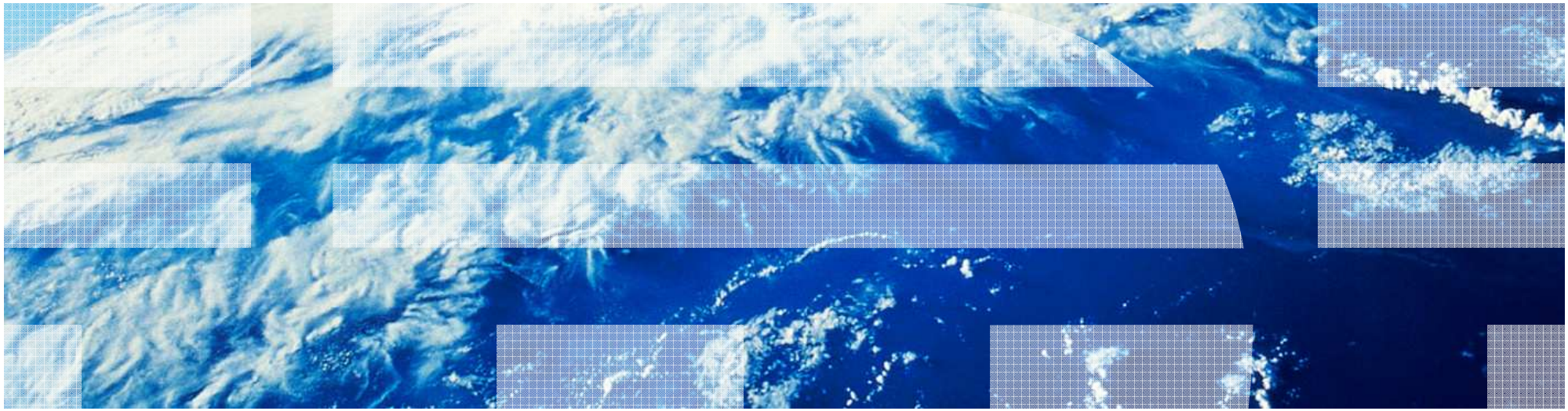
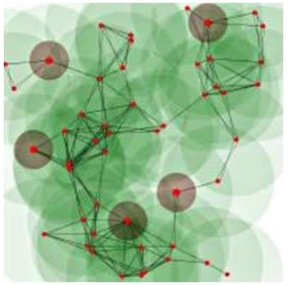
Kush R. Varshney – Thomas J. Watson Research Center – krvarshn@us.ibm.com

25 May 2011

Spatially-Correlated Sensor Discriminant Analysis



Sensor Networks



- Collection of spatially-distributed nodes that take measurements and communicate
- Detection or classification tasks
  - environmental monitoring
  - surveillance
- Measurements
  - temperature, sound, pressure, vibration, etc.
  - exhibit spatial correlation
- Resource-constrained

photographs from http://web.engr.oregonstate.edu/~thinhq/sensor_network.jpg and Zong, Houser, and Damarla (2006)

Distributed Detection or Classification

- Detection problem
  - use measurements to make decision about binary hypothesis
  - known likelihood functions of measurements and known prior probabilities
  - likelihood ratio test is Bayes optimal
  - distributed detection has much previous work
- Supervised classification problem
  - use measurements to make decision about binary hypothesis
  - unknown likelihood functions of measurements and unknown prior probabilities
  - given labeled training set: measurement vectors with their true hypotheses
  - machine learning algorithms: support vector machines, *linear discriminant analysis*, etc.
  - overfitting
  - distributed supervised classification does not have as much previous work
- Practically
  - in practice, oftentimes don't have knowledge of probability density a priori
  - easier to imagine situation in which training samples can be acquired

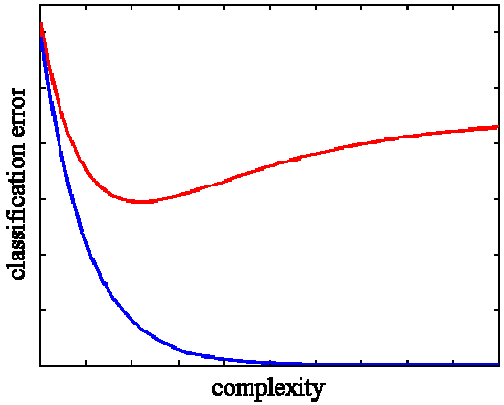
Outline

- Supervised classification problem and modern statistical learning theory
- Linear discriminant analysis generalization error approximations
- Gauss-Markov random field sensor model
- Mahalanobis distance and generalization error in sensor network
- Geometric probability approximation to Mahalanobis distance
- Simulations

