Theoretical Analysis For Exact Results in Stochastic Linear Learning

Rezaul Karim

LYNGBY 2004
IMM-EKS-2004-67

IMM

Theoretical Analysis For Exact Results in Stochastic Linear Learning

Rezaul Karim

LYNGBY 2004
IMM-EKS-2004-67

IMM

Abstract

This master thesis deals with the learning theory. It contains some analysis as well as derivations in Stochastic Linear learning. Derived results, for example, Generalization Error are exact in contrast with many available assumed or asymptotic results. The works are done chiefly basing on two papers: Hansen 1993 [13] and Hansen 2004 [20]. Some MATLAB simulations are done in order to prove the undoubted validity of the expressions for the Exact Generalization errors. Two expressions for the Generalization errors with respect to the sample size were derived in two different ways from linear models. The cross point of them were also detected. The properties of the curves were discussed throughout the whole sample size domain. At last, in the research part, there are some investigations about the undesired events of the curves while a discussion about the recovery from that situation is presented.

Keywords: Learning, stochastic, linear, asymptote, domain.

Preface

This master thesis works as documentation for the mandatory exam project in the requirements to achieve the M.Sc. degree from DTU, Denmark. The thesis has a workload of 35 ECTS points out of the 120 points for the two years International M.Sc. in Telecommunication program.

The thesis is done with the ”Intelligent Signal Processing (ISP) group”, IMM, DTU. It is worked out by two persons; Professor Dr. Lars Kai Hansen, acting as the supervisor and Rezaul Karim, a student of M.Sc. in Telecommunication, DTU.

Thesis Overview

This thesis document consists of mainly six chapters (chapter 0 to chapter 5) and two appendices.
Chapter 0 gives an introductory idea about learning theory with a useful idea from some probability distributions.

Chapter 1 talks about stochastic linear learning, aided with some thermodynamical and statistical mechanical concepts. The main achievement here is the exact expression for learning error, which is authenticated by MATLAB simulation.

Chapter 2 finds mainly the exact learning error in Linear Regression model that also passes the verification test through MATLAB simulation.

Chapter 3 looks for the behavior of these two learning curves (derived in chapter1 and chapter 2) and detect their cross point analytically. It also makes a mild comparison between these curves.

Chapter 4 investigates about the recovery from the undesired events of the cost function by using matrix regularization technique.

Chapter 5 makes a short conclusion of the over all thesis with a brief discussion about the merits and demerits of this thesis.

APPENDIX A is used for all calculations and proofs. Especially A.8 and A.9 gives the analytic explanation with large calculations for the pole and singularity of the test error function (which is found from chapter 1)

APPENDIX B deals about the terms and glossary.

At last, the Reference shows the sources used for this thesis.

Acknowledgements

Indubitably, a hefty amount of thanks goes to Professor Dr. Lars Kai Hansen for bringing such kind of abstract and analytical thesis inside my relaxing range due to his professionalism and dedicated efforts to motivate a student.
Bounties of thanks go to teachers Dr. Jan Larsen and Dr. Ole Winther for answering my several questions with inspiring actions. The same flows towards Post Doc Jesper (explaining statistical mechanics), Ph.D students Rasmus Olssen (friendly guidance and technical discussions) and Kaare (technical discussions). Ph.D students Anders, Peter, Michael, Daniel are pretty acknowledged for study help.
Thanks to Post Doc Finn and Sigurdur for their continuous politeness.
Ph.D students Rasmus Elsborg, Mads, Tue, Morten, Niels and others are appreciated for contribution to the existing nice environment at the ISP cluster, IMM where I spent my thesis time.
Fellows Sliman, Ling and Crillis are thanked for helpful knowledge sharing whereas Frederik must be thanked at least for his friendly manner. Martin Rune, Martin Vester and Denis are admitted for room sharing with nice communications.
Special thanks to Mr. Mogens Dyrdal (computer support) and Ms.Ulla Nørhave (official support).

Lyngby, August 31, 2004

Rezaul Karim, s030286

Table of contents

Chapter 0: Introduction

In this chapter we give some introductory idea about the issue that the project deals with.

0.0 Theoretical?

Although there is a huge improvement in the implementation of statistical meditations and models in almost all the fields of science and engineering, the intricacy of developing a satisfactory model depending on the information provided by a finite number of observations is not fully acknowledged. Absolutely, the theme of statistical model fabrication or recognition is greatly dependent on the results of the *theoretical analyses* of the subject under observation. Still, it must be considered that there is usually a big gap between the theoretical results and the practical recognition process. However, a good theory can overcome this problem efficiently. Therefore,

"Nothing is more practical than a good theory"
-------- *Vladimir Vapnik (Russian statistician).*

And here,

"I believe in Vapnik as r *believes in* v *so that* $\delta_r^v = 1$ *"*
-------- *Author.*

0.1 Stochastic linear learning

A process is said to be *stochastic* if it represents a time dependent statistical phenomenon according to the probabilistic laws. Usually, this time dependence is defined on some observation intervals. The statistical property of the phenomenon implies that it progresses with time in an inexact predictable manner from the observer's viewpoint.
This phenomenon can be a radar signal, a sequence of real-valued measurements of voltage, a sequence of coin tosses, digital computer data, the output from a communication channel, noise, etc. The inexactness in the prediction occurs due to some undesired effects such as interference or noise in a communication link or storage medium, etc.
By using the stochastic process theory, it is possible to enumerate the above impression and build mathematical models of real phenomena. The models involving only linear terms of the stochastic (input and output) variables are termed as *linear model* (with respect to those variables).
A *stochastic linear model* contains input and output variables that are connected by a number of adjustable parameters. The process of determining the values of these parameters on the basis of data set is called *stochastic linear learning*. In this process, *learning* is acknowledged when the relation between the input and output variables are changed in such a way as to reduce an error function surface. The result of the

stochastic linear learning is the set of adjustable parameters in the linear model. These parameters are also stochastic and depend on the specific, random training set.

0.2 Learning, generalization and the load parameter

In linear modeling, the main job is to fabricate a linear map between the stochastic input and output variables, which can recover the stochasticity. These variables may not come only from a specific set of examples, instead, from a lot of other new and unknown sets of data. Therefore, the vital goal in learning is not to memorize the specific training data, but rather to model the essential generator of the data. This type of model can make the best possible predictions for the output variables after it is trained and presented with a new value for the input variable.
As we have said before, the stochastic linear model contains a number of adjustable parameters that relate between the input and output variables. But often there is also a significant issue about their number; how many adjustable parameters should be in a stochastic linear model? This number is called the model dimension (or coefficient number). A model with too small input dimension is too little flexible. Therefore, it makes poor predictions in case of new data; poor *generalization* performance. On the contrary, a model with too large input dimension also makes paltry predictions in case of new data as it is too flexible and thus fits too much of the noise from the training data. Hence, it has a low *generalization* performance. Therefore, it is necessary to optimize the model dimension (or model complexity) in order to achieve the best generalization.
But how to notice whether any specific model dimension value is large or small? A feasible way to answer this question is to use the (training) sample size and compare it with the model dimension; for example, finding their ratio. Afterward, analyze the cost (error) function with respect to this ratio. The ratio between the model dimension and the training sample size is traded as *load parameter* or simply the number of examples per dimension. Mathematically,

$$load\ parameter = \frac{Sample\quad size}{Model\quad \dim ension}$$

In many literatures, it is denoted by α.

0.3 Example of stochastic output

Consider two distributions; the very well known Gaussian distribution and the Cauchy distribution.
A simple Gaussian distribution is given by:

$$p(x) = \frac{1}{\sqrt{2\pi}}\exp(-\frac{x^2}{2});\qquad x \in (-\infty, \infty)$$

And the Cauchy distribution is given by:

$$p(x) = \frac{1}{\pi} \frac{1}{1+x^2} \qquad ; \; x \in (-\infty, \infty)$$

They look like below:

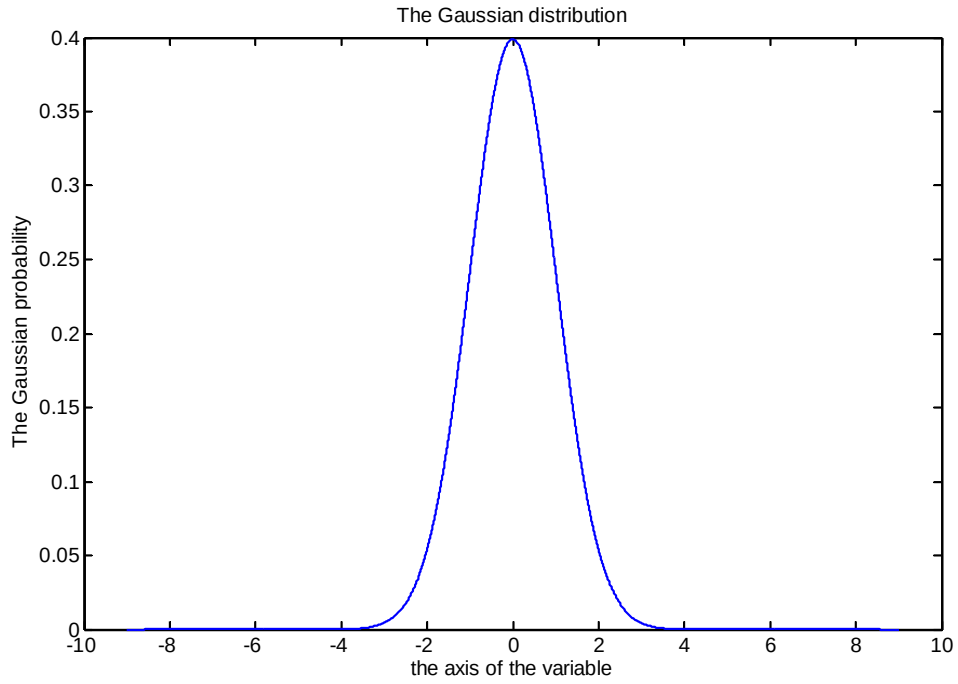

Figure 0.1: Simple Gaussian distribution (having zero mean and unit variance). It is a symmetric distribution.

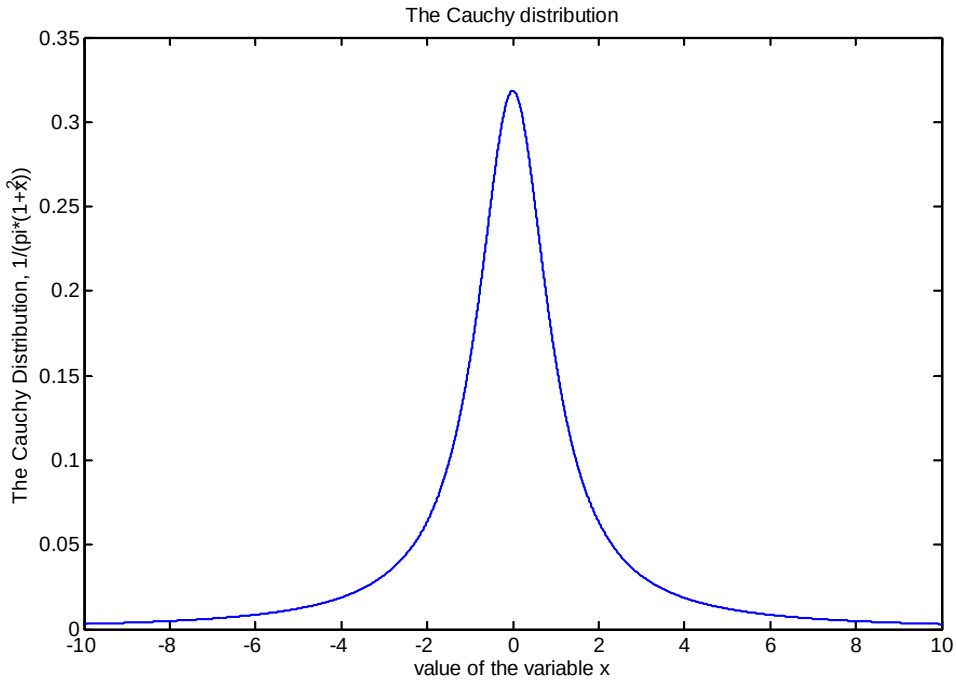

Figure 0.2: The Cauchy distribution. It is also a symmetric distribution like the Gaussian distribution.

The similarity between these two distributions is that both of them are symmetric. But if we look for their mean values, we see that they have opposite behavior; the Gaussian has a bounded mean whereas the Cauchy has an unbounded mean! This can be seen from the figures below:

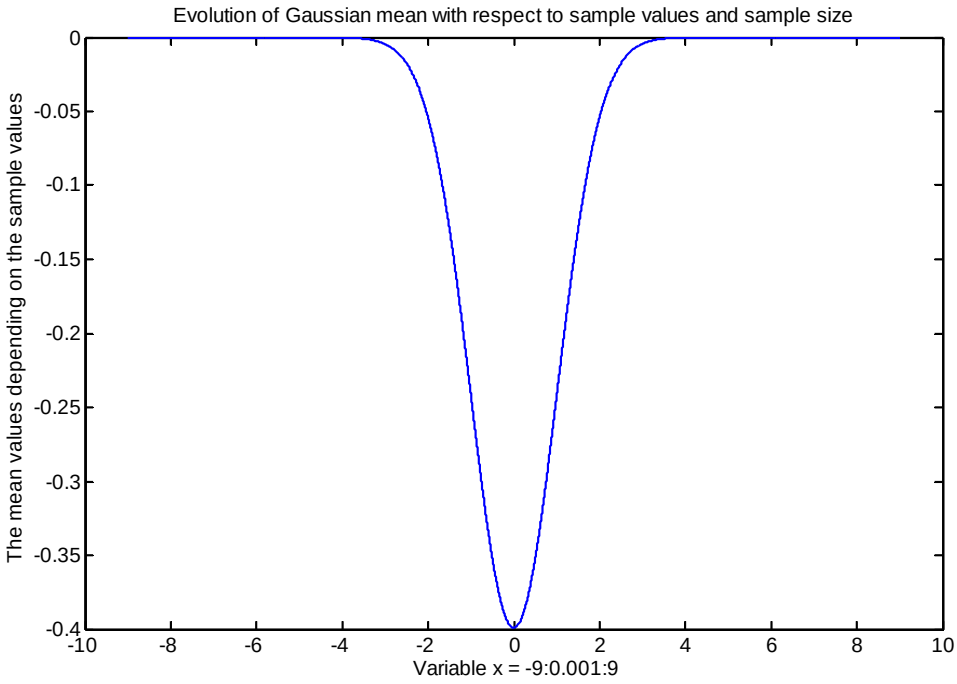

Figure 0.3: Evolution of the mean values of the simple Gaussian distribution. From the figure it is obvious that the mean value is bounded in contrast to the mean of the Cauchy distribution (Figure 0.4, below).

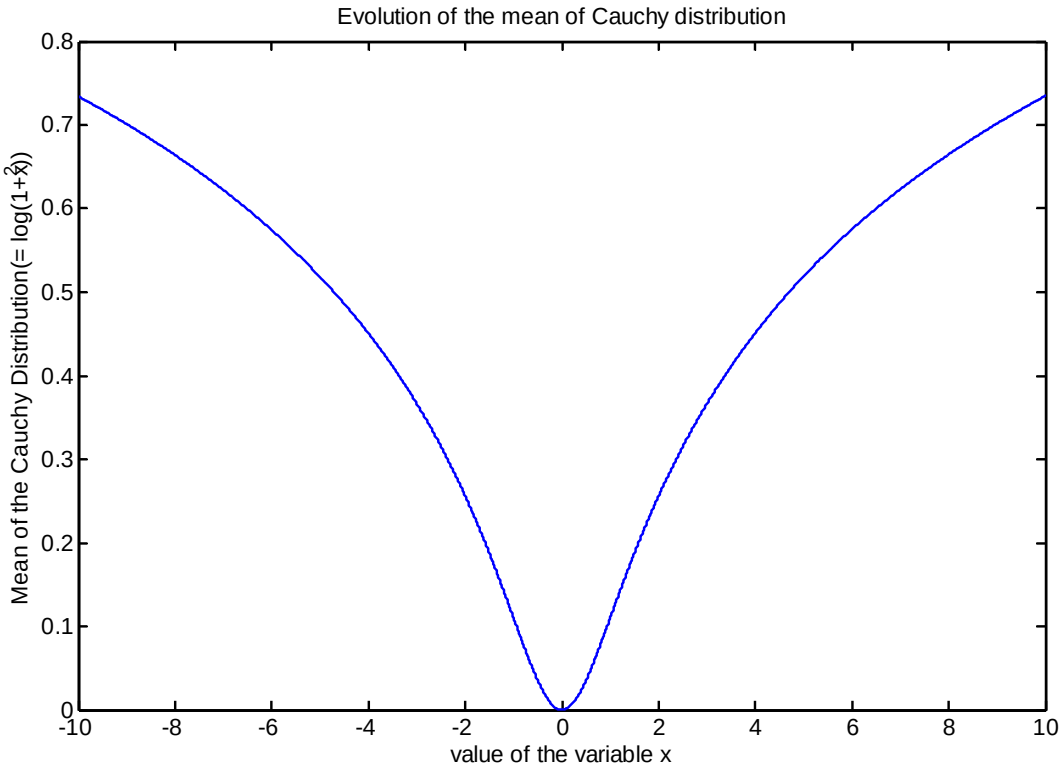

Figure 0.4: Evolution of the mean values of the Cauchy distribution. From the figure it is obvious that the mean value is unbounded in contrast to the mean of the Gaussian distribution (Figure 0.3).

(An explicit and independent explanation showing the reason behind the boundness of Gaussian mean and the unboundness of the Cauchy mean is given in APPENDIX A, A.7).
Thus, from this example, we can see that in some cases of the stochastic process, we can meet the situation that the sample values are finite whereas their mean value is infinite (example from the Cauchy distribution)! We keep this idea in our mind to proceed with the rest part as we are dealing with a stochastic process (the *stochastic linear learning*)!

0.4 Presented works related to learning and generalization

So far, the only standard results on test error estimation are given by Akaike [7] [8] that are valid for the large training sets (asymptotic) and basically linear models [9]. Recently, there have been a lot of concentrations on simple linear learning schemes [12] [2] [25] [11]. The benefit with simple learning schemes is that it can be analyzed analytically and the obtained results may either be applied directly to the more interesting non-linear models, such as feed forward neural networks with hidden representations, or may inspire future analysis of such models. Recent successful schemes for optimization of neural network architecture are driven basically from local linear approximations [26]. Fogel used Akaike's Information Criterion for comparing neural network architectures [27]. Hoffman and Larsen used Akaike's final Prediction Error Estimate (FPE) for optimal reduction of polynomial models [28]. These informations are aided by [13].

Krogh worked with the learning process in the limit of large models and large training sets introducing the load parameter and found a phase transition while the load parameter tends to one [11].
Hansen [13] [20] found the learning errors in exact forms that extend the above results (for linear model) for all valid range; i.e. also for deterministic case. He found a divergence in the test error like [11] but in the exact form [13].

0.5 Problem definition

We have seen that the conventional results in the stochastic linear learning are not exact; they are valid with some limitations. Our target is to derive the exact expressions for the vital terms in stochastic linear learning with consistent analyses. The most vital terms can be the average test and training errors for a stochastic linear model. Therefore, our main goal concerns with these terms. So far, there are only two available texts [13] [20] concerning the exact results of these terms. Thus, there is a good chance to compare our results with these texts. Further studies on the basis of the obtained results would be highly appreciated regarding the time duration of the project.

Chapter 1: Exact Test and Training Error Averages in stochastic Linear Learning

In this chapter, we study the statistical properties of stochastic linear learning. We derive the expressions for exact test and training errors for a linear model and a finite training set. These expressions are then compared with the results in [13]. Our result overcomes the limitation in Akaike's FPE as it is valid for more general case including the stochasticity of the process in contrast with Akaike's performance that is only for the special (asymptotic) case. We also make a vigilant MATLAB simulation that authenticates our theory.

The chapter is organized in the following way:
In 1.1, we formulate the model; while 1.2 discusses the post training distribution of parameters following a stochastic learning procedure. In 1.3, we compute analytically the estimates for test and training errors and these results are then compared with the others' in 1.4. Thus, the chapter is concluded in 1.5.

1.1 Linear Modeling

Consider a linear (with respect to x) model with d inputs x_j , $j = 1,2,...d$ and a single valued output

$$y(x) = \sum_{j=1}^{d} w_j x_j \tag{1.1}$$

The model parameters w_j will be estimated by the standard recursive gradient descent procedure.
The database or *training set* that we will use for estimating the optimal parameters is created by an unreliable teacher (i.e, static noise is included) having a set of weights w_j^* [11]:

$$y^{*\alpha} = \sum_{j=1}^{d} w_j^* x_j^\alpha + \nu^\alpha; \quad \alpha = 1,2,.....,N \tag{1.2}$$

(α works as a suffix representing the samples; not as a power!)
We consider a finite training set of N examples. The inputs to the teacher, $\mathbf{x}_j^\alpha$ and the noise, ν^α are independent (like as most of the models in practice) normally distributed: $\mathbf{x} \in N(0, \mathbf{\Sigma})$, $\nu \in N(0, \sigma_\nu^2)$. It is a critical point in the basic opinions leading to our achievements (decisions). The input covariance matrix $\sum = \left(\sum_{jj'}\right)$ that we will be working with is non-singular.

1.2 The Post Training Distribution

1.2.1The distribution

We try to reduce the distance between y^{α} and $y^{*\alpha}$. An optimal parameter set, w will minimize the additive distance for all α examples. We find this optimal parameter by training our model that leads to a distribution on the network configuration space. This distribution is noted as post training distribution.
We will use this distribution for computing average properties of an ensemble of networks; this will lead us to model the generic behavior of a network following a stochastic learning procedure. The distribution function of the ensemble reflects the learning procedure. Properties of *deterministic*[1] learning procedures can be obtained as a limit of the general results.

Levin, Tishby and Solla have presented general argument in favor of a *Gibbs distribution*[2] of weights [2]:

$$P_N(w) = Z_N^{-1} \exp\left(-\beta \sum_{\alpha=1}^{N} E^{\alpha}(w) \right) \tag{1.3}$$

where the error (or the distance) on the α'th example is given by:

$$E^{\alpha}(w) = \left(y(x^{\alpha}) - y^{*\alpha} \right)^2 = \left(\sum_{j=1}^{d} (w_j - w_j^*) x_j^{\alpha} - \nu^{\alpha} \right)^2 \tag{1.4}$$

and Z_N is the normalized integral, given by

$$Z_N = \int Dw \ \exp\left(-\beta \left(\sum_{\alpha=1}^{N} E^{\alpha}(w) \right) \right) \ [\text{With } Dw = \prod_{j=1}^{d} \int dw_j \,] \tag{*}$$

and β is a positive integration parameter that determines the sensitivity of the probability $P_N(w)$ to the error value $E^{\alpha}(w)$.
Relation (1.3), with the help of (1.4) and (*) can be regarded as the *post training* distribution in the weight space, the probability of each point w is reduced exponentially with the error of the network on the training set. That means, when for any w, this error is less, that w has higher probability and the w, for which this error is bigger, has a comparatively lower probability.
Here, we will say something about β and Z_N.

---

[1] Deterministic learning is the one where we are able to find an exact valid value of the quantity to be evaluated. For example, in equation (1.3), having a fixed β, when the error (sum) is the minimum, the probability for a certain w is the biggest and determinate (possible to be defined and measured).

[2] Something is said about Gibbs distribution in APPENDIX B.

1.2.2 The β factor

It is also called the effective inverse temperature as it is inversely related to the temperature, came from thermodynamics in the following way:

$$\beta = \frac{1}{k_B T}$$

given k_B is the Boltzman's constant and T is the temperature (non-negative).
In that case, $E^{\alpha}(w)$ (or their sum) in relation (1.3) or (1.4) can be treated as energy (or total energy) depending (mainly) on the parameter, w. This $E^{\alpha}(w)$ has the inverse dimension (or, unit) of β, which leads to a dimensionless quantity under the exponent in the R.H.S of relation (1.3) and thus makes this relation mathematically valid.
The idea of introducing β in our calculation comes from the feeling that the temperature T could be treated as a similar quantity like noise in our weight space; more temperature is equivalent to more noise (in the weight space) consistently, the less β value. Inverse of this phenomenon also holds. We will talk a bit about this with a short example at the end of this article when we discuss about its influence.
From [4], we have the effective temperature β^{-1} is given by

$$\beta^{-1} = \eta N^{-1} \sum_{\alpha=1}^{N} \left(F^{\alpha} - F_0\right)^2 \qquad (1.5)$$

with $F^{\alpha} = -\frac{\partial E^{\alpha}}{\partial w}$, $F_0 = N^{-1}\sum_{\alpha=1}^{N} F^{\alpha}$ and η is the step-size parameter as will be given in the relation (1.6) later.
We will now make a bit algebraic manipulation with (1.5) to find the influence of β in the weight space. Re-writing (1.5) we get

$$\begin{aligned}
\beta^{-1} &= \eta N^{-1} \sum_{\alpha=1}^{N} \left(F^{\alpha} - F_0\right)^2 \\
&= \eta \langle \left(F^{\alpha} - F_0\right)^2 \rangle \\
&= \eta \langle \left(F^{\alpha} - N^{-1}\sum_{\alpha=1}^{N} F^{\alpha}\right)^2 \rangle \\
&= \eta \langle \left(F^{\alpha} - \langle F^{\alpha} \rangle\right)^2 \rangle \\
&= \eta \left[\langle (F^{\alpha})^2 \rangle - (\langle F^{\alpha} \rangle)^2\right] \\
&= \eta \ \eta^{-2} \left[\langle (-\eta F^{\alpha})^2 \rangle - (\langle -\eta F^{\alpha} \rangle)^2\right] \\
&= \eta^{-1} \left[\langle (-\eta \frac{\partial E^{\alpha}}{\partial w})^2 \rangle - (\langle -\eta \frac{\partial E^{\alpha}}{\partial w} \rangle)^2\right] \\
&= \eta^{-1} \left[\langle (\delta w(\eta))^2 \rangle - (\langle \delta w(\eta) \rangle)^2\right] \text{ [using the idea in (1.6)]} \\
&= \eta^{-1} \, Variance(\delta w(\eta)) \\
\Rightarrow \ \beta^{-1} &= \eta^{-1} \, Variance(\textit{change in w parameter})
\end{aligned}$$

This shows that β is inversely proportional to the variance of the change in w parameter (which is also a function of η, chosen by the user); consistently temperature is directly proportional to this variance (or, the variance of the change) in w space, which makes sense.
Now, when $\beta \to \infty$ (or, equivalently, $T \to 0$) we find that there is no variance in the w space (or, no total change in the w parameter). In this case, the measurement of the probability in relation (1.3) is simply the previous distribution restricted to the zero error region in w space. We can express it mathematically in the following way,

$$P_N(w) = Z_N^{-1}\,\delta\left(E^{\alpha}(w)\right);$$

Where $\delta\left(E^{\alpha}(w)\right) = 1$ for $E^{\alpha}(w) = 0$ and 0 otherwise.
So, in a short, we say that in case of $\beta \to \infty$, we get no noise in the distribution of w and this distribution becomes a delta function.
But when $\beta \to 0$ (or, equivalently, temperature, $T \to \infty$), we have infinite variation in the w space. That means, w space is full of noise, every possible states of the weight vector in w space has the same probability, no useful information is available; this is worthless and undesired (but in some optimization algorithm, we start with higher T value and later we cool it down in order to get a better result, which is not so related here and may not be mixed with our discussion here).
In this chapter, first we will keep β to be non-zero finite in order to perform some formal derivations of the training and test errors.

1.2.3 The Z_N function

Sometimes this is called the partition function or the normalization constant. This gives the guaranty from the relation (1.3) that sum of all the probabilities will be equal to one.
It is also an error moment generating function[3] as by manipulating with it, we can obtain error functions (as we will see later of this chapter).
Although it is called a constant, it is not always a constant indeed (but it is a constant with respect to w); depends on some other variables. Temperature T is one of those variables.
We can verify its monotone property related to the error function and sample size in the following way:

We know that

$$\sum_{\alpha=1}^{N} E^{\alpha}(w) \;\le\; \sum_{\alpha=1}^{N+1} E^{\alpha}(w) \quad [\text{as } E^{\alpha}(w) \ge 0\ ]$$

$$\Rightarrow\; -\beta\sum_{\alpha=1}^{N} E^{\alpha}(w) \;\ge\; -\beta\sum_{\alpha=1}^{N+1} E^{\alpha}(w) \quad [\text{as } \beta \text{ is positive}]$$

$$\Rightarrow\; \exp\left(-\beta\sum_{\alpha=1}^{N} E^{\alpha}(w)\right) \;\ge\; \exp\left(-\beta\sum_{\alpha=1}^{N+1} E^{\alpha}(w)\right)$$

$$\Rightarrow\; \int \exp\left(-\beta\sum_{\alpha=1}^{N} E^{\alpha}(w)\right) \mathrm{D}w \;\ge\; \int \exp\left(-\beta\sum_{\alpha=1}^{N+1} E^{\alpha}(w)\right) \mathrm{D}w$$

[3] Something more about Moment generating function will be said later in this chapter.

$$\Rightarrow \quad Z_N \quad \geq \quad Z_{N+1} \quad \text{[Using the relation (*) above].}$$

Then this (the normalizing integral) function is a Semi-monotone decreasing function (considering the sample size).

