Sequential minimal optimization for SV models

Nazarov Ivan

ivan.nazarov@skolkovotech.ru

Skolkovo Institute of Science and Technology

January 30, 2017

Support Vector Models

Support Vector Classification

Consider an i.i.d. training sample $\mathcal{S} = (x_i, y_i)_{i=1}^m \sim D$ over $X \times \{-1, +1\}$.

The **S**upport **V**ector **C**lassification problem is

$$\begin{aligned} \underset{\beta_0 \in \mathbb{R}, \beta \in \mathcal{H}, \xi}{\texttt{minimize}} \quad & \frac{1}{2}\|\beta\|^2 + \sum_{i=1}^m C_i \xi_i \,, \\ \texttt{subject to} \quad & y_i \bigl(\langle \phi(x_i), \beta \rangle + \beta_0\bigr) \geq 1 - \xi_i \,, \\ & \xi_i \geq 0 \,,\, i = 1, \ldots, m \,. \end{aligned} \tag{SVC}$$

Here $(\mathcal{H}, \langle \cdot, \cdot \rangle)$ is the feature space of the kernel K with feature maps $\phi : X \mapsto \mathcal{H}$, $C_i \geq 0$ are the slack penalties, and ξ_i are slack varaibles.

Note: Typically in (SVC) C_i are set to a constant $C > 0$, however point-dependent penalties can be chosen to **fine-tune the balance** of $\mathcal{S}$.

The dual problem, corresponding to the primal (SVC) is

$$\begin{aligned} \underset{\alpha \in \mathbb{R}^{m\times 1}}{\texttt{minimize}} \quad & \frac{1}{2}\alpha' Q\alpha - \mathbf{1}'\alpha\,, \\ \texttt{subject to} \quad & y'\alpha = 0\,, \\ & \alpha_i \in [0, C_i]\,,\, i = 1, \ldots, m\,, \end{aligned} \tag{SVC-dual}$$

where $\mathbf{1}$ is the $m \times 1$ vector of ones, and $Q \in \mathbb{R}^{m\times m}$ has entries $Q_{ij} = y_i K(x_i, x_j) y_j$.

The solution to (SVC) is reconstructed from (SVC-dual)

$$\beta^* = \sum_{i=1}^{m} \alpha_i y_i \phi(x_i)\,,\ \text{and}\ \beta_0^* = \frac{1}{|\texttt{SV}|}\sum_{i\in\texttt{SV}} y_i - \langle \phi(x_i), \beta^* \rangle\,,$$

where $\texttt{SV} = \{i\,:\,\alpha_i \in (0, C_i)\}$ – the set of support vectors.

Support Vector Models

Support Vector Classification

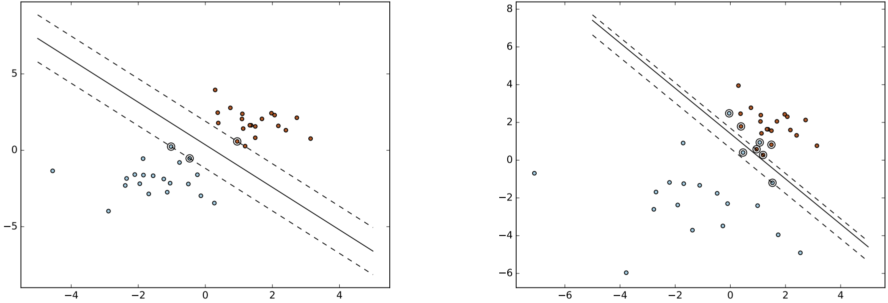

Figure: A sample decision boundary of SVC. Source: Scikit User Guide.

Support Vector Models

Support Vector Regression

An i.i.d. training sample $\mathcal{S} = (x_i, y_i)_{i=1}^m \sim D$ over $X \times \mathbb{R}$ and a fixed tolerance $\varepsilon > 0$.