Supervised Classification Notation

- True hypothesis or class label $Y \in \{-1,+1\}$
- Noisy measurement $\mathbf{X} \in \Omega \subseteq \mathbb{R}^p$
- Joint probability density function $f_{\mathbf{X},Y}(\mathbf{x},y)$
- Decision rule $\hat{y}(\mathbf{x}) : \Omega \rightarrow \{-1,+1\}$
  - learned using training samples $\{(\mathbf{x}_1,y_1), \ldots, (\mathbf{x}_n,y_n)\}$, but applied to unseen samples $\mathbf{x}$

- Generalization error: $\Pr[\hat{y}(\mathbf{X}) \neq Y]$
- Training error: $\frac{1}{n}\sum_{j=1}^{n} \mathbf{I}\left(\hat{y}\left(\mathbf{x}_j\right) \neq y_j\right)$
- Overfitting and structural risk minimization principle (Vapnik, 2000)



red: generalization error

blue: training error

Statistical Learning Theory Analysis

- Characterizations of generalization error in modern statistical learning theory are very loose
- "one should not be concerned about the quantitative value of the bound or even about its fundamental form but rather about the terms that appear in the bound. In that respect a useful bound is one which allows to understand which quantities are involved in the learning process. As a result, performance bounds should be used for what they are good for. They should not be used to actually predict the value of the expected error. Indeed, they usually contain prohibitive constants or extra terms that are mostly mathematical artifacts. They should not be used directly as a criterion to optimize since their precise functional form … may also be a mathematical artifact." (Bousquet, 2003)
- What if we want to actually predict the value of the generalization error and use it as an optimization criterion?
  - useful for sensor network system design
  - cannot use modern statistical learning theory results

Former Soviet Union Literature

- Very high quality approximations to generalization error of linear discriminant analysis and its variations
- Motivated by Kolmogorov
- Main contributor: Šarūnas Raudys
- Provide expressions that could be used as optimization criteria

- Linear discriminant analysis:

$$\hat{y}(\mathbf{x}) = \text{sign}\left(\mathbf{x}^T \left(\hat{\boldsymbol{\Sigma}}_- + \hat{\boldsymbol{\Sigma}}_+\right)^{-1} \left(\hat{\boldsymbol{\mu}}_+ - \hat{\boldsymbol{\mu}}_-\right) - \tfrac{1}{2} \left(\hat{\boldsymbol{\mu}}_- + \hat{\boldsymbol{\mu}}_+\right)^T \left(\hat{\boldsymbol{\Sigma}}_- + \hat{\boldsymbol{\Sigma}}_+\right)^{-1} \left(\hat{\boldsymbol{\mu}}_+ - \hat{\boldsymbol{\mu}}_-\right) \right)$$

  – with sample means and sample covariances of the two classes

Generalization Error Approximation and Mahalanobis Distance

- When true likelihood functions are Gaussian and true prior probabilities are equal
  - not known when learning $\hat{y}$ from n training samples
- Generalization error approximation:

$$\Pr\left[\hat{y}(\mathbf{X}) \neq Y\right] \approx \Phi\left(-\frac{\delta}{2}\left[\left(1+\frac{4p}{n\delta^2}\right)\frac{n}{n-p}\right]^{-1/2}\right)$$

  - where Φ is Gaussian cdf and δ is a Mahalanobis distance such that

$$\delta^2 = (\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-)^T \left(\frac{\mathbf{J}_- + \mathbf{J}_+}{2}\right)(\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-)$$

  - and $\mathbf{J}_- = \boldsymbol{\Sigma}_-^{-1}$ and $\mathbf{J}_+ = \boldsymbol{\Sigma}_+^{-1}$ are the inverse covariance matrices

- Use this generalization error approximation to analyze sensor networks with spatial correlation
  - not concerned with network/communication/power issues at this point

