Applied Bayesian Nonparametrics

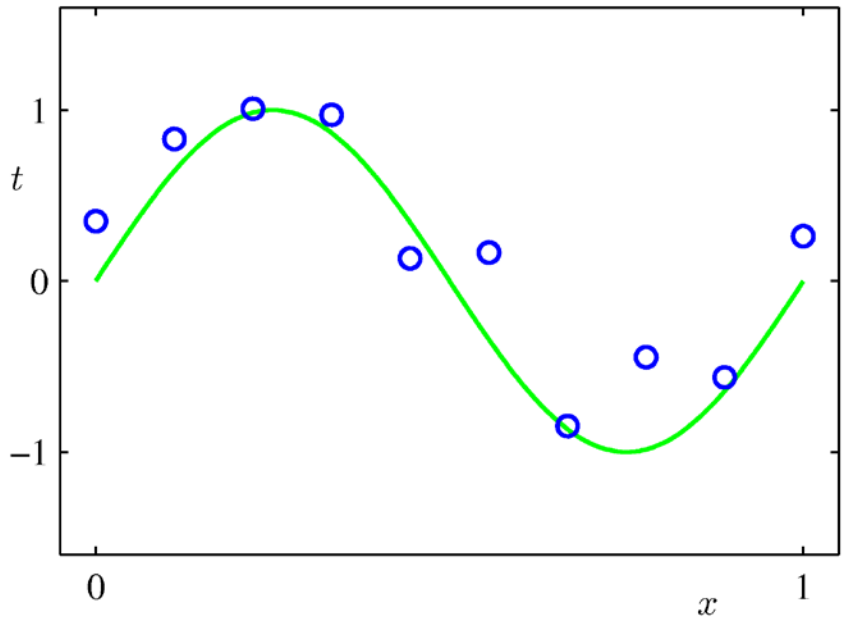
Special Topics in Machine Learning
Brown University CSCI 2950-P, Fall 2011

September 13: Gaussian Processes for Regression & Classification

Many figures courtesy Kevin Murphy's textbook
Machine Learning: A Probabilistic Perspective,
and Chris Bishop's textbook
Pattern Recognition and Machine Learning

Linear Basis Function Models (1)

- Example: Polynomial Curve Fitting



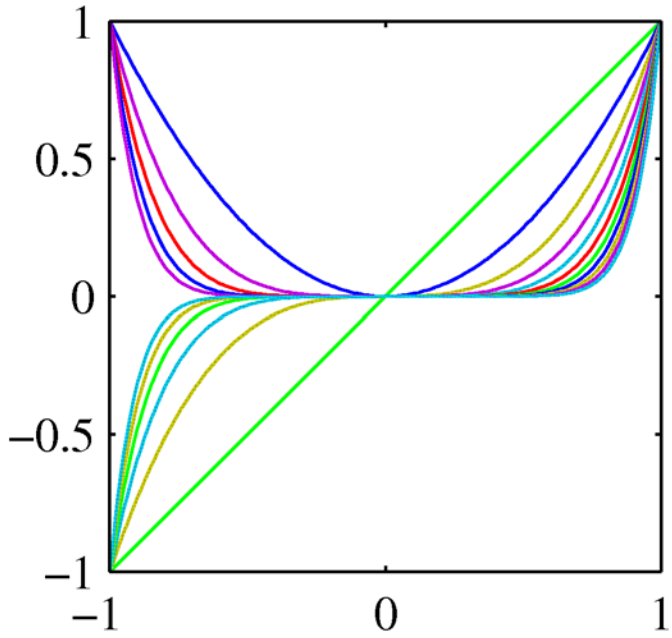
$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

Linear Basis Function Models (2)

- Polynomial basis functions:

$$\phi_j(x) = x^j.$$

- These are global; a small change in x affect all basis functions.

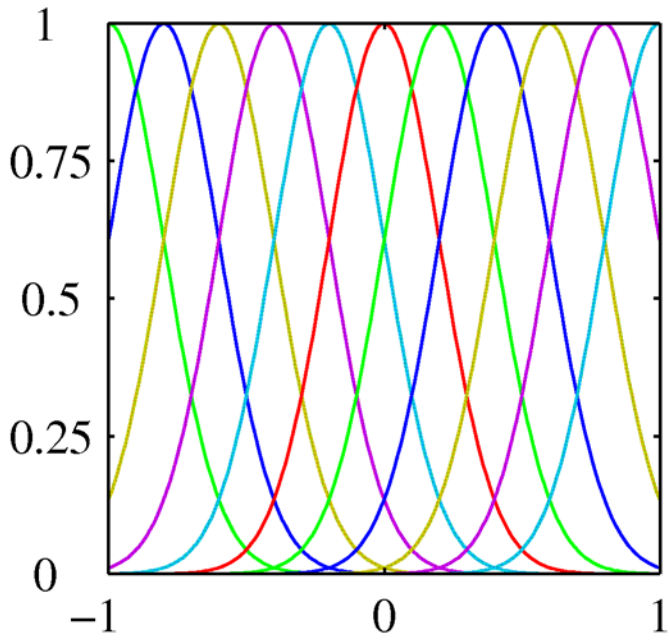


Linear Basis Function Models (3)

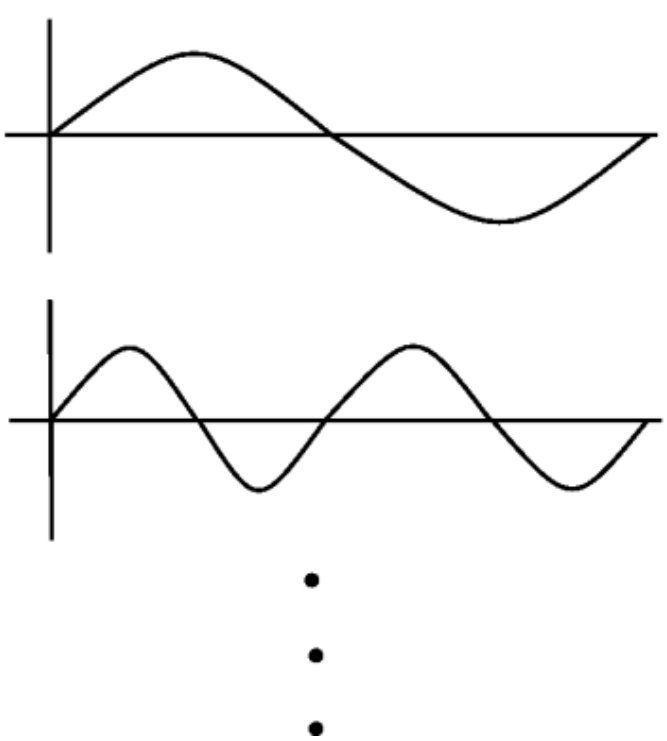
- Gaussian basis functions:

$$\phi_j(x) = \exp\left\{-\frac{(x-\mu_j)^2}{2s^2}\right\}$$

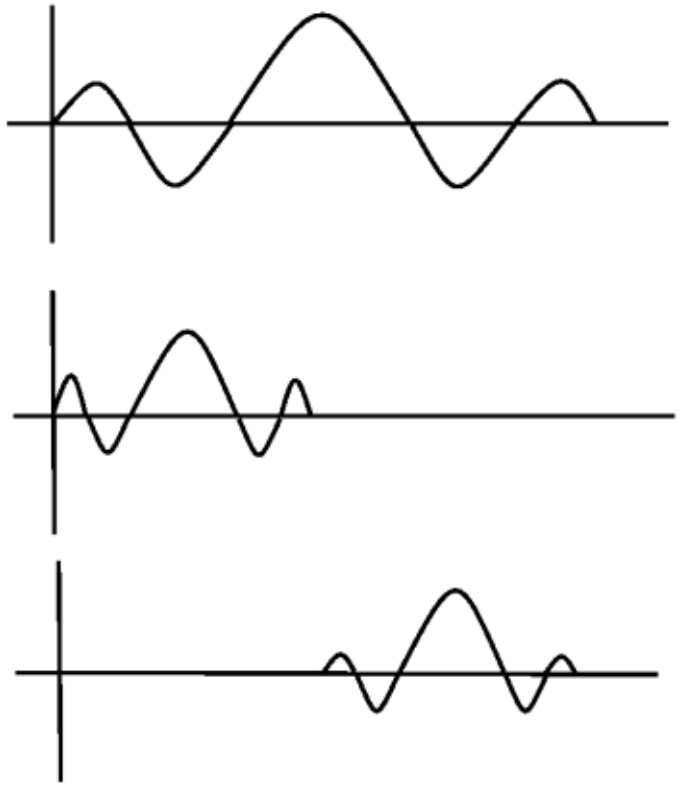
- These are local; a small change in x only affect nearby basis functions. Parameters control location and scale (width).



Linear Basis Function Models (4)

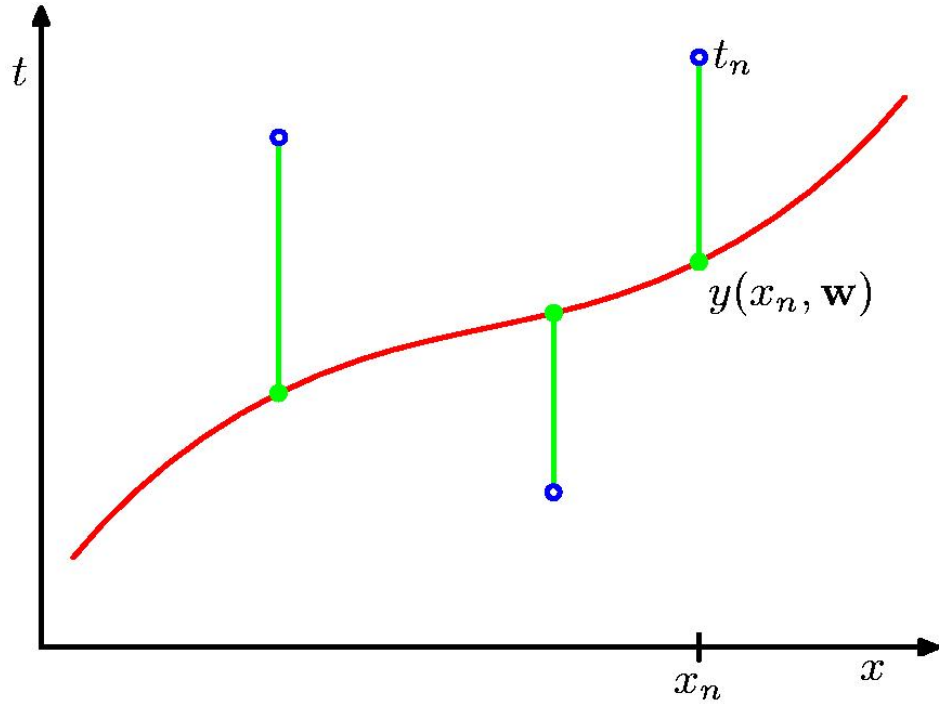


Fourier Basis



Wavelet Basis

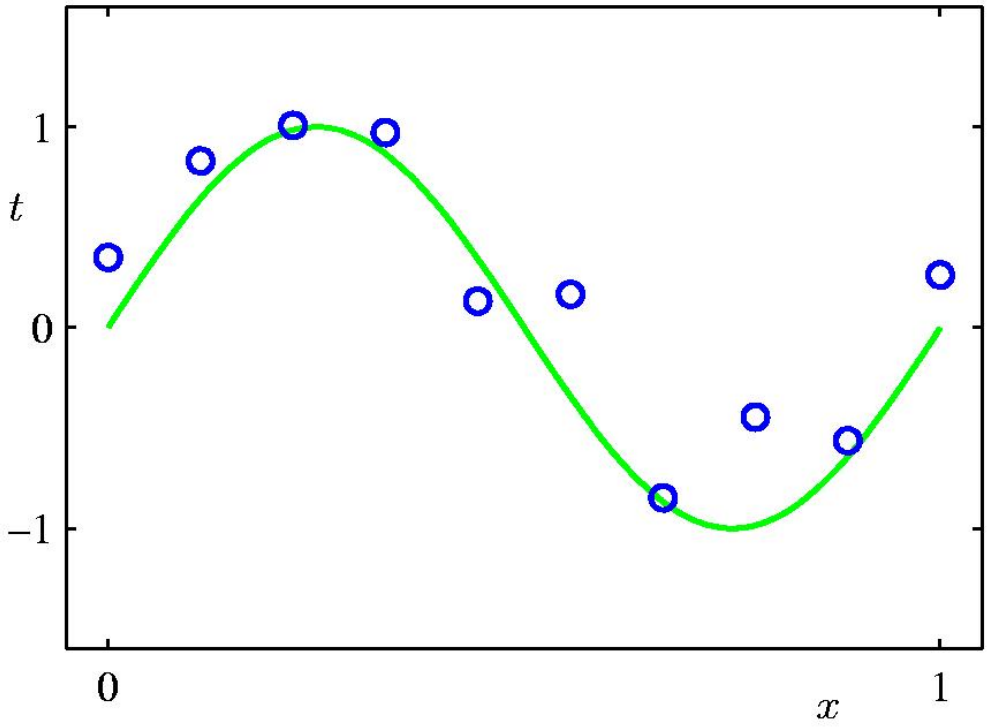
Sum-of-Squares Error Function



$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\}^2$$

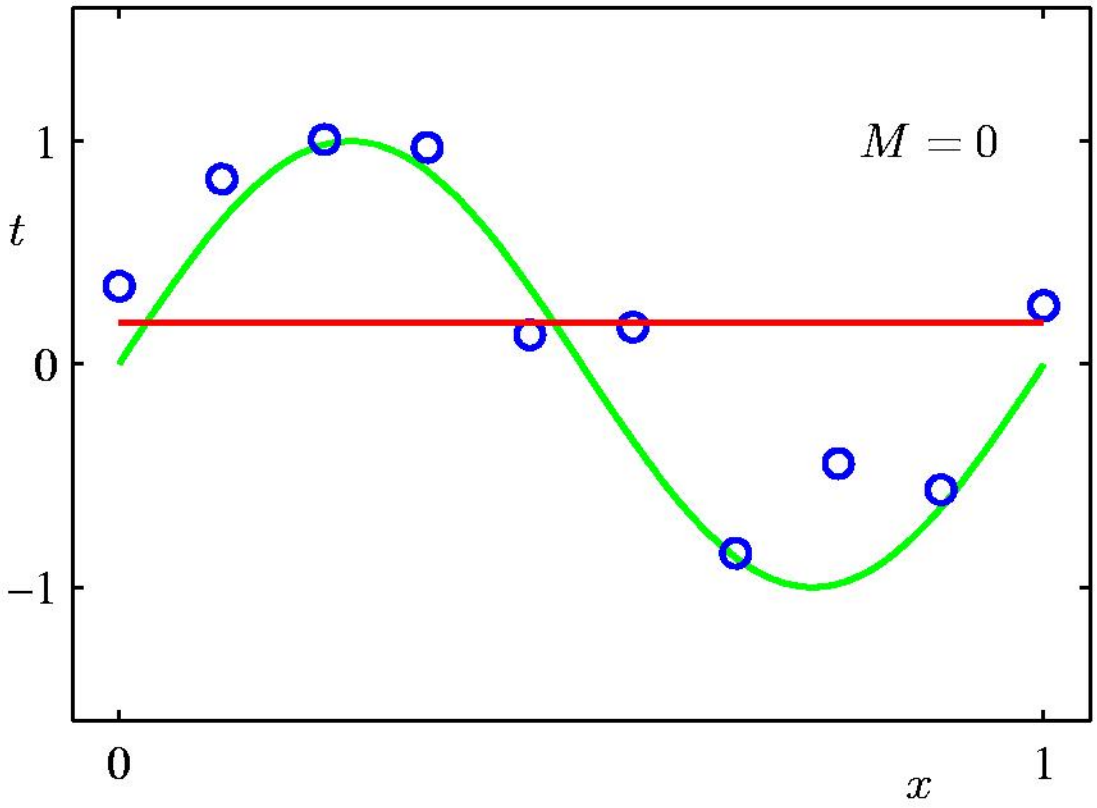
Board: Least squares and the normal equations

