Moderating the Outputs of Support Vector Machine Classifiers

James Tin-Yau Kwok
Department of Computer Science
Hong Kong Baptist University
Hong Kong
jamesk@comp.hkbu.edu.hk

Abstract

In this paper, we extend the use of moderated outputs to the support vector machine (SVM) by making use of a relationship between SVM and the evidence framework. The moderated output is more in line with the Bayesian idea that the posterior weight distribution should be taken into account upon prediction, and it also alleviates the usual tendency of assigning overly high confidence to the estimated class memberships of the test patterns. Moreover, the moderated output derived here can be taken as an approximation to the posterior class probability. Hence, meaningful rejection thresholds can be assigned and outputs from several networks can be directly compared. Experimental results on both artificial and real-world data are also discussed.

1 Introduction

In recent years, there has been a lot of interest in studying the support vector machine (SVM) [8, 9]. To date, SVM has been applied successfully to a wide range of problems, such as classification, regression, time series prediction and density estimation. In this paper, we will focus on classification problems.

Despite its many successes, SVM relies on only one weight solution (indirectly represented by the set of Lagrangian multipliers) in making predictions. However, from a Bayesian perspective, the weights of any machine, even after learning, still take a certain posterior distribution. Using just one weight solution as the sole representative thus neglects posterior uncertainty in the weights. This often leads to more extreme predicted outputs during testing, and in turn indicates an overly high confidence that the pattern belongs to a particular class. Under the Bayesian framework, the proper way to handle these weight parameters is by marginalization, which involves integrating them out from the conditional distribution. MacKay called the resultant marginalized output the *moderated output*, and this has been shown to be better in the context of neural networks [6].

Another concern is that the output from a SVM classifier has no clear relationship to posterior class probabilities. Such a connection, if exists, would offer a number of advantages [7], such as the meaningful assignment of rejection thresholds and a direct comparison or combination of outputs from several networks.

In this paper, we extend the use of moderated outputs to SVM by making use of a relationship [4] between the evidence framework [5] and the SVM. The evidence framework is a Bayesian framework proposed by MacKay and has been applied successfully to the learning of feedforward neural networks in both classification and regression problems. The rest of this paper is organized as follows. A brief summary of the SVM and its relationship to the evidence framework will be described in Sections 2 and 3 respectively. The moderated output for SVM will be derived in Section 4. Simulation results are then presented in Section 5, and the last section gives some concluding remarks.

2 Support Vector Machine for Classification

Let the training set D be $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$, with each input $\mathbf{x}_i \in \Re^m$ and the output label $y_i \in \{\pm 1\}$. The SVM first maps $\mathbf{x}$ from the input space $\Re^m$ to $\mathbf{z} = \phi(\mathbf{x})$ in a feature space $\mathcal{F}$. Here, we only consider the case when the data is not linearly separable in $\mathcal{F}$. The SVM algorithm introduces non-negative slack variables $\xi_i \geq 0,\ \ i = 1, \ldots, N$, and

$$\text{minimize } \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^N \xi_i, \tag{1}$$

subject to $y_i a(\mathbf{x}_i, \mathbf{w}) \geq 1 - \xi_i, i = 1, \ldots, N$. Here, C is a user-defined regularization parameter controlling the tradeoff between model complexity and training error, and ξ_i measures the (absolute) difference between

$a(\mathbf{x}_i, \mathbf{w})$ and y_i. This minimization problem can be transformed to a quadratic programming (QP) problem:

$$\text{maximize } W(\boldsymbol{\alpha}) = \boldsymbol{\alpha}^T \mathbf{1} - \frac{1}{2} \boldsymbol{\alpha}^T \mathbf{Q} \boldsymbol{\alpha},$$

subject to the constraints $\mathbf{0} \le \boldsymbol{\alpha} \le C\mathbf{1}$ and $\boldsymbol{\alpha}^T \mathbf{Y} = 0$. Here $\mathbf{Y}^T = (y_1, \ldots, y_N)$ and $\mathbf{Q}$ is a symmetric $N \times N$ matrix with elements $Q_{ij} = y_i y_j \mathbf{z}_i^T \mathbf{z}_j = y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$, where K is the *kernel*.

