Active Learning for Regression Based on Query by Committee

Robert Burbidge*, Jem J. Rowland, and Ross D. King

Department of Computer Science, University of Wales, Aberystwyth, Penglais, Aberystwyth, SY23 3DB, UK

Abstract. We investigate a committee-based approach for active learning of real-valued functions. This is a variance-only strategy for selection of informative training data. As such it is shown to suffer when the model class is misspecified since the learner's bias is high. Conversely, the strategy outperforms passive selection when the model class is very expressive since active minimization of the variance avoids overfitting.

1 Introduction

In process control we might wish to identify the effect of factors such as temperature, pH, etc. on output but obtaining such information, for example by running the system at various temperatures, pHs, etc., may be costly. In query learning, our goal is to provide criteria that a learning algorithm can employ to improve its performance by actively selecting data that are most informative. Given a small initial sample such a criterion might indicate that the system be run at particular temperatures, pHs, etc. in order for the relationship between these controls and the output to be better characterized.

We focus on supervised learning. Many machine learning algorithms are passive in that they receive a set of labelled data and then estimate the relationship from these data. We investigate a committee-based approach for *actively* selecting instantiations of the input variables x that should be labelled and incorporated into the training set. We restrict ourselves to the case where the training set is augmented one data point at a time, and assume that an experiment to gain the label y for an instance x is costly but computation is cheap. We investigate under what circumstances committee-based active learning requires fewer queries than passive learning.

Query by committee (QBC) was proposed by Seung, Opper and Sompolinksy [1] for active learning of classification problems. A committee of learners is trained on the available labelled data by the Gibbs algorithm. This selects a hypothesis at random from those consistent with the currently labelled data. The next query is chosen as that on which the committee members have maximal disagreement. They considered two toy models with perfectly realizable

* Thanks to Hugh Mallinson for initial inspiration. This work is supported by EPSRC grant reference S47649.

H. Yin et al. (Eds.): IDEAL 2007, LNCS 4881, pp. 209–218, 2007.

targets. The algorithm was implemented in the *query filtering* paradigm; the learner is given access to a stream of inputs drawn at random from the input distribution. With a two-member committee, any input on which the committee members make opposite predictions causes maximal disagreement and its label is queried. It was shown under these conditions that generalization error decreases exponentially with the number of labelled examples, but for random queries (i.e. passive learning), generalization error only decreased with an inverse power law.

Freund et al. [2] showed that QBC is an efficient query algorithm for the perceptron concept class with distributions close to uniform. This provided a rigorous proof of the earlier results, along with some relaxations in the requirements. They suggested a reasonable heuristic for filtering of queries would be to select and label those inputs expected to reduce prediction error. They note that this could be applied when the labels were not binary or even discrete. This is related to the variance-based regression methods described below. For the QBC approach to work, there must be some disagreement over the committee. In the original work, this was achieved by means of a randomized algorithm. An alternative approach is to use different subsets of the data, as in query by bagging and query by boosting [3]. This is also the approach taken in the regression framework investigated below.

2 Active Learning of Real-Valued Functions

The aim in active learning of a real-valued function is to query the labels of inputs such that the generalization error is minimized. The expected error is:

$$\int_x E_T\left[(\hat{y}(x;\mathcal{D}) - y(x))^2 | x\right] q(x) dx \ ,$$

where $E_T[\cdot]$ denotes expectation over $P(y|x)$ and over training sets $\mathcal{D}$, $q(x)$ is the input distribution and $\hat{y}(x;\mathcal{D})$ is the learner's output on input x, given the training set $\mathcal{D}$. The expectation in the integrand can be decomposed as [4]:

$$\begin{aligned} E_T\left[(\hat{y}(x;\mathcal{D}) - y(x))^2 | x\right] &= E\left[(y(x) - E[y|x])^2\right] + (E_{\mathcal{D}}[\hat{y}(x;\mathcal{D})] - E[y|x])^2 \\ &\quad + E_{\mathcal{D}}\left[(\hat{y}(x;\mathcal{D}) - E_{\mathcal{D}}[\hat{y}(x;\mathcal{D})])^2\right] \ , \end{aligned}$$

where $E_{\mathcal{D}}$ denotes expectation over training sets and the other expectations are with respect to the conditional density $P(y|x)$. The first term on the right hand side is the variance of y given x, i.e. the *noise*. The second term is the squared *bias*. The third term is the learner's *variance*. The bias and variance comprise the learner's mean squared error (MSE).