Now, we get back to the relation (1.3) & (1.4). As we said before, we will be looking for an optimal parameter set, w. For that searching (and learning; as here, learning is equivalent to the reduction of the cost function), we will use the standard recursive gradient descent[4] method with step –size parameter η:

$$\delta w_j^n \equiv w_j^{n+1} - w_j^n = -\eta\left(\frac{\partial E^{\alpha(n)}}{\partial w_j}\right) \tag{1.6}$$

that is, the weights are updated after each presented example. The presentations are from a random sequence (α (n)) drawn from the training set. It is done in this way as we will be working with the mean error.

Using [4], we arrive at the idea that as the distribution of weights in our system follows a stochastic learning procedure, it solves a Fokker-Planck equation[5]. Particularly, when we consider our covariance matrix of the input vector, $\sum_{jj'} = \sigma_x^2 \delta_{jj'}$ (isotropic covariance matrix) and the step size parameter (η) to be small (slow training) we get the stationary weight distribution is approximately Gibb's distribution as above (relation (1.3)). Throughout the rest of this chapter, we will continue with these considerations in order to assume our post training distribution to be Gibbsian. Our considerations will support our assumption to skip the limitations of [2] proved by [4].

[4] A standard technique in optimization problem. Sometimes, known as steepest descent. Idea behind this is mainly similar to the Bisection method (or, Newton-Raphson method) that is used to find the zero of a function. In gradient descent, (primarily) we also try to find the zero of the derivative of the cost function (here, E) with respect to the parameter (here, w). If the cost function's search region is convex, then for its positive gradient value we give the parameter a negative increment and for its negative gradient value we give the parameter a positive increment (or, for a concave regional cost function we do the opposite) in order hit the extremum of the cost function.

[5] Very shortly: A partial differential equation, mainly came from statistical mechanics; where the dependent variable is the probability of a state (with respect to particles) and the independent variable is time. That means, the Fokker-Planck equation talks about the rate of change of the probability densities of the states with respect to time. Here, it is compared by taking the probability density over the w space. Each set of w corresponds a state and time could be considered as the iteration.

1.3 Average Test and Training Errors

1.3.1 Training Error Average involving the Energy function and distribution

As the errors are functions of weights (coefficients of input data), we introduce their distribution in order to find the weighted average of the errors.
When a specific database is given, by using the post training distribution we can compute the *ensemble average* (or, group average having the same temperature, T or noise level **)** of the training error. We calculate it in the following way:

At first, we find the error on any α'th example (from the database), $E^{\alpha}(w)$ in any network of our model by using relation (1.4). Using $\tilde{w}_j = w_j - w_j^*$ in (1.4), we get $E^{\alpha}(w) = \left(\sum_{j=1}^{d} \tilde{w}_j x_j^{\alpha} - \nu \right)^2 = E^{\alpha}(\tilde{w})$. [here, we have introduced $E^{\alpha}(\tilde{w})$, which is just for notation only. It is for our convenience and throughout the rest of this chapter, we will use this.]. Then we add these errors, $E^{\alpha}(\tilde{w})$ for all given α ($\alpha = 1,2,.....,N$) in that network and divide this addition by the total number (N) of examples; this gives an average-error of this particular network. Then we consider infinite number of networks (as a result, we also have infinite numbers of weights, which leads $\tilde{w}$ to be continuous for any summation) and take a weighted average of the average-errors for all of the (considered) infinite networks being dependent on w (or, $\tilde{w}$). The weighted average is found by taking integration over the multiplication of the average-error per network, $\frac{1}{N}\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})$ and the post training distribution, $P_N(\tilde{w})$ (as it describes the probability of each weight on the whole weight space) with respect to $\tilde{w}$. This gives us at last

the *ensemble average* of the training error, $\in_T = \int D\tilde{w}\, P_N(\tilde{w}) \frac{1}{N}\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})$ (1.7)

with the notation: $\int Dw = \int D\left(w - w^*\right) = \int D\tilde{w} \equiv \prod_{j=1}^{d} \int d\tilde{w}_j$

1.3.2 Test Error Average involving the Energy function and distribution

The test or *generalization* error is computed as the joint average over the post training ensemble and a random example drawn from the same distribution as the training set (test samples are a part of the training sets).
The computation is done by using the following idea:

The test error in each network (due to its model and parameters) can be found by comparing relations (1.1) and (1.2)

$$E(w) = \left(y(x) - y^*\right)^2 = \left(\sum_{j=1}^{d}(w_j - w_j^*)x_j - \nu\right)^2 = \left(\sum_{j=1}^{d}\widetilde{w}_j x_j - \nu\right)^2 \quad .$$

By using infinite number of networks and the same post training distribution as above (as it describes the probability of each point $\widetilde{w}$ over the weight space), $P_N(\widetilde{w})$, we find the test error, weighted by the probabilities of the points (weight co-efficients of networks) $\widetilde{w}$ as $\int D\widetilde{w} P_N(\widetilde{w}) \left(\sum_{j=1}^{d}\widetilde{w}_j x_j - \nu\right)^2$. This can be treated as a partial average test error.

In order to find a full average test error, we now start to take the samples randomly from the training set pretending them as test samples and find the test errors. But these taken samples came from the distribution as for data $\mathbf{x} \in N(0, \boldsymbol{\Sigma})$ and for noise $\nu \in N(0, \sigma_\nu^2)$ that we mentioned at the beginning of this chapter. As soon as we insert them in our experiment, we need to introduce the probability densities of their components as they are taken randomly. That gives us the average test or generalization error in the following way:

$$\in_G = \left(\int D\widetilde{w} P_N(\widetilde{w})\right)\left(\int Dx P_\Sigma(x)\right)\left(\int d\nu P_{\sigma_\nu^2}(\nu)\right)\left(\sum_{j=1}^{d}\widetilde{w}_j x_j - \nu\right)^2 ; \text{ Where } Dx = \prod_{j=1}^{d}\int dx_j .$$

But since $\mathbf{x}$ and ν are independent, for our convenience, we write the above relation in the following form.

$$\in_G = \int D\widetilde{w} P_N(\widetilde{w}) \int Dx \int d\nu P_\Sigma(x) P_{\sigma_\nu^2}(\nu)\left(\sum_{j=1}^{d}\widetilde{w}_j x_j - \nu\right)^2 \qquad (1.8)$$

$$\Rightarrow \in_G = \int D\widetilde{w} P_N(\widetilde{w}) \iint \left(\sum_{j=1}^{d}\widetilde{w}_j x_j - \nu\right)^2 P_\Sigma(x) P_{\sigma_\nu^2}(\nu)\, Dx\, d\nu$$

$$\Rightarrow \in_G = \int D\widetilde{w} P_N(\widetilde{w})\, E\left[\left(\sum_{j=1}^{d}\widetilde{w}_j x_j - \nu\right)^2\right] \qquad [\text{Since}, E[m] = \int m\, p(m) dm\,]$$

$$\Rightarrow \in_G = \int D\widetilde{w} P_N(\widetilde{w})\, E\left[\sum_{j,j'=1}^{d}\widetilde{w}_j \widetilde{w}_{j'} x_j x_{j'} - 2\sum_{j=1}^{d}\widetilde{w}_j x_j \nu + \nu^2\right]$$

$$\Rightarrow \in_G = \int D\widetilde{w} P_N(\widetilde{w}) \left(E\left[\sum_{j,j'=1}^{d}\widetilde{w}_j \widetilde{w}_{j'} x_j x_{j'}\right] - 2E\left[\sum_{j=1}^{d}\widetilde{w}_j x_j \nu\right] + E\left[\nu^2\right]\right)$$

$$\Rightarrow \in_G = \int D\widetilde{w} P_N(\widetilde{w}) \left(\sum_{j,j'=1}^{d}\widetilde{w}_j \widetilde{w}_{j'} E\left[x_j x_{j'}\right] - 2\sum_{j=1}^{d}\widetilde{w}_j E\left[x_j\right]E[\nu] + \sigma_\nu^2\right)$$

$$\Rightarrow \in_G = \int D\widetilde{w} P_N(\widetilde{w}) \left(\sum_{j,j'}^{d}\widetilde{w}_j \widetilde{w}_{j'} \Sigma_{jj'} - 0 + \sigma_\nu^2\right)$$

[Using $E\left[x_j x_{j'}\right] = \Sigma_{jj'}$ from Art 1.1]

$$\Rightarrow \in_G = \int D\tilde{w} P_N(\tilde{w}) \left(\sum_{j,j'}^{d} \sum_{jj'} \tilde{w}_j \tilde{w}_{j'} + \sigma_v^2 \right). \tag{1.9}$$

1.3.3 The Moment generating functional

In case of post training distribution at section 1.2, in relation (1.3), we found a term Z_N as the normalization constant integral. We also defined it there mathematically by the relation (*). We see that this Z_N is a function of the error function, $E^{\alpha}(w)$ which is involved inside the exponential term under the integral. So, we can call Z_N as a *functional* of w. In addition, we notice that by differentiating Z_N with respect to the co-efficient of $E^{\alpha}(w)$ (here, β), it is possible to generate the moments of $E^{\alpha}(w)$ (or, $E^{\alpha}(\tilde{w})$). As a result, we can call Z_N as a *moment generating function* of $E^{\alpha}(w)$ (or, $E^{\alpha}(\tilde{w})$). In the same way, in order to get the moments (or products) of $\tilde{w}$ (or, w) we introduce an extra term involving $\tilde{w}$ under the exponent part of Z_N. This term is expressed as a linear combination of $\tilde{w}$ and h; where h is an auxiliary field that is useful in our calculation in the limit $h \to 0$ (that is, applying this limit we keep the validity (and originality) of our expressions and quantities). This extra term is multiplied with $-\beta$ and then added with $-\beta E^{\alpha}(w)$ under the exponent term of Z_N. Thus Z_N is having a new phase as

$$Z_N(h,\beta) = \int D\tilde{w} \exp\left(-\beta \left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j \right) \right) \tag{1.10}$$

[Here, one might ask about the dimensionality and the dependence of $\sum_{j=1}^{d} h_j \tilde{w}_j$ with respect to β for the consistency of (1.10) and the further manipulations with it. In that case, we inform that $\sum_{j=1}^{d} h_j \tilde{w}_j$ has the inverse dimensionality of β and $h_j \neq h_j(\beta)$ directly up to our uses level. We may not give detail explanations for this since it is beyond of interest.]

Now, from the relation (1.10), it is quite obvious that by manipulating with the derivatives of $Z_N(h,\beta)$, we can obtain the moments of the average training and generalization errors. So, it is a *moment generator*. In addition, $Z_N(h,\beta)$ is a function of function(s). So, it is a *functional*. Combining these two, we call $Z_N(h,\beta)$ as the *moment generating functional* that will be continued throughout the rest discussions.

1.3.4 Training Error Average involving the Moment Generating Functional

By making a careful observation (with comparison) on the relations (1.7) and (1.10) and applying the idea form the above discussions about $Z_N(h,\beta)$, it is noticeable that the average training error $\in_T$ can be expressed as below [13]:

$$\in_T = -\frac{1}{N}\frac{\partial}{\partial\beta}\left[\ln Z_N(h,\beta)\right]_{h=0} \tag{1.11}$$

(This $\in_T$ can be treated as the average training error with respect to samples)
(Proof of this (1.11) is given APPENDIX A: A.1)

1.3.5 Test Error Average involving the Moment Generating Functional

By making a powerful observation and investigation (with deep comparison) on the relations (1.9) and (1.10) and using the knowledge form the above discussions about $Z_N(h,\beta)$ it is also possible to detect that $\in_G$ can be expressed as below [13]:

$$\in_G = \frac{1}{\beta^2}\sum_{j,j'=1}^{d}\sum_{jj'}\left[\frac{\partial^2}{\partial h_j \partial h_{j'}}\ln Z_N(h,\beta) + \left(\frac{\partial}{\partial h_j}\ln Z_N(h,\beta)\right)\left(\frac{\partial}{\partial h_{j'}}\ln Z_N(h,\beta)\right)\right]_{h=0} + \sigma_\nu^2 \tag{1.12}$$

(This $\in_G$ can be treated as the average test error with respect to test samples and noise)

(Proof of this (1.12) is given APPENDIX A: A.2)

1.3.6 Explicit Evaluation of the Moment generating functional

Now we will try to evaluate the generating functional $Z_N(h,\beta)$ in a more explicit form. From relation (1.10), we have,

$$Z_N(h,\beta) = \int D\widetilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N}E^{\alpha}(\widetilde{w}) + \sum_{j=1}^{d}h_j\widetilde{w}_j\right)\right)$$

$$\Rightarrow Z_N(h,\beta) = \int D\widetilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N}\left(\sum_{j=1}^{d}\widetilde{w}_j x_j^{\alpha} - \nu\right)^2 + \sum_{j=1}^{d}h_j\widetilde{w}_j\right)\right)$$

$$\Rightarrow Z_N(h,\beta) = \int D\widetilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N}\left(\sum_{j,j'=1}^{d}\widetilde{w}_j\widetilde{w}_{j'}x_j^{\alpha}x_{j'}^{\alpha} - 2\sum_{j=1}^{d}\widetilde{w}_j x_j^{\alpha}\nu^{\alpha} + \left(\nu^{\alpha}\right)^2\right) + \sum_{j=1}^{d}h_j\widetilde{w}_j\right)\right)$$

If we define,

$$A_{jj'} = \sum_{\alpha=1}^{N} x_j^{\alpha} x_{j'}^{\alpha} \qquad \text{[Estimated input covariance matrix]} \tag{1.13}$$

$$a_j = h_j - 2\sum_{\alpha=1}^{N} x_j^{\alpha} \nu^{\alpha} \tag{1.14}$$

$$a_0 = \sum_{\alpha=1}^{N} \left(\nu^{\alpha}\right)^2 \tag{1.15}$$

We get

$$\ln Z_N(h,\beta) = -\frac{1}{2}\ln\left(\det(\beta\mathbf{A})\pi^{-N}\right) + \frac{\beta}{4}\sum_{j,j'}^{d} a_j a_{j'} \left(\mathbf{A}^{-1}\right)_{jj'} - \beta a_0 \tag{1.16}$$

[Proof of this relation is given in APPENDIX A.4]

1.3.7 Evaluating Training Error Average using the Moment Generating Functional

Now, we will apply (1.11) on (1.16) in order to obtain the average training error as we have mentioned earlier.

Applying (1.11) on (1.16) we get the average training error ($\in_T$) in the way below:

$$\in_T = -\frac{1}{N}\frac{\partial}{\partial\beta}\left[-\frac{1}{2}\ln\left(\det(\beta\mathbf{A})\pi^{-N}\right) + \frac{\beta}{4}\sum_{j,j'}^{d} a_j a_{j'}\left(\mathbf{A}^{-1}\right)_{jj'} - \beta a_0\right]_{h=0}$$

$\Rightarrow$

$$\in_T = -\frac{1}{N}\frac{\partial}{\partial\beta}\left[-\frac{1}{2}\ln\pi^{-d} - \frac{1}{2}\ln\left((\beta\lambda_1)(\beta\lambda_2).....(\beta\lambda_d)\right) + \frac{\beta}{4}\sum_{j,j'}^{d} a_j a_{j'}\left(\mathbf{A}^{-1}\right)_{jj'} - \beta a_0\right]_{h=0};$$

where $\beta\lambda_1, \beta\lambda_2, ..., \beta\lambda_d$ are the eigen values of the matrix $\beta\mathbf{A}$

[Since the determinant of a matrix is equal to the multiplication of its eigen values]

$$\Rightarrow \in_T = -\frac{1}{N}\frac{\partial}{\partial\beta}\left[-\frac{1}{2}\ln\beta^d - \frac{1}{2}\ln(\lambda_1\lambda_2.....\lambda_d) + \frac{\beta}{4}\sum_{j,j'}^{d} a_j a_{j'}\left(\mathbf{A}^{-1}\right)_{jj'} - \beta a_0\right]_{h=0}$$

$$\Rightarrow \in_T = -\frac{1}{N}\left[-\frac{d}{2\beta} + \frac{1}{4}\sum_{j,j'}^{d} a_j a_{j'}\left(\mathbf{A}^{-1}\right)_{jj'} - a_0\right]_{h=0}$$

$\Rightarrow$

$$\in_T = -\frac{1}{N}\left[-\frac{d}{2\beta} + \frac{1}{4}\sum_{j,j'}^{d}\left\{\left(h_j - 2\sum_{\alpha=1}^{N} x_j^{\alpha}\nu^{\alpha}\right)\left(h_{j'} - 2\sum_{\alpha=1}^{N} x_{j'}^{\alpha}\nu^{\alpha}\right)\left(\mathbf{A}^{-1}\right)_{jj'}\right\} - \sum_{\alpha=1}^{N}\left(\nu^{\alpha}\right)^2\right]_{h=0}$$

$$\Rightarrow \in_T = -\frac{1}{N}\left[-\frac{d}{2\beta} + \frac{1}{4}\sum_{j,j'}^{d}\left\{ \left(-2\sum_{\alpha=1}^{N} x_j^{\alpha}\nu^{\alpha} \right)\left(-2\sum_{\alpha=1}^{N} x_{j'}^{\alpha}\nu^{\alpha} \right)\left(\mathbf{A}^{-1}\right)_{jj'} \right\} - \sum_{\alpha=1}^{N}\left(\nu^{\alpha}\right)^2 \right]_{h=0}$$

$$\Rightarrow \in_T = -\frac{1}{N}\left[-\frac{d}{2\beta} + \sum_{j,j'=1}^{d}\sum_{\alpha,\alpha'=1}^{N} x_j^{\alpha} x_{j'}^{\alpha'}\nu^{\alpha}\nu^{\alpha'}\left(\mathbf{A}^{-1}\right)_{jj'} - \sum_{\alpha=1}^{N}\left(\nu^{\alpha}\right)^2 \right] \tag{1.17}$$

1.3.8 Evaluating Test Error Average using the Moment Generating Functional

We will also apply (1.12) on this (1.16) in order to obtain the average generalization error ($\in_G$) as we have mentioned earlier. It is found as below

$$\in_G = \sum_{k'k''}^{d} \Sigma_{k'k''}\left[\frac{1}{2\beta}\left(\mathbf{A}^{-1}\right)_{k'k''} + \sum_{j'j''}^{d}\sum_{\alpha'\alpha''}^{N} x_{j'}^{\alpha'} x_{j''}^{\alpha''}\nu^{\alpha'}\nu^{\alpha''}\left(\mathbf{A}^{-1}\right)_{k'j'}\left(\mathbf{A}^{-1}\right)_{k''j''} \right] + \sigma_{\nu}^2 \tag{1.18}$$

[Proof of this relation is given in APPENDIX A.5]

1.3.9 General average Training error

As the relation (1.11) for the average training error was derived by using a given training set and the expression (1.17) is obtained by following this relation primarily, the results in (1.17) is also valid for that specific training set only. But we are planning to derive expressions for the general case. Therefore, we have to average over all possible training sets.

Using $E\left[\nu^{\alpha}\nu^{\alpha'}\right] = \sigma_{\nu}^2\delta_{\alpha\alpha'}$, we obtain the general average training error ($\overline{\in_T}$) in the following way:

$$\in_T = -\frac{1}{N}\left[-\frac{d}{2\beta} + \sum_{j,j'=1}^{d}\sum_{\alpha,\alpha'=1}^{N} x_j^{\alpha} x_{j'}^{\alpha'}\nu^{\alpha}\nu^{\alpha'}\left(\mathbf{A}^{-1}\right)_{jj'} - \sum_{\alpha=1}^{N}\left(\nu^{\alpha}\right)^2 \right] \text{ [Re-writing (1.17)]}$$

$$\Rightarrow E\left[\in_T\right] = -\frac{1}{N}E\left[-\frac{d}{2\beta} + \sum_{j,j'=1}^{d}\sum_{\alpha,\alpha'=1}^{N} x_j^{\alpha} x_{j'}^{\alpha'}\nu^{\alpha}\nu^{\alpha'}\left(\mathbf{A}^{-1}\right)_{jj'} - \sum_{\alpha=1}^{N}\left(\nu^{\alpha}\right)^2 \right]$$

[Taking average on the both sides]

$$\Rightarrow \overline{\in_T} = \frac{d}{2\beta N} - \frac{1}{N}E\left[\sum_{j,j'=1}^{d}\sum_{\alpha,\alpha'=1}^{N} x_j^{\alpha} x_{j'}^{\alpha'}\nu^{\alpha}\nu^{\alpha'}\left(\mathbf{A}^{-1}\right)_{jj'} \right] + \frac{1}{N}E\left[\sum_{\alpha=1}^{N}\left(\nu^{\alpha}\right)^2 \right]$$

$$\Rightarrow \overline{\in_T} = \frac{d}{2\beta N} - \frac{1}{N}\sum_{\alpha,\alpha'=1}^{N}\left(E\left[\nu^{\alpha}\nu^{\alpha'}\right]\sum_{jj'=1}^{d} E\left[\left(x_j^{\alpha} x_{j'}^{\alpha'}\right)\left(\mathbf{A}^{-1}\right)_{jj'}\right] \right) + \frac{1}{N}\sum_{\alpha=1}^{N} E\left[\left(\nu^{\alpha}\right)^2\right]$$

$$\Rightarrow \overline{\epsilon_T} = \frac{d}{2\beta N} - \frac{1}{N}\sum_{\alpha,\alpha'=1}^{N}\left(\sigma_\nu^2\delta_{\alpha\alpha'}\sum_{jj'=1}^{d}E\left[\left(x_j^\alpha x_{j'}^{\alpha'}\right)\left(\mathbf{A}^{-1}\right)_{jj'}\right]\right) + \frac{1}{N}\sum_{\alpha=1}^{N}\sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_T} = \frac{d}{2\beta N} - \frac{1}{N}\,\sigma_\nu^2\sum_{\alpha=1}^{N}\left(\sum_{jj'=1}^{d}E\left[\left(x_j^\alpha x_{j'}^{\alpha}\right)\left(\mathbf{A}^{-1}\right)_{jj'}\right]\right) + \frac{1}{N}N\sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_T} = \frac{d}{2\beta N} - \frac{\sigma_\nu^2}{N}E\left[\left(\sum_{jj'=1}^{d}\left(\sum_{\alpha=1}^{N}x_j^\alpha x_{j'}^\alpha\right)\left(\mathbf{A}^{-1}\right)_{jj'}\right)\right] + \sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_T} = \frac{d}{2\beta N} - \frac{\sigma_\nu^2}{N}E\left[\left(\sum_{jj'=1}^{d}A_{jj'}\left(\mathbf{A}^{-1}\right)_{jj'}\right)\right] + \sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_T} = \frac{d}{2\beta N} - \frac{\sigma_\nu^2}{N}E\left[\left(\sum_{jj'=1}^{d}A_{j'j}\left(\mathbf{A}^{-1}\right)_{jj'}\right)\right] + \sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_T} = \frac{d}{2\beta N} - \frac{\sigma_\nu^2}{N}E\left[\left(\sum_{j'=1}^{d}\sum_{j=1}^{d}A_{j'j}\left(\mathbf{A}^{-1}\right)_{jj'}\right)\right] + \sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_T} = \frac{d}{2\beta N} - \frac{\sigma_\nu^2}{N}E\left[\left(\sum_{j'=1}^{d}\left(\mathbf{I}\right)_{j'j'}\right)\right] + \sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_T} = \frac{d}{2\beta N} - \frac{\sigma_\nu^2}{N}E[d] + \sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_T} = \frac{d}{2\beta N} - \frac{1}{N}d\,\sigma_\nu^2 + \sigma_\nu^2$$

$$\Rightarrow \quad \overline{\epsilon_T} = \frac{d}{2N\beta} + \left(1-\frac{d}{N}\right)\sigma_\nu^2 \qquad (1.19)$$

This is the general average training error with respect to the values of the samples. Here, we like to point out the issue that the general average training error ($\overline{\epsilon_T}$) in the above relation (1.19) is exact and no reference to the distribution of the inputs was used to calculate it; the only used condition is that the matrix, $\mathbf{A}$ must be non-singular in order to well define the Gaussian integrals that is used to express the moment generating functional, $Z_N(h,\beta)$ explicitly [from (1.13) to (1.16)].

From relation (1.19), it can easily be noticed that the value of the general average training error ($\overline{\epsilon_T}$) decreases with the increasing of β, which is desired. This can also be proved in a more general and analytic form by using the average training error (ϵ_T) in the following way:

$$\frac{1}{N}\frac{\partial \in_T}{\partial \beta} \;=\; -\left\langle \left[\textit{random mean train error} - \left\langle \textit{random mean train error} \right\rangle \right]^2 \right\rangle \;\le\; 0$$

(1.19-EXTRA)

[Proof of this (1.19-EXTRA) is given in APPENDIX A: A.3]

This tells us that the gradient of the average training error ($\in_T$) function is negative (or zero for special case) with respect to β. That implies, the average training error ($\in_T$) is a decreasing function of β. But the Expectation operator (taking the mean value), noted by $E[.]$ is a linear operator and $E[\in_T] = \overline{\in_T}$. This gives, there is a linear relation between $\in_T$ and $\overline{\in_T}$. Combining these, we conclude that the general average training error ($\overline{\in_T}$) is a decreasing function of β.

Using $\beta \to \infty$ (i.e, temperature $\to 0$), in (1.19), we get

$$\overline{\in_T} = \left(1 - \frac{d}{N}\right)\sigma_\nu^2 \quad \text{for } N \ge d\,.$$

In this case, we also see that for $N = d$, $\overline{\in_T} = 0$. But afterwards, $\overline{\in_T}$ starts to increase with the increasing of N. Anyway, this increasing ends for $N \to d$ by giving $\overline{\in_T} \to \sigma_\nu^2$, which is the maximum value of the general average training error ($\overline{\in_T}$). We will also analyze regarding this condition for our average test error later in this chapter.

