SYSTEMATIC UNDERPREDICTION OF VOLATILITY IN MAXIMUM LIKELIHOOD METHODS

M. MAGDON-ISMAIL, Y.S. ABU-MOSTAFA
Caltech 136-93, Pasadena,
CA 91125, USA

In forecasting a financial time series, the mean prediction can be validated by direct comparison with the value of the series. However, the volatility or variance can only be validated by indirect means such as the likelihood function. Systematic errors in volatility prediction have an 'economic value' since volatility is a tradable quantity (e.g., in options and other derivatives) in addition to being a risk measure. We analyze the fidelity of the likelihood function as a means of training (in sample) and validating (out of sample) a volatility model. We report several cases where the likelihood function leads to an erroneous model. We correct for this error by scaling the volatility prediction using a predetermined factor that depends on the number of data points.

Keywords: Validation, Volatility Prediction, Maximum Likelihood

1 Introduction

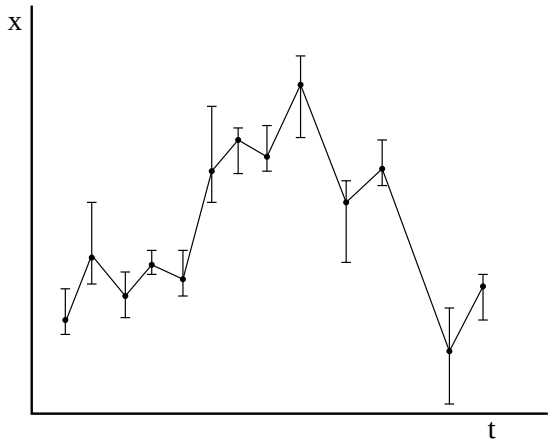

Figure 1.

Consider the time series depicted in Fig 1. Each point $x(t)$ is drawn from a distribution whose mean is $\mu(t)$ and variance is $\sigma^2(t)$. While we can only observe $x(t)$, we wish to learn about the mean $\mu(t)$ and the volatility $\tilde{\sigma}(t)$ (a normalized version of $\sigma(t)$). An accurate prediction of the mean tells us

about the expected behavior of the time series. An accurate prediction of the volatility is also important, especially in the case of a financial time series. Typically, volatility prediction is used as an explicit measure of risk in static hedging, portfolio selection and margining problems. It serves to place an error bar on the predicted value.

The question arises as to how one can judge various models that are predicting a non-explicit parameter like the variance. The variance falls into the class of non-explicit parameters because, on drawing a random variable from its distribution, no direct information on the variance is conveyed. If we have more than one drawing from the distribution then some direct information does exist on the variance but we would like to consider time series with time-varying distributions, so we only get one drawing from each distribution. As a result, one has to somehow infer information on the variance, and here lies the difficulty with variance prediction. We discuss how this can be done using maximum likelihood, and we take the special case of gaussian noise to illustrate our points.

The most striking result in this paper is that, for any finite number of data points, it is more likely than not that we will pick the worse of two specific models if we use the likelihood function to compare them. It turns out that maximum likelihood will lead to an *underestimation* of the volatility, even when the mean is predicted perfectly. This naturally leads to the question "can we correct for the systematic underestimation?". This allows us to choose from a class of models and then correct for the bias in the method of selection.

Volatility factors into a number of equations in finance. Black and Scholes (1973) derived option pricing models for which the expected future volatility is an important input. Kat (1993) has shown that more accurate volatility prediction will improve the replication efficiency of delta hedging strategies using Black–Scholes hedge ratios, even if the volatility is not constant. Crouchy (1995) shows that for path independent options, the option value depends only on the average volatility while the hedge ratio itself depends on the path of future volatility. The sensitivity of the hedge ratio to short term volatility (we are interested in time-varying volatility) is more of a problem for short term options than long term options. Nonetheless this sensitivity exists and so one would like to have an accurate estimate of the volatility.

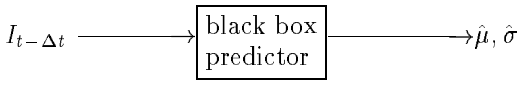

Figure 2.

