Machine Learning 11: Support Vector Machines, Aggregation

Master 2 Computer Science

Aurélien Garivier

2019-2020

Table of contents

1. Support Vector Machines
2. Super-learning: Ensemble Methods

Support Vector Machines

Margin for linear separation

- Training sample $S = \big\{(x_1, y_1), \dots, (x_m, y_m)\big\}$, where $x_i \in \mathbb{R}^d$ and $y_i \in \{\pm 1\}$.
- Linearly separable if there exists a halfspace $h = (w, b)$ such that $\forall i, y_i = \operatorname{sign}\big(\langle w, x_i \rangle + b\big)$.
- What is the best separating hyperplane for generalization?

Distance to hyperplane

If $\|w\| = 1$, then the distance from x to the hyperplane $h = (w, b)$ is $d(x, \mathcal{H}) = \big|\langle w, x \rangle + b\big|$.

Proof: Check that $\min\big\{\|x - v\|^2 : v \in h\big\}$ is reached at $v = x - \big(\langle w, x \rangle + b\big)w$.

Hard-SVM

Formulation 1:

$$\operatorname*{arg\,max}_{(w,b):\|w\|=1} \min_{1\le i\le m} \left|\langle w, x_i\rangle + b\right| \quad \text{such that } \forall i, y_i\big(\langle w, x_i\rangle + b\big) > 0 \; .$$

Formulation 2:

$$\min_{w,b} \|w\|^2 \quad \text{such that } \forall i, y_i\big(\langle w, x_i\rangle + b\big) \ge 1 \; .$$

Remark: b is not penalized.

Proposition

The two formulations are equivalent.

Proof of the useful implication: if (w_0, b_0) is the solution of Formulation 2, then $\hat{w} = \frac{w_0}{\|w_0\|}$, $\hat{b} = \frac{b_0}{\|w_0\|}$ is a solution of Formulation 1: if (w^*, b^*) is another solution, then letting $\gamma^* = \min_{1\le i\le m} y_i\big(\langle w, x_i\rangle + b\big)$ we see that $\left(\frac{w^*}{\gamma^*}, \frac{b^*}{\gamma^*}\right)$ satisfies the constraint of Formulation 2, hence $\|w_0\| \le \frac{\|w^*\|}{\gamma^*} = \frac{1}{\gamma^*}$ and thus $\min_{1\le i\le m} \left|\langle \hat{w}, x_i\rangle + \hat{b}\right| \ge \frac{1}{\|w_0\|} \ge \gamma^*$.

Sample Complexity

Definition

A distribution D over $\mathbb{R}^d \times \{\pm 1\}$ is *separable with a* (γ, ρ)*-margin* if there exists (w^*, b^*) such that $\|w^*\| = 1$ and with probability 1 on a pair $(X, Y) \sim D$, it holds that $\|X\| \leq \rho$ and $Y\big(\langle w^*, X\rangle + b\big) \geq \gamma$.

Remark: by multiplying the x_i by α, the margin is mutliplied by α.

Theorem

For any distribution D over $\mathbb{R}^d \times \{\pm 1\}$ that satisfies the (γ, ρ)-separability with margin assumption using a homogenous halfspace, with probability at least $1 - \delta$ over the training set of size m the $0 - 1$ loss of the output of Hard-SVM is at most

$$\sqrt{\frac{4(\rho/\gamma)^2}{m}} + \sqrt{\frac{2\log(2/\delta)]}{m}}\ .$$

Remark: depends on dimension d only thru ρ and γ.

Soft-SVM

When the data is not linearly separable, allow *slack variables* ξ_i:

$$\min_{w,b,\xi} \lambda\|w\|^2 + \frac{1}{m}\sum_{i=1}^{m} \xi_i \quad \text{such that } \forall i, y_i\big(\langle w, x_i\rangle + b\big) \geq 1 - \xi_i \text{ and } \xi_i \geq 0$$

$$= \min_{w,b} \lambda\|w\|^2 + L_S^{\mathrm{hinge}}(w,b) \quad \text{where } \ell^{\mathrm{hinge}}(u) = \max(0, 1-u)\ .$$