1.3.10 General average Test error

In relation (1.18), we got the average generalization error that was derived by making experiment on one training set as it is based on relation (1.12), which was for a specific training set only. Now, according to our target, we will find the general average generalization error ($\overline{\in_G}$). We find it by simply taking average on (1.18) in the following way:

$$\in_G = \sum_{k'k''}^{d} \sum\nolimits_{k'k''} \left[\frac{1}{2\beta} \left(\mathbf{A}^{-1}\right)_{k'k''} + \sum_{j'j''}^{d} \sum_{\alpha'\alpha''}^{N} x_{j'}^{\alpha'} x_{j''}^{\alpha''} \nu^{\alpha'} \nu^{\alpha''} \left(\mathbf{A}^{-1}\right)_{k'j'} \left(\mathbf{A}^{-1}\right)_{k''j''} \right] + \sigma_\nu^2$$

[Re-writing (1.18)]

$$\Rightarrow E[\in_G] = E\left[\sum_{k'k''}^{d} \sum\nolimits_{k'k''} \left[\frac{1}{2\beta} \left(\mathbf{A}^{-1}\right)_{k'k''} + \sum_{j'j''}^{d} \sum_{\alpha'\alpha''}^{N} x_{j'}^{\alpha'} x_{j''}^{\alpha''} \nu^{\alpha'} \nu^{\alpha''} \left(\mathbf{A}^{-1}\right)_{k'j'} \left(\mathbf{A}^{-1}\right)_{k''j''} \right] + \sigma_\nu^2 \right]$$

[Taking average on the both sides]

$$\Rightarrow \overline{\epsilon_G} =$$

$$\frac{1}{2\beta}\sum_{k'k''}^{d}\Sigma_{k'k''}E\left[\left(\mathbf{A}^{-1}\right)_{k'k''}\right]+\sum_{k'k''}^{d}\Sigma_{k'k''}E\left[\sum_{j'j''}^{d}\sum_{\alpha'\alpha''}^{N}x_{j'}^{\alpha'}x_{j''}^{\alpha''}\nu^{\alpha'}\nu^{\alpha''}\left(\mathbf{A}^{-1}\right)_{k'j'}\left(\mathbf{A}^{-1}\right)_{k''j''}\right]+E\left[\sigma_{\nu}^{2}\right]$$

$$\Rightarrow \overline{\epsilon_G} = \frac{1}{2\beta}\sum_{k'k''}^{d}\Sigma_{k'k''}\overline{\left(\mathbf{A}^{-1}\right)_{k'k''}}+\sum_{k'k''}^{d}\Sigma_{k'k''}E\left[\sum_{j'j''}^{d}\sum_{\alpha'\alpha''}^{N}x_{j'}^{\alpha'}x_{j''}^{\alpha''}\nu^{\alpha'}\nu^{\alpha''}\left(\mathbf{A}^{-1}\right)_{k'j'}\left(\mathbf{A}^{-1}\right)_{k''j''}\right]+\sigma_{\nu}^{2}$$

$$\Rightarrow \overline{\epsilon_G} = \frac{1}{2\beta}\sum_{k'k''}^{d}\Sigma_{k'k''}\overline{\left(\mathbf{A}^{-1}\right)_{k'k''}}+\sum_{k'k''}^{d}\Sigma_{k'k''}\left[\sum_{\alpha'\alpha''}^{N}\left(\left(E\left[\nu^{\alpha'}\nu^{\alpha''}\right]\right)\left(\sum_{jj'}^{d}E\left[x_{j'}^{\alpha'}x_{j''}^{\alpha''}\left(\mathbf{A}^{-1}\right)_{k'j'}\left(\mathbf{A}^{-1}\right)_{k''j''}\right]\right)\right)\right]+\sigma_{\nu}^{2}$$

$$\Rightarrow \overline{\epsilon_G} = \frac{1}{2\beta}\sum_{k'k''}^{d}\Sigma_{k'k''}\overline{\left(\mathbf{A}^{-1}\right)_{k'k''}}+\sum_{k'k''}^{d}\Sigma_{k'k''}\left[\sum_{\alpha'\alpha''}^{N}\left(\sigma_{\nu}^{2}\,\delta_{\alpha'\alpha''}\left(\sum_{jj'}^{d}E\left[x_{j'}^{\alpha'}x_{j''}^{\alpha''}\left(\mathbf{A}^{-1}\right)_{k'j'}\left(\mathbf{A}^{-1}\right)_{k''j''}\right]\right)\right)\right]+\sigma_{\nu}^{2}$$

$$\Rightarrow \overline{\epsilon_G} = \frac{1}{2\beta}\sum_{k'k''}^{d}\Sigma_{k'k''}\overline{\left(\mathbf{A}^{-1}\right)_{k'k''}}+\sigma_{\nu}^{2}\sum_{k'k''}^{d}\Sigma_{k'k''}\left[\sum_{\alpha=1}^{N}\left(\sum_{jj'}^{d}E\left[x_{j'}^{\alpha}x_{j''}^{\alpha}\left(\mathbf{A}^{-1}\right)_{k'j'}\left(\mathbf{A}^{-1}\right)_{k''j''}\right]\right)\right]+\sigma_{\nu}^{2}$$

$$\Rightarrow \overline{\epsilon_G} = \frac{1}{2\beta}\sum_{k'k''}^{d}\Sigma_{k'k''}\overline{\left(\mathbf{A}^{-1}\right)_{k'k''}}+\sigma_{\nu}^{2}\sum_{k'k''}^{d}\Sigma_{k'k''}E\left[\sum_{j'j''=1}^{d}\left(\sum_{\alpha=1}^{N}x_{j'}^{\alpha}x_{j''}^{\alpha}\right)\left(\mathbf{A}^{-1}\right)_{k'j'}\left(\mathbf{A}^{-1}\right)_{k''j''}\right]+\sigma_{\nu}^{2}$$

$$\Rightarrow \overline{\epsilon_G} = \frac{1}{2\beta}\sum_{k'k''}^{d}\Sigma_{k'k''}\overline{\left(\mathbf{A}^{-1}\right)_{k'k''}}+\sigma_{\nu}^{2}\sum_{k'k''}^{d}\Sigma_{k'k''}E\left[\sum_{j'=1}^{d}\sum_{j''=1}^{d}\left(\sum_{\alpha=1}^{N}x_{j'}^{\alpha}x_{j''}^{\alpha}\right)\left(\mathbf{A}^{-1}\right)_{k'j'}\left(\mathbf{A}^{-1}\right)_{k''j''}\right]+\sigma_{\nu}^{2}$$

$$\Rightarrow \overline{\epsilon_G} = \frac{1}{2\beta}\sum_{k'k''}^{d}\Sigma_{k'k''}\overline{\left(\mathbf{A}^{-1}\right)_{k'k''}}+\sigma_{\nu}^{2}\sum_{k'k''}^{d}\Sigma_{k'k''}E\left[\sum_{j'=1}^{d}\sum_{j''=1}^{d}A_{j'j''}\left(\mathbf{A}^{-1}\right)_{k'j'}\left(\mathbf{A}^{-1}\right)_{k''j''}\right]+\sigma_{\nu}^{2}$$

$$\Rightarrow \overline{\epsilon_G} = \frac{1}{2\beta}\sum_{k'k''}^{d}\Sigma_{k'k''}\overline{\left(\mathbf{A}^{-1}\right)_{k'k''}}+\sigma_{\nu}^{2}\sum_{k'k''}^{d}\Sigma_{k'k''}E\left[\sum_{j''=1}^{d}\sum_{j'=1}^{d}\left(\left(\mathbf{A}^{-1}\right)_{k'j'}A_{j'j''}\right)\left(\mathbf{A}^{-1}\right)_{k''j''}\right]+\sigma_{\nu}^{2}$$

$$\Rightarrow \overline{\epsilon_G} = \frac{1}{2\beta}\sum_{k'k''}^{d}\Sigma_{k'k''}\overline{\left(\mathbf{A}^{-1}\right)_{k'k''}}+\sigma_{\nu}^{2}\sum_{k'k''}^{d}\Sigma_{k'k''}E\left[\sum_{j''=1}^{d}\left(\left(\mathbf{I}\right)_{k'j''}\right)\left(\mathbf{A}^{-1}\right)_{k''j''}\right]+\sigma_{\nu}^{2}$$

$$\Rightarrow \overline{\epsilon_G} = \frac{1}{2\beta}\sum_{k'k''}^{d}\Sigma_{k'k''}\overline{\left(\mathbf{A}^{-1}\right)_{k'k''}}+\sigma_{\nu}^{2}\sum_{k'k''}^{d}\Sigma_{k'k''}E\left[\left(\mathbf{A}^{-1}\right)_{k'k''}\right]+\sigma_{\nu}^{2}$$

$$\Rightarrow \overline{\epsilon_G} = \frac{1}{2\beta}\sum_{k'k''}^{d}\Sigma_{k'k''}\overline{\left(\mathbf{A}^{-1}\right)_{k'k''}}+\sigma_{\nu}^{2}\sum_{k'k''}^{d}\Sigma_{k'k''}\left(\overline{\left(\mathbf{A}^{-1}\right)_{k'k''}}\right)+\sigma_{\nu}^{2}$$

$$\Rightarrow \quad \overline{\epsilon_G} = \left[\frac{1}{2\beta}+\sigma_{\nu}^{2}\right]\sum_{k'k''}^{d}\Sigma_{k'k''}\,\overline{\left(\left(\mathbf{A}^{-1}\right)_{k'k''}\right)}+\sigma_{\nu}^{2} \qquad (1.20)$$

The inverse covariance matrix of the input vectors, $\left(\mathbf{A}^{-1}\right)_{jj'}$, samples as an *Inverted Wishart distribution*[6] with N degrees of freedom, $W^{-1}(\sum^{-1}, N)$, for which the mean is given by [6] :

$$\overline{\left(\mathbf{A}^{-1}\right)_{jj'}} = \frac{1}{\left(N-d-1\right)}\left(\sum^{-1}\right)_{jj'} \quad ; \quad \left(N \succ d+1\right) \tag{1.21}$$

[Here, we insist on $N \succ d+1$ as in other case $\overline{\left(\mathbf{A}^{-1}\right)_{jj'}}$ does not have a finite consistent value. For example, if $N = d+1$, we get infinite valued elements in the entries of $\overline{\left(\mathbf{A}^{-1}\right)_{jj'}}$ for a non-singular matrix $\sum$ and if $N < d+1$, we get negative valued elements in the entries of $\overline{\left(\mathbf{A}^{-1}\right)_{jj'}}$ while $\left(\mathbf{A}^{-1}\right)_{jj'}$ is the inverse of a covariance matrix.]

Using this in relation (1.20) we get,

$$\overline{\epsilon_G} = \left[\frac{1}{2\beta} + \sigma_\nu^2\right] \sum_{k'k''}^{d} \sum_{k'k''} \frac{1}{N-d-1}\left(\sum^{-1}\right)_{k'k''} + \sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_G} = \left[\frac{1}{2\beta} + \sigma_\nu^2\right] \frac{1}{N-d-1} \sum_{k'=1}^{d}\left(\sum_{k''=1}^{d} \sum_{k'k''}\left(\sum^{-1}\right)_{k'k''}\right) + \sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_G} = \left[\frac{1}{2\beta} + \sigma_\nu^2\right] \frac{1}{N-d-1} \sum_{k'=1}^{d}\left(\sum_{k''=1}^{d} \sum_{k'k''}\left(\sum^{-1}\right)_{k'k''}\right) + \sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_G} = \left[\frac{1}{2\beta} + \sigma_\nu^2\right] \frac{1}{N-d-1} \sum_{k''=1}^{d}\left(\sum_{k'=1}^{d} \sum_{k''k'}\left(\sum^{-1}\right)_{k'k''}\right) + \sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_G} = \left[\frac{1}{2\beta} + \sigma_\nu^2\right] \frac{1}{N-d-1} \sum_{k''=1}^{d}\left(\mathbf{I}\right)_{k''k''} + \sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_G} = \left[\frac{1}{2\beta} + \sigma_\nu^2\right] \frac{1}{N-d-1} d + \sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_G} = \frac{1}{2\beta}\,\frac{d}{N-d-1} + \sigma_\nu^2\left(1 + \frac{d}{N-d-1}\right)$$

$$\overline{\epsilon_G} = \frac{d}{2\beta\left(N-d-1\right)} + \frac{N-1}{N-d-1}\sigma_\nu^2 \tag{1.22}$$

Here we emphasis that this is an exact (without any approximation) finite temperature (that means $T \neq \infty$ or $\beta \neq 0$) general average of the generalization error and it is valid for arbitrary correlation among the components of the multi-normal input vector as long as the correlation matrix $\sum$ is non-singular.

[6] A short discussion about (Inverted) Wishart distribution can be found in APPENDIX B

For $\beta \to \infty$ (that is, 0 temperature), from relation (1.22), we get

$$\overline{\in_G} = \frac{N-1}{N-d-1}\sigma_\nu^2 \qquad (1.22a)$$

$$\Rightarrow \overline{\in_G} = \left(1+\frac{d}{N-d-1}\right)\sigma_\nu^2$$

From this, we see that $\overline{\in_G}$ decreases with the increasing of N (number of sample size), which is reasonable and for $N \to \infty$, $\overline{\in_G}$ approaches to σ_ν^2, which is its minimum value.
This could be seen in the figure below:

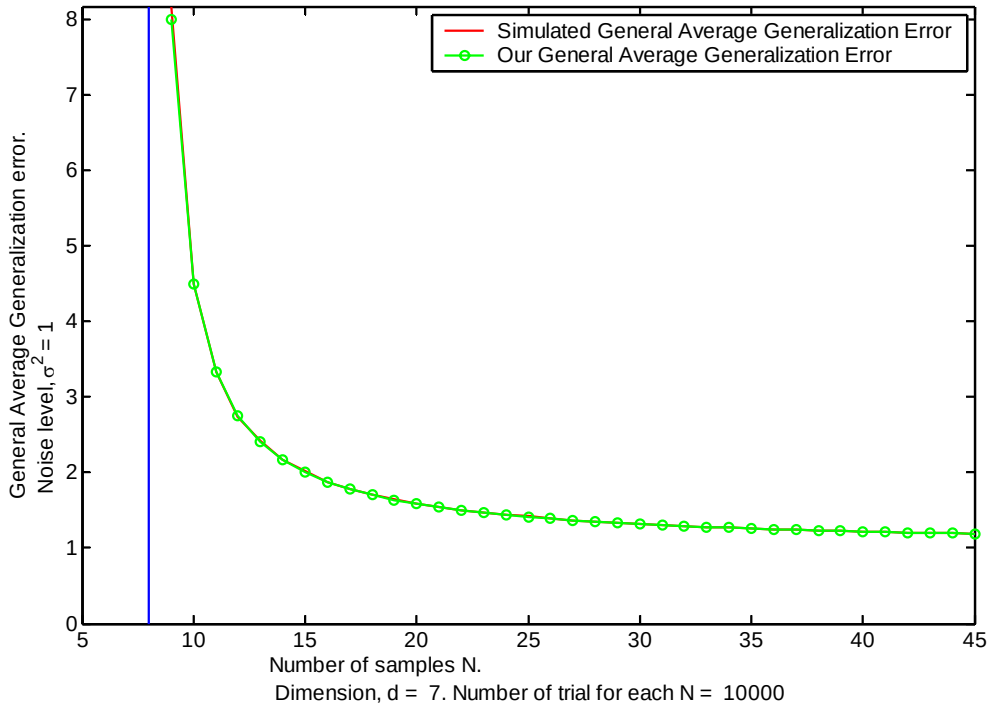

Figure 1.1: General Average Generalization error (in zero temperature) with respect to the sample size. The fresh (red) curve is from simulation while the circulated (green) curve is from our theoretical calculation following (1.22a). From figure, we see that they agree very strongly! Both of them have poles (or very high cost function values) at $N = 8\ (= d+1)$. Before this N value, the curves are not drawn as they are inconsistent for $N \le d+1$. In this figure, we also see that the cost function value gets smaller with the increasing of the sample size, N. When N gets even larger and larger, the cost function value approaches to 1, which is equal to the noise level, σ_ν^2 used for this simulation that was easily predictable before.

Here, one might complain about using any specific value of β. That means, it could be interesting to see the general average generalized error for any arbitrarily positive β (or equivalently, for arbitrarily positively finite temperature). In that case we can

skip using of β in our expression, but we will use the training error term. We do it in the following way:
From (1.19), we have

$$\overline{\epsilon_T} = \frac{d}{2N\beta} + \left(1 - \frac{d}{N}\right)\sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_T} - \left(\frac{N-d}{N}\right)\sigma_\nu^2 = \frac{d}{2N\beta}$$

$$\Rightarrow \frac{N\overline{\epsilon_T} - (N-d)\sigma_\nu^2}{N} = \frac{d}{2N\beta}$$

$$\Rightarrow 2\beta = \frac{d}{N\overline{\epsilon_T} - (N-d)\sigma_\nu^2}$$

$$\Rightarrow 2\beta = \frac{d}{E_T - (N-d)\sigma_\nu^2} \quad \text{[Using } E_T = N\overline{\epsilon_T} = \text{Training Error.]}$$

Using this expression for 2β in (1.22), we get

$$\overline{\epsilon_G} = \frac{d}{\frac{d}{E_T - (N-d)\sigma_\nu^2}(N-d-1)} + \frac{N-1}{N-d-1}\sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_G} = \frac{E_T - (N-d)\sigma_\nu^2}{N-d-1} + \frac{N-1}{N-d-1}\sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_G} = \frac{E_T}{N-d-1} + \frac{d-1}{N-d-1}\sigma_\nu^2 \qquad (1.22b)$$

$\Rightarrow$ General averaged generalization error = Training error term + Complexity term
or, equivalently,
Prediction error term = Training error term + Complexity term.

This expression can be compared to [15] .The specialty of this expression is that it is obtained regardless of any valid value of β .
Tracking the origin and the further path of (1.22b), we see:
for very small value of d (i.e, the dimension of weight vector) $\overline{\epsilon_G}$ is big as in that case E_T gives large value; on the other hand, for very big d value, $\overline{\epsilon_G}$ gets larger as in that case the terms of R.H.S of (1.22b) get rather bigger. Thus, the minimum value of General averaged generalization error ($\overline{\epsilon_G}$) represents a trade-off between these two competing effects.

1.4 Discussion and comparison with other results derived for the error functions

In conventional statistics, various results have been developed in the context of linear models, for assessing the generalization performance of trained models without the use of validation data. Similar criteria are also obtained by using similar properties in other areas like thermodynamics, statistical mechanics, etc. Parts of them are discussed below:

1.4.1 Comparing with Akaike's FPE

Combining expression (1.19) and (1.22) it is possible to find a relation between the general average training and generalization errors. We find it in the way below:

In (1.19) we have

$$\overline{\epsilon_T} = \frac{d}{2N\beta} + \left(1-\frac{d}{N}\right)\sigma_\nu^2$$

$$\Rightarrow \overline{\epsilon_T} - \frac{d}{2N\beta} = \left(1-\frac{d}{N}\right)\sigma_\nu^2$$

$$\Rightarrow \sigma_\nu^2 = \frac{\left(\overline{\epsilon_T} - \frac{d}{2N\beta}\right)}{\left(1-\frac{d}{N}\right)}$$

$$\Rightarrow \sigma_\nu^2 = \frac{\left(\overline{\epsilon_T} - \frac{d}{2N\beta}\right)}{\left(\frac{N-d}{N}\right)}$$

Using this expression in (1.22), we can express the average test error, $\overline{\epsilon_G}$ in terms of the average training error, $\overline{\epsilon_T}$ in the following way:

$$\overline{\epsilon_G} = \frac{d}{2\beta(N-d-1)} + \frac{N-1}{N-d-1}\frac{\left(\overline{\epsilon_T} - \frac{d}{2N\beta}\right)}{\left(\frac{N-d}{N}\right)}$$

$$\Rightarrow \overline{\epsilon_G} = \frac{d}{2\beta(N-d-1)} + \frac{N-1}{N-d-1}\frac{\overline{\epsilon_T}}{\left(\frac{N-d}{N}\right)} - \frac{N-1}{N-d-1}\frac{\frac{d}{2N\beta}}{\left(\frac{N-d}{N}\right)}$$

$$\Rightarrow \overline{\overline{\epsilon_G}} = \frac{N(N-1)}{(N-d-1)(N-d)}\overline{\overline{\epsilon_T}} + \frac{d}{2\beta(N-d-1)} - \frac{d}{2\beta}\frac{N-1}{N-d-1}\frac{1}{N-d}$$

$$\Rightarrow \overline{\overline{\epsilon_G}} = \frac{N(N-1)}{(N-d-1)(N-d)}\overline{\overline{\epsilon_T}} + \frac{d}{2\beta(N-d-1)}\left[1 - \frac{N-1}{N-d}\right]$$

$$\Rightarrow \overline{\overline{\epsilon_G}} = \frac{N(N-1)}{(N-d-1)(N-d)}\overline{\overline{\epsilon_T}} + \frac{d}{2\beta(N-d-1)}\left[\frac{-d+1}{N-d}\right]$$

$$\Rightarrow \overline{\overline{\epsilon_G}} = \frac{N(N-1)}{(N-d-1)(N-N)}\overline{\overline{\epsilon_T}} - \frac{d}{2\beta(N-d-1)}\left[\frac{d-1}{N-d}\right]$$

$$\Rightarrow \overline{\overline{\epsilon_G}} = \frac{N(N-1)}{(N-d-1)(N-d)}\overline{\overline{\epsilon_T}} - \frac{d-1}{(N-d-1)(N-d)}\frac{d}{2\beta} \qquad (1.23)$$

This can be compared to Akaike's Final Prediction Error estimate [7][8].
In [7] Akaike computes the ratio between the test and training errors in the *deterministic limit* $\beta \to \infty$:

$$\overline{\overline{\epsilon_G}} = \frac{N+d+1}{N-d-1}\overline{\overline{\epsilon_T}} \qquad (\#)$$

In [8] Akaike computes the ratio between the test and training errors in the *deterministic limit* $\beta \to \infty$:

$$\overline{\overline{\epsilon_G}} = \frac{N+d}{N-d}\overline{\overline{\epsilon_T}} \qquad (\#\,\#)$$

For comparatively larger values of N, d , relation (#) can be formed into relation (# #), which is the usual case. Thus, in general, we can notify the Akaike's Final Prediction Error estimate in the latter form; that is,

$$\overline{\overline{\epsilon_G}} = \frac{N+d}{N-d}\overline{\overline{\epsilon_T}} \qquad (1.24)$$

Using $\beta \to \infty$ in our expression (1.23) we find:

$$\overline{\overline{\epsilon_G}} = \frac{N}{N-d}\,\frac{N-1}{N-d-1}\overline{\overline{\epsilon_T}} \qquad (1.25)$$

This coincides with (1.24) in the limit of large training sets: $N \to \infty$ (as in both (1.24) and (1.25) if we apply $N \to \infty$ and $N >> d$, we get $\overline{\epsilon_G} \to \overline{\epsilon_T}$).

Proceeding with (1.25) for large N , we get

$$\overline{\overline{\epsilon_G}} = \frac{N}{N-d}\,\frac{N}{N-d}\overline{\overline{\epsilon_T}}$$

$$\Rightarrow \overline{\epsilon_G} = \frac{\frac{N}{d}}{\frac{N}{d}-1}\frac{N/d}{\frac{N}{d}-1}\overline{\epsilon_T}$$

$$\Rightarrow \overline{\epsilon_G} = \frac{\alpha}{\alpha-1}\frac{\alpha}{\alpha-1}\overline{\epsilon_T} \qquad [\text{Using } \alpha \equiv \frac{N}{d}]$$

$$\Rightarrow \overline{\epsilon_G} = \left(\frac{\alpha}{\alpha-1}\right)^2 \overline{\epsilon_T} \quad ; \text{Where}$$

$$\alpha \succ 1, \quad as \quad N \succ d+1 \Rightarrow \frac{N}{d} \succ 1+\frac{1}{d} \Rightarrow \alpha \succ 1+\frac{1}{d} \tag{1.26}$$

$$\Rightarrow \overline{\epsilon_G} = \left(\frac{1}{1-1/\alpha}\right)^2 \overline{\epsilon_T}$$

$$\Rightarrow \overline{\epsilon_G} = \left(1-\frac{1}{\alpha}\right)^{-2} \overline{\epsilon_T}$$

$$\Rightarrow \overline{\epsilon_G} = (1+2/\alpha)\overline{\epsilon_T} + 3\,O(\alpha^{-2})\,\overline{\epsilon_T} \qquad [\text{As } \alpha \succ 1 \Rightarrow \frac{1}{\alpha} \prec 1 \tag{1.26a}$$

Now, manipulating in the same way with Akaike's expression from (1.24), we get

$$\overline{\epsilon_G} = \frac{N/d+1}{N/d-1}\overline{\epsilon_T}$$

$$\Rightarrow \overline{\epsilon_G} = \frac{\alpha+1}{\alpha-1}\overline{\epsilon_T}$$

$$\Rightarrow \overline{\epsilon_G} = \frac{1+1/\alpha}{1-1/\alpha}\overline{\epsilon_T}$$

$$\Rightarrow \overline{\epsilon_G} = \left(-1+\frac{2}{1-1/\alpha}\right)\overline{\epsilon_T}$$

$$\Rightarrow \overline{\epsilon_G} = -\overline{\epsilon_T} + 2(1-1/\alpha)^{-1}\,\overline{\epsilon_T}$$

$$\Rightarrow \overline{\epsilon_G} = -\overline{\epsilon_T} + 2\left(1+\frac{1}{\alpha}\right)\overline{\epsilon_T} + 2\left(\frac{1}{\alpha}\right)^2 \overline{\epsilon_T} + \ldots\ldots$$

$$\Rightarrow \overline{\epsilon_G} = \left(1+\frac{2}{\alpha}\right)\overline{\epsilon_T} + 2O\left(\frac{1}{\alpha}\right)^2 \overline{\epsilon_T}$$

This is Akaike's expression in expanded and modified form. One message is obvious here that this expression agrees with our expression (1.26a) only to the first order in $\frac{1}{\alpha}$.

1.4.2 Comparing with others

Properties of linear models in the limit $N \to \infty$, but with constant load parameter, $\alpha (\equiv \frac{N}{d})$ have been investigated by several workers [9] [11] [12] [25].

Comparing with L.Ljung
In [9], L. Ljung found that the test error blows up for $N \to d$. This could be compared with our expression (1.26) for very large N , d . Because, for d is very large, $\frac{1}{d} \to 0 \Rightarrow 1 + \frac{1}{d} \to 1$ and we can use the assumption (hypothetically) $\alpha \to 1$ in relation (1.26). Using it, we find that the numerator of the R.H.S. of (1.26) tends to zero (as $\in_T \to 0$ for $\alpha \to 1$ or $N \to d$, found by analysis from (1.19)) in first degree; whereas the denominator of the R.H.S. of (1.26) tends to zero in second degree. As a result, their ratio diverges, which leads the test error to blow up.
This shows an agreement between L. Ljung's and our result for very large N , d (which is a common case).

Comparing with Krogh
In [11] [12] Krogh found phase-transition[7] like behavior for $\alpha \to 1^+$ (that means, $\alpha = 1 + q$ for q >0 and $q \to 0$), with divergent relaxation times[8] and infinite test-errors. This surely agrees with our expression (1.22a) where we showed that the test error blows up for $N \to d + 1 \Rightarrow \frac{N}{d} \to 1 + \frac{1}{d} \Rightarrow \alpha \to 1^+$. For very large N , d , the agreement between Krogh's result and our result can also be shown by using our expression (1.26) in the same way as we did for L. Lung's result.

1.5 Conclusion

We have derived exact averages of the training and test errors of a linear model for the general cases. Our results give the same as [13]. Our derivations are done in several phases; mainly considering $\beta \to \infty$ and without considering any value of β (that is, avoiding the β factor). We also have defined minimum and maximum value of the training and test errors. Our results are valid for a stochastic algorithm considering the following conditions:

- the post training distribution is Gibbsian.
- the difference between the sample size length and the model dimension length is more than one; i.e. $N \succ d + 1$

- inputs and noise are independent; their distributions are multinormal and normal respectively.

[7] The conversion by which a state leaves its cluster entirely and enters into a completely different type of cluster.
[8] Time taken for reaching a stationary situation. When this time goes to infinite or very large, it is considered to be a divergent relaxation time.

Our expressions for training and test errors are consistent for all valid sample size that extend those of Akaike to finite training set sizes for simple linear learning. They also agree with other's results like L.Ljung's and Krogh's.

Chapter 2: Exact Generalization Error in Linear Regression Model

In chapter1, we derived exact expressions for Generalization errors as a function of sample size, dimension and output noise level where the input sample covariance was probabilistic and unknown. In this chapter, we also derive expression for exact Generalization error involving two more parameters; the assumed known input covariance matrix and the true co-efficient set (weight vector). The derivation is done by an exact analysis of a simple linear regression model and the result is then compared to [20]. To my knowledge, our result (with [20]) is the first derived expression for Generalization error in regression analysis in such explicit form. We compare our theoretical result with MATLAB simulation and find a solid resemblance that recognises the undoubted validity of the theory.

Organization of this chapter:
In 2.1, we make an introductory discussion about linear regression while we deal with some of its common algebra in 2.2. In 2.3, we derive the expression of exact Generalization error for linear regression and we make simulation for its authentication. In 2.4, we also derive an expression for the covariance of the estimated weight vector. At last, we conclude the chapter in 2.5.

2.1 Introduction to Linear Regression

2.1.1 Regression

Regression Analysis may be broadly defined as the analysis of relationships among variables. It is one of the most widely used statistical tools because it provides a simple method for establishing functional relationship among variables. The relationship is expressed in the form of an equation connecting the response or the dependent variable and one or more independent variables. This equation is called the regression equation and the co-efficient (s) of the independent variable (s) is (are) called regression coefficient (s). When this equation is linear (with respect to the dependent /independent variable), we call it as linear regression. A regression containing only one independent variable is called a simple regression equation. But when the equation contains more than one independent variable, it is named as a multiple regression equation.

2.1.2 Its goal

The goal of regression analysis is to determine the values of parameters for a function that cause the function to best fit a set of data observations that is provided.

There are two primary reasons for fitting a regression equation to a set of data--first, to describe the data; second, to predict the response from the carrier. The justification behind the way the regression line is calculated is best seen from the point-of-view of prediction. A line gives a good

fit to a set of data if the points are close to it. Where the points are not tightly grouped about any line, a line gives a good fit if the points are closer to it than to any other line. For predictive purposes, this means that the predicted values obtained by using the line should be close to the values that were actually observed, that is, that the residuals should be small. Therefore, when assessing the fit of a line, the vertical distances of the points to the line are the only distances that matter. Perpendicular distances are not considered because errors are measured as vertical distances, not perpendicular distances.

When predictors are categorical, one can predict the response by simply averaging the responses observed in the training data. For numerical predictors, this is not possible since the predictor value we get in the future may not precisely match any value we have seen in the past. Right from the beginning, we need to impose constraints on how the predictors are related to the response.

2.1.3 Why Linear Regression

In real data analysis, relationships are rarely linear. But this does not necessarily mean that we have to use nonlinear regression. Fortunately, often there are simple transformations of the response and/or predictors which make the relationship linear. If possible, this approach is preferred since it generally leads to the simplest models and allows us to use tools developed for the linear case.

