Support Vector Machines

Case study for sensor analysis: the Shockfish project

Lagrangian optimization theory

Hard margin SVMs

Soft margin SVMs

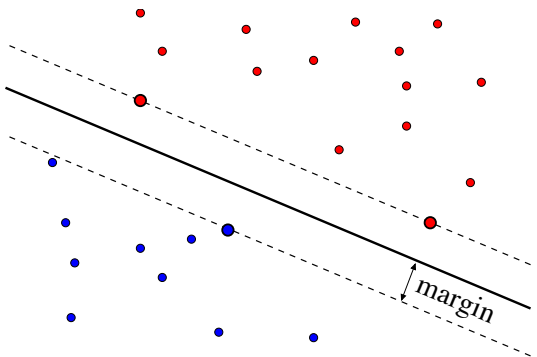
Joachim M. Buhmann

November 14, 2019

Support Vector Machine (SVM): Idea

Generalize perceptrons with margin and kernel

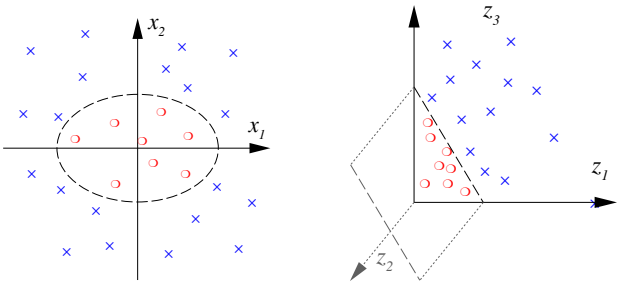
- **Margin**: use a linear classifier with margin m to foster robustness in classification!
- **Kernel**: employ a non-linear feature transformation to improve adaptivity of classifiers!



Nonlinear Transformation in Kernel Space

We pursue a similar strategy as in regression estimation: start with linear regression and “kernelize” it to Gaussian process regression.

$$\Phi : \mathbb{R}^2 \rightarrow \mathbb{R}^3$$
$$(x_1, x_2) \mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt{2}\, x_1 x_2, x_2^2)$$



The nonlinear transformation Φ linearly separates the blue crosses from red circles. In principle we could project all data points to the $(z_1, 0, z_3)$ plane.

Case Study: Parking lot occupancy prediction

Problem: Truck detection on highway rest areas

(SNF Project: Sharon Wulff, JMB & Shockfish AG, Lausanne)

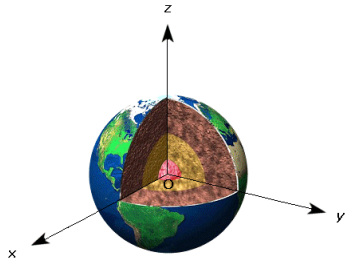
- The automatic detection system should accurately estimate how many truck parking lots are still available.
- Classification task: **Occupied** (8, 11, 15, 16) vs. **non-occupied** (all others)?

Project idea (Shockfish): TinyNode senses earth magnetic field

- TinyNode sensors measure deviations of the earth magnetic field.
- Deviations occur when a large ferromagnetic mass, such as a truck engine, is close to the TinyNode sensor.
- Deploy a sensor in each parking lot and use the sensor readings to detect the presence of a vehicle.

The Data: Sensor readings

The sensor measures the strength of the earth magnetic field along **3 axes**
$\Rightarrow$ vectors of 3-dimensional readings.
Rest readings: measurements collected when the parking lot is empty.



Center values:

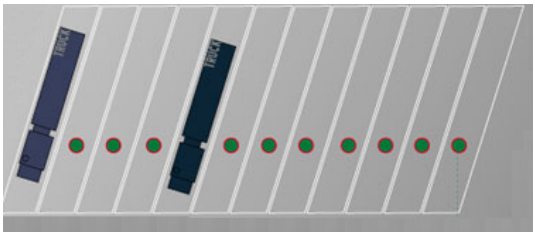
The rest readings drift as a function of the temperature and other physical variations.
$\Rightarrow$ Compute 3 dimensional center values to compensate for the drifts (offsets) from the readings.

Data set: signal(t) = reading(t) - center(t)

The readings as well as the center values are collected periodically. ($\approx$ 2 min)

Problem representation for parking space occupancy

Binary classification per parking lot



The learning system has to estimate if a parking lot is occupied or empty, based on one sensor reading per lot.

Topology of sensors

- One sensor per parking lot
- The parking lots are arranged in a chain
  $\Rightarrow$ each sensor has two immediate neighbors
- Problem: A truck also influences sensor measurements of neighboring sensors

  $\Rightarrow$ weak coupling of neighboring sensors

Design of a machine learning solution

Labeling of the measurements

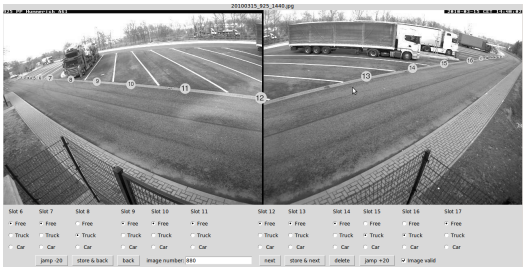
Convert the unsupervised learning problem in a supervised learning question.
Labels are extracted from video imagery.

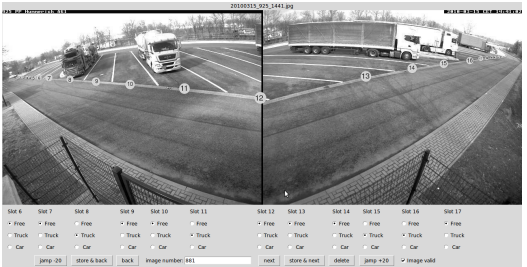
Label extraction system (half automatic, yet convenient; shown are two different days)

- Each image requires labels for $n = 16$ sensors.
- Occupancy indicators are copied to the next image. $\Rightarrow$ One confirmation click to generate the labeling suffices if occupancy states do not change.

Label Extraction

Easier to label images with day-time lighting...





- Often, unsupervised learning problems can be turned into supervised ones with additional labeling effort!
- Labeling can be outsourced to amazon mechanical turk Artificial Artificial Intelligence
- Learn to estimate the quality of the labeler.

What are the Machine Learning Challenges?

Tasks: ML design, label extraction, preprocessing

- Design a new algorithm for extracting occupation with increased reliability by modeling dependencies between neighboring sensors.
- Label the readings of three weeks based on camera imagery.
- Find appropriate data preprocessing filters to remove measurement drifts.

Performance measures

- **Accuracy**: The accuracy of the algorithm for a given time point is the Hamming distance between the predictions and the true occupancy state
- **Generalization**: Classifiers should generalize over sensors and parking lots.
- **Important insight**: carefully specify the conditioning, i.e., on specific sensor, on a parking lot, no conditioning.

Problems in Real World Applications

Non uniformity of sensors

- Initial assumption - conditioned on similar occupancy settings, the sensor readings are the same
- In reality the above assumption does not hold
- Possible explanation - sensors are not aligned, e.g. calibration errors

Signal for occupied lots are not homogeneous

- Every truck generates a different signal
  $\Rightarrow$ the class “occupied parking space” is not homogeneous.
- Signals depend on the steel mass of the truck and its position relative to the sensor.

Data Preprocessing

Goal

Find a **transformation** of the data such that ...

- ... sensors readings are as comparable as possible;
- ... signal discriminability between occupied and non-occupied is maintained.