Sensor Network Setup

- p sensors each with scalar-valued measurements $X_1, \ldots, X_p$
  - combined measurement vector $\mathbf{X} \in \mathbb{R}^p$
- Deployed randomly on the plane according to $f_{\mathbf{V}}(\mathbf{v})$
  - $f_{\mathbf{V}}(\mathbf{v})$ supported on a square with area p
  - region grows as more sensors are placed
- True Gaussian likelihood functions $f_{\mathbf{X}|Y}(\mathbf{x} \mid Y = -1) \sim N(\boldsymbol{\mu}_-, \boldsymbol{\Sigma}_-)$ and $f_{\mathbf{X}|Y}(\mathbf{x} \mid Y = +1) \sim N(\boldsymbol{\mu}_+, \boldsymbol{\Sigma}_+)$
- Covariance structure models spatial correlation between sensor measurements
  - Gauss-Markov random field model
- For simplicity:
  - let $\boldsymbol{\mu}_- = \mathbf{0}$ and $\boldsymbol{\mu}_+ = \mathbf{1}$
  - let $\boldsymbol{\Sigma}_- = \boldsymbol{\Sigma}_+ = \boldsymbol{\Sigma}$ and thus also $\mathbf{J}_- = \mathbf{J}_+ = \mathbf{J}$

Gauss-Markov Random Field with Nearest Neighbor Dependency

- Euclidean (undirected) nearest neighbor graph
  - edge between sensor *i* and *j* if *i* is nearest neighbor of *j* or *j* is nearest neighbor of *i*
  - denote edge set as *E*
- Markov random field covariance
- Diagonal elements of $\boldsymbol{\Sigma}$ all equal to σ^2
- Elements of $\boldsymbol{\Sigma}$ corresponding to edges in nearest neighbor graph:

$$\{\boldsymbol{\Sigma}\}_{ij} = \sigma^2 \, g\left(d\left(\mathbf{v}_i, \mathbf{v}_j\right)\right), \quad (i,j) \in E$$

  - $g(\cdot) : \mathbb{R}^+ \rightarrow (0,1)$ monotonically decreasing function to encode correlation decay with distance – known as semivariogram in geostatistics
- Off-diagonal elements of J corresponding to non-edges in nearest neighbor graph:

$$\{\mathbf{J}\}_{ij} = 0, \quad i \neq j, \ (i,j) \notin E$$

- Model used by e.g. Anandkumar, Tong, and Swami (2009)

Mahalanobis Distance for Gauss-Markov Random Field with Nearest Neighbor Dependency

- $\delta^2 = (\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-)^T \left(\frac{\mathbf{J}_- + \mathbf{J}_+}{2} \right) (\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-)$
- With simplifying assumptions $\boldsymbol{\mu}_- = \mathbf{0}$, $\boldsymbol{\mu}_+ = \mathbf{1}$, $\mathbf{J}_- = \mathbf{J}_+ = \mathbf{J}$: $\delta^2 = \sum_{i=1}^{p} \sum_{j=1}^{p} \{\mathbf{J}\}_{ij}$
- After substituting covariances of model and algebra: $\delta^2 = \frac{p}{\sigma^2} - \frac{2}{\sigma^2} \sum_{(i,j) \in E} \frac{g(d(\mathbf{v}_i, \mathbf{v}_j))}{1 + g(d(\mathbf{v}_i, \mathbf{v}_j))}$

Mahalanobis Distance Approximation by Geometric Probability

- Mahalanobis distance depends on d($\mathbf{v}_i$,$\mathbf{v}_j$), which depends on particular realization of random deployment of sensor locations
- Characterize average behavior of δ across realizations of {$\mathbf{v}_1$, …, $\mathbf{v}_p$}
- Average behavior of functionals of the nearest neighbor graph can be described using average behavior of homogenous Poisson point processes (Penrose and Yukich, 2003)
- Specifically, as *p* goes to infinity:

$$\frac{1}{p}\sum_{(i,j)\in E}\varphi\left(d\left(\mathbf{v}_i,\mathbf{v}_j\right)\right)\rightarrow\frac{1}{2}\int E\left[\sum_{(0,a)\in F}\varphi\left(\frac{d(\mathbf{0},\mathbf{w}_a)}{\sqrt{f_{\mathbf{v}}(\mathbf{v})}}\right)\right]d\mathbf{v}$$

- For us, φ(·) is g(·)/(1+g(·))
- Let *ζ*/2 equal the right side of limit, then:

$$\delta^2\approx\frac{p}{\sigma^2}(1-\zeta)$$

Optimal Number of Sensors

- Substituting Mahalanobis distance approximation into generalization error approximation:

$$\Pr\left[\hat{y}(\mathbf{X}) \neq Y\right] \approx \Phi\left(-\sqrt{\frac{p(1-\zeta)}{4\sigma^2}\left[\left(1+\frac{4\sigma^2}{n(1-\zeta)}\right)\frac{n}{n-p}\right]^{-1/2}}\right)$$

