
Regression NSS: An Alternative to Cross Validation

Michael P. Perrone

IBM T.J. Watson Research Center
Yorktown Heights, NY 10598
mpp@watson.ibm.com

and

Institute for Brain and Neural Systems
Brown University

Brian S. Blais

Institute for Brain and Neural Systems
Brown University
Providence, RI 02912
bblais@cns.brown.edu

Abstract

The Noise Sensitivity Signature (NSS), originally introduced by Grossman and Lapedes (1993), was proposed as an alternative to cross validation for selecting network complexity. In this paper, we extend NSS to the general problem of regression estimation. We also present results from regularized linear regression simulations which indicate that for problems with few data points, NSS regression estimates perform better than Generalized Cross Validation (GCV) regression estimates [7].

1 The Noise Sensitivity Signature (NSS)

One of the fundamental problems in neural networks and regression estimation in general is to develop reliable methods for avoiding overfitting to finite data. Grossman and Lapedes (1993) proposed the NSS for classification problems as a method for selecting neural network complexity to match the complexity of the data available and in this way reduce overfitting. Unlike other methods for avoiding overfitting, such as cross validation [5], which uses only part of the available data, and unlike common penalty functions [8, 4] which cannot adapt to new data, the NSS approach uses all of the data in training and is manifestly data-adaptive.

The basic idea behind NSS is to generate a “noisy” data set, \mathcal{N} , by adding artificial noise to a given data set, \mathcal{D} , and to use \mathcal{N} to train a network. The performance of the network optimized on \mathcal{N} is then tested using the original data set, \mathcal{D} . If the training error is lower than the testing error, then the network has overfit to the artificially added noise and therefore the network has more than sufficient complexity to model the data. If on the other hand, the testing error is lower than the training error, then the network has insufficient complexity. Ideally, one would choose the network complexity such that the average training and testing performance are equal.

Grossman and Lapedes (1993) implemented this selection process for a binary classification problem in the following way: Define $Q_n(p)$ as the percent correct classification from a neural network trained on noisy data for which a fraction, $p \in [0, 1]$, of the total data set is randomly selected and has had noise added. For classification data, noise is added by flipping the class label of the input point. Define $Q_f(p)$ as the percent correct classification of the same network tested on the noise free data. The $Q_n(p)$ and $Q_f(p)$ are random variables that depend on the finite training set and the choice of noise added. For each p in a range of p ,

one then trains several networks of varying complexities and plots Q_n and Q_f vs. p for each network. The NSS criterion is to select the network complexity for which the plots of Q_n and Q_f vs. p are most similar. Once a complexity is selected, training is performed on all of the noise free data.

In Section 2, we generalize the NSS complexity criterion to regression. In Section 3, we give a specific example by showing how NSS can be applied to ridge regression. We present experimental results of NSS complexity selection in Section 4 and conclude with a discussion in Section 5.

2 Regression NSS

Consider the standard regularized regression problem. We are given the set $\mathcal{D} = \{(x_i, y_i)\}$ where y_i is generated by

$$y = g(x) + \epsilon, \quad (1)$$

$g(\cdot)$ is some unknown function, and ϵ is a random variable with zero mean and unknown variance. We would like to find the best possible estimate \hat{f} of g . In general, we minimize a cost, $C(f, \mathcal{D}, \lambda)$, given by

$$C(f, \mathcal{D}, \lambda) = E(f, \mathcal{D}) + \lambda R(f) \quad (2)$$

where $E(f, \mathcal{D})$ is a measure of how well the regression function f fits the data; $R(f)$ is a measure of the complexity of f ; and λ is a non-negative tuning parameter which determines the relative importance of the solution's fit and complexity.

NSS was specifically designed to identify the optimal complexity for a classification network. We therefore extend NSS to regression by using it to select an optimal value for the tuning parameter, λ , and thus the complexity of f . This selection can be accomplished in the following way.