Comparison of readings

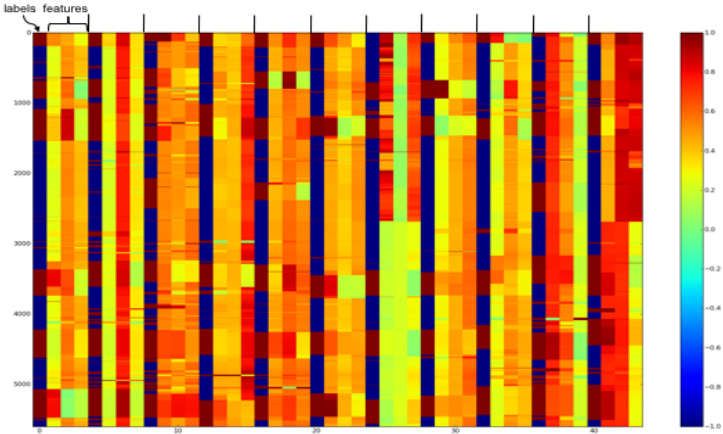
Trucks generate different signals $\Rightarrow$ use **rest readings** to compare different sensors

- Due to neighboring effects we redefine the rest readings as

Readings taken when the **entire** parking lot is empty!

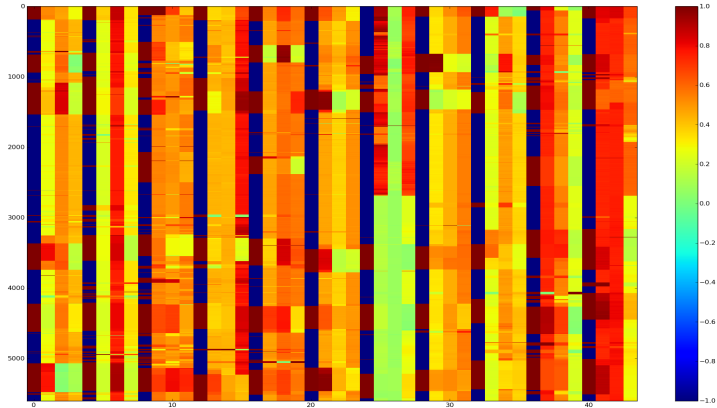
Data Preprocessing

Raw readings, (4 columns: 1 label + 3 features for 11 sensors)



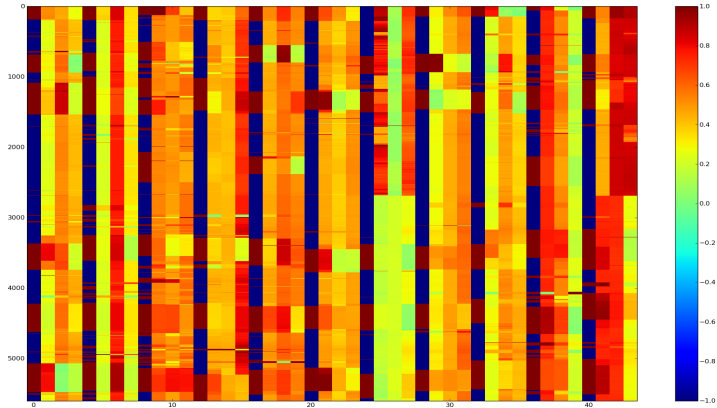
Data Preprocessing

Readings - Center values



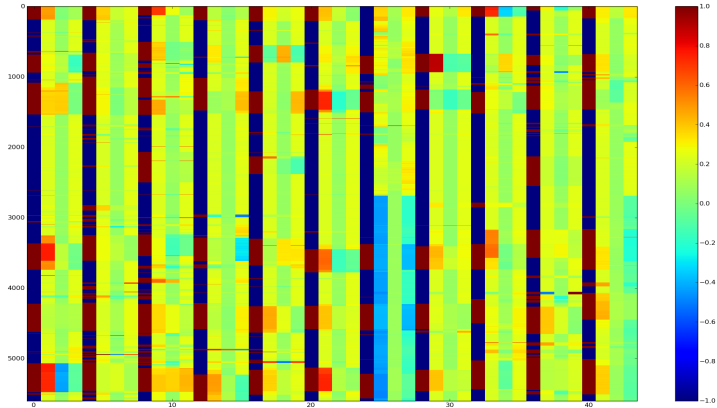
Data Preprocessing

Subtract median of rest readings (per sensor)



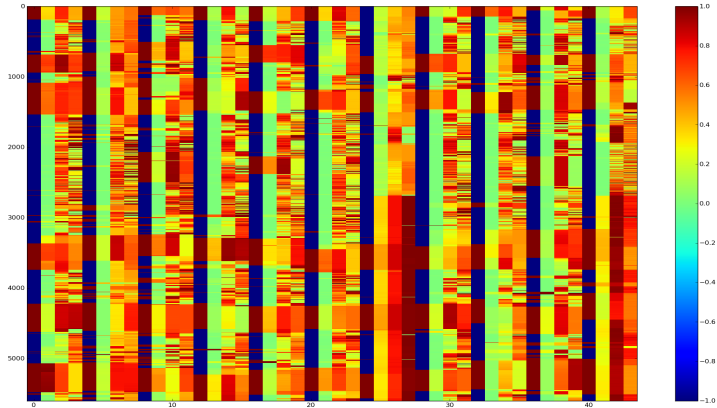
Data Preprocessing

Subtract minimal rest readings (across sensors)



Data Preprocessing

Spherical coordinates



Modeling: classify parking space occupancy

Types of information

- Spacial information:
  - Individual sensors: readings, centers
  - Neighborhood: neighbors (to which degree?) readings, centers
- Transition information: changes in consecutive readings

Classifiers

- Support Vector Machines
- Random Forest
- Graphical Models

Its all about the features..

Measurements

Z-axis most informative, X-axis not informative for most sensors

Spherical coordinates

$$r = \sqrt{x^2 + y^2 + z^2} \quad \theta = \arccos(\tfrac{z}{r}) \quad \varphi = \arctan 2(y, x)$$

- The radius r is very informative
- The angles θ and φ exhibit high variations in consecutive readings in the non-occupied state

Time of the day

The occupancy likelihood varies between different times of the day

Features

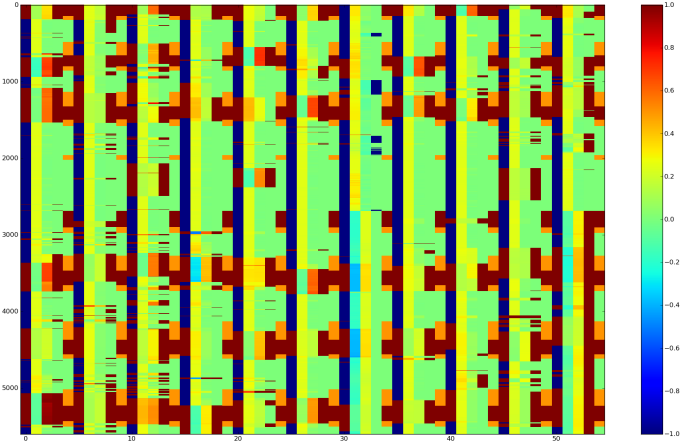


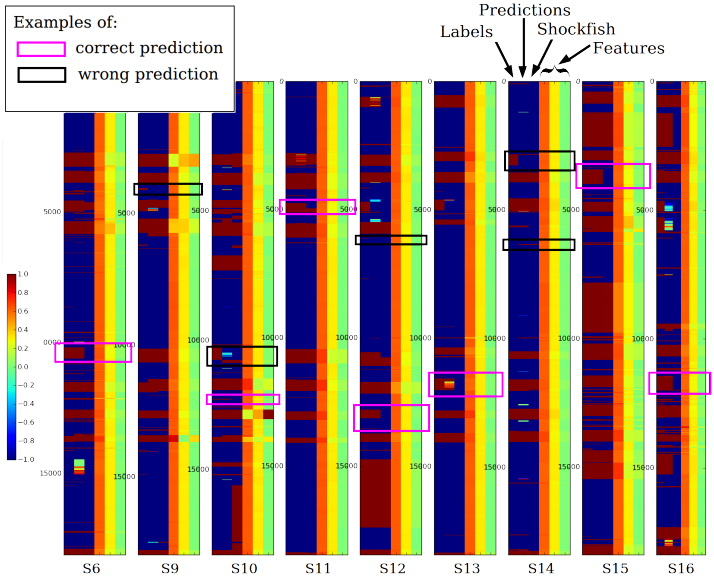
Figure: Features: z-axis, radius, shockfish predictions, time of the day

Benchmark

Experimental setup

- **Data**:
  - Sensors 6-16 (visible to the camera)
  - According to Shockfish sensors 12 and 16 were broken and are therefore not included.
  - A week of labeled readings from Hummerich, parking lot in Germany
  - Number of readings per sensor: between 7800 - 5400 (4500 when aligned)
- **Data splits**:
  1. Generalizing over sensors: “leave one sensor out” accuracy results
  2. Generalizing over time: include all sensors, train on first time period predict on the second

Benchmark Results: Graphical Summary



Benchmark Results: Detection Rate

Average detection rate

The linear SVM has as good performance as a Gaussian mixture model in time; the performance of the comparison solution (Shockfish) has been significantly improved, even with “broken” sensors 12,16.