- Find p that minimizes generalization error approximation
  - differentiate with respect to p and set to zero
- Optimal p is $n/2$, irrespective of σ^2 and ζ
  - don't always throw down more sensors
  - overfitting
- Minimum generalization error is:

$$\Pr\left[\hat{y}(\mathbf{X}) \neq Y\right] \approx \Phi\left(-\frac{1}{4}\sqrt{\frac{n^2(1-\zeta)^2}{n\sigma^2(1-\zeta)+4\sigma^2}}\right)$$

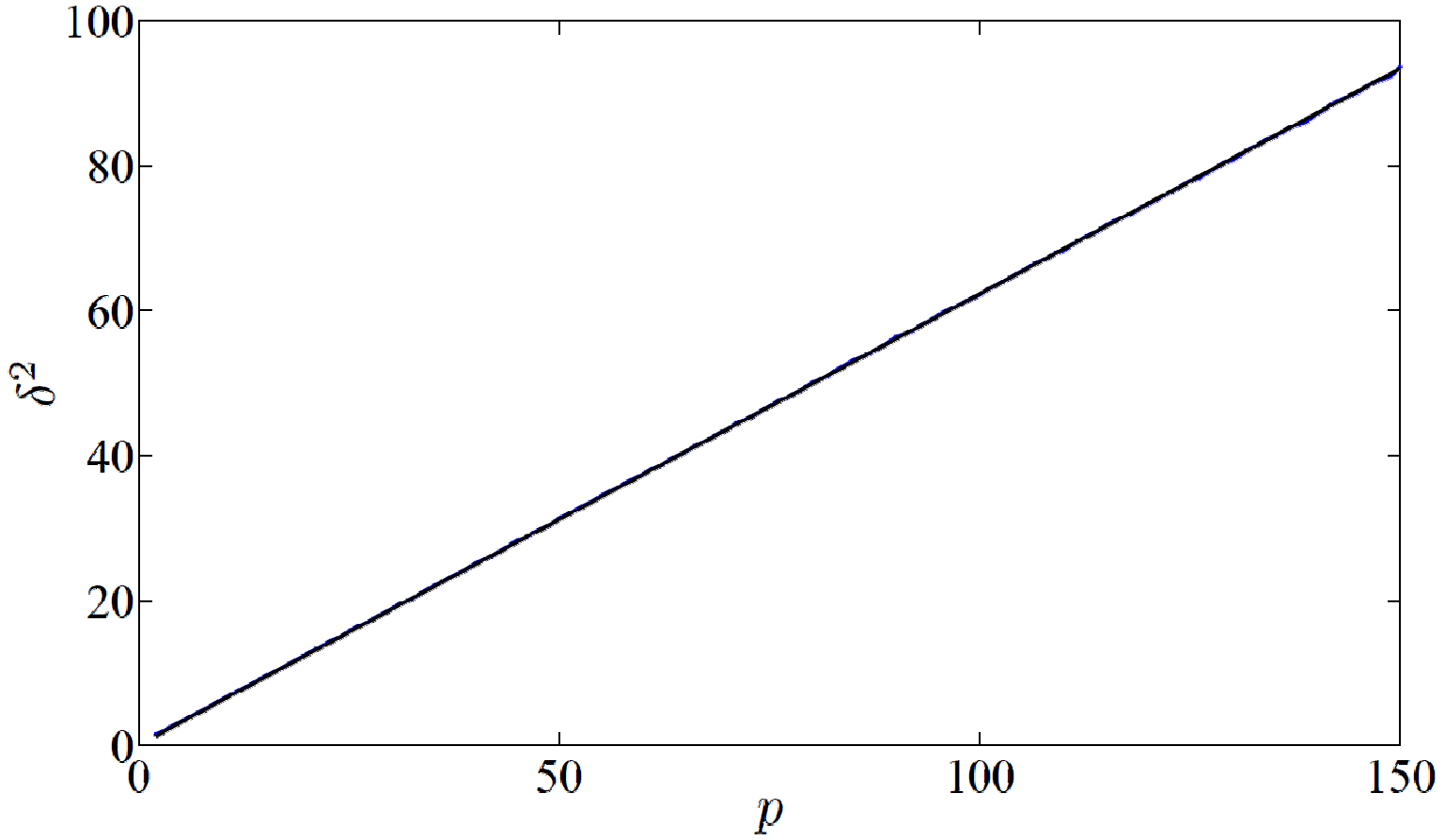
- Minimum generalization error
  - monotonically increasing in σ^2
  - monotonically decreasing in n
  - monotonically increasing in ζ □ [0,1/2]
    - sensor placement distribution $f_{\mathbf{V}}(\mathbf{v})$ should be chosen to minimize ζ