Assuming the learner is approximately unbiased, Cohn, Ghahramani and Jordan [5] propose to select inputs for labelling that minimize the variance of the learner. It is assumed that an estimate of $\sigma^2_{\tilde{y}}$, the variance of the learner at x is available. For a new input $\tilde{x}$, an estimate of the learner's new variance at x given $\tilde{x}$ is computed. This requires an estimate of the distribution $P(\tilde{y}|\tilde{x})$, or, at least, estimates of its mean and variance. This gives an estimate of $\left\langle \tilde{\sigma}^2_{\tilde{y}} \right\rangle$, the expected

variance of the learner at x after querying at $\tilde{x}$. This is integrated over the input space to give an estimate of the integrated average variance of the learner. In practice, a Monte Carlo approximation was used, $\left\langle \tilde{\sigma}^2_{\hat{y}} \right\rangle$ was evaluated at 64 *reference points* and $\tilde{x}$ chosen to minimize the average expected variance over the reference points. In [6], this approach was used to select data to train a neural network. In [5], it was extended to mixtures of Gaussians and to locally weighted regression. Active data selection based on minimizing variance was superior to random data selection for a toy 2-degree-of-freedom robot arm problem. This is a noisy problem where the target function is not perfectly realizable. Note, however, that this technique of selecting statistically 'optimal' training data cannot be applied to all machine learning algorithms.

Krogh and Vedelsby [7] considered committees of neural networks for learning real-valued functions. The committee consists of k networks and the output of network α on input x is $\hat{y}^\alpha(x)$. They defined the *ambiguity* at an input point $\tilde{x}$ as the variance in the predictions of the committee members:

$$\overline{a}(\tilde{x}) = \sum_\alpha \left(\hat{y}^\alpha(x) - \overline{y}(x)\right)^2 \quad .$$

This provided a reliable estimate of the generalization error of the committee, and to determine the optimal voting weights of the committee members in determining the committee's predictions. These two contributions of that work have subsequently been cited a number of times in the literature. However, a third aspect of the work that is rarely cited relates to active learning. They propose to query at each step the label of the input for which the ambiguity is maximal, i.e. where the committee's variance is highest. This can be seen to be a minimax approach to the problem of minimizing the learner's variance over the input distribution. The networks were trained on the same set of labelled examples, starting from one labelled example and adding one labelled example at a time. The disagreement in the predictions of the individual committee members arises from the differing random initializations of the network weights. A committee of five neural networks, each with 20 hidden nodes, was trained to approximate the univariate square wave function. This is a noise-free problem where the target function is not realizable. Active selection of training data led to improved performance compared to random selection.

RayChaudhuri and Hamey [8] used a similar approach to [7]. However, the disagreement among committee members arose from their being trained on different sub-samples of the available data. An initial random sample of ten points was queried. Ten neural networks were each trained on random sub-samples of half the data. An unlabelled input with maximal variance over the committee is then selected. This approach is similar to query by bagging used in classification [3]. It was argued that this approach should lead to better performance than [7]. The target function was generated by a univariate neural network with three hidden nodes. The committee members all had the same architecture. The emphasis was not to minimize generalization error, but to minimize data collection. Active learning was shown to require fewer queries than passive learning to reach

agreement among the committee members, but its advantage was reduced when a small amount of noise was added (signal-to-noise ratio (SNR) ca. 40). Passive learning was superior when more noise was added (SNR ca. 2.5). In [9], a committee of five neural networks with one hidden node was used to approximate the step function using the criterion of [8]. Limited evidence was provided that their approach outperformed that of [7]. In [10], it was demonstrated that this active learning approach could be used for system identification by training a neural network on all of the data actively selected and using it as a feed-forward controller. In [11], RayChaudhuri and Hamey propose a similar criterion to estimate the variance of a learner at x. Instead of ambiguity they use a jackknifed [12] estimate of the variance combined with a noise estimate. Generalization error when using this selection criterion is not significantly different from that obtained using ambiguity for the problems considered here.

