ν-SVM

The hyperparameters of SVM

$$\min_{w,\xi}\left(\frac{1}{2}\|w\|^2+\frac{C}{M}\sum_{j=1}^{M}\xi_j\right)$$

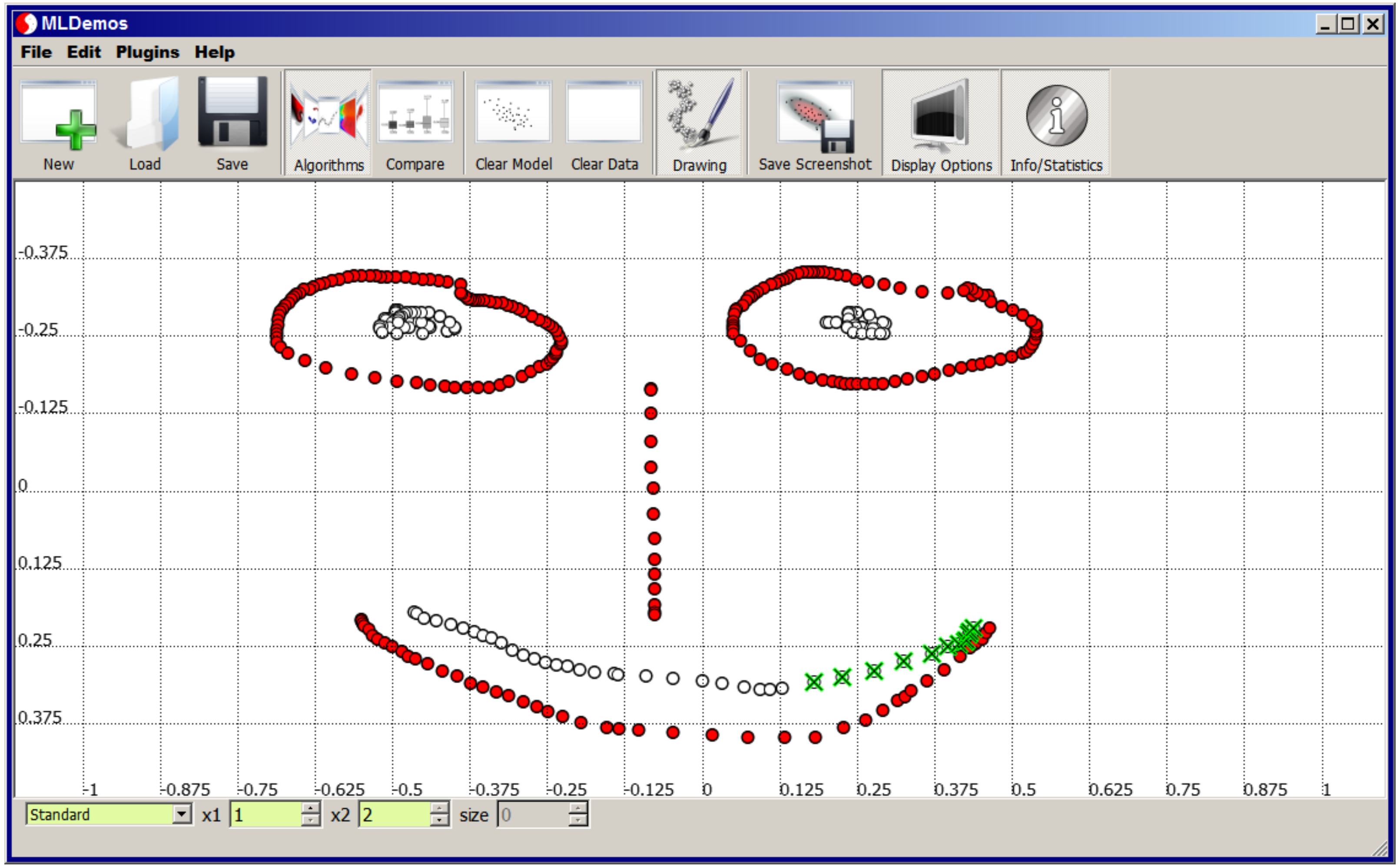
C that determines the costs associated to incorrectly classifying datapoints is an open parameter of the optimization function

u.c.