We begin in analogy to [2] by defining an artificial noise process which takes \mathcal{D} to $\mathcal{D} + \delta$ such that each (x, y) becomes $(x + \epsilon_x, y + \epsilon_y)$ where ϵ_x and ϵ_y are zero mean random variables. The amount of noise is regulated by its variance. Using this notation, we see that Q_n and Q_f are given by

$$Q_n(f) = C(f, \mathcal{D} + \delta, \lambda) \quad (3)$$

and

$$Q_f(f) = C(f, \mathcal{D}, \lambda). \quad (4)$$

We now define

$$f^* = \arg \min_f Q_n(f). \quad (5)$$

According to the NSS criterion,

$$Q_f(f^*) > Q_n(f^*) \quad (6)$$

suggests that complexity should be increased while

$$Q_f(f^*) < Q_n(f^*) \quad (7)$$

suggests that complexity should be decreased; or in other words, the Q_f and Q_n will be almost equal at the optimal complexity. To capture this behavior, we choose the λ which minimizes

$$\lambda_{\text{NSS}} = \arg \min_{\lambda} \int E \left[(Q_f(f^*) - Q_n(f^*))^2 \right] d\sigma^2 \quad (8)$$

where the expected value is over all configurations of δ for a given noise variance, σ^2 ; and the integral is over all possible variances. We note at this point that this integral is in general unbounded¹. This unboundedness can be handled by adding a measure on the variance space, in effect making the integral into an expected value. However, this approach leaves the problem of which measure to choose. Here we can be guided by the intrinsic variance of the data set, \mathcal{D} , which sets a natural scale for any artificial noise that we might add.

¹We will see this for a particular regression setting in Eqn. 16 where we solve the problem by renormalizing.

Once λ_{NSS} is found using Eqn. 8, the optimal regression function is given by

$$f_{\text{NSS}} = \arg \min_f C(f, \mathcal{D}, \lambda_{\text{NSS}}). \quad (9)$$

The original NSS also made use of a measure of the distance, D , between minimizers of Q_n given various noise configurations. We generalize this measure to regression by defining

$$D_\lambda = E[(f_1 - f_2)^2] \quad (10)$$

where f_i minimizes $Q_n(f, \lambda, \mathcal{D}, \delta_i)$; δ_1 and δ_2 are independent noise configurations; and the expected value is taken over all δ_1 and δ_2 with the same variance. D_λ can be thought of as another measure of complexity (or “stiffness”) since as complexity increases, f_1 and f_2 will wiggle around more and thus D_λ will increase. However, it is not clear how to use this “stiffness” measure in the regression setting. (See Section 3.2.)

3 NSS and Ridge Regression

In this section we give an explicit example of how Regression NSS can be used in a linear regression problem. In the case of linear regression, Eqn. (1) becomes

$$y_i = \beta^t x_i + n_i \quad (11)$$

where β and x_i are d dimensional vectors and n_i is noise with zero mean and unknown variance. Defining Y as the N dimensional vector of y_i , X as the matrix of x_i vectors and N as the number of data points in \mathcal{D} , we find that ridge regression [6] sets

$$C(\beta, \mathcal{D}, \lambda) = (Y - X^t \beta)^t (Y - X^t \beta) + \lambda \beta^t \beta. \quad (12)$$

In order to derive a Q_f and Q_n from C , we must specify the type of artificial noise we will use. We have chosen $\epsilon_y \sim \mathcal{N}(0, \sigma^2)$ and $\epsilon_x = 0$; and define ϵ as the N dimensional vector of ϵ_y 's. With this choice, Q_n becomes

$$Q_n(\beta, \mathcal{D} + \delta, \lambda) = (Y + \epsilon - X^t \beta)^t (Y + \epsilon - X^t \beta) + \lambda \beta^t \beta. \quad (13)$$

Minimizing Q_n gives

$$\beta^* = (X X^t + \lambda)^{-1} X (Y + \epsilon). \quad (14)$$

For convenience define

$$\Delta Q = Q_n(\beta^*, \lambda, \mathcal{D}, \epsilon) - Q_f(\beta^*, \lambda, \mathcal{D}, \epsilon) \quad (15)$$

We can now use Eqns. 14 and 15 to write the integrand of Eqn. 8 as

$$E[N^2(\Delta Q)^2] = \sigma^4 \left[\text{tr}^2(1 - 2A) + 2\text{tr}(1 - 2A)^2 \right] + \sigma^2 Y^t (1 - A)^2 Y \quad (16)$$

where $A \equiv X^t (X X^t + \lambda)^{-1} X$. One should note two things about Eqn. 16. First, it is reassuring to note that as the noise variance approaches zero, we recover the usual ridge regression term; and second, the integrand is unbounded unless both $\text{tr}^2(1 - 2A) + 2\text{tr}(1 - 2A)^2 = 0$ and $Y^t (1 - A)^2 Y = 0$ which is not true in general. Thus the integral in Eqn. 8 is unbounded and must be renormalized. We renormalize here by dividing the integral by $\int \sigma^4 d\sigma^2$, restricting both integrals to a finite interval and then taking the limit as the interval increases to include all possible variances. This renormalization results in the following NSS selection rule for ridge regression:

$$\lambda_{\text{NSS}} = \arg \min_\lambda \left(\text{tr}^2(1 - 2A) + 2\text{tr}(1 - 2A)^2 \right). \quad (17)$$

We use this criterion to derive our experimental results. (See Section 4.)