We consider models that predict the mean and variance at time t as in Fig. 2. A variety of techniques exist for predicting variance or volatility. One can use the option prices to compute 'implied' volatilities, such as those derived from the Black-Scholes pricing equations. Another alternative is a multifactor model. Usually combinations of such models work best, but these models all include some constancy in the volatility. Schewert (1989) tests for the constancy of the volatility and strongly rejects this hypothesis. Thus one would like to have models that reasonably account for changing volatility. One can modify the Black-Scholes equations if the volatility is some known function of time by replacing the actual volatility by the average over the remaining life of the option. Autoregressive Conditional Heteroscedasticity (ARCH) type models introduced by Engle(1982) stipulate the conditional variance as a function of past innovations. Generalizations of this are GARCH (Bollerslev, 1986) and EGARCH (Nelson, 1991) where the conditional variance is a function of past innovations and variances. Hull (1987) has tried models that have stochastic parts to them.

One would like to have a reliable method for choosing between volatility models. Our goal is the selection of the optimal model given a number of models. Training can be viewed as a generalization of this where one chooses from a *class* of models (e.g., neural networks with a given architecture). We will evaluate maximum likelihood as a selection criterion between models.

2 Defining the Volatility

2.1 Variability versus Volatility

It is important to distinguish how "jagged" or "choppy" a time series is from the volatility. One might more accurately refer to the former as the variability. At any given time, if the mean value is known, the tendency of the actual value at that time to wander about this mean value is related to the volatility. In the case where the mean and the variance are constant, the variability will reflect the volatility. We are more concerned with time varying variance so this distinction should be made. A measure of this tendency for the actual outcome to be scattered about the mean is the variance of the value at time t.

2.2 Correspondence with the Black-Scholes Volatility Models

In modeling the time variation of a stock price, Black-Scholes assume an Ito Process (Ito, 1951) of the form $\Delta S = \tilde{\mu} S \Delta t + \tilde{\sigma} S \epsilon \sqrt{\Delta t}$. $\epsilon \sim N(0,1)$, S is the instrument price and $\tilde{\sigma}$ is defined as the volatility and is the standard deviation of the proportional change in the stock price in unit time. It is this $\tilde{\sigma}$ that

enters into the calculation of the hedge ratios, option prices, etc. Hence, in the Black-Scholes model we can identify the volatility as

$$\tilde{\sigma} = \frac{\sigma}{S_{t-\Delta t}\sqrt{\Delta t}} \tag{1}$$

So to calculate option prices, hedge ratios, etc., according to the Black-Scholes prescription, it suffices to know σ^2, the variance. With this consideration in mind, we now restrict our analysis to the prediction of variance.

3 Setting Up the Problem of Variance Estimation

3.1 Basic Setup

We will consider the case of noisy time series, financial series being a special case. The problem is set up in the following way. Given the history of information (including the full history of values of the time series), there exists some conditional probability distribution for the next value. We label the time series variable x, then

$$f(x_t \mid I_{t-\Delta t}) = \begin{array}{l}\text{probability density function for } x_t, \\ \text{the value of the series at time t}\end{array} \tag{2}$$

where $I_{t-\Delta t}$ is the information available at time $t-\Delta t$. Usually, $I_{t-\Delta t}$ is taken as the past few values of the variable x. We will focus on the first two moments of f, $\mu(I_{t-\Delta t})$ and $\sigma(I_{t-\Delta t})$. A model consists of a "black box" that takes as input $I_{t-\Delta t}$ and outputs $\{\hat{\mu}^t, \hat{\sigma}^t\}$, predictions of the mean and variance for time t (Fig 2). A collection of models consists of a set of such pairs of functions $\{\hat{\mu}_i^t, \hat{\sigma}_i^t\}_{i=1}^M$. The index i refers to which model we are talking about. In this paper, we are not concerned with exactly what goes into the black box of Fig 2. All we know is that we are given a set of models (e.g., GARCH, Neural Networks,...) that take the input $I_{t-\Delta t}$ and provide estimates of the mean and variance as output.

3.2 Choosing Between the Models

In order to choose between the models, one requires some 'validation' data. Our goal is to predict the mean μ and the variance σ^2. Unfortunately one does not usually have access to the actual values of the mean and variance. All one usually has are data points $(\{d_\alpha\}_{\alpha=1}^n)$ drawn from the distributions $f(x_t \mid I_{t-\Delta t}^\alpha)$. Based on the data, one has to construct an error measure that evaluates the estimates $\hat{\mu}$ and $\hat{\sigma}$ without knowing the actual μ and σ. One then evaluates this error measure for each model and picks the model that

minimizes the error. The question now arises as to what kind of error measure to take.