$$y^j\left(w^T\cdot x^j+b\right)\geq 1-\xi_j,$$

$$\xi_j\geq 0 \qquad \forall\ \text{j=1,...M}$$

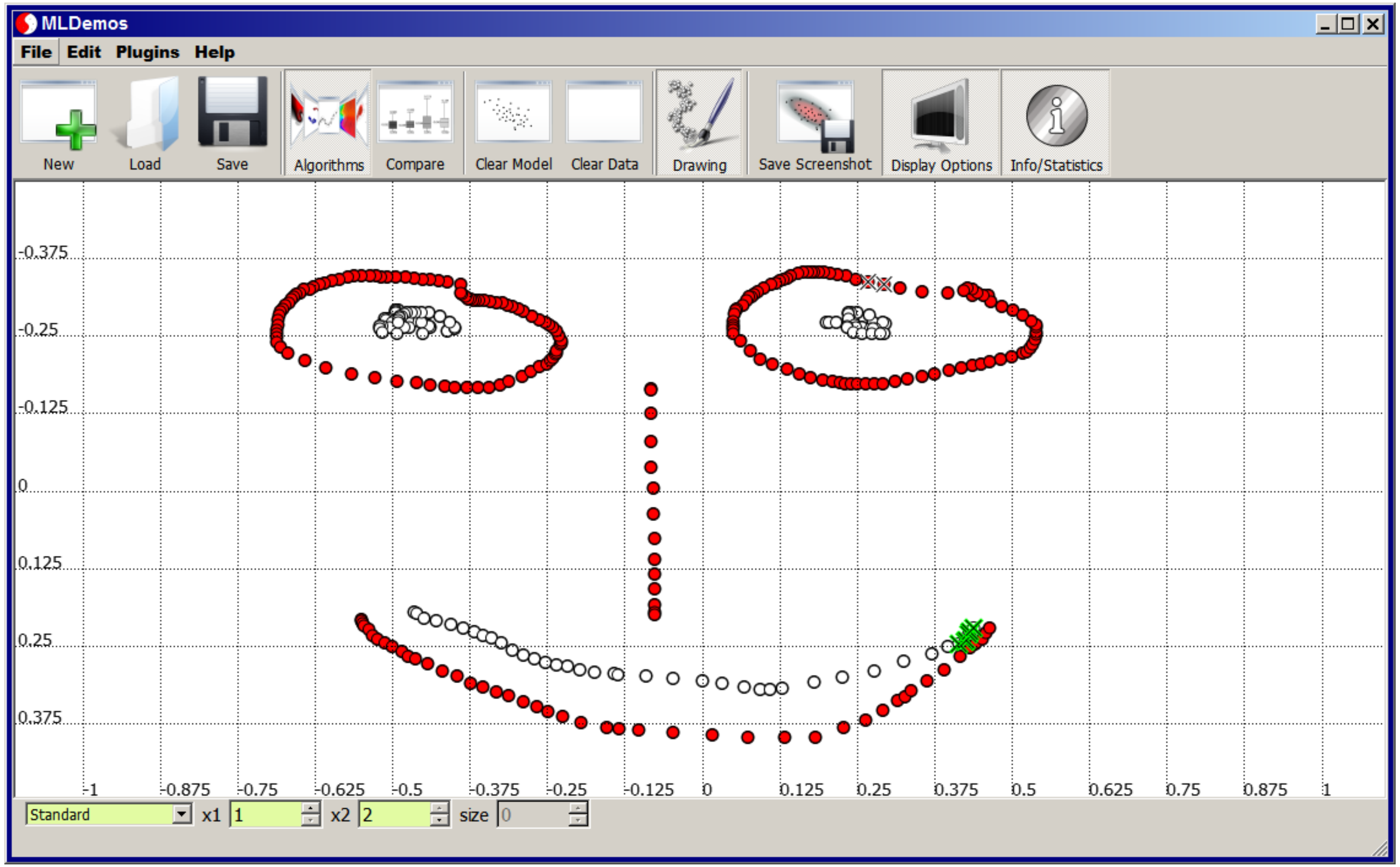
Effect of the penalty factor C



x1 1 x2 2 size 0

RBF kernel width=0.20; C=1000; several misclassified datapoints

Effect of the penalty factor C



RBF kernel width=0.20; C=2000; fewer misclassified datapoints

Hyperparameters for SVM

The original objective function:

$$\min_{w,\xi}\left(\frac{1}{2}\|w\|^2+\frac{C}{M}\sum_{j=1}^{M}\xi_j\right)$$

Determining C may be difficult in practice

ν-SVM

Introduce a new variable ρ to control for the lower bound on $\|w\|$ and add a hyperparameter $0 \leq \nu \leq 1$ to control for its effect in the objective function.

The optimization problem becomes:

$$\min_{w,\xi,\rho} \left(\|w\|^2 - \enclose{circle}{\nu\rho} + \frac{1}{M} \sum_{i=1}^{M} \xi_i \right),$$

$$\text{subject to} \quad y^i \left(\left\langle w, x^i \right\rangle + b \right) \geq \enclose{circle}{\rho} - \xi_i$$

$$\text{and} \quad \xi_i \geq 0, \enclose{circle}{\rho \geq 0}.$$

ν-SVM

ν is an upper bound on the fraction of margin error (i.e. the number of datapoints misclassified in the margin)

ν is a lower bound on the ratio: support vectors / number of datapoints.

$$\nu \leq \frac{p}{M}, \qquad p: \text{number of SV}$$

ν-SVM: Exercise

Show that ν is a lower bound on the ratio:

$$\nu \le \frac{p}{M}, \qquad p: \text{number of SV}, \; M: \text{number of datapoints}$$

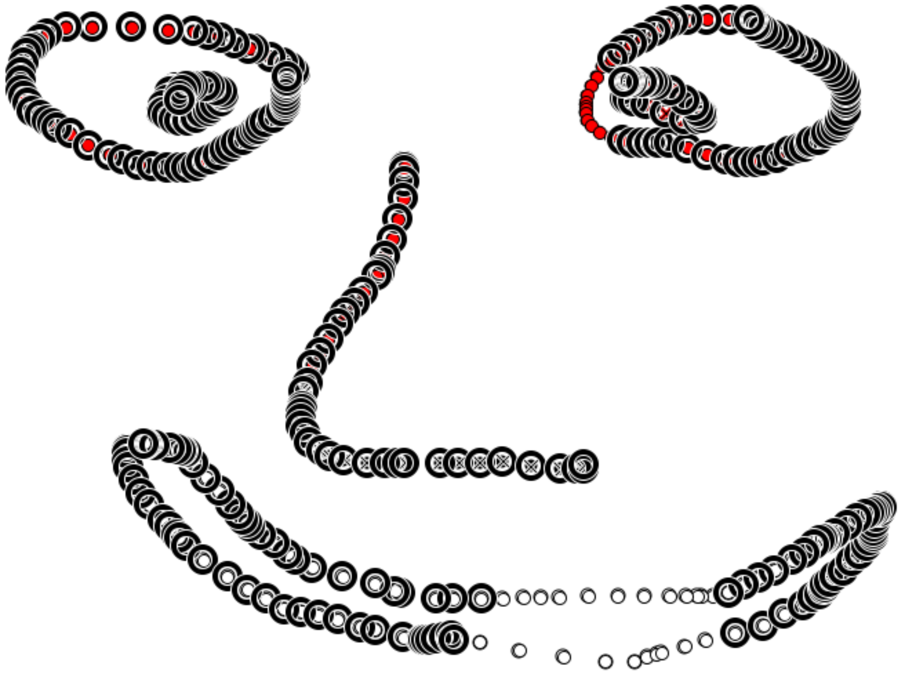
Hint: The dual problem is

$$\min_{\alpha_1,\ldots.\alpha_M} \left(L\left(\alpha_1,\ldots.\alpha_M\right) = -\frac{1}{2}\sum_{i,j=1}^{M} \alpha_i \alpha_j k\left(x^i, x^j\right) \right)$$

subject to

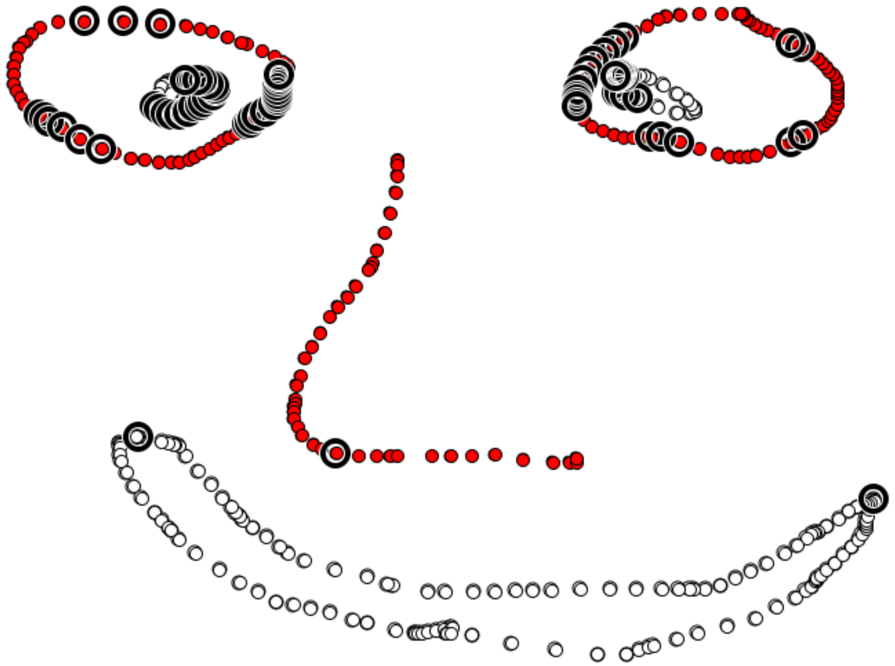
$$\sum_{i=1}^{M} \alpha_i y_i = 0; \quad 0 \le \alpha_i \le 1/M; \qquad \sum_{i=1}^{M} \alpha_i \ge \nu$$