3.1 Alternative NSS Formulation

Another approach is to find the λ for which

$$\int E[\Delta Q] d\sigma^2 = 0. \quad (18)$$

This equation is more like what is proposed by Grossman and Lapedes (1993) than Eqn. 8 since they compare Q_n and Q_f only after averaging over different configurations of the noise. In the ridge regression framework, Eqn. 18 can be written

$$\text{tr}[1 - 2A] \int \sigma^2 d\sigma^2 = 0 \quad (19)$$

and so a single value of λ satisfies Eqn. 18 for all values of the artificial noise variance and no renormalization is needed. Equation 19 implies that

$$\text{tr}[X^t (XX^t + \lambda)^{-1} X] = \frac{N}{2}. \quad (20)$$

The left hand side of this equation is commonly referred to as the “effective number of parameters” in the model [3]. In this case we see that the NSS chooses the effective number of parameters to be half the number of data points.

Diagonalizing XX^t , the above condition for selecting λ can be re-written as

$$\sum_i \frac{\lambda_i}{\lambda_i + \lambda} = \frac{N}{2} \quad (21)$$

where the λ_i are the eigenvalues of XX^t . Since we require that $\lambda > 0$, we find that $N < 2d$ where d is the dimensionality of the input space. This suggests that the NSS criterion may only be useful when the number of data points is not much larger than the number of dimensions. Although this result is derived here for the case where $E[\Delta Q] = 0$, it appears to be more fundamental since similar behavior is observed in our experimental results using Eqn. 8. (See Section 4.)

In our initial work, we attempted to use Eqn. 18 to select λ . However, this approach did not give satisfactory results which may be due to the fact that although the average ΔQ was zero, no constraint was placed on the variance of ΔQ . Equation 8 is a method for overcoming this problem. Since Eqn. 8 minimizes the second moment of ΔQ , it is simultaneously minimizes the the variance and the squared mean of ΔQ . One further approach is to minimize the variance directly. This direction will be pursued in future work.

3.2 D_λ and Ridge Regression

For ridge regression, we can evaluate Eqn. 10 in closed form to find

$$D_\lambda = 2\sigma^2 \text{tr}[A^2] \quad (22)$$

Note that D_λ is linear in σ^2 and thus the original NSS approach of looking at the sign of the curvature of D_λ does not work here. Another rule used by the original NSS was to increase the complexity if D_λ goes to zero as σ^2 goes to zero. Here D_λ goes to zero for all values of complexity suggesting that we should set $\lambda = 0$ (i.e. minimize stiffness and maximize complexity). This results in basic linear regression and is not satisfactory. For these reasons, we did not use D_λ in our simulations. However D_λ may still be useful in some way particularly because of its close relation to the effective number of parameters. (See Eqns. 19 and 20.) It would be interesting to see how the stiffness could be usefully incorporated into the NSS criterion presented in this paper.

4 Experimental Results

In order to test the NSS complexity selection criterion outlined in Section 3, we performed a series of ridge regression simulations. In each case, we calculated the MSE between the true solution and the estimate

found using the NSS choice of λ (Eqn. 17), the Generalized Cross Validation (GCV) choice of λ given by [7]

$$\lambda_{\text{GCV}} = \arg \min_{\lambda} \frac{Y^t(1-A)^2Y}{[\text{tr}(1-A)]^2} \quad (23)$$

and simple linear regression (i.e. choosing $\lambda = 0$).

