A Practical Bayesian Framework for Backpropagation Networks

David J.C. MacKay
1992

Bardia Mojra
September 16, 2020
Robotic Vision Lab
University of Texas at Arlington

Introduction

- A quantitative and practical Bayesian framework for learning of mapping in feed-forward networks.
- Framework features:

1. Objective solution comparison using alternative network architectures
2. Objective stopping rules for network pruning and growing procedures
3. Objective choice of magnitude and type of weight decay terms and additive regularizers (for penalizing large weights, etc.)
4. A measure of the effective number of well determined parameters and on network output
5. Quantified estimates of the error bars on network parameters in a model
6. Objective comparisons with alternative learning and interpolation models

Introduction

- Bayes' rule:

$$P(A|B)=\frac{P(B|A)P(A)}{P(B)}$$

- P(A|B): Posterior - probability of A occurring given that B has occurred
- P(B|A): Likelihood – likelihood of B occurring given A has occurred
- P(A): Prior – general probability or initial degree of belief that A has occurred
- P(B): Marginalization – general probability that B has occurred

Introduction

- Bayes' rule:
  - Remember your priors (Bayes' rate neglect)
  - Imagine your theory is wrong. Would the world look different?
  - Update incrementally (snow flakes of evidence)

Backpropagation

- Involves tuning learning parameters:
  I. Parameters that change the effective learning model i.e. number of hidden units and weight decay terms.
  II. Parameters concerned with function optimization technique i.e. “momentum” terms.

Backpropagation

$$E_D(D|w,A)=\sum_m (\frac{1}{2})[y(x^m;w,A)-t^m]^2$$

- D = {x^m , t^m}: Set of input target pairs
- If a set of values w is assigned to the connections in the network, the network defines a mapping y(x;w,A) from the input activities x to the output activities y.
- In plain backpropagation, we use some optimization technique such as gradient descent to minimized E_D over w-space.

Regularization with Energy Model

$$E_w(w|A) = \sum_i \left(\frac{1}{2} w_i^2\right)$$

- $E_w(w|A)$: some loss function (this is '[energy function]')
- Weight-dependent energy terms have decaying effect
- Energy of a system in physics represents movement

Model

$$M = \alpha E_w(w|A) + \beta E_D(D|w, A)$$

- M: Trained model with connections A, weights w, and data D
- α: Regularizing constant, decay rate (not to be confused with "momentum")
- β: Decay rate for weight parameters of the network
- Here the data is divided into two sets:
  - A training set that is used to optimize the parameters w of the network, $E_D(D|w,A)$
  - A test set that is used to optimize control parameters such as α and the architecture A, $E_w(w|A)$

MacKay's Inverse Probability

"In this paper the emphasis will be on quantifying the relative plausibilities of many alternative solutions to an interpolation or classification task; that task is defined by a single data set produced by the real world, and we do not know the prior ensemble from which the task comes. This is called *inverse probability*."

MacKay's Inverse Probability

"In this paper the emphasis will be on quantifying the relative plausibilities of many alternative solutions to an interpolation or classification task; that task is defined by a single data set produced by the real world, and we do not know the prior ensemble from which the task comes. This is called *inverse probability*."

$$P(A|B)=\frac{P(B|A)P(A)}{P(B)}$$

Likelihood

$$P(t^m|x^m, w, \beta, A) = \frac{e^{-\beta E(t^m|x^m, w, A)}}{Z_m(\beta)}$$

- P(t | x,w,β ,A): Probability distribution of target output t given as function input x, a network with specified architecture A and connections w

$$Z_m(\beta) = \int dt \exp(-\beta E)$$

- E: Error for a single datum
- β: measure of the presumed noise included in t
- If E is the quadratic error function then this corresponds to the assumption that t includes additive Gaussian noise with variance $\sigma_Y^2 = 1/\beta$

Prior

$$P(w|\alpha, A, R) = \frac{\exp[-\alpha E_w(w|A)]}{Z_w(\alpha)}$$

- P(w|α,A,R): a prior probability is assigned to alternative network connection strengths w

$$Z_w(\alpha) = \int d^k w \exp(-\alpha E_w)$$

- α: A measure of the characteristic expected connection magnitude
- If E_w is quadratic (i.e. energy function), then weights are expected to come from a Gaussian with zero mean and variance $\sigma_w^2 = 1/\alpha$
- R: Regularizer function (each can use different energy function E_w, which corresponds to alternative hypotheses about the statistics of the environment.

Posterior Probability

$$P(w|D,\alpha,\beta,A,R)=\frac{\exp[-\alpha E_w-\beta E_D]}{Z_M(\alpha,\beta)}$$

- P(w|D,α,β,A,R): Represent "the probability of the connections w" as a measure of plausibility that the model's parameters should have a specified value w. Note that this has nothing to do with the probability that particular algorithm might converge to w.

$$Z_M(\alpha,\beta)=\int d^k w \exp(-\alpha E_w-\beta E_D)$$

- Minimizing M is identical to finding the local most probable parameters w_{MP}. $M=\alpha E_w+\beta E_D$.
- Minimizing E_D by itself is identical to finding the maximum likelihood parameters w_{ML}.

Relevant Work

- Le Cun et al. (1990) demonstrated how to estimate the “saliency” of a weight, which is the change in M when the weight is deleted. This is used to simplify neural networks by pruning excessive nodes. However no stopping rule for weight deletion was offered other than using a test set as test environment control.
- Denker and Le Cun (1991) demonstrated how the Hessian of M can be used to assign error bars to the parameters and output of a network. These error bar can be quantified only once Beta is calculated. We used estimated Beta from training set as quantified prior knowledge when predicting based on new test data.