ν-SVM: Example of effect of choice of ν



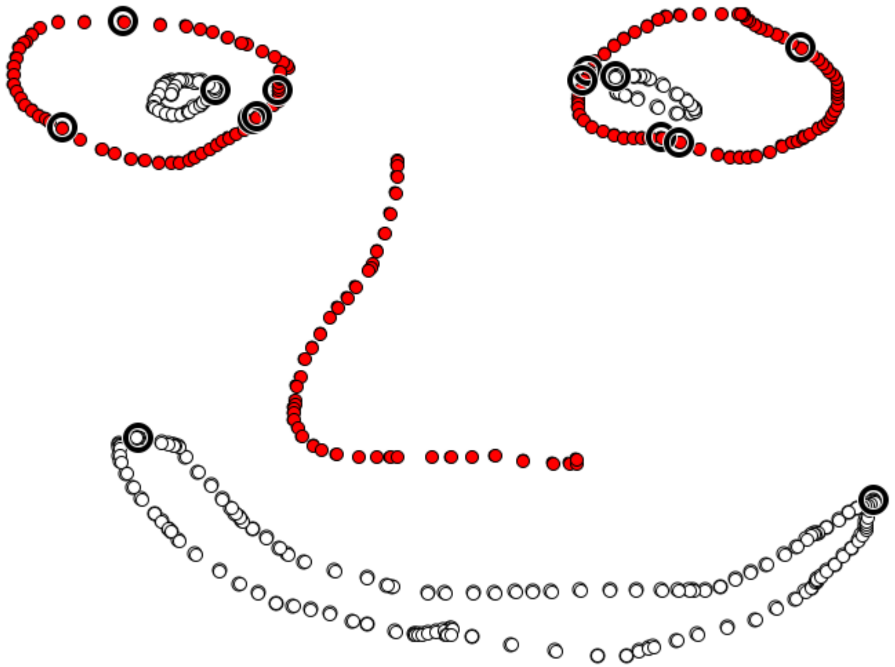
Increase in the number of SV-s with ν=0.9

ν-SVM: Example of effect of choice of ν



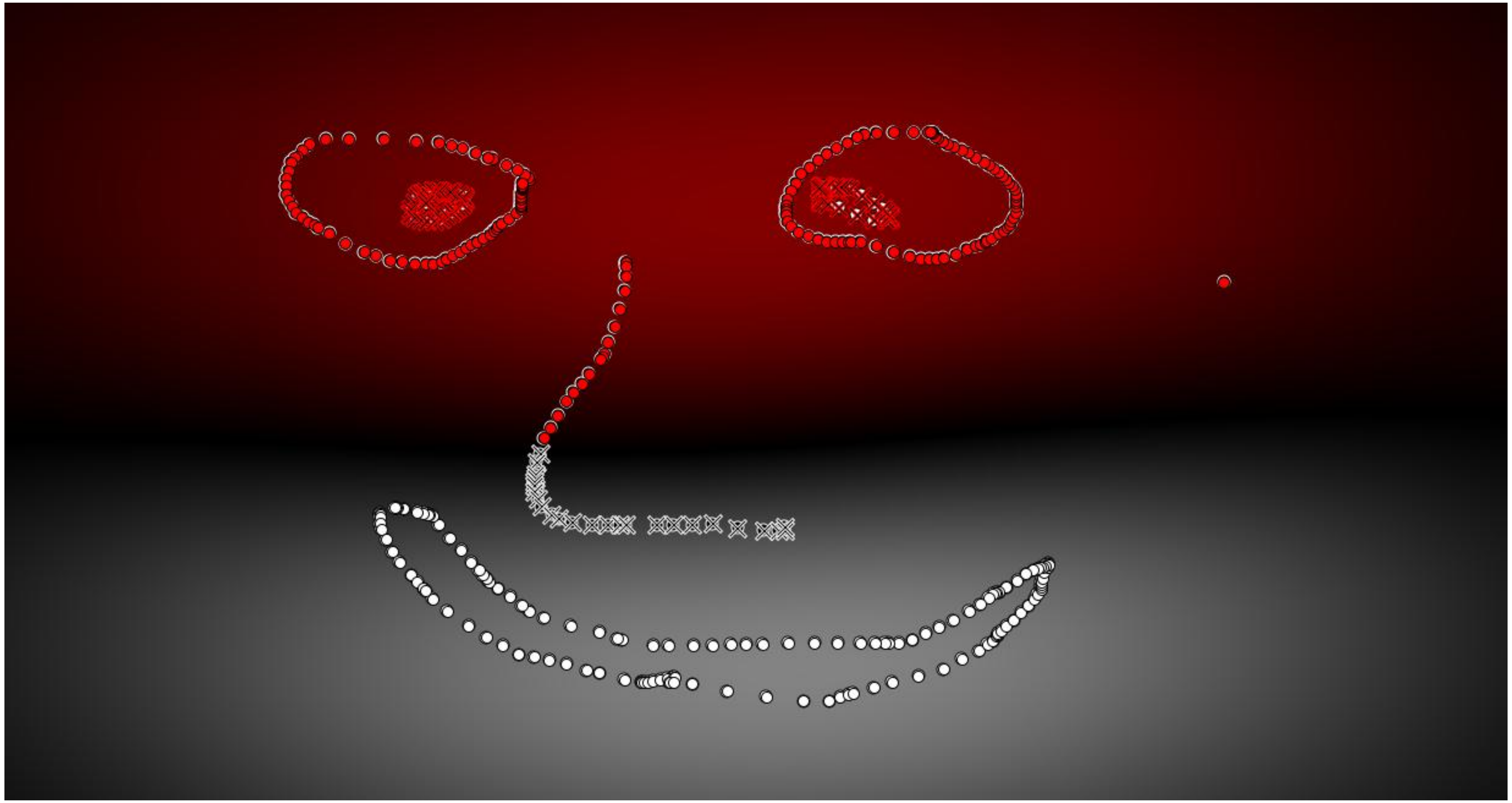
Increase in the number of SV-s with ν=0.2