Polynomial Curve Fitting

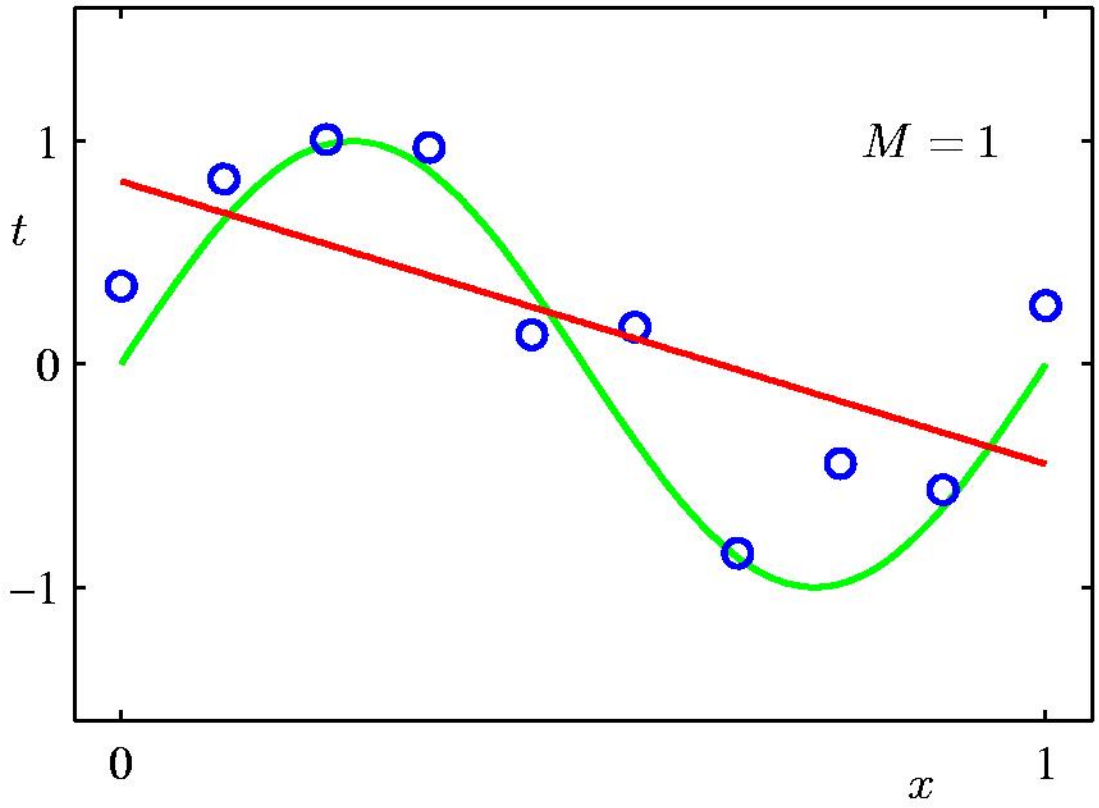


$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

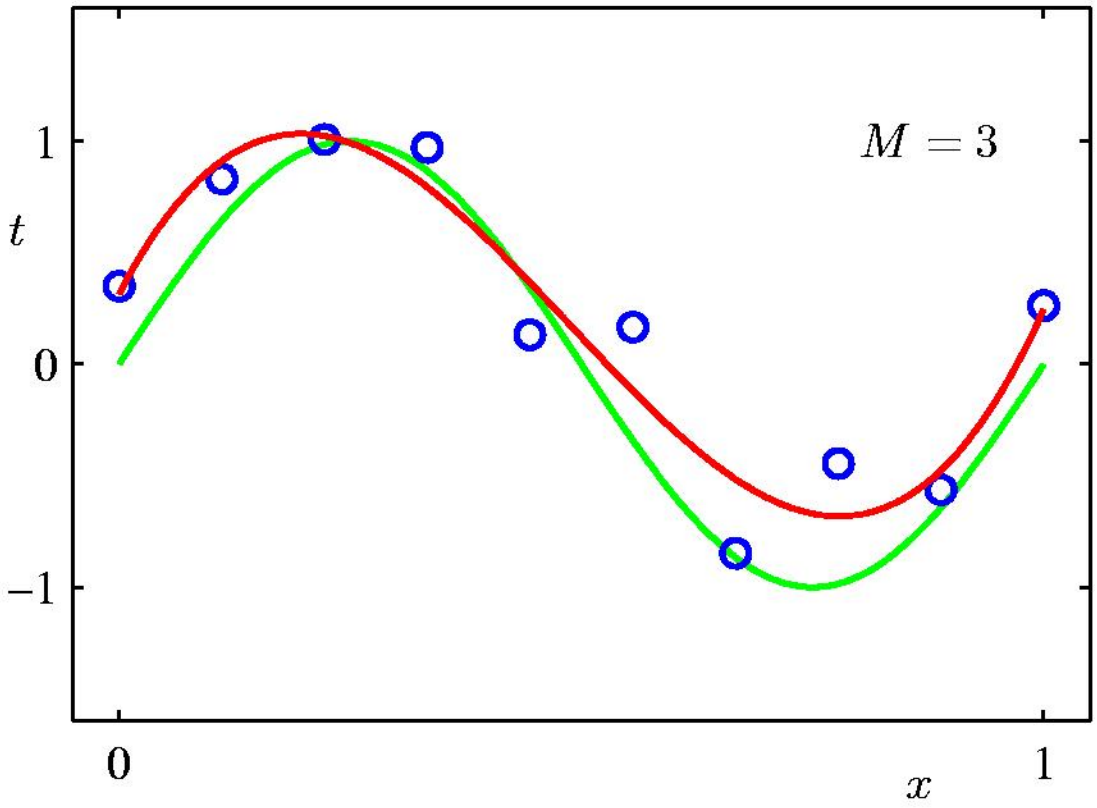
0^{th} Order Polynomial



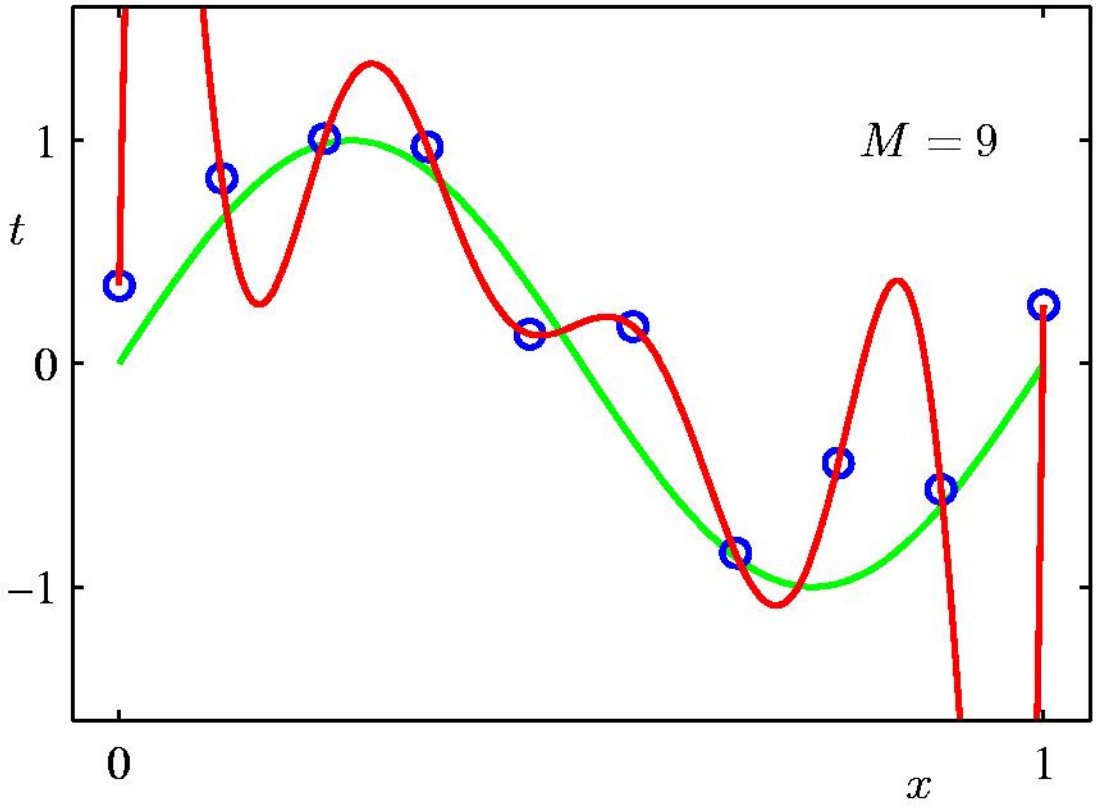
1st Order Polynomial



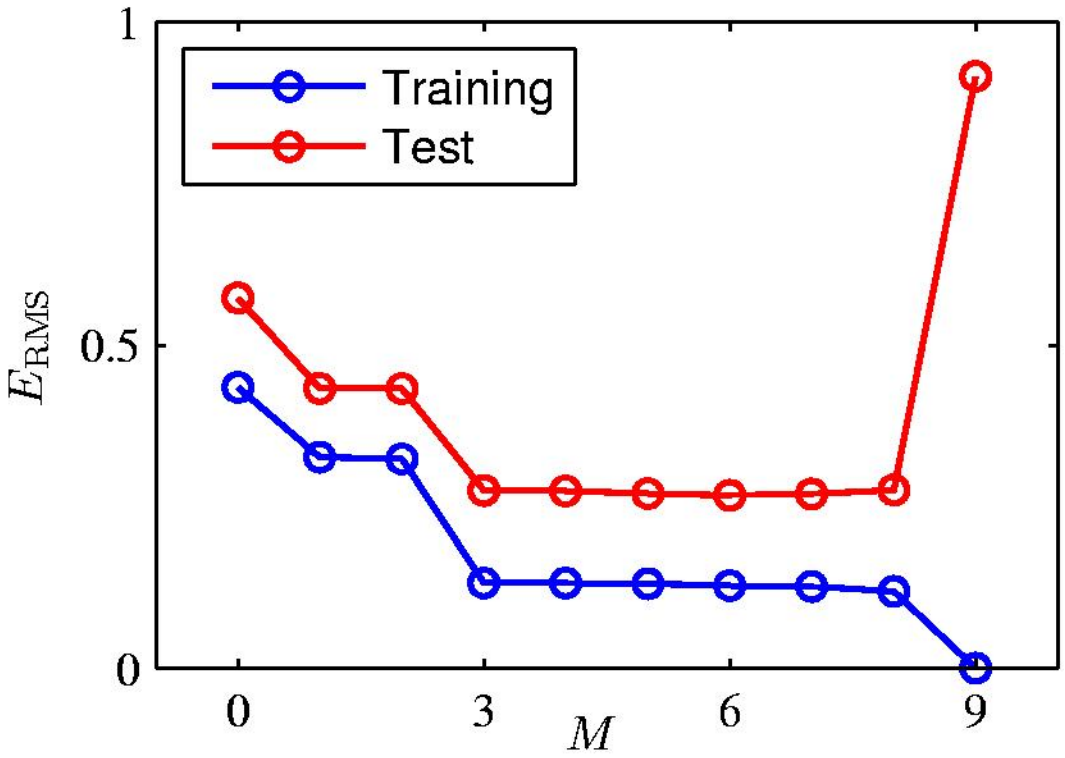
3rd Order Polynomial



9th Order Polynomial



Over-fitting



Root-Mean-Square (RMS) Error: $E_{\mathrm{RMS}} = \sqrt{2E(\mathbf{w}^{\star})/N}$

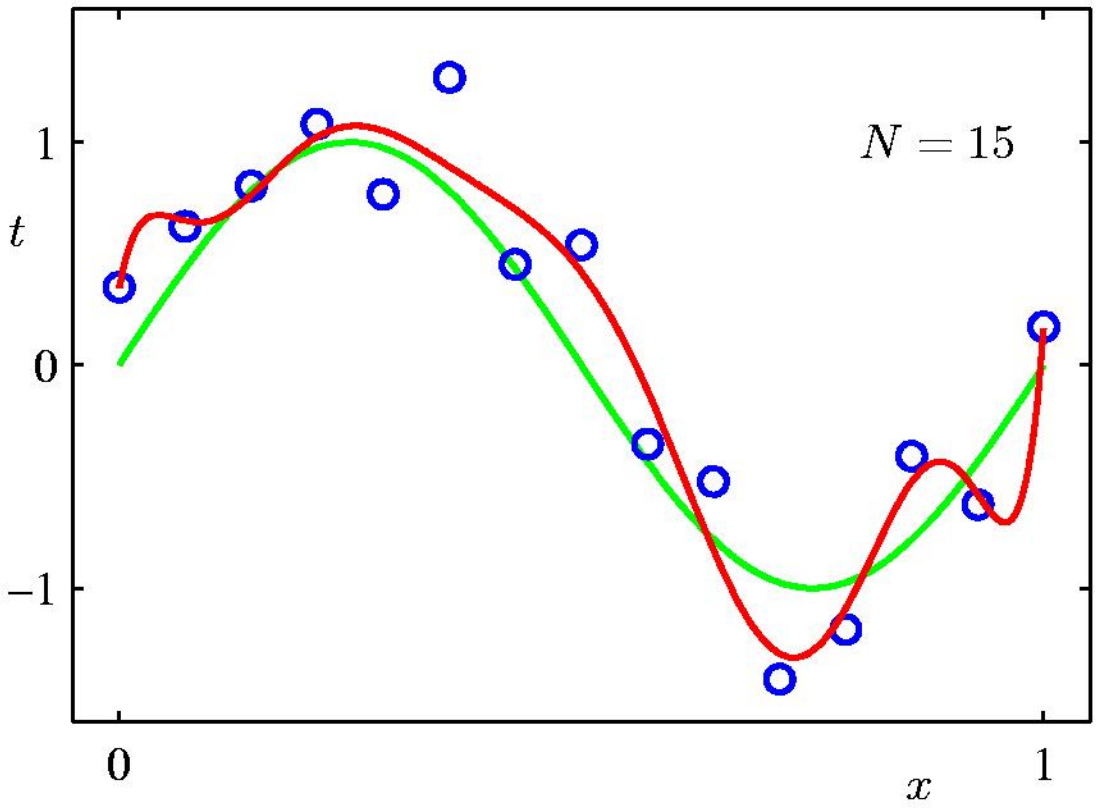
Polynomial Coefficients

	$M=0$	$M=1$	$M=3$	$M=9$
$w_0^\star$	0.19	0.82	0.31	0.35
$w_1^\star$		-1.27	7.99	232.37
$w_2^\star$			-25.43	-5321.83
$w_3^\star$			17.37	48568.31
$w_4^\star$				-231639.30
$w_5^\star$				640042.26
$w_6^\star$				-1061800.52
$w_7^\star$				1042400.18
$w_8^\star$				-557682.99
$w_9^\star$				125201.43

Good priors oppose "extreme" model configurations

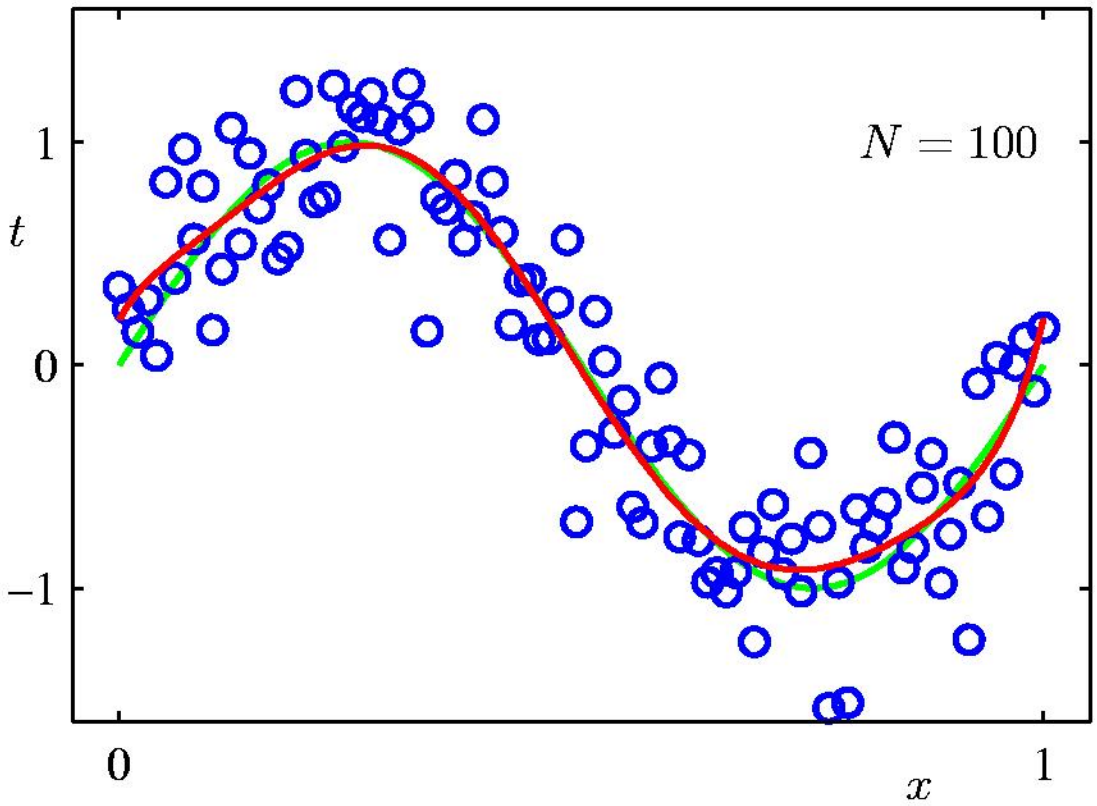
Data Set Size: $N = 15$

9th Order Polynomial



Data Set Size: $N = 100$

9th Order Polynomial



Model complexity can grow as additional data is observed

Regularized Least Squares (1)

- Consider the error function:

$$E_D(\mathbf{w}) + \lambda E_W(\mathbf{w})$$

Data term + Regularization term

- With the sum-of-squares error function and a quadratic regularizer, we get

$$\frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n)\}^2 + \frac{\lambda}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w}$$