Determination of Alpha and Beta

$$P(\alpha,\beta|D,A,R)=\frac{P(D|\alpha,\beta,A,R)P(\alpha,\beta)}{P(D|A.R)}$$

$$P(D|\alpha,\beta,A,R)=\frac{Z_M(\alpha,\beta)}{Z_W(\alpha)\cdot Z_D(\beta)}$$

$$Z_D=\int d^N\cdot D\cdot e^{-\beta E_D}$$

- N: Degrees of freedom in the data set which is the number of output units times the number of data pairs
- K: The number free parameters which is the dimension of w.

$$Z_D=(2\pi/\beta)^{N/2} \qquad Z_W=(2\pi/\alpha)^{k/2}$$

Determination of Alpha and Beta

$$Z_M(\alpha,\beta)=\int d^k\cdot w\cdot e^{-M(w,\alpha,\beta)}$$

- Suppose M has a single minimum as a function of w, at w_{MP}, and assuming we can locally approximate M as quadratic there. Then Z_M is approximated by:

$$Z_M\simeq e^{-M(w_{MP})}(2\pi)^{k/2}det^{-1/2}A$$

- $A=\nabla\nabla M$ is the Hessian of M evaluated at w_{MP}
- The maxima of $P(D|\alpha,\beta,A,R)$ has the following properties:
  - $X_w^2 \equiv 2\alpha E_W = \gamma$
  - $X_D^2 \equiv 2\beta E_D = N-\gamma$
- γ: It is the effective number of parameters determined by data.
- λ: It is the eigenvalues of the quadratic from βE_D in the natural basis of E_w.

$$\gamma=\sum_{a=1}^{k}\frac{\lambda_a}{\lambda_a+\alpha}$$

Comparison of Different Models

$$P(D|A,R)=\int P(D|\alpha,\beta,A,R)\cdot P(\alpha,\beta)\cdot d\alpha\cdot d\beta$$

- This quantifies the evidence (data) for a given architecture A and regularizer function R.
- This enables comparing models in light of the data or given evidence.
- If M(w) is not quadratic, M would have many local minima and if network has symmetry, M will also be symmetric under permutation of its parameters.
- Then we know M(w) must share that symmetry so that each minima belongs to a family of symmetric minima of M.

Comparison of Different Models

- Consider a local minima at w*, and define a solution S_{w^*} as the ensemble of networks in the neighborhood of w*, and all symmetric permutations of that ensemble
- Then we can define posterior probability of alternative solutions S_{w^*} and parameters α and β:

$$P(S_{\dot{w}}, \alpha, \beta, A, R|D) \propto g \frac{\dot{Z}_M(\dot{w}, \alpha, \beta)}{Z_w(\alpha) \cdot Z_D(\beta)} P(\alpha, \beta) P(A, R)$$

- Where g is the permutation factor and,

$$\dot{Z}_M(\dot{w}, \alpha, \beta) = \int_{S_{\dot{w}}} d^k w \, e^{-M(w, \alpha, \beta)}$$

- Where the integral is performed only over the neighborhood of the minimum at w*.
- We also define the following as the evidence for α, β, and S_{w^*}:

$$g \frac{\dot{Z}_M(\dot{w}, \alpha, \beta)}{Z_w(\alpha) \cdot Z_D(\beta)}$$

Comparison of Different Models

- The parameters α and β will be chosen to maximize this evidence.
- Then the quantity we want to evaluate to compare alternative solutions is the evidence for S_{w*},

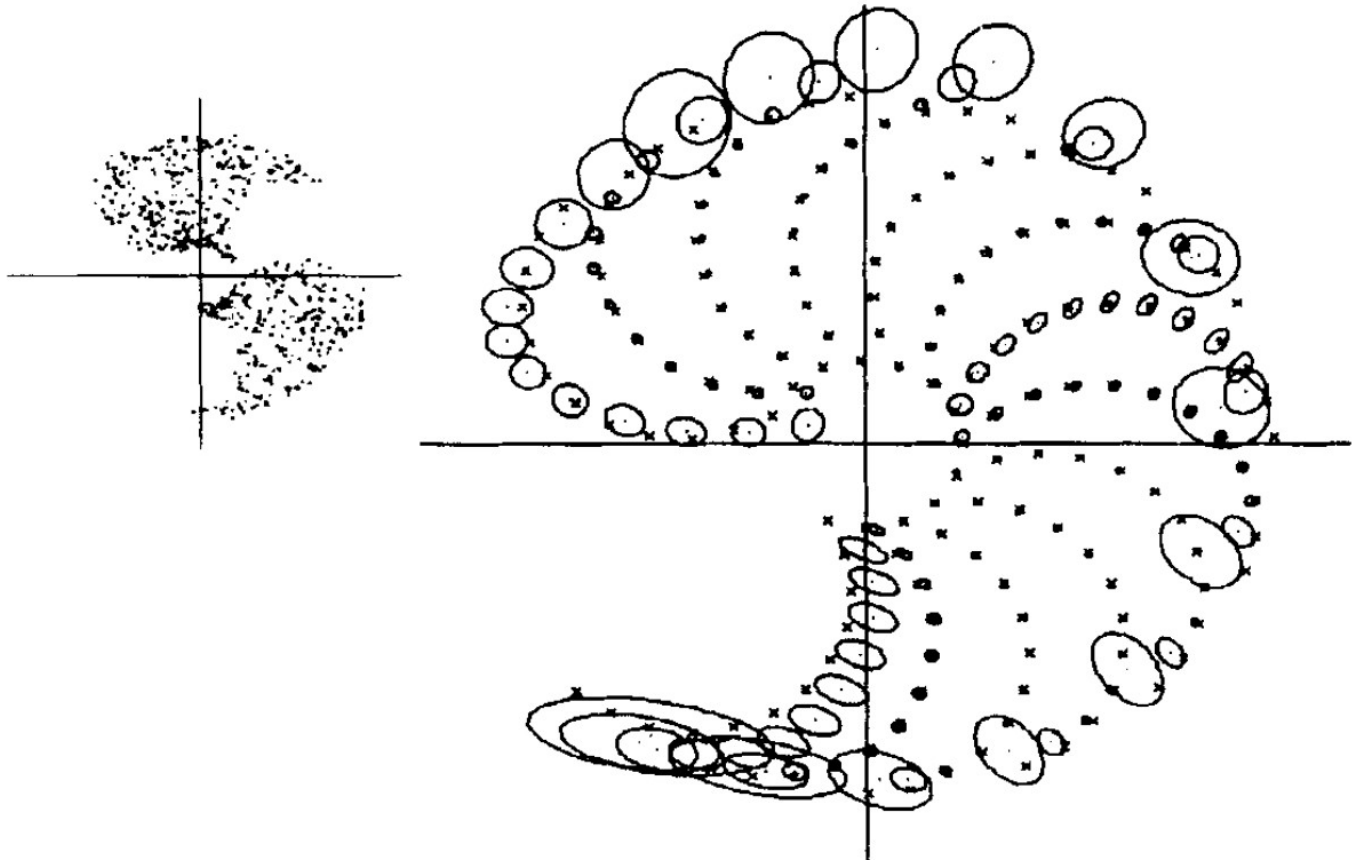
$$P(D, S_{\dot{w}} | A, R) = \int g \frac{\dot{Z}_M(\dot{w}, \alpha, \beta)}{Z_w(\alpha) Z_D(\beta)} P(\alpha, \beta) d\alpha d\beta$$