2.2 Algebra in Linear Regression

2.2.1 Linear Equation

With the unknown properties of the dependent and independent variables, it is not possible to determine their functional relationship (or, the regression function) a priori. It has to be estimated from the data and must therefore be suitably parametrized. Thus, we consider a noisy linear relation

$$y = \mathbf{w} \bullet \mathbf{x} + \varepsilon \qquad (2.1)$$

Where $\mathbf{y}$ is the depending variable that depends on the independent variable $\mathbf{x}$. We will try to estimate the relation between $\mathbf{y}$ and $\mathbf{x}$ based on N examples $D = \{(\mathbf{x_n}, y_n) \mid n = 1,2,...,N\}$, with $y_n = \mathbf{w_0} \bullet \mathbf{x_n} + \varepsilon_n$. Here, the $\mathbf{x_n}$ s are independent, probabilistic and a $d \times 1$ vector whereas $\mathbf{w_0}$ is a fixed $d \times 1$ (coefficient) vector. We assume that the input and the additive noise are independent and both of them belong to zero mean normal distribution, that is $\mathbf{x} \in N(\mathbf{0}, \sum)$, $\varepsilon \in N(0, \sigma^2)$. At this stage, we will assume that $\sum$ is unknown whereas the noise level σ^2 will be assumed to be known throughout the whole process.

We define the estimated covariance matrix (of the input covariance matrix $\sum$) as

$$\mathbf{A} = \frac{1}{N}\left[\sum_{n=1}^{N} \mathbf{x}_n \mathbf{x}_n^T\right]$$

But from Chapter1, we found that while the matrix $\mathbf{A}$ is non-singular with probability one for $N \geq d$, the mean of its inverse is infinite for $N \leq d+1$. Therefore, for the moment, in order to use $\mathbf{A}^{-1}$, we consider $N \succ d+1$, so that it can be well defined.

2.2.2 Least Square Estimate (LSE)

For estimating the functional relation between $\mathbf{y}$ and $\mathbf{x}$, we have to estimate $\mathbf{w_0}$. The best estimate (with the used assumptions) is found by searching for that estimate of $\mathbf{w_0}$ for which squared sum of the difference between the estimated and observed values of $\mathbf{y}$ is the minimum. This estimation is called the Least Square Estimation (LSE) or simply Ordinary Least Square (OLS) Estimation.

We make the OLS estimate of $\mathbf{w_0}$ in the following way:

$$\hat{\mathbf{w}}(\mathbf{D}) \;=\; \mathbf{A}^{-1}\mathbf{a}\,. \qquad (2.2)$$

With $\mathbf{a} = \frac{1}{N}\sum_{n=1}^{N} \left[\mathbf{x}_n \; y_n\right]$

[Proof of this relation is given in APPENDIX A.6.]

One of the good properties of this estimator is that it is an unbiased estimator that can be seen below:

$$E\left[\hat{\mathbf{w}}(\mathbf{D})\right]$$

$$= E\left[\mathbf{A}^{-1}\mathbf{a}\right]$$

$$= E\left[\left(\frac{1}{N}\left[\sum_{n=1}^{N}\mathbf{x}_n\mathbf{x}_n^T\right]\right)^{-1}\left(\frac{1}{N}\sum_{n=1}^{N}\left[\mathbf{x}_n \; y_n\right]\right)\right]$$

$$= E\left[\left(\sum_{n=1}^{N}\mathbf{x}_n\mathbf{x}_n^T\right)^{-1}\left(\sum_{n=1}^{N}\left[\mathbf{x}_n(\mathbf{x}_n^T\mathbf{w_0}+\varepsilon_n)\right]\right)\right]$$

$$= E\left[\left(\sum_{n=1}^{N}\mathbf{x}_n\mathbf{x}_n^T\right)^{-1}\left(\sum_{n=1}^{N}\mathbf{x}_n\mathbf{x}_n^T\right)\mathbf{w_0}\right] \;+\; E\left[\left(\sum_{n=1}^{N}\mathbf{x}_n\mathbf{x}_n^T\right)^{-1}\sum_{n=1}^{N}(\mathbf{x}_n\;\varepsilon_n)\right]$$

$$= E\left[\mathbf{w_0}\right] + \;\mathbf{0}$$

$$\Rightarrow\; E\left[\hat{\mathbf{w}}(\mathbf{D})\right] = \mathbf{w_0}$$

2.2.3 Generalization error

The generalization error or the final prediction error is counted as the mean square loss averaged with respect to test and training set in the following way:
Generalization error,

$$E = \langle \langle (y-\hat{y})^2 \rangle_{(y,\mathbf{x})} \rangle_D$$

$$\Rightarrow E = \langle \langle (y-\hat{\mathbf{w}}(\mathbf{D})\bullet\mathbf{x})^2 \rangle_{(y,\mathbf{x})} \rangle_D$$

By following the track of the derivation of $\hat{\mathbf{w}}(\mathbf{D})$, it is possible to realize that this Generalization error, E can be a function of the model dimension (d), the sample size (N), the noise level (σ^2), the true coefficient parameter ($\mathbf{w_0}$) and the input covariance matrix ($\sum$). Therefore, we can re-write it in the following form

$$E(d,N,\sigma^2,\mathbf{w_0},\Sigma) = \langle \langle (y-\hat{\mathbf{w}}(\mathbf{D})\bullet\mathbf{x})^2 \rangle_{(y,\mathbf{x})} \rangle_D \qquad (2.3)$$

But so far, we have only the estimated input covariance matrix $\mathbf{A}$, not the true one ($\sum$) and it is very hard to find the explicit form of the relation (2.3) using this **A.** However, a good property of this **A** is that it is an unbiased estimate of $\sum$ as we can see in the following:

$$E[\mathbf{A}] = E\left[\frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n\mathbf{x}_n^T\right] = \frac{1}{N}\sum_{n=1}^{N}E[\mathbf{x}_n\mathbf{x}_n^T] = \frac{1}{N}\sum_{n=1}^{N}\Sigma = \Sigma.$$

And from the Law of Large numbers, we can say that for a very large value of N or correspondingly, in the presence of a large set of data, $\mathbf{A}\to\sum$ (that means, when a large amount of examples are available, the input covariance matrix can be assumed to be known). Thus, in that case, we can re-write the above relation (2.2) in the following way:

$$\hat{\mathbf{w}}(\mathbf{D}) \cong \sum^{-1}\mathbf{a} \qquad (2.4)$$

Now, we will try to express relation (2.3) in an explicit form involving the assumed known input covariance matrix $\sum$.

2.3 Generalization Error with known input distribution in Linear Regression

2.3.1 Derivation of the expression for exact Generalization error

From (2.3) we have

$$\begin{aligned} E(d,N,\sigma^2,\mathbf{w_0},\Sigma) &= \langle \langle (y-\hat{\mathbf{w}}(\mathbf{D})\bullet\mathbf{x})^2 \rangle_{(y,\mathbf{x})} \rangle_D \\ &= \langle \langle (y-\mathbf{x}^T\hat{\mathbf{w}}(\mathbf{D}))^2 \rangle_{(y,\mathbf{x})} \rangle_D \\ &= \langle \langle (y_n-\mathbf{x}_n^T\hat{\mathbf{w}}(\mathbf{D}))^2 \rangle_{(y,\mathbf{x})} \rangle_D \quad [\text{Writing with the suffix where } n=1,2,...,N\ ] \end{aligned}$$

$$= \langle \langle (y_n - \mathbf{x}_n^T \hat{\mathbf{w}}(\mathbf{D}))^T (y_n - \mathbf{x}_n^T \hat{\mathbf{w}}(\mathbf{D})) \rangle_{(y,\mathbf{x})} \rangle_D$$
$$= \langle \langle (y_n^T - \hat{\mathbf{w}}^T(\mathbf{D})\mathbf{x}_n)(y_n - \mathbf{x}_n^T \hat{\mathbf{w}}(\mathbf{D})) \rangle_{(y,\mathbf{x})} \rangle_D$$

$$= \langle \langle (y_n^T y_n - y_n^T \mathbf{x}_n^T \hat{\mathbf{w}}(\mathbf{D}) - \hat{\mathbf{w}}^T(\mathbf{D})\mathbf{x}_n y_n + \hat{\mathbf{w}}^T(\mathbf{D})\mathbf{x}_n \mathbf{x}_n^T \hat{\mathbf{w}}(\mathbf{D})) \rangle_{(y,\mathbf{x})} \rangle_D$$

$$\Rightarrow E(d, N, \sigma^2, \mathbf{w_0}, \Sigma)$$
$$= \langle \langle y_n^T y_n \rangle_{(y,\mathbf{x})} - \langle y_n^T \mathbf{x}_n^T \rangle_{(y,\mathbf{x})} \hat{\mathbf{w}}(\mathbf{D}) - \hat{\mathbf{w}}^T(\mathbf{D}) \langle \mathbf{x}_n y_n \rangle_{(y,\mathbf{x})} + \hat{\mathbf{w}}^T(\mathbf{D}) \langle \mathbf{x}_n \mathbf{x}_n^T \rangle_{(y,\mathbf{x})} \hat{\mathbf{w}}(\mathbf{D}) \rangle_D$$

In the following, we will do some small vector & matrix calculations that will be used in the above expression.

We have,
$y_n = \mathbf{x}_n^T \mathbf{w_0} + \varepsilon_n$. Then $y_n^T = \mathbf{w}_0^T \mathbf{x_n} + \varepsilon_n^T$

[In fact, both y_n and ε_n are scalar quantities; still here we work with their transposes in order to show a consistent way of calculation]

Thus $\langle y_n^T y_n \rangle_{(y,\mathbf{x})} = \langle (\mathbf{w}_0^T \mathbf{x_n} + \varepsilon_n^T)(\mathbf{x}_n^T \mathbf{w_0} + \varepsilon_n) \rangle_{(y,\mathbf{x})}$

$= \mathbf{w}_0^T \langle \mathbf{x_n} \mathbf{x}_n^T \rangle_{(y,\mathbf{x})} \mathbf{w_0} + \mathbf{w}_0^T \langle \mathbf{x_n} \rangle_{(y,\mathbf{x})} \langle \varepsilon_n \rangle_{(y,\mathbf{x})} + \langle \varepsilon_n^T \rangle_{(y,\mathbf{x})_n} \langle \mathbf{x}_n^T \rangle_{(y,\mathbf{x})} \mathbf{w_0} + \langle \varepsilon_n^T \varepsilon_n \rangle_{(y,\mathbf{x})}$

[As $\mathbf{x_n}$ & ε_n are independent.]

$= \mathbf{w}_0^T \Sigma \mathbf{w_0} + \sigma^2$ [As both the input distribution and additive noise are of zero mean]

And

$$\langle y_n^T \mathbf{x}_n^T \rangle_{(y,\mathbf{x})} = \langle (\mathbf{w}_0^T \mathbf{x_n} + \varepsilon_n^T) \mathbf{x}_n^T \rangle_{(y,\mathbf{x})}$$
$$= \mathbf{w}_0^T \langle \mathbf{x_n} \mathbf{x}_n^T \rangle_{(y,\mathbf{x})} + \langle \varepsilon_n^T \rangle_{(y,\mathbf{x})} \langle \mathbf{x}_n^T \rangle_{(y,\mathbf{x})}$$
$$= \mathbf{w}_0^T \Sigma$$

Also

$$\langle \mathbf{x}_n y_n \rangle_{(y,\mathbf{x})} = \langle \mathbf{x_n} (\mathbf{x}_n^T \mathbf{w_0} + \varepsilon_n) \rangle_{(y,\mathbf{x})}$$
$$= \langle \mathbf{x_n} \mathbf{x}_n^T \rangle_{(y,\mathbf{x})} \mathbf{w_0} + \langle \mathbf{x}_n \rangle_{(y,\mathbf{x})} \langle \varepsilon_n \rangle_{(y,\mathbf{x})}$$
$$= \Sigma \mathbf{w_0}$$

Inserting these values in the above expression for $E(d, N, \sigma^2, \mathbf{w_0}, \Sigma)$, we get

$$E(d, N, \sigma^2, \mathbf{w_0}, \Sigma)$$
$$= \langle \mathbf{w}_0^T \Sigma \mathbf{w_0} + \sigma^2 - (\mathbf{w}_0^T \Sigma) \hat{\mathbf{w}}(\mathbf{D}) - \hat{\mathbf{w}}^T(\mathbf{D})(\Sigma \mathbf{w_0}) + \hat{\mathbf{w}}^T(\mathbf{D}) \Sigma \hat{\mathbf{w}}(\mathbf{D}) \rangle_D$$

$$= \sigma^2 + \mathbf{w}_0^T \Sigma \mathbf{w_0} - \langle (\mathbf{w}_0^T \Sigma)(\Sigma^{-1}\mathbf{a}) - (\Sigma^{-1}\mathbf{a})^T (\Sigma \mathbf{w_0}) + \hat{\mathbf{w}}^T(\mathbf{D}) \Sigma \hat{\mathbf{w}}(\mathbf{D}) \rangle_D$$

$$= \sigma^2 + \mathbf{w}_0^T \Sigma \mathbf{w_0} - \langle (\mathbf{w}_0^T \Sigma \Sigma^{-1}\mathbf{a}) + (\mathbf{a}^T \Sigma^{-1} \Sigma \mathbf{w_0}) - \hat{\mathbf{w}}^T(\mathbf{D}) \Sigma \hat{\mathbf{w}}(\mathbf{D}) \rangle_D$$

$$\Rightarrow E(d, N, \sigma^2, \mathbf{w_0}, \Sigma) = \sigma^2 + \mathbf{w}_0^T \Sigma \mathbf{w_0} - \mathbf{w}_0^T \langle \mathbf{a} \rangle_D - \langle \mathbf{a}^T \rangle_D \mathbf{w_0} + \langle \hat{\mathbf{w}}^T(\mathbf{D}) \Sigma \hat{\mathbf{w}}(\mathbf{D}) \rangle_D$$

As we want to express the generalization error, E in terms of the known parameters $d, N, \sigma^2, \mathbf{w_0}, \Sigma$; that means, we want $E = E(d, N, \sigma^2, \mathbf{w_0}, \Sigma)$, we also need to find out the values of $\langle \mathbf{a} \rangle_D$, $\langle \mathbf{a}^T \rangle_D$ and $\langle \hat{\mathbf{w}}^T(\mathbf{D}) \Sigma \hat{\mathbf{w}}(\mathbf{D}) \rangle_D$ using these parameters. We will find $\langle \mathbf{a} \rangle_D$ and $\langle \mathbf{a}^T \rangle_D$ in the following at first.

$$\langle \mathbf{a} \rangle_D = \langle \frac{1}{N} \sum_{n=1}^{N} (\mathbf{x}_n \; y_n) \rangle_D = \frac{1}{N} \sum_{n=1}^{N} \langle \mathbf{x}_n \; y_n \rangle_D = \frac{1}{N} \sum_{n=1}^{N} \Sigma \mathbf{w_0} = \frac{1}{N} N \Sigma \mathbf{w_0} = \Sigma \mathbf{w_0}$$

And

$$\langle \mathbf{a}^T \rangle_D = \langle \left(\frac{1}{N} \sum_{n=1}^{N} (\mathbf{x}_n \; y_n) \right)^T \rangle_D = \frac{1}{N} \sum_{n=1}^{N} \langle y_n^T \mathbf{x}_n^T \rangle_D = \frac{1}{N} \sum_{n=1}^{N} \mathbf{w_0'} \Sigma = \frac{1}{N} N \mathbf{w}_0^T \Sigma =$$

$\mathbf{w}_0^T \Sigma$.

Inserting these in the above expression for E, we get

$$E(d, N, \sigma^2, \mathbf{w_0}, \Sigma) = \sigma^2 + \mathbf{w}_0^T \Sigma \mathbf{w_0} - \mathbf{w}_0^T \Sigma \mathbf{w_0} - \mathbf{w}_0^T \Sigma \mathbf{w_0} + \langle \hat{\mathbf{w}}^T(\mathbf{D}) \Sigma \hat{\mathbf{w}}(\mathbf{D}) \rangle_D$$

$$\Rightarrow E(d, N, \sigma^2, \mathbf{w_0}, \Sigma) = \sigma^2 - \mathbf{w}_0^T \Sigma \mathbf{w_0} + \langle \hat{\mathbf{w}}^T(\mathbf{D}) \Sigma \hat{\mathbf{w}}(\mathbf{D}) \rangle_D \qquad (\alpha)$$

Now we will compute $\langle \hat{\mathbf{w}}^T(\mathbf{D}) \Sigma \hat{\mathbf{w}}(\mathbf{D}) \rangle_D$ in terms of $d, N, \sigma^2, \mathbf{w_0}, \Sigma$ in the way below.

$\langle \hat{\mathbf{w}}^T(\mathbf{D}) \Sigma \hat{\mathbf{w}}(\mathbf{D}) \rangle_D$

$= \mathrm{Tr} [\langle \hat{\mathbf{w}}^T(\mathbf{D}) \Sigma \hat{\mathbf{w}}(\mathbf{D}) \rangle_D]$ [As $\hat{\mathbf{w}}(\mathbf{D})$ is a $d \times 1$ vector and Σ is a $d \times d$ matrix].

$= \mathrm{Tr} [\langle \Sigma \hat{\mathbf{w}}(\mathbf{D}) \hat{\mathbf{w}}^T(\mathbf{D}) \rangle_D]$

$= \mathrm{Tr} [\langle \Sigma (\Sigma^{-1}\mathbf{a})(\Sigma^{-1}\mathbf{a})^T \rangle_D]$

$= \mathrm{Tr} [\langle \Sigma (\Sigma^{-1}\mathbf{a}\mathbf{a}^T \Sigma^{-1} \rangle_D]$

$\Rightarrow \langle \hat{\mathbf{w}}^T(\mathbf{D}) \Sigma \hat{\mathbf{w}}(\mathbf{D}) \rangle_D = \mathrm{Tr} [\langle \mathbf{a}\mathbf{a}^T \rangle_D \Sigma^{-1}]$ $\qquad (\beta)$

So, in order to perform the above calculation, we need to find out $\langle \mathbf{a}\mathbf{a}' \rangle_D$. We will do it in the following way

$\langle \mathbf{a}\mathbf{a}^T \rangle_D$

$$= \langle \left[\frac{1}{N} \sum_{n=1}^{N} (\mathbf{x}_n \; y_n) \right] \left[\frac{1}{N} \sum_{n=1}^{N} (y_n^T \mathbf{x}_n^T) \right] \rangle_D$$

$$= \langle \left[\frac{1}{N} \sum_{n=1}^{N} \mathbf{x_n} (\mathbf{x}_n^T \mathbf{w_0} + \varepsilon_n) \right] \left[\frac{1}{N} \sum_{n=1}^{N} (\mathbf{w}_0^T \mathbf{x_n} + \varepsilon_n^T) \mathbf{x}_n^T \right] \rangle_D$$

$$= \langle \left[\frac{1}{N} \sum_{n=1}^{N} (\mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} + \mathbf{x_n} \varepsilon_n) \right] \left[\frac{1}{N} \sum_{n=1}^{N} (\mathbf{w}_0^T \mathbf{x_n} \mathbf{x}_n^T + \varepsilon_n^T \mathbf{x}_n^T) \right] \rangle_D$$

$$= \langle \left[\frac{1}{N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} (\mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} + \mathbf{x_n} \varepsilon_n)(\mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T + \varepsilon_m^T \mathbf{x}_m^T) \right] \rangle_D$$

$$= \langle \left[\frac{1}{N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} \left(\mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T + \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{x}_m^T \varepsilon_m^T + \mathbf{x_n} \mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T \varepsilon_n + + \mathbf{x}_n \mathbf{x}_m^T \varepsilon_n \varepsilon_m^T \right) \right] \rangle_D$$

$$= \frac{1}{N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} \left[\langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T \rangle_D + \langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{x}_m^T \varepsilon_m^T \rangle_D + \langle \mathbf{x_n} \mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T \varepsilon_n \rangle_D + \langle \mathbf{x}_n \mathbf{x}_m^T \varepsilon_n \varepsilon_m^T \rangle_D \right]$$

$$= \frac{1}{N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} \left[\langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T \rangle_D + \langle \mathbf{x}_n \mathbf{x}_m^T \rangle_D \langle \varepsilon_n \varepsilon_m^T \rangle_D \right]$$

$$= \frac{1}{N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} \langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T \rangle_D + \frac{1}{N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} \Sigma \sigma^2 \delta_{mn}$$

$$\Rightarrow \langle \mathbf{a}\mathbf{a}^T \rangle_D = \frac{1}{N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} \langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T \rangle_D + \frac{1}{N} \Sigma \sigma^2 \qquad (\gamma)$$

Now we calculate the first term of the R.H.S. of the above relation, (γ) as below.

$$\frac{1}{N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} \langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T \rangle_D$$

$$= \frac{1}{N^2} \langle \left[\sum_{n=1}^{N} \sum_{m=1}^{N} \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T \right] \rangle_D$$

$$= \frac{1}{N^2} \langle \sum_{n=1}^{N} \sum_{m=1}^{N} \left[(\mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T)(1 - \delta_{mn}) + (\mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T) \delta_{mn} \right] \rangle_D$$

$$= \frac{1}{N^2} \left[N(N-1) \langle (\mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T) \rangle + N \langle (\mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_n \mathbf{x}_n^T) \rangle \right]_D \qquad [\text{For } m \neq n]$$

$$= \frac{1}{N^2} \left[N(N-1) \langle \mathbf{x_n} \mathbf{x}_n^T \rangle_D \mathbf{w_0} \mathbf{w}_0^T \langle \mathbf{x}_m \mathbf{x}_m^T \rangle_D + N \langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_n \mathbf{x}_n^T \rangle_D \right]$$

$$= \frac{1}{N^2} N(N-1) \Sigma \mathbf{w_0} \mathbf{w}_0^T \Sigma + \frac{1}{N^2} N \langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_n \mathbf{x}_n^T \rangle_D$$

$$\Rightarrow \frac{1}{N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} \langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T \rangle_D =$$

$$(1 - \frac{1}{N}) \Sigma \mathbf{w_0} \mathbf{w}_0^T \Sigma + \frac{1}{N} \langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_n \mathbf{x}_n^T \rangle_D \qquad (\delta)$$

Further, we have to calculate $\langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_n \mathbf{x}_n^T \rangle_D$ and we will do so by the following technique.

$\langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_n \mathbf{x}_n^T \rangle_D$

$$= \langle \left[\sum_{q=1}^{d} x_{\mathbf{n}p} x_{\mathbf{n}q} w_{\mathbf{0}q} \right] \left[\sum_{s=1}^{d} w_{\mathbf{0}s} x_{nr} x_{ns} \right] \rangle_D \qquad ; p,q,r,s = 1,2,....,d$$

[Here we remind ourselves once again that n represents the suffixes of the samples (or, examples) organized as column vectors and each of p,q,r,s represent the tuples (or the rows) of these column vectors; where $n = 1,2,...,N$ and $p,q,r,s = 1,2,....,d$.]

$$= \sum_{q=1}^{d} \sum_{s=1}^{d} w_{\mathbf{0}q}\ w_{\mathbf{0}s} \langle x_{\mathbf{n}p} x_{\mathbf{n}q} x_{nr} x_{ns}\ \rangle_D$$

$$= \sum_{q=1}^{d} \sum_{s=1}^{d} w_{\mathbf{0}q}\ w_{\mathbf{0}s} \left[\langle x_{\mathbf{n}p} x_{\mathbf{n}q} \rangle_D \langle x_{nr} x_{ns}\ \rangle_D + \langle x_{\mathbf{n}p} x_{\mathbf{n}r} \rangle_D \langle x_{nq} x_{ns}\ \rangle_D + \langle x_{\mathbf{n}p} x_{\mathbf{n}s} \rangle_D \langle x_{nq} x_{nr}\ \rangle_D \right]$$

[Using the Gaussian Joint Variable Theorem [16]]

$$= \sum_{q=1}^{d} \sum_{s=1}^{d} w_{\mathbf{0}q}\ w_{\mathbf{0}s} \left[(\Sigma)_{pq} (\Sigma)_{rs} + (\Sigma)_{pr} (\Sigma)_{qs} + (\Sigma)_{ps} (\Sigma)_{qr} \right]$$

$$= \left(\sum_{q=1}^{d} w_{\mathbf{0}q} (\Sigma)_{pq} \right) \left(\sum_{s=1}^{d} w_{\mathbf{0}s} (\Sigma)_{rs} \right) + (\Sigma)_{pr} \left(\sum_{q=1}^{d} \sum_{s=1}^{d} w_{\mathbf{0}q}\ w_{\mathbf{0}s} (\Sigma)_{qs} \right) + \left(\sum_{q=1}^{d} w_{\mathbf{0}q} (\Sigma)_{qr} \right) \left(\sum_{s=1}^{d} w_{\mathbf{0}s} (\Sigma)_{ps} \right)$$

$$= (\Sigma \mathbf{w}_{\mathbf{0}})(\Sigma \mathbf{w}_{\mathbf{0}})^T + (\Sigma)(Tr(\Sigma \mathbf{w}_{\mathbf{0}} \mathbf{w}_0^T)) + (\Sigma \mathbf{w}_{\mathbf{0}})(\Sigma \mathbf{w}_{\mathbf{0}})^T$$

$$\Rightarrow \langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_n \mathbf{x}_n^T \rangle_D = 2 \Sigma \mathbf{w_0} \mathbf{w}_0^T \Sigma + (\mathbf{w}_0^T \Sigma \mathbf{w_0})(\Sigma)$$

Using this value of $\langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_n \mathbf{x}_n^T \rangle_D$ in the above relation (δ), we find

$$\frac{1}{N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} \langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T \rangle_D =$$

$$(1 - \frac{1}{N}) \Sigma \mathbf{w_0} \mathbf{w}_0^T \Sigma + \frac{1}{N} \left[2 \Sigma \mathbf{w_0} \mathbf{w}_0^T \Sigma + (\mathbf{w}_0^T \Sigma \mathbf{w_0})(\Sigma) \right]$$

$$\Rightarrow \frac{1}{N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} \langle \mathbf{x_n} \mathbf{x}_n^T \mathbf{w_0} \mathbf{w}_0^T \mathbf{x}_m \mathbf{x}_m^T \rangle_D = (1 + \frac{1}{N}) \Sigma \mathbf{w_0} \mathbf{w}_0^T \Sigma + \frac{1}{N} (\mathbf{w}_0^T \Sigma \mathbf{w_0})(\Sigma)$$

Using this expression in the above relation (γ), we get

$$\langle \mathbf{a} \mathbf{a}^T \rangle_D = (1 + \frac{1}{N}) \Sigma \mathbf{w_0} \mathbf{w}_0^T \Sigma + \frac{1}{N} (\mathbf{w}_0^T \Sigma \mathbf{w_0})(\Sigma) + \frac{1}{N} \Sigma \sigma^2 \qquad (2.5)$$

Further, using this (2.5) in the above relation (β), we get

$$\langle \hat{\mathbf{w}}^T(\mathbf{D}) \Sigma \hat{\mathbf{w}}(\mathbf{D}) \rangle_D = \text{Tr} \left[\left\{ (1 + \frac{1}{N}) \Sigma \mathbf{w_0} \mathbf{w}_0^T \Sigma + \frac{1}{N} (\mathbf{w}_0^T \Sigma \mathbf{w_0})(\Sigma) + \frac{1}{N} \Sigma \sigma^2 \right\} \Sigma^{-1} \right]$$

$$= \text{Tr} \left[\left\{ (1 + \frac{1}{N}) \Sigma \mathbf{w_0} \mathbf{w}_0^T + \frac{1}{N} (\mathbf{w}_0^T \Sigma \mathbf{w_0}) \mathbf{I} + \frac{1}{N} \mathbf{I} \sigma^2 \right\} \right]$$

$$= (1 + \frac{1}{N}) Tr(\Sigma \mathbf{w_0} \mathbf{w}_0^T) + \frac{1}{N} (\mathbf{w}_0^T \Sigma \mathbf{w_0}) Tr(\mathbf{I}) + \frac{1}{N} Tr(\mathbf{I}) \sigma^2$$

$$\Rightarrow \quad \langle \hat{\mathbf{w}}^T(\mathbf{D}) \sum \hat{\mathbf{w}}(\mathbf{D}) \rangle_D \quad =$$

$$(1+\frac{1}{N})\left(\mathbf{w}_0^T \sum \mathbf{w_0}\right)+\frac{d}{N}(\mathbf{w}_0^T \sum \mathbf{w_0})+\frac{d}{N}\sigma^2$$

At last, using this expression in the above relation (α), we get

$$E(d,N,\sigma^2,\mathbf{w_0},\Sigma)= \sigma^2 - \mathbf{w}_0^T \sum \mathbf{w_0} + \ (1+\frac{1}{N})\left(\mathbf{w}_0^T \sum \mathbf{w_0}\right)+\frac{d}{N}(\mathbf{w}_0^T \sum \mathbf{w_0})+\frac{d}{N}\sigma^2$$

$$\Rightarrow E(d,N,\sigma^2,\mathbf{w_0},\Sigma)= \sigma^2 \ +\frac{d}{N}\sigma^2+(\frac{1+d}{N})\left(\mathbf{w}_0^T \sum \mathbf{w_0}\right)$$

$$\Rightarrow E(d,N,\sigma^2,\mathbf{w_0},\Sigma) = (1+\frac{d}{N}+\frac{d+1}{N}\ \frac{\mathbf{w}_0^T \sum \mathbf{w_0}}{\sigma^2}\ )\ \sigma^2 \qquad (2.6)$$

But we have found before that the average total energy of the signal is
$\langle y^2{}_n \rangle = \langle y'_n y_n \rangle = \mathbf{w}_0^T \sum \mathbf{w_0} + \sigma^2$ = signal energy + noise energy.