The ε–**S**upport **V**ector **R**egression problem is

$$\begin{aligned}
\underset{\beta_0\in\mathbb{R},\beta\in\mathcal{H},\xi^+,\xi^-}{\texttt{minimize}} \quad & \frac{1}{2}\|\beta\|^2 + \sum_{i=1}^m C_i^+ \xi_i^+ + \sum_{i=1}^m C_i^- \xi_i^- \,, \\
\texttt{subject to} \quad & \big(\langle \phi(x_i), \beta\rangle + \beta_0\big) - y_i \leq \varepsilon + \xi_i^+ \,, \\
& y_i - \big(\langle \phi(x_i), \beta\rangle + \beta_0\big) \leq \varepsilon + \xi_i^- \,, \\
& \xi_i^+, \xi_i^- \geq 0 \,,\, i = 1, \ldots, m \,.
\end{aligned} \tag{ε-SVR}$$

Here $(C_i^+)_{i=1}^m \geq 0$ are $(C_i^-)_{i=1}^m \geq 0$ are the slack penalties.

Note: $C_i^+ = C^+$ and $C_i^- = C^-$ permits the model to be fine-tuned for asymmetric costs of under- and over- prediction of the target.

Support Vector Models

Support Vector Regression

The dual problem, corresponding to the primal (ε-SVR) is

$$\begin{aligned} & \underset{\alpha^+,\alpha^-\in\mathbb{R}^{m\times 1}}{\texttt{minimize}} && \frac{1}{2}\begin{pmatrix}\alpha^+\\ \alpha^-\end{pmatrix}'\begin{pmatrix}K & -K\\ -K & K\end{pmatrix}\begin{pmatrix}\alpha^+\\ \alpha^-\end{pmatrix} + \begin{pmatrix}\mathbf{1}\varepsilon + y\\ \mathbf{1}\varepsilon - y\end{pmatrix}'\begin{pmatrix}\alpha^+\\ \alpha^-\end{pmatrix}, \\ & \texttt{subject to} && \begin{pmatrix}\mathbf{1}\\ -\mathbf{1}\end{pmatrix}'\begin{pmatrix}\alpha^+\\ \alpha^-\end{pmatrix} = 0, \text{ and } \alpha_i^+ \in [0, C_i^+], \alpha_i^- \in [0, C_i^-]. \end{aligned} \tag{ε-SVR-dual}$$

where $K \in \mathbb{R}^{m\times m}$ has entries $K_{ij} = K(x_i, x_j)$.

- the $2m \times 2m$ matrix in (ε-SVR-dual) is **positive semi-definite** iff $K \succeq 0$
- the dual solution has $\alpha_i^+\alpha_i^- = 0$

Solution to (ε-SVR): If $\texttt{SV}^\square = \{i : \alpha_i^\square \in (0, C_i^\square)\}$ and $r_i = y_i - \langle \phi(x_i), \beta^*\rangle$ then

$$\beta^* = \sum_{i=1}^m (\alpha_i^- - \alpha_i^+)\phi(x_i), \text{ and } \beta_0^* = \frac{\sum_{i\in\texttt{SV}^+\uplus\texttt{SV}^-} r_i}{|\texttt{SV}^+| + |\texttt{SV}^-|} + \varepsilon\frac{|\texttt{SV}^+| - |\texttt{SV}^-|}{|\texttt{SV}^+| + |\texttt{SV}^-|}.$$

Support Vector Models

Support Vector Regression

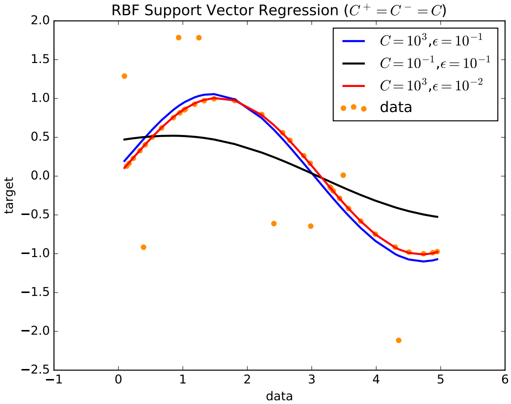

Figure: The regression using ε-SVR. Source: Scikit User Guide.

Support Vector Models

One-Class SVM

An i.i.d. training sample $\mathcal{S} = (x_i)_{i=1}^m \sim D$ over X and a fixed confidence $\nu \in (0,1)$.

