**ISML-II: Machine Learning** **Spring 2014**

# Lecture 20- Variables Selection

*Lecturer: Lorenzo Rosasco*

In many practical situations, beyond predictions it is important to obtain interpretable results. Interpretability is often related to detecting which factors have determined our prediction. We look at this question from the perspective of variable selection.

Consider a linear model

$$f_w(x) = w^T x = \sum_{i=1}^{v} w^j x^j. \tag{20.1}$$

Here we can think of the components $x^j$of an input as of specific measurements: pixel values in the case of images, dictionary word counting in the case of texts, etc. Given a training set, the goal of variable selection is to detect which variables are important for prediction. The key assumption is that the best possible prediction rule is sparse, that is only few of the coefficients in (20.1) are different from zero.

## 20.1 Subset Selection

A brute force approach would be to consider all the training sets obtained considering all the possible subsets of variables. More precisely we could begin by considering only the training set where we retain the first variable of each input points. Then the one where we retain only the second, and so on and so forth. Next, we could pass to consider a training set with pairs of variables, then triplets etc. For each training set one would solve the learning problem and eventually end selecting the variables for which the corresponding training set achieves the best performance.

The described approach has an exponential complexity and becomes unfeasible already for relatively small $D$. If we consider the square loss, it can be shown that the corresponding problem could be written as

$$\min_{w \in \mathbb{R}^D} \frac{1}{n} \sum_{i=1}^{n} (y_i - f_w(x_i))^2 + \lambda \|w\|_0, \tag{20.2}$$

where

$$\|w\|_0 = |\{j \mid w^j \neq 0\}|$$

is called the $\ell_0$ norm and counts the number of non zero components in $w$. In the following we focus on the least squares loss and consider different approaches to find approximate solution to the above problem, namely *greedy methods* and *convex relaxation*.

## 20.2 Greedy Methods: (Orthogonal) Matching Pursuit

Greedy approaches are often considered to find approximate solutions to problem (20.2). This class of approaches to variable selection generally encompasses the following steps:

1. initialize the residual, the coefficient vector, and the index set,
2. find the variable most correlated with the residual,
3. update the index set to include the index of such variable,
4. update/compute coefficient vector,
5. update residual.

The simplest such procedure is called forward stage-wise regression in statistics and matching pursuit (MP) in signal processing. To describe the procedure we need some notation. Let $X_n$ be the $n \times D$ data matrix and $X^j \in \mathbb{R}^n$, $j = 1, \dots, D$ be the columns of $X_n$. Let $Y_n \in \mathbb{R}^n$ be the output vector. Let $r, w, I$ denote the residual, the coefficient vector, an index set, respectively.

The MP algorithm starts by initializing the residual $r \in \mathbb{R}^n$, the coefficient vector $w \in \mathbb{R}^D$, and the index set $I \subseteq \{1, \dots, D\}$,

$$r_0 = Y_n, \quad , w_0 = 0, \quad I_0 = \emptyset.$$

The following procedure is then iterated for $i = 1, \dots, T-1$. The variable most correlated with the residual is given by

$$k = \arg \max_{j=1,\dots,D} a_j, \quad a_j = \frac{(r_{i-1}^T X^j)^2}{\|X^j\|^2},$$

where we note that

$$v^j = \frac{r_{i-1}^T X^j}{\|X^j\|^2} = \arg\min_{v \in \mathbb{R}} \|r_{i-1} - X^j v\|^2, \quad \|r_{i-1} - X^j v^j\|^2 = \|r_{i-1}\|^2 - a_j$$

The selection rule has then two interpretations. We select the variable such that the projection of the output on the corresponding column is larger, or, equivalently, we select the variable such that the corresponding column best explains the the output vector in a least squares sense.