In analogy to Breiman (1992), we begin by selecting two β 's: β^{flat} has all elements equal to $1/\sqrt{20}$ and β^{ramp} has “ramped” elements such that the k th element is $\alpha k/20$ and α is chosen to normalize β^{ramp} . We define $x \sim \mathcal{N}(0, I)$ where I is a 20×20 identity matrix. Using this distribution, we generate X_i which is the i th instance of a $20 \times N$ matrix of N x vectors. Using X_i , β^{flat} , β^{ramp} , Eqn. 11 and $n \sim \mathcal{N}(0, \sigma_n^2)$, we generate the 20 dimensional vector Y_i^{flat} and Y_i^{ramp} . We now use X_i , Y_i^{flat} and Y_i^{ramp} to calculate the NSS, GCV and unregularized estimates of β^{flat} and β^{ramp} and their corresponding MSE's. Thus, we can vary N and σ_n^2 and compare the relative performances of these estimates.

This comparison is shown in Figs. 1 and 2 for the NSS and GCV estimates averaged over 2000 different runs. Each figure contains three graphs: one for each of three values of σ_n^2 . Each graph shows the average MSE (calculated relative to the true β) as a function of the number of data points in the training set. The results for linear regression are not shown but they were dramatically worse than either NSS or GCV. This result was expected since $d < N < 2d$ for these experiments. The graphs also include the “optimal” average MSE (i.e. the average MSE given that the optimal λ is known for each data set). The optimal MSE is not available in practice but is included here to show what the best possible average performance is.

The most notable aspect of the graphs is that for small data sets and small to moderate noise levels, the NSS estimator has lower MSE than the GCV estimate. Note however that as the amount of data increases or as the noise increases, the GCV MSE falls below the NSS MSE. This suggests that NSS is a desirable technique when the noise is low and/or the data is sparse.

Note also that Eqn. 17 does not depend on Y . This implies that in this case, the NSS criterion has no knowledge of the noise variance and can not adjust for it as it increases. This may explain the lower performance at higher noise variances. This also suggests that this problem can be overcome by choosing an appropriate measure for the integral in Eqn. 8. We are investigating this further.

Fig. 3 presents similar results using a scaled version of β^{flat} such that the resulting β had length $\sqrt{5}$. The effect was to raise the signal to noise ratio² by a factor of 5 leading to a corresponding increase in the performance of the NSS MSE. Comparing between graphs with the same signal to noise ratio in Figs. 1, 2 and 3; and we see that graphs with comparable ratios are qualitatively similar. This suggests that the observed behavior of the NSS is universal.

In another experiment, we recalculated the graphs for β^{ramp} in a 10 dimensional space and allowed for $N > 2d$. The results are shown in Fig. 4. In this experiment we see for low N qualitatively the same behavior that that we saw in Figs. 1 and 2 (since they are at the same signal to noise ratio); however, since N is allowed to increase beyond $2d$ we see some new behavior. At $N > 26$ (in this case), the NSS criterion begins to select $\lambda = 0$. We have observed that this behavior is a general feature of the NSS criterion: Typically, the average value of λ chosen by NSS drops linearly with N until the positivity constraint on λ forces all λ to zero for N high enough.

Fig. 5 shows the standard deviation of the average MSEs shown in Fig. 1. Note that in the region where NSS performs well in Fig. 1, the standard deviation of the NSS MSE in Fig. 5 is much lower than the standard deviation of the GCV MSE suggesting that the NSS estimators are more robust in this regime.

5 Discussion

We have presented a framework for extending the NSS criterion [2] to regression. This framework adds rigor to the original NSS criterion by making explicit the the complexity selection rule and by allowing for detailed mathematical analysis of the resulting equations. One disadvantage of the original NSS was that its selection rule was subjective. The method presented here removes this problem.

²Here we define the signal to noise ratio to be the ratio between the variance of $\beta^t x$ and σ_n^2 .

Further, we have presented a realization of this framework for the case of linear regression which removes the computational burden of actually adding artificial noise to the data set. And we have presented computer simulations of the linear regression case which indicate that NSS performs better than GCV when both the noise level and the number of data points are low.

Acknowledgements

The authors would like to thank Tal Grossman for introducing us to NSS and for many useful discussions; and the anonymous reviewers for suggestions which helped improve this paper.