All of the criteria defined above were proposed within the framework of *selective sampling*. It is assumed that a set of unlabelled inputs is provided and we wish to query the labels of as few inputs as possible whilst minimizing the generalization error. In the toy problems considered in [5,6,7,8,9,10,11], it was assumed that the input distribution, $q(x)$, was uniform, and that the label of any input point, $\tilde{x}$, drawn from $q(x)$ could be queried. There are thus two basic approaches to optimizing the query criterion. As suggested in [5], the criterion could be optimized by hillclimbing on $\partial \left\langle \tilde{\sigma}^2_{\hat{y}} \right\rangle / \partial \tilde{x}$. The same idea could be applied to the other two criteria, by using a gradient free search to find a local maximum. In practice, in low dimensions, it is computationally more efficient to draw m candidate points from $q(x)$ at each iteration and choose the best $\tilde{x}$ from these. This is known as *pool-based* selective sampling. For example, [5] choose $m = 64$, [7] choose $m = 800$ and [8] choose $m = 100$. In situations where active learning outperforms passive learning, we would expect a large m to be beneficial. Conversely, in situations where active learning performs badly, a large m could lead to a substantial deterioration in performance[1].

Use of a committee to estimate the variance of a learner as a query criterion for active learning does not appear to have been pursued further than the foregoing references[2]. The aim of this paper is to investigate the performance of these criteria under various conditions.

3 Numerical Examples

The passive (P) selection strategy selects the next input point, $\tilde{x}$ at random from the unlabelled examples. For the active strategy, a committee of $k = 5$ learners is maintained. Each is trained on a subset of the labelled data by leaving out disjoint subsets of size $\lfloor n/k \rfloor$, where n is the number of labelled data. The

[1] Expressed in the saying 'give 'em enough rope and they'll hang themselves.'

[2] According to CiteSeer (`http://citeseer.ist.psu.edu/`), apart from the work by RayChaudhuri and Hamey, [7] has only been cited once with reference to active learning, viz. [13]. The work by RayChaudhuri and Hamey has also only been cited once, viz. [14]. Neither of these later works took a committee-based approach.

ambiguity (A) selection strategy selects $\tilde{x}$ to maximize $\overline{a}(x)$, $\tilde{x}$ is chosen from a pool of $m = 1000$ unlabelled examples. Results are averaged over 1000 runs.

We consider the toy 1-d problem described in [15]. The input dimension is $d = 1$ and the target function is:

$$f(x) = 1 - x + x^2 + \delta r(x) \ .$$

where

$$r(x) = \frac{z^3 - 3z}{\sqrt{6}} \text{ with } z = \frac{x - 0.2}{0.4} \ .$$

The number of queries is $n = 100$ and the labels are corrupted by i.i.d. noise $\sim N(0, 0.3^2)$. The test input density is $q(x) = N(0.2, 0.4^2)$ and is assumed to be known. The model class is linear regression with a polynomial kernel of order two. Three cases are considered, $\delta = 0, 0.005, 0.05$, termed *correctly specified*, *approximately correct*, and *misspecified*, respectively.

The signal-to-noise (SNR) ratio for this problem is $0.4^2/0.3^2 = 1.8$. This is roughly the same as the 'highly noise' problem in [8]. It was observed therein that ambiguity criterion did not outperform a passive selection strategy in the 'highly noisy' case. Following [8], we also consider low-noise, $\sigma_\epsilon = 0.1$ and zero-noise, $\sigma_\epsilon = 0.0$, versions of the same problem.