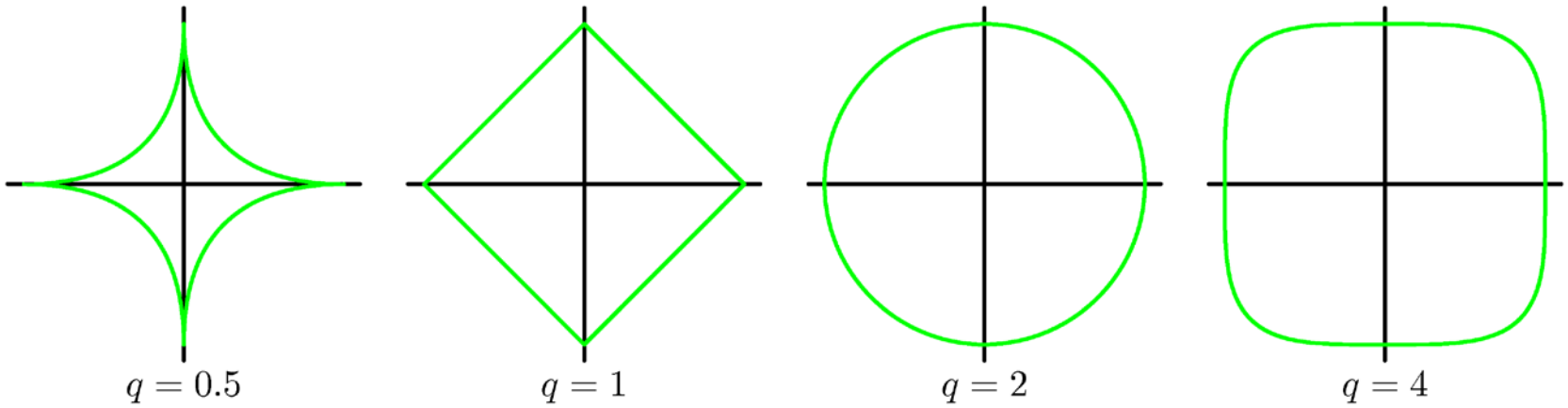
- which is minimized by

$$\mathbf{w} = \left(\lambda\mathbf{I} + \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}.$$

Regularized Least Squares (2)

- With a more general regularizer, we have

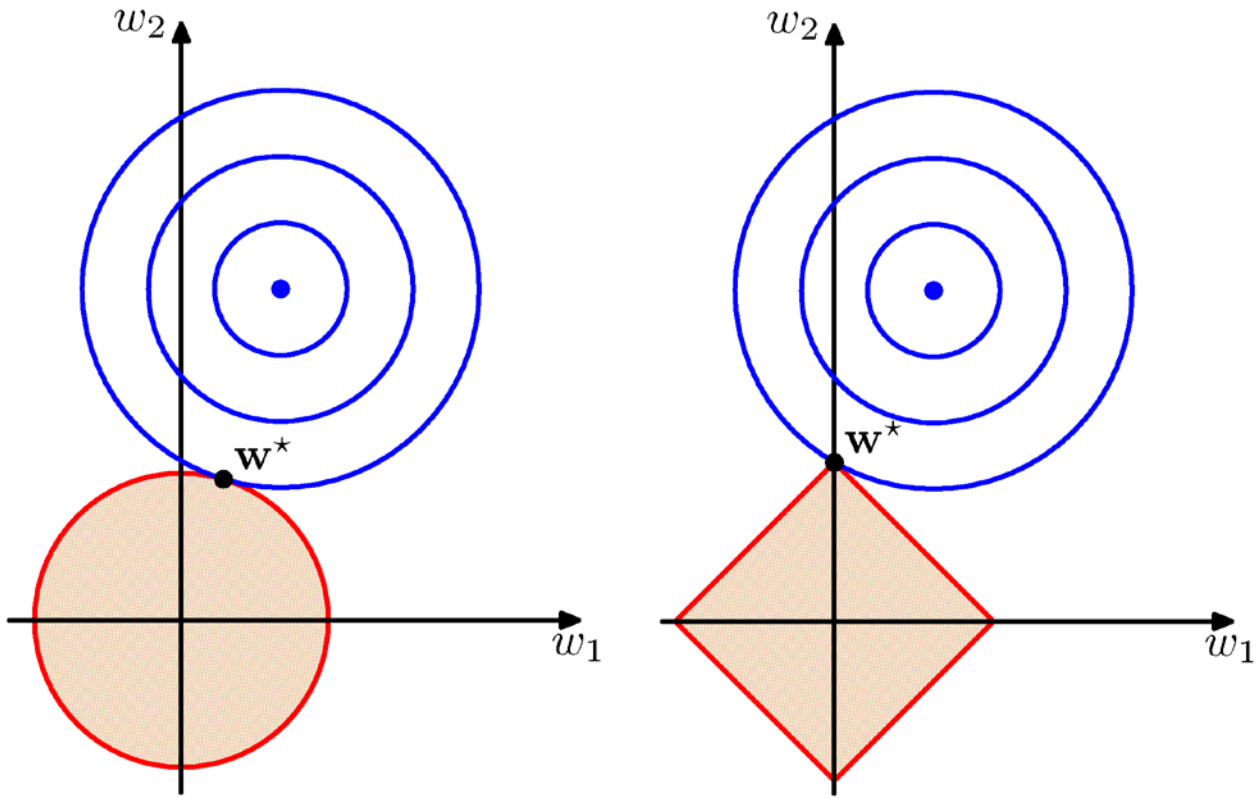
$$\frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n)\}^2 + \frac{\lambda}{2}\sum_{j=1}^{M}|w_j|^q$$



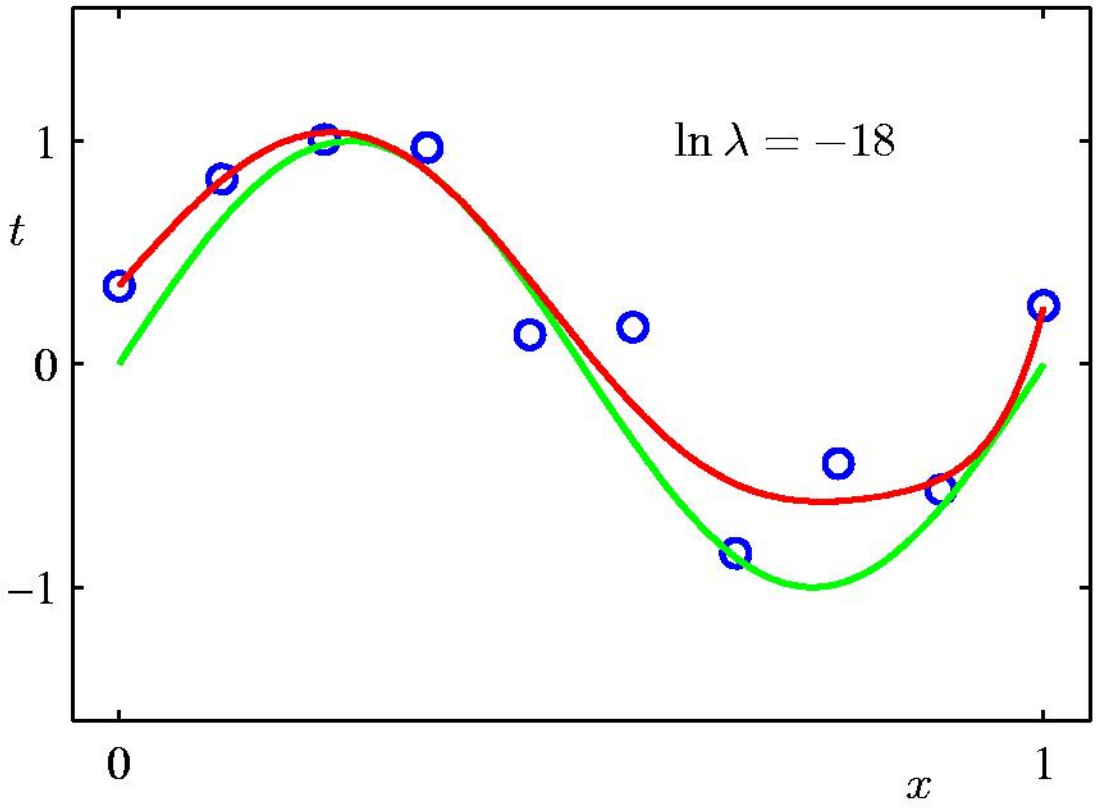
Lasso Quadratic

Regularized Least Squares (3)

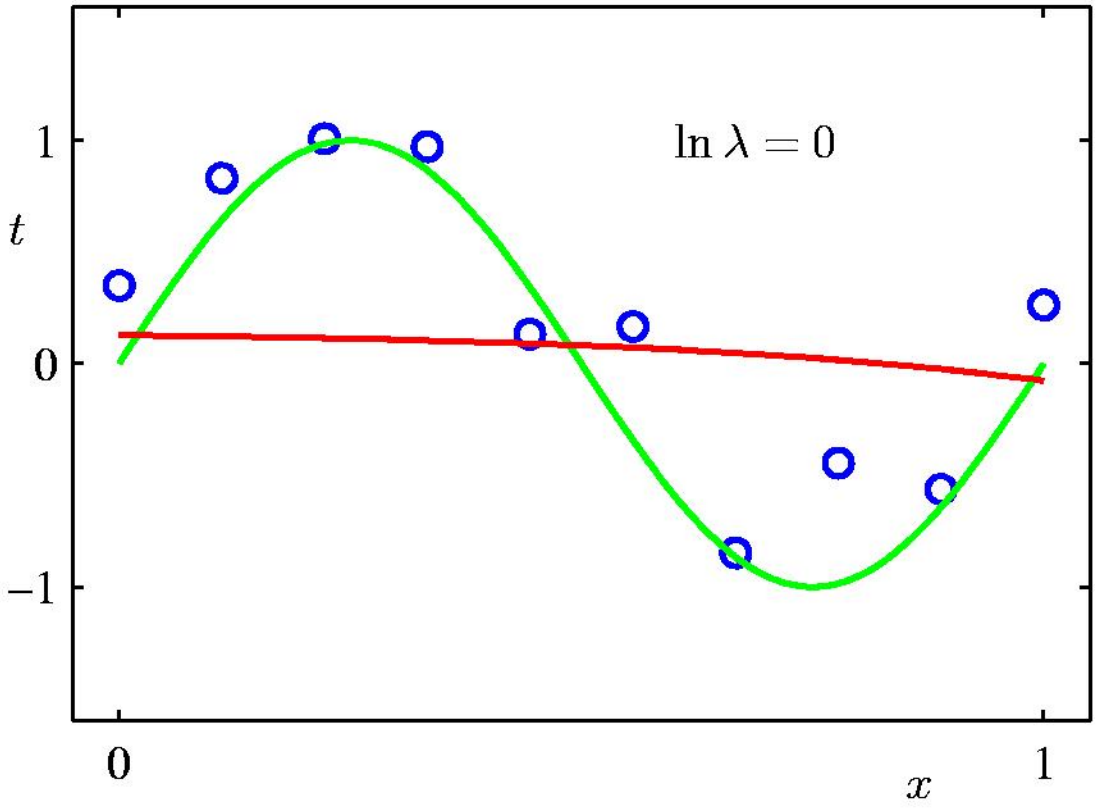
- Lasso tends to generate sparser solutions than a quadratic regularizer.



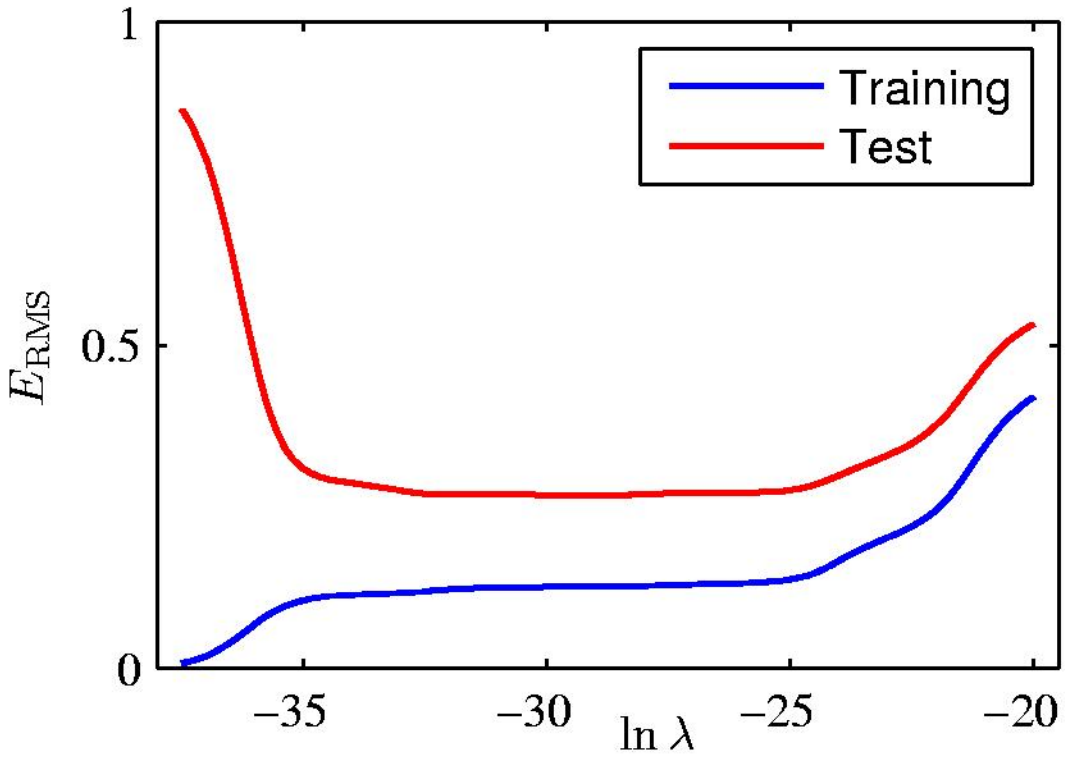
Regularization: $\ln \lambda = -18$



Regularization: $\ln \lambda = 0$



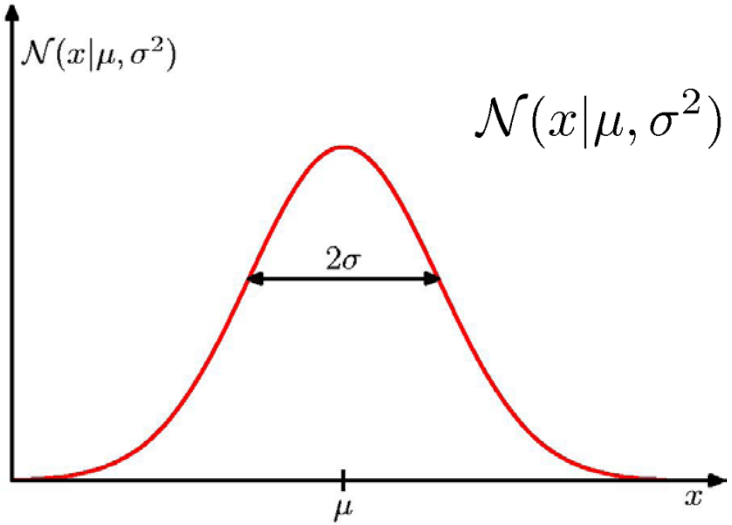
Regularization: E_{RMS} vs. $\ln \lambda$



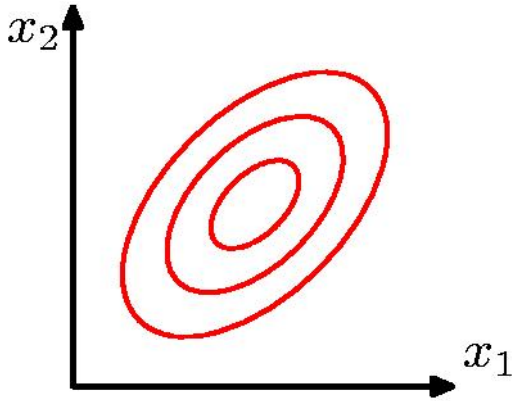
Polynomial Coefficients

	$\ln \lambda = -\infty$	$\ln \lambda = -18$	$\ln \lambda = 0$
$w_0^\star$	0.35	0.35	0.13
$w_1^\star$	232.37	4.74	-0.05
$w_2^\star$	-5321.83	-0.77	-0.06
$w_3^\star$	48568.31	-31.97	-0.05
$w_4^\star$	-231639.30	-3.89	-0.03
$w_5^\star$	640042.26	55.28	-0.02
$w_6^\star$	-1061800.52	41.32	-0.01
$w_7^\star$	1042400.18	-45.95	-0.00
$w_8^\star$	-557682.99	-91.53	0.00
$w_9^\star$	125201.43	72.68	0.01