The **O**ne-**C**lass SVM estimates the support of a high-dimensional distibution by a soft-margin supporting hyperplane:

$$\begin{aligned} \underset{\rho\in\mathbb{R},\beta\in\mathcal{H},\xi}{\texttt{minimize}} \quad & \frac{1}{2}\|\beta\|^2 - \rho + \frac{1}{\nu C}\sum_{i=1}^m C_i\xi_i\,, \\ \texttt{subject to} \quad & \langle\phi(x_i),\beta\rangle \geq \rho - \xi_i\,, \\ & \xi_i \geq 0\,, i=1,\ldots,m\,. \end{aligned} \tag{OC-SVM}$$

Here $(C_i)_{i=1}^m \geq 0$ are the sample weights, $C = \sum_{i=1}^m C_i > 0$.

Note: For $C_i = 1$ the parameter ν determines the fraction of support vectors, i.e. points with $\langle\phi(x_i),\beta^*\rangle \leq \rho^*$.

Support Vector Models

One-Class SVM

The dual problem, corresponding to (OC-SVM) is

$$\begin{aligned} \underset{\alpha \in \mathbb{R}^{m \times 1}}{\texttt{minimize}} \quad & \frac{1}{2}\alpha' K \alpha \,, \\ \texttt{subject to} \quad & \mathbf{1}'\alpha = \nu C \,, \\ & \alpha_i \in [0, C_i] \,, \, i = 1, \ldots, m \,. \end{aligned} \tag{OC-SVM-dual}$$

The solution to the original problem us

$$\beta^* = \frac{1}{\nu C} \sum_{i=1}^{m} \alpha_i \phi(x_i) \,, \text{ and } \rho^* = \frac{1}{|\texttt{SV}|} \sum_{i \in \texttt{SV}} \langle \phi(x_i), \beta^* \rangle \,.$$

The soft support, $\texttt{supp}(\mathcal{S})$, is $\{x \in X \,:\, d(x) \geq 0\}$, where $d(x) = \langle \phi(x_i), \beta^* \rangle - \rho^*$.

Support Vector Models

One-Class SVM

A sample image of two-clusters enveloped by a soft hyperplane

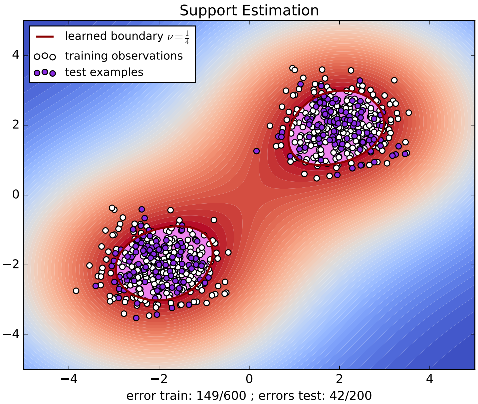

Figure: Distribution support estimation with OC-SVM. Source: Scikit User Guide.

Sequential minimal optimization

Quadratic Problem

The problems (SVC-dual), (ε-SVR-dual), and (OC-SVM-dual) are instantces of the **same quadratic optimization problem** with linear and box constraints:

$$\begin{aligned} &\underset{\alpha}{\texttt{minimize}} && f(\alpha) = \frac{1}{2}\alpha' Q\alpha + p'\alpha\,, \\ &\texttt{subject to} && z'\alpha = \Delta\,,\ \text{and}\ \alpha_i \in [0, C_i]\,,\ i = 1, \ldots, m\,. \end{aligned} \tag{QP}$$

Here $Q \in \mathbb{R}^{m\times m}$ is a **positive definite** matrix, $p \in \mathbb{R}^{m\times 1}$, $z \in \{-1, +1\}^{m\times 1}$, $\Delta \geq 0$, and $C_i > 0$ for all $i = 1, \ldots, m$.