Therefore, the Signal to Noise Ratio (SNR) is

$$\text{SNR} = \frac{\textit{Signal Power}}{\textit{Noise Power}} \equiv \frac{\textit{Signal Energy}}{\textit{Noise Energy}} = \frac{\mathbf{w}_0^T \sum \mathbf{w_0}}{\sigma^2}$$

So, we can re-write the relation (2.6) in the form below

$$\Rightarrow E(d,N,\sigma^2,\mathbf{w_0},\Sigma) = (1+\frac{d}{N}+\frac{d+1}{N}\ SNR\ )\ \sigma^2 \qquad (2.7)$$

This is an exact Generalization Error in case of Linear Regression model and is a function of the signal to noise ratio in addition to the model dimension, sample size and the input noise level. From this expression we also see that the error increases with the increasing of the model complexity (dimension) and decreases with in increasing of the sample size, which is a pretty common idea in learning theory. But the peculiarity of this expression is that it shows the increasing of error with the increasing of the signal to noise ratio.

2.3.2 Simulation and comparison for the Generalization error calculation

In order to verify the purity of our theory (2.6), we make a MATLAB simulation that compares our result with the simulated one. Figure 2.1 shows that comparison (below):

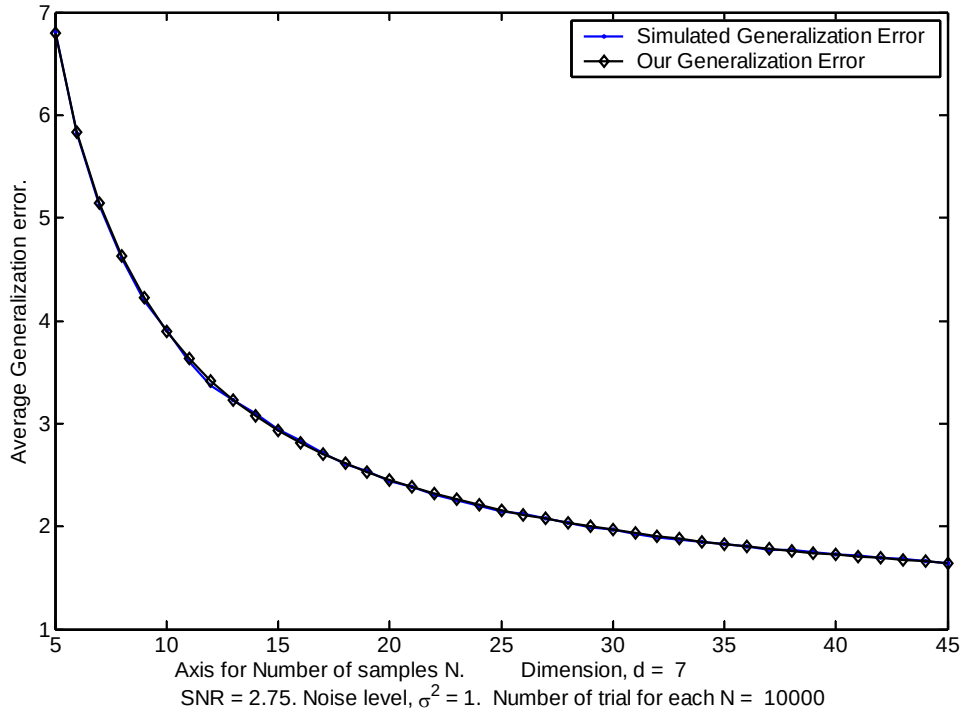

Figure 2.1: Generalization Error curves with respect to the sample size. The dotted (blue) one is the simulated curve whereas the crystal (black) one is from our theoretical calculation (found from (2.6) or (2.7)). Figure shows their strong agreement!! From figure, we also see that the cost function value gets smaller with the increasing of N . When N gets even more and more large, the cost function approaches to 1, which is equal to the noise level for this simulation. This was easily predictable from relations (2.6) or (2.7). This figure is obtained using the known input distribution.

2.4 Covariance of the estimated weight vector

The covariance of $\hat{\mathbf{w}}(\mathbf{D})$ is given by

$$Cov(\hat{\mathbf{w}}(\mathbf{D}))$$

$$= \left\langle \left(\hat{\mathbf{w}}(\mathbf{D}) - E\left[\hat{\mathbf{w}}(\mathbf{D})\right]\right)\left(\hat{\mathbf{w}}(\mathbf{D}) - E\left[\hat{\mathbf{w}}(\mathbf{D})\right]\right)^T \right\rangle$$

$$= \left\langle \left(\hat{\mathbf{w}}(\mathbf{D}) - \mathbf{w}_0\right)\left(\hat{\mathbf{w}}(\mathbf{D}) - \mathbf{w}_0\right)^T \right\rangle$$

$$= \langle\, \hat{\mathbf{w}}(\mathbf{D})\, \hat{\mathbf{w}}^T(\mathbf{D}) \,\rangle - \langle \hat{\mathbf{w}}(\mathbf{D}) \rangle \mathbf{w}_0^T - \mathbf{w}_0 \langle\, \hat{\mathbf{w}}^T(\mathbf{D}) \,\rangle + \mathbf{w}_0 \mathbf{w}_0^T$$

$$= \Sigma^{-1} \langle \mathbf{a}\mathbf{a}^T \rangle \Sigma^{-1} - \Sigma^{-1} \langle \mathbf{a} \rangle \mathbf{w}_0^T - \mathbf{w}_0 \langle\, \mathbf{a}^T \rangle \Sigma^{-1} + \mathbf{w}_0 \mathbf{w}_0^T$$

$$= \Sigma^{-1} \left[\left(1 + \frac{1}{N}\right) \Sigma \mathbf{w}_0 \mathbf{w}_0^T \Sigma + \frac{1}{N} (\mathbf{w}_0^T \Sigma \mathbf{w}_0)(\Sigma) + \frac{1}{N} \Sigma \sigma^2 \right] \Sigma^{-1} - \Sigma^{-1} \Sigma \mathbf{w}_0 \mathbf{w}_0^T - \mathbf{w}_0 \mathbf{w}_0^T \Sigma \Sigma^{-1} + \mathbf{w}_0 \mathbf{w}_0^T$$

[Using the results from (2.5) and some others above]

$$\Rightarrow Cov(\hat{\mathbf{w}}(\mathbf{D})) = \frac{1}{N} \mathbf{w}_0 \mathbf{w}_0^T + \frac{1}{N} \left[\mathbf{w}_0^T \Sigma \mathbf{w}_0 + \sigma^2 \right] \Sigma^{-1} \qquad (2.8)$$

2.5 Conclusion

We successfully managed to derive the exact expression for Generalization error in Linear Regression model that is an explicit function of the model dimension, the sample size, associative additive noise level, true regression coefficient parameter and the known input covariance matrix. Our result gives the same as [20]. The parameters in our expression arise in such a form that the error can be treated also as a function of Signal to Noise Ratio. Figure (2.1) shows a severely strong similarity between our theoretical curve and the simulated one. This proves a strong validity of our theory.
We also derived the exact expression for the covariance of our (unbiased) estimated weight vector. But as it is in the matrix form, it is really hard and long process to make any proper graphical comparison for it, which is not in high priority with respect to the time limitation of this thesis.

Chapter 3: Generalized Cross-over

In chapter1, we derived expressions for generalized error with respect to the sample size, input data dimension and the input additive noise level, where the input covariance matrix was probabilistic and unknown. In chapter 2, we derived this expression in terms of input sample size, input dimension, true weight vector, additive noise level and the assumed known input covariance matrix. In this chapter, we make short mathematical analyses with these expressions and thus search for the classes and behaviors of the curves produced by them. Then the cross point of these curves are traced. At last, we conclude by making a short comparison between them regarding their usefulness.

The chapter is organized in the following way: In 3.1, we analyze the expressions of the generalized curves. In 3.2, we find the cross point of two learning curves obtained from these two expressions of the Generalized curves. In 3.3, we make a mild comparison between the methods regarding their merits depending on the cost functions produced by them. And at last, we make a very short conclusion in 3.4.

3.1 Expressions of the generalized errors and their properties

From chapter1, relation (1.22a), we have

$$\overline{\in_G} \;=\; \frac{N-1}{N-d-1}\sigma_v^2 \qquad [\text{For } N \succ d+1\ ] \tag{3.1}$$

And from chapter 2, relation (2.7), we have

$$E(d,N,\sigma^2,\mathbf{w_0},\Sigma) = (1+\frac{d}{N}+\frac{d+1}{N}\ SNR\ )\ \sigma^2 \tag{3.2}$$

As both of the above expressions talk about the Generalization error, in a sense they are of the same type. Therefore, while manipulating with them, we will use different but same type of notations for them. This is just for manipulating comfort and decoration only. We simply denote $\overline{\in_G}$ or $\overline{\in_G}(d,N,\sigma^2)$ of (3.1) by E_1 and $E(d,N,\sigma^2,\mathbf{w_0},\Sigma)$ of (3.2) by E_2. Using these notations, (3.1) and (3.2) can be re-written as

$$E_1 \;=\; \frac{N-1}{N-d-1}\sigma_v^2 \tag{3.3}$$

And

$$E_2 = (1+\frac{d}{N}+\frac{d+1}{N}\ SNR\ )\ \sigma^2 \tag{3.4}$$

Now we will analyze with their mathematical form and find the classes of the curves produced by them.

3.1.1 Generalized error expression in the 1st way

In (3.3) we have

$$E_1 = \frac{N-1}{N-d-1}\sigma_\nu^2$$

$$\Rightarrow E_1 = \sigma_\nu^2 + \frac{d}{N-d-1}\sigma_\nu^2$$

$$\Rightarrow E_1 - \sigma_\nu^2 = \frac{d}{N-d-1}\sigma_\nu^2$$

$$\Rightarrow (E_1 - \sigma_\nu^2)(N-d-1) = d\,\sigma_\nu^2$$

$$\Rightarrow \widetilde{E}_1\,\widetilde{N} = C_1 \qquad (3.5)$$

[Where $\widetilde{E}_1 = E_1 - \sigma_\nu^2$; $\widetilde{N} = N-d-1$; $C_1 = d\,\sigma_\nu^2$ = Constant]

This is an equation of a *Rectangular Hyperbola* (in the form of $x\ y = Cons\tan t$) [18].
The two perpendicular asymptotes are at

$$\widetilde{E}_1 = 0$$

$$\Rightarrow E_1 - \sigma_\nu^2 = 0$$

$$\Rightarrow E_1 = \sigma_\nu^2$$

And

$$\widetilde{N} = 0$$

$$\Rightarrow N-d-1 = 0$$

$$\Rightarrow N = d+1$$

That means, the curve produced by relation (3.3) or, (3.1) (or, 1.22a in chapter 1) representing our 1st way of generalization error has two asymptotes; one is the line $E_1 = \sigma_\nu^2$ and the other one is the line $N = d+1$.

The line $E_1 = \sigma_\nu^2$ (parallel to the sample size axis) gives us the minimum value of the generalized error with the maximum value of the sample size N (here, infinitely large).

On the other hand, the line $N = d+1$ (parallel to the Generalized Error axis) tells us the maximum value (here, infinitely large) of the Generalized Error at the minimum valid value (with open boundary!) of the sample size. Below is the figure of the Generalization Error in the 1st way with its two asymptotes.

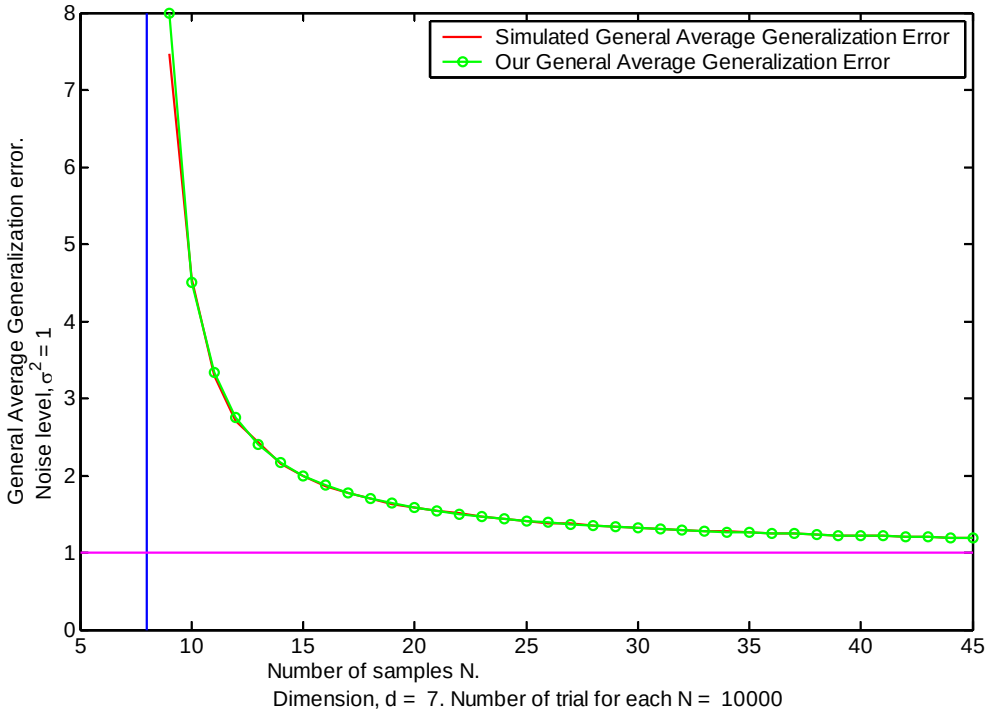

Figure 3.1: Generalization error with respect to the sample size in the 1st way with its two perpendicular asymptotes. In chapter1, we also used the same figure but with the vertical asymptote only. The vertical asymptote is the (blue) line $N = d + 1$, parallel to the Error axis and passes the sample size axis at d+1. The horizontal asymptote is the (magenta) line $E_1 = \sigma_\nu^2$, parallel to the sample size axis and passes the Error axis at σ_ν^2.

3.1.2 Generalized error expression in the 2nd way

In (3.4) we have

$$E_2 = (1 + \frac{d}{N} + \frac{d+1}{N} SNR\ )\ \sigma^2$$

$$\Rightarrow E_2 - \sigma^2 = \frac{d + (d+1)SNR}{N}\ \sigma^2$$

$$\Rightarrow (E_2 - \sigma^2)\ N = \left[d + (d+1)SNR\right]\ \sigma^2$$

$$\Rightarrow \widetilde{E}_2\ N = C_2 \qquad (3.6)$$

[Where, $\widetilde{E}_2 = E_2 - \sigma^2$; $C_2 = d + (d+1)SNR$ = Constant]

This is also an equation of a *Rectangular Hyperbola* (in the form of $x\ y = Cons\tan t$) [18]. The two perpendicular asymptotes are at

$\widetilde{E}_2 = 0$

$\Rightarrow\ E_2 - \sigma_v^2 = 0$

$\Rightarrow\ E_2 = \sigma_v^2$

And

$N = 0$

Thus, the learning curve produced by relation (3.4) or, (3.2) (or 2.7 in chapter 2) representing our 2nd way of Generalization Error (or, cost function) has two asymptotes; one is the line $E_2 = \sigma^2$ and the other one is the line $N = 0$.

The line $E_2 = \sigma^2$ is the same as $E_1 = \sigma_v^2$. Thus the two generalized error curve have the same horizontal asymptote that gives the lower limit of the error in case of the upper limit of the sample size. We denote these two lines in one expression as $E = \sigma^2$ [considering $\sigma_v^2 = \sigma^2$].

On the other hand, the line $N = 0$ is the generalized error axis; this tells us that average error goes infinite when there is no sample. This is quite ridiculous, impractical… and we will not focus on it at all.

Below is the figure of the Generalization Error in the 2nd way with its two asymptotes

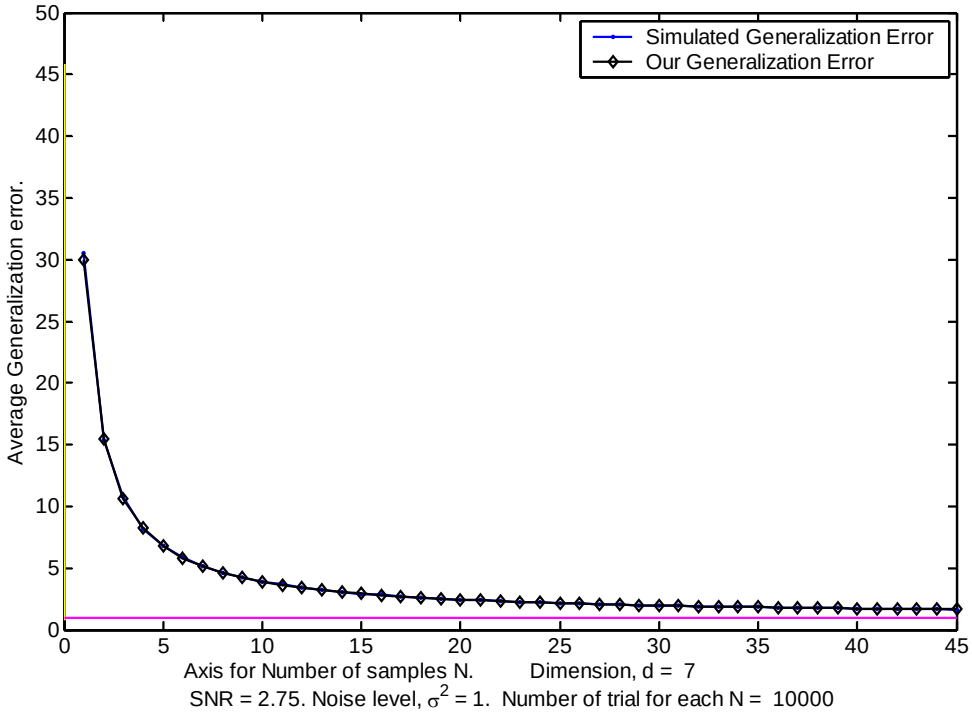

Figure 3.2: Generalization Error with respect to the sample size in the 2nd way with its two perpendicular asymptotes. In chapter2, we also used the same figure but without showing any asymptote. The vertical asymptote is the (yellow) line $N = 0$, the Error axis. The horizontal asymptote is the (magenta) line $E_2 = \sigma^2$, parallel to the sample size axis and passes the error axis at σ^2, which is the same in case of the 1st way of generalization error.

Up to now, we have discussed about the properties of the two generalized error curves. Now, we will find their cross point and some other relevant values.

3.2 Finding the cross point of the two learning curves:

In the above analysis, we found that the two curves, produced by relations (3.3) and (3.4) have the four asymptotes as $E = \sigma^2$, $N = d + 1$ and $N = 0$. Therefore, the cross point of these two curve must lie in the truncated plane defined by $E \succ \sigma^2$ and $N \succ d + 1$. We will investigate it in the following.

As both of (3.3) and (3.4) represent the generalized error, they are equivalent. Comparing these two relations, we get

$\frac{N-1}{N-d-1}\sigma^2 = (1 + \frac{d}{N} + \frac{d+1}{N} SNR\ )\ \sigma^2$ [considering $\sigma_v^2 = \sigma^2$, as their different value does not give big difference meaning of the context here; it only brings the complexity of the calculation .]

$$\Rightarrow \frac{N-1}{N-d-1} = 1+\frac{d}{N}+\frac{d+1}{N}\,SNR \quad [\text{As}\,\sigma^2 \succ 0\,]$$

$$\Rightarrow \frac{N-1}{N-d-1} - 1 = \frac{d}{N}+\frac{d+1}{N}\,SNR$$

$$\Rightarrow \frac{d}{N-d-1} = \frac{d}{N}+\frac{d+1}{N}\,SNR$$

$$\Rightarrow d = \frac{d}{N}(N-d-1)+\frac{d+1}{N}(N-d-1)\,SNR$$

$$\Rightarrow d = d-\frac{d^2}{N}-\frac{d}{N}+(d+1)SNR-\frac{(d+1)^2}{N}\,SNR$$

$$\Rightarrow \frac{d^2}{N}+\frac{d}{N}-(d+1)SNR+\frac{(d+1)^2}{N}\,SNR = 0$$

$$\Rightarrow d^2+d+(d+1)^2\,SNR = N(d+1)SNR$$

$$\Rightarrow N = \frac{d^2+d+(d+1)^2\,SNR}{(d+1)SNR}$$

$$\Rightarrow N = \frac{(d+1)\{d+(d+1)\,SNR\}}{(d+1)SNR}$$

$$\Rightarrow N = \frac{\{d+(d+1)\,SNR\}}{SNR}$$

$$\Rightarrow N = d+1+\frac{d}{SNR} \qquad (3.7)$$

This is the value of N (sample size) where the two generalization curves cross each other.
Inserting this value of N in relation (3.3), we get

$$E_1 = \frac{d+1+\frac{d}{SNR}-1}{d+1+\frac{d}{SNR}-d-1}\,\sigma^2$$

$$\Rightarrow E = \frac{d+\frac{d}{SNR}}{\frac{d}{SNR}}\,\sigma^2$$ [As at the cross point, E represents both types of learning curves]

$$\Rightarrow E = (1+SNR)\,\sigma^2$$

$$\Rightarrow E = (1+\frac{\mathbf{w}_0^T\sum\mathbf{w_0}}{\sigma^2})\,\sigma^2$$

$$\Rightarrow E = \sigma^2 + \mathbf{w}_0^T\sum\mathbf{w_0} \qquad (3.8)$$

$$\Rightarrow E = y^2$$

$\Rightarrow E$ = Total Energy of signal in case of the 2nd curve.

This shows that at the cross point, the error is fixed. It is not directly dependent on d (dimension) and independent of N (sample size).

Now, we will investigate this cross point obtained in (3.7) and (3.8).
We have, $\sigma^2 \succ 0$ and $\mathbf{w}_0^T \sum \mathbf{w}_0 \succ 0$ for any non-zero $\mathbf{w}_0$ and covariance matrix $\sum$.
Therefore, $SNR = \dfrac{\mathbf{w}_0^T \sum \mathbf{w}_0}{\sigma^2}$ is positively finite. Thus, $N = d + 1 + \dfrac{d}{SNR} \succ d + 1$.
Moreover, $E = \sigma^2 + \mathbf{w}_0^T \sum \mathbf{w}_0 \succ \sigma^2$. Thus the cross point lies in the above mentioned truncated plane and valid.

A figure of showing the cross-pint is given below:

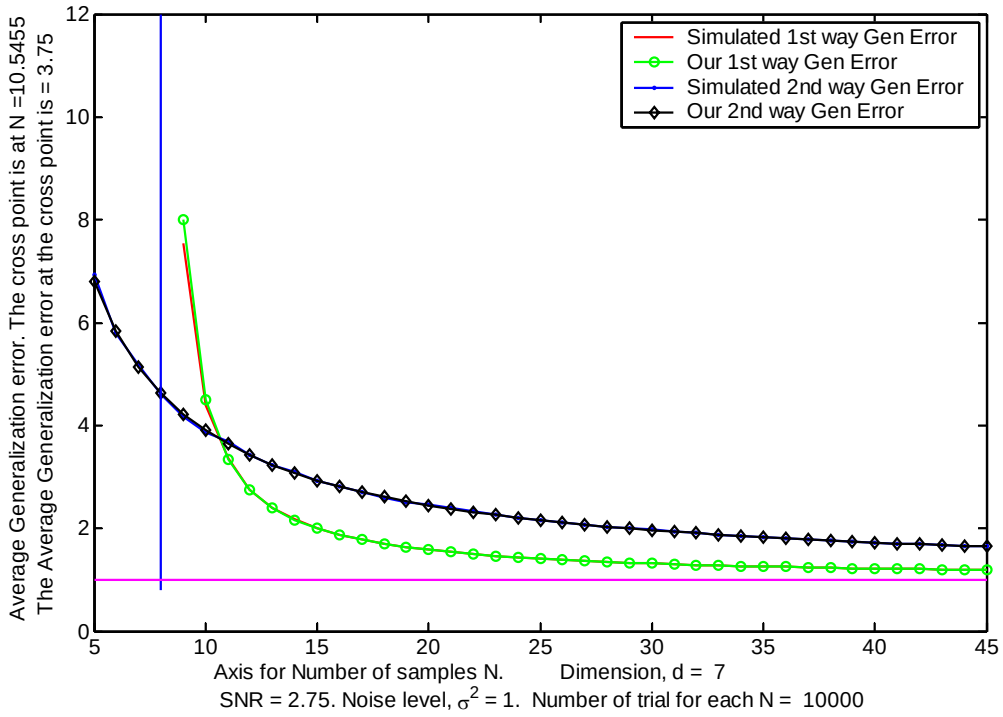

Figure 3.3: Crossing of the two curves representing the Generalization Errors with respect to sample size in the 1st and 2nd way. These two curves have a common horizontal (magenta colored) asymptote, $E = \sigma^2$. But they have different vertical asymptotes: $N = d + 1$ (blue colored) for the 1st way and $N = 0$ (not shown here because of its least importance in order to make a better figure) for the 2nd way. According to relations (3.7) and (3.8), at the cross point, the value of the sample size, N should be equal to 10.5455 and the error value should be equal to 3.75 (total energy of the 2nd curve). Figure shows a strong support to this statement.

3.3 Comparison between these two methods

For comparing these two methods, first of all we re-write the expressions (3.3) and (3.4) once again in the following way:

The 1st method gives the generalized error in the form

$$E_1 = \frac{N-1}{N-d-1}\sigma_v^2$$

$$\Rightarrow E_1 = \left(1+\frac{d}{N-d-1}\right)\sigma^2 \quad [\text{Using } \sigma_v^2 = \sigma^2] \tag{3.9}$$

And the 2nd method gives the generalized error in the form

$$E = (1 + \frac{d}{N} + \frac{d+1}{N} SNR\ )\ \sigma^2$$

$$\Rightarrow E = \left(1 + \frac{d}{N}\right)\sigma^2 + \left(\frac{d+1}{N}\right)(\mathbf{w}_0^T \sum \mathbf{w_0}) \quad [\text{As } SNR = \frac{\mathbf{w}_0^T \sum \mathbf{w_0}}{\sigma^2}] \tag{3.10}$$

Now, if we remove the stochasticity of the process by setting $\sigma^2 = 0$ in relation (3.9) and (3.10), we obtain:

Generalization error in the 1st way (from (3.9)) becomes zero, which is desired.
In contrast, the Generalization error in the 2nd case (from (3.10)) still remains with the value $\left(\frac{d+1}{N}\right)(\mathbf{w}_0^T \sum \mathbf{w_0})$, which is undesired! Although for very large N (>>d), this error may converge to zero (depending on the signal energy, $\mathbf{w}_0^T \sum \mathbf{w_0}$), it has a significant effect on the cost function for small N .
Therefore, in the non-stochastic process (which is almost impossible in real life!!), we should always use the 1st method.
Now, for comparing these two methods in case of stochastic process, at first, we will find the difference of the learning curve values (or, error values) produced from them.
Subtracting (3.9) from (3.10) we get

$$E_2 - E_1 = \left(\frac{d+1}{N}\right)(\mathbf{w}_0^T \sum \mathbf{w_0}) + \left(\frac{d}{N} - \frac{d}{N-d-1}\right)\sigma^2$$

$$\Rightarrow E_2 - E_1 = \left(\frac{d+1}{N}\right)(\mathbf{w}_0^T \sum \mathbf{w_0}) - \left(\frac{d(d+1)}{N(N-d-1)}\right)\sigma^2 \tag{3.11}$$

Now, as for $N \le d+1$, E_1 is not defined (from (3.9) or (3.1)) and the 1st method is not valid; hence the 2nd method is comparatively better for modeling.
Afterwards, we will be looking for the event $N \succ d+1$. In that case,
$E_2 \succ E_1$ if

$$\left(\frac{d+1}{N}\right)(\mathbf{w}_0^T \sum \mathbf{w_0}) \succ \left(\frac{d(d+1)}{N(N-d-1)}\right)\sigma^2$$

$$\Rightarrow N \succ d+1+\frac{d}{SNR} \text{ [After doing a tiny simple algebra]}$$

; Where $SNR = \frac{\mathbf{w}_0^T \sum \mathbf{w_0}}{\sigma^2}$

That means, from just after $N = d+1+\frac{d}{SNR}$ to any finite value of N, $E_2 \succ E_1$ (that is the cost function from the 2nd method is larger than the cost function from the 1st method); this can also be seen from the figure 3.3. Therefore, after this value of N (Sample size), the first method is comparatively better for modeling. But this N value is the cross point of the two learning curves. So, in narrative form, we say that if the given training sample size is larger than this cross point sample size value, then it is better to model by following the 1st method.
Here, we also see that a low SNR gives comparatively larger value for N as the crossing point; in this case, the 2nd method is recommended for comparatively larger

sample sizes. But for extremely low *SNR* implies that the 2nd method is a little improved form of the 1st method.