References

- [1] BREIMAN, L. Stacked regression. Technical Report TR-367, Department of Statistics, University of California, Berkeley, August 1992.
- [2] GROSSMAN, T., AND LAPEDES, A. Use of bad training data for better prediction. In *Advances in Neural Information Processing Systems* (1993), J. D. Cowan, G. Tesauro, and J. Alspector, Eds., vol. 6, Morgan Kaufmann, pp. 342–350.
- [3] MOODY, J. E. The effective number of parameters: An analysis of generalization and regularization in nonlinear learning systems. In *Advances in Neural Information Processing Systems*, J. E. Moody, S. J. Hanson, and R. P. Lippmann, Eds., vol. 4. Morgan Kaufmann, San Mateo, CA, 1992, pp. 847–854.
- [4] RISSANEN, J. Stochastic complexity and modeling. *Annals of Statistics* 14, 3 (1986), 1080–1100.
- [5] STONE, M. Cross-validatory choice and assessment of statistical predictions (with discussion). *J. Royal Statistics Society B* 36 (1974), 111–147.
- [6] VINOD, H. D., AND ULLAH, A. *Recent Advances in Regression Methods*. Marcel Dekker, Inc., 1981.
- [7] WAHBA, G. *Spline Models for Observational Data*. SIAM, Philadelphia, 1990.
- [8] WEIGEND, A. S., RUMELHART, D. E., AND HUBERMAN, B. A. Backpropagation, weight-elimination and time series prediction. In *Proceedings of the 1990 Connectionist Models Summer School* (1990), Morgan Kaufmann, pp. 105–116.

Figure 1: β^{flat} Experiments in 20 Dimensions: Graphs of the Average Mean Squared Error as a function of the number of points in the training set for each of three noise variances. The signal to noise ratios for the top, middle and lower graphs are 1:1, 1:0.2 and 1:0.04, respectively.

Figure 2: β^{ramp} Experiments in 20 Dimensions: Graphs of the Average Mean Squared Error as a function of the number of points in the training set for each of three noise variances. The signal to noise ratios for the top, middle and lower graphs are 1:1, 1:0.2 and 1:0.04, respectively.

Figure 3: $\sqrt{5}\beta^{\text{flat}}$ Experiments in 20 Dimensions: Graphs of the Average Mean Squared Error as a function of the number of points in the training set for each of three noise variances. The signal to noise ratios for the top, middle and lower graphs are 1:0.2, 1:0.04, 1:0.008, respectively.

Figure 4: β^{ramp} Experiments in 10 Dimensions: Graphs of the Average Mean Squared Error as a function of the number of points in the training set for each of three noise variances. The signal to noise ratios for the top, middle and lower graphs are 1:1, 1:0.2, 1:0.04, respectively.

Figure 5: β^{flat} Experiments in 20 Dimensions: Graphs of the standard deviation of the Mean Squared Error as a function of the number of points in the training set for each of three noise variances.