ν-SVM: Example of effect of choice of ν



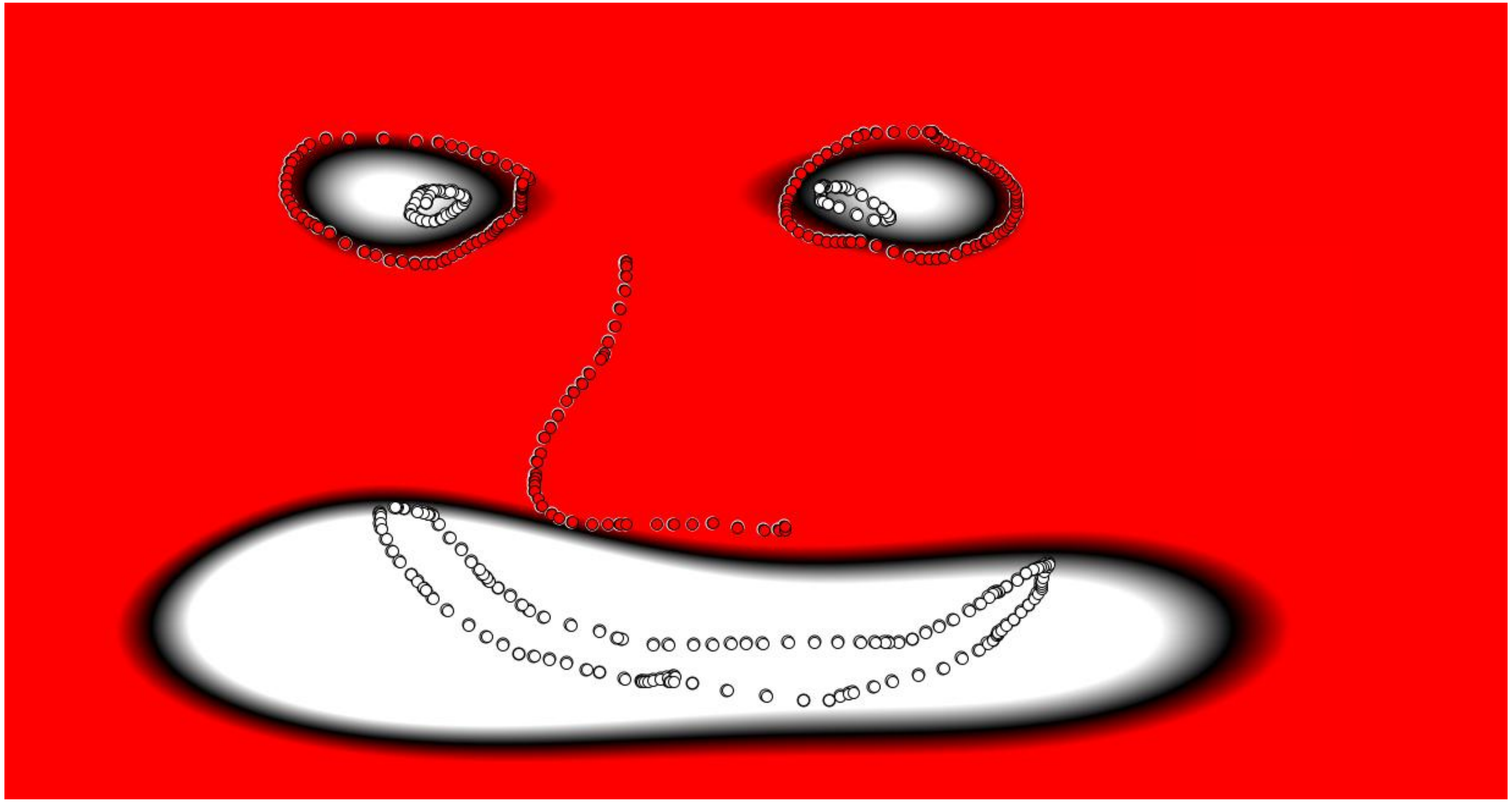
ν-svm ν=0.001, rbf kernel width 0.1

ν-SVM: Example of effect of choice of ν



Increase in the error with ν=0.9

ν-SVM: Example of effect of choice of ν

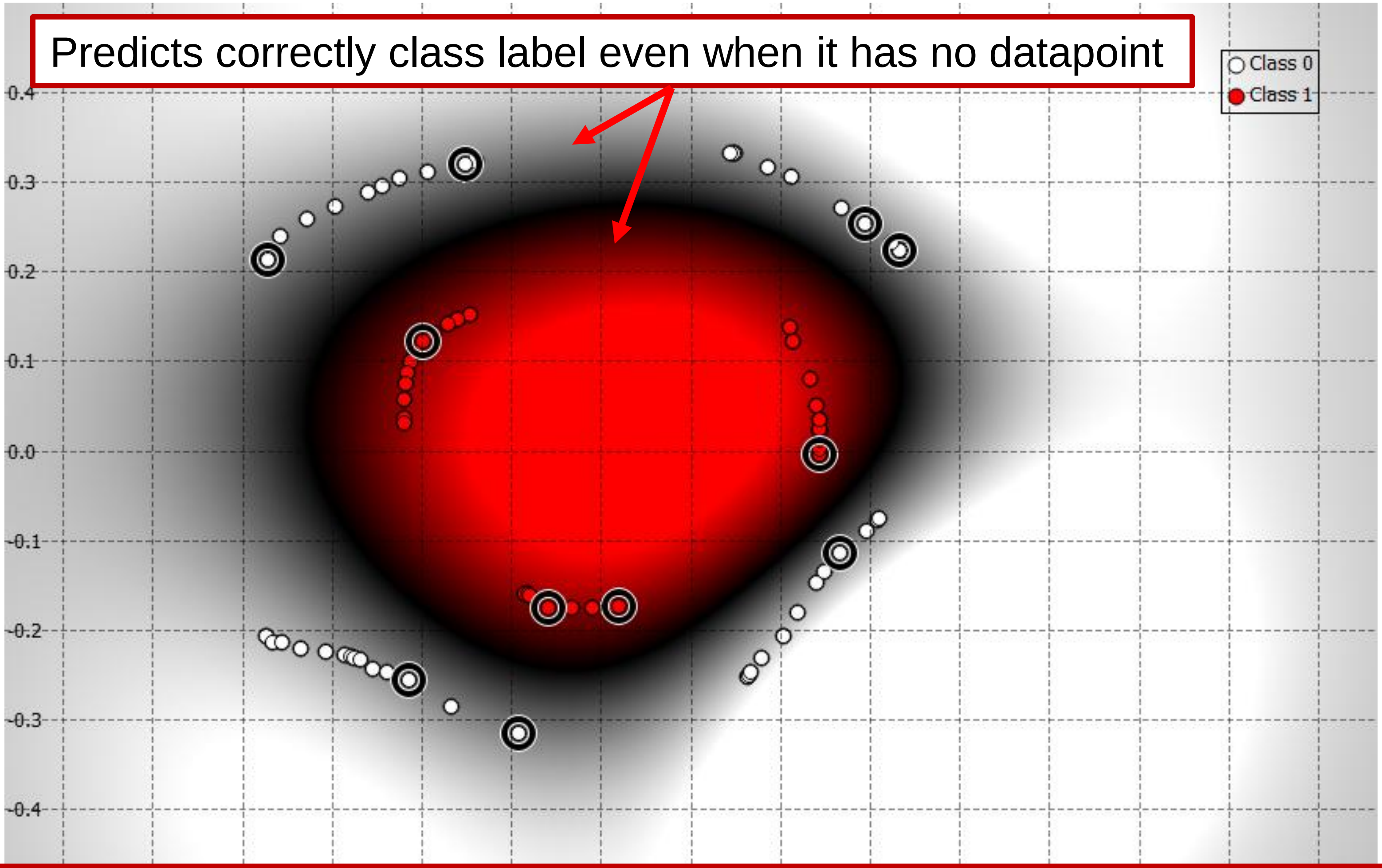


Good classification with ν=0.2

Relevance Vector Machine (RVM)
(sparse SVM)