High Noise. Box-plots of the generalization error for the high noise case are in figure 1 (*top*). The mean and standard deviation of the generalization error are given in table 1. The passive committee learner does not have significantly different generalization error than reported in [15]. For the correctly specified and approximately correct cases, active selection does not have significantly different performance to passive selection. For the misspecified case, active selection performs significantly worse than passive learning. We do not necessarily expect the active strategy to work well in this case since the assumption that the bias is approximately zero has been violated.

Low Noise. Box-plots of the generalization error for the low noise case are in figure 1 (*middle*). The mean and standard deviation of the generalization error are given in table 2. There are no significant differences in performance between the passive and active strategies for the correctly specified case. In the misspecified and approximately correct cases, the performance of the active strategy is significantly worse than that of the passive strategy.

Table 1. The mean and standard deviation of the generalization error in the high-noise case. The best method and comparable ones by the t-test at 95% are emphasized in bold. All values in the table are multiplied by 10^3.

	$\delta = 0$	$\delta = 0.005$	$\delta = 0.05$
P	**2.86±2.35**	**2.90±2.62**	**6.11±3.44**
A	**2.74±4.24**	**3.09±4.45**	46.5±14.6

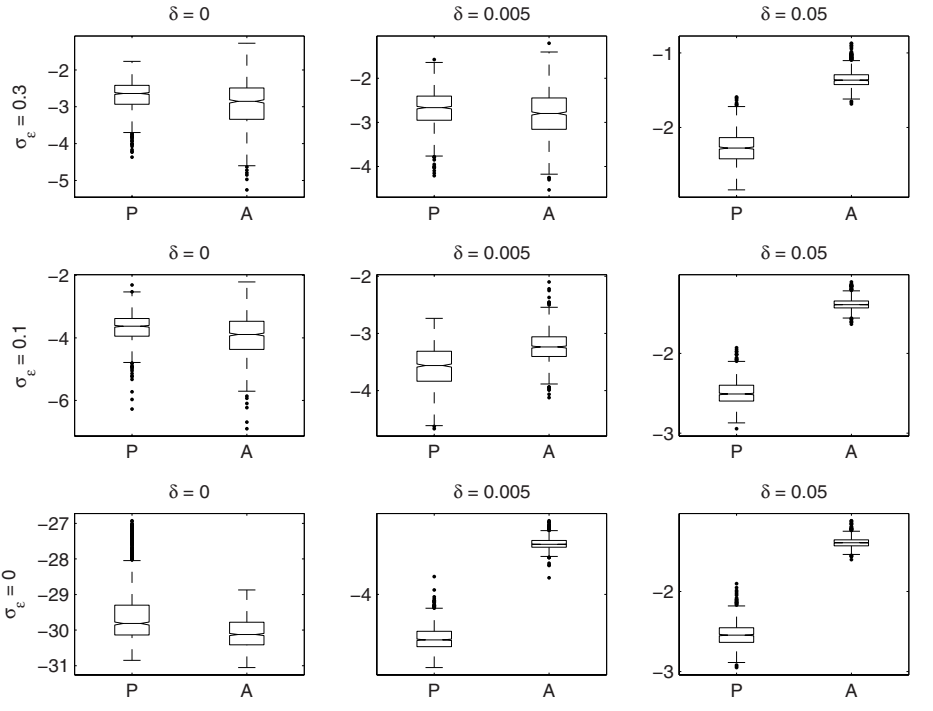

Fig. 1. Box-plots of the generalization error for the passive (P) and ambiguity (A) selection strategies. The box-plots show the distribution of $\log_{10}$ of the mean squared error on 1000 test points. The notches indicate robust estimates of the median generalization error.

Zero Noise. Box-plots of the generalization error for the zero noise case are given in figure 1 (*bottom*). The mean and standard deviation of the generalization error are given in table 3.