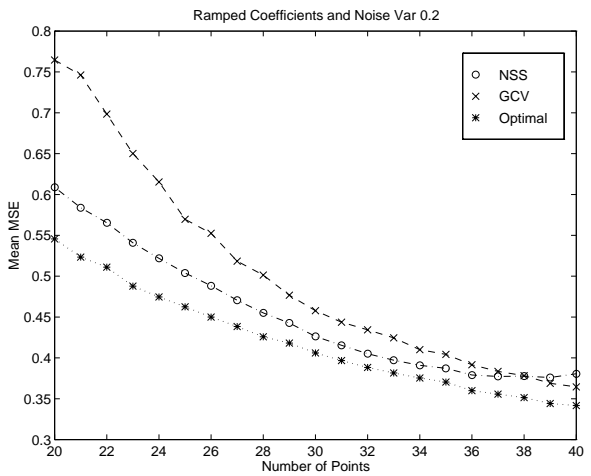
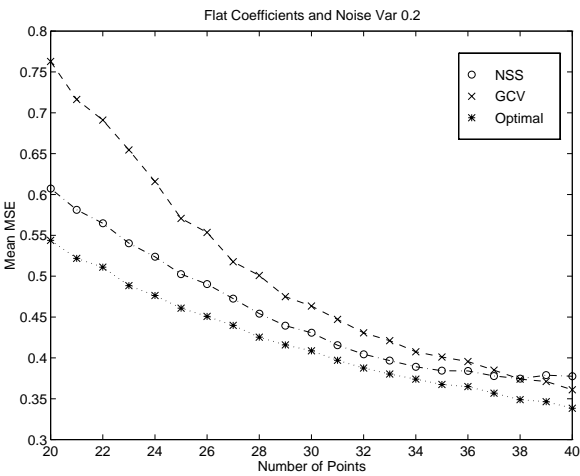
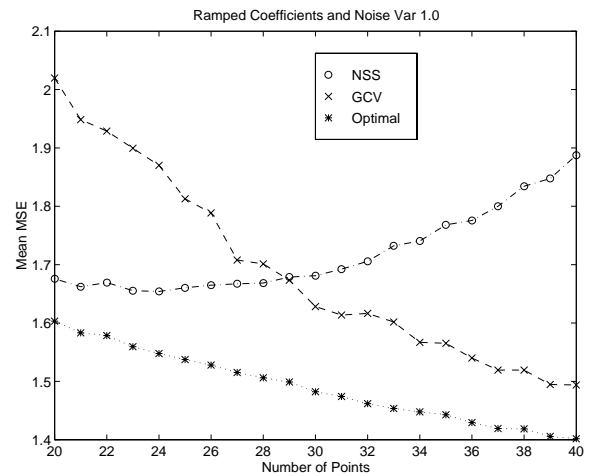
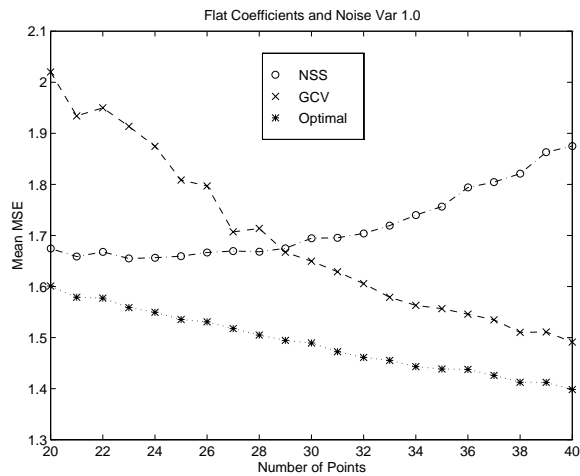
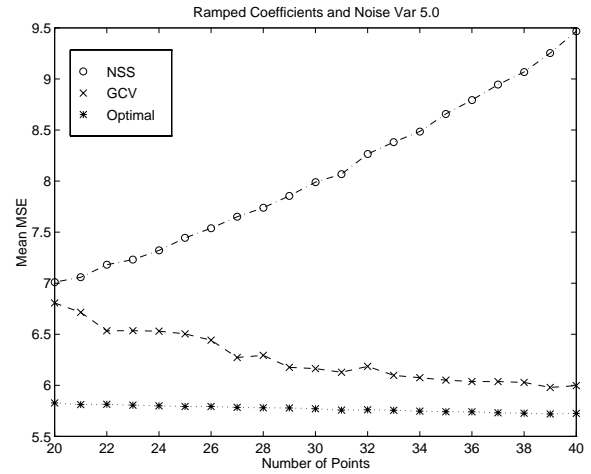
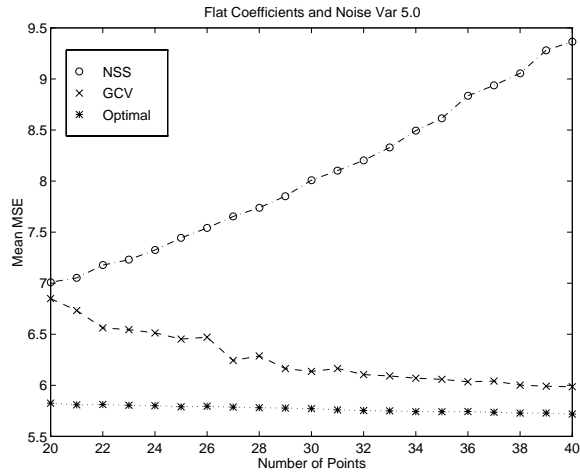


Figure 1: β^{flat} Experiments in 20 Dimensions: Graphs of the Average Mean Squared Error as a function of the number of points in the training set for each of three noise variances. The signal to noise ratios for the top, middle and lower graphs are 1:1, 1:0.2 and 1:0.04, respectively.

Figure 2: β^{ramp} Experiments in 20 Dimensions: Graphs of the Average Mean Squared Error as a function of the number of points in the training set for each of three noise variances. The signal to noise ratios for the top, middle and lower graphs are 1:1, 1:0.2 and 1:0.04, respectively.