Sensor	SVM	GM in time	Shockfish
6	**94.88**	94.62	94.75
9	**97.27**	97.04	96.22
10	94.98	**95.55**	80.53
11	**99.29**	98.82	97.80
12	**97.12**	96.99	92.24
13	**97.67**	97.23	94.82
14	96.40	**96.87**	96.03
15	**99.37**	98.44	95.73
16	93.32	**94.91**	77.95
	96.7 $\pm$ 1.9	96.7 $\pm$ 1.3	91.7 $\pm$ 6.8

Support Vector Machines (SVM)

Ingredients:

- **Labeled dataset** $(\mathbf{x}_1, z_1), \ldots, (\mathbf{x}_n, z_n) \in \mathbb{R}^d \times \{-1, +1\}$.
- **Transformation** $\phi : \mathbb{R}^d \to \mathbb{R}^e$.

Goal: To compute a linear classifier

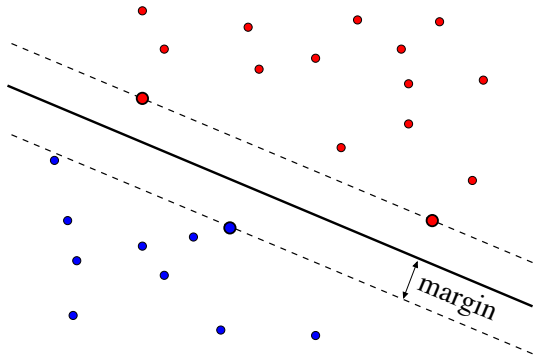
$$\mathbf{w}^\top \phi(\cdot) + \mathbf{w}_0$$

that maximizes the margin.

We let $\mathbf{y}_i = \phi(\mathbf{x}_i)$, for $i \leq n$.

Support Vector Machines (SVM)

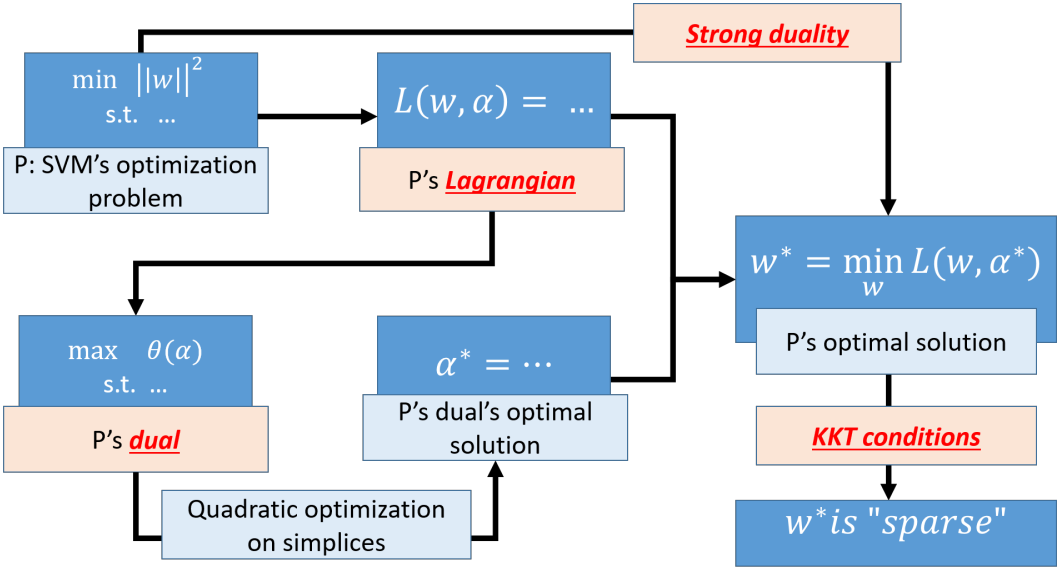
Find hyperplane that maximizes the **margin** m with



The vectors $\mathbf{y}_i$ that define the margin are the **support vectors**.

If the points are linearly separable, then there are at least two support vectors.

Roadmap: How to compute SVMs



Agenda

- Optimization with constraints.
- Hard-margin SVMs.
- Soft-margin SVMs.

Lagrange multipliers

Assume that f and g_i are continuously differentiable.

$$\mathcal{P}: \quad \begin{array}{lll} \min & f(\mathbf{w}), & \mathbf{w} \in \mathbb{R}^d \\ \text{s.t.} & g_i(\mathbf{w}) = 0, & i \leq m \end{array}$$

What to do?

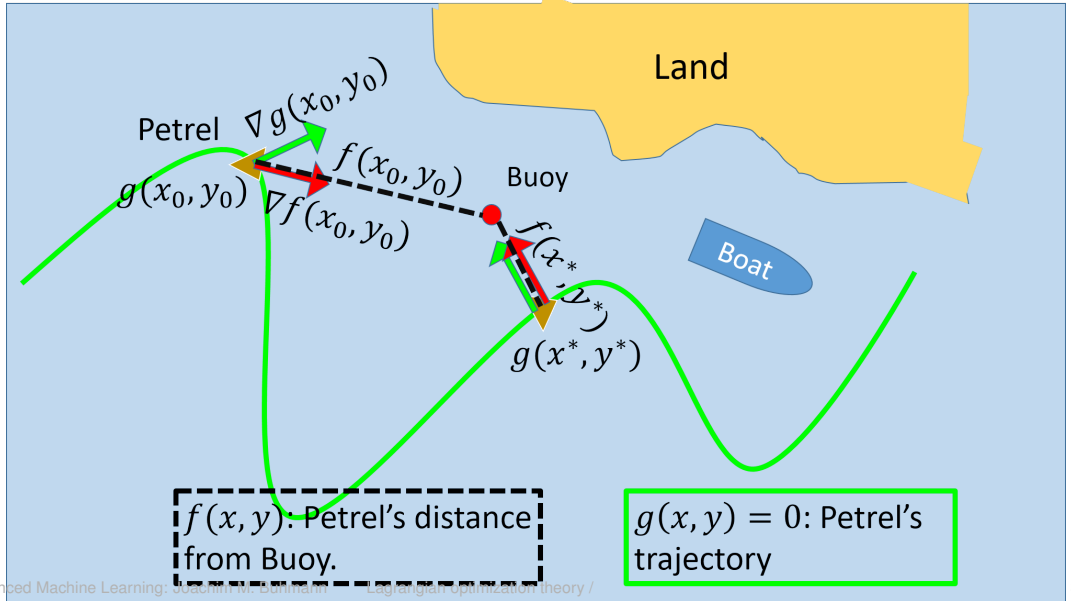
1. Compute $\mathcal{P}$'s Lagrangian

$$\mathcal{L}(\mathbf{w}, \boldsymbol{\lambda}) = f(\mathbf{w}) + \sum_{i \leq m} \lambda_i g_i(\mathbf{w}).$$

2. An optimal solution must satisfy $\nabla \mathcal{L} = 0$.

Intuition: the petrel and the buoy

Intuition: the petrel and the buoy



Intuition: the petrel and the buoy

At (x^*, y^*), where the Petrel is the closest to the Buoy, we have that

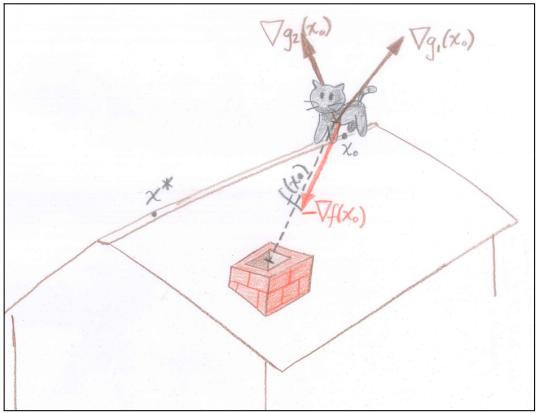
$$\nabla f\left(x^*, y^*\right) + \lambda^* \nabla g\left(x^*, y^*\right) = 0.$$

Moreover, we also have that $g\left(x^*, y^*\right) = 0$.