Active learning outperforms the passive strategy in the correctly specified case. This confirms the result of [8]. In the approximately correct and misspecified cases the active learning strategy is significantly worse than the passive strategy. This result is at odds with that reported in [7], although they used

Table 2. The mean and standard deviation of the generalization error in the low-noise case. The best method and comparable ones by the t-test at 95% are emphasized in bold. All values in the table are multiplied by 10^3.

	$\delta = 0$	$\delta = 0.005$	$\delta = 0.05$
P	**0.31±0.33**	**0.36±0.28**	**3.44±1.35**
A	**0.30±0.48**	0.76±0.64	41.8±7.34

Table 3. The mean and standard deviation of the generalization error in the zero-noise case. The best method and comparable ones by the t-test at 95% are emphasized in bold. All values in the table are multiplied by 10^3, except the values for $\delta = 0$, which are multiplied by 10^{30}.

	$\delta = 0$	$\delta = 0.005$	$\delta = 0.05$
P	42.5±139	**0.03±0.01**	**3.11±1.24**
A	**1.35±1.65**	0.42±0.07	41.3±6.18

a different method to create diversity across the ensemble. The generalization performance actually *worsened* as more points were queried.

In summary, active learning outperforms the passive strategy when the model class is correctly specified and there is no output noise. This performance gain is lost when the outputs are noisy or the model class is misspecified. Further consideration of this issue is given below.

3.1 Discussion

The problems of stagnation and deterioration in performance for the active learning strategy could be avoided by tracking the variance. This could be estimated from the working set, as in [8], or from a separate reference set, as in [5]. If the variance doesn't decrease then training can be halted, or the learner can switch to a passive strategy. A stochastic approach could also be taken, whereby the learner chooses an active or passive strategy probabilistically based on the change in average variance induced by the previous query. We do not investigate these ideas further but consider why the active strategy does not perform well.

In active learning the input density, $p(x)$, of training points differs from the input density, $q(x)$, of unlabelled points. This is known as the *covariate shift*. When $\delta = 0, \sigma_\epsilon = 0$, the active strategy outperform the passive strategy and empirically $p(x)$ is as shown in the left of figure 2. For the other cases, the active strategy does not outperform the passive strategy and $p(x)$ is as shown in the right of figure 2. In the previous work [7,8,9,10] $q(x)$ was taken to be uniform. This is likely to be the case in system identification and control [10]. We repeated the above experiments with $q(x)$ uniform having the same mean and standard deviation. Passive learning outperformed the active strategy in all cases.

When the model used for learning is correctly specified, the covariate shift does not matter since ordinary least squares (OLS) regression is unbiased under a mild condition [15]. In this case the median generalization error[3] of (A) is significantly lower than that of (P). When the model is misspecified, OLS is no longer unbiased. The active strategy investigated here is a variance-only method so we do not necessarily expect it to perform well when $\delta > 0$ and it does not. It is known that a form of weighted least squares is asymptotically unbiased for

[3] The expected generalization error of (A) is not significantly different from that of (P) since the active strategy is more variable in performance (see tables 1–3).

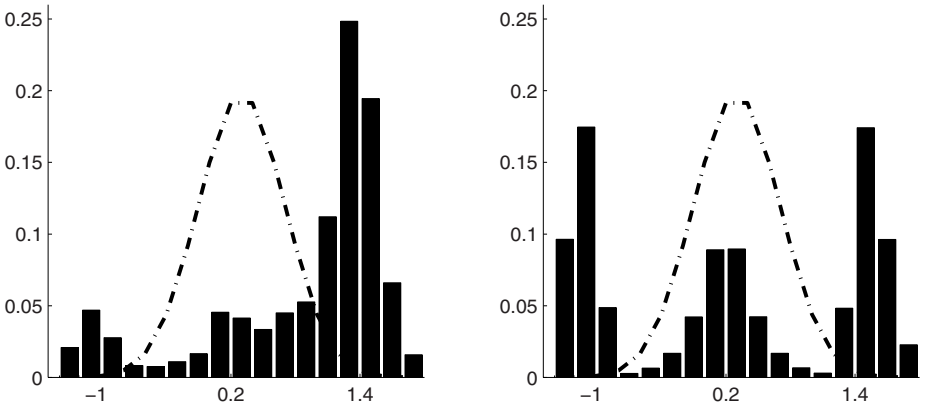

Fig. 2. Empirical input density, $p(x)$, for the active strategy. For the correctly specified, zero-noise case (*left*), the active strategy outperform the passive strategy. For the other cases (*right*), the active strategy perform the same or worse as the passive strategy. The test input distribution, $q(x)$, is superimposed.