3.3 Using Maximum Likelihood to Choose Between Models

If we know the functional dependence of $f(x_t \mid I_{t-\Delta t})$ on the parameters that we are trying to predict, we can evaluate the likelihood that the data occurred under the assumption that model i is correct. We will assume that the data have been drawn independently and that f is gaussian. In particular, Black-Scholes models satisfy our gaussian assumption on f. A similar analysis could be done for any other assumption on f. The likelihood that the data occurred under a particular model i is given by

$$l(\vec{d} \mid \text{model } i) = \prod_{\alpha=1}^{n} \frac{1}{\sqrt{2\pi\hat{\sigma}_{\alpha}^{i\,2}}} e^{-\frac{(d_\alpha - \hat{\mu}_{\alpha}^{i})^2}{2\hat{\sigma}_{\alpha}^{i\,2}}} \tag{3}$$

Maximizing the likelihood, or equivalently maximizing the Log(*likelihood*) is the criterion that we will use to differentiate between the models. As an example, consider the training of Neural Networks to predict the variance using maximum likelihood as the objective function to optimize (Weigend, 1995). In this case, the models i correspond to all the functions that the Neural Network can implement and we are choosing one of them using maximum likelihood as the criterion.

4 Analysis of the Maximum Likelihood Scheme

4.1 Expected Value of the Likelihood and Log(likelihood)

Consider the likelihood function as a function of the data d_α. Thus it is itself a random variable, for which the distribution is known given the distribution of the data point d_α. One can calculate 'the expectation of $l(d_\alpha)$'=$\langle l_i \rangle$ and 'the expectation of $\log(l(d_\alpha))$'=$\langle \log l_i \rangle$ for model i. For simplicity in notation, we will drop the α index.

$$\begin{aligned} \log\langle l_i \rangle &= -\frac{1}{2}\left[\frac{(\mu - \hat{\mu}_i)^2}{\sigma^2 + \hat{\sigma}_i^2} + \log(\sigma^2 + \hat{\sigma}_i^2) + \log 2\pi\right] \\ \langle \log l_i \rangle &= -\frac{1}{2}\left[\frac{(\mu - \hat{\mu}_i)^2 + \sigma^2}{\hat{\sigma}_i^2} + \log(\hat{\sigma}_i^2) + \log 2\pi\right] \end{aligned} \tag{4}$$

$\langle l_i \rangle$ is maximized when the predicted mean and variance are equal to the true mean and variance. So if we expect the observed value of the Log(*likelihood*) to

be close to its expected value, which will be true with enough data points, then it seems reasonable to maximize the Log(*likelihood*) in order to predict $\{\hat{\mu}, \hat{\sigma}\}$.

The expectation of the likelihood itself is not maximized at the correct values. Its maximum is when $\hat{\sigma} \to 0$. This rules out likelihood itself as a comparator because its expected behavior is not desirable. To investigate this issue further we define what it means for a model to be 'better' than another. Suppose we have two models, model 1 and model 2 with $\mid \hat{\mu}_1 - \mu \mid > \mid \hat{\mu}_2 - \mu \mid$ and $\mid \hat{\sigma}_1 - \sigma \mid > \mid \hat{\sigma}_2 - \sigma \mid$. We will then say that model 1 is worse than model 2.

If model 1 is worse than model 2 then its expected Log(*likelihood*) should also be worse. Unfortunately one can find common situations where this is not the case. This becomes evident from Fig. 3.