see supplement (Tipping, IJML 2001) – sparse classification technique

SVM Limitation

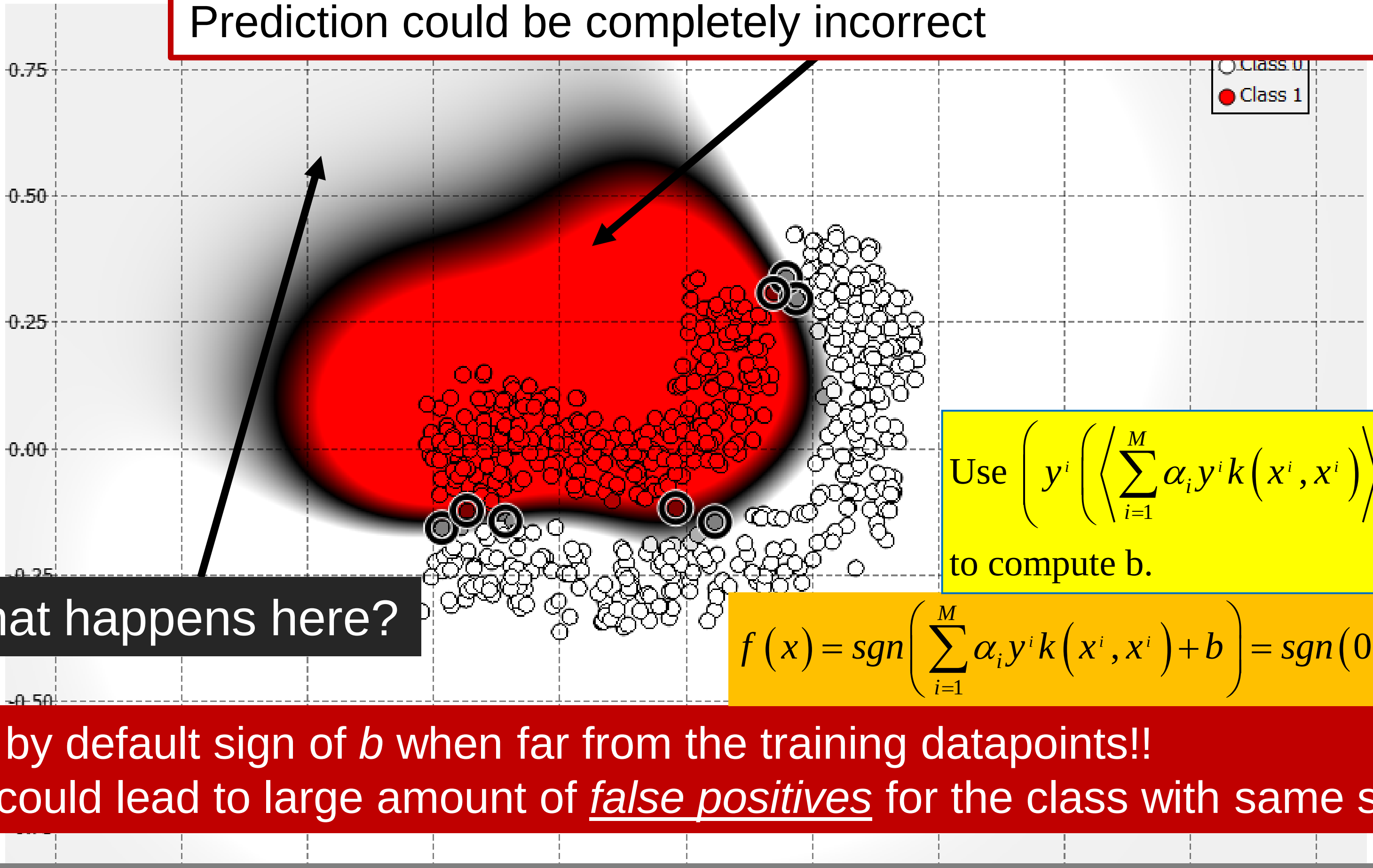


Predicts correctly class label even when it has no datapoint

SVM does not entail a notion of confidence! (no notion of likelihood)
→ You cannot tell if prediction is correct or not!
→ Crossvalidation is hence crucial, but still no guarantee that prediction is correct

SVM – no confidence in prediction

Generalization: Predicts class label even when it has no datapoint
Prediction could be completely incorrect



What happens here?

Use $\left(y^i\left(\left\langle\sum_{i=1}^{M}\alpha_i y^i k\left(x^i,x^i\right)\right\rangle+b\right)-1\right)=0$ to compute b.

$$f(x)=sgn\left(\sum_{i=1}^{M}\alpha_i y^i k\left(x^i,x^i\right)+b\right)=sgn(0+b)=sgn(b)$$

Predicts by default sign of *b* when far from the training datapoints!!
→ This could lead to large amount of *false positives* for the class with same sign as *b!*

Doing crossvalidation would not prevent this effect as it would use only points at your disposal. It does not test for unseen datapoints.

SVM – confidence in prediction

- False positives must be treated with care.

- Imagine you classify images of cancer tissue and you want to predict if the tissue has a tumor (positive class) or no tumor (negative class); **you cannot afford false positive for the negative class.**

- To prevent this to happen, you should:
  - Verify that the sign of *b* is not the sign of the class you care about.
  - Run crossvalidation by generating a testing set from points never seen – far from your training set.

Relevance Vector Machine (RVM)

RVM was offered to address three shortcomings of standard SVM:

1) Even though SVM usually results in a relatively small number of support vectors compared to total number of data-points, nothing ensures that a sparse solution is obtained. The number of SV tends to grow “linearly” with the number of training datapoints.

2) Unlike other bayesian techniques (e.g. GMM classification using naïve Bayes), SVM’s prediction are not accompanied by a metric measuring the confidence of the model’s prediction.

3) SVM requires also to find hyper-parameters (C, ν) and to have special form for the basis function (the kernel must satisfy the Mercer conditions).

RVM relaxes assumption 3 and takes a Bayesian approach to estimate the model’s parameters. The Bayesian framework captures the uncertainty of the prediction. It also results in a sparse version of classical SVM.

Relevance Vector Machine

Start from the solution of SVM (dropping the sign function – provides regression solution first, see slides on non-linear regression)

$$y(x) = f(x) = \sum_{i=1}^{M} \alpha_i \underbrace{k\left(x, x^i\right)}_{\psi_i(x)} + b$$

Rewrite the solution of SVM as a linear combination over M basis functions

A sparse solution has a majority of entries with alpha zero.

$$\alpha = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ . \\ . \\ . \\ . \\ \alpha_M \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ 0 \\ \alpha_3 \\ 0 \\ 0 \\ . \\ \alpha_M \end{bmatrix}$$

In the (binary) classification case, $y \in [0;1]$.

In the regression case, $y \in \mathbb{R}$.