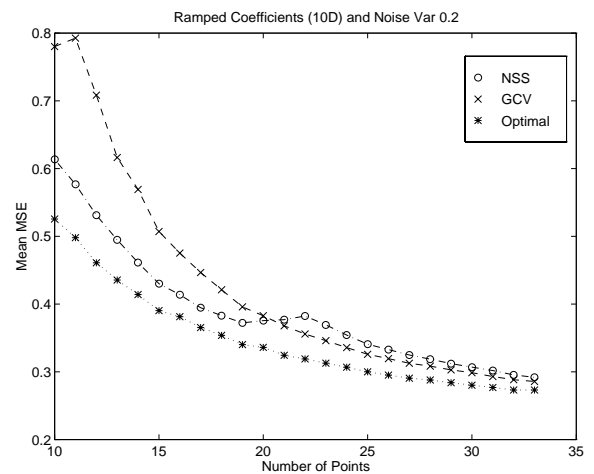
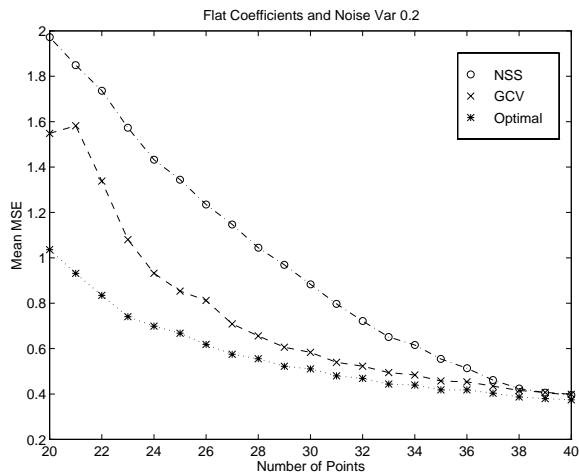
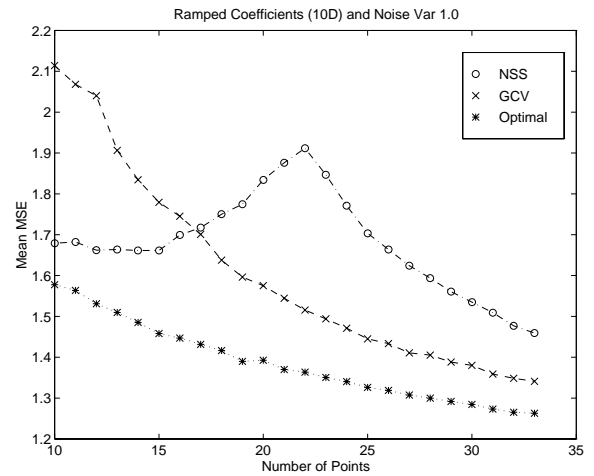
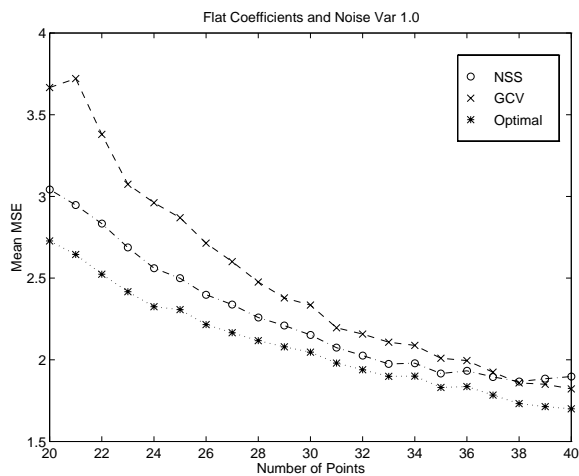
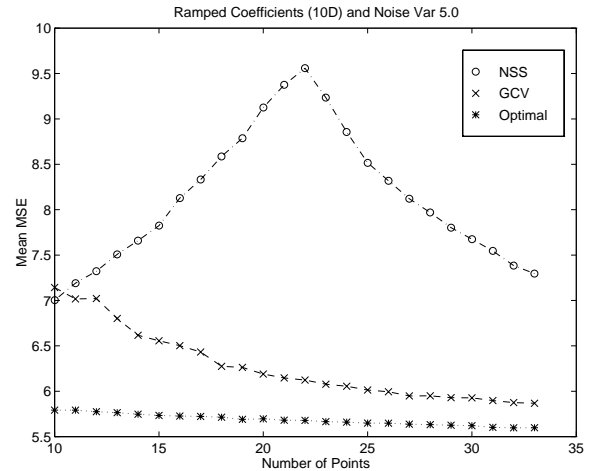
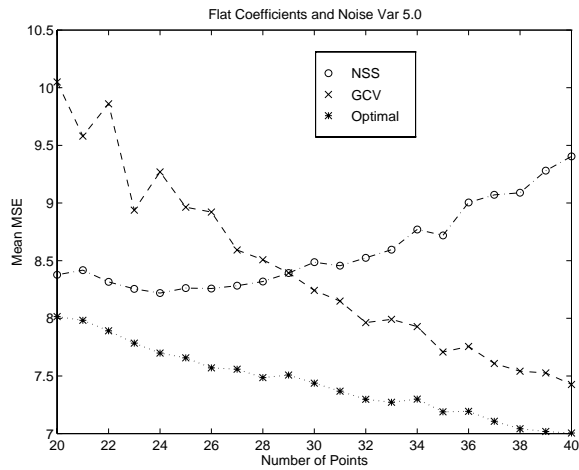