Both conditions are equivalent to the following:

$$\nabla \left(\underbrace{f\left(x^*, y^*\right) + \lambda^* g\left(x^*, y^*\right)}_{\mathcal{L}(x^*, y^*, \lambda^*)} \right) = 0.$$

Intuition: The cat on the roof



- $f(x)$, distance from point x to the chimney.
- $g_1(x) = 0$ and $g_2(x) = 0$ are the hyperplanes where the two sections of the roof are.
- The cat walks along the roof's ridge ($g_1(x) = g_2(x) = 0$.)
- What's the point x^* in the ridge that is closest to the chimney?

Intuition: The cat on the roof

At x^*, where the cat is the closest to the chimney, we have that

$$\nabla f(x^*) + \lambda_1^* \nabla g_1(x^*) + \lambda_2^* \nabla g_2(x^*) = 0.$$

Moreover, we also have that $g_1(x^*) = 0$ and that $g_2(x^*) = 0$.

These conditions are equivalent to the following:

$$\nabla \left(\underbrace{f(x^*) + \lambda_1^* g_1(x^*) + \lambda_2^* g_2(x^*)}_{\mathcal{L}(x^*, \lambda_1^*, \lambda_2^*)} \right) = 0.$$

Lagrange multipliers

Assume that f and h_j are convex and that g_i is affine.

$$\mathcal{P}: \quad \begin{array}{lll} \min & f(\mathbf{w}), & \mathbf{w} \in \mathbb{R}^d \\ \text{s.t.} & g_i(\mathbf{w}) = 0, & i \leq m \\ & h_j(\mathbf{w}) \leq 0, & j \leq n \end{array}$$

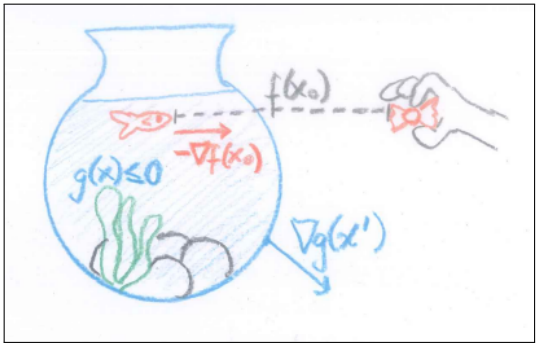
What to do?

1. Compute $\mathcal{P}$'s Lagrangian

$$\mathcal{L}(\mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\alpha}) = f(\mathbf{w}) + \sum_{i \leq m} \lambda_i g_i(\mathbf{w}) + \sum_{j \leq n} \alpha_j h_j(\mathbf{w}), \quad \text{with each } \alpha_j \geq 0.$$

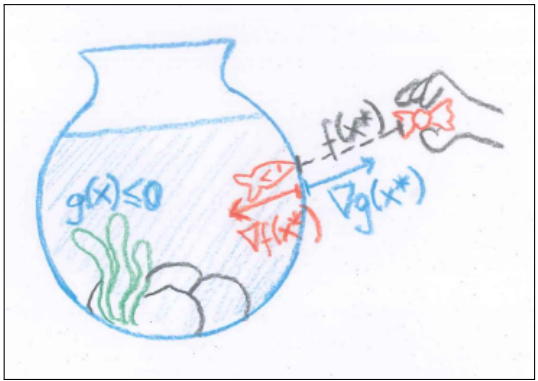
2. An optimal solution must satisfy
   - $\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = 0$ and $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\lambda}} = 0$,
   - $\alpha_j h_j(\mathbf{w}) = 0$, and
   - $\alpha_j \geq 0$, for each $j \leq n$.

Intuition: The fish in the aquarium



- Suppose we know that the candy is outside the aquarium.
- $f(x)$, distance from point x to the candy.
- $g(x) \geq 0$ defines the volume filled with water.
- What's the point x^* in the aquarium that is closest to the candy?

Intuition: The fish in the aquarium



- At the closest point x^*, we have that $\nabla f(x^*)$ and $\nabla g(x^*)$ point in opposite directions!
- What about the general case, when we do not know if the candy is inside or outside the aquarium?

Intuition: The fish in the aquarium

Two cases:

- The optimal point x^* is inside the aquarium. Then $\nabla f(x^*) = 0$.
- The optimal point x^* is outside the aquarium. Then $\nabla f(x^*) + \alpha^* \nabla g(x^*) = 0$, with $\alpha \geq 0$ as $\nabla g(x^*)$ and $\nabla f(x^*)$ must be pointing in opposite directions!!.

These conditions are equivalent to the following:

$$\frac{\partial \mathcal{L}}{\partial x}(x^*, \alpha^*) = 0 \qquad \alpha^* g(x^*) = 0 \qquad \alpha^* \geq 0.$$

Examples

Formal analysis

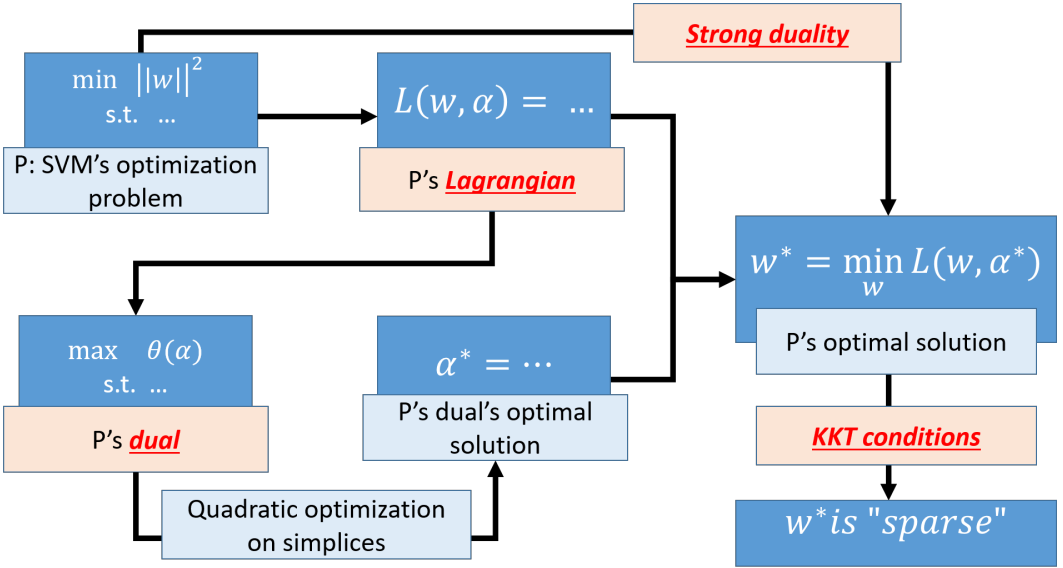
How about the general case?

$$\mathcal{P}: \quad \begin{array}{lll} \min & f(\mathbf{w}), & \mathbf{w} \in \mathbb{R}^d \\ \text{s.t.} & g_i(\mathbf{w}) = 0, & i \leq m \\ & h_j(\mathbf{w}) \leq 0, & j \leq n \end{array}$$