Theorem

Let D be a distribution over $B(0,\rho) \times \{\pm 1\}$. If $A(S)$ is the output of the soft-SVM algorithm on the sample S of D of size m,

$$\mathbb{E}\Big[L_D^{0-1}\big(A(S)\big)\Big] \leq \mathbb{E}\Big[L_D^{\mathrm{hinge}}\big(A(S)\big)\Big] \leq \inf_u L_D^{\mathrm{hinge}}(u) + \lambda\|u\|^2 + \frac{2\rho^2}{\lambda m}\ .$$

For every $B > 0$, setting $\lambda = \sqrt{\frac{2\rho^2}{B^2 m}}$ yields:

$$\mathbb{E}\Big[L_D^{0-1}\big(A(S)\big)\Big] \leq \mathbb{E}\Big[L_D^{\mathrm{hinge}}\big(A(S)\big)\Big] \leq \inf_{w:\|w\|\leq B} L_D^{\mathrm{hinge}}(w) + \sqrt{\frac{8\rho^2 B^2}{m}}\ .$$

Dual Form of the SVM Optimization Problem

To simplify, we consider only the homogeneous case of hard-SVM. Let

$$g(w) = \max_{\alpha \in [0,+\infty)^m} \sum_{i=1}^{m} \alpha_i \big(1 - y_i \langle w, x_i \rangle\big) = \begin{cases} 0 & \text{if } \forall i, y_i \langle w, x_i \rangle \geq 1, \\ +\infty & \text{otherwise .} \end{cases}$$

Then the hard-SVM problem is equivalent to

$$\begin{aligned} \min_{w : \forall i, y_i \langle w, x_i \rangle \geq 1} \frac{1}{2}\|w\|^2 &= \min_{w} \frac{1}{2}\|w\|^2 + g(w) \\ &= \min_{w} \max_{\alpha \in [0,+\infty)^m} \frac{1}{2}\|w\|^2 + \sum_{i=1}^{m} \alpha_i \big(1 - y_i \langle w, x_i \rangle\big) \\ &\overset{\text{min}-\text{max thm}}{=} \max_{\alpha \in [0,+\infty)^m} \min_{w} \frac{1}{2}\|w\|^2 + \sum_{i=1}^{m} \alpha_i \big(1 - y_i \langle w, x_i \rangle\big) \ . \end{aligned}$$

The inner min is reached at $w = \sum_{i=1}^{m} \alpha_i y_i x_i$ and can thus be written as

$$\max_{\alpha \in \mathbb{R}^m, \alpha \geq 0} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{1 \leq i,j \leq m} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle \ .$$

Support vectors

Still for the homogeneous case of hard-SVM:

Property

Let w_0 be a solution of and let $I = \{i : \left|\langle w_0, x_i \rangle\right| = 1\}$. There exist $\alpha_1, \ldots, \alpha_m$ such that

$$w_0 = \sum_{i \in I} \alpha_i x_i \ .$$

The dual problem involves the x_i only thru scalar products $\langle x_i, x_j \rangle$.

It is of size m (independent of the dimension d).

These computations can be extended to the non-homogeneous soft-SVM

$\rightarrow$ **Kernel trick**.

Numerically solving Soft-SVM

$f(w) = \frac{\lambda}{2}\|w\|^2 + L_S^{\mathrm{hinge}}(w)$ is λ-strongly convex.
$\rightarrow$ Stochastic Gradient Descent with learning rate $1/(\lambda t)$. Stochastic subgradient of $L_S^{\mathrm{hinge}}(w)$: $v_t = -y_{I_t} x_{I_t} \mathbb{1}\big\{ y_{I_t} \langle w, x_{I_t} \rangle < 1 \big\}$.

$$w_{t+1} = w_t - \frac{1}{\lambda t}(\lambda w_t + v_t) = \frac{t-1}{t} w_t - \frac{1}{\lambda t} v_t = -\frac{1}{\lambda t} \sum_{s=1}^{t} v_s \ .$$

Algorithm: SGD for Soft-SVM

Set $\theta_0 = 0$
for $t = 0 \dots T-1$ **do**
  Let $w_t = \frac{1}{\lambda t} \theta_t$
  Pick $I_t \sim \mathcal{U}\big(\{1, \dots, m\} \big)$
  if $y_{I_t} \langle w_t, x_{I_t} \rangle < 1$ **then**
    $\theta_{t+1} \leftarrow \theta_t + y_{I_t} x_{I_t}$
  else
    $\theta_{t+1} \leftarrow \theta_t$
return $\bar{w}_T = \frac{1}{T} \sum_{t=0}^{T-1} w_t$