During testing, for a test vector $\mathbf{x} \in \Re^m$, we first compute

$$a(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T \mathbf{z} + b = \sum_i \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) + b, \qquad (2)$$

and then its class label $o(\mathbf{x}, \mathbf{w})$ is given by:

$$o(\mathbf{x}, \mathbf{w}) = \begin{cases} 1 & a(\mathbf{x}, \mathbf{w}) > 0, \\ -1 & \text{otherwise.} \end{cases} \qquad (3)$$

3 Relationship between The Evidence Framework and the SVM

The evidence framework is divided into three levels of inference. In this section, we recapitulate the relationship between the first level of inference and the SVM as described in [4].

A model $\mathcal{H}$, with a k-dimensional parameter vector $\mathbf{w}$, consists of its functional form f, the distribution $p(D|\mathbf{w}, \mathcal{H})$ that the model makes about the data D, and a prior parameter distribution $p(\mathbf{w}|\lambda, \mathcal{H})$ with a regularization parameter λ. The first level of inference infers the posterior distribution of $\mathbf{w}$ for a given value of λ by using the Baye's rule:

$$p(\mathbf{w}|D, \lambda, \mathcal{H}) \propto p(\mathbf{w}|\lambda, \mathcal{H}) p(D|\mathbf{w}, \mathcal{H}). \qquad (4)$$

Now, consider the following probability model:

- The prior over $\mathbf{w}$ is the Gaussian prior

$$p(\mathbf{w}|\lambda, \mathcal{H}) \propto \exp(-\frac{\lambda}{2} \|\mathbf{w}\|^2).$$

- The probability distribution $p(y_i|\mathbf{x}_i, \mathbf{w}, \mathcal{H})$ for $y_i = \pm 1$ is given by

$$\begin{aligned} p(y_i|\mathbf{x}_i, \mathbf{w}, \mathcal{H}) & \\ = & \frac{\exp(-[1 - y_i a_i]_+)}{\exp(-[1 - a_i]_+) + \exp(-[1 + a_i]_+)} \qquad (5) \\ \simeq & \exp(-[1 - y_i a_i]_+) = \exp(-\xi_i), \qquad (6) \end{aligned}$$

where $a_i = a(\mathbf{x}_i, \mathbf{w})$ in (2) and $[u]_+ = u I_{\{u>0\}}$.

Substituting these probabilities into (4) and assuming that the patterns are i.i.d., we obtain

$$-\log p(\mathbf{w}|D, \lambda, \mathcal{H}) = \frac{\lambda}{2} \|\mathbf{w}\|^2 + \sum_i \xi_i - \sum_i \log p(\mathbf{x}_i) + \text{constant}.$$

The last two terms on the right do not depend on $\mathbf{w}$. Hence, by setting $C = 1/\lambda$, optimizing (1) can be regarded as finding the *maximum a posteriori* (MAP) estimate $\mathbf{w}_{MP}$ of $\mathbf{w}$. In other words, by using the approximated probabilities in (6) (instead of the true ones in (5)), training of the SVM can be regarded as performing the first level of inference in the evidence framework. Moreover, in the light of this correspondence, traditional SVM can be considered as using $\mathbf{w}_{MP}$ as the sole representative of the whole posterior distribution upon prediction. A plot of the true and approximated probability distributions is shown in Figure 1.

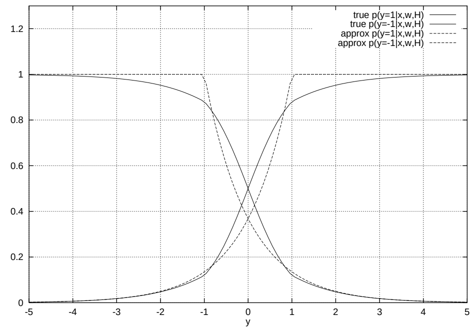


Figure 1: True and approximated probabilities.