- Where Gaussian approximation is used for Z_M* and can be estimated by:

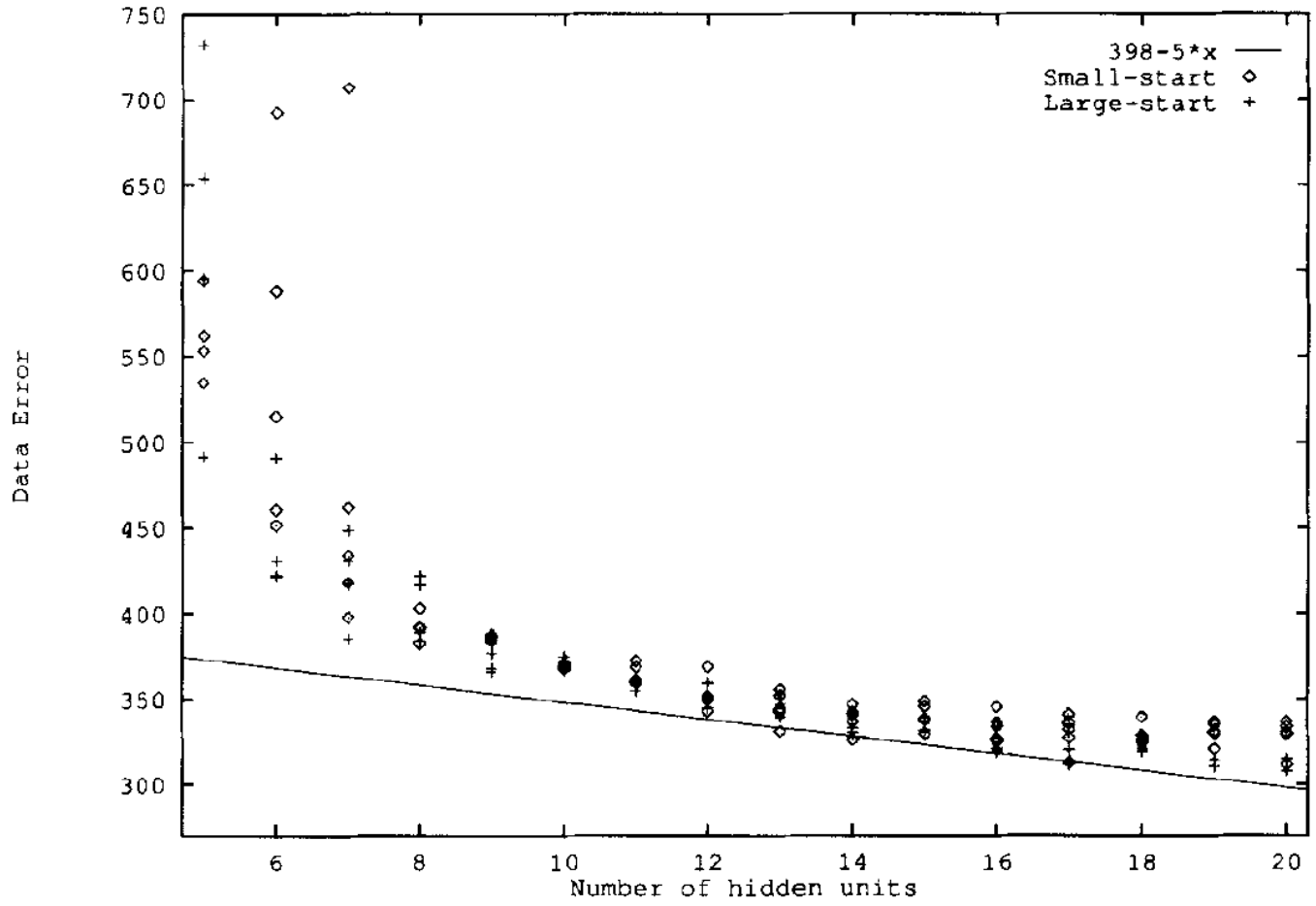
$$\dot{Z}_M \simeq e^{-M(\dot{w})} (2\pi)^{k/2} det^{-1/2} A$$

- A is again the Hessian of M evaluated at w*.
- Generalization of α and β is probably unacceptable but approximation of A only needs to accurate for a small range of α and β close to their most probable value.