misspecifed models [16]. Each training input point, x_i, is weighted by $q(x_i)/p(x_i)$. This requires an estimate of $p(x)$. We repeated the above experiments, at each iteration estimating $p(x)$ and using weighted least squares. The density $p(x)$ was estimated as $N(\mu, \sigma)$ where μ and σ are the maximum likelihood estimates. This did not improve the performance of the active strategy. It would be preferable to estimate $p(x)$ as a mixture of Gaussians, but this is not reliable with so few data, especially in higher dimensions.

For most real-life problems in function estimation and systems control, the target function or system is noisy and not perfectly realizable. When the model class is not correctly specified, this variance-only active learning strategy seems at best useless and at worst counterproductive. We now consider a different scenario and outline how to avoid the above problems.

When the model class is correctly specified, the active strategy is not significantly better than passive learning. When the model class is not correctly specified, the active strategy may be worse than passive learning. However, we have only considered cases where the model class is underspecified, i.e. learning tasks for which the learner is underfitting the available data. In most real-world situations, the functional form of the system is unknown. Ideally, we would use a very expressive model class in order to approximate the system as closely as possible. For example, it is known that a neural network with enough hidden nodes is able to approximate any function to arbitrary accuracy. This is not possible with noisy data since it will lead to overfitting. Overfitting occurs when the model variance is high [17]. Therefore, one way to avoid overfitting is to actively minimize the variance of the learner. This is exactly the approach taken by the active learning strategy considered here. Moreover, when the model class is very expressive the learner's bias is small.

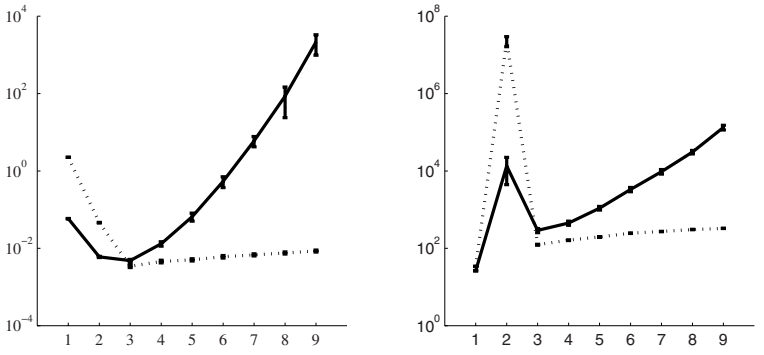

Fig. 3. Generalization error after 100 queries for the passive (*solid line*) and ambiguity (*dotted line*) selection strategies with a polynomial kernel as a function of the order of the polynomial for the toy data set (*left*) and the Boston housing data set (*right*). The error bars indicate 95% confidence limits for the mean.

In the following we consider only the misspecified, highly noisy case as this is the most realistic. The generalization error of (P) and (A) with polynomial kernels of order $\theta = 1, \ldots, 9$ is illustrated in figure 3 (*left*). When $\theta = 1, 2$, the learner underfits and (P) has lower generalization error than (A). As θ increases the model class becomes more expressive and performance of (P) deteriorates. However, (A) is robust to overspecification and has lower generalization error than (P) for $\theta \geq 4$. The lowest generalization error after 100 queries is for $\theta = 3$.

The Boston housing data set [18] has 506 examples and 13 attributes. The passive and active strategies were used to query the label of 100 points with polynomial kernels of orders $\theta = 1, \ldots, 9$. The generalization error averaged over 1000 runs is shown in figure 3 (*right*). The lowest error is at $\theta = 1$, when the passive strategy slightly outperforms the active. However, again the active strategy is much more robust to overspecification of the model class.