Consider the Lagrangian

$$\mathcal{L}(\mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\alpha}) = f(\mathbf{w}) + \sum_{i \leq m} \lambda_i g_i(\mathbf{w}) + \sum_{j \leq n} \alpha_j h_j(\mathbf{w}), \quad \text{with each } \alpha_j \geq 0.$$

Roadmap


Strong duality
min $||w||^2$
s.t. ...
P: SVM's optimization problem
$L(w,\alpha) = \ldots$
P's Lagrangian
$w^* = \min_w L(w,\alpha^*)$
P's optimal solution
max $\theta(\alpha)$
s.t. ...
P's dual
$\alpha^* = \cdots$
P's dual's optimal solution
KKT conditions
Quadratic optimization on simplices
$w^* is\ "sparse"$

The duals

$$\mathcal{P}: \quad \begin{array}{lll} \min & f(\mathbf{w}), & \mathbf{w} \in \mathbb{R}^d \\ \text{s.t.} & g_i(\mathbf{w}) = 0, & i \leq m \\ & h_j(\mathbf{w}) \leq 0, & j \leq n \end{array}$$

Observe the following:

$$\mathcal{L}(\mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\alpha}) \leq f(\mathbf{w}).$$

In particular, for any $\boldsymbol{\lambda}, \boldsymbol{\alpha}$ with each $\alpha_j \geq 0$,

$$\theta(\boldsymbol{\lambda}, \boldsymbol{\alpha}) := \inf_{\mathbf{w} \in dom(\mathcal{P})} \mathcal{L}(\mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\alpha}) \leq \mathcal{L}(\mathbf{w}^*, \boldsymbol{\lambda}, \boldsymbol{\alpha}) \leq f(\mathbf{w}^*), \tag{1}$$

where $\mathbf{w}^*$ is an optimal solution for $\mathcal{P}$.

The dual problem

So $\theta(\boldsymbol{\lambda}, \boldsymbol{\alpha})$ estimates a lower bound for $f(\mathbf{w}^*)$.

What is then the best lower bound $\theta(\boldsymbol{\lambda}, \boldsymbol{\alpha})$ we can get for $f(\mathbf{w}^*)$?

$$\tilde{\mathcal{P}}: \quad \begin{array}{ll} \max & \theta(\boldsymbol{\lambda}, \boldsymbol{\alpha}), \quad \boldsymbol{\lambda}, \boldsymbol{\alpha} \in \mathbb{R}^{m+n} \\ \text{s.t.} & \alpha_i \geq 0, \quad i \leq n \end{array}$$

We refer to $\tilde{\mathcal{P}}$ as the dual of $\mathcal{P}$. We also refer to $\mathcal{P}$ and $\tilde{\mathcal{P}}$ as the primal and the dual problems, respectively.

Let $\boldsymbol{\lambda}^*, \boldsymbol{\alpha}^*$ be an optimal solution for $\tilde{\mathcal{P}}$. Then

$$\theta(\boldsymbol{\lambda}^*, \boldsymbol{\alpha}^*) \leq f(\mathbf{w}^*). \qquad \text{Weak duality.}$$

When $\theta(\boldsymbol{\lambda}^*, \boldsymbol{\alpha}^*) = f(\mathbf{w}^*)$, then we say that strong duality holds.

Slater's condition

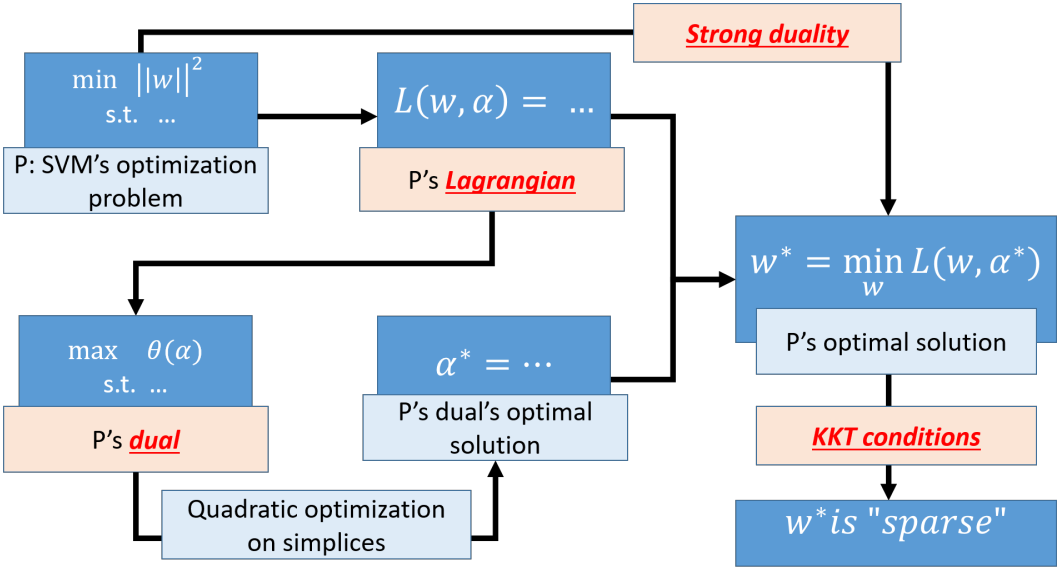
$$\mathcal{P}: \quad \begin{array}{lll} \min & f(\mathbf{w}), & \mathbf{w} \in \mathbb{R}^d \\ \text{s.t.} & g_i(\mathbf{w}) = 0, & i \leq m \\ & h_j(\mathbf{w}) \leq 0, & j \leq n \end{array}$$

If f and h_j are convex and g_i is affine, then:

Slater's condition: If there is a feasible $\mathbf{w}$ such that $h_j(\mathbf{w}) < 0$, for all $j \leq n$, then strong duality holds for $\mathcal{P}$.

We will later see that the optimization problem that is solved for training SVMs fulfills Slater's condition. Therefore, strong duality holds for that problem.

Roadmap


Strong duality
$\min \ ||w||^2$
s.t. ...
P: SVM's optimization problem
$L(w, \alpha) = \ ...$
P's Lagrangian
$w^* = \min_w L(w, \alpha^*)$
P's optimal solution
$\max \ \theta(\alpha)$
s.t. ...
P's dual
$\alpha^* = \cdots$
P's dual's optimal solution
KKT conditions
Quadratic optimization on simplices
$w^* is \ "sparse"$

Complementary slackness

What happens when strong duality holds?

$$
\begin{aligned}
f(\mathbf{w}^*) &= \theta(\boldsymbol{\lambda}^*, \boldsymbol{\alpha}^*) \\
&= \inf_{\mathbf{w}} \left(f(\mathbf{w}) + \sum_{i \leq m} \lambda_i^* g_i(\mathbf{w}) + \sum_{i \leq n} \alpha_i^* h_i(\mathbf{w}) \right) \\
&\leq f(\mathbf{w}^*) + \sum_{i \leq m} \lambda_i^* g_i(\mathbf{w}^*) + \sum_{i \leq n} \alpha_i^* h_i(\mathbf{w}^*) \\
&\leq f(\mathbf{w}^*).
\end{aligned}
$$

But since $f(\mathbf{w}^*) = f(\mathbf{w}^*)$, two things must happen:

- $\mathbf{w}^* = \min_{\mathbf{w}} \mathcal{L}(\mathbf{w}, \boldsymbol{\lambda}^*, \boldsymbol{\alpha}^*)$.
- Complementary slackness: $\sum_{j \leq n} \alpha_j^* h_j(\mathbf{w}^*) = 0$. This means that $\alpha_j^* h_j(\mathbf{w}^*) = 0$, for each $j \leq n$.

