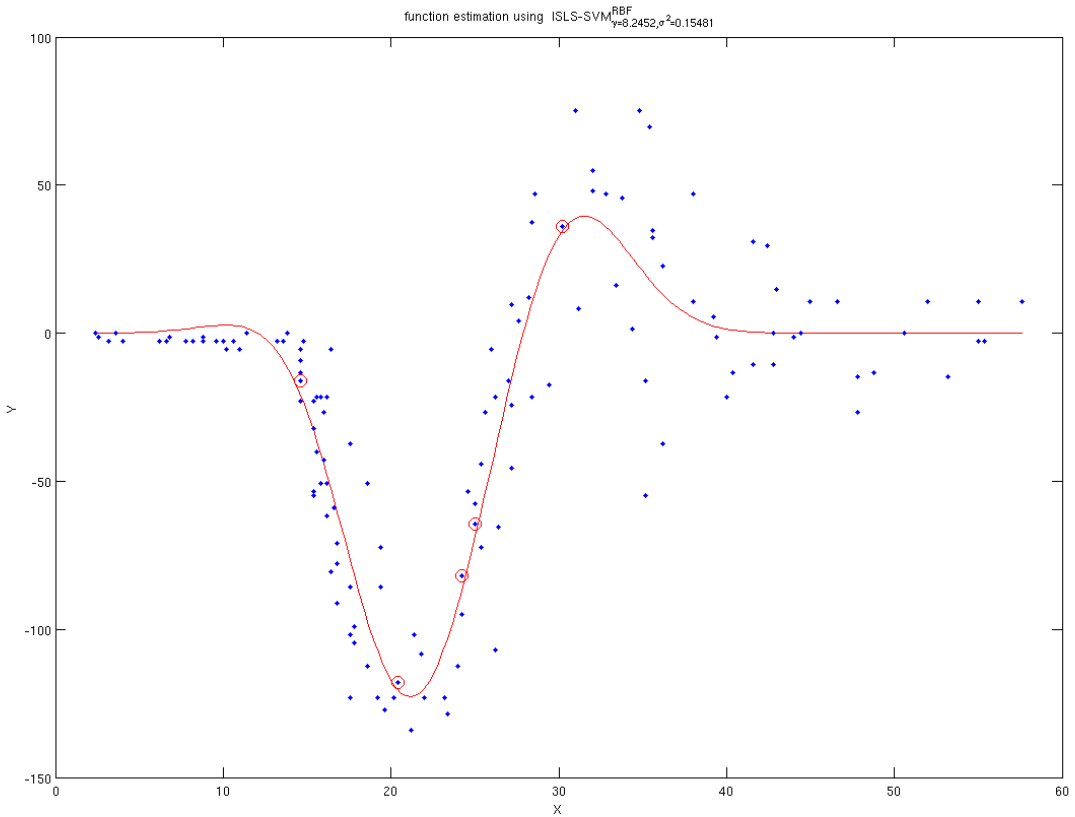
- 40 SVs out of 862 patterns, spread close to decision border.
- Easy problem (no error), but highly nonlinear.