The Gaussian Distribution

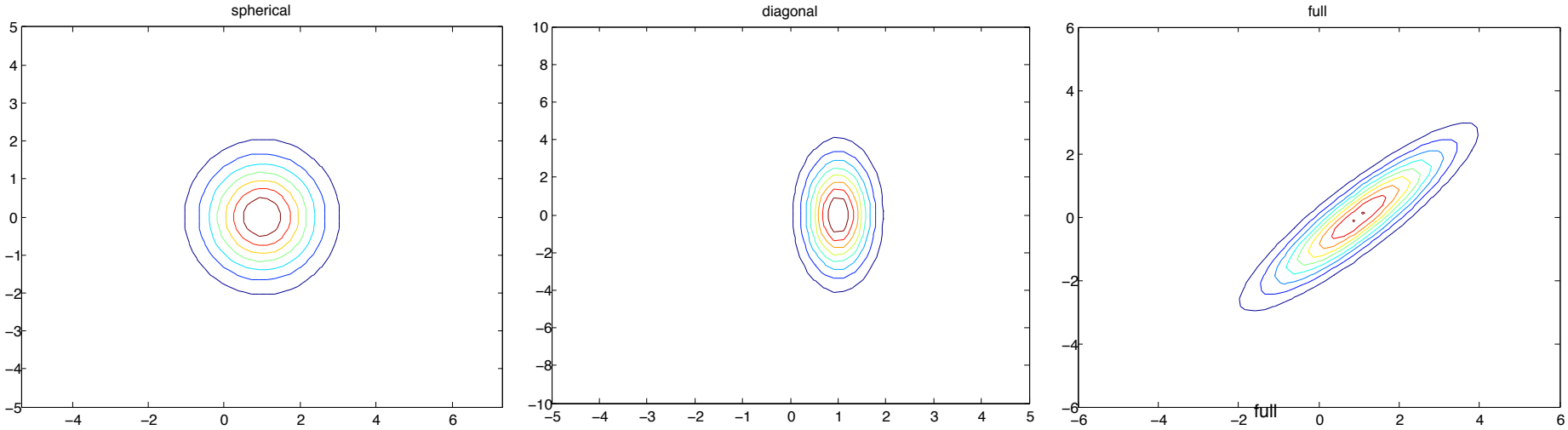


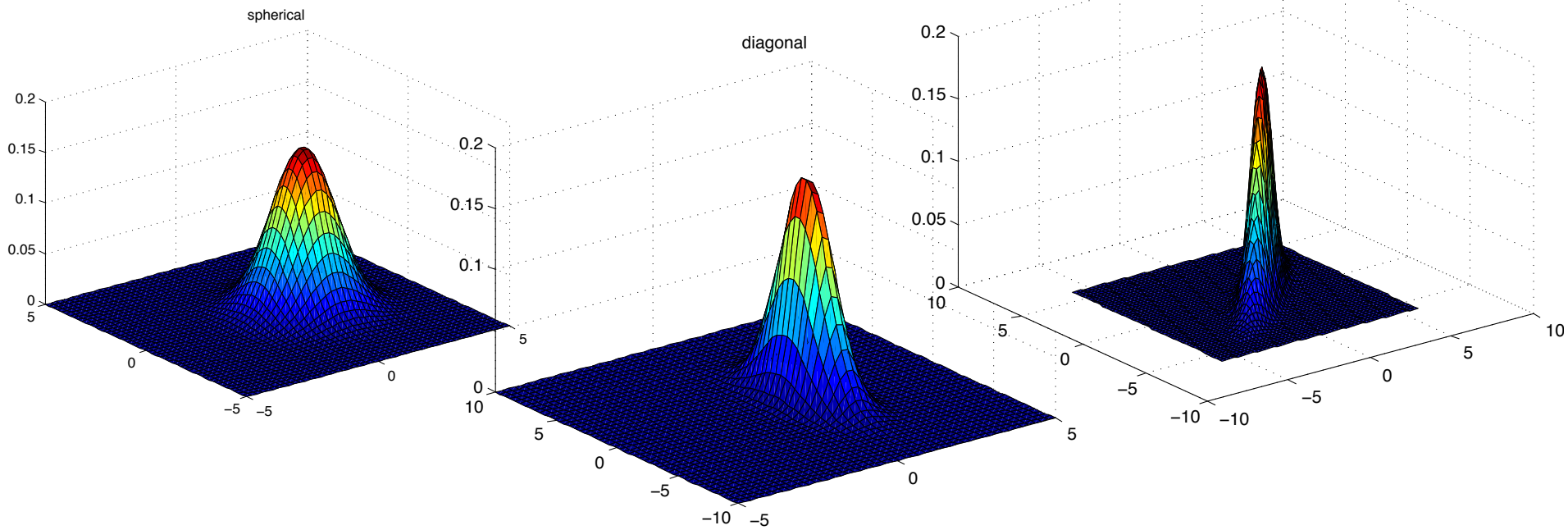
$$\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\}$$



$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}$$

Two-Dimensional Gaussians





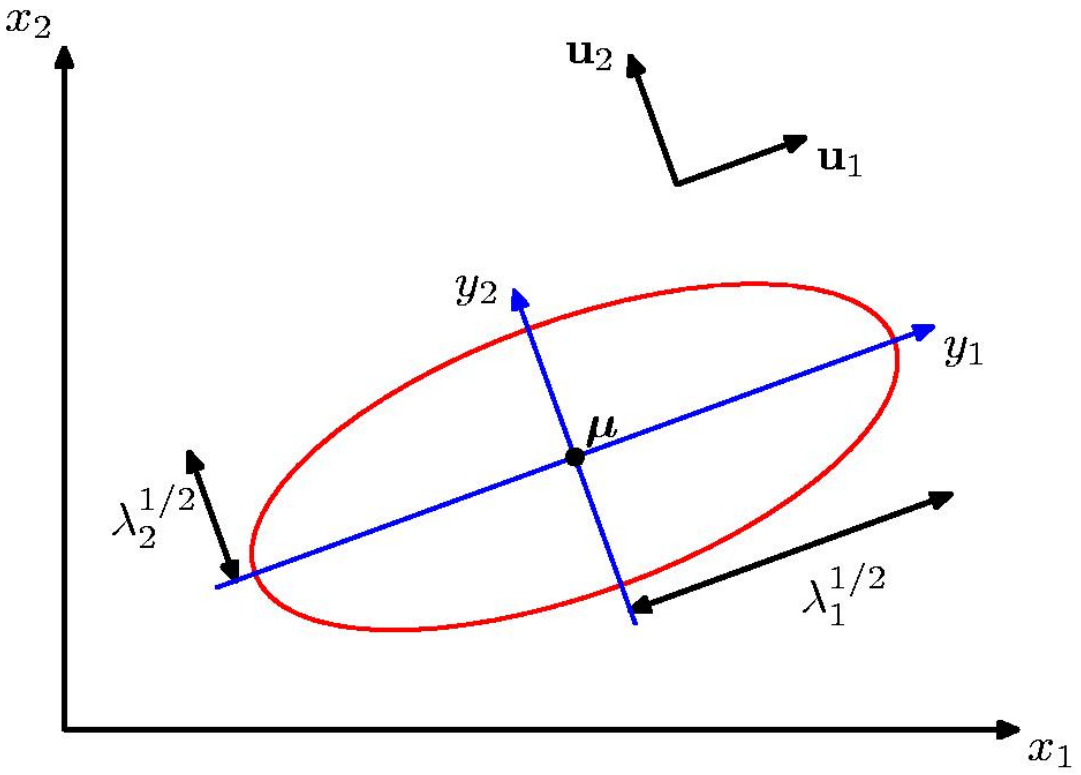
Geometry of the Multivariate Gaussian

$$\Delta^2 = (\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})$$

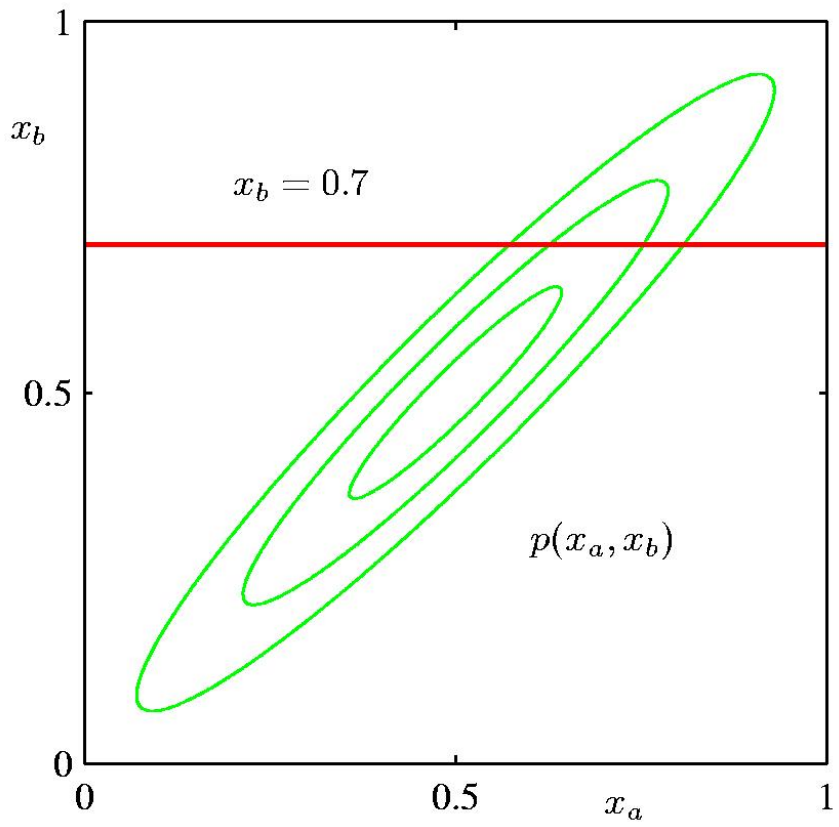
$$\boldsymbol{\Sigma}^{-1} = \sum_{i=1}^{D} \frac{1}{\lambda_i}\mathbf{u}_i\mathbf{u}_i^{\mathrm{T}}$$

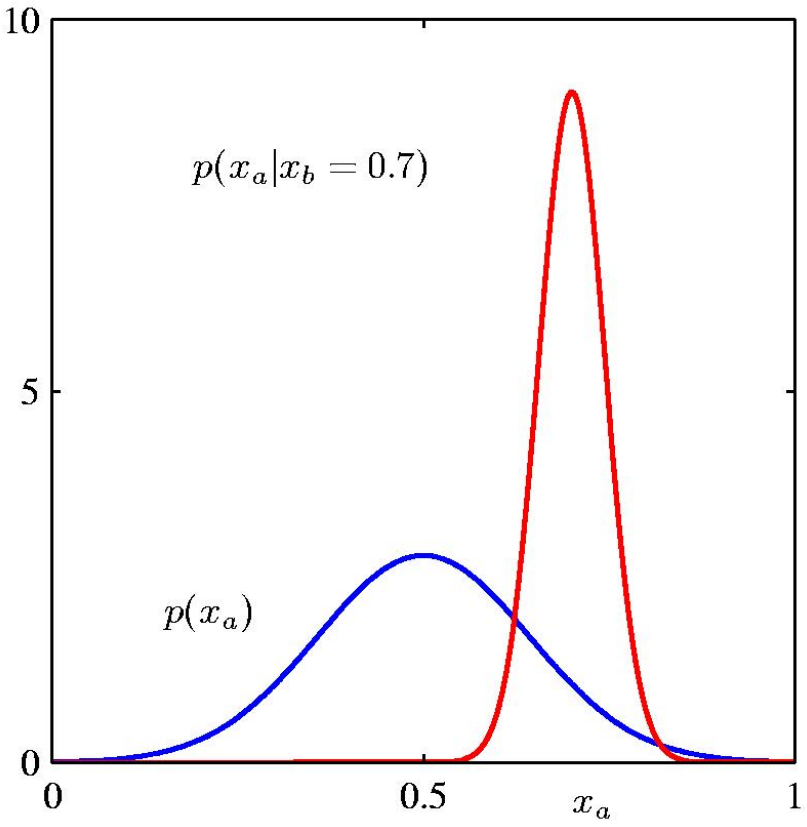
$$\Delta^2 = \sum_{i=1}^{D} \frac{y_i^2}{\lambda_i}$$

$$y_i = \mathbf{u}_i^{\mathrm{T}}(\mathbf{x}-\boldsymbol{\mu})$$



Conditional & Marginal Distributions





Maximum Likelihood & Least Squares (1)

- Assume observations from a deterministic function with added Gaussian noise:

$$t = y(\mathbf{x}, \mathbf{w}) + \epsilon \quad \text{where} \quad p(\epsilon|\beta) = \mathcal{N}(\epsilon|0, \beta^{-1})$$

- which is the same as saying,

$$p(t|\mathbf{x}, \mathbf{w}, \beta) = \mathcal{N}(t|y(\mathbf{x}, \mathbf{w}), \beta^{-1}).$$

- Given observed inputs, $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, and targets, $\mathbf{t} = [t_1, \ldots, t_N]^{\mathrm{T}}$, we obtain the likelihood function

$$p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n), \beta^{-1}).$$

Maximum Likelihood & Least Squares (2)

- Taking the logarithm, we get

$$\begin{aligned}\ln p(\mathbf{t}|\mathbf{w},\beta) &= \sum_{n=1}^{N} \ln \mathcal{N}(t_n|\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n),\beta^{-1}) \\ &= \frac{N}{2}\ln\beta - \frac{N}{2}\ln(2\pi) - \beta E_D(\mathbf{w})\end{aligned}$$

- where

$$E_D(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n)\}^2$$

- is the sum-of-squares error.

Maximum Likelihood & Least Squares (3)

- Computing the gradient and setting it to zero yields

$$\nabla_{\mathbf{w}} \ln p(\mathbf{t}|\mathbf{w}, \beta) = \beta \sum_{n=1}^{N} \left\{ t_n - \mathbf{w}^{\mathrm{T}} \boldsymbol{\phi}(\mathbf{x}_n) \right\} \boldsymbol{\phi}(\mathbf{x}_n)^{\mathrm{T}} = \mathbf{0}.$$

- Solving for w, we get

$$\mathbf{w}_{\mathrm{ML}} = \left(\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{\Phi} \right)^{-1} \boldsymbol{\Phi}^{\mathrm{T}} \mathbf{t}$$

- where

$$\boldsymbol{\Phi} = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \cdots & \phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \cdots & \phi_{M-1}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \cdots & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}.$$

Bayesian Linear Regression (1)

- Define a conjugate prior over w

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}_0, \mathbf{S}_0).$$

- Combining this with the likelihood function and using results for marginal and conditional Gaussian distributions, gives the posterior

- where

$$p(\mathbf{w}|\mathbf{t}) = \mathcal{N}(\mathbf{w}|\mathbf{m}_N, \mathbf{S}_N)$$

$$\begin{aligned} \mathbf{m}_N &= \mathbf{S}_N \left(\mathbf{S}_0^{-1}\mathbf{m}_0 + \beta \mathbf{\Phi}^{\mathrm{T}} \mathbf{t} \right) \\ \mathbf{S}_N^{-1} &= \mathbf{S}_0^{-1} + \beta \mathbf{\Phi}^{\mathrm{T}} \mathbf{\Phi}. \end{aligned}$$

Bayesian Linear Regression (2)

- A common choice for the prior is

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha^{-1}\mathbf{I})$$

- for which

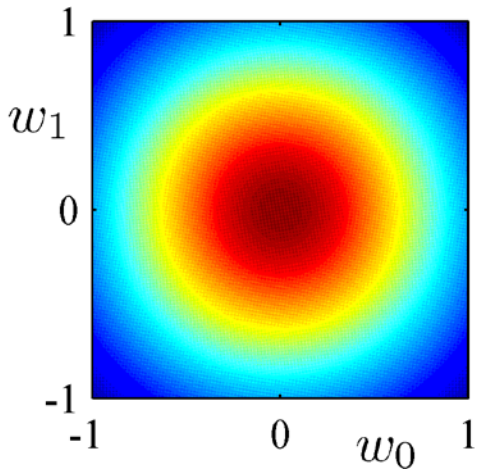
$$\begin{aligned} \mathbf{m}_N &= \beta \mathbf{S}_N \boldsymbol{\Phi}^{\mathrm{T}} \mathbf{t} \\ \mathbf{S}_N^{-1} &= \alpha \mathbf{I} + \beta \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{\Phi}. \end{aligned}$$

- Next we consider an example …

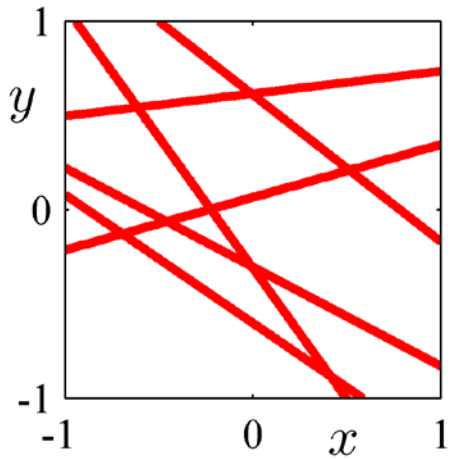
Bayesian Linear Regression (3)

0 data points observed

Prior



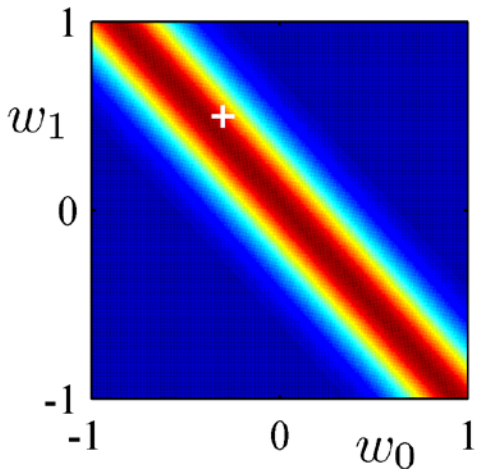
Data Space



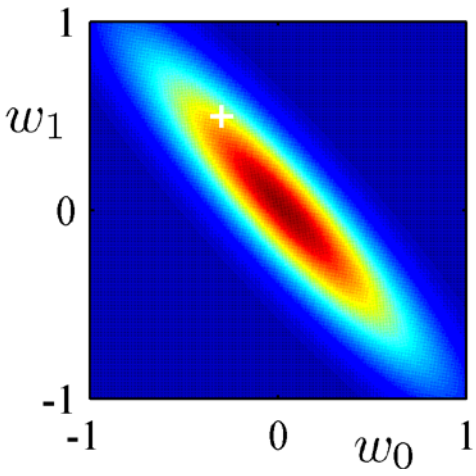
Bayesian Linear Regression (4)