4 Moderating the SVM Output

As in the evidence framework, we assume that the posterior distribution of $\mathbf{w}$ can be approximated by a single Gaussian at $\mathbf{w}_{MP}$. Since $a(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T \mathbf{z} + b$, therefore the posterior distribution of $a(\mathbf{x}, \mathbf{w})$ will also be Gaussian $\mathrm{N}(a_{MP}, s^2)$, with mean $a_{MP}(\mathbf{x}) = a(\mathbf{x}, \mathbf{w}_{MP})$ and variance

$$s^2(\mathbf{x}) = \mathbf{z}^T \mathbf{A}^{-1} \mathbf{z}, \qquad (7)$$

where $\mathbf{A} = \nabla^2 M = \nabla^2 (\lambda E_W + \sum_{i=1}^N \xi_i)$ is the hessian. As mentioned in Section 1, the Bayesian approach of handling the unwanted parameter $\mathbf{w}$ is to integrate it out from the conditional distribution. Thus,

$$\begin{aligned} p(y = 1|D, \mathbf{x}) & \\ = & \int_{-\infty}^{1} e^{-1+a} \mathrm{N}(a_{MP}, s^2) da + \int_1^\infty \mathrm{N}(a_{MP}, s^2) da \\ = & \frac{1}{2} \exp\left(\frac{s^2}{2} + a_{MP} - 1 \right) \operatorname{erfc}\left(\frac{s^2 + a_{MP} - 1}{\sqrt{2} s} \right) \end{aligned}$$

$$+\frac{1}{2}\text{erfc}\left(\frac{1-a_{MP}}{\sqrt{2}s}\right), \quad (8)$$

and

$$\begin{aligned} p(y=-1|D,\mathbf{x}) & \\ = & \int_{-\infty}^{-1} \mathrm{N}(a_{MP}, s^2)da + \int_{-1}^{\infty} e^{-1-a}\mathrm{N}(a_{MP}, s^2)da \\ = & \frac{1}{2}\exp\left(\frac{s^2}{2} - a_{MP} - 1\right)\text{erfc}\left(\frac{s^2 - a_{MP} - 1}{\sqrt{2}s}\right) \\ & +\frac{1}{2}\text{erfc}\left(\frac{1+a_{MP}}{\sqrt{2}s}\right). \quad (9) \end{aligned}$$

Here, $\text{erfc}(x) = \frac{2}{\sqrt{\pi}}\int_x^{\infty} e^{-t^2}dt$ is the complementary error function and can be readily computed. Recall that $p(y_i|\mathbf{x}_i, \mathbf{w}, \mathcal{H})$ is only an approximation, consequently (8) and (9) have to be normalized as

$$\tilde{p}(y=1|D,\mathbf{x}) = \frac{p(y=1|D,\mathbf{x})}{p(y=1|D,\mathbf{x}) + p(y=-1|D,\mathbf{x})},$$
$$\tilde{p}(y=-1|D,\mathbf{x}) = 1 - \tilde{p}(y=1|D,\mathbf{x}).$$

We call $\tilde{p}(y=1|D,\mathbf{x})$ the *moderated output* of the SVM for pattern $\mathbf{x}$. Figure 2 shows the variation of the moderated output as a function of a_{MP} and s. As can be seen, when s increases, changes in the moderated output becomes more gradual to changes in a_{MP}. And when s becomes very large, the moderated output stays near 0.5. As the moderated output is an approximation of the posterior class probability, this indicates maximum uncertainty in predicting the class membership of $\mathbf{x}$.

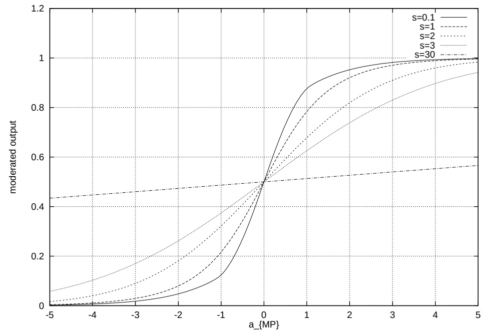


Figure 2: Moderated output as a function of a_{MP} and s.