Karush Kuhn Tucker conditions: necessary conditions for an optimal solution

$$\begin{array}{rlll} & \min & f(\mathbf{w}), & \mathbf{w} \in \mathbb{R}^d \\ \mathcal{P}: & \text{s.t.} & g_i(\mathbf{w}) = 0, & i \leq m \\ & & h_j(\mathbf{w}) \leq 0, & j \leq n \end{array}$$

Any tuple $\mathbf{w}^*, \boldsymbol{\lambda}^*, \boldsymbol{\alpha}^*$ that optimizes the primal and the dual with strong duality must satisfy the following conditions:

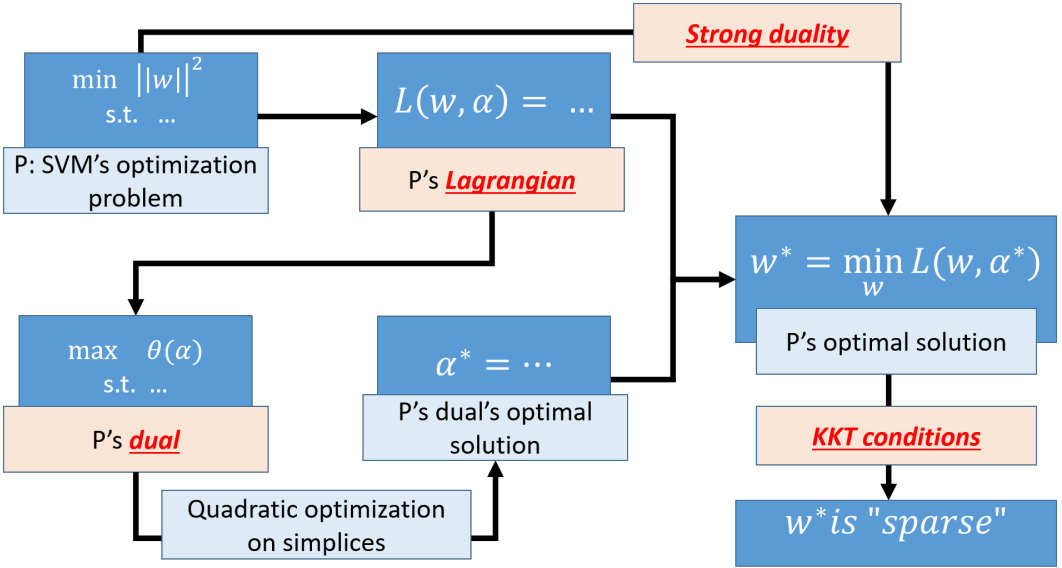
- (Feasible) $g_i(\mathbf{w}^*) = 0$, for $i \leq m$.
- (Feasible) $h_j(\mathbf{w}^*) \leq 0$, for $j \leq n$.
- (Complementary slackness) $\alpha_j^* h_j(\mathbf{w}^*) = 0$, for each $j \leq n$.
- (Minimizes Lagrangian) $\frac{\partial}{\partial \mathbf{w}} \mathcal{L}(\mathbf{w}, \boldsymbol{\lambda}^*, \boldsymbol{\alpha}^*) \mid_{\mathbf{w}=\mathbf{w}^*} = 0$.

KKT conditions are sufficient for convex optimization problems!

$$\mathcal{P}: \quad \begin{array}{lll} \min & f(\mathbf{w}), & \mathbf{w} \in \mathbb{R}^d \\ \text{s.t.} & g_i(\mathbf{w}) = 0, & i \leq m \\ & h_j(\mathbf{w}) \leq 0, & j \leq n \end{array}$$

If f is convex, h_j is convex, and g_i is affine, then any tuple $\mathbf{w}^*, \boldsymbol{\lambda}^*, \boldsymbol{\alpha}^*$ that meets the KKT conditions yields an optimal solution with strong duality.

Roadmap


Strong duality
min ||w||^2
s.t. ...
P: SVM's optimization problem
L(w, α) = ...
P's Lagrangian
w* = min_w L(w, α*)
P's optimal solution
max θ(α)
s.t. ...
P's dual
α* = ⋯
P's dual's optimal solution
KKT conditions
Quadratic optimization on simplices
w* is "sparse"

Support Vector Machines (SVM)

Idea: linear classifier with margin and feature transformation.

Transformation from original feature space to nonlinear feature space.

$$\mathbf{y}_i = \phi(\mathbf{x}_i) \quad \text{e.g. Polynomial, Radial Basis Function, ...}$$

$$\phi : \mathbb{R}^d \to \mathbb{R}^e \text{ with } d \ll e$$

$$z_i = \begin{cases} +1 \\ -1 \end{cases} \quad \text{if } \mathbf{x}_i \text{ in class } \begin{cases} 1 \\ 2 \end{cases}$$

Training vectors should be linearly separable after mapping!

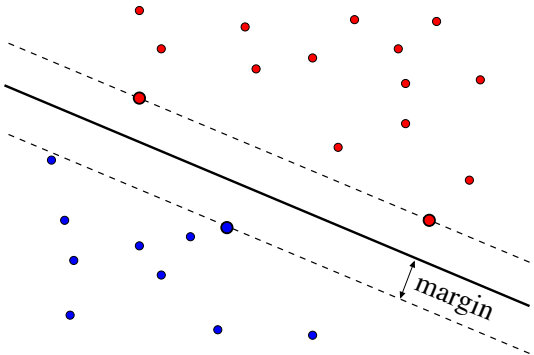
Linear discriminant function:

$$g(\mathbf{y}) = \mathbf{w}^\mathsf{T}\mathbf{y} + w_0$$

Support Vector Machine (SVM)

Find hyperplane that maximizes the **margin** m with

$$z_i\, g(\mathbf{y}_i) = z_i(\mathbf{w}^\mathsf{T}\mathbf{y} + w_0) \geq m \quad \text{for all } \mathbf{y}_i \in \mathcal{Y}$$



Vectors $\mathbf{y}_i$ with $z_i g(\mathbf{y}_i) = m$ are the **support vectors**.

Maximal Margin Classifier

Constraints for classification

$$z_i = \begin{cases} +1 & \mathbf{w}^\mathsf{T}\mathbf{y}_i + w_0 \geq m \\ -1 & \mathbf{w}^\mathsf{T}\mathbf{y}_i + w_0 \leq -m \end{cases} \qquad \forall i$$

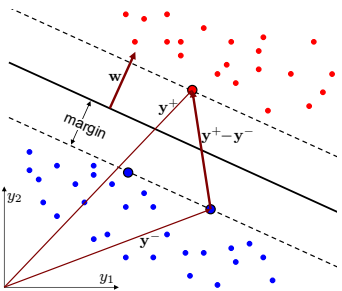
Objective: maximize margin m such that the joint conditions $z_i\,(\mathbf{w}^\mathsf{T}\mathbf{y}_i + w_0) \geq m,\ 1 \leq i \leq n$ are met.

What is the margin m ?

Consider two points $\mathbf{y}^+, \mathbf{y}^-$ of class 1,2 which are located on both sides of the margin boundaries ($\mathbf{w}^\mathsf{T}\mathbf{y}^+ + w_0 = m, \mathbf{w}^\mathsf{T}\mathbf{y}^- + w_0 = -m$) and project the difference vector $\mathbf{y}^+ - \mathbf{y}^-$ onto the normal of the hyperplane.

$$2 \times \text{margin} = \frac{\mathbf{w}^\mathsf{T}}{\|\mathbf{w}\|}(\mathbf{y}^+ - \mathbf{y}^-)$$

$$= \frac{1}{\|\mathbf{w}\|}\left(+\mathbf{w}^\mathsf{T}\mathbf{y}^+ + w_0 - (\mathbf{w}^\mathsf{T}\mathbf{y}^- + w_0)\right) = 2\frac{m}{\|\mathbf{w}\|}$$



Invariance of Maximal Margin Classifier

Rescaling of margin

The relation margin $= \frac{m}{\|\mathbf{w}\|}$ suggests that we have two parameters m and $\|\mathbf{w}\|$ to specify the optimization problem.