3.4 Conclusion

We found out the type and the properties of the learning curves produced in two methods. We were also able to locate their cross point in the consistently finite plane of the tuple: (sample size, generalized error). The error value at this cross-point, which is independent of the sample size, was also detected. Finally, we made a mild comparison between the two methods with respect to their merits depending on the learning curves produced from them.

Chapter 4: Matrix Regularization

In Chapter 1, we found that the generalization error becomes unbounded and invalid while the sample size $\in$ [1, 1+ model dimension]. In this chapter, we will investigate that issue by making regularization of the estimated input covariance matrix.

Organization of the topics in this chapter:
In 4.1, we talk about the regularization of the estimated input covariance matrix to find its necessity and action. In 4.2, we make simulations in order to investigate this issue further. At last, we conclude the chapter in 4.3.

4.1 Explanation about the regularization

4.1.1 Necessity of Regularization

When we use only **A** as an estimate of the input covariance matrix, then we are in a little risk for the lower number of sample size; specially, for the sample size value near to the value of model dimension. In that case, the fluctuation of the elements of **A,** leads to an increased probability of the small eigen value of **A** . Even some cases, one or many of the eigen values may become zero or very close to it; i.e. .the rank of **A** goes down (less than the full rank). This will cause the determinant of **A** to be zero, i.e. $|\mathbf{A}| = 0$. Therefore, the inverse of **A** or $\mathbf{A}^{-1}$ will have infinite valued elements, which will be still infinite after taking the mean of $\mathbf{A}^{-1.}$ This leads the general averaged generalization error $\overline{\in}_G$ to have infinite value [A.8]. In order to avoid this phenomenon, we need to tune or regularize this estimated input covariance matrix **A.**

4.1.2 Actions of Regularization

If we would choose our sample covariance matrix as $\mathbf{A} + \lambda \mathbf{I}$, where **I** is the unit (identity) matrix and λ is the regularizer with $\lambda \succ 0$, this λ would somehow increase the number of degrees of freedom of the elements in $\left(\overline{A^{-1}}\right)$; i.e. the number of degrees of freedom of the elements in ($\overline{(\mathbf{A} + \lambda \mathbf{I})^{-1}}$) is greater than the numbers of degrees of freedom of the elements in $\left(\overline{A^{-1}}\right)$ [A.9]. Thus, $\overline{(\mathbf{A} + \lambda \mathbf{I})^{-1}}$ will have finite valued elements [A.8] [A.9] and therefore, the estimated coefficient vector (or estimated weight vector) will become finite. As a result, the general averaged generalization error $\overline{\in}_G$ will have finite value. We can also say it another way, as $\lambda \mathbf{I}$ has a full rank, $\mathbf{A} + \lambda \mathbf{I}$ will have full rank too without caring the fluctuation in **A** . Therefore, $(\mathbf{A} + \lambda \mathbf{I})^{-1}$ will have finite valued element and thus their mean i.e. $\left(\overline{\mathbf{A} + \lambda \mathbf{I}}\right)^{-1}$ is expected (usually) to have finite element. This will lead the general averaged generalization error $\overline{\in}_G$ to have finite value. This treatment can be called Matrix Regularization.

But in this case, we face another problem. That is, we may loose part of our information because of introducing $\lambda \mathbf{I}$ that was not related to our input distribution.
We can also find a difference between these two estimators (regularized and unregularized) with respect to the biased unbiased property. It can be seen below:

For only **A** as estimated input covariance matrix, we have

$E[\mathbf{A}] = E[\sum_{\alpha=1}^{N} x_j^{\alpha} x_{j'}^{\alpha}]$ [Using the expression of **A** from chapter 1]

$\Rightarrow E[\mathbf{A}] = N\boldsymbol{\Sigma}$

This is proportionally unbiased.

But in contrast, for $\mathbf{A} + \lambda \mathbf{I}$, we get $E[\mathbf{A} + \lambda \mathbf{I}] = E[\mathbf{A}] + \lambda \mathbf{I} = N\boldsymbol{\Sigma} + \lambda \mathbf{I}$, which is biased.

Still, by tuning this λ - value properly, we can recover this information loosing, which may not cure the problem completely but at least can give a better solution.

Below we have shown some simulations regarding the estimated matrix regularization.

4.2 Simulations

4.2.1 Simulation concerning sample size (N), model dimension (d) and the determinant of the estimated input covariance matrix

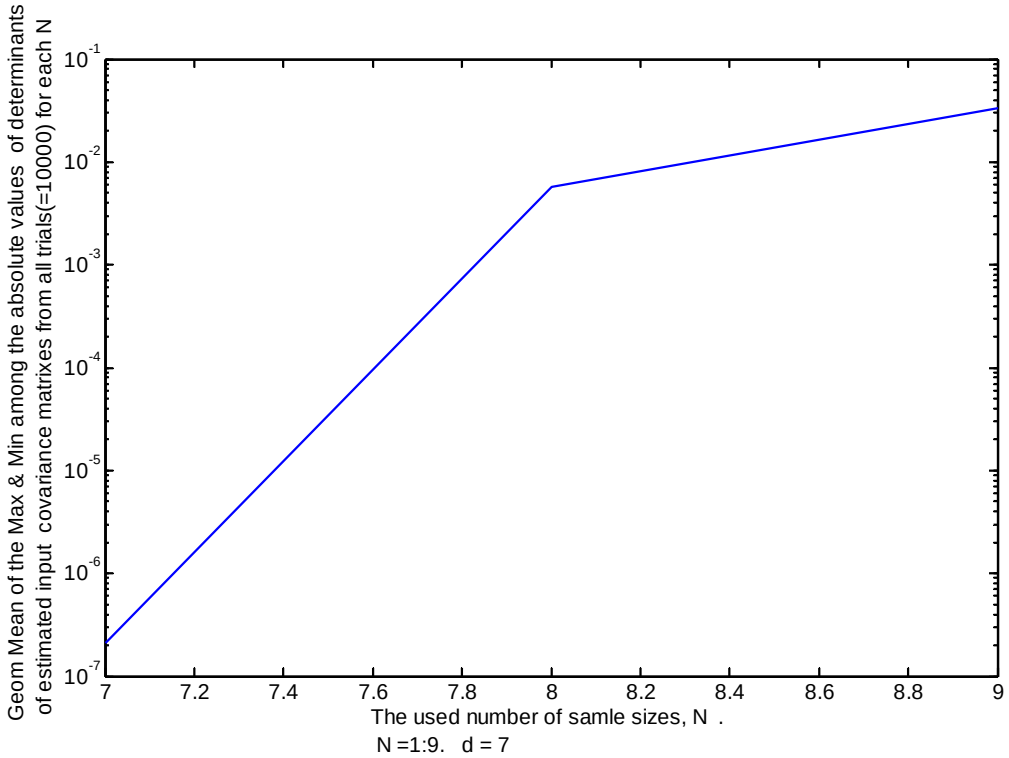

Figure 4.1: Showing the determinant values of the estimated input covariance matrix when the sample size value is near the number of dimension. As the determinant of a square matrix is a multiplicative property of the eigen values of the matrix, we took the geometric mean of the absolute maximum and absolute minimum values of the determinant of the (unregularized) estimated input covariance matrix taken out from all the trials for each sample size value. We made simulations for the sample size (N) values from 1 to 9. But the values before the sample size equals the model dimension (d =7), these determinant values are extremely low. Therefore, it is not shown in the simulation. We took the absolute values of the determinants as MATLAB sometimes gives negative valued determinant of a covariance matrix when it is almost singular. From this figure 4.1, we also see that the determinant values are too low when the sample size value (N) is equal or less than one more than the model dimension value (d) [i.e. for $N \leq d+1$] and thus it starts to increase afterwards, as we have discussed in section 4.1.1.

4.2.2 Simulation concerning the regularization of the estimated input covariance matrix

Simulation of the cost functions with unregularized estimated input covariance matrix:

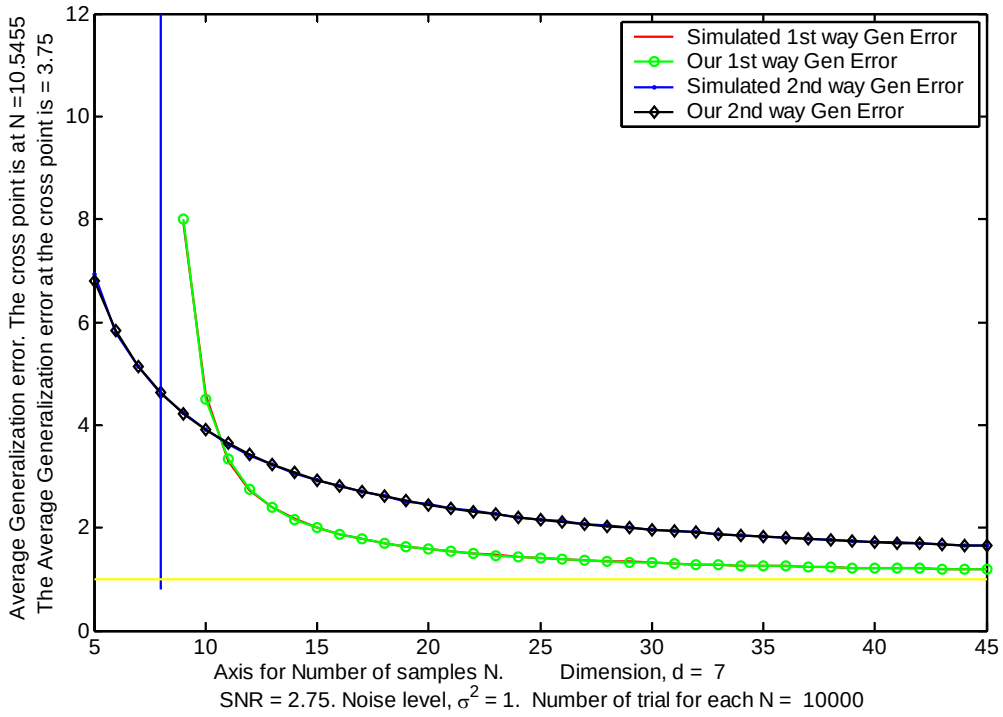

Figure 4.2: Showing the unboundness of the generalization error without using any regularization of the estimated input covariance matrix **A**. In figure, we see that for the 1st method (derived in chapter1) of generalization error, both our theoretical result's curve (green) and the simulated curve (red) becomes unbounded when the sample size value (N) is lesser or equal to one more than the model dimension value (d) [i.e. for $N \leq d+1$] and thus they start to give finite values afterwards, as we have discussed in section 4.1.1. The two perpendicular straight lines are the asymptotes of the two simulated curves obtained from the 1st and 2nd method. Same type of this figure 4.2 we used in chapter 3, Figure 3.3.

Simulation of the cost functions with regularized estimated input covariance matrix:

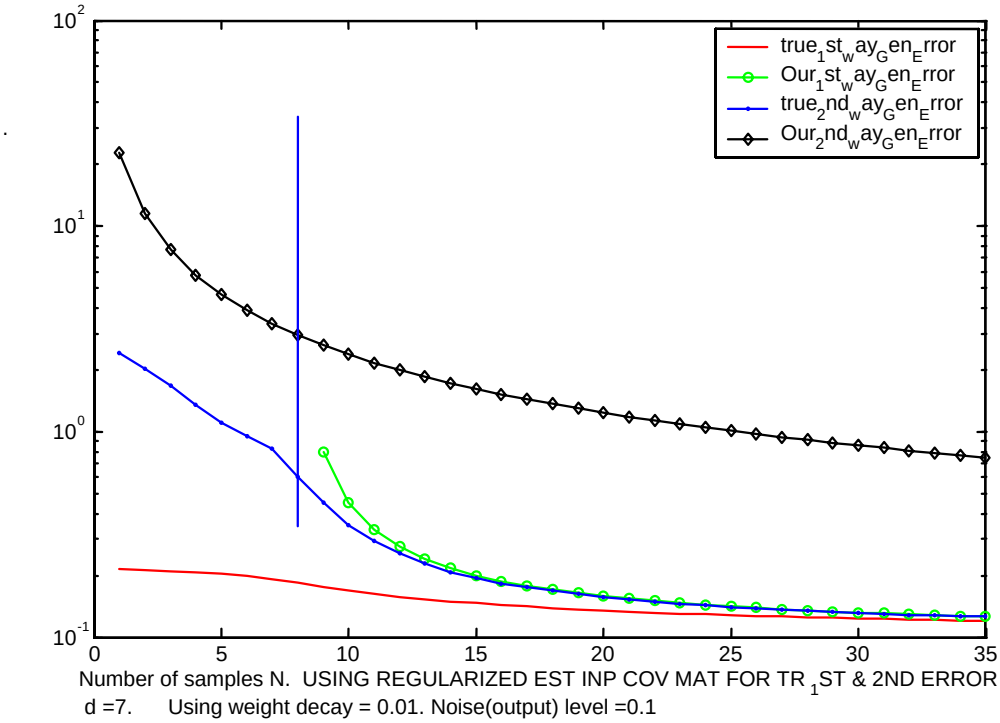

Figure 4.3: Showing the boundness of the generalization error after using regularization of the estimated input covariance matrix **A** by introducing a regularization parameter (or the so called 'weight decay') λ. After regularization, the new (regularized) estimated input covariance matrix becomes $\mathbf{A} + \lambda \mathbf{I}$. In figure, we see that for the 1st method (derived in chapter1) of generalization error, our theoretical result's curve (green) is still unbounded while $N \leq d + 1$ as it is a function of $N - d$. But the simulated curve (red) becomes bounded for all the valid sample size value (N), which is better than the case of Figure 4.2 above. This is the benefit of making regularization of the estimated input covariance matrix as discussed in section 4.1.2. In the same way, we see that the simulated generalization error curve (dotted blue) obtained from the 2nd method (derived in chapter 2) is bounded for all valid sample size value (N), which is better than the case of the same curve in Figure 4.2 (above). This benefit is obtained by regularization as we have mentioned once before. From the figure, we also see that our theoretical curve from the 2nd method is always bounded for all valid sample size as usual since it has no pole with respect to d.

4.2.3 Simulation concerning the values of regularization parameter (or 'weight decay')

From Figure 4.2 and Figure 4.3 above, we see that the weight decay parameter λ plays an important role in order to recover from the unbounded generalization error phenomenon. But there is no given fixed value for that. We will now try to see the influence of the values of this weight decay parameter by observing the variation in the generalized error function with the change in this parameter as.

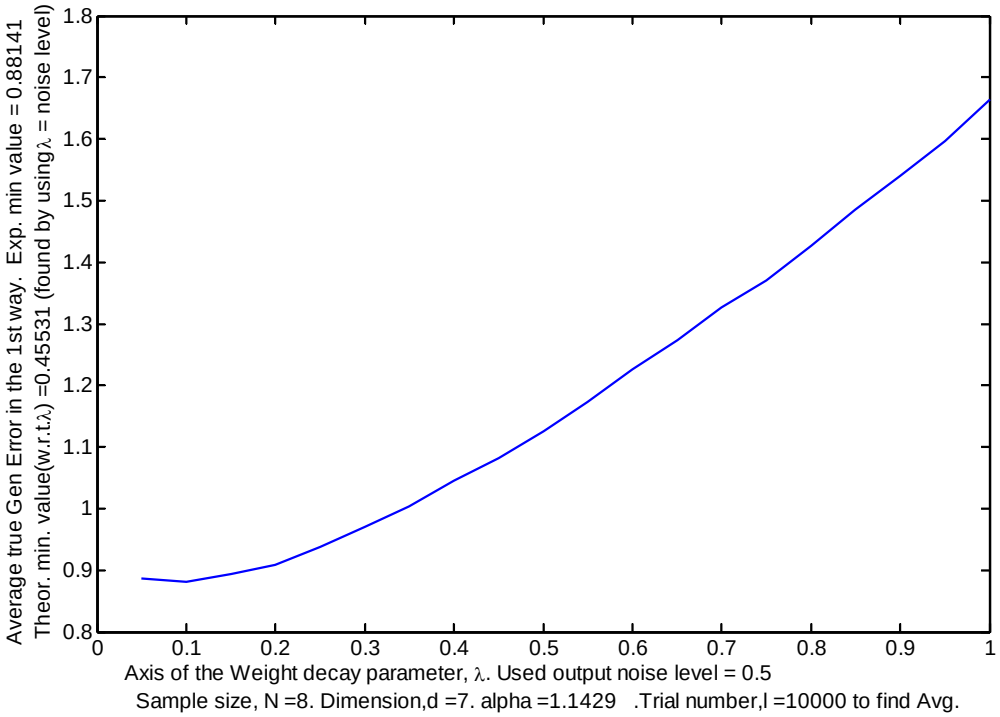

Figure 4.4: Figure showing the generalization error as a function of the regularization parameter (weight decay) λ. From figure we see that after certain value of λ, generalization error increases as λ increases. Therefore, this certain value could be thought as the optimum value of λ regarding our model and its assumptions. From [11] we can have a little idea about the optimum value of this λ and the minimum generalization error by using this value of λ. We made a theoretical calculation to approximate this minimum generalization value, which is called (in this figure at the Y-label) as the 'Theo. Min. value (wrt λ)' and was found as 0.4531 whereas the experimental minimum value, called (in this figure at the Y-label) as 'Exp. Min. value' and was found as 0.88141, which is 0.42831 from the theoretical one. This difference came due to the simulation technique and the issues regarding the model selection with the used assumptions in the model. A deep analysis considering all of these issues can give better approximation of the optimal λ.

4.3 Conclusion

In this chapter, we managed to talk about the necessity of estimated input covariance matrix regularization and its action (with merit and demerit). We were able to investigate the related issues through simulations. We tried to find the optimal value of the regularizer λ implementing the idea from [11]. We were able to get closer to the optimal value λ excluding the model specialty and used assumption.
A further deep analysis would take us nearer to the optimum value of λ.

Chapter 5: Conclusion

In this chapter we make a succinct summary of the whole project.

Organization of this chapter:
In 5.1, we discussed about the work done so far whereas we talk about the least further possible work in 5.2

5.1 So Far

This is a theoretical project. Works here are in principal to derive and analyze some crucial results in stochastic linear learning with a chaste authentication. A reasonable amount of work has also been done to compare the derived results with other standard results (if there exists any!). Main target was to obtain exact expressions especially, for the vital terms in stochastic learning algorithms and thus investigate them in several aspects. Therefore, in a short, the target was fulfilled with full satisfaction! But how is it so? Below traces the answer:

We, combining the statistical mechanical and other concepts, were able to derive the expressions of the exact training and test errors averages for a linear model [13], which are salient terms in learning theory in order to judge the quality of a model. These derivations were done using only three simple conditions:

i) The post training distribution is Gibbsian
ii) Noise and inputs are independent having the normal and multinormal distribution respectively
iii) Difference between the sample size's length and the model dimension's length is more than one.

And the results are mainly functions of three terms only: the sample size, model dimension and the associated additive noise level; but not the unknown input covariance.
Our results are consistent for all valid sample size range defeating the conventional results so far, which perform in limited cases. For example, the most frequently referred one is the Akaike's FPE [7] [8]. This is valid only for very large sample size. Our test error result easily passed the unsullied verification by using MATLAB simulation (with 10000 trials for each sample size value) [chapter 1, figure: 1.1].

In this thesis, we also derived the exact expression of the generalization error in Linear Regression model [20] with respect to the sample size, model dimension, associated additive noise level, true weight vector and the known input covariance matrix. In my knowledge, this is the first and only derived expression for the generalization error in that phase considering the whole machine learning area. We also made an immaculate MATLAB simulation regarding this result. The simulation result (with 10000 trials for each sample size value) proves the validity of our theorem [chapter 2, Fig: 2.1].

We detected the cross point of the two curves produced from the two exact expressions of the generalization errors mentioned above. This detection was done

after making a short analytic inspection about their properties. The cross point was found in the valid sample size, error domain. It was also shown that at the cross point, the generalization error is not an explicit function of the either sample size or the model dimension [chapter 3].

This thesis was also able to make a useful investigation for the case of unbounded generalization error by making regularization of the estimated covariance matrix [chapter 4].

It also finds mathematical explanations for the case of pole and singularity of the generalization error function [A.8].

5.2 Further work

Due to the time limitation of the project, it was not possible to go through the deepest detail in all the aspects. Thus, if there would be further work in this field, the following issues would be considered to focus first:

i) Finding the geometric interpretation behind the unboundness of the averaged generalization error $\overline{\epsilon_G}$ when the sample size length is one more than the length of the model dimension [related to chapter 1].

ii) Measuring the amount of penalty (for example, lost information) due to the regularization of the estimated input covariance matrix [related to chapter 4].

iii) Finding the optimum value of this regularization parameter that gives the least generalization error.

APPENDIX A (CALCULATIONS AND PROOFS)

A.1

PROOF OF RELATION (1.11)

Statement: $\in_T = -\frac{1}{N}\frac{\partial}{\partial\beta}\left[\ln Z_N(h,\beta)\right]_{h=0}$

Proof:
From relation (1.10) we have

$$Z_N(h,\beta) = \int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)$$

$$\Rightarrow \ln\ Z_N(h,\beta) = \ln\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)$$

$$\Rightarrow \frac{\partial}{\partial\beta}\left[\ln Z_N(h,\beta)\right] = \frac{\partial}{\partial\beta}\left[\ln\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)\right]$$

$$\Rightarrow \frac{\partial}{\partial\beta}\left[\ln Z_N(h,\beta)\right] = \frac{\frac{\partial}{\partial\beta}\left(\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)\right)}{\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)}$$

$$\Rightarrow \frac{\partial}{\partial\beta}\left[\ln Z_N(h,\beta)\right] = \frac{\int D\tilde{w}\left(-\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)}{\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)}$$

$$\Rightarrow \frac{\partial}{\partial\beta}\left[\ln Z_N(h,\beta)\right]_{h=0} = \frac{\int D\tilde{w}\left(-\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\right)}{\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\right)}$$

$$\Rightarrow \frac{\partial}{\partial\beta}\left[\ln Z_N(h,\beta)\right]_{h=0} = \frac{\int D\tilde{w}\left(-\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right) P_N(\tilde{w})\, Z_N}{\int D\tilde{w}\ P_N(\tilde{w})\, Z_N}$$

[In the above line, we have used relation (1.3), where we have

$$P_N(w) = Z_N^{-1}\exp\left(-\beta\sum_{\alpha=1}^{N} E^{\alpha}(w)\right)$$

$$\Rightarrow P_N(\tilde{w})\, Z_N = \exp\left(-\beta \sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right) \quad]$$

$$\Rightarrow \frac{\partial}{\partial \beta}\left[\ln Z_N(h,\beta)\right]_{h=0} = \frac{\int D\tilde{w}\left(-\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right) P_N(\tilde{w})}{\int D\tilde{w}\; P_N(\tilde{w})}$$

[As Z_N is (normalization) constant with respect to $\tilde{w}$ or w. Because from the expression $Z_N = \int D\tilde{w}\; \exp\left(-\beta \left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right) \right)$ we see that the variable $\tilde{w}$ or w is taken out by the integration.]

$$\Rightarrow \frac{\partial}{\partial \beta}\left[\ln Z_N(h,\beta)\right]_{h=0} = \int D\tilde{w}\left(-\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right) P_N(\tilde{w}) \quad [\text{As } \int D\tilde{w}\; P_N(\tilde{w}) = 1]$$

$$\Rightarrow -\frac{1}{N}\frac{\partial}{\partial \beta}\left[\ln Z_N(h,\beta)\right]_{h=0} = \int D\tilde{w}\left(\frac{1}{N}\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right) P_N(\tilde{w})$$

$$\Rightarrow -\frac{1}{N}\frac{\partial}{\partial \beta}\left[\ln Z_N(h,\beta)\right]_{h=0} = \in_T$$

[Using relation (1.7) $\Rightarrow \int D\tilde{w}\left(\frac{1}{N}\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right) P_N(\tilde{w}) = \in_T$]

[**Proved**]

A.2

PROOF OF RELATION (1.12)

Statement:

$$\in_G = \frac{1}{\beta^2}\sum_{j,j'=1}^{d} \Sigma_{jj'}\left[\frac{\partial^2}{\partial h_j \partial h_{j'}}\ln Z_N(h,\beta) + \left(\frac{\partial}{\partial h_j}\ln Z_N(h,\beta)\right)\left(\frac{\partial}{\partial h_{j'}}\ln Z_N(h,\beta)\right)\right]_{h=0} + \sigma_v^2$$

Proof:

From relation (1.10) we have

$$Z_N(h,\beta) = \int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)$$

$$\Rightarrow \ln\ Z_N(h,\beta) = \ln\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)$$

$$\Rightarrow \frac{\partial}{\partial h_j}\ln\ Z_N(h,\beta) = \frac{\dfrac{\partial}{\partial h_j}\left(\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)\right)}{\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)}$$

$$\Rightarrow \frac{\partial}{\partial h_j}\ln Z_N(h,\beta) = \frac{\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)\beta\,\tilde{w}_j}{Z_N(h,\beta)}$$

Similarly, we can get

$$\frac{\partial}{\partial h_{j'}}\ln Z_N(h,\beta) = \frac{\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)\beta\,\tilde{w}_{j'}}{Z_N(h,\beta)}$$

Then

$$\frac{\partial^2}{\partial h_j \partial h_{j'}}\ln Z_N(h,\beta) \quad = \quad \frac{\partial}{\partial h_j}\left[\frac{\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)\beta\,\tilde{w}_{j'}}{Z_N(h,\beta)}\right]$$

$$= \frac{(Z_N(h,\beta))\dfrac{\partial}{\partial h_j}\left[\int D\tilde{w}\exp\left(-\beta\left(\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})+\sum_{j=1}^{d}h_j\tilde{w}_j\right)\right)\beta\,\tilde{w}_{j'}\right]-\left(\int D\tilde{w}\exp\left(-\beta\left(\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})+\sum_{j=1}^{d}h_j\tilde{w}_j\right)\right)\beta\,\tilde{w}_{j'}\right)\dfrac{\partial}{\partial h_j}(Z_N(h}{(Z_N(h,\beta))^2}$$

$$= \frac{(Z_N(h,\beta))\left[\int D\tilde{w}\exp\left(-\beta\left(\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})+\sum_{j=1}^{d}h_j\tilde{w}_j\right)\right)\tilde{w}_{j'}\tilde{w}_j\,\beta^2\right]}{(Z_N(h,\beta))^2}$$

$$- \frac{\left(\int D\tilde{w}\exp\left(-\beta\left(\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})+\sum_{j=1}^{d}h_j\tilde{w}_j\right)\right)\tilde{w}_{j'}\,\beta\right)\int D\tilde{w}\exp\left(-\beta\left(\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})+\sum_{j=1}^{d}h_j\tilde{w}_j\right)\right)\tilde{w}_j\,\beta}{(Z_N(h,\beta))^2}$$

Then $\left[\dfrac{\partial^2}{\partial h_j\partial h_{j'}}\ln Z_N(h,\beta)\right]_{h=0}$

$$= \frac{\left[\int D\tilde{w}\ \exp\left(-\beta\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)\right]\left[\int D\tilde{w}\exp\left(-\beta\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)\tilde{w}_{j'}\tilde{w}_j\,\beta^2\right]}{\left[\int D\tilde{w}\ \exp\left(-\beta\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)\right]^2}$$

$$- \frac{\beta^2\left(\int D\tilde{w}\exp\left(-\beta\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)\tilde{w}_{j'}\right)\left(\int D\tilde{w}\exp\left(-\beta\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)\tilde{w}_j\right)}{\left[\int D\tilde{w}\ \exp\left(-\beta\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)\right]^2}$$

$$= \frac{\beta^2\left[\int D\tilde{w}\ P_N(\tilde{w})Z_N\right]\left[\int D\tilde{w}\ P_N(\tilde{w})Z_N\tilde{w}_{j'}\tilde{w}_j\right]-\beta^2\left(\int D\tilde{w}\ P_N(\tilde{w})Z_N\,\tilde{w}_{j'}\right)\left(\int D\tilde{w}\ P_N(\tilde{w})Z_N\,\tilde{w}_j\right)}{\left[\int D\tilde{w}\ P_N(\tilde{w})Z_N\right]^2}$$

