Machine Learning

A Geometric Approach

Linear Classification:
Support Vector Machines (SVM)

Professor Liang Huang

some slides from Alex Smola (CMU)

Daume book
Chap 7.7
Linear Separator

Ham

Spam
Linear Separator

Ham

Spam
From Perceptron to SVM

1959: Rosenblatt invention
1962: Novikoff proof
1964: Vapnik/Chervonenkis

1969: fall of USSR

1997: Cortes/Vapnik SVM
1999: Freund/Schapire voted/avg: revived
2002: Collins structured
2003: Crammer/Singer MIRA
2005*: McDonald/Crammer/Pereira structured MIRA
2006: Singer group aggressive
2007-2010*: Singer group Pegasos

1997: +max margin +kernels +soft-margin

2007: online approx. subgradient descent
2006: minibatch

AT&T Research
ex-AT&T and students

*mentioned in lectures but optional
(others papers all covered in detail)
From Perceptron to SVM

1959 Rosenblatt invention
1962 Novikoff proof
1964 Vapnik

1969 Chervonenkis fall of USSR

1974 Inseparable case

1997 Cortes/Vapnik SVM +max margin +kernels +soft-margin

1999 Freund/Schapire voted/avg: revived

2002 Collins structured

2003 Crammer/Singer MIRA

2006 Singer group aggressive
2007-2010 Singer group Pegasos

2005 McDonald/Crammer/Pereira structured MIRA

AT&T Research ex-AT&T and students

+max margin +kernels +soft-margin


*mentioned in lectures but optional (others papers all covered in detail)
Large Margin Classifier

\[ \langle w, x \rangle + b \leq -1 \]

\[ \langle w, x \rangle + b \geq 1 \]

linear function

\[ f(x) = \langle w, x \rangle + b \]
Large Margin Classifier

\[ f(x) = \langle w, x \rangle + b \]

linear function

\[ \langle w, x \rangle + b \leq -1 \]

\[ \langle w, x \rangle + b \geq 1 \]
Why large margins?

- Maximum robustness relative to uncertainty
- Symmetry breaking
- Independent of correctly classified instances
- Easy to find for easy problems
Feature Map $\Phi$

- SVM is often used with kernels
Feature Map $\Phi$

- SVM is often used with kernels
Large Margin Classifier

geometric margin:
\[ \langle w, x \rangle + b = 1 \]

functional margin:
\[ y_i (w \cdot x_i) \]

geometric margin:
\[ \frac{y_i (w \cdot x_i)}{\|w\|} = \frac{1}{\|w\|} \]
Large Margin Classifier

SVM objective (max version):
\[
\max_{w} \frac{1}{\|w\|} \quad \text{s.t. } \forall (x, y) \in D, y(w \cdot x) \geq 1
\]

max. geometric margin
s.t. functional margin is at least 1
Large Margin Classifier

Q1: what if we want functional margin of 2?

SVM objective (max version):

\[
\max_w \frac{1}{\|w\|} \quad \text{s.t.} \quad \forall (x, y) \in D, y(w \cdot x) \geq 1
\]

max. geometric margin
s.t. functional margin
is at least 1
Large Margin Classifier

Q1: what if we want functional margin of 2?
Q2: what if we want geometric margin of 1?

SVM objective (max version):

\[
\max_w \frac{1}{\|w\|} \quad \text{s.t.} \quad \forall (x, y) \in D, y(w \cdot x) \geq 1
\]

max. geometric margin
s.t. functional margin is at least 1
Large Margin Classifier

SVM objective (min version):

$$\min_w \|w\| \quad \text{s.t.} \quad \forall (x, y) \in D, y(w \cdot x) \geq 1$$

min. weight vector s.t. functional margin is at least 1
SVM objective (min version):

\[
\min \| \mathbf{w} \| \quad \text{s.t.} \quad \forall (\mathbf{x}, y) \in D, y(\mathbf{w} \cdot \mathbf{x}) \geq 1
\]

interpretation: small models generalize better

\[
\langle \mathbf{w}, \mathbf{x} \rangle \geq 1
\]

min. weight vector s.t. functional margin is at least 1
Large Margin Classifier

\[ \langle w, x \rangle \leq -1 \quad \text{for green points} \]
\[ \langle w, x \rangle \geq 1 \quad \text{for red points} \]