1 data point observed

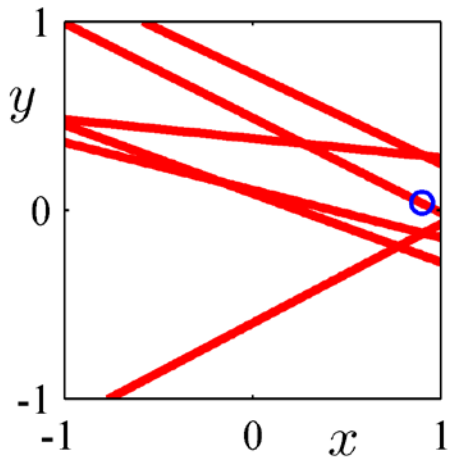
Likelihood



Posterior



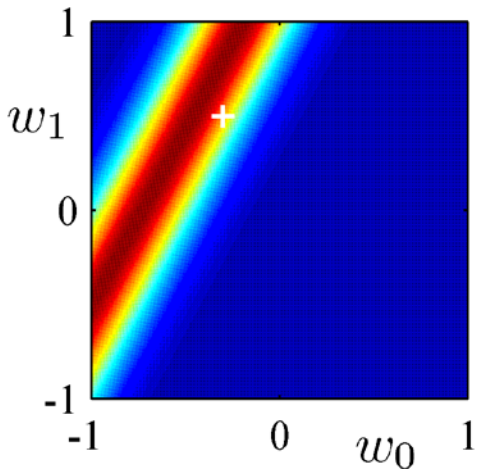
Data Space



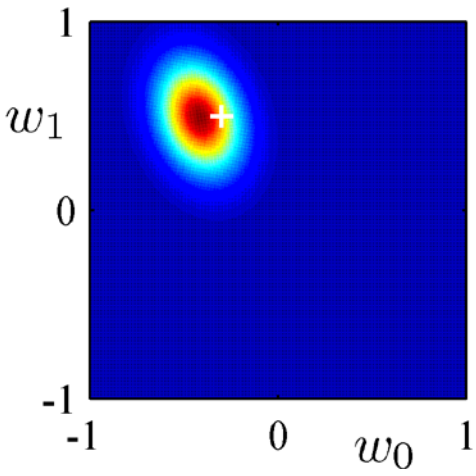
Bayesian Linear Regression (5)

2 data points observed

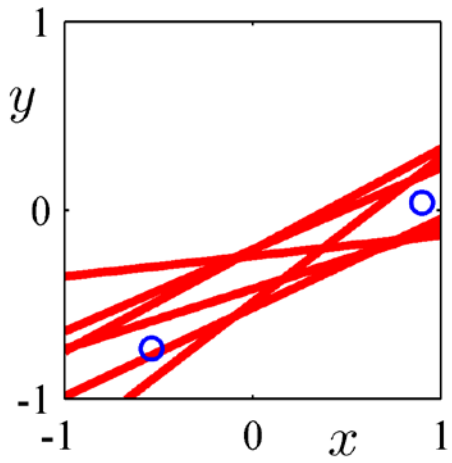
Likelihood



Posterior



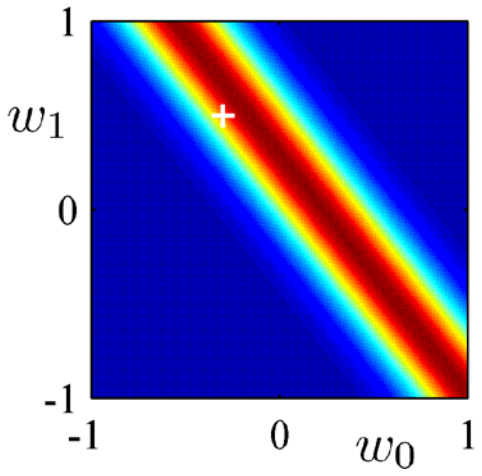
Data Space



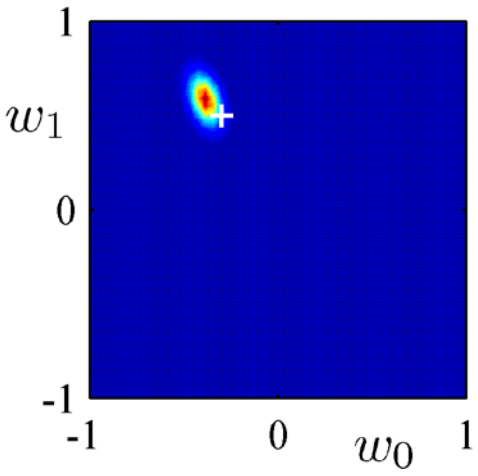
Bayesian Linear Regression (6)

20 data points observed

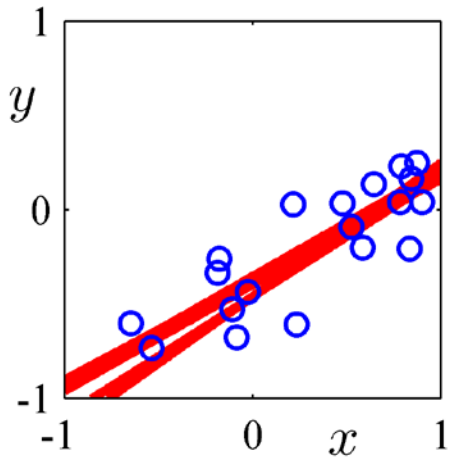
Likelihood



Posterior



Data Space



Predictive Distribution (1)

- Predict t for new values of x by integrating over w:

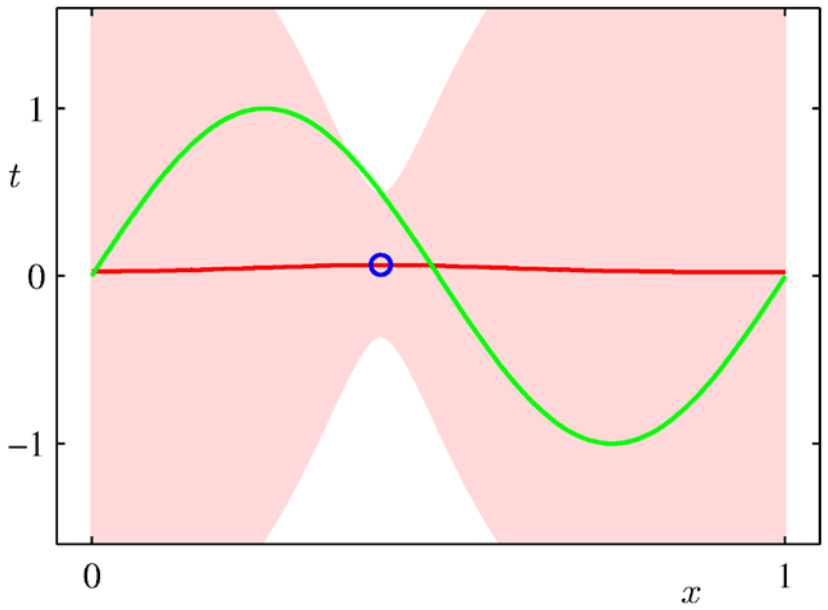
$$p(t|\mathbf{t},\alpha,\beta) = \int p(t|\mathbf{w},\beta)p(\mathbf{w}|\mathbf{t},\alpha,\beta)\,\mathrm{d}\mathbf{w}$$
$$= \mathcal{N}(t|\mathbf{m}_N^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}),\sigma_N^2(\mathbf{x}))$$

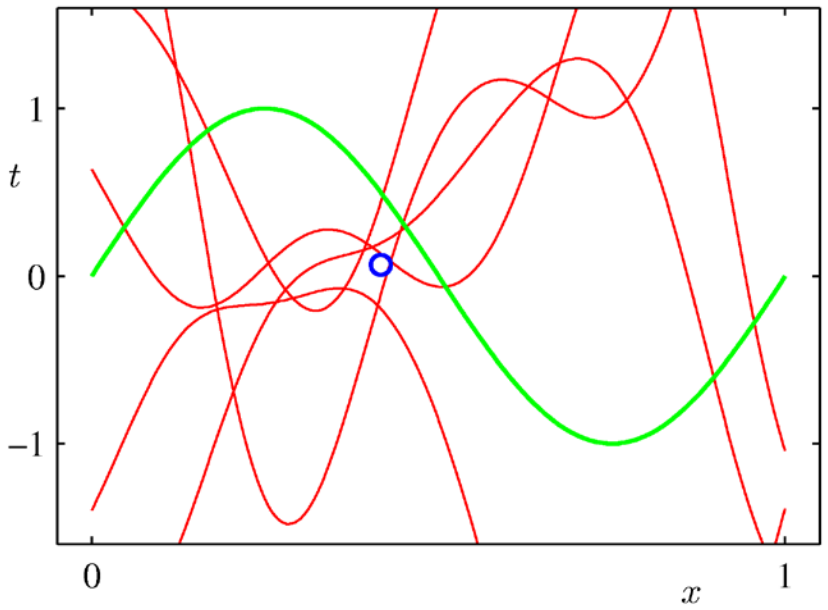
- where

$$\sigma_N^2(\mathbf{x}) = \frac{1}{\beta} + \boldsymbol{\phi}(\mathbf{x})^{\mathrm{T}}\mathbf{S}_N\boldsymbol{\phi}(\mathbf{x}).$$

Predictive Distribution (2)

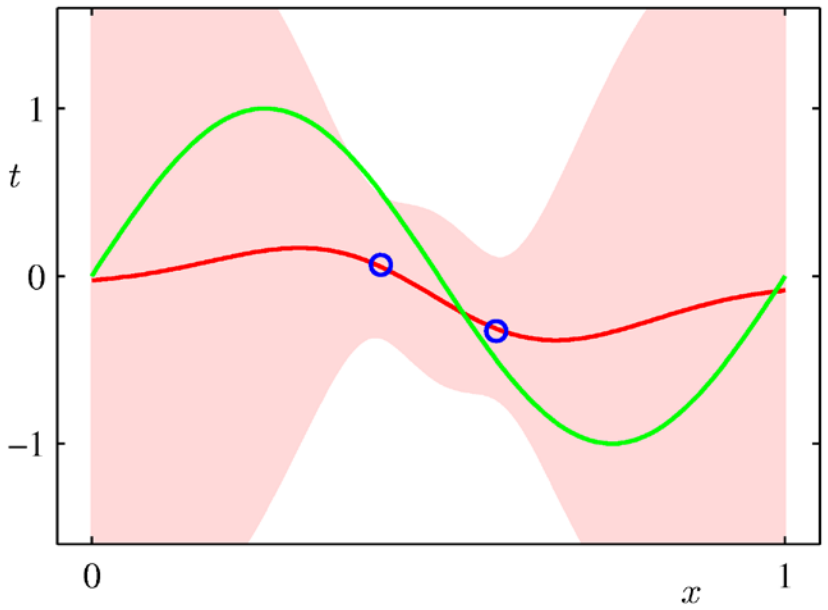
- Example: Sinusoidal data, 9 Gaussian basis functions, 1 data point

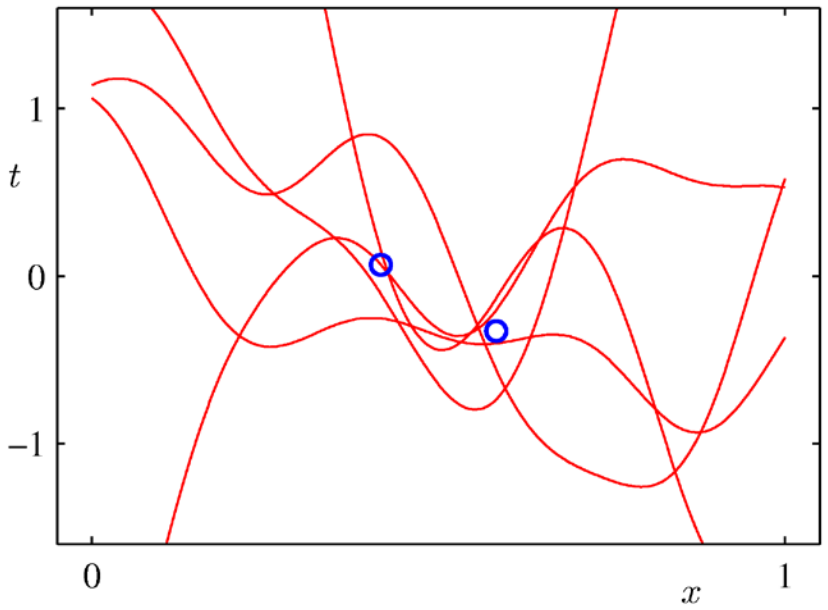




Predictive Distribution (3)

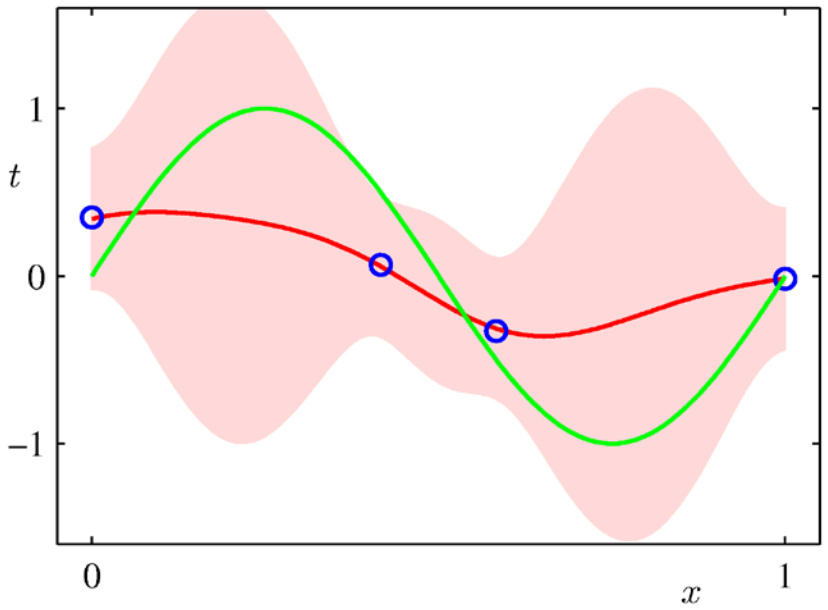
- Example: Sinusoidal data, 9 Gaussian basis functions, 2 data points

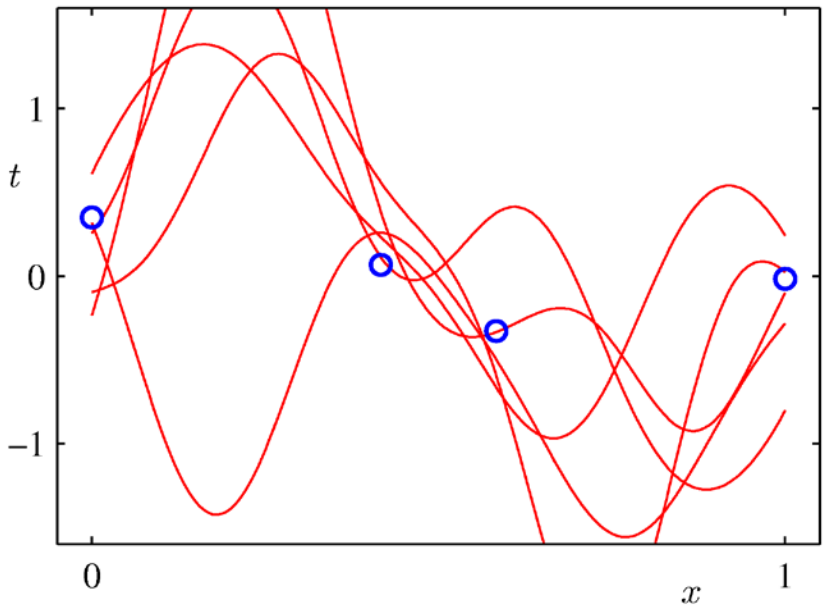




Predictive Distribution (4)

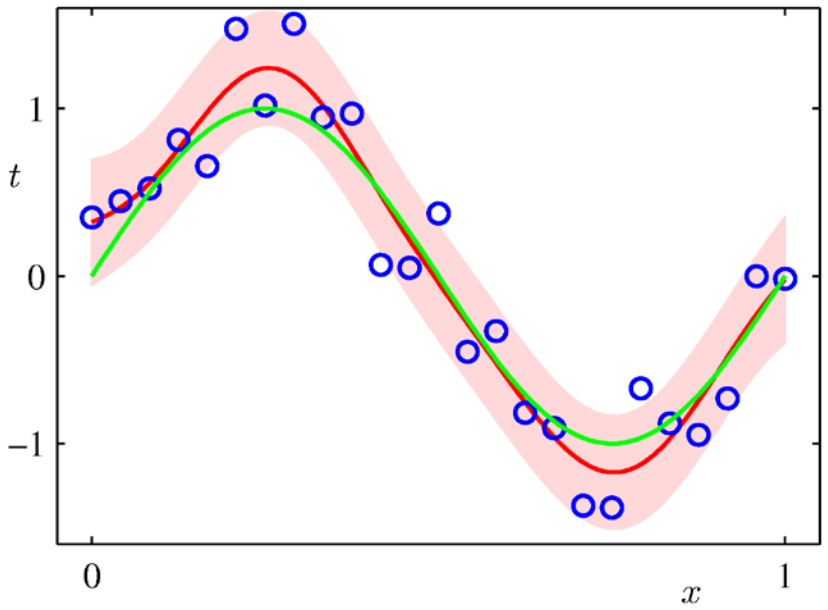
- Example: Sinusoidal data, 9 Gaussian basis functions, 4 data points