Relevance Vector Machine

Rewrite the solution of SVR in a compact form such that the problem is linear in the parameters:

$$y(x) = f(x) = \sum_{i=1}^{M} \alpha_i \underbrace{k\left(x, x^i\right)}_{\psi_i(x)} + \boxed{\underset{=\alpha_0}{b}}$$

$$y(x) = \alpha^T \Psi(x), \qquad \Psi(x) = \left[\psi_0(x)\ \psi_1(x) ....\psi_M(x)\right]^T, \ \boxed{\psi_0(x) = 1.}$$

The problem consists in finding the parameters α.
The problem is made easy in that it is *linear in the parameters.*

Relevance Vector Machine

Rewrite the solution of SVR in a compact form such that the problem is linear in the parameters:

$$y(x) = f(x) = \sum_{i=1}^{M} \alpha_i \underbrace{k(x, x^i)}_{\psi_i(x)} + \underbrace{b}_{=\alpha_0}$$

$$y(x) = \alpha^T \Psi(x), \quad \Psi(x) = \left[\psi_0(x)\ \psi_1(x) ....\psi_M(x)\right]^T, \ \psi_0(x) = 1.$$

Take a Bayesian approach and assume that all samples y_i

are i.i.d and that they are measurements of the real value $\alpha^T \Psi(x^i)$

subjected to white noise ε, i.e.:

$$y_i(x) = \alpha^T \Psi(x^i) + \varepsilon, \quad \varepsilon \sim N(0, \sigma_\varepsilon)$$

Relevance Vector Machine

Since the measurement are i.i.d, the likelihood of the model is given by:

$$L(\alpha)=\prod_{i=1}^{M} p\left(y_i \mid x^i;\alpha,\sigma_\varepsilon\right)\sim\prod_{i=1}^{M} e^{-\frac{1}{\sigma_\varepsilon^2}\left\|y_i-\alpha\Psi\left(x^i\right)\right\|^2}$$

Question: What is the result if we estimate the α parameters through maximum likelihood?

Doing maximum likelihood would lead to overfitting, as we have as many parameters as datapoints (or more if we consider the variance of the noise too).

E.g. with rbf kernel for basis functions, all alphas are +/-1 putting one rbf function on each datapoint.

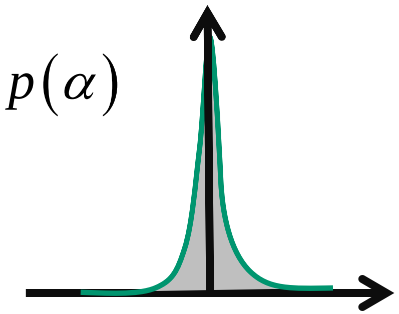
→ idea: approximate the distribution of the α with a probability density function which reduces the number of parameters to estimate.

Relevance Vector Machine

$$y(x) = \alpha^T \Psi(x), \qquad \Psi(x) = \left[\psi_0(x)\ \psi_1(x) \ldots\ldots \psi_M(x)\right]^T,\ \psi_0(x) = 1.$$

Introduces a prior on the distribution of the parameters, i.e. $p(\alpha)$, to prevent them from taking arbitrary values.

Sparsity is obtained when the distribution is sharply peaked at zero, e.g. $E\{p(\alpha)\} \sim 0$ and $\mathrm{var}\{p(\alpha)\} << 1$



Relevance Vector Machine

Prior distribution is zero-mean with variance σ.

σ is a set of M parameters that controls for the breadth of values taken by the α:

$$p(\alpha_i) = \sim N(\alpha_i; 0, \sigma_i)$$

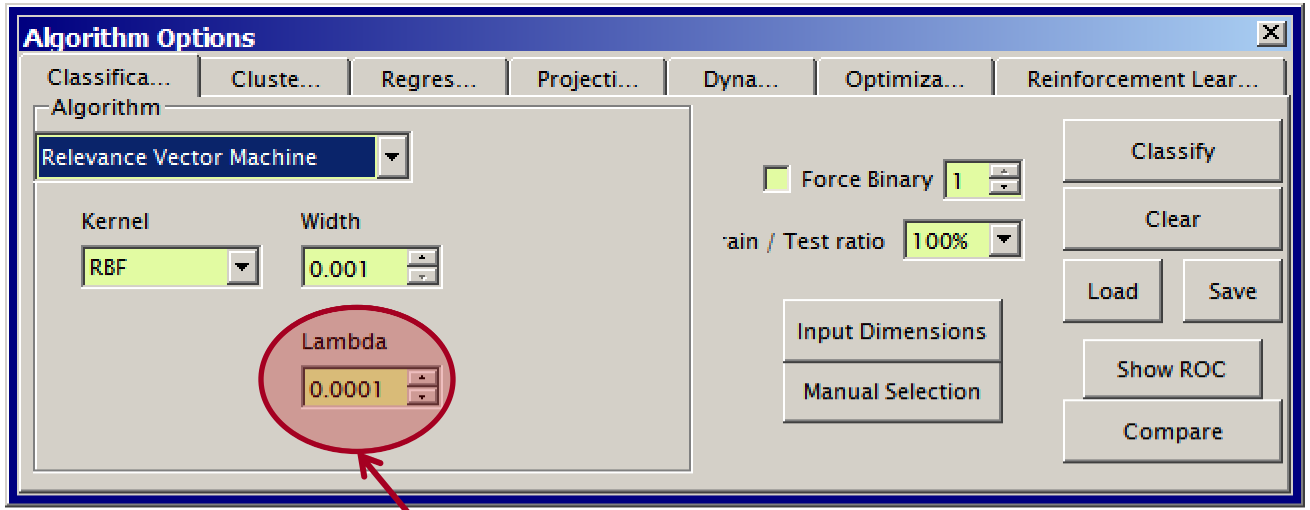
Question 1: Why zero-mean?

Equivalent to the KKT condition: $\sum_i y_i \alpha_i = 0$

Solving the problem now requires estimating the optimal set of parameters, i.e. all the $\{\alpha_i, \sigma_i,\}_{i=0..M}$ and σ_ε

One cannot compute the optimal alpha in closed form. One must use an iterative procedure similar to expectation maximization. The procedure differs depending on whether we consider the classification or regression case.
(see Tipping 2001, supplementary material, for details).