Simulations

- Overall message: these are really good approximations that can be used for system optimization
- Semivariogram g(d) = 0.5exp(–d/2)
- Sensor location distribution with support over square with area p
  - appropriately scaled and shifted beta distribution, i.i.d. in both spatial dimensions
  - both parameters taken to equal β
    - $\beta = 1$: uniform
    - $\beta > 1$: concentrated in middle of square
    - $\beta < 1$: concentrated at edges of square
- For different values of p
  - 20 realizations of **V**
  - 10 realizations of training set per realization of **V**
    - different values of n
  - 100000 test samples per training set

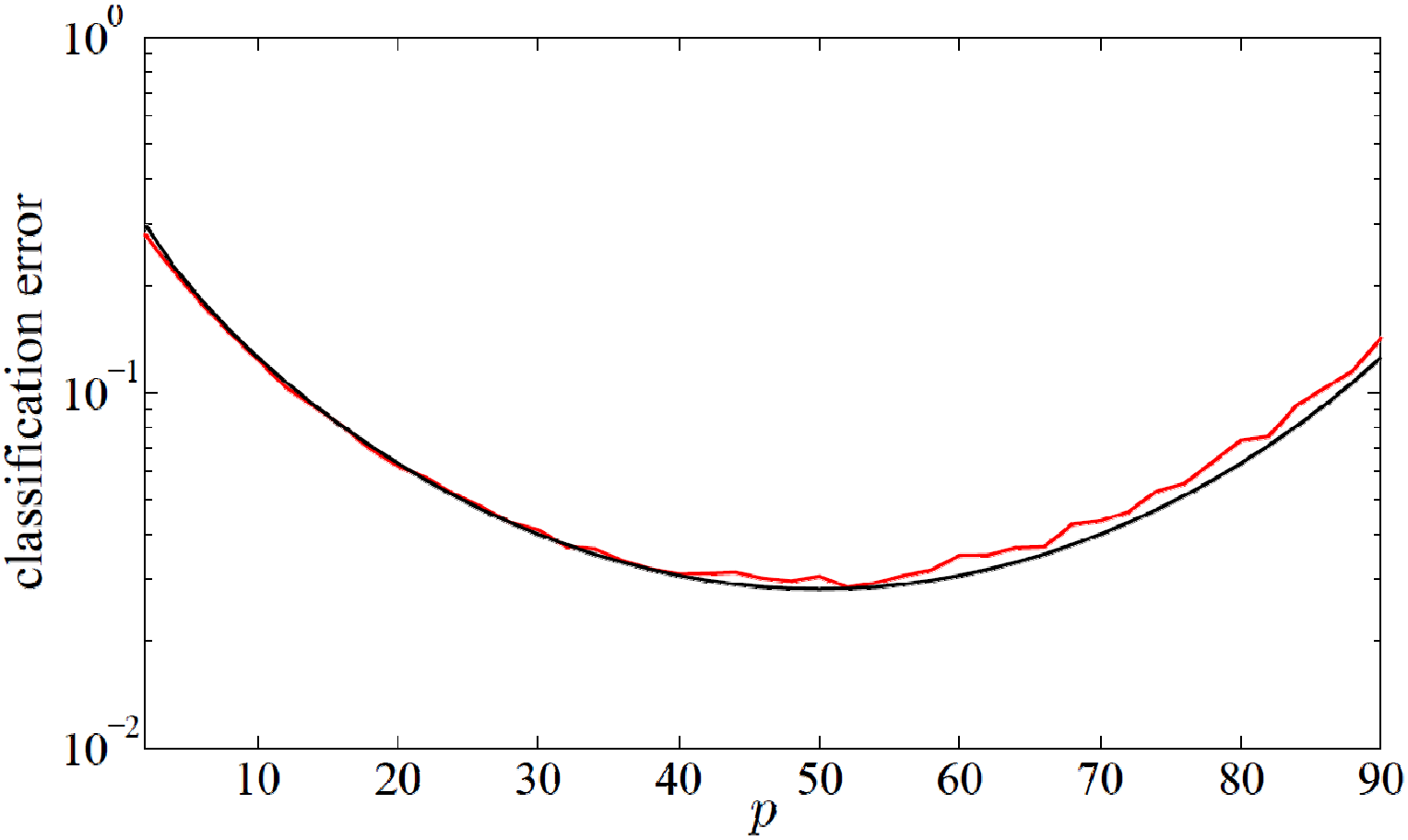
Mahalanobis Distance Approximation

- Blue line: empirical
- Black line: approximation



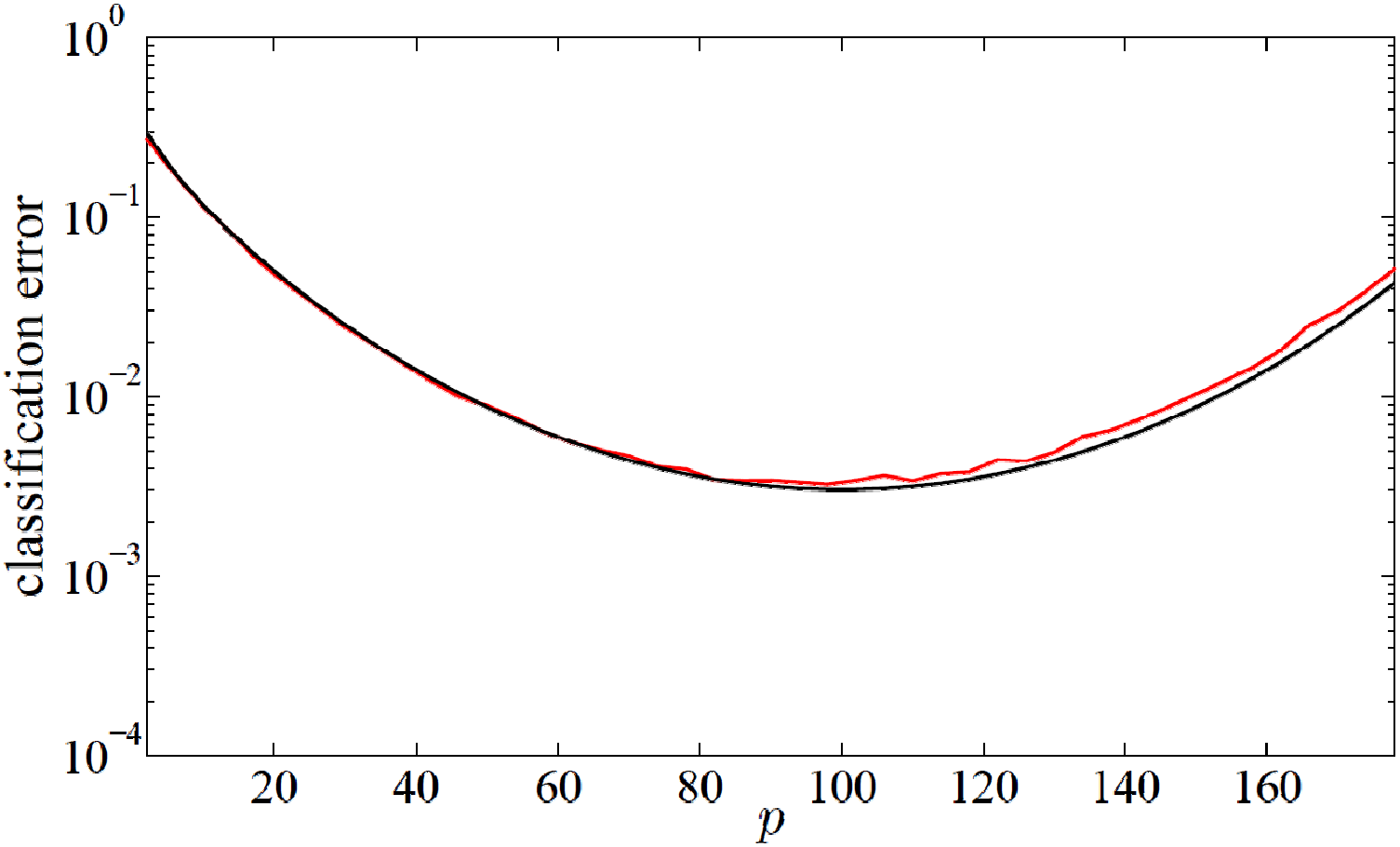
Generalization Error Approximation (n = 100)