4 Conclusion

We have investigated a seemingly forgotten strategy for active selection of training data in real-valued function estimation. The main idea is to train a committee of learners and query the labels of input points where the committee's predictions differ, thus minimizing the variance of the learner by training on input points where variance is largest. This approach only works when the learner's bias is small. Its main advantage is that it is more robust to overspecification of the model order and thus less prone to overfitting than the passive strategy. The attractiveness of a committee-based approach is that there are no restrictions on the form of the committee members provided they are not identical.

References

1. Seung, H., Opper, M., Sompolinsky, H.: Query by committee. In: Proceedings of the Fifth Workshop on Computational Learning Theory, pp. 287–294. Morgan Kaufmann, San Mateo, CA (1992)
2. Freund, Y., Seung, H., Shamir, E., Tishby, N.: Selective sampling using the query by committee algorithm. Machine Learning 28(2–3), 133–168 (1997)
3. Abe, N., Mamitsuka, H.: Query learning strategies using boosting and bagging, pp. 1–9. Morgan Kaufmann, San Francisco, CA (1998)
4. Geman, S., Bienenstock, E., Doursat, R.: Neural networks and the bias/variance dilemma. Neural Computation 4, 1–58 (1992)
5. Cohn, D.A., Ghahramani, Z., Jordan, M.I.: Active learning with statistical models. Journal of Artificial Intelligence Research 4, 129–145 (1996)
6. Cohn, D.: Neural network exploration using optimal experiment design. Morgan Kaufmann, Denver, CO (1996)
7. Krogh, A., Vedelsby, J.: Neural network ensembles, cross validation, and active learning. In: Tesauro, G., Touretzky, D., Leen, T. (eds.) Advances in Neural Information Processing Systems 7, pp. 231–238. MIT Press, Denver, CO (1995)
8. RayChaudhuri, T., Hamey, L.G.: Minimisation of data collection by active learning. In: Proceedings of the 1995 IEEE International Conference on Neural Networks, Perth, Australia, vol. 3, pp. 1338–1341 (1995)
9. RayChaudhuri, T., Hamey, L.G.: Accurate modelling with minimised data collection — an active learning algorithm. In: Proceedings of the Seventh Australian Conference on Neural Networks, pp. 11–15. Australian National University, Canberra, Australia (1996)
10. RayChaudhuri, T., Hamey, L.G.: Active learning for nonlinear system identification and control. In: Gertler, J., Cruz, J.J.B., Peshkin, M. (eds.) Proceedings of 13th International Federation of Automatic Control World Congress, vol. F, pp. 193–197. Pergamon, San Francisco, CA (1996)
11. RayChaudhuri, T., Hamey, L.G.: Cost-effective querying leading to dual control. Technical Report 96-07, School of MPCE, Macquarie University, New South Wales 2109, Australia (June 1996)
12. Quenouille, M.: Approximate tests of correlation in time series. Journal of the Royal Statistical Society B 11, 68–84 (1949)
13. Zhang, C., Chen, T.: An active learning framework for content-based information retrieval. IEEE Transactions on multimedia 4(2), 260–268 (2002)
14. Lindenbaum, M., Markovitch, S., Rusakov, D.: Selective sampling for nearest neighbor classifiers. Machine Learning 54(2), 125–152 (2004)
15. Sugiyama, M.: Active learning in approximately linear regression based on conditional expectation of generalization error. Journal of Machine Learning Research 7, 141–166 (2006)
16. Weins, D.: Robust weights and designs for biased regression models: least squares and generalized M-estimation. Journal of Statistical Planning and Inference 83(2), 395–412 (2000)
17. Bishop, C.: Neural Networks for Pattern Recognition. Clarendon Press, Oxford (1995)
18. Harrison, D., Rubinfeld, D.: Hedonic prices and the demand for clean air. J. Environ. Economics and Management 5, 81–102 (1978)