Relevance Vector Machine



Algorithm Options

Algorithm: Relevance Vector Machine

Kernel: RBF

Width: 0.001

Lambda: 0.0001

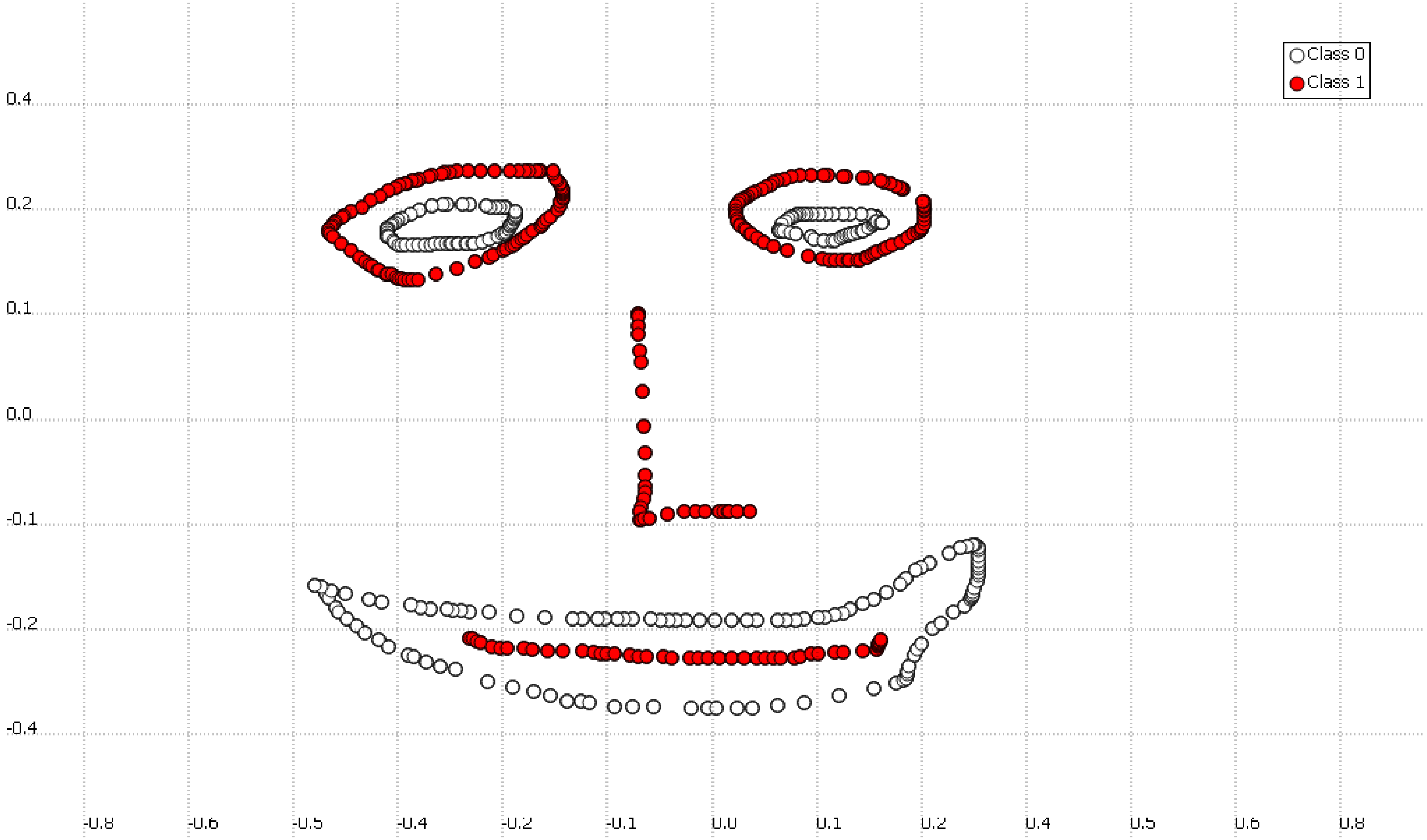
□ Force Binary 1

ain / Test ratio 100%

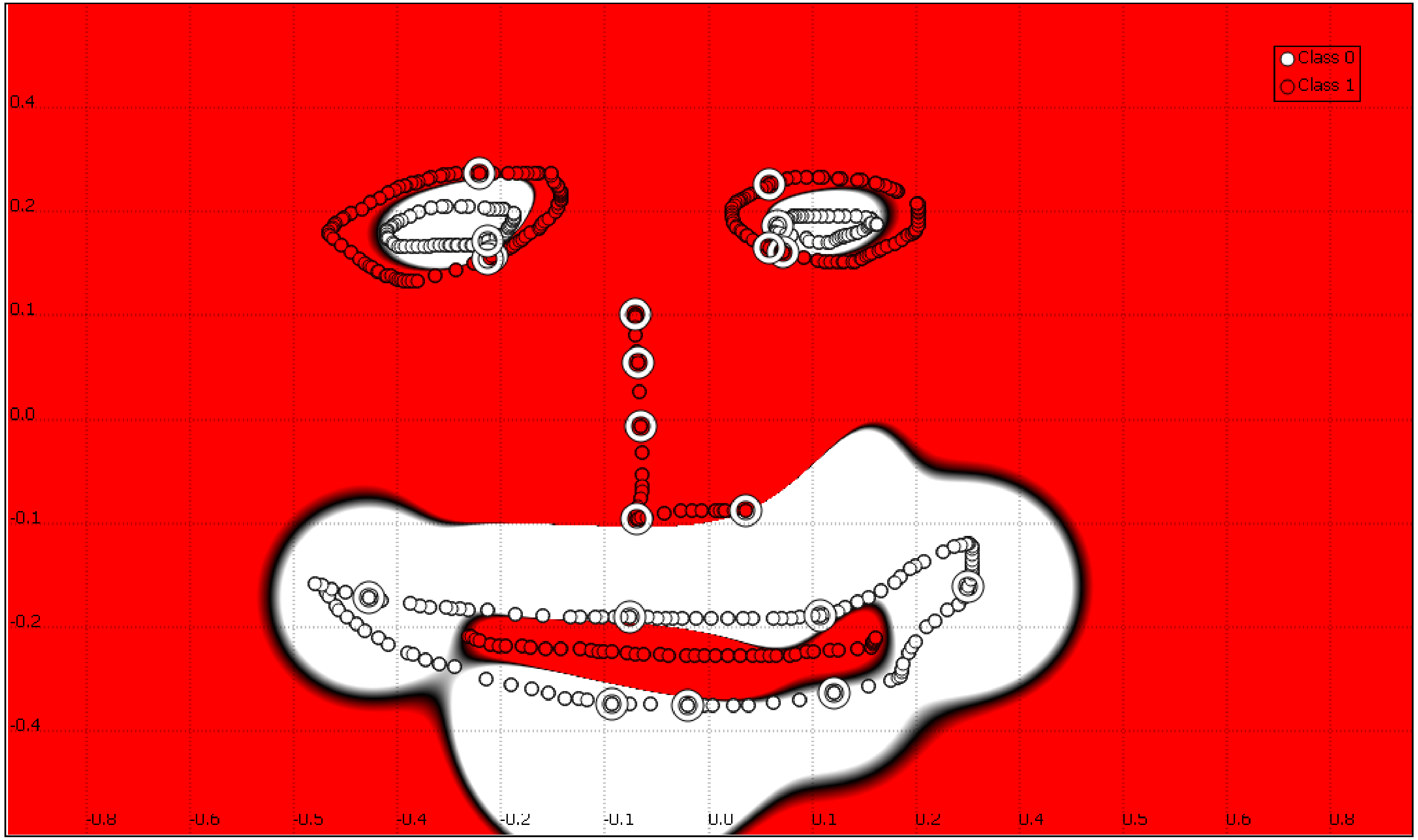
$\lambda \in [0,1]$

This parameter is a stopping criterion for the optimization. It determines how good the fit is. The smaller the value, the closer the true parameters are fitted.

