Statistical Learning and Applications

10/02/2014 – Supervised Learning Course 3

Contents

1 Support Vector Machines (continued)

1.1 Reminder

SVM are a linear method for binary classification. 2 methods,

- hard margin SVM: for linearly separable problems, hyperplane with largest margin
- soft margin SVM: allow points to be on the wrong side of the margin

Soft margin can be written as a penalised empirical risk minimization problem.

1.2 Algorithms

- QP with box constraints. In practice, faster dedicated algorithm exist.
- SimpleSVM : solve sub-problem with a restricted set of points, then iteratively add the points which most violate the constraints
- Stochastic Gradient Descent

2 l_1 penalties

2.1 l_1 norm

l_1 norm :

$$\Omega(\theta) = \|\theta\|_1 = \sum_{j=1}^{p} |\theta_j|$$

The l_1 norm leads in practice to *sparse* estimators. A problem constrained by the l_1 norm can be thought of as a *convex relaxation* of the problem constrained by the l_0 norm.

2.2 Algorithms for the Lasso

Lasso:

$$\min_{\alpha} \|X\alpha - y\|^2 + \lambda\|\alpha\|_1$$

Can be formulated as a QP problem and solved with generic toolboxes. However, there exists other algorithms, *e.g.* coordinate descent which are often faster.

3 Relationship to maximum likelihood estimation

3.1 Model for the Data

So far, risk minimization without model for the data. Model $P(D \mid \theta)$ of data D, for instance

$$y = \bar{\theta}^T x + \varepsilon$$

where ε has a given distribution. It is common to estimate θ by the value which maximizes the likelihood of the data under the model

$$\hat{\theta} = \arg\max_{\theta} P(D \mid \theta)$$

In practice, it is easier to minimize the negative log likelihood

$$\hat{\theta} = \arg\min_{\theta} -logP(D \mid \theta)$$

, which can be thought of like an empirical risk minimization problem with loss function:

$$L(D, \theta) = -logP(D \mid \theta)$$

.

3.2 Bayesian Statistics

Prior distribution $P(\theta)$ over the parameter θ. *Posterior* distribution $P(\theta \mid D)$:

$$P(\theta \mid D) = \frac{P(D \mid \theta)P(\theta)}{P(D)} \propto P(D \mid \theta)P(\theta)$$

Estimate θ through maximization of its *posterior* likelihood :

$$\hat{\theta}_{MAP} = \arg\max_{\theta} P(\theta \mid D)$$

is a penalized empirical risk minimization problem, with penalty $\Omega(\theta) = -logP(\theta)$.

4 Validation

Need to estimate the population risk :

$$R(f) = \int_{\mathcal{X}\times\mathcal{Y}} L(y, f(x))d\mathbb{P} = \mathbb{E}[L(y, f(x))]$$

4.1 Hold Out Procedure

Split available data between training and test sets :

$$\hat{R}^{HO}(\hat{f}, D_n, I^{(t)}) = \frac{1}{n_v} \sum_{i \in D_n^{(v)}} L(y_i, \hat{f}_{D_n^{(t)}}(x_i))$$

where

- D_n : full set of n available data points
- $I^{(t)}$: subset of indices used for training
- $D_n^{(t)}$: set of data point restricted to training indices
- $D_n^{(v)}$: complement of $D_n^{(t)}$
- $\hat{f}$: learning algorithm whose risk we want to estimate
- $\hat{f}_{D_n^{(t)}}$: function learnt by applying the algorithm to the training data $D_n^{(t)}$

4.2 Cross Validation

Averaging several hold out estimators of the risk corresponding to different data splits :

$$\hat{R}^{CV}(\hat{f}, D_n, (I_j^{(t)})_{1\leq j\leq B}) = \frac{1}{B}\sum_{j=1}^{B} \hat{R}^{HO}(\hat{f}, D_n, I_j^{(t)})$$

where $I_1^{(t)}, \ldots, I_B^{(t)}$ are non empty proper subsets of $\{1, \ldots, n\}$. CV estimators :

- V-fold CV : partition D_n into V sets of approximately equal cardinality $\frac{n}{V}$
- Leave-one-out : V-fold with $V = n$
- Monte-Carlo CV, leave-p-out CV, ...

4.3 Model Selection vs Assessment

CV can be used for *model assessment* (see above), but also for *model selection*. Some care is necessary when doing both, for instance split the data in Train/Validation/Test set or perform double cross validation.