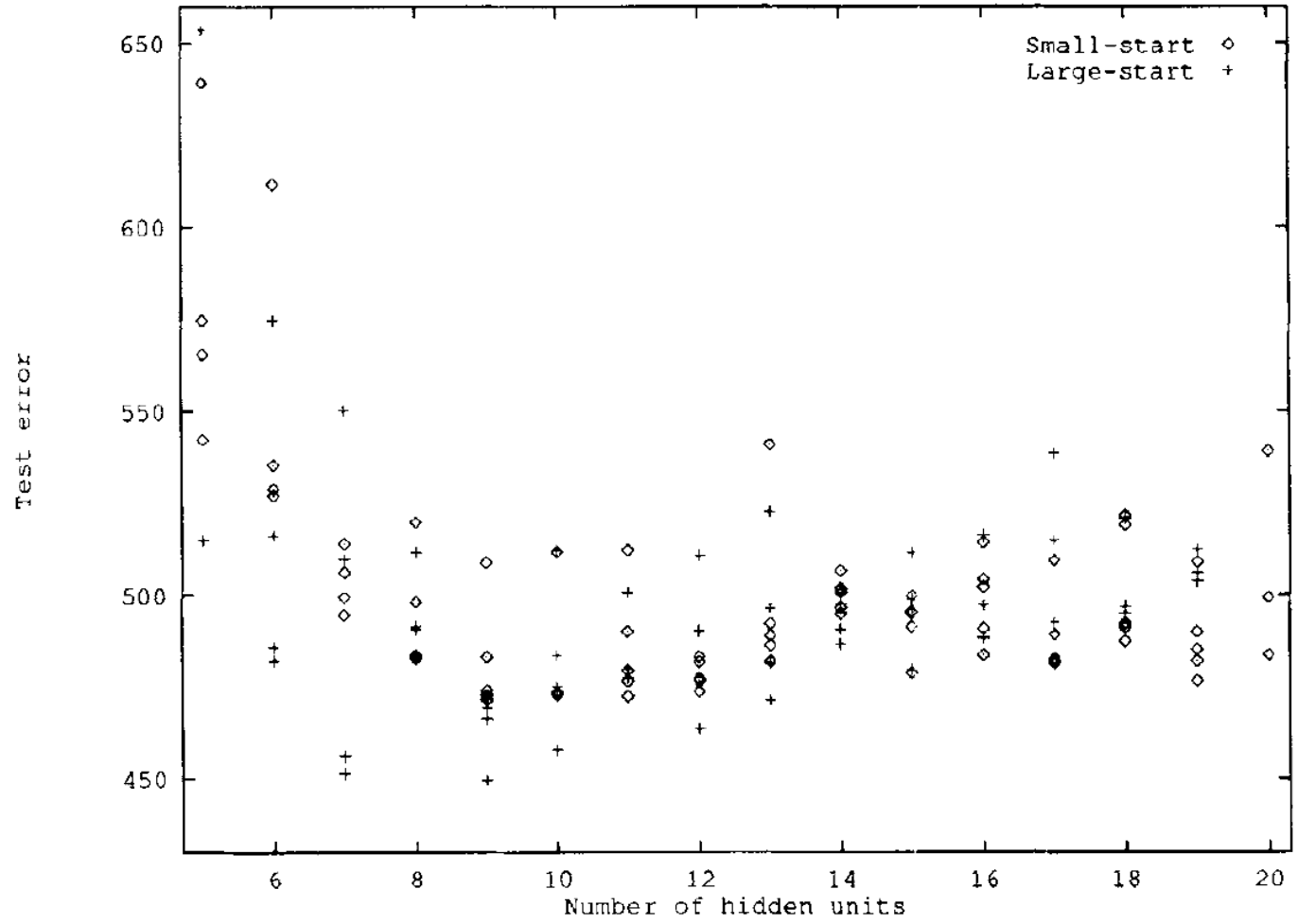
Demonstration – Error Bar



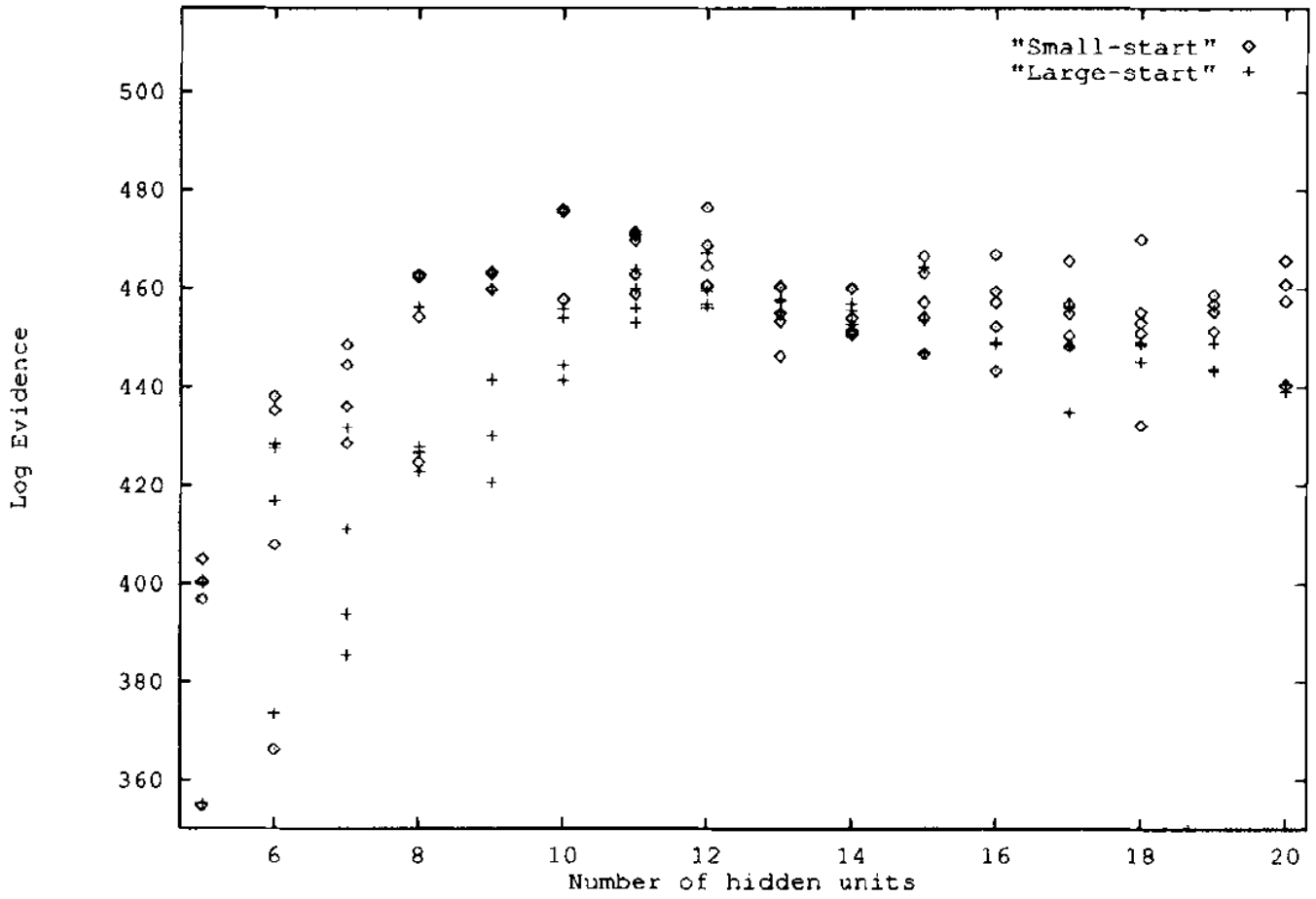