Relevance Vector Machine



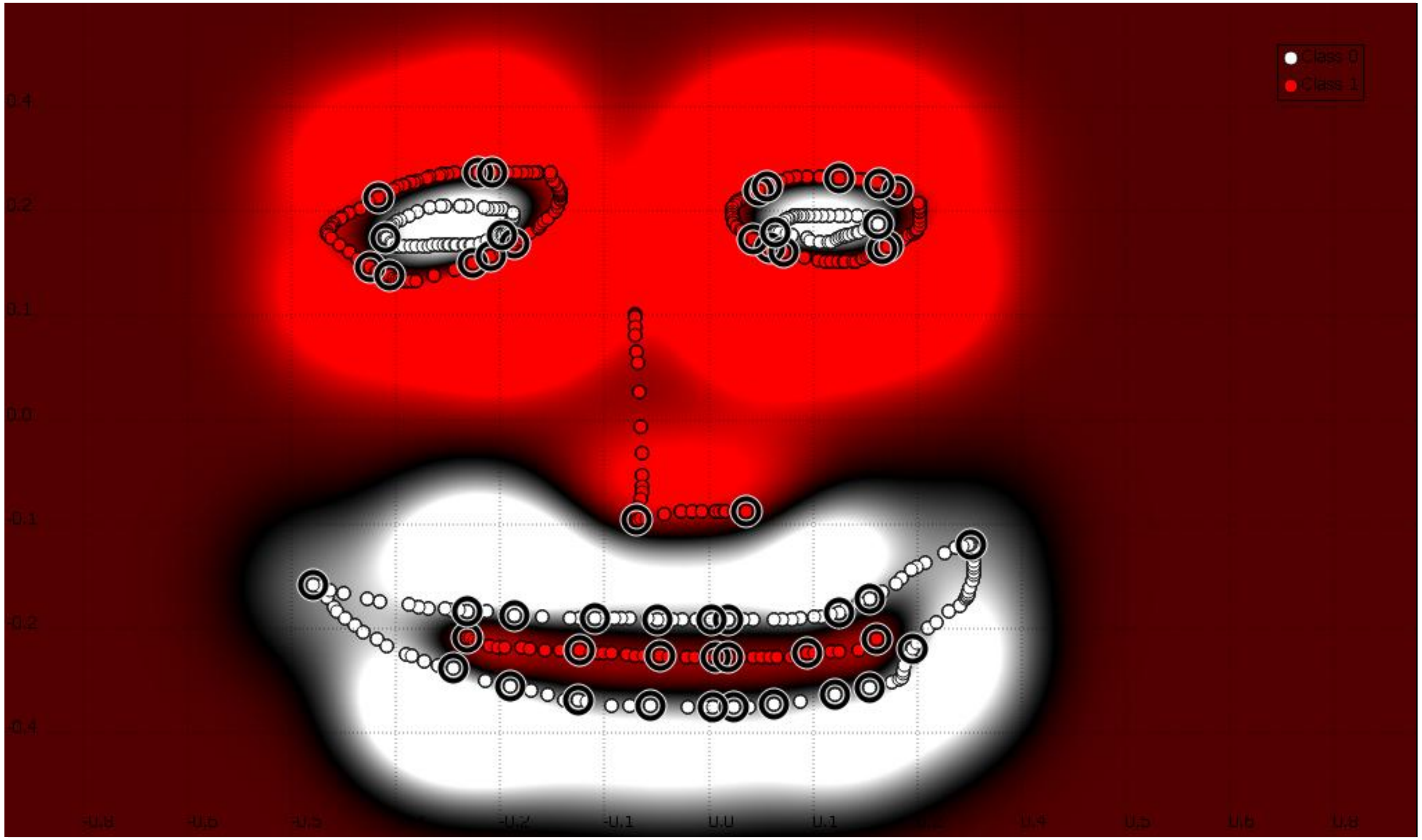
Relevance Vector Machine



RVM with kernel width = 0.01

477 datapoints, 19 support vectors

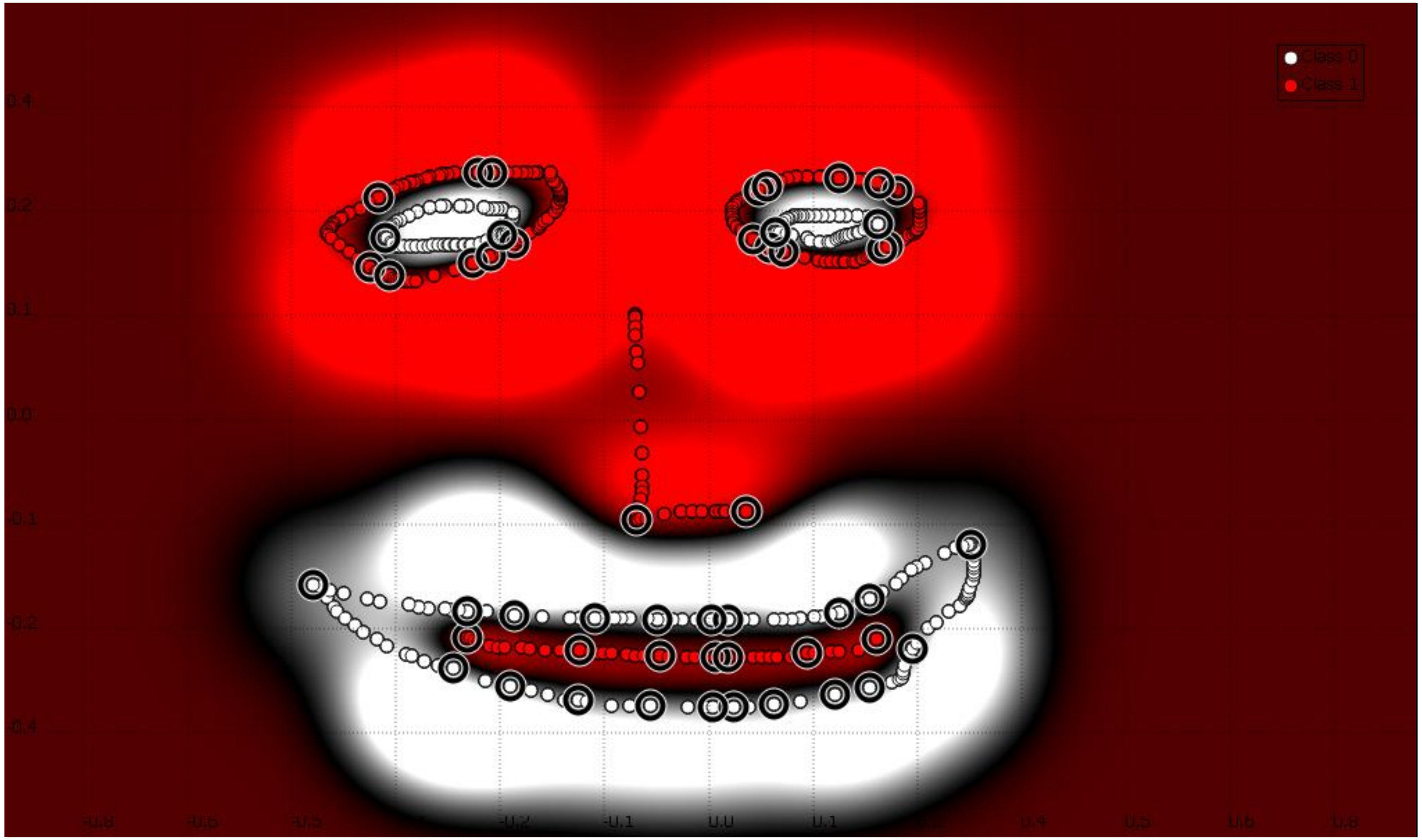
Relevance Vector Machine



SVM with C=1000, kernel width = 0.01

477 datapoints, 51 support vectors

Relevance Vector Machine



Notice the notion of uncertainty of the model encapsulated in the distribution (shadings of grey and red)