[As Z_N is constant with respect to $\tilde{w}$ or w]

$$= \frac{\beta^2\left[\int D\tilde{w}\ P_N(\tilde{w})\right]\left[\int D\tilde{w}\ P_N(\tilde{w})\tilde{w}_{j'}\tilde{w}_j\right]-\beta^2\left(\int D\tilde{w}\ P_N(\tilde{w})\ \tilde{w}_{j'}\right)\left(\int D\tilde{w}\ P_N(\tilde{w})\ \tilde{w}_j\right)}{\left[\int D\tilde{w}\ P_N(\tilde{w})\right]^2}$$

$$\left[\frac{\partial^2}{\partial h_j \partial h_{j'}} \ln Z_N(h,\beta)\right]_{h=0}$$

$$= \beta^2 \left[\int D\tilde{w}\ P_N(\tilde{w})\, \tilde{w}_{j'} \tilde{w}_j - \left(\int D\tilde{w}\ P_N(\tilde{w})\ \tilde{w}_{j'}\right)\ \left(\int D\tilde{w}\ P_N(\tilde{w})\ \tilde{w}_j\right)\right]$$

Also,

$$\Rightarrow \left[\frac{\partial}{\partial h_j} \ln Z_N(h,\beta)\right]_{h=0} = \left[\frac{\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)\beta\, \tilde{w}_j}{Z_N(h,\beta)}\right]_{h=0}$$

$$= \frac{\int D\tilde{w}\ \exp\left(-\beta \sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\beta\, \tilde{w}_j}{\int D\tilde{w}\ \exp\left(-\beta \sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)}$$

$$= \frac{\beta \int D\tilde{w}\ P_N(\tilde{w}) Z_N\ \tilde{w}_j}{\int D\tilde{w} P_N(\tilde{w}) Z_N} = \frac{\beta \int D\tilde{w}\ P_N(\tilde{w})\ \tilde{w}_j}{\int D\tilde{w} P_N(\tilde{w})} = \beta \int D\tilde{w}\ P_N(\tilde{w})\ \tilde{w}_j$$

And similarly,

$$\left[\frac{\partial}{\partial h_{j'}} \ln Z_N(h,\beta)\right]_{h=0} = \beta \int D\tilde{w}\ P_N(\tilde{w})\ \tilde{w}_{j'}$$

Using the above quantities in

$$\frac{1}{\beta^2} \sum_{j,j'=1}^{d} \Sigma_{jj'} \left[\frac{\partial^2}{\partial h_j \partial h_{j'}} \ln Z_N(h,\beta) + \left(\frac{\partial}{\partial h_j} \ln Z_N(h,\beta)\right)\left(\frac{\partial}{\partial h_{j'}} \ln Z_N(h,\beta)\right)\right]_{h=0} + \sigma_\nu^2$$

we get,

$$\frac{1}{\beta^2} \sum_{j,j'=1}^{d} \Sigma_{jj'} \left[\begin{array}{l} \left(\beta^2 \left[\int D\tilde{w}\ P_N(\tilde{w}) \tilde{w}_{j'} \tilde{w}_j - \left(\int D\tilde{w}\ P_N(\tilde{w})\ \tilde{w}_{j'}\right)\ \left(\int D\tilde{w}\ P_N(\tilde{w})\ \tilde{w}_j\right)\right]\right) \\ +\left(\beta \int D\tilde{w}\ P_N(\tilde{w})\ \tilde{w}_j\right)\left(\beta \int D\tilde{w}\ P_N(\tilde{w})\ \tilde{w}_{j'}\right) \end{array}\right] + \sigma_\nu^2$$

$$= \frac{1}{\beta^2}\sum_{j,j'=1}^{d} \Sigma_{jj'}\left[\beta^2 \int D\tilde{w}\ P_N(\tilde{w})\,\tilde{w}_{j'}\tilde{w}_j\right]+\sigma_\nu^2$$

$$= \int D\tilde{w}\ P_N(\tilde{w})\ \sum_{j,j'=1}^{d} \Sigma_{jj'}\tilde{w}_{j'}\tilde{w}_j+\sigma_\nu^2$$

$$= \int D\tilde{w}\ P_N(\tilde{w})\ \sum_{j,j'=1}^{d} \Sigma_{jj'}\tilde{w}_{j'}\tilde{w}_j+\sigma_\nu^2 \int D\tilde{w}\ P_N(\tilde{w}) \quad [\text{As } \int D\tilde{w}\ P_N(\tilde{w}) = 1]$$

$$= \int D\tilde{w}\ P_N(\tilde{w})\left(\sum_{j,j'=1}^{d} \Sigma_{jj'}\,\tilde{w}_{j'}\tilde{w}_j+\sigma_\nu^2\right) \quad [\text{As } \sigma_\nu^2 \text{ is constant}]$$

$= \in_G$ [Using relation (1.9)]

$$\Rightarrow \in_G = \frac{1}{\beta^2}\sum_{j,j'=1}^{d} \Sigma_{jj'}\left[\frac{\partial^2}{\partial h_j \partial h_{j'}}\ln Z_N(h,\beta)+\left(\frac{\partial}{\partial h_j}\ln Z_N(h,\beta)\right)\left(\frac{\partial}{\partial h_{j'}}\ln Z_N(h,\beta)\right)\right]_{h=0}+\sigma_\nu^2$$

[Proved]

A.3

PROOF OF RELATION (1.19-EXTRA)

Statement:

$$\frac{1}{N}\frac{\partial \in_T}{\partial \beta} \;=\; -\left\langle \;\left[\textit{mean training error} - \left\langle \textit{mean training error} \right\rangle \right]^2 \;\right\rangle \;\le\; 0$$

Proof:

$$\frac{1}{N}\frac{\partial \in_T}{\partial \beta}$$

$$= \frac{1}{N}\frac{\partial}{\partial \beta}\left(-\frac{1}{N}\frac{\partial}{\partial \beta}\left[\ln Z_N(h,\beta)\right]_{h=0} \right) \quad \text{[Using the statement of relation (1.11 or } \mathbf{A.1}\text{)}$$

$$= -\frac{1}{N^2}\frac{\partial}{\partial \beta}\left(\frac{\partial}{\partial \beta}\left[\ln Z_N(h,\beta)\right]_{h=0} \right)$$

$$= -\frac{1}{N^2}\frac{\partial}{\partial \beta}\left(\frac{\int D\tilde{w}\left(-\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right)\exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right)\right)}{\int D\tilde{w}\,\exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right)\right)} \right)$$

[Using the sixth line of the proof in **A.1**]

$$= -\frac{1}{N^2}\left(\frac{\left[\int D\tilde{w}\,\exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right)\right)\right]\frac{\partial}{\partial \beta}\left[\int D\tilde{w}\left(-\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right)\exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right)\right)\right]}{\left[\int D\tilde{w}\,\exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right)\right)\right]^2} \right.$$

$$\left. - \frac{\left[\int D\tilde{w}\left(-\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right)\exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right)\right)\right]\frac{\partial}{\partial \beta}\left[\int D\tilde{w}\,\exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right)\right)\right]}{\left[\int D\tilde{w}\,\exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) \right)\right)\right]^2} \right)$$

$$= -\frac{1}{N^2}\left(\frac{\left[\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\right)\right]\left[\int D\tilde{w}\left(-\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)^2 \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\right)\right] - \left[\int D\tilde{w}\left(-\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\right)\right]\left[\int D\tilde{w}\left(-\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\right)\right]}{\left[\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\right)\right]^2} \right)$$

$$= -\frac{1}{N^2}\left(\frac{\left[\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\right)\right]\left[\int D\tilde{w}\left(-\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)^2 \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\right)\right] - \left[\int D\tilde{w}\left(-\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\right)\right]^2}{\left[\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)\right)\right]^2} \right)$$

$$= -\frac{1}{N^2}\left(\frac{\left[\int D\tilde{w}\ P_N(\tilde{w}) Z_N\right]\left[\int D\tilde{w}\left(-\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right)^2 P_N(\tilde{w}) Z_N\right] - \left[\int D\tilde{w}\left(-\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right) P_N(\tilde{w}) Z_N\right]^2}{\left[\int D\tilde{w}\ P_N(\tilde{w}) Z_N\right]^2} \right)$$

[In the above line, we have used relation (1.3), where we have

$$P_N(w) = Z_N^{-1} \exp\left(-\beta \sum_{\alpha=1}^{N} E^{\alpha}(w)\right) \qquad \Rightarrow P_N(\tilde{w}) Z_N = \exp\left(-\beta \sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w})\right) \quad]$$

$$=$$

$$-\frac{1}{N^2}Z_N^2\left(\frac{\left[\int D\tilde{w}\ P_N(\tilde{w})\right]\left[\int D\tilde{w}\left(-\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)^2 P_N(\tilde{w})\right]-\left[\int D\tilde{w}\left(-\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)P_N(\tilde{w})\right]^2}{Z_N^2\left[\int D\tilde{w}\ P_N(\tilde{w})\right]^2}\right)$$

[Since here Z_N is constant with respect to $\tilde{w}$]

$$=-\frac{1}{N^2}\left(\left[\int D\tilde{w}\left(-\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)^2 P_N(\tilde{w})\right]-\left[\int D\tilde{w}\left(-\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)P_N(\tilde{w})\right]^2\right)$$

[As $\int D\tilde{w}\ P_N(\tilde{w})=1$]

$$=-\left(\left[\int D\tilde{w}\left(-\frac{1}{N}\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)^2 P_N(\tilde{w})\right]-\left[\int D\tilde{w}\left(-\frac{1}{N}\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)P_N(\tilde{w})\right]^2\right)$$

$$=-\left(\left\langle\left(-\frac{1}{N}\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)^2\right\rangle-\left[\left\langle\left(-\frac{1}{N}\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)\right\rangle\right]^2\right)$$

$$=-\left(\left\langle\left(\frac{1}{N}\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)^2\right\rangle-\left[\left\langle\left(\frac{1}{N}\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right)\right\rangle\right]^2\right)$$

$$\Rightarrow\frac{1}{N}\frac{\partial\in_T}{\partial\beta}=-\left\langle\left(\left[\frac{1}{N}\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right]-\left\langle\left[\frac{1}{N}\sum_{\alpha=1}^{N}E^{\alpha}(\tilde{w})\right]\right\rangle\right)^2\right\rangle$$

$$\Rightarrow\frac{1}{N}\frac{\partial\in_T}{\partial\beta}=$$

$$-\left\langle\left[\text{random mean train error}-\left\langle\text{random mean train error}\right\rangle\right]^2\right\rangle$$

$$\Rightarrow\frac{1}{N}\frac{\partial\in_T}{\partial\beta}\leq 0$$

[Proved]

A.4

PROOF OF RELATION (1.16)

Statement:

$$\ln Z_N(h,\beta) = -\frac{1}{2}\ln\left(\det(\beta\mathbf{A})\pi^{-N}\right) + \frac{\beta}{4}\sum_{j,j'}^{d} a_j a_{j'}\left(\mathbf{A}^{-1}\right)_{jj'} - \beta a_0$$

Proof:

From relation (1.10), we have

$$Z_N(h,\beta) = \int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N} E^{\alpha}(\tilde{w}) + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)$$

$$\Rightarrow Z_N(h,\beta) = \int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N}\left(\sum_{j=1}^{d}\tilde{w}_j x_j^{\alpha} - \nu\right)^2 + \sum_{j=1}^{d} h_j \tilde{w}_j\right)\right)$$

$$\Rightarrow Z_N(h,\beta) = \int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N}\left(\sum_{j,j'=1}^{d}\tilde{w}_j\tilde{w}_{j'}x_j^{\alpha}x_{j'}^{\alpha} - 2\sum_{j=1}^{d}\tilde{w}_j x_j^{\alpha}\nu^{\alpha} + \left(\nu^{\alpha}\right)^2\right) + \sum_{j=1}^{d} h_j\tilde{w}_j\right)\right)$$

$$\Rightarrow Z_N(h,\beta) = \exp\left(-\beta\sum_{\alpha=1}^{N}\left(\nu^{\alpha}\right)^2\right)\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N}\sum_{j,j'=1}^{d}\tilde{w}_j\tilde{w}_{j'}x_j^{\alpha}x_{j'}^{\alpha} - 2\sum_{\alpha=1}^{N}\sum_{j=1}^{d}\tilde{w}_j x_j^{\alpha}\nu^{\alpha} + \sum_{j=1}^{d} h_j\tilde{w}_j\right)\right)$$

$$\Rightarrow Z_N(h,\beta) = \exp\left(-\beta\sum_{\alpha=1}^{N}\left(\nu^{\alpha}\right)^2\right)\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{\alpha=1}^{N}\sum_{j,j'=1}^{d}\tilde{w}_j\tilde{w}_{j'}x_j^{\alpha}x_{j'}^{\alpha} + \sum_{j=1}^{d}\tilde{w}_j\left(h_j - 2\sum_{\alpha=1}^{N}x_j^{\alpha}\nu^{\alpha}\right)\right)\right)$$

$$\Rightarrow Z_N(h,\beta) = \exp\left(-\beta\sum_{\alpha=1}^{N}\left(\nu^{\alpha}\right)^2\right)\int D\tilde{w}\ \exp\left(-\beta\left(\sum_{j,j'=1}^{d}\tilde{w}_{j'}\left(\sum_{\alpha=1}^{N}x_j^{\alpha}x_{j'}^{\alpha}\right)\tilde{w}_j + \sum_{j=1}^{d}\left(h_j - 2\sum_{\alpha=1}^{N}x_j^{\alpha}\nu^{\alpha}\right)\tilde{w}_j\right)\right)$$

If we define

$$A_{jj'} = \sum_{\alpha=1}^{N} x_j^{\alpha}x_{j'}^{\alpha} \qquad \text{[Estimated input covariance matrix]}$$

$$a_j = h_j - 2\sum_{\alpha=1}^{N} x_j^{\alpha}\nu^{\alpha}$$

$$a_0 = \sum_{\alpha=1}^{N}\left(\nu^{\alpha}\right)^2$$

we get

$$\Rightarrow Z_N(h,\beta) = \exp(-\beta a_0) \int D\tilde{w} \exp\left(-\beta \left(\sum_{j,j'=1}^{d} \tilde{w}_{j'} \mathbf{A}_{\mathbf{jj'}} \tilde{w}_j + \sum_{j=1}^{d} a_j \tilde{w}_j \right) \right)$$

$$\Rightarrow Z_N(h,\beta) = \exp(-\beta a_0) \int D\tilde{w} \exp\left(-\frac{1}{2} \sum_{j,j'=1}^{d} \tilde{w}_{j'} (2\beta \mathbf{A}_{\mathbf{jj'}}) \tilde{w}_j + \sum_{j=1}^{d} (-\beta a_j) \tilde{w}_j \right)$$

Comparing this expression with (B.15) (page-446) of Bishop's book ("Neural Networks for Pattern Recognition") and then following the result of (B.22) of the same source, we get

$$\Rightarrow Z_N(h,\beta) = \exp(-\beta a_0) (2\pi)^{\frac{d}{2}} (\det(2\beta \mathbf{A}))^{-\frac{1}{2}} \exp\left(\frac{1}{2} (-\beta a_j)(2\beta \mathbf{A})^{-1} (-\beta a_j)^T \right)$$

[**A** is non-singular]

$$\Rightarrow Z_N(h,\beta) = \exp(-\beta a_0) (2)^{\frac{d}{2}} (\pi)^{\frac{d}{2}} (2)^{-\frac{d}{2}} (\det(\beta \mathbf{A}))^{-\frac{1}{2}} \exp\left(\frac{1}{2} \frac{\beta}{2} (a_j)(\mathbf{A})^{-1} (a_j)^T \right)$$

$$\Rightarrow Z_N(h,\beta) = \exp(-\beta a_0) \left((\pi)^{-d} (\det(\beta \mathbf{A})) \right)^{-\frac{1}{2}} \exp\left(\frac{\beta}{4} \sum_{j,j'}^{d} a_j a_{j'} (\mathbf{A}^{-1})_{jj'} \right)$$

Taking logarithm (natural) on the both sides, we get

$$\ln Z_N(h,\beta) = -\frac{1}{2} \ln\left(\det(\beta \mathbf{A}) \pi^{-N} \right) + \frac{\beta}{4} \sum_{j,j'}^{d} a_j a_{j'} (\mathbf{A}^{-1})_{jj'} - \beta a_0$$

[**Proved**]

A.5

PROOF OF RELATION (1.18)

Statement:

$$\in_G = \sum_{k'k''}^{d} \Sigma_{k'k''} \left[\frac{1}{2\beta} \left(\mathbf{A}^{-1}\right)_{k'k''} + \sum_{j'j''}^{d} \sum_{\alpha'\alpha''}^{N} x_{j'}^{\alpha'} x_{j''}^{\alpha''} \nu^{\alpha'} \nu^{\alpha''} \left(\mathbf{A}^{-1}\right)_{k'j'} \left(\mathbf{A}^{-1}\right)_{k''j''} \right] + \sigma_\nu^2$$

Proof:
We get (1.18) by using (1.16) in (1.12). In order to do so, first we find the following:

$$\frac{\partial}{\partial h_j} \ln Z_N(h,\beta) = \frac{\partial}{\partial h_j} \left[-\frac{1}{2} \ln\left(\det(\beta\mathbf{A})\pi^{-d}\right) + \frac{\beta}{4} \sum_{j,j'}^{d} a_j a_{j'} \left(\mathbf{A}^{-1}\right)_{jj'} - \beta a_0 \right]$$

$$\Rightarrow \frac{\partial}{\partial h_j} \ln Z_N(h,\beta) = \frac{\partial}{\partial h_j} \left[\frac{\beta}{4} \sum_{j,j'}^{d} a_j a_{j'} \left(\mathbf{A}^{-1}\right)_{jj'} \right] \quad \text{[Since } \mathbf{A}, a_0, \beta \text{ are not functions}$$

of h]

$$\Rightarrow \frac{\partial}{\partial h_j} \ln Z_N(h,\beta) = \frac{\beta}{4} \frac{\partial}{\partial h_j} \left[\sum_{j,j'}^{d} \left\{ \left(h_j - 2\sum_{\alpha=1}^{N} x_j^\alpha \nu^\alpha \right) \left(h_{j'} - 2\sum_{\alpha=1}^{N} x_{j'}^\alpha \nu^\alpha \right) \left(\mathbf{A}^{-1}\right)_{jj'} \right\} \right]$$

$$\Rightarrow \frac{\partial}{\partial h_j} \ln Z_N(h,\beta) = \frac{\beta}{4} \frac{\partial}{\partial h_j} \left[\sum_{j,j'}^{d} \left\{ \left(h_j h_{j'} - 2\sum_{\alpha=1}^{N} x_{j'}^\alpha \nu^\alpha h_j - 2\sum_{\alpha=1}^{N} x_j^\alpha \nu^\alpha h_{j'} + 4 \sum_{\alpha,\alpha'=1}^{N} x_j^\alpha x_{j'}^{\alpha'} \nu^\alpha \nu^{\alpha'} \right) \left(\mathbf{A}^{-1}\right)_{jj'} \right\} \right]$$

$$= \frac{\beta}{4} \left[\frac{\partial}{\partial h_j} \left(\left(\sum_{j=1}^{d} h_j \right) \left(\sum_{j'=1}^{d} h_{j'} \right) \right) \left(\mathbf{A}^{-1}\right)_{jj'} - 2\sum_{\alpha=1}^{N} \sum_{j'=1}^{d} \left(\frac{\partial}{\partial h_j} \left(\sum_{j=1}^{N} x_{j'}^\alpha \nu^\alpha h_j \right) \right) \left(\mathbf{A}^{-1}\right)_{jj'} - 2\sum_{\alpha=1}^{N} \sum_{j=1}^{d} \left(\frac{\partial}{\partial h_j} \left(\sum_{j'=1}^{d} x_j^\alpha \nu^\alpha h_{j'} \right) \right) \left(\mathbf{A}^{-1}\right)_{jj'} \right]$$

$$= \frac{\beta}{4} \left[\left(\left(\sum_{j=1}^{d} h_j \right) \delta_{jj'} + \left(\sum_{j'=1}^{d} h_{j'} \right) \delta_{jj} \right) \left(\mathbf{A}^{-1}\right)_{jj'} - 2\sum_{\alpha=1}^{N} \sum_{j'=1}^{d} x_{j'}^\alpha \nu^\alpha \left(\mathbf{A}^{-1}\right)_{jj'} - 2\sum_{\alpha=1}^{N} \sum_{j=1}^{d} x_j^\alpha \nu^\alpha \left(\mathbf{A}^{-1}\right)_{jj'} \right]$$

$$= \frac{\beta}{4} \left[\left(\left(\sum_{j=1}^{d} h_j \right) \delta_{jj'} + \left(\sum_{j'=1}^{d} h_{j'} \right) \right) \left(\mathbf{A}^{-1}\right)_{jj'} - 2\sum_{\alpha=1}^{N} \sum_{j'=1}^{d} x_{j'}^\alpha \nu^\alpha \left(\mathbf{A}^{-1}\right)_{jj'} - 2\sum_{\alpha=1}^{N} \sum_{j'=1}^{d} x_{j'}^\alpha \nu^\alpha \left(\mathbf{A}^{-1}\right)_{j'j} \right]$$

[Interchanging between the suffixes j *and* j' in the last

term]

$$\Rightarrow \frac{\partial}{\partial h_j}\ln Z_N(h,\beta) = \frac{\beta}{4}\left[\left(\left(\sum_{j=1}^{d} h_j\right)\delta_{jj'} + \left(\sum_{j'=1}^{d} h_{j'}\right)\right)\left(\mathbf{A}^{-1}\right)_{jj'} - 4\sum_{\alpha=1}^{N}\sum_{j'=1}^{d} x_{j'}^{\alpha}\nu^{\alpha}\left(\mathbf{A}^{-1}\right)_{jj'}\right]$$

[Since, from (1.13) we see that $A_{jj'} = A_{j'j}$]

Similarly, we can get

$$\frac{\partial}{\partial h_{j'}}\ln Z_N(h,\beta) = \frac{\beta}{4}\left[\left(\left(\sum_{j'=1}^{d} h_{j'}\right)\delta_{j'j} + \left(\sum_{j=1}^{d} h_j\right)\right)\left(\mathbf{A}^{-1}\right)_{j'j} - 4\sum_{\alpha=1}^{N}\sum_{j=1}^{d} x_j^{\alpha}\nu^{\alpha}\left(\mathbf{A}^{-1}\right)_{j'j}\right]$$

$$\Rightarrow \frac{\partial}{\partial h_{j'}}\ln Z_N(h,\beta) = \frac{\beta}{4}\left[\left(\left(\sum_{j'=1}^{d} h_{j'}\right)\delta_{j'j} + \left(\sum_{j=1}^{d} h_j\right)\right)\left(\mathbf{A}^{-1}\right)_{jj'} - 4\sum_{\alpha=1}^{N}\sum_{j=1}^{d} x_j^{\alpha}\nu^{\alpha}\left(\mathbf{A}^{-1}\right)_{jj'}\right]$$

Then

$$\frac{\partial^2}{\partial h_j\,\partial h_{j'}}\ln Z_N(h,\beta) = \frac{\partial}{\partial h_j}\left(\frac{\partial}{\partial h_{j'}}\ln Z_N(h,\beta)\right)$$

$$= \frac{\beta}{4}\,\frac{\partial}{\partial h_j}\left[\left(\left(\sum_{j'=1}^{d} h_{j'}\right)\delta_{j'j} + \left(\sum_{j=1}^{d} h_j\right)\right)\left(\mathbf{A}^{-1}\right)_{jj'} - 4\sum_{\alpha=1}^{N}\sum_{j=1}^{d} x_j^{\alpha}\nu^{\alpha}\left(\mathbf{A}^{-1}\right)_{jj'}\right]$$

$$= \frac{\beta}{4}\left[\frac{\partial}{\partial h_j}\left(\left(\sum_{j'=1}^{d} h_{j'}\right)\delta_{j'j} + \left(\sum_{j=1}^{d} h_j\right)\right)\left(\mathbf{A}^{-1}\right)_{jj'}\right]$$

$$= \frac{\beta}{4}\left[\left(1.\delta_{jj} + 1\right)\left(\mathbf{A}^{-1}\right)_{jj'}\right]$$

$$\Rightarrow \frac{\partial^2}{\partial h_j\,\partial h_{j'}}\ln Z_N(h,\beta) = \frac{\beta}{2}\left(\mathbf{A}^{-1}\right)_{jj'}$$

Now re-writing (1.12) we get

$$\in_G = \frac{1}{\beta^2}\sum_{j,j'=1}^{d}\Sigma_{jj'}\left[\frac{\partial^2}{\partial h_j\partial h_{j'}}\ln Z_N(h,\beta) + \left(\frac{\partial}{\partial h_j}\ln Z_N(h,\beta)\right)\left(\frac{\partial}{\partial h_{j'}}\ln Z_N(h,\beta)\right)\right]_{h=0} + \sigma_\nu^2$$

Using the above expressions, we find

$$\in_G$$

$$= \frac{1}{\beta^2} \sum_{j,j'=1}^{d} \Sigma_{jj'} \left[\begin{array}{c} \frac{\beta}{2}\left(\mathbf{A}^{-1}\right)_{jj'} + \left(\frac{\beta}{4}\left[\left(\left(\sum_{j=1}^{d} h_j \right) \delta_{jj'} + \left(\sum_{j'=1}^{d} h_{j'} \right) \right) \left(\mathbf{A}^{-1}\right)_{jj'} - 4 \sum_{\alpha=1}^{N} \sum_{j'=1}^{d} x_{j'}^{\alpha} \nu^{\alpha} \left(\mathbf{A}^{-1}\right)_{jj'} \right] \right) \\ \left(\frac{\beta}{4}\left[\left(\left(\sum_{j'=1}^{d} h_{j'} \right) \delta_{j'j} + \left(\sum_{j=1}^{d} h_j \right) \right) \left(\mathbf{A}^{-1}\right)_{jj'} - 4 \sum_{\alpha=1}^{N} \sum_{j=1}^{d} x_j^{\alpha} \nu^{\alpha} \left(\mathbf{A}^{-1}\right)_{jj'} \right] \right) \end{array} \right]_{h=0} +$$

$$\Rightarrow \in_G = \frac{1}{\beta^2} \sum_{j,j'=1}^{d} \Sigma_{jj'} \left[\frac{\beta}{2}\left(\mathbf{A}^{-1}\right)_{jj'} + \left(\frac{\beta}{4}\left[-4 \sum_{\alpha=1}^{N} \sum_{j'=1}^{d} x_{j'}^{\alpha} \nu^{\alpha} \left(\mathbf{A}^{-1}\right)_{jj'} \right] \right) \left(\frac{\beta}{4}\left[-4 \sum_{\alpha=1}^{N} \sum_{j=1}^{d} x_j^{\alpha} \nu^{\alpha} \left(\mathbf{A}^{-1}\right)_{jj'} \right] \right) \right] + \sigma_{\nu}^2$$

$$\Rightarrow \in_G = \sum_{k'k''}^{d} \Sigma_{k'k''} \left[\frac{1}{2\beta} \left(\mathbf{A}^{-1}\right)_{k'k''} + \sum_{j'j''}^{d} \sum_{\alpha'\alpha''}^{N} x_{j'}^{\alpha'} x_{j''}^{\alpha''} \nu^{\alpha'} \nu^{\alpha''} \left(\mathbf{A}^{-1}\right)_{k'j'} \left(\mathbf{A}^{-1}\right)_{k''j''} \right] + \sigma_{\nu}^2$$

[Proved]

A.6

PROOF OF RELATION (2.2)

Statement:

$$\hat{\mathbf{w}}(\mathbf{D}) \;=\; \mathbf{A}^{-1}\mathbf{a}$$

With $\mathbf{a} = \frac{1}{N}\sum_{n=1}^{N}\left[\mathbf{x}_n\ y_n\right]$ and $\mathbf{A} = \frac{1}{N}\left[\sum_{n=1}^{N}\mathbf{x}_n\mathbf{x}_n^T\right]$

Proof:

We have,

$y_n = \mathbf{w_0} \bullet \mathbf{x_n} + \varepsilon_n$

$\Rightarrow y_n = \mathbf{x}_n^T\mathbf{w_0} + \varepsilon_n$

Then the (error)^2 from the n-th relation (input-output map),

$$R_n = (y_n - \mathbf{x}_n^T\mathbf{w_0})^2$$

and the Residual Sum of Squares (RSS) can be calculated as

$$\text{RSS} \;=\; \sum_{n=1}^{N} R_n$$

$$\Rightarrow \text{RSS} \;=\; \sum_{n=1}^{N} (y_n - \mathbf{x}_n^T\mathbf{w_0})^2$$

We will be looking for a set of $\mathbf{w}_{0j}$; $j = 1,2,...,d$ that minimizes RSS. This vector $\mathbf{w}_{0j}$ is called the OLS (Ordinary Least Square) estimate of $\mathbf{w}_0$ and will be denoted by $\hat{\mathbf{w}}(\mathbf{D})$.