Reductions:

- (SVC-dual): set $Q = K$, $p = -\mathbf{1}$, $z = y$, and $\Delta = 0$
- (ε-SVR-dual): set $Q = \begin{pmatrix} K & -K \\ -K & K \end{pmatrix}$, $p = \begin{pmatrix} \mathbf{1}\varepsilon + y \\ \mathbf{1}\varepsilon - y \end{pmatrix}$, $z = \begin{pmatrix} \mathbf{1} \\ -\mathbf{1} \end{pmatrix}$, and $\Delta = 0$
- (OC-SVM-dual): set $Q = K$, $p = \mathbf{0}$, $z = \mathbf{1}$, and $\Delta = \nu C$

Sequential minimal optimization

Algorithm Properties

SMO is a powerful yet simple iterative procedure that efficiently sloves (QP)

Starting from a feasible α^1 perform a sequence of updates such that after k-th step

- $f(\alpha^{k+1}) < f(\alpha^k)$
- α^{k+1} is admissible
  - $z'\alpha^{k+1} = \Delta$
  - $\alpha_i^k \in [0, C_i]$ for all $i = 1, \ldots, m$

SMO, as proposed in [1], constructs a sequence, which

- progressively improves $f(\alpha)$ until its minimum
- offers the linear convergence rate to the optimum of (QP)
- performs each step in $O(m)$ time
- requires $O(2m)$ storage (with clever memory usage)

Sequential minimal optimization

Key Idea

Situation: We have a feasible α^k: $z'\alpha^k = \Delta$, and $\alpha_i^k \in [0, C_i]$.

Goal: Find a simple and quick adjustment $\delta \in \mathbb{R}^{m\times 1}$ such that $\alpha^{k+1} = \alpha^k + \delta$ is feasible and $f(\alpha^{k+1}) < f(\alpha^k)$.

Observations:

- $f(\alpha^{k+1}) - f(\alpha^k) = \delta'\nabla f(\alpha^k) + \frac{1}{2}\delta' Q\delta$, where $\nabla f(\alpha^k) = Q\alpha^k + p$
- $z'\alpha^{k+1} = \Delta$ if and only if $z'\delta = 0$
- The simplest δ is the one with the most **zeros**

Cannot use the coordinate-wise descent due to $z'\delta = 0$ constraint.

- use "two coordinate" descent: fix $\delta_l = 0$ for $l \notin \{i, j\}$, and minimize over δ_i and δ_j
- use a "clever" strategy to pick $i, j \in \{1, \ldots, m\}$ on each iteration

Sequential minimal optimization

the Subproblem

For a given pair $\{i,j\}$ and this "sparse" δ we have $z'\delta = z_i\delta_i + z_j\delta_j = 0$ and

$$f(\alpha^{k+1}) - f(\alpha^k) = \frac{1}{2}\big(\delta_i^2 Q_{ii} + \delta_j^2 Q_{jj} + 2\delta_i\delta_j Q_{ij}\big) + \delta_i\nabla_i + \delta_j\nabla_j\,,$$

where ∇_l is $\nabla_l f(\alpha^k)$ for short.

Consider the subproblem

$$\begin{aligned} \underset{\delta_i,\delta_j}{\texttt{minimize}} \quad & \frac{1}{2}\big(\delta_i^2 Q_{ii} + \delta_j^2 Q_{jj} + 2\delta_i\delta_j Q_{ij}\big) + \delta_i\nabla_i + \delta_j\nabla_j\,, \\ \texttt{subject to} \quad & z_i\delta_i + z_j\delta_j = 0\,. \end{aligned} \tag{Aux$_{ij}$}$$

What about the box constraints $\alpha_i^{k+1} \in [0, C_i]$?

- first, solve for the best δ, and worry about the box later

Sequential minimal optimization

Solving the subproblem

the problem (Aux_{ij}) can be solved easily

- substitute $d_l = \delta_l z_l$, $l \in \{i, j\}$
- notice that $d_j = -d_i$
- use the fact that $z_l \in \{-1, +1\}$ implies $z_l^2 = 1$
- solve an an even more simpler equivalent problem:

$$\underset{d_i}{\texttt{minimize}} \quad \frac{1}{2}\bigl(Q_{ii} + Q_{jj} - 2 z_i z_j Q_{ij}\bigr) d_i^2 + d_i (z_i \nabla_i - z_j \nabla_j)\,. \tag{Aux'_{ij}}$$