Super-learning: Ensemble Methods

Aggregating Predictions from Weak Learners

Weak learners:

- Stumps
- Decision trees

High bias, high individual variance

But quick and light $\implies$ can be combined efficiently

Decision Trees: CART and co

Idea: recursive splitting of the feature space $\mathcal{X}$. *Inhomogeneity* of a cell:

- classification: 0 when all labels are equal, maximal when the labels are evenly distributed. Ex: if $p =$ frequency of label 1, $h(p) = \max(p, 1-p), p(1-p)$, binary entropy
- regression: empirical variance of the labels

1. Expansion phase: top-down

- Start with tree root $= \mathcal{X}$
- Repeat for each in-homogeneous leaf:
  - find variable v and threshold s such that splitting according to
    - [quantitative variable] $v < s$ versus $v \geq s$
    - [qualitative variable] $v \in s$ versus $v \notin s$

    improves most homogeneity
  - replace that leaf by a node with the two corresponding children
- Stop when all leaves are homogeneous or contain fewer that K data points

2. Pruning phase: bottom-up

- In each leaves' parent, test if the split is significant
- If not, remove the leaves: the parent is now a leave (and start again)

Outline

Support Vector Machines

Super-learning: Ensemble Methods

- Bagging
- Boosting

Bootstrap: a Resampling scheme

- Setting:
  - observation space $\mathcal{X}$, model $\mathcal{M} \subset \mathfrak{M}_1(\mathcal{X})$,
  - target: $\psi(P)$ for $P \in \mathfrak{M}$,
  - data: $S_m = (X_1, \ldots, X_m) \overset{iid}{\sim} P$,
  - empirical measure $P_m = \frac{1}{m}\sum_{i=1}^m \delta_{X_i}$ is "close to" P
  - statistic: $\psi(P_m)$
- Problem: how close is $\psi(P_m)$ from $\psi(P)$?
  If we had several samples, we could experiment...
- Idea: since P_n is close to P, we can use it as a substitute to P:
  $\tilde{X}_i \overset{iid}{\sim} P_m$
- Sampling from P_n amounts to *resampling with replacement* from S_m
- The distribution of the estimator $\psi(P_m)$ might be close to that of $\psi(\tilde{P}_m)$, where $\tilde{P}_m = \frac{1}{m}\sum_{i=1}^m \delta_{\tilde{X}_i}$
- We can "see" the distribution of $\psi(\tilde{P}_m)$ by forming a large number M of such "bootstrap samples".
- From this distribution we can build confidence intervals, etc. (needs to be justified theoretically!)

Bagging: Bootstrap Aggregating

Input:

Sample: $S_m = \big((X_1, Y_1), \dots, (X_m, Y_m)\big)$

Weak learner: $\Phi_m : S_m \mapsto h_m$, where $h_m : \mathcal{X} \to \mathcal{Y}$ is a decision rule

1. Build M bootstrap samples $\tilde{S}_m^1, \dots, \tilde{S}_m^M$.
2. For each $1 \le j \le M$, call weak classifier on $\tilde{S}_m^j$ so as to obtain rule $\hat{h}_m^j = \Phi_m(\tilde{S}_m^j)$.
3. Aggregate all decision rules into a strong classifier $\hat{h}_m$:
   - for classification: by majority vote
     $$\hat{h}_m(x) = \arg\max_{y \in \mathcal{Y}} \sum_{j=1}^{m} \mathbb{1}\left\{\hat{h}_m^j(x) = y\right\} ;$$
   - for regression: by (uniform) averaging
     $$\hat{h}_m(x) = \frac{1}{m} \sum_{j=1}^{m} \hat{h}_m^j(x) .$$

Out-of-bag error estimate

Random Forest

- Bagging with decision trees
- No need to optimize too much on the tree (for speed, but not only):
  - no pruning
  - simplified splitting rule (see below)
  - limited depth (sometimes to 2)
- extra variance:
  - consider a subset of variables only as candidates for splitting
  - split at average (or median) value

Measure the importance of each variable:

- (rough) number of occurrences of the variable in the forest
- mean decreasse Gini: sum of the heterogeneity measure decrease caused by the variable

Outline

Support Vector Machines

Super-learning: Ensemble Methods

Bagging

Boosting

Boosting

See Rob Schapire's excellent slides:

https://www.csie.ntu.edu.tw/~mhyang/course/u0030/papers/schapire.pdf