We find the OLS estimate of $\mathbf{w}_0$ by setting

$$\frac{\partial}{\partial \mathbf{w}_0}(RSS) \;= 0$$

[It comes as $\sum_{n=1}^{N} \frac{\partial R_n}{\partial \mathbf{w}_0} = 0$

$$\Rightarrow \frac{\partial}{\partial \mathbf{w_0}}\left(\sum_{n=1}^{N} R_n\right) = 0$$

$$\Rightarrow \frac{\partial}{\partial \mathbf{w_0}}(RSS) = 0\,]$$

$$\Rightarrow \frac{\partial}{\partial \mathbf{w}_0}\left(\sum_{n=1}^{N} (y_n - \mathbf{x}_n^T \mathbf{w_0})^2\right) = 0$$

$$\Rightarrow \frac{\partial}{\partial \mathbf{w}_0}\left(\sum_{n=1}^{N} (y_n - \mathbf{x}_n^T \mathbf{w_0})^T (y_n - \mathbf{x}_n^T \mathbf{w_0})\right) = 0$$

$$\Rightarrow \frac{\partial}{\partial \mathbf{w}_0}\left(\sum_{n=1}^{N} (y_n^T y_n - \mathbf{w}_0^T \mathbf{x}_n y_n - y_n^T \mathbf{x}_n^T \mathbf{w_0} + \mathbf{w}_0^T \mathbf{x}_n \mathbf{x}_n^T \mathbf{w_0})\right) = 0$$

$$\Rightarrow \frac{\partial}{\partial \mathbf{w}_0}\left(\sum_{n=1}^{N} (-2 y_n^T \mathbf{x}_n^T \mathbf{w_0} + \mathbf{w}_0^T \mathbf{x}_n \mathbf{x}_n^T \mathbf{w_0})\right) = 0$$

$$\Rightarrow \sum_{n=1}^{N} \left[-2 y_n^T \frac{\partial}{\partial \mathbf{w}_0}(\mathbf{x}_n^T \mathbf{w_0}) + \frac{\partial}{\partial \mathbf{w}_0}(\mathbf{w}_0^T \mathbf{x}_n \mathbf{x}_n^T \mathbf{w_0}) \right] = 0$$

$$\Rightarrow \sum_{n=1}^{N} \left[-2 y_n^T \mathbf{x}_n + 2 \mathbf{x}_n \mathbf{x}_n^T \mathbf{w_0} \right] = 0 \qquad \text{[Using relation (67) \& (43) from [19]]}$$

$$\Rightarrow \sum_{n=1}^{N} \left[\mathbf{x}_n \; y_n \right] = \left[\sum_{n=1}^{N} \mathbf{x}_n \mathbf{x}_n^T \right] \mathbf{w_0}$$

$$\Rightarrow \frac{1}{N} \sum_{n=1}^{N} \left[\mathbf{x}_n \; y_n \right] = \frac{1}{N} \left[\sum_{n=1}^{N} \mathbf{x}_n \mathbf{x}_n^T \right] \mathbf{w_0}$$

$$\Rightarrow \mathbf{a} = \mathbf{A w_0} \qquad \text{[Where, } \mathbf{a} = \frac{1}{N} \sum_{n=1}^{N} \left[\mathbf{x}_n \; y_n \right] \text{]}$$

$$\Rightarrow \mathbf{w_0} = \mathbf{A}^{-1} \mathbf{a}$$

This is the value of $\mathbf{w_0}$ that minimizes RSS. So, this is the OLS estimate of $\mathbf{w_0}$. Therefore,

$$\hat{\mathbf{w}}(\mathbf{D}) = \mathbf{A}^{-1} \mathbf{a} .$$

[Proved]

[An almost similar proof can also be found in [23]]

A.7

Means of Gaussian and Cauchy distributions

Mean of Gaussian distribution and reason for its boundness:

Gaussian distribution, $p(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2}\right)$; $x \in (-\infty, \infty)$

[With unit (constant) variance and μ mean]

Before we talk about (or evaluate) the mean, we need to find the mean function. We find it as

$$\mu_f^G = \frac{1}{\sqrt{2\pi}} \int x \exp\left(-\frac{(x-\mu)^2}{2}\right) dx$$

$$= \frac{1}{\sqrt{2\pi}} \int (\mu + z) \exp\left(-\frac{z^2}{2}\right) dz \quad [\text{Using } x - \mu = z \Rightarrow dx = dz\,]$$

$$= \frac{\mu}{\sqrt{2\pi}} \int \exp\left(-\frac{z^2}{2}\right) dz + \frac{\mu}{\sqrt{2\pi}} \int z \exp\left(-\frac{z^2}{2}\right) dz$$

$$\Rightarrow \mu_f^G = \frac{\mu}{\sqrt{2\pi}} \int \exp\left(-\frac{z^2}{2}\right) dz - \frac{\mu}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right)$$

But $\exp\left(-\frac{z^2}{2}\right)$ is a bounded function as $\exp\left(-\frac{z^2}{2}\right) \to 0$ for both $z \to -\infty$ and $z \to \infty$. Therefore, the Gaussian mean function μ_f^G is a bounded function or simply the mean of Gaussian distribution is bounded.

Mean of Cauchy's distribution and reason for its unboundness:

Cauchy's distribution, $p(x) = \frac{1}{\pi} * \frac{1}{1+x^2}$

Before we talk about (or evaluate) the mean, we need to find the mean function. We find it as

$$\mu_f^C = \int x\, p(x) dx$$

$$= \frac{1}{2\pi} \int \frac{2x}{1+x^2} dx$$

$$\Rightarrow \mu_f^C = \frac{1}{2\pi} \ln\left(1+x^2\right)$$

But $\ln\left(1+x^2\right)$ is an unbounded function as $\ln\left(1+x^2\right)\to\infty$ for both $x\to-\infty$ and $x\to\infty$. Therefore, the Cauchy mean function μ_f^C is an unbounded function or simply the mean of Cauchy distribution is unbounded (or undefined).

<u>This could also be proved in an alternating way that is given below:</u>

Let for any integer $N\succ 0$ the N th sample from Cauchy's distribution is $x_N = a \succ 0$
Then the maximum mean value for Cauchy up to this point is

$$\mu_N = \frac{1}{2\pi}\left[\ln\left(1+x^2\right)\right]_0^a \quad .$$

$$= \frac{1}{2\pi}\ln\left(1+a^2\right)$$

[We said this is the maximum value of the mean at this point as we used the lower limit of $x=0$. If we take any non-zero lower limit of x , e.g. $x=b$, then the mean becomes $\frac{1}{2\pi}[\ln(1+a^2)-\ln(1+b^2)]$, which is smaller than $\frac{1}{2\pi}\ln(1+a^2)$]

$$\Rightarrow \mu_N = \ln\left(1+a^2\right)^{\frac{1}{2\pi}}$$

Now, let the $N+1$st sample from Cauchy distribution is $x=a+t$ for t ≥ 0
i.e. $x_{N+1}=a+t$.

Now we can obviously write, $(1+a^2)^{\frac{1}{2\pi}} \prec e^{a+t}$

[As from the Binomial expansion, we have

$$(1+a^2)^{\frac{1}{2\pi}} = \left[\left(1+a^2\right)^{\frac{1}{2}}\right]^{\frac{1}{\pi}} = \left[1+a+.....\right]^{\frac{1}{\pi}}$$

But $e^{(a+t)} = 1+(a+t)+\frac{(a+t)^2}{2!}+..... \succ \left[1+a+.....\right]^{\frac{1}{\pi}} = (1+a^2)^{\frac{1}{2\pi}}]$

$$\Rightarrow \ln(1+a^2)^{\frac{1}{2\pi}} \prec a+t$$

$$\Rightarrow \frac{1}{2\pi}\ln(1+a^2) \prec a+t$$

$$\Rightarrow \mu_N \prec x_{N+1}$$

From this, we will show that, the mean of Cauchy is unbounded as N gets larger.
We have, $\mu_N \prec x_{N+1}$,

$$\Rightarrow x_{N+1} \succ \mu_N$$

$$\Rightarrow x_{N+1} \succ \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$\Rightarrow \frac{x_{N+1}}{\sum_{i=1}^{N} x_i} \succ \frac{1}{N}$$

$$\Rightarrow \frac{x_{N+1}}{\sum_{i=1}^{N} x_i} + 1 \succ \frac{1}{N} + 1$$

$$\Rightarrow \frac{\sum_{i=1}^{N+1} x_i}{\sum_{i=1}^{N} x_i} \succ \frac{N+1}{N}$$

$$\Rightarrow \frac{1}{N+1}\sum_{i=1}^{N+1} x_i \succ \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$\Rightarrow \mu_{N+1} \succ \mu_N$$

That means, for Cauchy's distribution, the mean for any sample size is lesser than the mean for the sample size that is larger than before.

A.8

Reason for the asymptotic behavior of the error function:

From the relation (1.20) in Chapter1, we have

$$\overline{\epsilon_G} = \left[\frac{1}{2\beta} + \sigma_\nu^2\right] \sum_{k'k''}^{d} \Sigma_{k'k''} \overline{\left(\left(\mathbf{A}^{-1}\right)_{k'k''}\right)} + \sigma_\nu^2 \; ; \qquad \text{(A.8.1)}$$

Where $\left(\mathbf{A}^{-1}\right)_{k'k''}$ is the inverse of the estimated $d \times d$ input covariance matrix,

$\mathbf{A_{k'k''}} = \sum_{\alpha=1}^{N} x_{k'}^{\alpha} x_{k''}^{\alpha}$.

Now, determinant of $\mathbf{A_{k'k''}}$, denoted by det ($\mathbf{A_{k'k''}}$) = | $\mathbf{A_{k'k''}}$ | = $\lambda_1 \lambda_2 .....\lambda_d$ where $\lambda_1, \lambda_2, ..., \lambda_d$ are the eigen values of $\mathbf{A_{k'k''}}$ and all of them are positive as $\mathbf{A_{k'k''}}$ is a estimated covariance matrix. Thus, we can write,

$$\left(\mathbf{A}^{-1}\right)_{k'k''} = \frac{a\ non-\sin gular\ matrix}{\left|\mathbf{A_{k'k''}}\right|}$$

$$\Rightarrow \left(\mathbf{A}^{-1}\right)_{k'k''} \propto \frac{1}{\left|\mathbf{A_{k'k''}}\right|} = \frac{1}{\lambda_1 \lambda_2 .....\lambda_d}$$

$$\Rightarrow E\left[\left(\mathbf{A}^{-1}\right)_{k'k''}\right] = \mathbf{\Phi}\ E\left[\frac{1}{\lambda_1 \lambda_2 .....\lambda_d}\right]$$

[Here, $\mathbf{\Phi}$ is a constant that depends on the distribution of $\mathbf{A_{k'k''}}$ following some constrains.]

$$\Rightarrow \overline{\left(\mathbf{A}^{-1}\right)_{k'k''}} = \mathbf{\Phi}\ E\left[\frac{1}{\lambda_1\lambda_2.....\lambda_d}\right] \qquad \text{(A.8.2)}$$

But the other terms, involved in the R.H.S. of (A.8.1) are $\frac{1}{2\beta}$, σ_v^2 and $\sum_{k'k''}$, which are positive, finite and fixed in the aspect of mean of the general average generalized error ($\overline{\in_G}$). Therefore, the value of $\overline{\in_G}$ mainly depends on $\overline{\left(\mathbf{A}^{-1}\right)_{k'k''}}$.
As a result, we can write,

$$\overline{\in_G} \propto \overline{\left(\mathbf{A}^{-1}\right)_{k'k''}}$$

$$\Rightarrow \overline{\in_G} \propto \mathbf{\Phi}\ E\left[\frac{1}{\lambda_1\lambda_2.....\lambda_d}\right] \quad \text{[Using (A.8.2)]}$$

Now, we see that $\overline{\in_G}$ depends on a composite term involving $\mathbf{\Phi}$ and the eigen values ($\lambda_1, \lambda_2, ..., \lambda_d$). Therefore, its singularity or pole can occur due to any of them.
For $N \prec d$, as we have said before, $\left(\mathbf{A}^{-1}\right)_{k'k''}$ is not defined (at the same time $\mathbf{\Phi}$ is undefined and also at least on of the d eigen values tends to zero leading $E\left[\frac{1}{\lambda_1\lambda_2.....\lambda_d}\right] \to \infty$). Therefore, $\overline{\in_G}$ becomes undefined (or singular).
For $N = d$ (in non-asymptotic case), $\mathbf{\Phi}$ brings negative values (from A.9.2) and thus $\overline{\in_G}$ becomes invalid.

For $N = d+1$, $\mathbf{\Phi}$ becomes infinite (from A.9.2) and thus $\overline{\in_G} \to \infty$.

A.9

About the mean of the estimated input covariance matrix:

(This is an alternate proof of [6])

Statement: $\overline{\left(\left(A^{-1}\right)_{jj'}\right)} = \frac{1}{N-d-1}\left(\mathbf{\Sigma}^{-1}\right)_{jj'}$

Proof: we have, $A_{jj'} = \sum_{\alpha=1}^{N} x_j^{\alpha} x_{j'}^{\alpha}$ $[\ j, j' = 1,2,...,d\ ]$ [From relation (1.13)]

$= \sum_{\alpha=1}^{N} x_{j\alpha} x_{j'\alpha}$ [As α is just a suffix index, not a power index]

But $A_{jj'}$ is the sample of the covariance matrix of $\boldsymbol{\Sigma}$. So, $A_{jj'}$ has the distribution of $W_d(\boldsymbol{\Sigma}, N)$ and $\mathbf{A}^{-1}$ has the distribution of $W_d^{-1}\left(\boldsymbol{\Sigma}^{-1}, N\right)$. But our $\boldsymbol{\Sigma}$ is a diagonal matrix. Therefore, $\boldsymbol{\Sigma}^{-1}$ is also a diagonal matrix.

Thus, $\overline{\left(\left(A^{-1}\right)_{j\,j'}\right)} = E\left[A^{-1}_{j\,j'}\right]$ is also a diagonal matrix and we will write

$$\overline{\left(\left(A^{-1}\right)_{j\,j'}\right)} = E\left[\left(\sum_{\alpha=1}^{N} x_{j\alpha}\, x_{j'\alpha}\right)^{-1}\right]\delta_{j\,j'}\left(\boldsymbol{\Sigma}^{-1}\right)_{jj'} \quad \text{[Using [17]]}$$

$$[\, j = 1,2,...,d\,]$$

$$\Rightarrow \quad \overline{\left(\left(A^{-1}\right)_{j\,j'}\right)} = E\left[\left(x_{j1}^2 + x_{j2}^2 + ... + x_{jN}^2\right)^{-1}\right]\left(\boldsymbol{\Sigma}^{-1}\right)_{jj} \qquad (*)$$

Now, by a little analysis, we can easily realize that if we have a vector with d correlated input, then the necessary relation to find each of them is d-1.
Again, from [29], we know that
Degrees of freedom in terms of sample size
= number of observations minus the number of necessary relations.
Then from the above relation (*) , each term of $E\left[\left(x_{j1}^2 + x_{j2}^2 + ... + x_{jN}^2\right)^{-1}\right]$ has the
degress of freedom = $N-(d-1) = N-d+1$

Thus each (diagonal) element of $\overline{\left(\left(A^{-1}\right)_{j\,j'}\right)}$, i.e. $E\left[\left(x_{j1}^2 + x_{j2}^2 + ... + x_{jN}^2\right)^{-1}\right]\left(\boldsymbol{\Sigma}^{-1}\right)_{jj}$ is said to have an inverse Gamma distribution with $(N-d+1)$ degrees of freedom and with the scale parameter $\lambda = \Sigma_{jj}^{-1}$;
Now if we notify each of the (diagonal) elements of $\left(\overline{A^{-1}}\right)$ as w_j^{-1}, then we can write, $w_j^{-1} \in (k\,,\,\lambda)$; where, $k = N-d+1$ & $\lambda = \Sigma_{jj}^{-1}$.
This has a density function as below [14] :

$$f\left(w_j^{-1}\right) = \frac{(\lambda/2)^{k/2}\,(w_j^{-1})^{\frac{k}{2}-1}\exp\left[-\frac{\lambda w_j^{-1}}{2}\right]}{\Gamma(k/2)}$$

Thus, $\mathrm{E}\left[w_j^{-1}\right] = \int_{0^+}^{\infty} w_j^{-1}\, f\left(w_j^{-1}\right)\, dw_j$

$$= \frac{(\frac{\lambda}{2})^{\frac{k}{2}}}{\Gamma\left(\frac{k}{2}\right)}\int_{0^+}^{\infty} (w_j^{-1})\,(w_j^{-1})^{\frac{k}{2}-1} \exp\left[-\frac{\lambda w_j^{-1}}{2}\right] dw_j$$

$$= \frac{(\frac{\lambda}{2})^{\frac{k}{2}}}{\Gamma\left(\frac{k}{2}\right)}\int_{0^+}^{\infty} (w_j^{-1})^{\frac{k}{2}} \exp\left[-\frac{\lambda w_j^{-1}}{2}\right] dw_j$$

Now Let, $w_j^{-1} = z$

$\Rightarrow -\frac{1}{w_j^2} dw_j = dz$

$\Rightarrow dw_j = - w_j^2 \, dz$

[As $w_j^{-1} = z, \Rightarrow w_j = z^{-1}$, $\Rightarrow w_j^2 = z^{-2}$. and $w_j \to 0$, $\Rightarrow z = \infty$;
$w_j \to \infty$; $z \to 0$]

Then we can write, $E[w_j^{-1}] = \frac{(\frac{\lambda}{2})^{k/2}}{\Gamma(k/2)} \int_{\infty}^{0} z^{\frac{k}{2}} \exp.[-\frac{\lambda z}{2}] \, (-z^{-2} \quad dz)$

$\Rightarrow E[w_j^{-1}] = \frac{(\frac{\lambda}{2})^{k/2}}{\Gamma(k/2)} \int_0^{\infty} z^{\frac{k}{2}-2} \quad \exp\left[-\frac{\lambda z}{2}\right] dz$

Now let, $\frac{\lambda z}{2} = m$

$\Rightarrow z = \frac{2m}{\lambda}$

$\Rightarrow dz = \frac{2dm}{\lambda}$, and $z \to 0 \Rightarrow m \to 0$, $z \to \infty \Rightarrow m \to \infty$.

Then we get, $E[w_j^{-1}] = \frac{(\frac{\lambda}{2})^{k/2}}{\Gamma(k/2)} \int_0^{\infty} (\frac{2m}{\lambda})^{\frac{k}{2}-2} \quad \exp(-m) \frac{2}{\lambda} dm$.

$= \frac{(\frac{\lambda}{2})^{k/2}}{\Gamma(k/2)} \frac{2^{\frac{k}{2}-2}}{(\lambda)\left(\frac{k}{2}-2\right)} \frac{2}{\lambda} \int_0^{\infty} m^{\frac{k}{2}-2} \quad \exp(-m) dm.$

$= \frac{2^{-1}}{\lambda^{-1}\Gamma\left(\frac{k}{2}\right)} \int_0^{\infty} m^{(\frac{k}{2}-1)-1} \exp(-m) dm$

$\Rightarrow E[w_j^{-1}] = \frac{\lambda}{2\Gamma(k/2)} \Gamma(\frac{k}{2}-1)$ [As, $\Gamma a = \int_0^{\infty} t^{(a-1)} \exp(-t) dz$]

$= \frac{\lambda \quad \Gamma(\frac{k}{2}-1)}{2\left(\frac{k}{2}-1\right)\Gamma\left(\frac{k}{2}-1\right)}$ [As, $\Gamma(n+1) = n\Gamma n$]

$= \frac{\lambda}{2(\frac{k}{2}-1)}$

$\Rightarrow E[w_j^{-1}] = \frac{\lambda}{k-2}$

$$\Rightarrow \mathrm{E}[w_j^{-1}] = \frac{\Sigma_{jj'}^{-1}}{N-d-1} \quad [\text{ since } k = N-d+1 \text{ }]$$

$$\text{But}\left(\overline{A^{-1}}\right) = \mathrm{E}[w_j^{-1}] \;;\; j = 1,2,...,d$$

$$\Rightarrow \left(\overline{A^{-1}}\right) = \frac{1}{N-d-1}\left(\Sigma^{-1}\right)_{jj'} \tag{A.9.1}$$

[**Proved**].

From the above derivation, we see that the mean of the inverse Gamma distribution has special weakness to the number 2 regarding the degrees of freedom of the distribution. And we have also seen that according to our choice of the input samples, we get inverse Gamma distribution for the mean of the inverse of our sample Covariance matrix. Keeping this choice, if we like to avoid the infinite value of the general averaged generalization error $\overline{\epsilon_G}$, we should follow the track for which Number of degrees of freedom in the element of the mean of the inverse sample covariance matrix -2 > 0

$\Rightarrow N-d-1 \succ 0$

$\Rightarrow N \succ d+1$

Otherwise, for $N = d+1$ the elements act similar like Cauchy's distribution that discussed in Chapter 0 (Introduction) of this thesis.

Again, comparing (A.8.2) and (A.9.1) we see find

$$\mathbf{\Phi}\, E\left[\frac{1}{\lambda_1\lambda_2.....\lambda_d}\right] = \frac{1}{N-d-1}\left(\Sigma^{-1}\right)_{jj'}$$

$$\Rightarrow \mathbf{\Phi} = \frac{1}{N-d-1}\left(\Sigma^{-1}\right)_{jj'}\left(E\left[\frac{1}{\lambda_1\lambda_2.....\lambda_d}\right]\right)^{-1} \tag{A.9.2}$$

[This relation is helpful for A.8]

G

Gibbs distribution:

It is a probability distribution that mainly came from thermodynamics describing the probability of states with respect to energy. In our context, these states are compared with the possible combinations of weights in the weight space and energy is compared to the error due to the weights. It has a general form:

$$P_N(w) = Z_N^{-1} \exp\left(-\beta \sum_{\alpha=1}^{N} E^{\alpha}(w)\right);$$

where Z_N is the normalization integral and is a function of β

β is compared to the inverse of the noise in the weight space

and $\sum_{\alpha=1}^{N} E^{\alpha}(w)$ is the error function.

We have chosen this distribution as it is the only distribution that corresponds directly to the error minimization.

In fact, any distribution can be expressed as Gibb's distribution if its energy function (normally the quadratic function that depends on the variable representing the degrees of freedom) is proportional to the negative $\log P$ and the co-efficient of the energy function (here, β) is fixed. We can analyze this in the following way:

Suppose, $P(x)$ is any distribution. We can write

$$P(x) = \exp(\ln(P(x)))$$

Here, we have $\int P(x)dx = 1$. But if we raise $P(x)$ to the power $\beta \neq 1$, then $\int (P(x))^{\beta} dx \neq 1$. In this case we need to normalize the probability in order to get the total probability summed to 1. Thus, if we know β, we can write,

$$P(x \mid \beta) = Z^{-1}(\beta) \exp\left[\ln\left(P(x)\right)^{\beta}\right];$$

where $Z(\beta)$ is the normalizing constant

and $Z(\beta) = \int dx \exp\left(\ln\left(P(x)\right)^{\beta}\right)$ with $P(x) > 0$ and $\ln P(x) < 0$

Though this distribution may appear unlikely for some training methods (the stationary probability distribution in the general case is non-Gibbsian; it becomes Gibbsian only when the covariance matrix of the backpropagated gradients is *isotropic* and independent of the weights whose distribution is being sought. [4]), it arises naturally for stochastic algorithms.

W

Wishart distribution: This distribution deals with the distribution of the sample covariance matrix.
Consider N samples $\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_N}$ where each of these samples is a P dimensional vector with N>P and they come from the distribution $N_P(\boldsymbol{\mu}, \Sigma)$. Let **S** is the sample covariance matrix in the following way:

$$\mathbf{S} = \frac{1}{N-1}\sum_{\alpha=1}^{N}(\mathbf{x}_\alpha - \bar{\mathbf{x}})(\mathbf{x}_\alpha - \bar{\mathbf{x}})'; \text{ where } \bar{\mathbf{x}} = \frac{1}{N}\sum_{\alpha=1}^{N}\mathbf{x}_\alpha$$

Then **S** is an unbiased estimator of the population covariance matrix Σ
If we consider a positive definite matrix **A** such that

$$\mathbf{A} = (N-1)\mathbf{S}$$

then the distribution of **A** (or, **S**) is often called the Wishart distribution. The distribution of **A** is denoted by $W_p(\Sigma, n)$, where n = N-1 = Number of degrees of freedom.

Inverted Wishart distribution: This deals with the distribution of the inverse of the sample covariance matrix.
If a positive definite matrix **A** has the distribution $W_p(\Sigma, n)$ and another positive definite matrix **B = A^{-1},** then the distribution of **B** is called the Inverted Wishart distribution and is denoted by $W_p^{-1}(\Sigma^{-1}, n)$. In this case, Σ^{-1} is termed as precision matrix.

References:

[1] Douglas C. Montgomery, Goerge C. Rnger (1999). Applied Statistics and Probability For Engineers. John Wiley.

[2] Levin, E., Tishby, N., and Solla, S.A. (1990). A Statistical Approach to Generalization in Layered Neural Networks. *Proceeding of the* IEEE, 78, 1568-1574

[3] Radons, G., Schuster, H.G., and Werner, D. (1990): Drift and Diffusion in Back propagation Learning. In R. Eckmiller (Eds.), Parallel Processing in Neural Systems and Computers, (pp 261-266). Elsevier Science Publishers.

[4] Hansen, L.K., Pathria, R., and Salamon, P. (1991). Stochastic Dynamics of Supervised Learning. *Electronics Institute Preprint,* The Technical University of Denmark.

[5] Peter Ahrendt (2004), IMM, Technical University of Denmark. The Multivariate Gaussian Probability Distribution.

[6] Anderson, T.W. (1984). An Introduction to Multivariate Statistical Analysis. New York: John Wiley & Sons.

[7] Akaike, H. (1969). Fitting Autoregressive Models for Prediction. *Annals of The Institute of Statistical Mathematics,* 21, 243-247.

[8] Akaike H. (1974). A new Look at the Statistical Model Identification. IEEE Transactions on automatic control, AC-19, 716-723

[9] Ljung, L. (1987). System Identification: Theory for the User. Englewood Cliffs, New Jersey: Prentice Hall.

[10] Anders Meng (2004), IMM, DTU. An Introduction to Variational Calculus in Machine Learning.

[11] Krogh, A. and Hertz, J.A., (1992). Generalization in a Linear Perceptron in the Presence of Noise. *Journal of Physics A: Math. Gen.,Vol 25, issue 5, 1135-47.*

[12] A Krogh, J A Hertz and G I Thorbergsson (1989): Phase Transitions in Simple Learning. *Journal of Physics A: Math. Gen.,* 22**,** 2133-2150.

[13] Lars Kai Hansen (1993). Stochastic Linear Learning: Exact Test and Training Error Averages. Neural Networks, Vol 6, issu 3, 393-396.

[14] James D. Hamilton: Time Series Analysis. (1994) By Princeton University Press,New Jersey.

[15] Christopher M. Bishop: Neural Network for Pattern Recognition. (2003) Oxford University Press.

[16] Gaussian Joint Variable Theorem: http://icl.pku.edu.cn/yujs/MathWorld/math/g/g092.htm

[17] K.V. Mardia, J.T. Kent and J.M. Bibby (1995).: Multivariate Analysis. Academic Press.

[18] Rectangular Hyperbola: http://mathworld.wolfram.com/RectangularHyperbola.html

[19] Michael Syskind Pedersen, ISP, IMM, DTU: Matricks.

http://www.imm.dtu.dk/pubdb/views/edoc_download.php/2976/pdf/imm2976.pdf

[20] Lars Kai Hansen (2004). Generalized cross-over in linear learning with known input distribution: Exact results.

[21] Robert M. Gray and Lee D. Davisson: An introduction to Statistical Signal Processing.

[22] Cauchy distribution: http://www.itl.nist.gov/div898/handbook/eda/section3/eda3663.htm

[23] Milos Hauskrecht: Cs 1571 Introduction to AI, Lecture 26

[24] Samprit Chatterjee and Bertram Price (1977): Regression Analysis By Example. Wiley.

[25] Le Cun, Y. , Canter, I., and Solla, S. A. (1991). Eigenvalues of Covariance Matrices: Application to neural-Network Learning. Physics of Review Letters, 66, 2396-99.

[26] Le Cun, Y., Denker, J.S., and Solla, S.A. (1989). Optimal Brain Damage. In D. Touretzky (Ed.) Neural Information Processing Systems, Denver 1989, (pp 598-605). Morgan Kaufman.

[27] Fogel, D.B. (1991). An Information Criterion for Optimal Neural Network Selection. *IEEE Transactions on Neural Networks*, 2, 490-497.

[28] Hoffmann, N. and Larsen, J. (1991). Algorithms for model Order Reduction in Nonlinear Filters. *Electronics Institute Preprint*, The Technical University of Denmark.

[29] Dr. Chong Ho (Alex) Yu. Illustrating degrees of freedom in terms of sample size and dimensionality: http://seamonkey.ed.asu.edu/~alex/computer/sas/df.html