Problem: Scaling m is compensated by scaling the length of the weight vector, i.e.,

$$(\mathbf{w}, w_0) \leftarrow (\lambda\mathbf{w}, \lambda w_0),$$
$$m \leftarrow \lambda m$$

does neither change the margin nor the inequality constraints!

Two equivalent solutions for well-posedness

1. geometric margin problem: maximize m for $\|\mathbf{w}\| = 1$.
2. functional margin problem: minimize $\|\mathbf{w}\|$ for $m = 1$

SVM Lagrangian for functional margin formulation

Objective: Minimize $||\mathbf{w}||$ for a given margin $m = 1$

$$\begin{array}{lrcl} \text{minimize} & \mathcal{T}(\mathbf{w}) & = & \frac{1}{2}\mathbf{w}^{\mathsf{T}}\mathbf{w} \\ \text{subject to} & \forall i: \; z_i(\mathbf{w}^{\mathsf{T}}\mathbf{y}_i + w_0) & \geq & 1 \end{array}$$

Generalized Lagrange Function:

$$L(\mathbf{w}, w_0, \boldsymbol{\alpha}) = \frac{1}{2}\mathbf{w}^{\mathsf{T}}\mathbf{w} - \sum_{i=1}^{n} \alpha_i \left[z_i(\mathbf{w}^{\mathsf{T}}\mathbf{y}_i + w_0) - 1 \right]$$

Functional and geometric margin: The problem formulation with margin $m = 1$ is called the functional margin setting; The original formulation refers to the geometric margin.

Stationarity of Lagrangian

Extremality condition:

$$\frac{\partial L(\mathbf{w}, w_0, \boldsymbol{\alpha})}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i \leq n} \alpha_i z_i \mathbf{y}_i = 0 \quad \Rightarrow \quad \mathbf{w} = \sum_{i \leq n} \alpha_i z_i \mathbf{y}_i$$

$$\frac{\partial L(\mathbf{w}, w_0, \boldsymbol{\alpha})}{\partial w_0} = -\sum_{i \leq n} \alpha_i z_i = 0$$

Resubstituting $\frac{\partial L}{\partial \mathbf{w}} = 0$, $\frac{\partial L}{\partial w_0} = 0$ into the Lagrangian function $L(\mathbf{w}, w_0, \boldsymbol{\alpha})$ yields

$$\begin{aligned} L(\mathbf{w}, w_0, \boldsymbol{\alpha}) &= \frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{w} - \sum_{i \leq n} \alpha_i \left[z_i(\mathbf{w}^\mathsf{T}\mathbf{y}_i + w_0) - 1 \right] \\ &= \frac{1}{2} \sum_{i \leq n} \sum_{j \leq n} \alpha_i \alpha_j z_i z_j \mathbf{y}_i^\mathsf{T} \mathbf{y}_j - \sum_{i \leq n} \sum_{j \leq n} \alpha_i \alpha_j z_i z_j \mathbf{y}_i^\mathsf{T} \mathbf{y}_j + \sum_{i \leq n} \alpha_i \\ &= \sum_{i \leq n} \alpha_i - \frac{1}{2} \sum_{i \leq n} \sum_{j \leq n} \alpha_i \alpha_j z_i z_j \mathbf{y}_i^\mathsf{T} \mathbf{y}_j \qquad \text{(note the scalar product!)} \end{aligned}$$

Dual Problem

The **Dual Problem** for support vector learning is

$$\text{maximize} \quad W(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} z_i z_j \alpha_i \alpha_j \mathbf{y}_i^{\mathsf{T}} \mathbf{y}_j$$

$$\text{subject to} \quad \forall i: \ \alpha_i \geq 0 \quad \wedge \quad \sum_{i=1}^{n} z_i \alpha_i = 0$$

The optimal hyperplane $\mathbf{w}^*, w_0^*$ is given by

$$\mathbf{w}^* = \sum_{i=1}^{n} \alpha_i^* z_i \mathbf{y}_i, \quad w_0^* = -\frac{1}{2} \left(\min_{i:z_i=1} \mathbf{w}^{*\mathsf{T}} \mathbf{y}_i + \max_{i:z_i=-1} \mathbf{w}^{*\mathsf{T}} \mathbf{y}_i \right)$$

where $\boldsymbol{\alpha}^*$ are the optimal Lagrange multipliers maximizing the Dual Problem.

The optimal bias w_0^* is determined by the support vector constraints $\min_{i:z_i=1} \mathbf{w}^{*\mathsf{T}} \mathbf{y}_i + w_0^* = 1$ and $\max_{i:z_i=-1} \mathbf{w}^{*\mathsf{T}} \mathbf{y}_i + w_0^* = -1$. Adding both constraints and solving for w_0^* yields the formula.

Support Vectors

The **Kuhn-Tucker Conditions** for the maximal margin SVM are

$$\begin{aligned} \alpha_i^*(z_i g^*(\mathbf{y}_i) - 1) &= 0, & i = 1, \ldots, n \\ \alpha_i^* &\geq 0, & i = 1, \ldots, n \\ z_i g^*(\mathbf{y}_i) - 1 &\geq 0, & i = 1, \ldots, n \end{aligned}$$

The first one is known as the **Kuhn-Tucker complementary condition**. The conditions imply

$$\begin{aligned} z_i g^*(\mathbf{y}_i) = 1 &\Rightarrow \alpha_i^* \geq 0 && \text{Support Vectors (SV)} \\ z_i g^*(\mathbf{y}_i) \neq 1 &\Rightarrow \alpha_i^* = 0 && \text{Non Support Vectors} \end{aligned}$$

Optimal Decision Function

Sparsity:

$$
\begin{aligned}
g^*(\mathbf{y}) &= \mathbf{w}^{*\mathsf{T}}\mathbf{y} + w_0^* = \sum_{i=1}^{n} z_i \alpha_i^* \mathbf{y}_i{}^{\mathsf{T}}\mathbf{y} + w_0^* \\
&= \sum_{i \in \mathsf{SV}} z_i \alpha_i^* \mathbf{y}_i{}^{\mathsf{T}}\mathbf{y} + w_0^*
\end{aligned}
$$

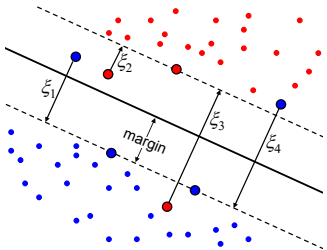
Remark: only few support vectors enter the sum to evaluate the decision function! $\Rightarrow$ efficiency and interpretability

Optimal margin: $\mathbf{w}^{\mathsf{T}}\mathbf{w} = \sum_{i \in \mathsf{SV}} \alpha_i^*$

Non-separable case: Soft Margin SVM

How to treat samples that violate the constraint?

$$z_i \left(\mathbf{w}^\mathsf{T}\mathbf{y}_i + w_0\right) < m, \text{ for some } i$$



Answer: Introduce slack variables $\xi_i \geq 0$ such that

$$z_i \left(\mathbf{w}^\mathsf{T}\mathbf{y}_i + w_0\right) \geq m(1 - \xi_i)$$

Soft Margin SVM

Slack variables relax the margin constraints for the samples that violate the constraints.

Invariance: Since the constraint for sample i is relaxed by a fraction $1 - \xi_i$ the invariance still exists.