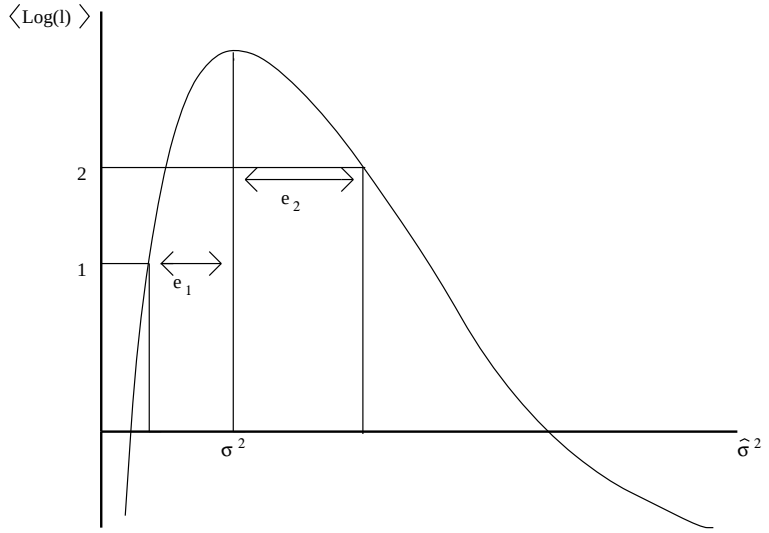

Figure 3. Plot of $\langle \log l_i \rangle$ as a function of $\hat{\sigma}_i^2$. The curve is not symmetric about the true variance. In the figure is shown how model 2 has a higher expected likelihood despite having a higher prediction error ($e_2 > e_1$)

Thus we see that even log(*likelihood*) can lead to an expected worse choice. However, note thattraining (say neural networks) by maximizing log(likelihood) (Weigend,1995) using small perturbations will select 'better' models on the

average. Note that by initially choosing a model with lower log(*likelihood*), training could be faster owing to the asymmetry of the curve about the actual variance. If the neural network cannot implement the maximum of the curve then training may stop at a worse value than is necessary, owing to this asymmetry.

4.2 Probability of Choosing the Wrong Model

Suppose that you have two models, model 1 worse than model 2 as described above. One can ask whether it is possible that model 1 will be picked more often than model 2. This is distinct from the analysis of the expected value because it is possible for the expected value to be greater for model 1 despite the fact that the probability that model 1's Log(*likelihood*) is higher is very small. In this case one might still be content because most of the time one will be choosing the better model even though its expected Log(*likelihood*) is lower. Unfortunately, we show that this too is not the case.

We ask the question: Is it possible that one might pick a 'worse' model (in the sense described in the previous section) more often than one might pick a 'better' model? To answer this question we will compare a 'worse' model with the ground truth model. We shall see that this will lead to some striking results.

Set up the problem in the following way. We have n data points $\{I_{t-\Delta t}, d_\alpha\}$ where each of the d_α have been drawn from a (possibly *different*) distribution $\sim N(\mu_\alpha, \sigma_\alpha)$. We will consider the case where $\hat{\mu}^i_\alpha = \mu_\alpha$ (perfect prediction of the mean), while the variances predicted by the models are given by $\hat{\sigma}^i_\alpha = \lambda^i_\alpha \sigma_\alpha$, $\lambda^i_\alpha \geq 0$ thus the models are parametrized by λ^i_α,$\alpha = 1, \ldots, n$. The likelihood of model i with parameters $\vec{\lambda}_i$ is

$$l(\vec{d} \mid \vec{\lambda}_i) = \prod_\alpha N_{d_\alpha}(\mu_\alpha, \lambda^i_\alpha \sigma_\alpha) \tag{5}$$

We are going to compare the two models $\{\lambda_\alpha = 1\}$, the perfect model, and $\{\lambda_\alpha\}$. We seek the probability ($P_{\vec{\lambda}}$) that $l_{\vec{\lambda}} \geq l_{\vec{1}}$ where $l_{\vec{1}}$ is the likelihood of the ground truth model and $l_{\vec{\lambda}}$ is the likelihood of model $\vec{\lambda}$.

Now treat the data outcome $\vec{d}$ as a random variable $\vec{x}$. Then this likelihood itself is a random variable, depending on the parameter $\vec{\lambda}$, so we can ask the question: what is the probability that the random variable with parameter $\vec{\lambda}$ is greater than the random variable with parameter $\vec{\lambda} = \vec{1}$, i.e., we are asking for the frequency with which the 'worse' model is chosen over the *actual* model. For the case $\vec{\lambda} = \lambda\vec{1}$, the result is shown in Appendix A. For λ slightly less than 1, P_λ is greater than $\frac{1}{2}$. This means that a worse model will be chosen over

the actual model more than $\frac{1}{2}$ the time. *This holds for any number of data points.* The smallest value that λ can be with $P_\lambda \geq \frac{1}{2}$ ($\lambda_{min}(n)$) is tabulated for various n in table 1.