Fourclass dataset: LS-SVM



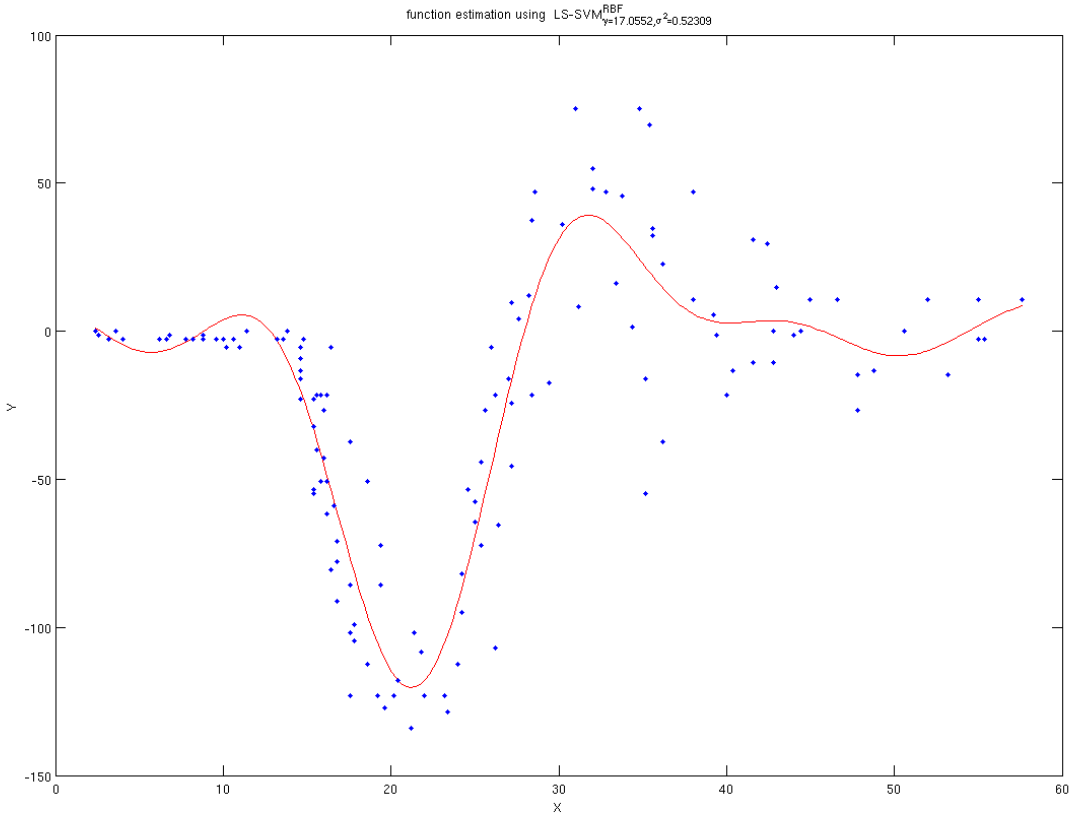
- All 862 patterns are SVs.
- Also perfect performance and very similar decision function.

Motorcycle dataset: ISLS-SVM



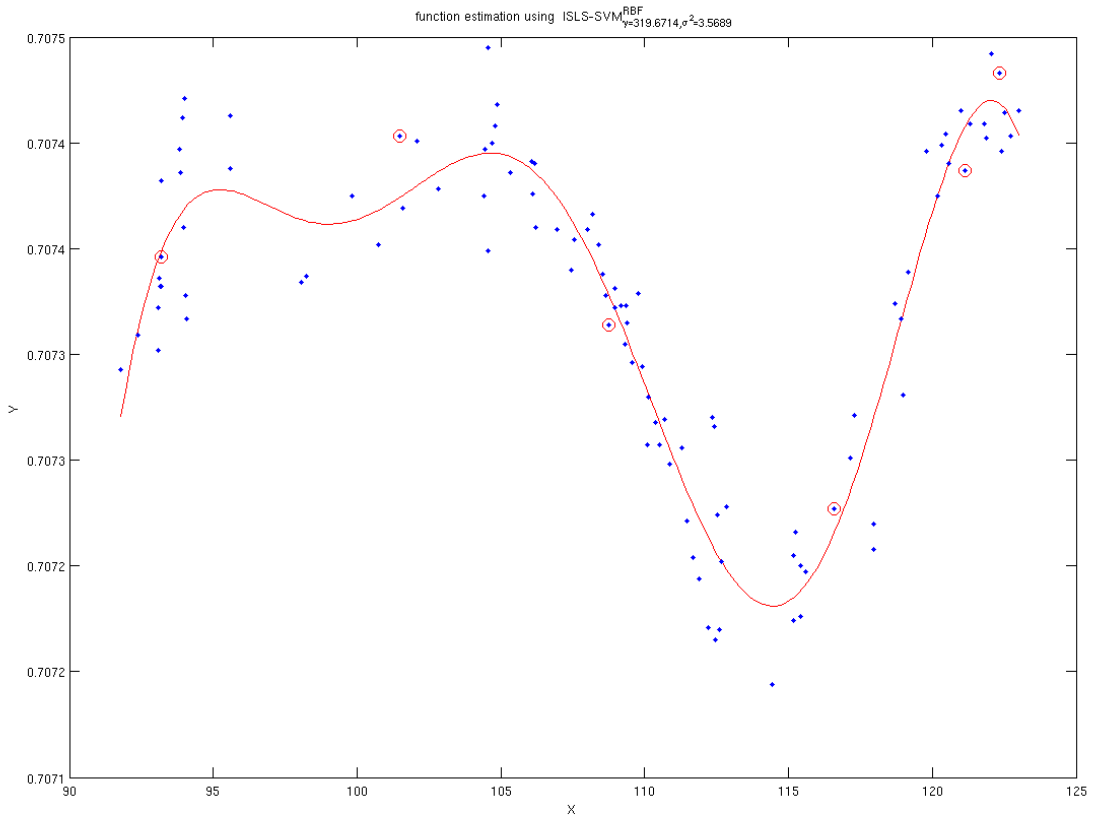
- 5 SVs out of 133 patterns.
- Difficult problem and highly nonlinear.