The solution d_i^* of (Aux'_{ij}) and its minimal value $\texttt{Opt}_{ij}$ are

$$d_i^* = -\frac{z_i \nabla_i - z_j \nabla_j}{a_{ij}}\,, \text{ and } \texttt{Opt}_{ij} = -\frac{1}{2}\frac{\bigl(z_i \nabla_i - z_j \nabla_j\bigr)^2}{a_{ij}}\,,$$

where $a_{ij} = Q_{ii} + Q_{jj} - 2 z_i z_j Q_{ij}$. **Note**: Since $Q \succ 0$, a_{ij} is always positive!

Sequential minimal optimization

Adjusting for the box constraints

The optimal δ^* in (Aux_{ij}) is $\delta_i^* = z_i d_i^*$ and $\delta_j^* = -z_j d_i^*$ with $\delta_j^* = -z_j z_i \delta_i^*$.

The candidate $\hat{\alpha} \in \mathbb{R}^m$ with $\hat{\alpha}_l = \alpha_l^k + \delta_l^*$ for $l \in \{i, j\}$, and $\hat{\alpha}_l = \alpha_l^k$ for $l \notin \{i, j\}$:

- satisfies the linear constraint $z'\hat{\alpha} = z'\alpha^k + (z_i\delta_i^* + z_j\delta_j^*) = \Delta + 0$
- possibly violates the box constraints only at $\hat{\alpha}_A = (\hat{\alpha}_i, \hat{\alpha}_j)$, since
  - α^k is feasible $\Rightarrow \hat{\alpha}_l \in [0, C_l]$, $l \notin \{i, j\}$

To project the candidate solution $\hat{\alpha}$ back into the box we to consdier two cases.

- $z_i \neq z_j$: $\alpha_A^k \to \hat{\alpha}_A$ is along 45° rays in $\mathbb{R}^2$
- $z_i = z_j$: $\alpha_A^k \to \hat{\alpha}_A$ is along 135° rays

Sequential minimal optimization

Adjusting for the box constraints

When $z_i \neq z_j$ we have $\delta_j^* = \delta_i^*$ and $\hat{\alpha}_j - \hat{\alpha}_i = \alpha_j^k - \alpha_i^k$.

If $\hat{\alpha}$ is infeasible:

- α^k is feasible $\Rightarrow$ $\hat{\alpha}$ is **not** in NA
- project along 45° rays into the box

Projection for each valid region of $\hat{\alpha}$:

I: $\alpha_i^{k+1} \leftarrow C_i$, $\alpha_j^{k+1} \leftarrow \hat{\alpha}_j - (\hat{\alpha}_i - C_i)$

II: $\alpha_j^{k+1} \leftarrow C_j$, $\alpha_i^{k+1} \leftarrow \hat{\alpha}_i - (\hat{\alpha}_j - C_j)$

III: $\alpha_j^{k+1} \leftarrow 0$, $\alpha_i^{k+1} \leftarrow \hat{\alpha}_i - \hat{\alpha}_j$

IV: $\alpha_i^{k+1} \leftarrow 0$, $\alpha_j^{k+1} \leftarrow \hat{\alpha}_j - \hat{\alpha}_i$

- $z'\alpha^{k+1} = z'\hat{\alpha} = \Delta$

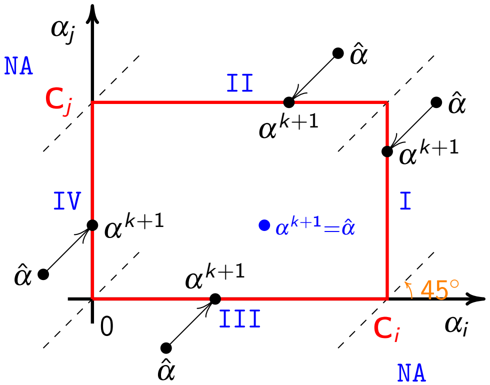

Figure: Projections for $z_i \neq z_j$.

Sequential minimal optimization

Adjusting for the box constraints

When $z_i = z_j$ we have $\delta_j^* = -\delta_i^*$ and $\hat{\alpha}_j + \hat{\alpha}_i = \alpha_j^k + \alpha_i^k$.