Demonstration – Hidden Units



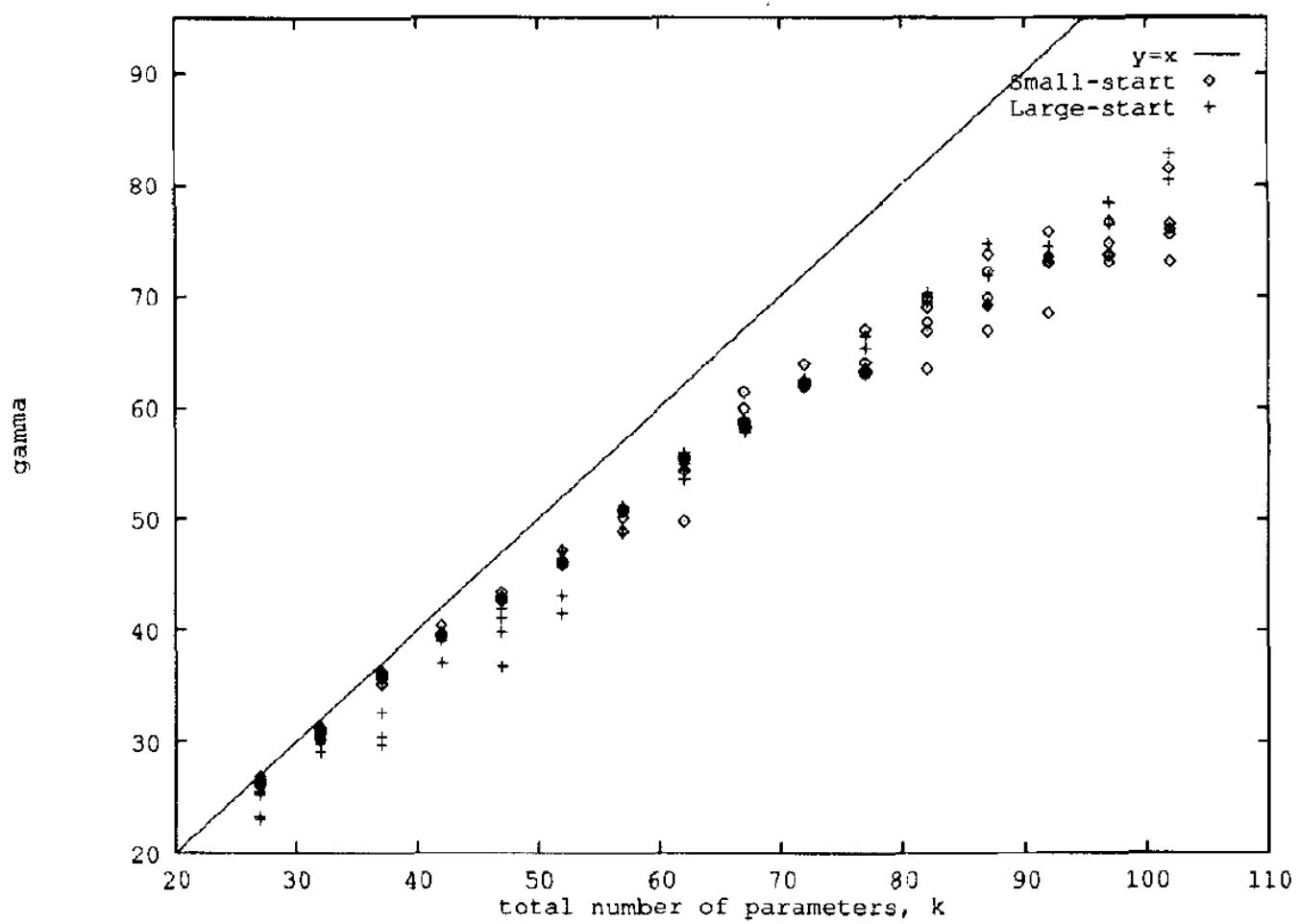
Demonstration – Occam's Razor