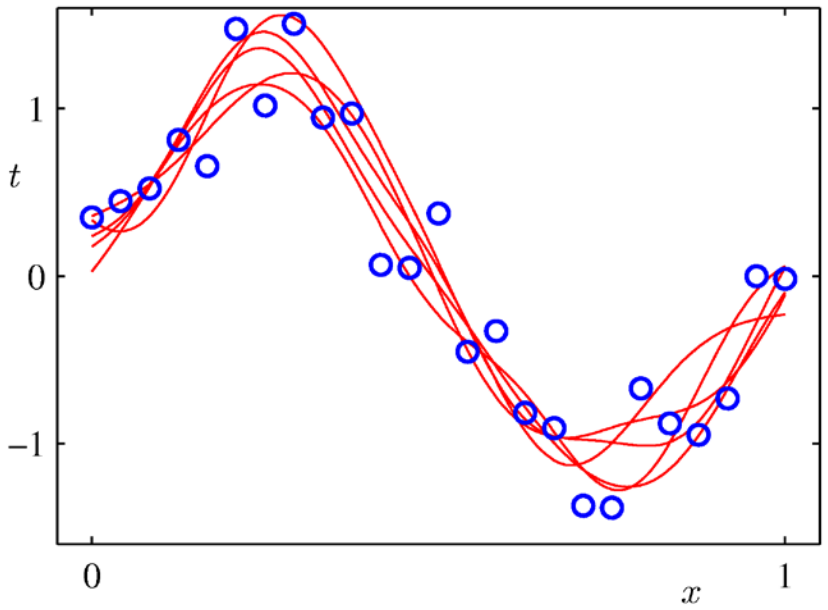




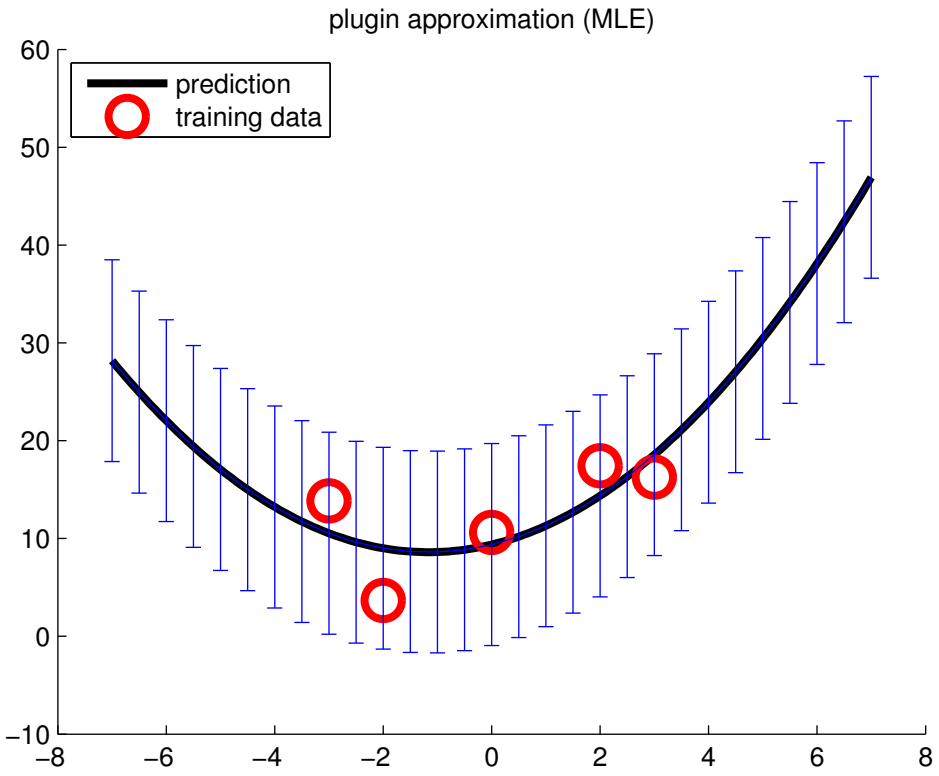
Predictive Distribution (5)

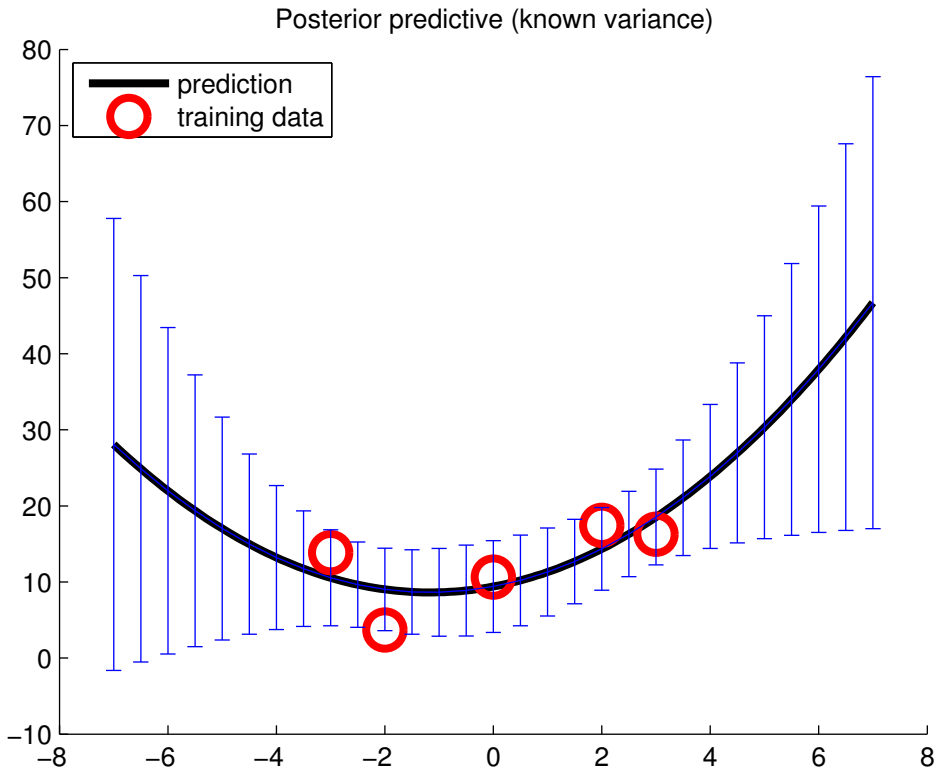
- Example: Sinusoidal data, 9 Gaussian basis functions, 25 data points



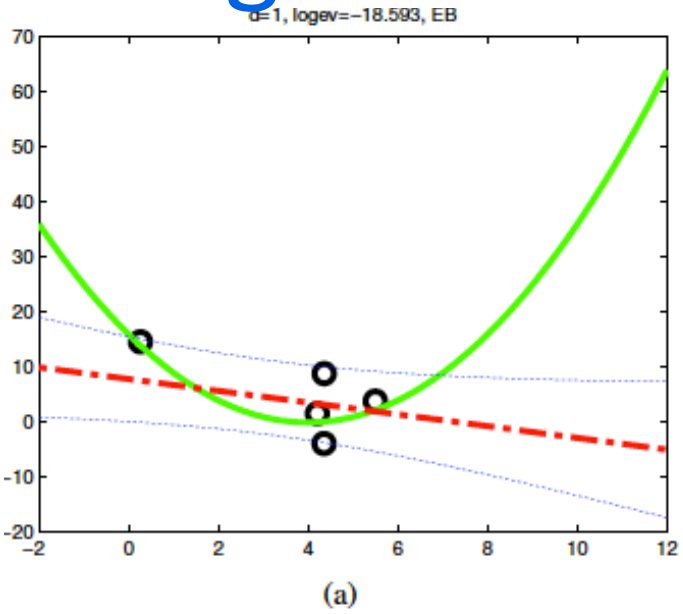


Estimation vs. Predictive Distributions

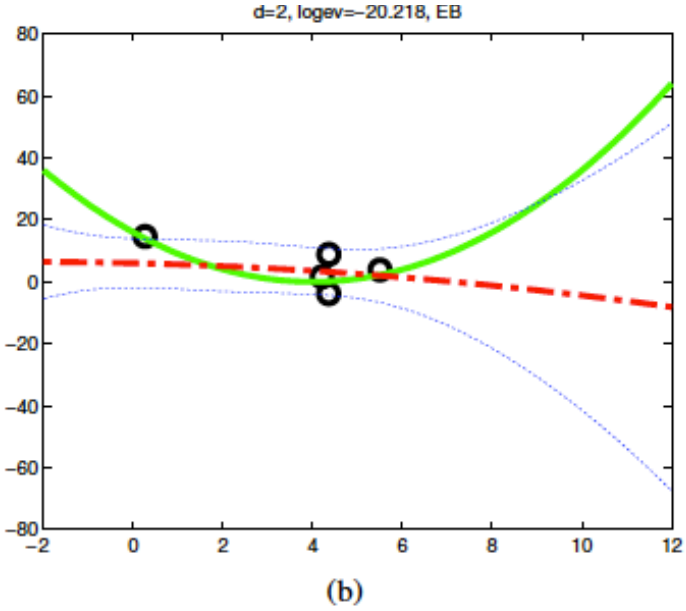




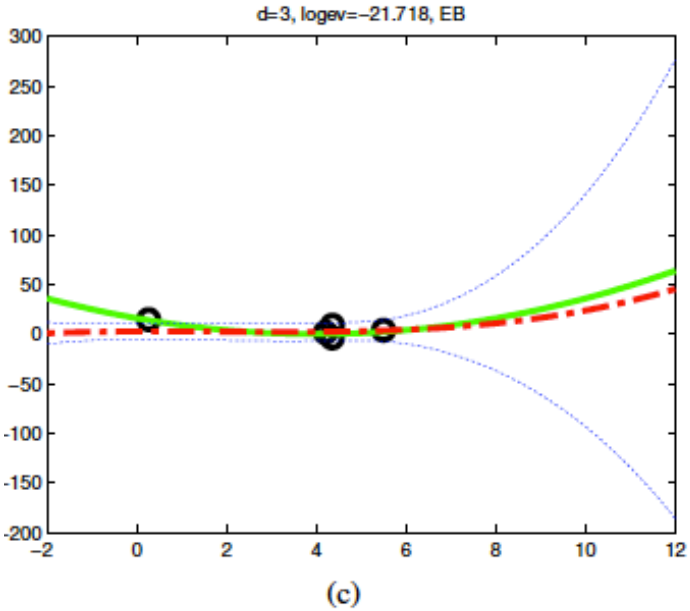
Marginal Data Likelihood: N=5



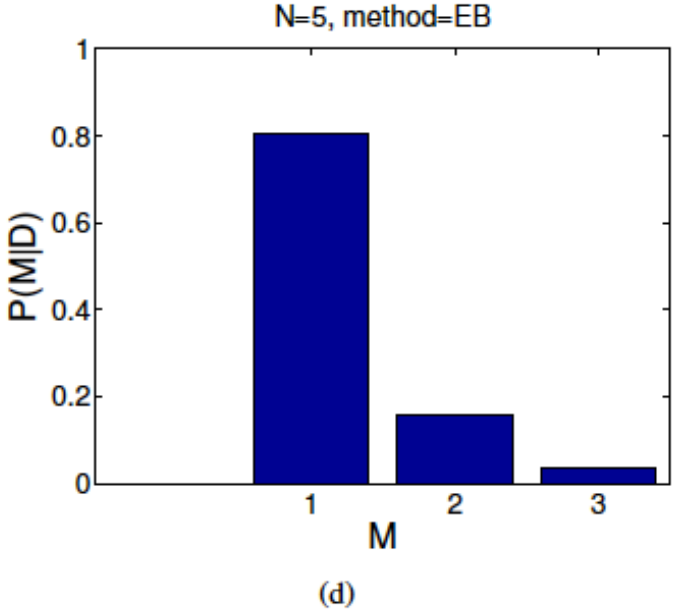
(a)



(b)

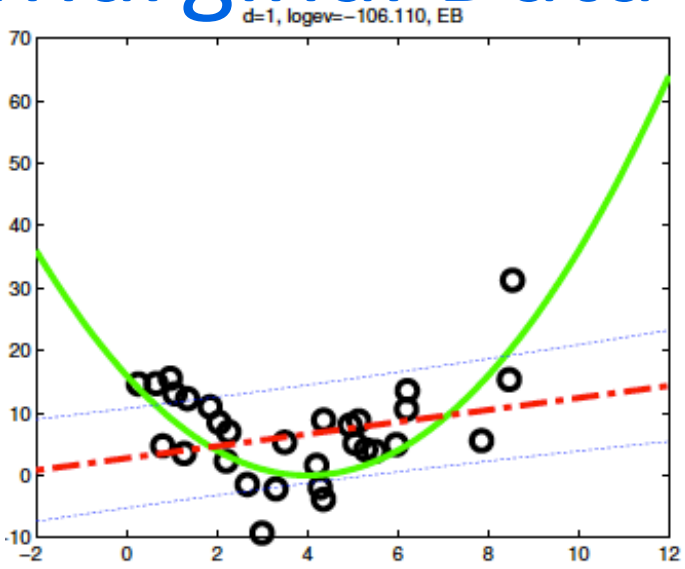


(c)

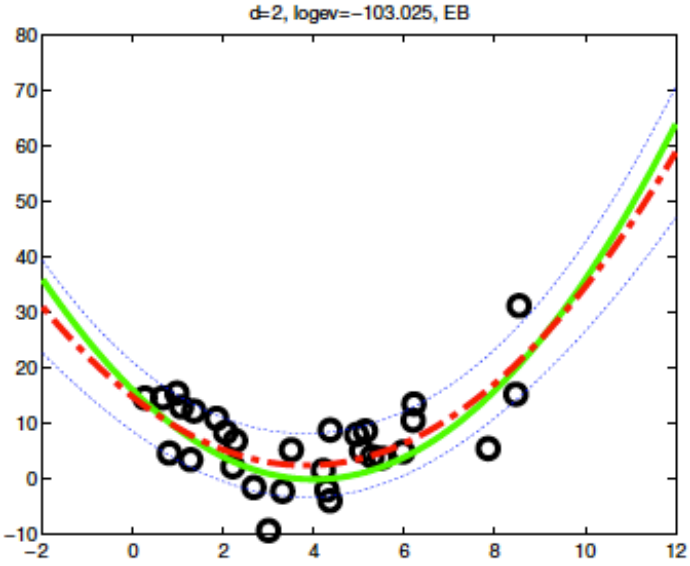


(d)

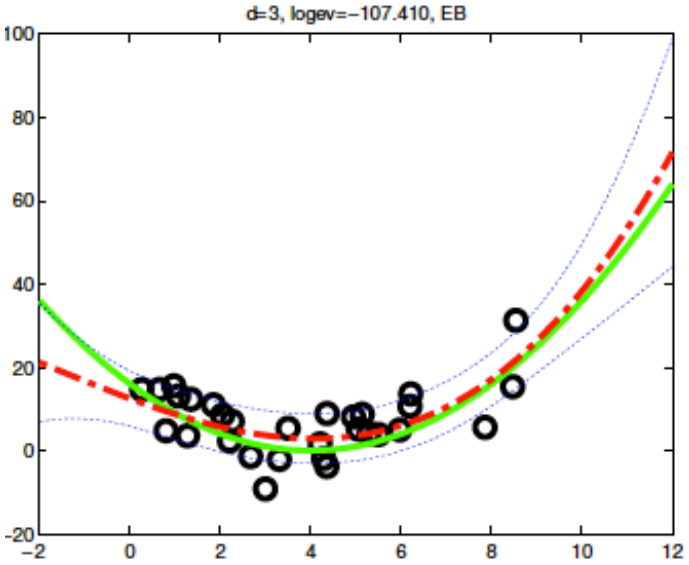
Marginal Data Likelihood: N=30



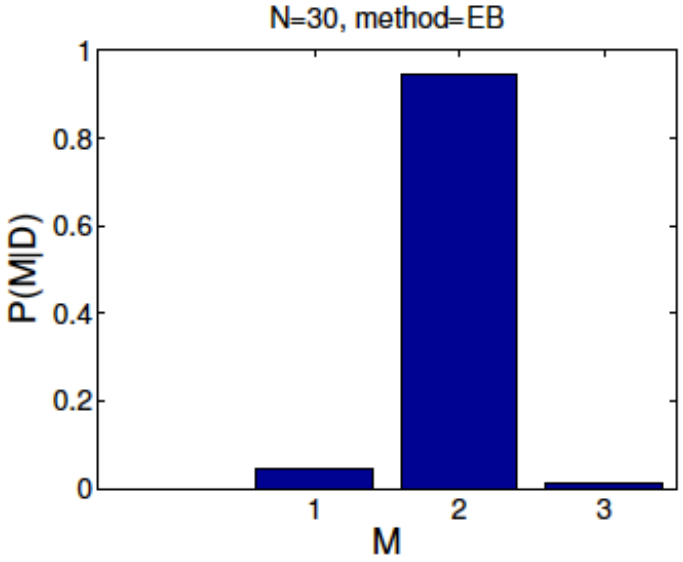
(a)



(b)

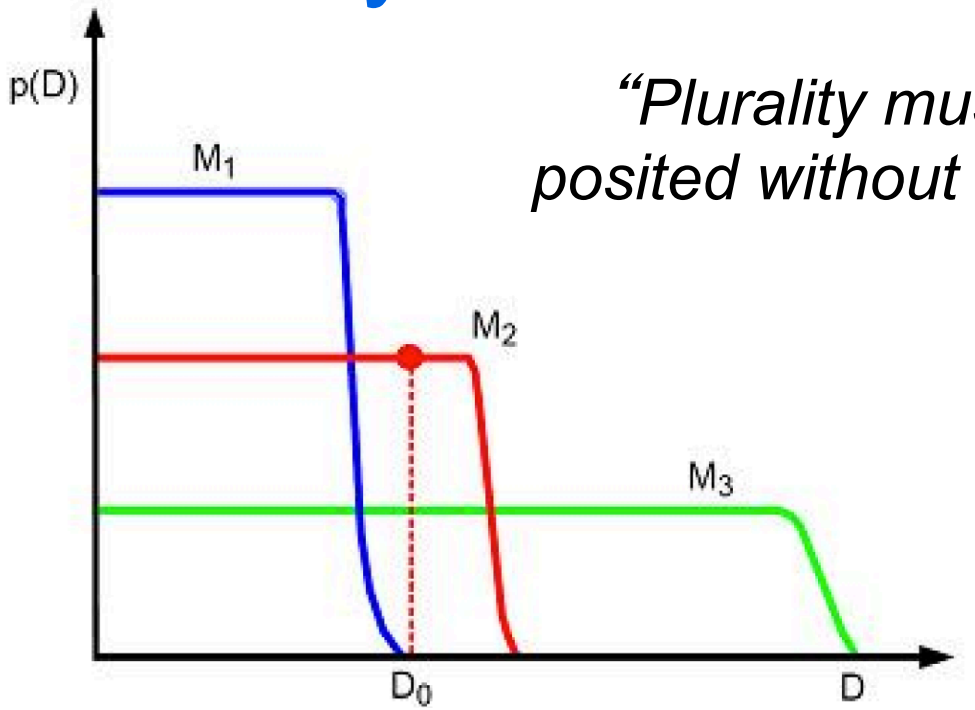


(c)



(d)

Bayesian Ockham's Razor



"Plurality must never be posited without necessity."