- Red line: empirical
- Black line: approximation



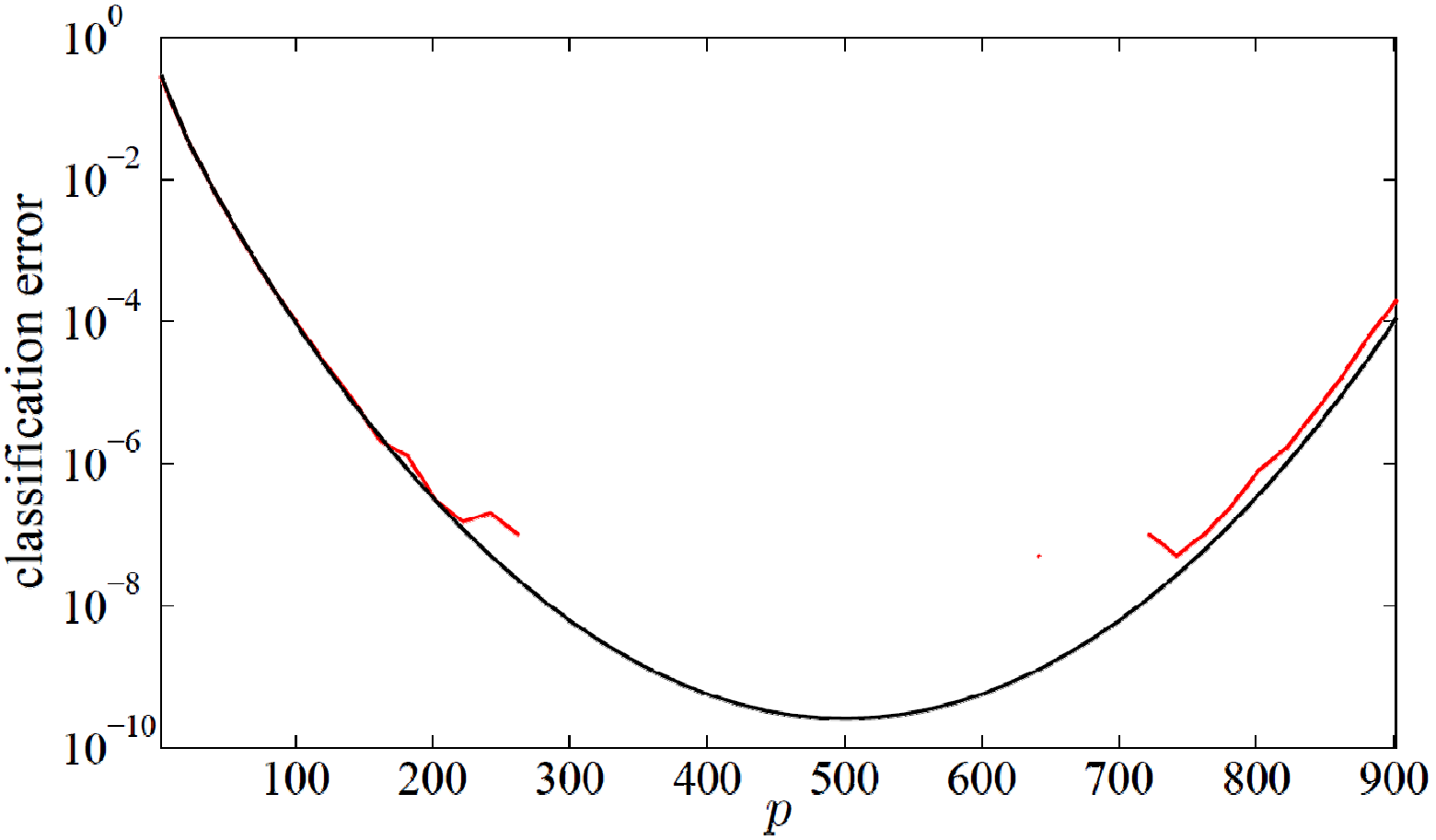
Generalization Error Approximation (n = 200)