SVM objective (min version):
\[
\min_{w} \frac{1}{2} \| w \|^2 \quad \text{s.t.} \quad \forall (x, y) \in D, y(w \cdot x) \geq 1
\]

min. weight vector s.t. functional margin is at least 1
Large Margin Classifier

SVM objective (min version):
\[
\min_w \frac{1}{2} \|w\|^2 \quad \text{s.t.} \quad \forall (x, y) \in D, y(w \cdot x) \geq 1
\]

min. weight vector

s.t. functional margin

is at least 1

\[|w| \text{ not differentiable, } |w|^2 \text{ is.}\]
**SVM vs. MIRA**

- SVM: min weight vector to enforce functional margin of at least 1 on ALL EXAMPLES
- MIRA: min weight change to enforce functional margin of at least 1 on THIS EXAMPLE
- MIRA is 1-step or online approximation of SVM
- Aggressive MIRA $\rightarrow$ SVM as $p \rightarrow 1$

\[
\begin{align*}
\min_w & \frac{1}{2} \|w\|^2 \quad \text{s.t. } \forall (x, y) \in D, y(w \cdot x) \geq 1 \\
\min_{w'} & \|w' - w\|^2 \\
\text{s.t. } & w' \cdot x \geq 1
\end{align*}
\]
Convex Hull Interpretation

max. distance between convex hulls

Class $B$ | | Class $A$
Convex Hull Interpretation

max. distance between convex hulls

weight vector is determined by the support vectors alone

c.f. perceptron: \( \mathbf{w} = \sum_{(x, y) \in \text{errors}} y \cdot x \)

What about MIRA?
**Convex Hull Interpretation**

max. distance between convex hulls

Class $B$ | Class $A$

weight vector is determined by the support vectors alone

c.f. perceptron: $\mathbf{w} = \sum_{(x,y) \in \text{errors}} y \cdot \mathbf{x}$

how many support vectors in 2D?

what about MIRA?
Convex Hull Interpretation

max. distance between convex hulls

weight vector is determined by the support vectors alone

c.f. perceptron: \[ w = \sum_{(x,y) \in \text{errors}} y \cdot x \]

what about MIRA?
Convex Hull Interpretation

Convex Hull Interpretation

max. distance between convex hulls

weight vector is determined by the support vectors alone

c.f. perceptron: \[ w = \sum_{(x,y) \in \text{errors}} y \cdot x \]

why don’t use convex hulls for SVMs in practice??
Convex set. A point set \( C \in \mathbb{R}^d \) is convex if the line segment \([x,y]\) connecting any two points \(x\) and \(y\) in \(C\) lies entirely in \(C\).

Convex hull. Smallest convex set containing \(C\).

\[
\text{ch}(C) := \left\{ \sum_i \alpha_i x_i : x_i \in C, \alpha_i \geq 0, \sum_i \alpha_i = 1 \right\}.
\]
Convex set. A point set $C \in \mathbb{R}^d$ is convex if the line segment $[x, y]$ connecting any two points $x$ and $y$ in $C$ lies entirely in $C$.

Convex hull. Smallest convex set containing $C$.

$$
\text{ch}(C) := \left\{ \sum_{i} \alpha_i x_i : x_i \in C, \alpha_i \geq 0, \sum_{i} \alpha_i = 1 \right\}.
$$
Optimization

- Primal optimization problem
  \[ \min_{\mathbf{w}} \frac{1}{2} \| \mathbf{w} \|^2 \quad \text{s.t.} \quad \forall (\mathbf{x}, y) \in D, y(\mathbf{w} \cdot \mathbf{x}) \geq 1 \]

- Convex optimization: convex function over convex set!

- Quadratic prog.: quadratic function w/ linear constraints
MIRA as QP

- MIRA is a trivial QP; can be solved geometrically
- what about multiple constraints (e.g. minibatch)?

\[
\min_{\mathbf{w}'} \| \mathbf{w}' - \mathbf{w} \|^2 \\
\text{s.t. } \mathbf{w}' \cdot \mathbf{x} \geq 1
\]
Optimization

- Primal optimization problem
  \[ \min_w \frac{1}{2} \|w\|^2 \quad \text{s.t. } \forall (x, y) \in D, y(w \cdot x) \geq 1 \]

- Convex optimization: convex function over convex set!

- Lagrange function
  \[ L(w, \alpha) = \frac{1}{2} \|w\|^2 - \sum_i \alpha_i \left[ y_i \langle x_i, w \rangle + b \right] - 1 \]

Derivatives in \( w \) need to vanish
Optimization

• Primal optimization problem
\[
\min_w \frac{1}{2} \|w\|^2 \quad \text{s.t.} \quad \forall (x, y) \in D, y(w \cdot x) \geq 1
\]

• Convex optimization: convex function over convex set!