Figure 3: $\sqrt{5}\beta^{\text{flat}}$ Experiments in 20 Dimensions: Graphs of the Average Mean Squared Error as a function of the number of points in the training set for each of three noise variances. The signal to noise ratios for the top, middle and lower graphs are 1:0.2, 1:0.04, 1:0.008, respectively.

Figure 4: β^{ramp} Experiments in 10 Dimensions: Graphs of the Average Mean Squared Error as a function of the number of points in the training set for each of three noise variances. The signal to noise ratios for the top, middle and lower graphs are 1:1, 1:0.2, 1:0.04, respectively.

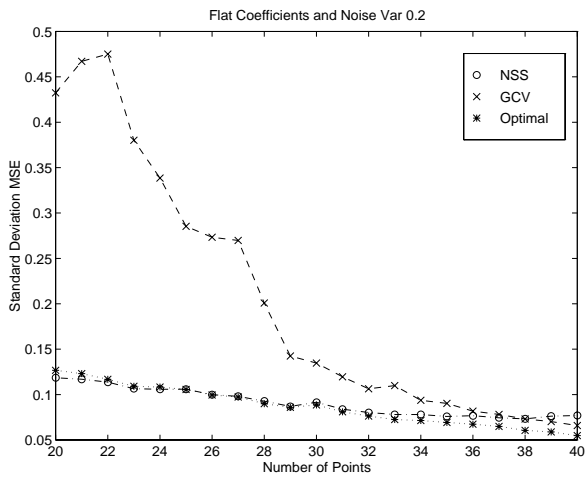
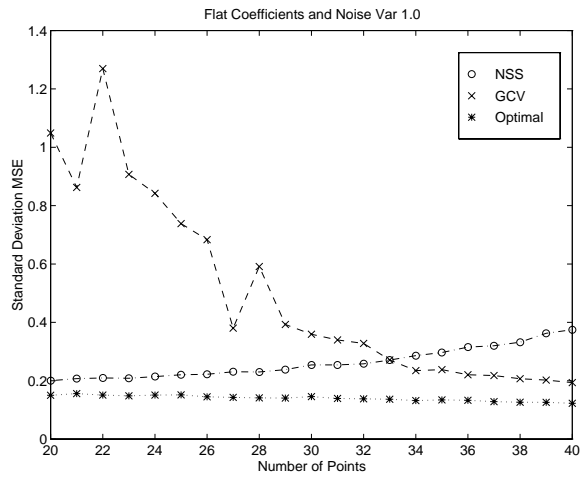
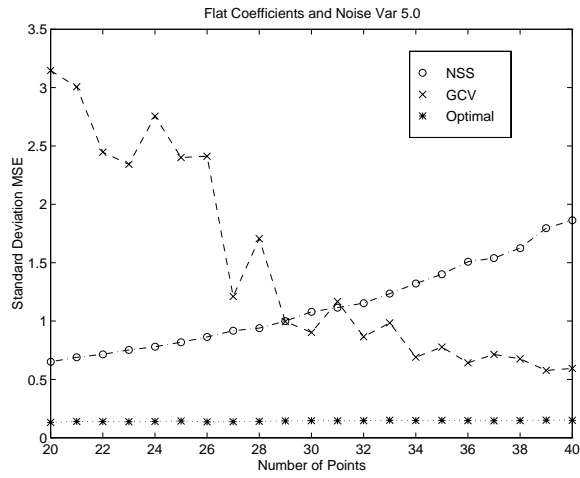


Figure 5: β^{flat} Experiments in 20 Dimensions: Graphs of the standard deviation of the Mean Squared Error as a function of the number of points in the training set for each of three noise variances.