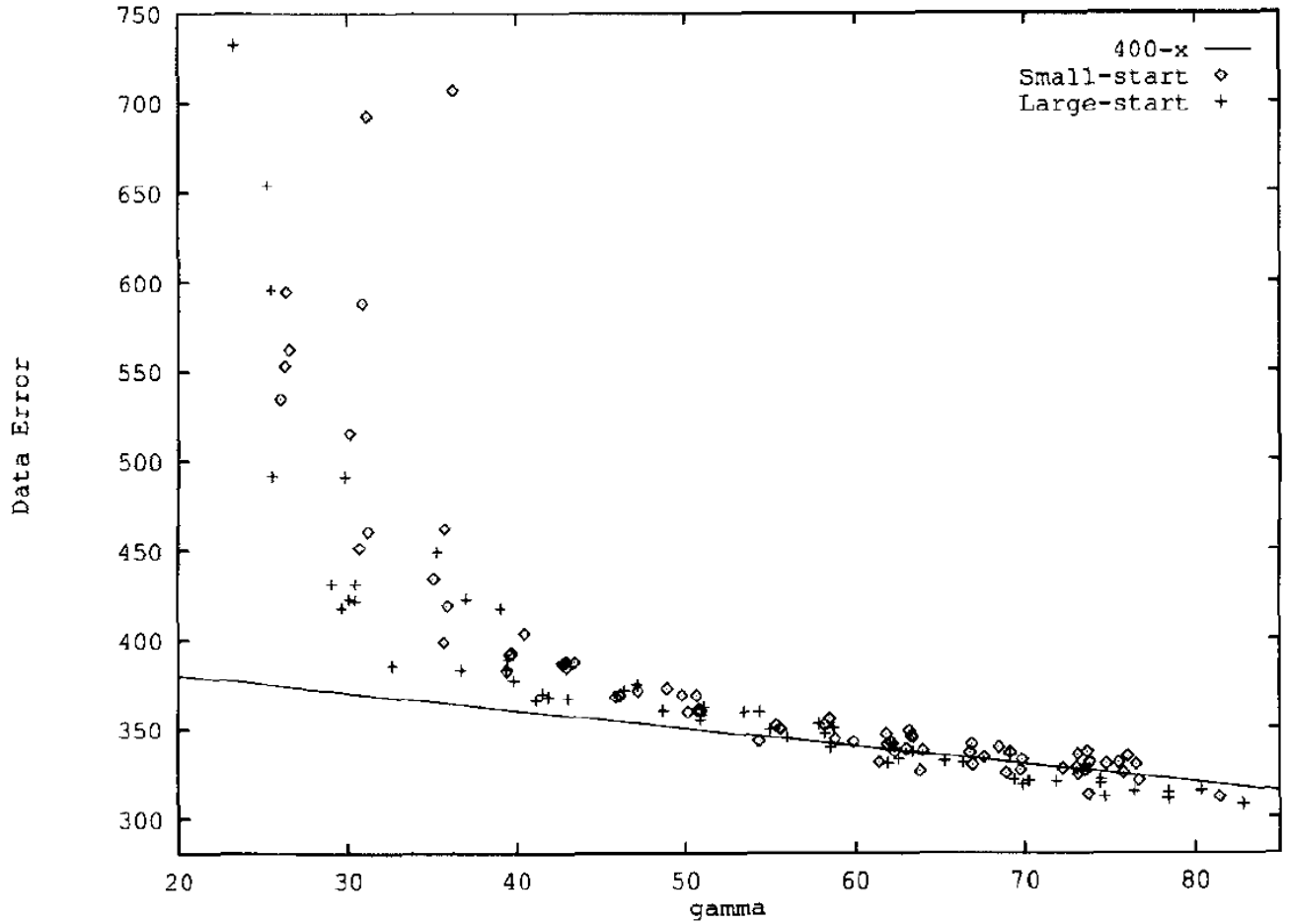
Demonstration – Occam's Razor



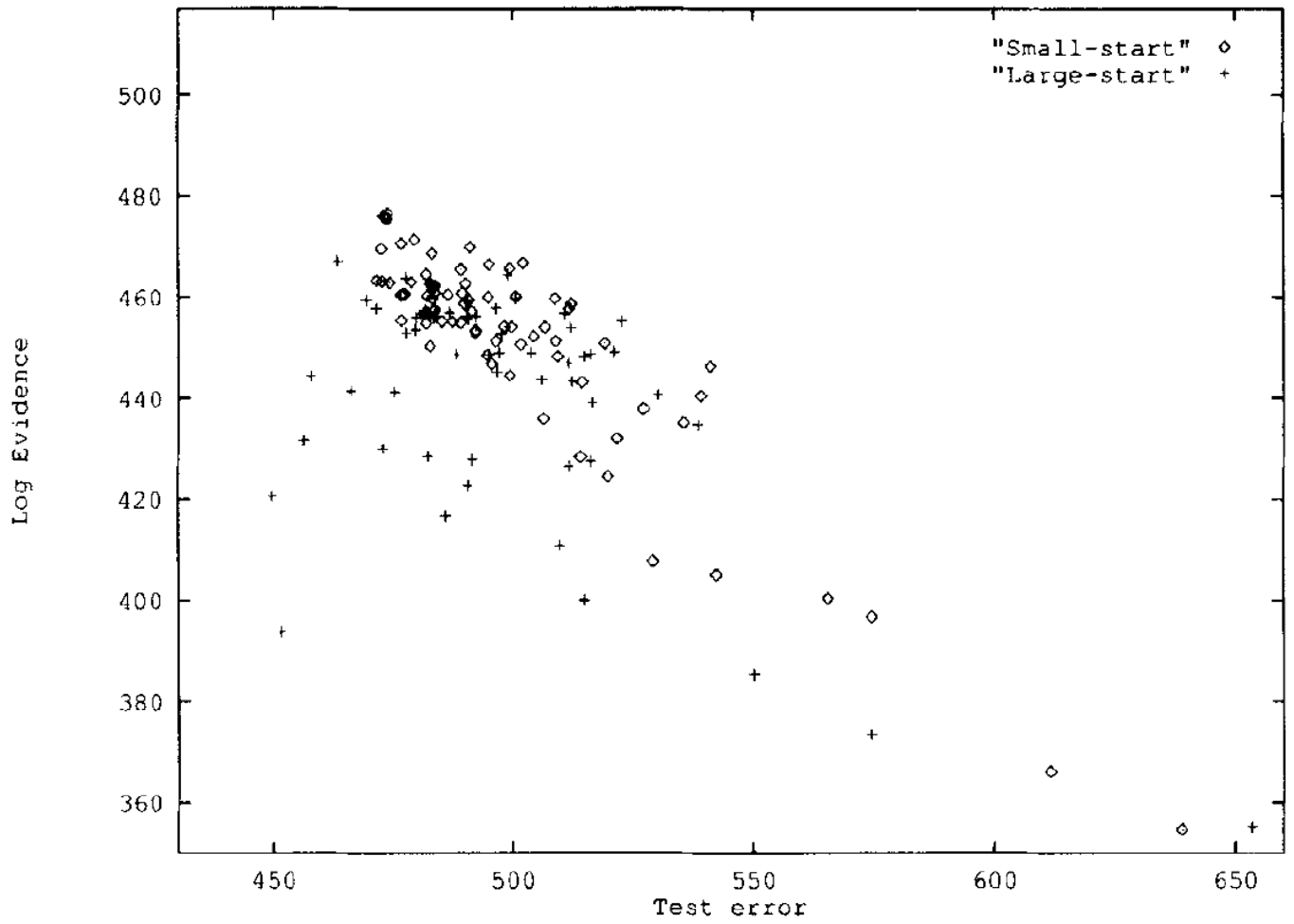