• Lagrange function
\[
L(w, \alpha) = \frac{1}{2} \|w\|^2 - \sum_i \alpha_i [y_i \langle x_i, w \rangle + b] - 1
\]

Derivatives in \( w \) need to vanish
\[
\partial_w L(w, \alpha) = w - \sum_i \alpha_i y_i x_i = 0
\]
\[
w = \sum_i y_i \alpha_i x_i
\]
Optimization

- Primal optimization problem
  \[
  \min_w \frac{1}{2} \|w\|^2 \quad \text{s.t.} \quad \forall (x, y) \in D, y(w \cdot x) \geq 1
  \]

- Convex optimization: convex function over convex set!

- Lagrange function
  \[
  L(w, \alpha) = \frac{1}{2} \|w\|^2 - \sum_i \alpha_i [y_i \langle x_i, w \rangle + b] - 1
  \]

Derivatives in \( w \) need to vanish

\[
\partial_w L(w, \alpha) = w - \sum_i \alpha_i y_i x_i = 0
\]

Model is a linear combo of a small subset of input (the support vectors) i.e., those with \( \alpha_i > 0 \).
Lagrangian & Saddle Point

- equality: \( \min x^2 \) s.t. \( x = 1 \)
- inequality: \( \min x^2 \) s.t. \( x \geq 1 \)
- Lagrangian: \( L(x, \alpha) = x^2 - \alpha(x-1) \)
- derivative in \( x \) need to vanish
- optimality is at saddle point with \( \alpha \)
- \( \min_x \) in primal \( \Rightarrow \max_\alpha \) in dual
Constrained Optimization

- Quadratic Programming
- Quadratic Objective
- Linear Constraints

\[
\begin{align*}
\text{minimize} & \quad \frac{1}{2} \|w\|^2 \\
\text{subject to} & \quad y_i [\langle x_i, w \rangle + b] \geq 1
\end{align*}
\]

Karush–Kuhn–Tucker (KKT) condition (complementary slackness) optimal point is achieved at active constraints where \( \alpha_i > 0 \) (\( \alpha_i = 0 \) => inactive)

\[
\alpha_i [y_i \langle w, x_i \rangle + b] - 1 = 0
\]
KKT $\Rightarrow$ Support Vectors

\[
\begin{align*}
\text{minimize} & \quad \frac{1}{2} \|w\|^2 \\
\text{subject to} & \quad y_i [\langle x_i, w \rangle + b] \geq 1
\end{align*}
\]

\[
w = \sum_i y_i \alpha_i x_i
\]

Karush Kuhn Tucker (KKT) Optimality Condition

\[
\alpha_i \left[ y_i \langle w, x_i \rangle + b \right] - 1 = 0
\]

\[
\alpha_i = 0
\]

\[
\alpha_i > 0 \implies y_i \langle w, x_i \rangle + b = 1
\]
Properties

\[ w = \sum_{i} y_i \alpha_i x_i \]

- Weight vector \( w \) as weighted linear combination of instances
- Only points on margin matter (ignore the rest and get same solution)
- Only inner products matter
  - Quadratic program
  - We can replace the inner product by a kernel
- Keeps instances away from the margin
Alternative: Dual Problem

• Lagrange function

\[ L(w, \alpha) = \frac{1}{2} \|w\|^2 - \sum_i \alpha_i [y_i \langle x_i, w \rangle - 1] \]

• Derivatives in \( w \) need to vanish

\[ \partial_w L(w, \alpha) = w - \sum_i \alpha_i y_i x_i = 0 \]

\[ w = \sum_i y_i \alpha_i x_i \]

• Plugging \( w \) back into \( L \) yields

\[ \max_{\alpha} - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle + \sum_i \alpha_i \]

subject to dual variables \( \alpha_i \geq 0 \)
Primal vs. Dual

**Primal**

\[
\begin{align*}
\text{minimize} \quad & \frac{1}{2} \|w\|^2 \\
\text{subject to} \quad & y_i \left( \langle x_i, w \rangle + b \right) \geq 1 \\
& w = \sum_i y_i \alpha_i x_i
\end{align*}
\]

**Dual**

\[
\begin{align*}
\text{maximize} \quad & - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle + \sum_i \alpha_i \\
\text{subject to} \quad & \text{dual variables } \alpha_i \geq 0
\end{align*}
\]
Solving the optimization problem