Then, the index set is updated as $I_i = I_{i-1} \cup \{k\}$, and the coefficients vector is given by

$$w_i = w_{i-1} + w_k, \quad w_k k = v_k e_k \tag{20.3}$$

where $e_k$ is the element of the canonical basis in $\mathbb{R}^D$, with $k$-th component different from zero. Finally, the residual is updated

$$r_i = r_{i-1} - X w^k.$$

A variant of the above procedure, called Orthogonal Matching Pursuit, is also often considered. The corresponding iteration is analogous to that of MP, but the coefficient computation (20.3) is replaced by

$$w_i = \arg\min_{w\in\mathbb{R}^D} \|Y_n - X_n M_{I_i} w\|^2,$$

where the $D \times D$ matrix $M_I$ is such that $(M_I w)^j = w^j$ if $j \in I$ and $(M_I w)^j = 0$ otherwise. Moreover, the residual update is replaced by

$$r_i = Y_n - X_n w_i.$$

## 20.3 Convex Relaxation: LASSO & Elastic Net

Another popular approach to find an approximate solution to problem (20.2) is based on a convex relaxation. Namely, the $\ell_0$ norm is replaced by the $\ell_1$ norm,

$$\|w\|_1 = \sum_{j=1}^{D} |w^j|,$$

so that, in the case of least squares, problem (20.2) is replaced by

$$\min_{w\in\mathbb{R}^D} \frac{1}{n}\sum_{i=1}^{n}(y_i - f_w(x_i))^2 + \lambda\|w\|_1. \tag{20.4}$$

The above problem is called *LASSO* in statistics and *Basis Pursuit* in signal processing. The objective function defining the corresponding minimization problem is convex but not differentiable. Tools from non-smooth convex optimization are needed to find a solution. A simple yet powerful procedure to compute a solution is based on the so called Iterative Soft Thresholding Algorithm (ISTA). The latter is an iterative procedure where, at each iteration, a non linear soft thresholding operator is applied to a gradient step. More precisely, ISTA is defined by the following iteration

$$w_0 = 0, \quad w_i = S_{\lambda\gamma}(w_{i-1} - \frac{2\gamma}{n} X_n^T(Y_n - X_n w_{i-1})), \quad i = 1, \ldots, T_{\max}$$

which should be run until a convergence criterion is met, e.g. $\|w_i - w_{i-1}\| \leq \epsilon$, for some precision $\epsilon$, or a prescribed maximum number of iterations $T_{\max}$ is reached. To ensure convergence we should choose the step-size $\gamma = \frac{n}{2\|X_n^T X_n\|}$ .

Note that the argument of the soft thresholding operator corresponds to a step of gradient descent. Indeed,

$$\frac{2}{n} X_n^T(Y_n - X_n w_{i-1})$$

The soft thresholding operator acts component-wise on a vector $w$, so that

$$S_\alpha(u) = ||u| - \alpha|_+ \frac{u}{|u|}.$$

The above expression shows that the coefficients of the solution of problem (20.2) as computed by ISTA can be exactly zero: this can be contrasted with Tikhonov regularization where this is hardly the case.

Indeed, it is possible to see that, on the one hand, while Tikhonov allows to compute a stable solution, in general its solution is not sparse. On the other hand, the solution of LASSO might not be stable. The elastic net algorithm, defined as

$$\min_{w\in\mathbb{R}^D} \frac{1}{n}\sum_{i=1}^{n}(y_i - f_w(x_i))^2 + \lambda(\alpha\|w\|_1 + (1-\alpha)\|w\|_2^2), \quad \alpha\in[0,1], \tag{20.5}$$

can be seen as hybrid algorithm which is interpolated between Tikhonov and LASSO. The ISTA procedure can be adapted to solve the elastic net problem, where the gradient descent step incorporates also the derivative of the $\ell^2$ penalty term. The resulting algorithm is

$$w_0 = 0, \quad w_i = S_{\lambda\alpha\gamma}((1-\lambda\gamma(1-\alpha))w_{i-1} - \frac{2\gamma}{n}X_n^T(Y_n - X_n w_{i-1})), \quad i = 1,\dots,T_{\max}$$

To ensure convergence, we should choose the step-size $\gamma = \frac{n}{2(\|X_n^T X_n\| + \lambda(1-\alpha))}$ .