n	$\lambda_{min}(n)$	$P_{max}(n)$	$\langle\frac{1}{\lambda}\rangle$=$\alpha_{correc}$
1	0.4937	0.6831	∞
2	0.7072	0.6321	1.7725
5	0.8729	0.5842	1.1894
10	0.9349	0.5594	1.0837
20	0.9670	0.5422	1.0397
50	0.9866	0.5268	1.0153
100	0.9934	0.5186	1.0076
500	0.9986	0.5088	1.0015
1000	0.9993	0.5059	1.0008

Table 1: Showing the results for the analysis of the probability that model $\vec{\lambda}$ is chosen more often than the actual model.For $\lambda_{min}(n) \leq \lambda \leq 1$, $P_\lambda \geq \frac{1}{2}$. $\langle\frac{1}{\lambda}\rangle$ is the expected correction factor α_{correc} for the new prediction given the maximum likelihood model prediction ($\hat{\sigma} \rightarrow \alpha_{correc}\hat{\sigma}$). $P_{max}(n)$ is the maximum value that P_λ can take and occurs for $\lambda \rightarrow 1^-$

Referring to table 1, we note that for n=100, a model with a 0.7% error in the prediction of σ will be chosen more often than the actual model. This is a fairly significant error when dealing with a tradable quantity.

5 Correcting for the Maximum Likelihood Prediction Error

Up to now we have diagnosed some problems with the maximum likelihood scheme. We now try to tackle the problem of compensating for this error using our understanding of how it fails. The models that are more probable than the actual model are models which underestimate the variance (assuming the mean is predicted perfectly). One expects that the model chosen using maximum likelihood will have a systematic bias for predicting a variance lower than the actual variance. Suppose that we have a model that is predicting $\hat{\sigma} = \lambda\sigma$ on the average. We would like to correct our prediction arrived at using maximum likelihood methods by multiplying our prediction by some correction factor to get a better prediction.

$$\hat{\sigma} \rightarrow \alpha_{correc}\hat{\sigma} \tag{6}$$

In order to calculate α_{correc} we need the probability distribution for obtaining a particular model with parameter λ. The method of compensating will depend on the the exact method that was used to arrive at the model. Here we will consider the case that seems appropriate to neural networks that are trained on the data using maximum likelihood. The general philosophy will become apparent.

5.1 Probability Distribution Over λ

We have the class of models that are parametrised by λ. Training proceeds in the following way. Start with a random λ and perturb λ a little toward the higher range and toward the lower range. So we are dealing with the three λ's $\{\lambda - \frac{d\lambda}{2}, \lambda, \lambda + \frac{d\lambda}{2}\}$. Now using the data we compare the likelihood of these three models and choose the model that produces the highest likelihood. We continue the process until no change in λ results. In actuality the learning proceeds not by varying λ but by varying the parameters of the model (in the case of neural networks, these are the weights). One might question why the model's λ should be the same for all the possible inputs. This will not be the case in general, but we are asking what the correction factor is *on the average*. To calculate this we look at the probability distribution that we end up with a model with a certain λ (for which the correction factor is $\frac{1}{\lambda}$).

What is the probability that we stop at the value λ? We consider the region $[\lambda, \lambda + d\lambda]$ and derive the probability density over λ, which we can use to calculate various desired properties like expectations. The detailed derivation of this probability density is lengthy. The final result is given in Appendix B, eq. 1. We are interested in $\left\langle \frac{1}{\lambda} \right\rangle$, the expected correction factor shown in Appendix B eq. 2. Values of the correction factor for various values of n are given in table 1. Note that for $n = 100$ the correction is about 0.8% which is not trivial.

We summarize the correction method, assuming that the models are predicting the mean well: Train the models using maximum likelihood and arrive at a model to be used for future predictions. Now given a new test input, obtain this model's prediction and correct by the correction factor described above.