• Dual problem

\[
\text{maximize } \prod_{\alpha} \left( -\frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle + \sum_i \alpha_i \right)
\]

subject to \( \sum_{i} \text{dual variables } \alpha_i \geq 0 \)

• If problem is small enough (1000s of variables) we can use off-the-shelf solver (CVXOPT, CPLEX, OOQP, LOQO)

• For larger problem use fact that only SVs matter and solve in blocks (active set method).
Quadratic Program in Dual

- Dual problem

\[
\begin{align*}
\text{maximize} & \quad \frac{1}{2} \alpha^T Q \alpha - \alpha^T b \\
\text{subject to} & \quad \alpha \geq 0
\end{align*}
\]

Q: what’s the Q in SVM primal? how about Q in SVM dual?

- Quadratic Programming
  - Objective: Quadratic function
  - Q is positive semidefinite
  - Constraints: Linear functions

- Methods
  - Gradient Descent
  - Coordinate Descent
    - aka., Hildreth Algorithm
  - Sequential Minimal Optimization (SMO)
Convex QP

• if $Q$ is positive (semi)definite, i.e., $x^T Q x \geq 0$, then convex QP $\Rightarrow$ local min/max is global min/max
• if $Q = 0$, it reduces to linear programming
• general QP is NP-hard; convex QP is polynomial
Convex QP

- if $Q$ is positive (semi)definite, i.e., $x^TQx \geq 0$, then convex QP $\Rightarrow$ local min/max is global min/max
- if $Q = 0$, it reduces to linear programming
- general QP is NP-hard; convex QP is polynomial
Hildreth Algorithm

• idea 1:
  • update one coordinate while fixing all other coordinates
  • e.g., update coordinate $i$ is to solve:
    $$\argmax_{\alpha_i} - \frac{1}{2} \alpha^T Q \alpha - \alpha^T b$$
    subject to $\alpha \geq 0$

Quadratic function with only one variable
Maximum $\Rightarrow$ first-order derivative is 0
Hildreth Algorithm

- idea 2:
  - choose another coordinate and repeat until meet stopping criterion
  - reach maximum or
  - increase between 2 consecutive iterations is very small or
  - after some # of iterations
- how to choose coordinate: sweep pattern
  - Sequential:
    - 1, 2, ..., n, 1, 2, ..., n, ...
    - 1, 2, ..., n, n-1, n-2, ..., 1, 2, ...
  - Random: permutation of 1, 2, ..., n
  - Maximal Descent
    - choose i with maximal descent in objective
Hildreth Algorithm

initialize $\alpha_i = 0$ for all $i$

repeat

pick $i$ following sweep pattern

solve

$$\alpha_i \leftarrow \arg\max_{\alpha_i} - \frac{1}{2} \alpha^T Q \alpha - \alpha^T b$$

subject to $\alpha \geq 0$

until meet stopping criterion
Hildreth Algorithm

\[
\text{maximize } -\frac{1}{2} \alpha^T \begin{pmatrix} 4 & 1 \\ 1 & 2 \end{pmatrix} \alpha - \alpha^T \begin{pmatrix} -6 \\ -4 \end{pmatrix}
\]

subject to \( \alpha \geq 0 \)

- choose coordinates
- 1, 2, 1, 2, ...
Hildreth Algorithm

• **pros:**
  • extremely simple
  • no gradient calculation
  • easy to implement

• **cons:**
  • converges slow, compared to other methods
Linear Separator

Ham

Spam
Linear Separator

Ham

Spam
Linear Separator

Ham

Spam
Linear Separator

Ham

Spam
Support Vector Machines: CLASSIFIERS
Large Margin Classifier

\[ \langle w, x \rangle + b \leq -1 \]

\[ \langle w, x \rangle + b \geq 1 \]

linear function

\[ f(x) = \langle w, x \rangle + b \]
Large Margin Classifier

\[ \langle w, x \rangle + b \leq -1 \]

\[ \langle w, x \rangle + b \geq 1 \]

Linear function

\[ f(x) = \langle w, x \rangle + b \]
Large Margin Classifier

\[ \langle w, x \rangle + b \leq -1 \]

\[ \langle w, x \rangle + b \geq 1 \]

linear function

\[ f(x) = \langle w, x \rangle + b \]

linear separator

is impossible
Theorem (Minsky & Papert)
Finding the minimum error separating hyperplane is NP hard
Theorem (Minsky & Papert)
Finding the minimum error separating hyperplane is NP hard
Adding slack variables

\[ \langle w, x \rangle + b \leq -1 + \xi \]

\[ \langle w, x \rangle + b \geq 1 - \xi \]