- Red line: empirical
- Black line: approximation



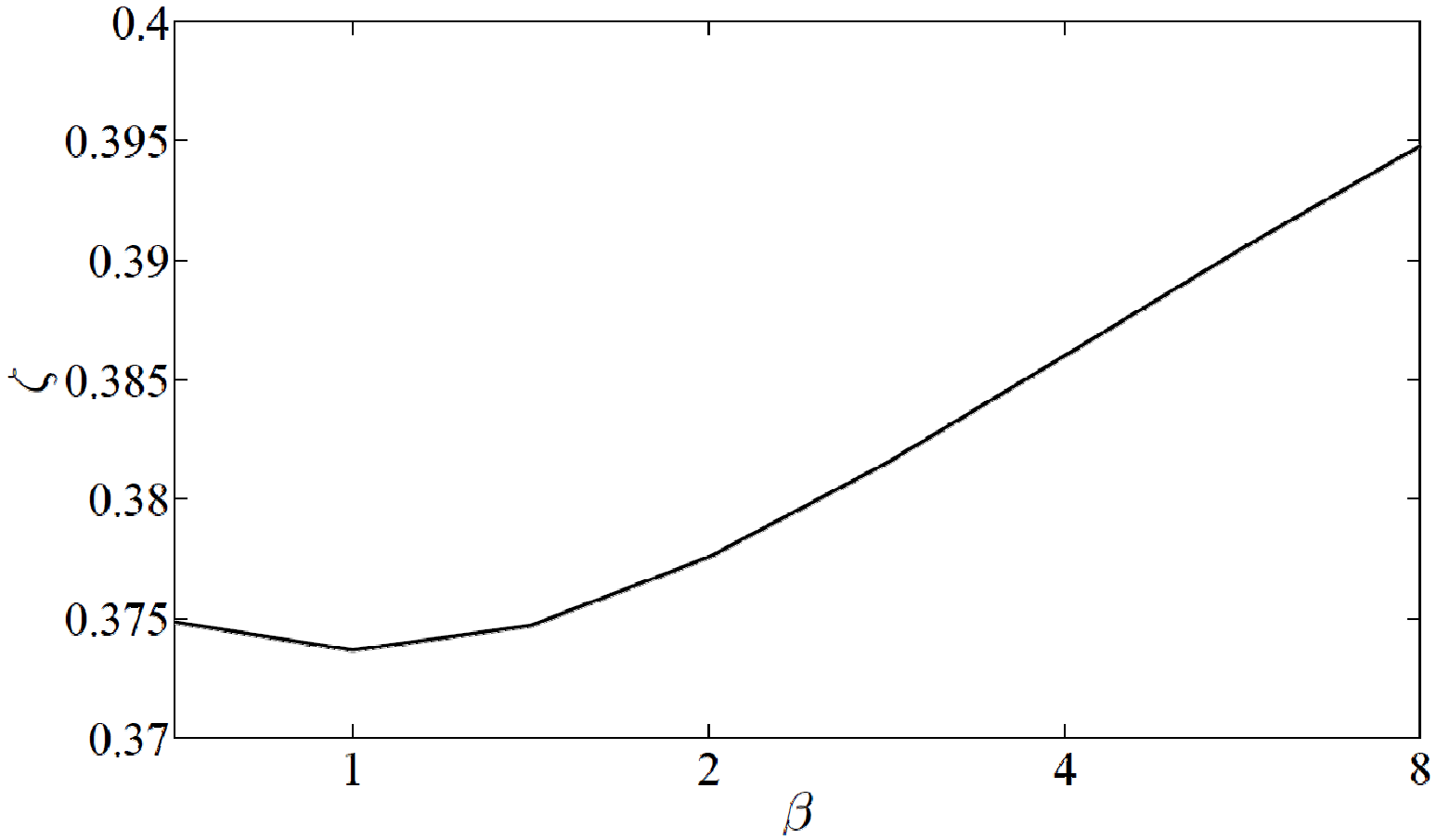
Generalization Error Approximation (n = 1000)

- Red line: empirical
- Black line: approximation



Different Sensor Placement Distributions

- Uniform is best



Conclusion

- Time always limited resource, so training set always finite
- Optimal to use precisely half the number of sensors as training samples in sensor network with local Gauss-Markov dependency
- Finite rather than infinite number of sensors follows from overfitting

- Generalization error approximation that involves Mahalanobis distance
- Exact statement of Mahalanobis distance for Gauss-Markov sensor measurements
- Approximation of Mahalanobis distance using geometric probability
- Approximations closely match empirical results
- Uniform sensor placement is good

Questions