Labeling Image Patches by Boosting based Median Classifier

Songfeng Zheng
SongfengZheng@MissouriState.edu

Department of Mathematics, Missouri State University
901 S. National Ave., Springfield, MO 65897, USA

Since the data in computer vision problems usually lie in very high dimensional spaces, it is possible that the data are from long-tail distribution or have heavy outliers. In such situations, it is well-known that the median is usually more stable than other statistics. As such, it is worthwhile trying to design classifiers based on the median information.

Consider the following model:

$$Y^* = h(\mathbf{x}) + \varepsilon \quad \text{and} \quad Y = I(Y^* \geq 0),$$

where Y^* is a continuous latent variable, $h(\cdot)$ is the true model for Y^*, ε is a disturb, and $Y \in \{0,1\}$ is the observed label given the feature vector $\mathbf{x} \in \mathbf{R}^p$, $I(\cdot)$ is the indicator function. Let the median of the latent variable Y^* be $f(\mathbf{x},\beta)$ with β as the parameter vector, i.e., $M(Y^*|\mathbf{x}) = f(\mathbf{x},\beta)$, we can prove that the conditional median of the binary variable Y can be modeled as

$$M(Y|\mathbf{x}) = I\left(f(\mathbf{x},\beta) \geq 0\right).$$

Once the model is fitted, i.e., the parameter vector β is estimated as $\mathbf{b}$, we can make prediction by

$$\hat{Y} = I(f(\mathbf{x},\mathbf{b}) \geq 0), \tag{1}$$

where $\hat{Y}$ is the predicted label for the input feature vector $\mathbf{x}$. We call Eqn. (1) as a median classifier.

Given training dataset $\{(\mathbf{x}_i, Y_i), i = 1, \cdots, n\}$, with $\mathbf{x}_i \in \mathbf{R}^p$ and $Y_i \in \{0,1\}$, the median classifier is approximately learned by solving

$$\mathbf{b} = \arg\max_{\beta} \left\{ S(\beta,h) = \sum_{i=1}^{n} [Y_i - 0.5] K\left(\frac{f(\mathbf{x}_i,\beta)}{h}\right) \right\}, \tag{2}$$

where h is a small positive number, and $K(t)$ is smoothed version of the indicator function.

We propose to maximize the objective function in Eqn. (2) by gradient ascent in the framework of functional gradient method [2]. Let $f^{[m]}(\cdot)$ be the fitted function at the m-th iteration, we obtain the Median Boost algorithm, which is shown as Algorithm 1.

Algorithm 1 Median Boost Algorithm

0: Initialize $f^{[0]}(\mathbf{x}) = 0$.
1: **for** $m = 1$ to M **do**
2: Compute the gradient $\frac{\partial}{\partial f} l(Y_i, f)$ at $f^{[m-1]}(\mathbf{x}_i)$, for $i = 1, \cdots, n$:

$$U_i = \left.\frac{\partial l(Y_i,f)}{\partial f}\right|_{f=f^{[m-1]}(\mathbf{x}_i)} = \frac{Y_i - 0.5}{h} K'\left(\frac{f^{[m-1]}(\mathbf{x}_i)}{h}\right).$$