To compute $s^2(\mathbf{x})$ in (7), we have to determine the hessian $\mathbf{A}$. First, define the sigmoid function $\sigma(u) = 1/(1+e^{-\eta u})$ for some $\eta > 0$, and $r(u) = u\sigma''(u) + 2\sigma'(u)$. Using results in [4], $\mathbf{A}$ can be approximated as

$$\mathbf{A} \simeq \mathbf{P}\mathbf{\Lambda}\mathbf{P}^T, \quad (10)$$

where

$$\mathbf{P} = (\mathbf{v}_1, \ldots, \mathbf{v}_n) = (\sum_{i=1}^{N} \mu_{1i}\mathbf{z}_i, \ldots, \sum_{i=1}^{N} \mu_{ni}\mathbf{z}_i), \quad (11)$$

and

$$\mathbf{\Lambda} = diag(\lambda + \rho_1, \ldots, \lambda + \rho_n). \quad (12)$$

Here, the $\mathbf{v}_l$'s are orthogonal, n is the number of significant eigenvalues in the $N \times N$ matrix $\tilde{\mathbf{K}}$ with entries $\tilde{K}_{ij} = r(|y_i - a(\mathbf{x}_i, \mathbf{w})|)K(\mathbf{x}_i, \mathbf{x}_j) \equiv r_i K(\mathbf{x}_i, \mathbf{x}_j)$, while ρ_l and $\boldsymbol{\mu}_l = (\mu_{l1}, \ldots, \mu_{lN})^T$ are obtained from the eigen system $\rho_l \boldsymbol{\mu}_l = \tilde{\mathbf{K}}\boldsymbol{\mu}_l$.

Later on, it will be more convenient to have $\mathbf{P}^{-1} = \mathbf{P}^T$. This is satisfied by ensuring that the set $\{\mathbf{v}_l\}_l$ is orthonormal. Let the required normalizing factor for $\mathbf{v}_l$ be c_l and define $\tilde{\boldsymbol{\mu}}_l \stackrel{\text{def}}{=} (\mu_{l1}/r_1, \ldots, \mu_{lN}/r_N)^T$. Then,

$$\begin{aligned} 1 &= \left\| c_l \sum_{i=1}^{N} \mu_{li}\mathbf{z}_i \right\|^2 = c_l^2 \sum_{i,j=1}^{N} \mu_{li}\mu_{lj}K(\mathbf{x}_i, \mathbf{x}_j) \\ &= c_l^2 \sum_{i,j=1}^{N} \frac{\mu_{li}}{r_i}\mu_{lj}\tilde{K}_{ij} = c_l^2 \tilde{\boldsymbol{\mu}}_l^T \tilde{\mathbf{K}}\boldsymbol{\mu}_l \\ &= c_l^2 \rho_l \tilde{\boldsymbol{\mu}}_l^T \boldsymbol{\mu}_l = c_l^2 \rho_l \sum_{i=1}^{N} \frac{\mu_{li}^2}{r_i}. \end{aligned}$$

Hence, the required normalizing factor for $\mathbf{v}_l$ is $c_l = 1/\sqrt{\rho_l \sum_{i=1}^{N} \mu_{li}^2/r_i}$.

Utilizing (10), (11), (12) and the fact that $\mathbf{P}^{-1} = \mathbf{P}^T$, $s^2(\mathbf{x})$ in (7) can then be computed as

$$\begin{aligned} s^2 &= \mathbf{z}^T\mathbf{A}^{-1}\mathbf{z} = \mathbf{z}^T\mathbf{P}\mathbf{\Lambda}^{-1}\mathbf{P}^T\mathbf{z} = (\mathbf{P}^T\mathbf{z})^T\mathbf{\Lambda}^{-1}(\mathbf{P}^T\mathbf{z}) \\ &= \sum_{i=1}^{n} \frac{1}{\lambda + \rho_i}\left(\sum_{j=1}^{N} \mu_{ij}K(\mathbf{x}_j, \mathbf{x})\right)^2. \quad (13) \end{aligned}$$

5 Simulation

In this section, we report simulation results on three data sets. The polynomial kernel $K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T\mathbf{x}_j + 1)^d$ is used throughout.

5.1 Toy Problem

This is an artificial two-class classification problem, with data generated from 5 Gaussians (Figure 3). The training set has 500 patterns and the test set has 10,201 patterns. Polynomial kernel with $d = 3$ is used.

Figure 4 compares the various outputs obtained from the SVM: The MAP output does not show any meaningful pattern in general. Both the clipped MAP output and the unmoderated probability output show high confidence (with output values near ± 1 and 0/1 respectively) in most parts of the input space, even far beyond the decision boundary and in regions of low data density.