5.2 Example: Training Neural Networks Using maximum Likelihood

In this example the model is a neural network with 131 weights that are involved in the training of the model to predict σ (Weigend, 1995). The input variable which we have called $I_{t-\Delta t}$ is $x \in [0, \pi]$. The mean $\mu(x)$ and variance

$\sigma^2(x)$ are functions of x and the model is a mapping $x \rightarrow \begin{bmatrix} \hat{\mu}(x) \\ \hat{\sigma}^2(x) \end{bmatrix}$. The network was trained on n examples by altering the weight in the direction that increased the likelihood that the data occurred under the model. The final network arrived at is used as the final model. It is to this network that we wish to apply the correction factor. A question arises as to what the effective number of data points (n_{eff}) is. Each parameter can be regarded as being trained on some of the data points. So n_{eff} should be approximately proportional to the number of examples, $n_{eff} \sim \gamma n$. Using this relationship we can get a theoretical prediction to compare with the experimental value. The results are summarized in table 2 [a]

n	$\langle\lambda\rangle$		$\langle\frac{1}{\lambda}\rangle$=$\alpha_{correc}$	
	expt.	theory	expt	theory
50	0.68±0.18	0.56±0.07	1.25±0.18	1.98±0.58
100	0.66±0.04	0.70±0.07	1.24±0.03	1.27±0.12
150	0.67±0.05	0.77±0.06	1.22±0.04	1.16±0.06
300	0.88±0.14	0.86±0.04	1.07±0.09	1.07±0.02
500	0.92±0.05	0.91±0.03	1.04±0.03	1.04±0.01

Table 2: Comparison of the theoretical results that are predicted from the induced probability distribution over λ with the experimentally obtained values for learning the variance with a neural network. The theoretical predictions were obtained by assuming a proportional relationship between n_{eff} and n, fitting γ to the data using the expected theoretical form. For $\langle\lambda\rangle$, γ was 0.006±0.002 and for$\langle\frac{1}{\lambda}\rangle$, γ was 0.044±0.014. It is interesting to compare $\frac{1}{\gamma}$ to the number of weights=N_w. $\frac{1}{N_w} = 0.008$.

The agreement between the theoretical values and the experimental values seems convincing especially as the number of data points increases. Also note that the variance in the experimental values is relatively small implying that the network has indeed settled on a model with almost a constant parameter

[a] We thank Zehra Cataltepe of the Learning Systems Group at Caltech for use of the results from her experiments in verifying the method of Weigend(1995). $\langle\alpha_{correc}\rangle$ was computed from the results only where σ was larger than a threshold because where σ is small, the behaviour is erratic. The mean as expected was learned well, so we can apply the analysis above where we assumed the mean was being predicted exactly.

of λ. How one would calculate n_{eff} for a general class of models with a given learning algorithm is not yet obvious. In our discussion we have assumed that the class of models is good enough to implement the various models with parameter λ. Exactly how we search through this space and how the models are parametrized are expected to affect n_{eff}.

6 Conclusions

It seems appropriate to summarize the path that we have followed in this paper. We started out by setting up a framework for comparing between models. In this framework, we used maximum likelihood to compare between models and we found that this leads to choosing wrong models. The results of the maximum likelihood analysis can be summarized by the picture in Fig. 4 .

	IN SAMPLE	OUT OF SAMPLE
FINITE DATA	Choosing the model that maximizes the likelihood will yield a model that systematically predicts lower variance, even if the mean is predicted well.	Probability of choosing the wrong model is $> \frac{1}{2}$ for some 'worse' models.
EXPECTED VALUE	NOT APPLICABLE	Possible to find models 1,2 with model 1 worse than 2 but $\langle \log l_1 \rangle > \langle \log l_2 \rangle$.

Figure 4: Diagram depicting what could go wrong with the Maximum likelihood scheme in the 3 possible cases.

When the mean is predicted well, we attempt to correct for the systematic underprediction of the variance by multiplying by a correction factor, α_{correc}. We find that this correction factor is economically significant even for a large number of data points. In this way we are able to choose a model from a class of models that were trained on different data, and then improve that model

using the correction factor. Unfortunately this will not work when the mean is not being predicted well. In this case, the variance will tend to be over predicted to compensate for the bias. Thus one would like to have a way to predict the variance without having to predict the mean. This is a direction open for future work as is the question of combining models that have been trained on different data. It is necessary, however, to have a "good" way to compare models before one even starts to think about combining models.