Motorcycle dataset: LS-SVM



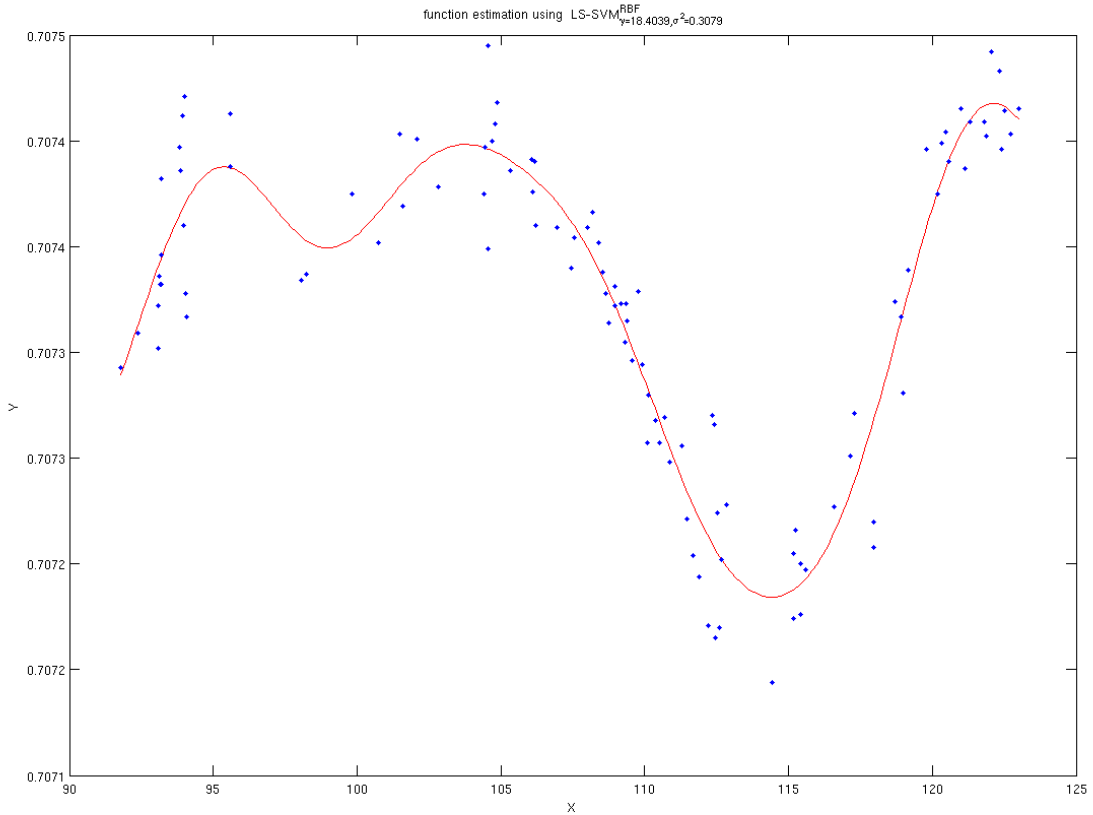
- All 133 patterns are SVs.
- Fitter to data at end, more oscillating at beginning.