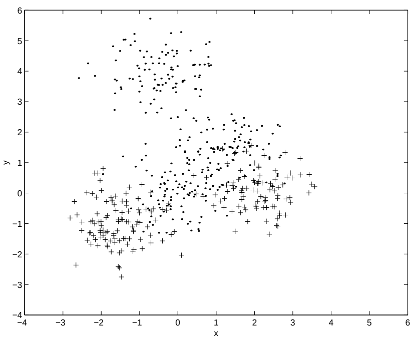


Figure 3: Training data for the toy problem.

In comparison, the moderated output becomes less certain as the pattern moves away from the training data.

Notice that the moderated output passes through 0.5 at $a_{MP} = 0$. Hence, if the class label of $\mathbf{x}$ is obtained by thresholding the moderated output at 0.5, then it will make no difference as the MAP-based decision rule in (3). In the following, as in [6], we use instead

$$G = -\sum_{i=1}^{N_t} \frac{1+y_i}{2} \log o_i + \frac{1-y_i}{2} \log(1-o_i). \qquad (14)$$

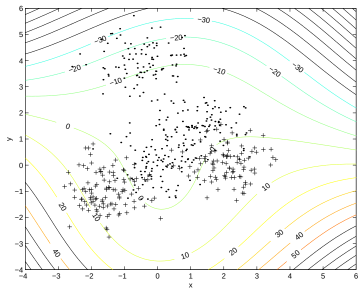
to compare the performance of the SVM outputs. Here, $o_i \in [0,1]$ is the output for pattern i, and N_t is the size of the independent test set (equals 10201 in this experiment). As mentioned in [6], $\exp(-G)$ can be interpreted as the data likelihood. The smaller the value of G, the more likely is the data generated by the model. Table 1 shows the results for the various outputs. For the clipped MAP solution[1], there are some mis-classified test patterns (i.e., $y_i = -1$ but $a_{MP} > 1$, or vice versa) and G goes infinite. For the estimate based on the unmoderated output, the value of G obtained is inferior to that based on the moderated output.

5.2 Image Segmentation Problem

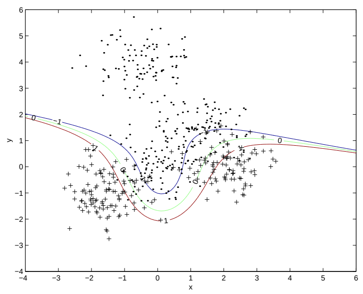
The second data set is the image segmentation data from the UCI machine learning repository[1]. Each pattern has 19 continuous attributes and corresponds to a 3×3 region of an outdoor image. The problem is to classify the pattern into one of the seven classes: brickface, cement, foliage, grass, path, sky and window. There are 210 patterns in the training set and 2,100 patterns in the test set (each class has 300 test patterns). Polynomial kernel with $d = 1$ is used.

We take the usual approach of formulating this multi-class classification problem as a series of binary classi-

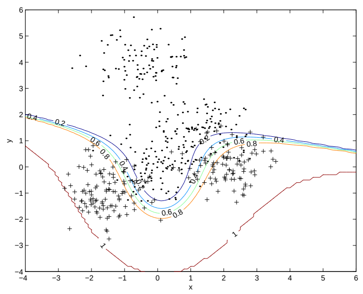
[1] Notice that the clipped MAP output ranges from $[-1, 1]$, and so has to be translated back to $[0, 1]$ before using (14).



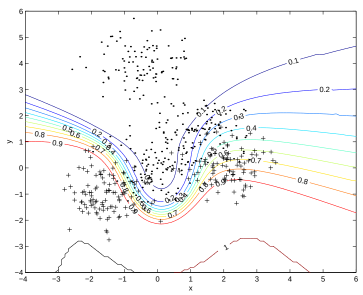
(a) MAP output a_{MP}



(b) a_{MP} clipped at ± 1



(c) Unmoderated output



(d) Moderated output

Figure 4: Various outputs from the SVM.

Table 1: Values of G for various outputs from the SVM.

clipped MAP	unmoderated	moderated
∞	3801	1417

fication problems [2]. Seven classifiers are constructed, with one for each class (e.g. the brickface classifier separates patterns belonging to the brickface class from those that do not). For a particular test pattern, the classifier with the highest output value is selected as winner and the corresponding class label assigned. Table 2 shows the misclassification errors on incorporating the classes one by one (in alphabetical order). The performance of the moderated output is better than that of the MAP output[2]. One can further improve the performance by rejecting the uncertain patterns. As the moderated output approximates the posterior class probability, meaningful rejection thresholds can be set on the moderated output of the winning classifier. Table 3 shows the resultant improvement. In contrast, as the output from traditional SVMs has no clear relationship to the posterior class probabilities, selection of a meaningful rejection threshold is not easy.