If $\hat{\alpha}$ is infeasible:

- α^k is feasible $\Rightarrow$ $\hat{\alpha}$ is not in NA
- project along 135° rays into the box

Used projection in each valid region:

I: $\alpha_i^{k+1} \leftarrow C_i,\ \alpha_j^{k+1} \leftarrow \hat{\alpha}_j + (\hat{\alpha}_i - C_i)$

II: $\alpha_j^{k+1} \leftarrow C_j,\ \alpha_i^{k+1} \leftarrow \hat{\alpha}_i + (\hat{\alpha}_j - C_j)$

III: $\alpha_j^{k+1} \leftarrow 0,\ \alpha_i^{k+1} \leftarrow \hat{\alpha}_i + \hat{\alpha}_j$

IV: $\alpha_i^{k+1} \leftarrow 0,\ \alpha_j^{k+1} \leftarrow \hat{\alpha}_j + \hat{\alpha}_i$

- $z'\alpha^{k+1} = z'\hat{\alpha} = \Delta$

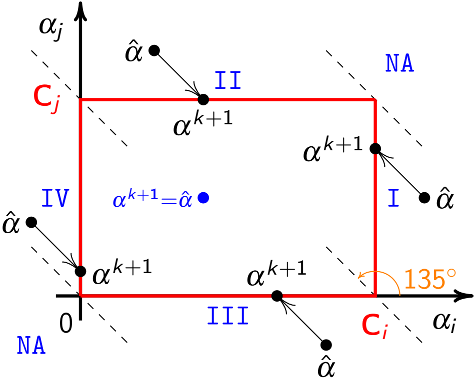

Figure: Projections for $z_i = z_j$.

Sequential minimal optimization

Selecting the most promising pair i, j

Suppose (QP) is feasible and Q is positive semi-definite.

For a feasible α in (QP) let $L_\alpha, U_\alpha \subset \{1, \ldots, m\}$ be

$$\begin{aligned} L_\alpha &= \{i \,:\, \alpha_i > 0 \,\&\, z_i = +1 \text{ or } \alpha_i < C_i \,\&\, z_i = -1\}\,, \\ U_\alpha &= \{i \,:\, \alpha_i > 0 \,\&\, z_i = -1 \text{ or } \alpha_i < C_i \,\&\, z_i = +1\}\,. \end{aligned}$$

Result in [1]: α is optimal in (QP) iff for some $b \in \mathbb{R}$

$$m(\alpha) = \max_{i \in U_\alpha} -z_i \nabla_i f(\alpha) \leq b \leq \min_{j \in L_\alpha} -z_j \nabla_i f(\alpha) = M(\alpha)\,. \tag{*}$$

α is **not** optimal in (QP) iff $m(\alpha) - M(\alpha) > 0$

- the **inferiority** of α^k can be measured by $\mathtt{err}^k = m(\alpha^k) - M(\alpha^k)$

Sequential minimal optimization

Selecting the most promising pair i, j

Violating pair

- If α^k violates (*) then it does not solve (QP), and there must be $(i, j) \in U_{\alpha^k} \times L_{\alpha^k}$

$$-z_i \nabla_i f(\alpha) > -z_j \nabla_j f(\alpha) .$$

Optimal δ_i^* and δ_j^* in (Aux_{ij}) are $z_i d^*$ and $-z_j d^*$ with

$$d^* = -\frac{z_i \nabla_i - z_j \nabla_j}{Q_{ii} + Q_{jj} - 2 z_i z_j Q_{ij}} .$$

Note: For any violating pair (i, j) we have $d^* > 0$.

Sequential minimal optimization

Selecting the most promising pair i, j

- If (i, j) is a violating pair then the line segment $[\alpha^k, \hat{\alpha}]$ passes **through** the box
- (Aux'_{ij}) is minimal at $\hat{\alpha}$ and $f(\alpha) < f(\alpha^k)$ at every α on $(\alpha^k, \hat{\alpha}]$

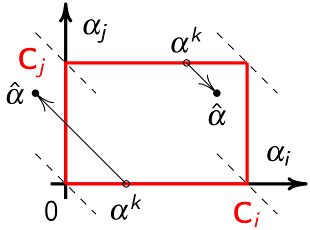

Figure: Box crossings for $z_i = z_j$.