Fossil dataset: ISLS-SVM



- 6 SVs out of 106 patterns.
- Highly nonlinear problem.

Fossil dataset: LS-SVM



- All 106 patterns are SVs.
- More oscillating at beginning.

Summary

Pros

- Straightforward algorithm to find sparse LS-SVM models.
- Small number of iterations (10-20 for most cases).
- Final models an order of magnitude (10-30x) sparser than LS-SVMs.
- Performance only degrades slightly.
- Applicable to other formulations e.g. Multiclass LS-SVMs.

Cons

- Only local minimum attained, basing on initial weights given by LS-SVM.
- Assumed model $W = \sum_i v_i \phi(X_i)$.
- Computationally costly (each iteration is $\mathcal{O}(N^3)$).
- Occasional numerical instabilities.

(Possible) solutions (I)

Numerical instabilities

- Solving (4) instead of (5).
- System looks more robust since λ and C are in the same term.

Local minimum

- Apparently no solution, since finding global minimum is NP-hard.

Computational cost

- $\mathcal{O}(N^3)$ cost cause every system solved with operator $\backslash$.
- Too costly for medium or large-scale problems.
- Possible to reduce to $\mathcal{O}(N^2)$ using the SMO algorithm in the dual [11].
- This also allows for caching kernel matrix, so large-scale can be issued.

[11] J. López, J.A.K. Suykens. First and Second Order SMO Algorithms for Large Scale LS–SVM Training. Internal Report 09–179, ESAT–SISTA, K.U. Leuven, 2009.

(Possible) solutions (II)

Assumption on $\overrightarrow{w}$

- Is it possible to remove assumption on $\overrightarrow{w}$ in (3)?

$$\min_{\overrightarrow{w}^t, b^t, \overrightarrow{\xi}^t} \quad \frac{1}{2}\sum_{i=1}^{n_F} \lambda_i^t (w_i^t)^2 + \frac{C}{2}\sum_{i=1}^{N} (\xi_i^t)^2$$

$$s.t. \qquad \overrightarrow{w}^t \cdot \phi(\overrightarrow{x_i}) + b^t = y_i - \xi_i^t, \qquad \forall i = 1, ..., N.$$

- Solving in the primal we get

$$\left[\begin{array}{c|c} N & \overrightarrow{1}^T \Phi \\ \hline \Phi^T \overrightarrow{1} & \Phi^T \Phi + \frac{\mathrm{diag}(\overrightarrow{\lambda}^t)}{C} \end{array}\right] \left[\begin{array}{c} b^t \\ \hline \overrightarrow{w}^t \end{array}\right] = \left[\begin{array}{c} \overrightarrow{y}^T \overrightarrow{1} \\ \hline \Phi^T \overrightarrow{y} \end{array}\right]$$

- Same as (4), but with matrix $\Phi^T = (\phi(X_1) \ldots \phi(X_N))$.
- Intractable cause Φ usually unknown, unless we have an estimate $\widehat{\Phi}$.
- Solving in the dual we get

$$\left[\begin{array}{c|c} 0 & \overrightarrow{1}^T \\ \hline \overrightarrow{1} & \Phi\,\mathrm{diag}(\overrightarrow{\lambda}^t)^{-1}\Phi^T + \frac{I}{C} \end{array}\right] \left[\begin{array}{c} b^t \\ \hline \overrightarrow{\beta}^t \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline \overrightarrow{y} \end{array}\right]$$

- Same as (5), but with matrix $\Phi^T\,\mathrm{diag}(\overrightarrow{\lambda}^t)^{-1}\Phi$ instead of K.
- Intractable cause $\Phi\Phi^T = K$, but $\Phi\,\mathrm{diag}(\overrightarrow{\lambda}^t)^{-1}\Phi^T$ usually unknown, unless we have an estimate $\widehat{\Phi}$.

Farewell

Thank you for your attention!