6 Conclusion

In this paper, we extend the use of moderated output to SVM based on the evidence framework. Because of the Bayesian nature of this approach, the moderated output derived serves naturally as an approximation to the posterior class probability. Consequently, meaningful rejection thresholds can be assigned and outputs from several networks (as in using dichotomous networks for multi-class problems) can be directly compared. Moreover, unlike the use of unmoderated outputs in approximating the posterior class probabilities (as in [3]), the moderated output is more in line with the Bayesian idea that the posterior weight distribution should be taken into account on prediction, and this alleviates the usual tendency of assigning overly high confidence to its predictions in regions of low training data density. Experimentally, performance improvements on both artificial and real-world data sets are observed.

In this paper, we have focused on classification problems. Extension to regression problems for the calculation of error bars is straight-forward and will be reported elsewhere. Finally, as this work is based on the evidence framework, variability in the hyperparameter (i.e. the regularization parameter λ in (4)) is ignored. This issue will be studied in the future.

[2]For comparison, the percentage error of the one nearest-neighbor classifier (for the discrimination of all seven classes) is 12.3%.

Acknowledgments

This research has been partially supported by the Research Grants Council of the Hong Kong Special Administrative Region under HKBU2063/98E. The author would like to thank Robert Vanderbei for the LOQO quadratic programming solver used in the experiments.

References

[1] C. Blake, E. Keogh, and C.J. Merz. UCI repository of machine learning databases, 1998. http://www.ics.uci.edu/~mlearn/MLRepository.html, University of California, Irvine, Department of Information and Computer Sciences.

[2] V. Blanz, B. Schölkopf, H. Bulthoff, C. Burges, V. Vapnik, and T. Vetter. Comparison of view-based object recognition algorithms using realistic 3D models. In *Proceedings of the International Conference on Artificial Neural Networks*, 1996.

[3] S. Dumais, J. Platt, D. Heckerman, and M. Sahami. Inductive learning algorithms and representations for text categorization. In *7th International Conference on Information and Knowledge Management*, 1998.

[4] J.T. Kwok. Integrating the evidence framework and the support vector machine. In *Proceedings of the European Symposium on Artificial Neural Networks*, pages 177–182, Bruges, Belgium, April 1999.

[5] D.J.C. MacKay. Bayesian interpolation. *Neural Computation*, 4(3):415–447, May 1992.

[6] D.J.C. MacKay. The evidence framework applied to classification networks. *Neural Computation*, 4(5):720–736, September 1992.

[7] M.D. Richard and R.P. Lippmann. Neural network classifiers estimate Bayesian *a posteriori* probabilities. *Neural Computation*, 3:461–483, 1991.

[8] V. Vapnik. *The Nature of Statistical Learning Theory.* Springer Verlag, 1995.

[9] V. Vapnik. *Statistical Learning Theory.* Wiley, 1998.

Table 2: Results on the image segmentation problem.

classes combined	# test patterns in the classes	# error (moderated)	% error (moderated)	# error (MAP)	% error (MAP)
brickface + cement	600	18	3.0	25	4.2
brickface + cement + foliage	900	40	4.4	48	5.3
brickface + cement + foliage + grass	1200	43	3.6	59	4.9
brickface + cement + foliage + grass + path	1500	60	4.0	71	4.7
brickface + cement + foliage + grass + path + sky	1800	62	3.4	72	4.0
brickface + cement + foliage + grass + path + sky + window	2100	180	8.6	205	9.8

Table 3: Performance after setting a rejection threshold on the moderated output of the winning classifier.

rejection threshold	% test patterns rejected	%error
0.1	0.0	8.6
0.2	0.0	8.5
0.3	0.0	7.8
0.4	0.24	6.4
0.5	12.7	4.9
0.6	24.9	3.9
0.7	39.3	3.4
0.8	47.9	2.6
0.9	57.4	3.3