Consider again two points $\mathbf{y}^+, \mathbf{y}^-$ of class 1,2 which are located on both sides of the margin boundaries ($\mathbf{w}^\mathsf{T}\mathbf{y}^+ + w_0 = m(1-\xi^+), \mathbf{w}^\mathsf{T}\mathbf{y}^- + w_0 = -m(1-\xi^-)$) and project the difference vector $\mathbf{y}^+ - \mathbf{y}^-$ onto the normal of the hyperplane.

$$2 \times \text{margin} = \frac{\mathbf{w}^\mathsf{T}}{\|\mathbf{w}\|}(\mathbf{y}^+ - \mathbf{y}^-)$$

$$= \frac{1}{\|\mathbf{w}\|}\left(+\mathbf{w}^\mathsf{T}\mathbf{y}^+ + w_0 - (\mathbf{w}^\mathsf{T}\mathbf{y}^- + w_0)\right) = 2\frac{m}{\|\mathbf{w}\|}\left(1 - \frac{\xi^+ + \xi^-}{2}\right)$$

Functional Soft Margin

Support Vectors $\mathbf{y}_i$ are characterized by active constraints

$$z_i(\mathbf{w}^{\mathsf{T}}\mathbf{y}_i + w_0) = m(1 - \xi_i) \ .$$

margin $= \dfrac{m}{\|\mathbf{w}\|}\left(1 - \dfrac{\xi^+ + \xi^-}{2}\right)$ is invariant under rescaling

$$m \leftarrow \lambda m, \ \mathbf{w} \leftarrow \lambda \mathbf{w}, \ w_0 \leftarrow \lambda w_0.$$

functional margin requires to define the scale as $m = 1$ and to minimize the length of the weight vector $\|\mathbf{w}\|$.

Learning the Soft Margin SVM

Slack variables are penalized by L_1 norm.

$$
\begin{array}{lrcl}
\text{minimize} & \mathcal{T}(\mathbf{w}, \boldsymbol{\xi}) & = & \frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{w} + C\sum_{i=1}^{n}\xi_i \\
\text{subject to } \forall i: & z_i(\mathbf{w}^\mathsf{T}\mathbf{y}_i + w_0) & \geq & 1 - \xi_i \\
& \xi_i & \geq & 0
\end{array}
$$

C controls the amount of constraint violations vs. margin maximization!

Lagrange function for soft margin SVM

$$
\begin{aligned}
L(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) &= \frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{w} + C\sum_{i=1}^{n}\xi_i \\
&- \sum_{i=1}^{n}\alpha_i\left[z_i(\mathbf{w}^\mathsf{T}\mathbf{y}_i + w_0) - 1 + \xi_i\right] - \sum_{i=1}^{n}\beta_i\xi_i
\end{aligned}
$$

Stationarity of Primal Problem

Differentiation of primal Lagrange function

$$\begin{aligned}
\frac{\partial L(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta})}{\partial \mathbf{w}} &= \mathbf{w} - \sum_{i=1}^{n} \alpha_i z_i \mathbf{y}_i = 0 \quad \Rightarrow \quad \mathbf{w} = \sum_{i=1}^{n} \alpha_i z_i \mathbf{y}_i \\
\frac{\partial L(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta})}{\partial \xi_i} &= C - \alpha_i - \beta_i = 0 \\
\frac{\partial L(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta})}{\partial w_0} &= -\sum_{i=1}^{n} \alpha_i z_i = 0
\end{aligned}$$

Resubstituting stationarity conditions into Lagrangian

$$\begin{aligned}
L(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) &= \frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{w} + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \beta_i \xi_i \\
&\quad - \sum_{i=1}^{n} \alpha_i \left(z_i(\mathbf{w}^\mathsf{T}\mathbf{y}_i + w_0) - 1 + \xi_i \right)
\end{aligned}$$

Stationarity of Primal Problem

$$
\begin{aligned}
L(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) &= \tfrac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j z_i z_j \mathbf{y}_i^{\mathsf{T}}\mathbf{y}_j + C\sum_{i=1}^{n}\xi_i \\
&\quad - \sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j z_i z_j \mathbf{y}_i^{\mathsf{T}}\mathbf{y}_j \\
&\quad + \sum_{i=1}^{n}\alpha_i(1-\xi_i) - \sum_{i=1}^{n}\beta_i\xi_i \\
&= \sum_{i=1}^{n}\alpha_i - \tfrac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j z_i z_j \mathbf{y}_i^{\mathsf{T}}\mathbf{y}_j \\
&\quad + \sum_{i=1}^{n}\underbrace{(C-\alpha_i-\beta_i)}_{=\frac{\partial L}{\partial \xi_i}=0}\xi_i \\
&= \sum_{i=1}^{n}\alpha_i - \tfrac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j z_i z_j \mathbf{y}_i^{\mathsf{T}}\mathbf{y}_j
\end{aligned}
$$

The dual of the soft SVM Lagrangian has the same form as the dual Lagrangian in the hard SVM case. The differences arise in the dual constraints which include the knowledge on the constraint violations!

Constraints of the Dual Problem

The dual objective function is the same as for the maximal margin SVM. The only difference is the constraint

$$\frac{\partial L(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta})}{\partial \xi_i} = C - \alpha_i - \beta_i = 0$$

Together with $\beta_i \geq 0$ it implies

$$\alpha_i \leq C$$

The Kuhn-Tucker complementary conditions

$$\begin{aligned} \alpha_i(z_i(\mathbf{w}^\mathsf{T}\mathbf{y}_i + w_0) - 1 + \xi_i) &= 0, \qquad i = 1, \ldots, n \\ \xi_i(\alpha_i - C) &= 0, \qquad i = 1, \ldots, n \end{aligned}$$

imply that nonzero slack variables can only occur when $\alpha_i = C$.

Dual Problem of Soft Margin SVM

The **Dual Problem** for support vector learning is

$$\begin{aligned} &\text{maximize} && W(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} z_i z_j \alpha_i \alpha_j \mathbf{y}_i^\top \mathbf{y}_j \\ &\text{subject to} && \sum_{j=1}^{n} z_j \alpha_j = 0 \ \wedge \ \forall i \ C \geq \alpha_i \geq 0 \end{aligned}$$

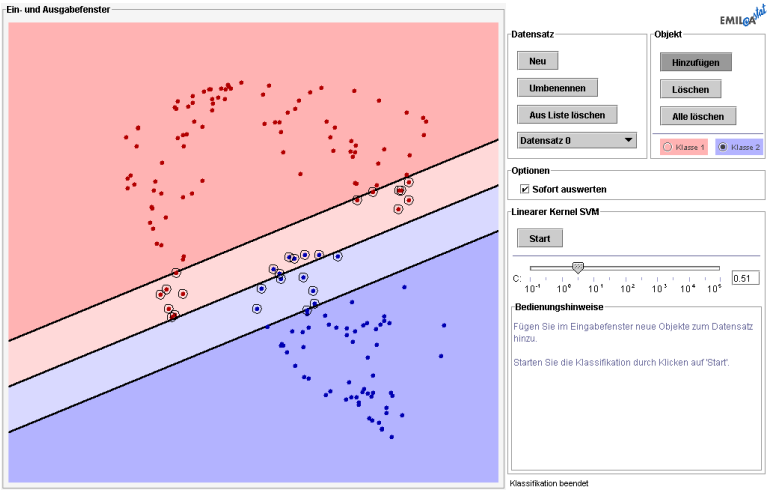
The optimal hyperplane $\mathbf{w}^*$ is given by

$$\mathbf{w}^* = \sum_{i=1}^{n} \alpha_i^* z_i \mathbf{y}_i$$

where $\boldsymbol{\alpha}^*$ are the optimal Lagrange multipliers maximizing the Dual Problem.

$\alpha_i^* > 0$ holds only for **support vectors**.

Applet HTML Page



Linear Programming Support Vector Machines

Idea: Minimize an estimate of the number of positive multipliers $\sum_{i=1}^{n} \alpha_i$ which improves bounds on the generalization error.

The **Lagrangian** for the LP-SVM is

$$\begin{aligned} \text{minimize} \quad & W(\boldsymbol{\alpha}, \xi) = \sum_{i=1}^{n} \alpha_i + C \sum_{i=1}^{n} \xi_i \\ \text{subject to } \forall i: \quad & z_i \left[\sum_{j=1}^{n} \alpha_j \mathbf{y}_i^{\mathsf{T}} \mathbf{y}_j + w_0 \right] \geq 1 - \xi_i, \\ & \alpha_i \geq 0, \; \xi_i \geq 0 \end{aligned}$$

Advantage: efficient LP solver can be used.

Disadvantage: theory is not as well understood as for standard SVMs.