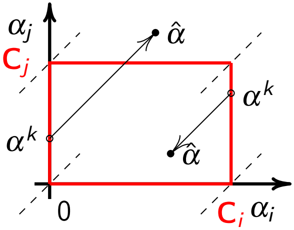

Figure: Box crossings for $z_i \neq z_j$.

Note: projection $\hat{\alpha} \to \alpha^{k+1}$ into the box still yields $f(\alpha^{k+1}) < f(\alpha^k)$

Sequential minimal optimization

Selecting the most promising pair i, j

Results so far: for a violating pair (i, j)

- the move $\alpha^k \to \hat{\alpha}$ decreases the objective: $f(\alpha^k) > f(\hat{\alpha})$
- the clipping $\hat{\alpha} \to \alpha^{k+1}$ does not increase f much: $f(\alpha^k) > f(\alpha^{k+1}) \geq f(\hat{\alpha})$

Pair heuristic (WSS2 in [1]): for $\nabla_I = \nabla_I f(\alpha^k)$

- if (*) is volated, then $i \in \texttt{argmax}_{i \in U_{\alpha^k}} -z_i \nabla_i$ is in a violating pair
- pick an accomplice $j \in L_{\alpha^k}$ with the lowest value of (Aux_{ij}):

$$j \in \texttt{argmin}\left\{ -\frac{1}{2} \frac{\left(z_i \nabla_i - z_j \nabla_j\right)^2}{Q_{ii} + Q_{jj} - 2 z_i z_j Q_{ij}} \,:\, j \in L_{\alpha^k}\,,\ \text{and } z_i \nabla_i < z_j \nabla_j \right\}. \qquad \text{(WSS2)}$$

We are guaranteed to make an update $\alpha^k \to \alpha^{k+1}$ with a **large enough** decrease.

Sequential minimal optimization

the whole Algorithm

We arrive at SMO iterative solver for (QP) ($\eta > 0$ – tolerance)

Set α^1 to some feasible point in (QP);
while α^k *is not stationary within* $\mathtt{err}^k > \eta$ **do**
    Pick a **good** pair $\{i, j\} \subset \{1, \ldots, m\}$ with (WSS2);
    Move $\alpha^k \to \hat{\alpha}$ by solving (Aux_{ij});
    Move $\hat{\alpha} \to \alpha^{k+1}$ by projecting back into $[0, C_i] \times [0, C_j]$;
    `// Here` $z'\alpha^{k+1} = \Delta$`, and` $\alpha_l^{k+1} \in [0, C_l]$
    $k \leftarrow k + 1$;
end
return α^k;

Algorithm 1: SMO

Sequential minimal optimization

the whole Algorithm

Theorem 4 in [1]:

- the sequence $(\alpha^k)_{k\geq 1}$ converges to the unique global solution $\bar{\alpha}$ of (QP).

Theorem 6 in [1]: there is $c \in (0,1)$

- $f(\alpha^{k+1}) - f(\bar{\alpha}) < c\big(f(\alpha^k) - f(\bar{\alpha})\big)$
- for any $\eta > 0$ there is $\bar{k}$ such that within $\bar{k} + O\log\frac{1}{\eta}$ we have $f(\alpha^k) - f(\bar{\alpha}) < \eta$

Asymptotics of the algorithm:

- (WSS2) is a good heuristic that takes $O(2m)$ (instead of $O(m^2)$ for the best pair)
- SMO algorithm has robust linear convergence rate!

The core of `libsvm` is **SMO** with **computation reuse**, clever **caching** and **speed-ups**.

References

- Rong-En Fan, Pai-Hsuen Chen, and Chih-Jen Lin. "Working Set Selection Using Second Order Information for Training Support Vector Machines". In: *J. Mach. Learn. Res.* 6 (Dec. 2005), pp. 1889–1918. ISSN: 1532-4435. URL: http://dl.acm.org/citation.cfm?id=1046920.1194907.
- F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.