Demonstration – Well-Determined Parameters



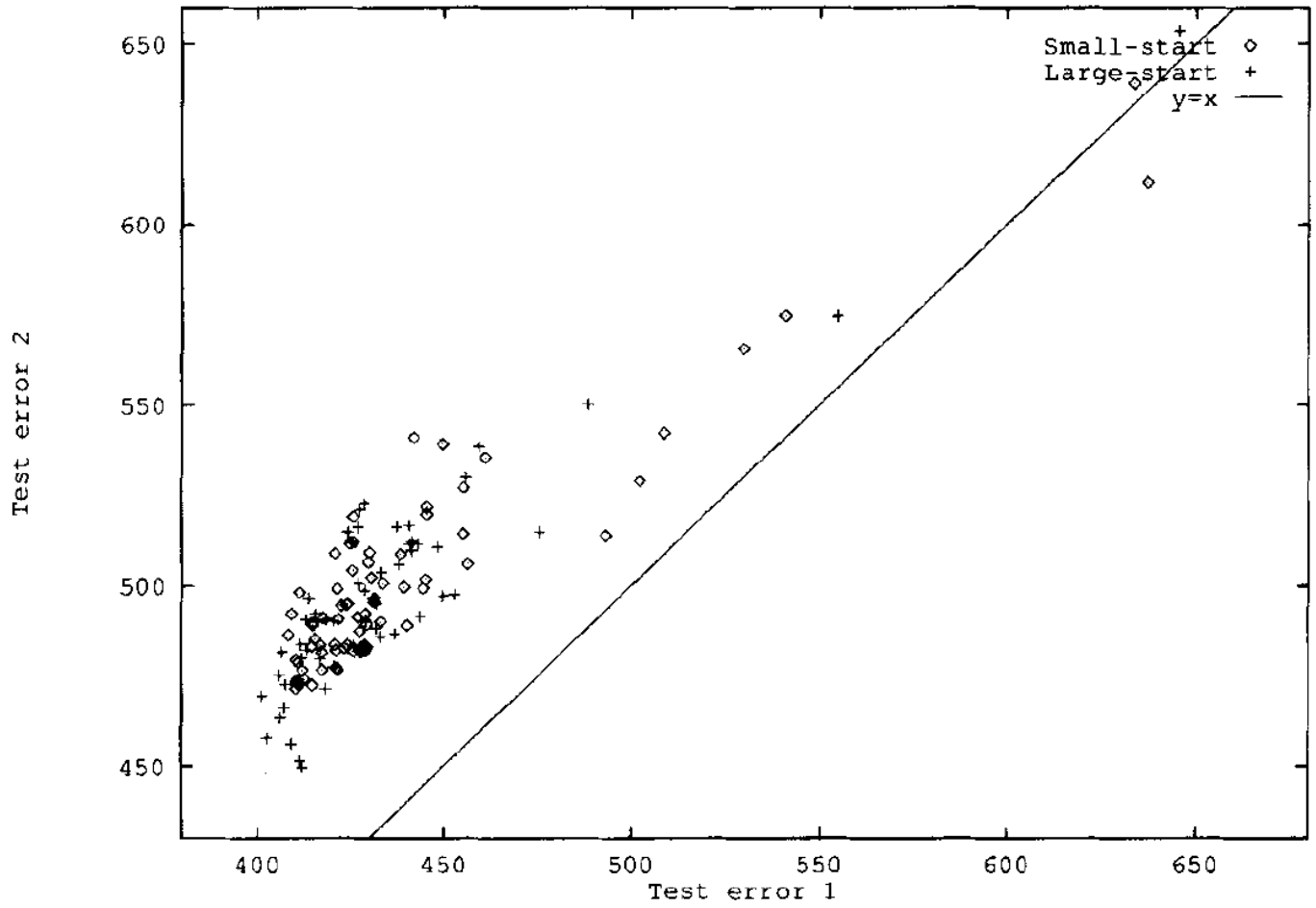
Demonstration – Well-Determined Parameters