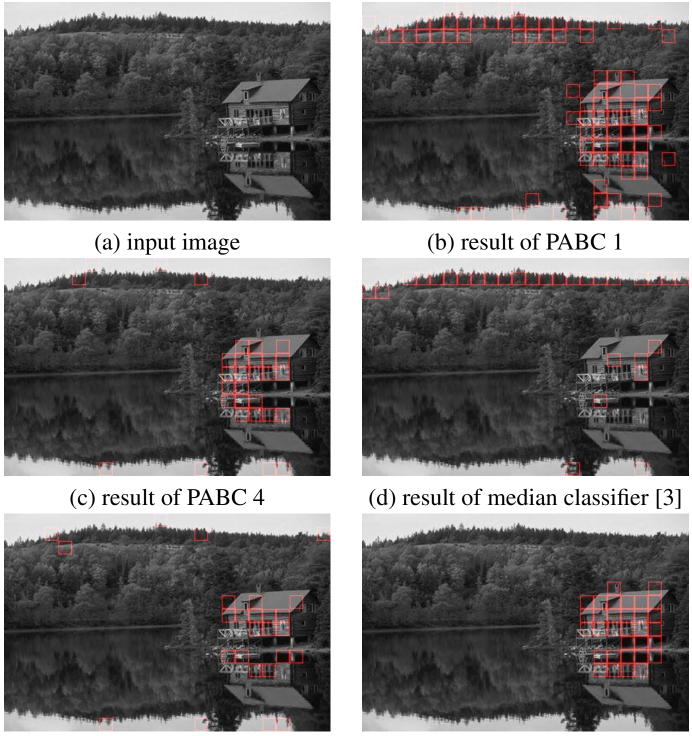
3: Fit the gradients $U_1, \cdots, U_n$ to $\mathbf{x}_1, \cdots, \mathbf{x}_n$ by the base procedure:

$$\{(\mathbf{x}_i, U_i),\ i = 1, \cdots, n\} \longrightarrow g^{[m]}(\cdot).$$

4: Update the estimation by $f^{[m]}(\cdot) = f^{[m-1]}(\cdot) + \nu g^{[m]}(\cdot)$, where ν is a step-length factor.
5: **end for**
6: Output the classifier $I(f^{[M]}(\mathbf{x}) \geq 0)$.

We test the proposed Median Boost on the task of labeling building blocks in natural images [4, 5]. Each image is divided into non-overlapping 16×16 patches. The ground truth was generated by manually labeling every patch as *building* or *non-building*. For each image patch, we extract about 10,000 features, including features described in [4, 5], mean and variance values of different filter responses inside sub-windows, histograms of different filter responses, etc.

The Median Boost algorithm used the standard normal cumulative distribution function with $h = 0.1$. We fix the step size parameter at $\nu = 0.1$, as suggested by [1, 2]. In the third step of the Median Boost algorithm, the simple linear regression model with only one predictor was used as weak learner for its simplicity. The Median Boost classifier was ran for 120 iterations. The Probabilistic AdaBoost Cascade [6] was tested with the same set of features. The cascade structure contains four AdaBoost nodes, and each AdaBoost node runs 120 iterations, with decision stump as weak learner. As a comparison, we also tested the median classifier defined in [3] on this problem.



(a) input image (b) result of PABC 1
(c) result of PABC 4 (d) result of median classifier [3]
(e) result of Median Boost (f) Manually labeled result

Figure 1: The experimental result on image patch labeling.

Fig. 1 shows the detection results on a testing image. As seen from (b) and (c), with more AdaBoost nodes, cascade can remove some false positives. (e) presents the results obtained by Median Boost, and we see the results are comparable to the results in (c), but visually better than the result in (b). (d) is the result obtained by the median classifier in [3], which shows significantly more false positives than both (c) and (e). The results from Median Boost and AdaBoost cascade are visually quite close to the manually labeled result which is shown in (f). However, AdaBoost cascade uses 4 AdaBoost nodes, totally 480 features, while Median Boost only selects 120 features.

[1] P. Bühlmann and T. Hothorn. Boosting Algorithms: Regularization, Prediction and Model Fitting. *Statistical Science*, 22, 516-522, 2007.

[2] J. H. Friedman. Greedy Function Approximation: A Gradient Boosting Machine. *Annals of Stat.*, 29, pp. 1189-1232, 2001.

[3] P. Hall, D. M. Titterington, and J. H. Xue. Median-based Classifiers for High-Dimensional Data. *JASA*, vol. 104, pp. 1597-1608, 2009.

[4] S. Kumar and M. Hebert. Man-Made Structure Detection in Natural Images using a Causal Multiscale Random Field. In *CVPR*, 2003.

[5] S. Kumar and M. Hebert. Discriminative Random Fields: A Discriminative Framework for Contextual Interaction in Classification. In *ICCV*, 2003.

[6] S. Zheng. Probabilistic Cascade Random Fields for Man-made Structure Detection. In *ACCV*, 2009.