William of Ockham

- **Parametric Bayes:** Consider a finite list of possible models, average according to posterior probability (or in practice, just select the most probable)
- **Nonparametric Bayes:** Consider a single infinite model, integrate over parameters when making predictions or infer which finite subset is exhibited in your dataset

From Features to Kernels

- Nonparametric Gaussian regression:
  Would like to let the number of features $M \to \infty$

Prior:

$$p(w) = \mathcal{N}(w \mid 0, \alpha^{-1} I_M)$$

$$\mathbf{\Phi} = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \cdots & \phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \cdots & \phi_{M-1}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \cdots & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}.$$

Predictions:

$$y = \Phi w$$

$$\begin{aligned} p(y) &= \mathcal{N}(y \mid 0, \alpha^{-1} \Phi \Phi^T) \\ &= \mathcal{N}(y \mid 0, K) \end{aligned}$$

- Gaussian process models replace feature functions with direct specification of a *positive definite kernel function*

Mercer Kernel Functions

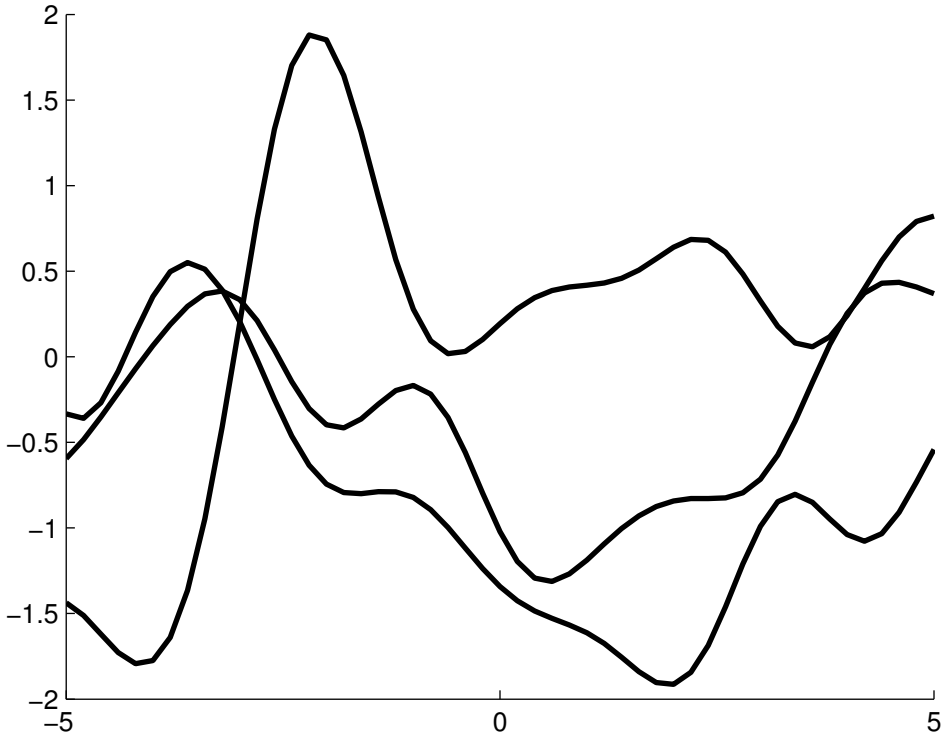
$$p(y) = \mathcal{N}(y \mid 0, \alpha^{-1}\Phi\Phi^T)$$
$$= \mathcal{N}(y \mid 0, K)$$
$$K_{ij} = k(x_i, x_j) = \phi(x_i)^T \phi(x_j)$$

Details on the board:

- Positive definite kernel functions
- Mercer's theorem
- What features lead to valid kernels?
- Examples of kernel functions

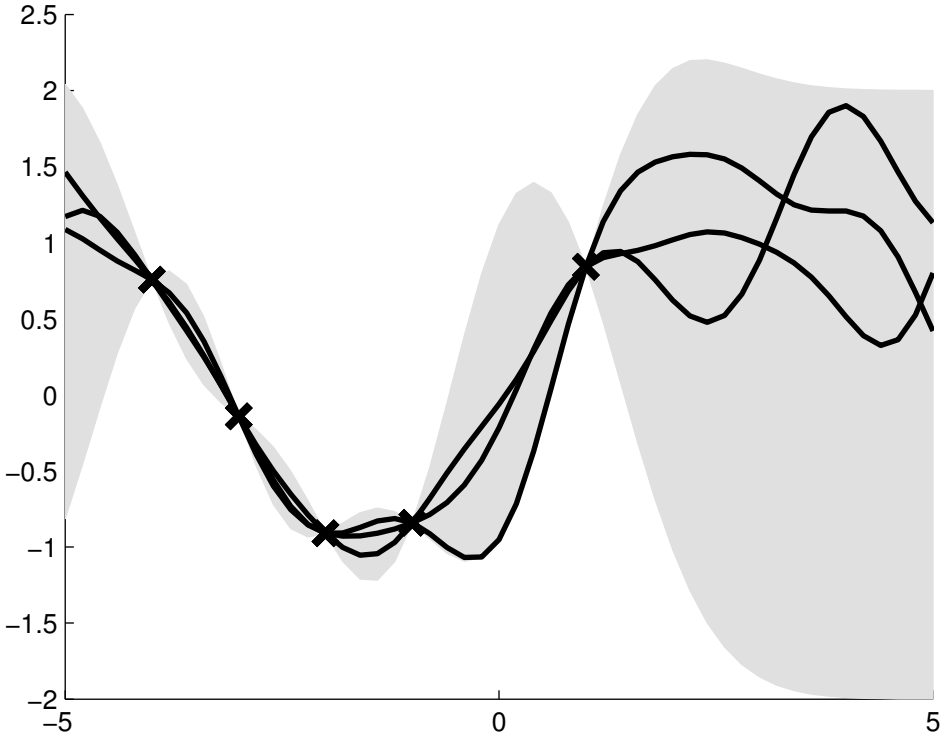
Feature representations and kernel representations are dual views of the same model families. The kernel representation is useful when M is large relative to N (number of features large relative to amount of data).

1D Gaussian Process Regression



Samples from Prior

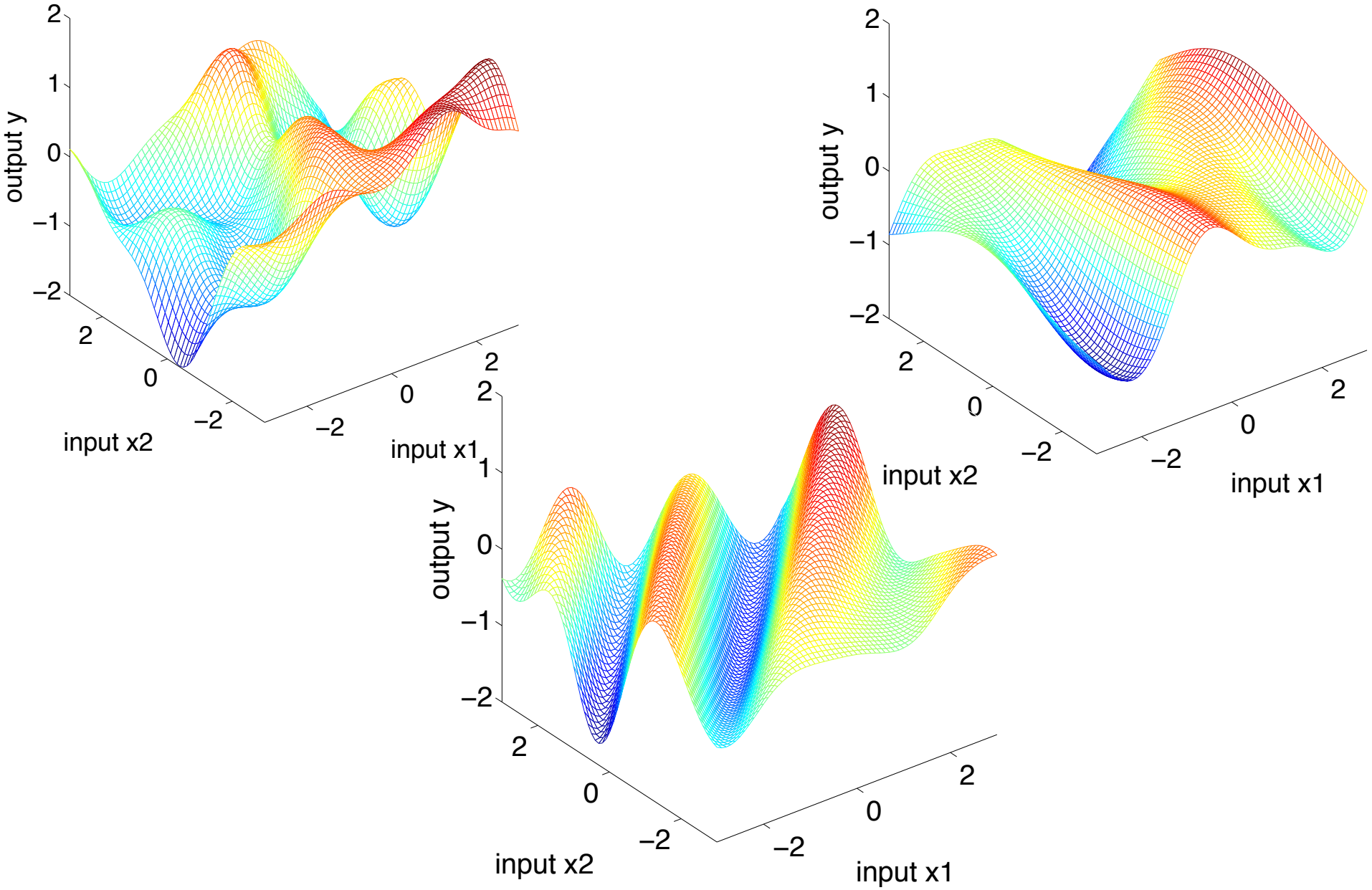
$$\kappa(x, x') = \sigma_f^2 \exp(-\frac{1}{2\ell^2}(x - x')^2)$$



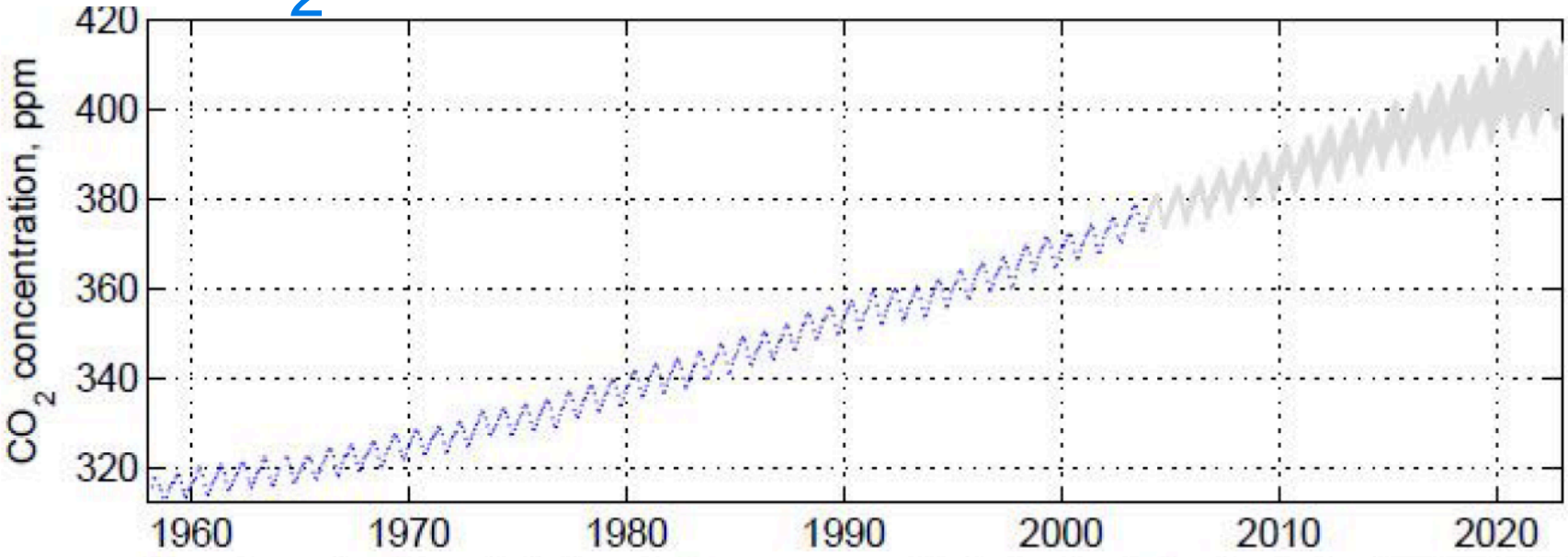
Posterior Given 5 Noise-Free Observations

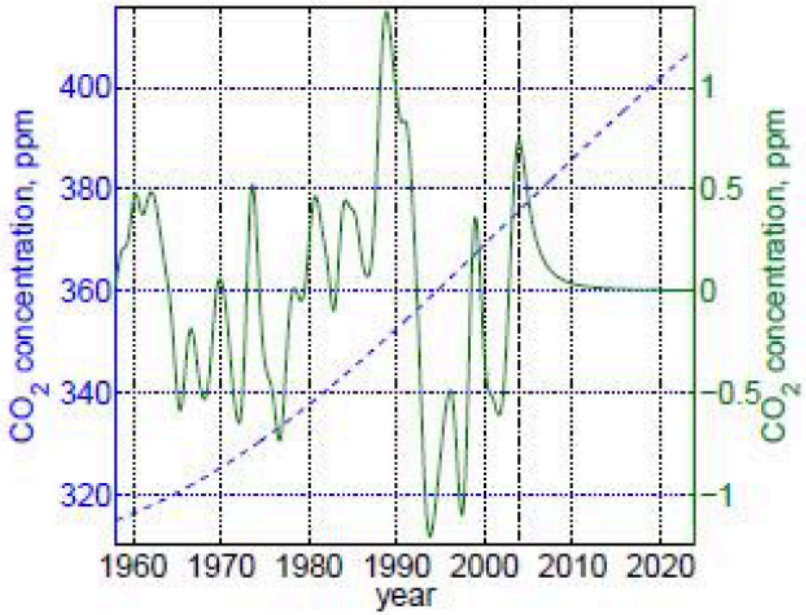
Squared exponential kernel or radial basis function (RBF) kernel has a countably *infinite* set of underlying feature functions