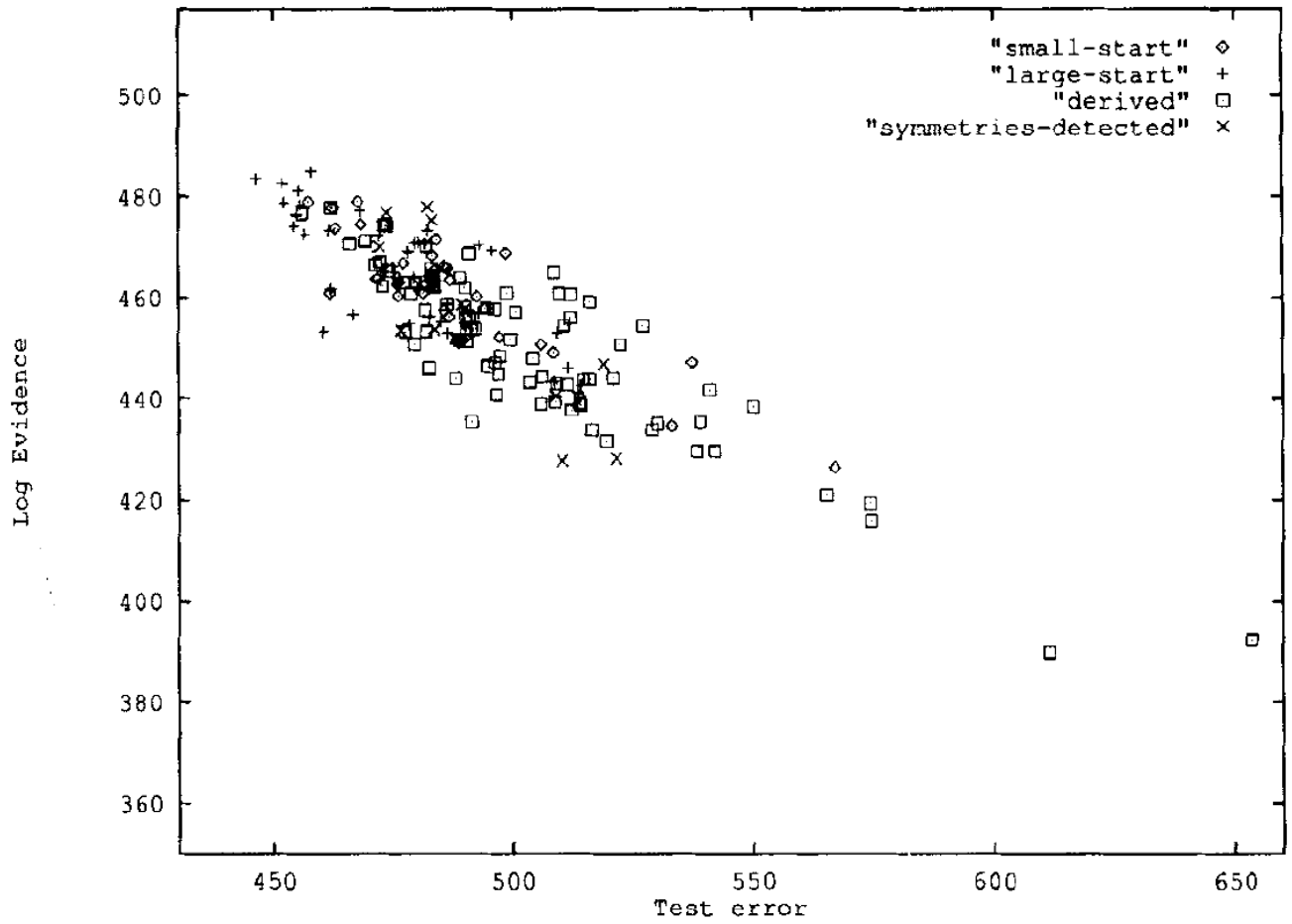
Demonstration – High Variance Prior w



Demonstration – 1st Prior vs. 2nd Prior



Demonstration – Logging 2nd Prior



Last Word

- Since Bayesian approximation is heavily based on the assumption that mentioned probabilities have a Gaussian distribution, the evaluation of evidence breaks down significantly for N/k < 3 +/- 1.

- Thank you!