Convex optimization problem
Adding slack variables

Convex optimization problem

\[ \langle w, x \rangle + b \leq -1 + \xi \]

\[ \langle w, x \rangle + b \geq 1 - \xi \]

minimize amount of slack
Adding slack variables

• Hard margin problem

\[
\begin{align*}
\text{minimize} & \quad \frac{1}{2} \|w\|^2 \\
\text{subject to} & \quad y_i \left[ \langle w, x_i \rangle + b \right] \geq 1
\end{align*}
\]

• With slack variables

\[
\begin{align*}
\text{minimize} & \quad \frac{1}{2} \|w\|^2 + C \sum_i \xi_i \\
\text{subject to} & \quad y_i \left[ \langle w, x_i \rangle + b \right] \geq 1 - \xi_i \text{ and } \xi_i \geq 0
\end{align*}
\]

Problem is always feasible. Proof:

\[w = 0 \text{ and } b = 0 \text{ and } \xi_i = 1\] (also yields upper bound)
Intermezzo
Convex Programs for Dummies

• Primal optimization problem

$$\min_x f(x) \text{ subject to } c_i(x) \leq 0$$

• Lagrange function

$$L(x, \alpha) = f(x) + \sum_i \alpha_i c_i(x)$$

• First order optimality conditions in x

$$\partial_x L(x, \alpha) = \partial_x f(x) + \sum_i \alpha_i \partial_x c_i(x) = 0$$

• Solve for x and plug it back into L

$$\max_{\alpha} L(x(\alpha), \alpha)$$

(keep explicit constraints)
Primal optimization problem

\[
\text{minimize } \frac{1}{2} \| w \|^2 + C \sum_i \xi_i \\
\text{subject to } y_i [\langle w, x_i \rangle + b] \geq 1 - \xi_i \text{ and } \xi_i \geq 0
\]

Lagrange function

\[
L(w, b, \alpha) = \frac{1}{2} \| w \|^2 + C \sum_i \xi_i - \sum_i \alpha_i [y_i [\langle x_i, w \rangle + b] + \xi_i - 1] - \sum_i \eta_i \xi_i
\]

Optimality in \((w, \xi)\) is at saddle point with \(\alpha, \eta\)

Derivatives in \((w, \xi)\) need to vanish
Dual Problem

• Lagrange function

\[ L(w, b, \alpha) = \frac{1}{2} \|w\|^2 + C \sum_i \xi_i - \sum_i \alpha_i [y_i \langle x_i, w \rangle + b] + \xi_i - 1 - \sum_i \eta_i \xi_i \]

• Derivatives in \( w \) need to vanish

\[ \partial_w L(w, b, \xi, \alpha, \eta) = w - \sum_i \alpha_i y_i x_i = 0 \]
\[ \partial_b L(w, b, \xi, \alpha, \eta) = \sum_i \alpha_i y_i = 0 \]
\[ \partial_{\xi_i} L(w, b, \xi, \alpha, \eta) = C - \alpha_i - \eta_i = 0 \]

• Plugging terms back into \( L \) yields

\[ \maximize_{\alpha} - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle + \sum_i \alpha_i \]

subject dual variables \( \alpha_i \in [0, C] \)
Karush Kuhn Tucker Conditions

\[ L(w, b, \alpha) = \frac{1}{2} \|w\|^2 + C \sum_i \xi_i - \sum_i \alpha_i [y_i \langle x_i, w \rangle + b] + \xi_i - 1 - \sum_i \eta_i \xi_i \]

\[ \partial_w L(w, b, \xi, \alpha, \eta) = w - \sum_i \alpha_i y_i x_i = 0 \]

\[ w = \sum_i y_i \alpha_i x_i \]

\[ \alpha_i [y_i \langle w, x_i \rangle + b] + \xi_i - 1 = 0 \]

\[ \eta_i \xi_i = 0 \]

\[ 0 \leq \alpha_i = C - \eta_i \leq C \]

\[ \alpha_i = 0 \implies y_i \langle w, x_i \rangle + b \geq 1 \]

\[ 0 < \alpha_i < C \implies y_i \langle w, x_i \rangle + b = 1 \]

\[ \alpha_i = C \implies y_i \langle w, x_i \rangle + b \leq 1 \]