Acknowledgments

We would like to thank Dr. Amir Atiya, Joseph Sill and Zehra Cataltepe for helpful discussion.

Appendix A

Treat the data outcome $\vec{d}$ as a random variable $\vec{x}$. Then the likelihood itself is a random variable, depending on the parameter $\vec{\lambda}$. We seek the probability that this random variable with parameter $\vec{\lambda}$ is greater than the random variable with parameter $\vec{\lambda} = \vec{1}$, i.e., we are asking for the frequency with which the 'worse' model is chosen over the *actual* model. This probability is given by

$$P_{\vec{\lambda}} = \text{Prob}[l_{\vec{\lambda}} \geq l_{\vec{1}}] = \int\limits_{\{l_{\vec{\lambda}} \geq l_{\vec{1}}\}} \mathrm{d}^n \mathrm{x}\, l(\vec{x} \mid \vec{\lambda} = \vec{1}) \qquad (A-1)$$

With some manipulation one finally gets for $\vec{\lambda} = \lambda\vec{1}$ and n even

$$P_\lambda = 1 - e^{-\beta n} \sum_{k=0}^{\frac{n}{2}-1} \frac{(\beta n)^k}{k!} \qquad (A-2)$$

where $\beta = \frac{\lambda^2 \log \lambda}{\lambda^2 - 1}$ and $0 < \lambda < 1$.

Appendix B

By infinitessimally changing the λ parameter one eventually ends up at a model that maximizes the likelihood. The probability that this results in a model with parameter λ is (after lengthy calculation)

$$P(\lambda)d\lambda = \frac{n^{\frac{n}{2}} e^{-n\frac{\lambda^2}{2}}}{\Gamma(\frac{n}{2})} \left(\frac{\lambda^2}{2}\right)^{\frac{n}{2}-1} \lambda d\lambda \qquad (B-1)$$

The expected correction factor can now be calculated

$$\alpha_{correc} = \left\langle \frac{1}{\lambda} \right\rangle = \sqrt{\left(\frac{n}{2}\right)} \frac{\Gamma(\frac{n-1}{2})}{\Gamma(\frac{n}{2})} \qquad (B-2)$$

References

Black, F. and Scholes, M.S. 1973, The Pricing of Options and Corporate Liabilities. *Journal of Political Economy* **3**, 637-654.

Bollerslev, T. 1986, Generalized Autoregressive Conditional Heteroscedasticity. *Journal of Econometrics* **31**, 307-327.

Crouhy,M. and Galai,D. 1995, Hedging With a Volatility Term Structure. *The Journal of Derivatives* **Spring**, 45-52.

Engle, R.F. 1982, Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of U.K. Inflation. *Econometrica* **50**, 987-1008.

French, K.R., Schwert, G.W. and Stambaugh, R.F. 1987, Expected Stock Returns and Volatility. *Journal of Financial Economics* **19**, 3-29.

Hull, John C., *Options, Futures and other Derivative Securities* 2^{nd} *Ed.*, Englewood Cliffs, New Jersey: Prentice Hall, 1993.

Hull, J. and White, A. 1987, The Pricing of Options on Assets with Stochastic Volatilities. *Journal of Finance* **2**, 281-300.

Ito, K. 1951, On Stochastic Differential Equations. *Memoirs, American Mathematical Society* **4**, 1-51.

Kat,H.M.,'Replicating Ordinary Call Options: A Stochastic Simulation Study', Presented at the 13th AMEX Options and Derivatives Colloquium, New York (1993).

Nelson, D.B. 1991, Conditional Heteroscedasticity in Asset Returns: A New Approach. *Econometrica* **59**, 347-370.

Poterba, J. and Summers, L. 1986, The Persistence of Volatility and Stock Market Fluctuations. *American Economic Review* **76**, 1142-1151.

Schwert, G.W. 1989, Why Does Stock Market Volatility Change Over Time. *Journal of Finance* **44**, 1115-1153.

Shiller, Robert J., *Market Volatility*, Cambridge, Massachusetts: The MIT Press, 1993.

Weigend, A.S., Nix, D.A. 1995. Learning Local Error Bars for Nonlinear Regression. In *Advances in Neural Information Processing Systems (NIPS): Proceedings of NIPS'94*, Tesauro G., Touretzky D., Leen T. (eds), 489-496. Morgan Kauffman.