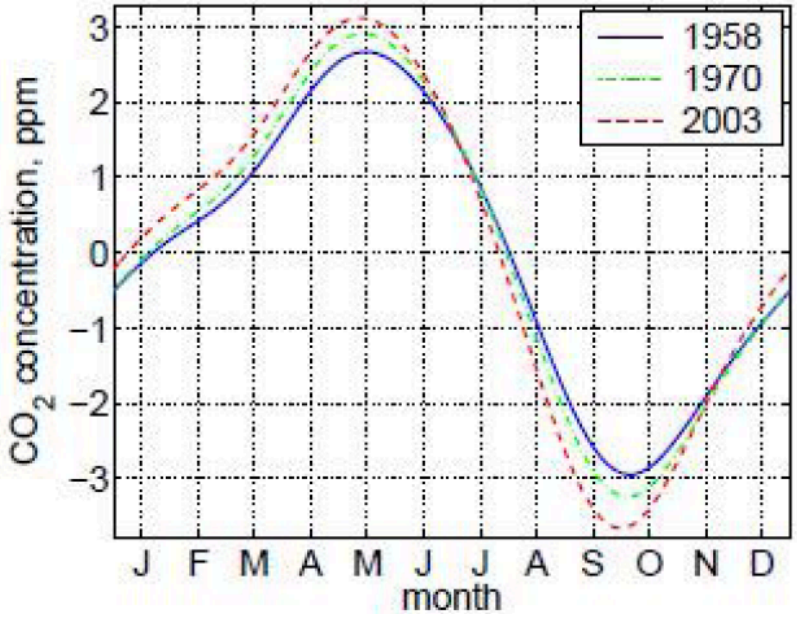
2D Gaussian Processes



CO_2 Concentration Over Time







Mauna Loa Observatory in Hawaii, analyzed by Rasmussen & Williams 2006

Mixing Kernels for CO_2 GP Regression

Smooth global trend

$$\kappa_1(x, x') = \theta_1^2 \exp\left(-\frac{(x - x')^2}{2\theta_2^2}\right)$$

Seasonal periodicity

$$\kappa_2(x, x') = \theta_3^2 \exp\left(-\frac{(x - x')^2}{2\theta_4^2} - \frac{2\sin^2(\pi(x - x'))}{\theta_5^2}\right)$$

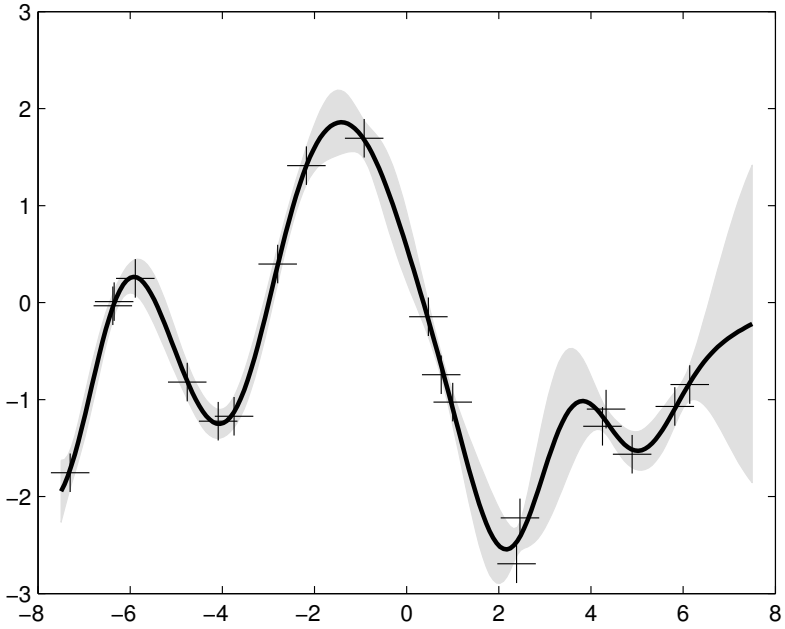
Medium term irregularities

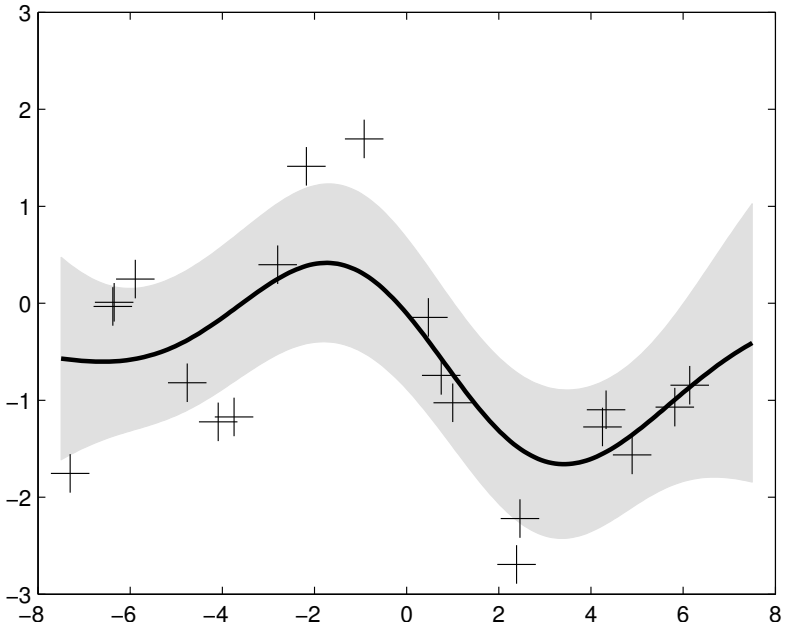
$$\kappa_3(x, x') = \theta_6^2 \left(1 + \frac{(x - x')^2}{2\theta_8\theta_7^2}\right)^{-\theta_8}$$

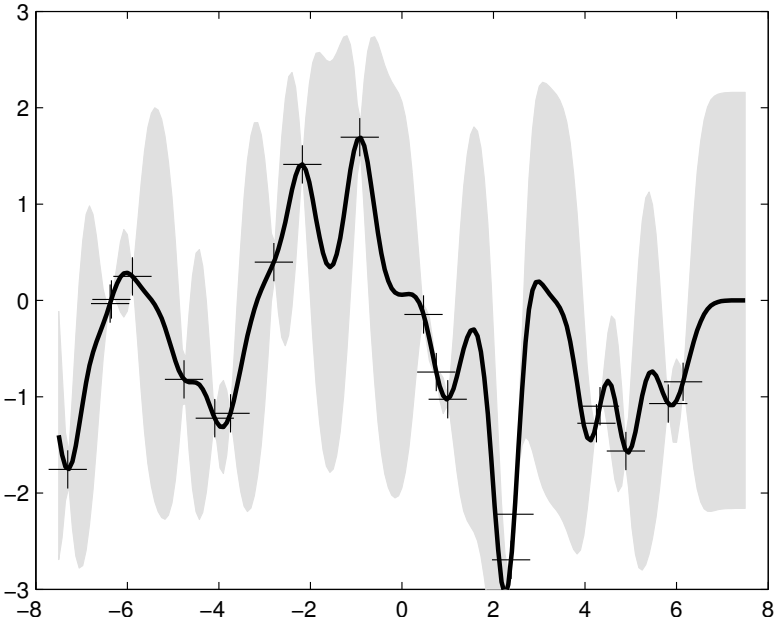
Correlated Observation Noise

$$\kappa_4(x_p, x_q) = \theta_9^2 \exp\left(-\frac{(x_p - x_q)^2}{2\theta_{10}^2}\right) + \theta_{11}^2\delta_{pq}$$

Gaussian Process Hyperparameters





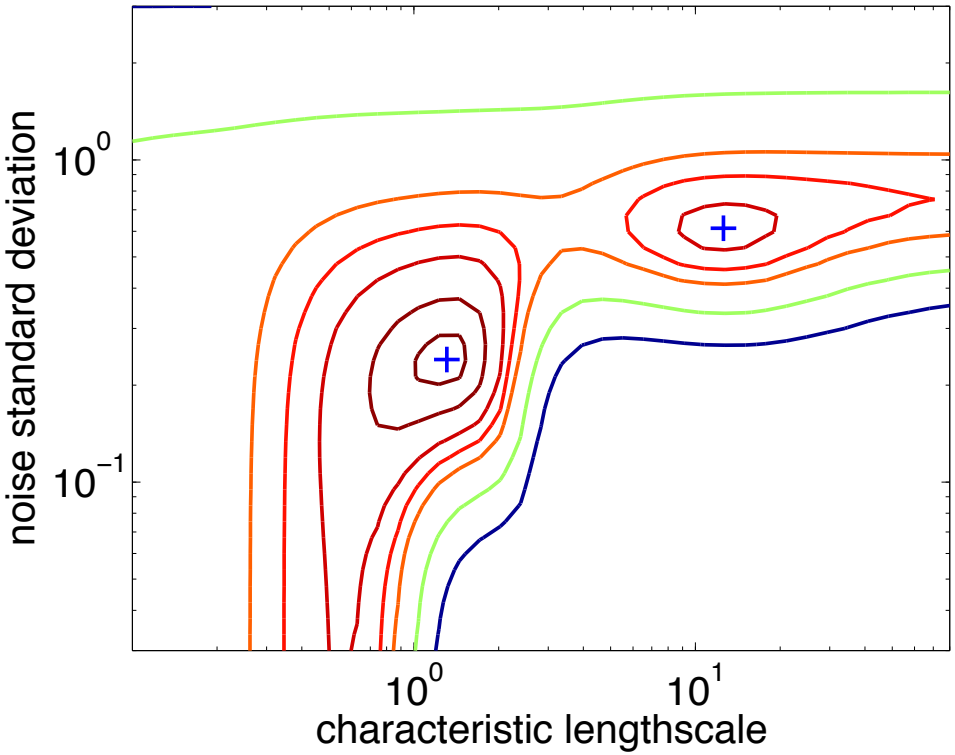


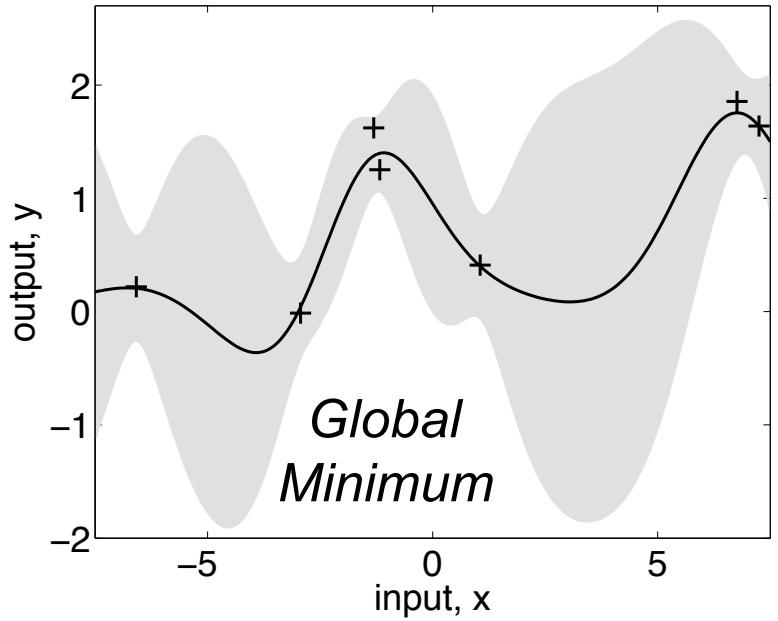
$$\kappa(x, x') = \sigma_f^2 \exp(-\frac{1}{2\ell^2}(x - x')^2)$$

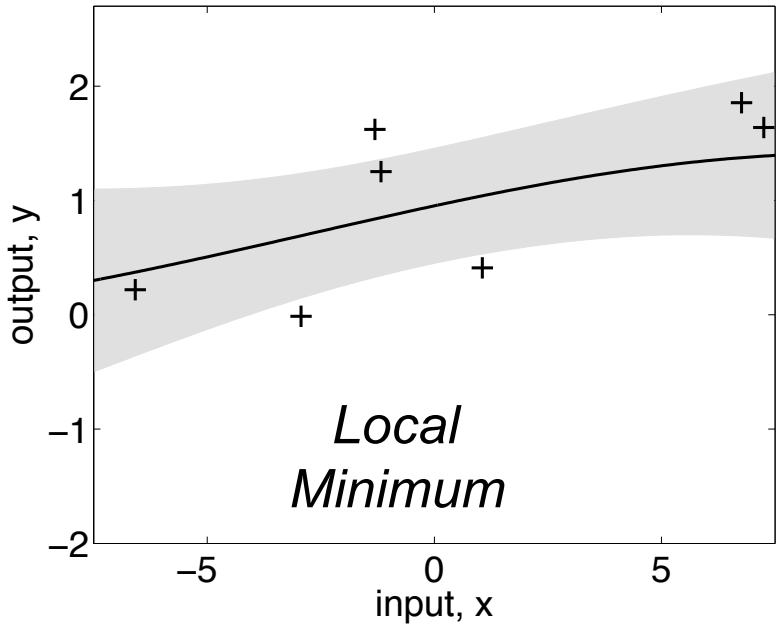
How should we fit to data?

- Cross-validation
- Full Bayesian analysis
- Maximize marginal likelihood (empirical Bayes, tractable for GP regression)

Hyperparameter Marginal Likelihoods





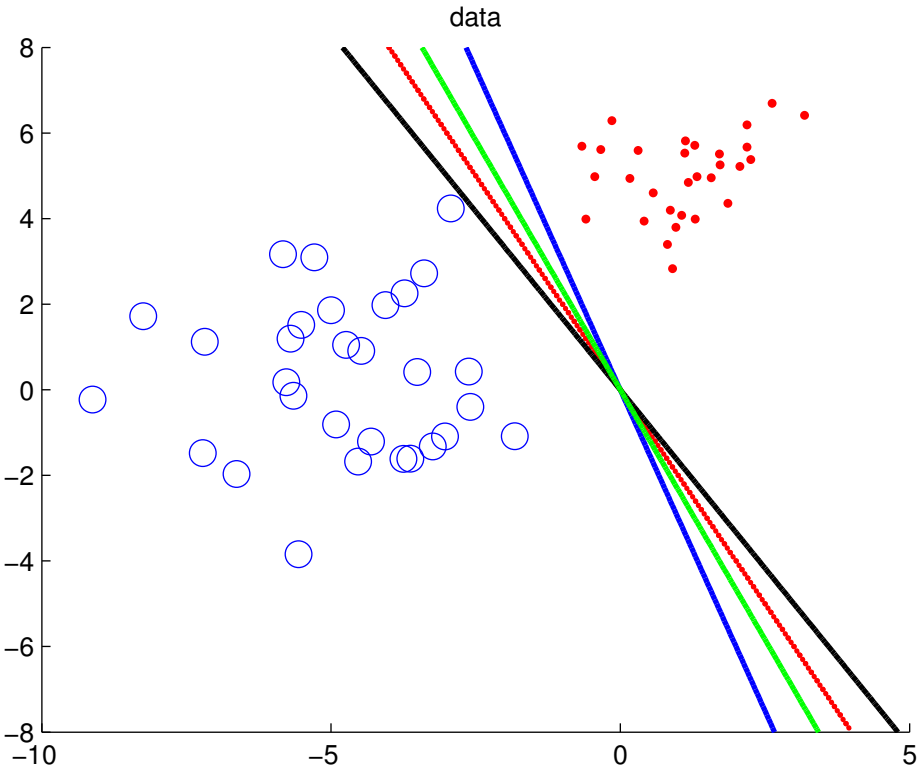


$$p(\mathbf{y}|\mathbf{X}) = \int p(\mathbf{y}|\mathbf{f},\mathbf{X})p(\mathbf{f}|\mathbf{X})d\mathbf{f}$$

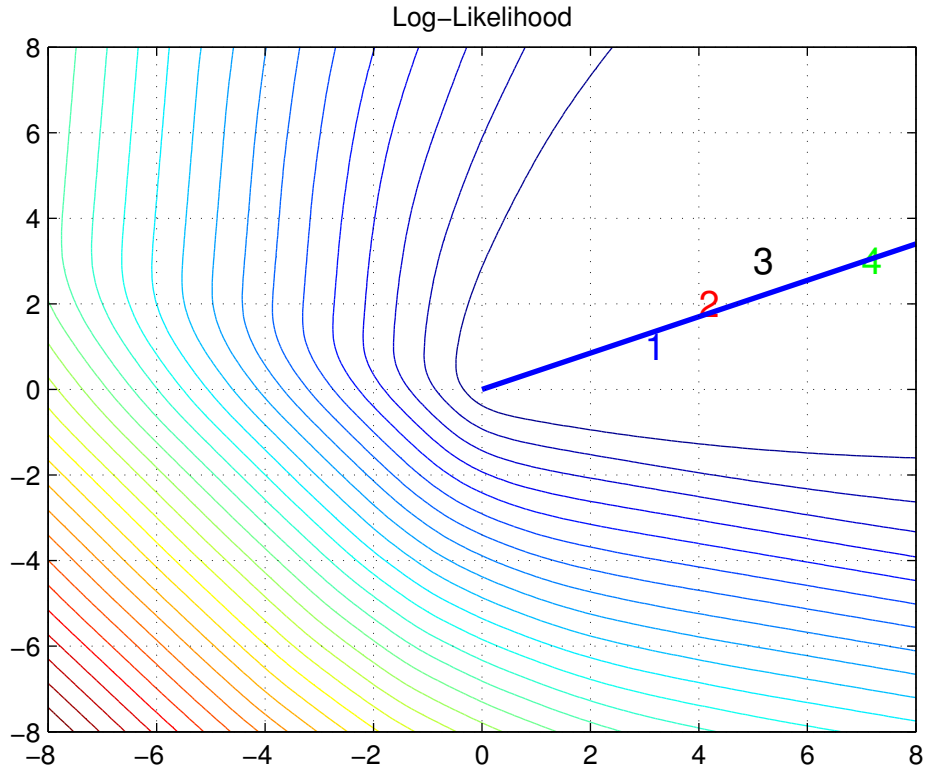
$$\log p(\mathbf{y}|\mathbf{X}) = -\frac{1}{2}\mathbf{y}\mathbf{K}_y^{-1}\mathbf{y} - \frac{1}{2}\log|\mathbf{K}_y| - \frac{N}{2}\ln(2\pi)$$

GPs for Binary Classification

Board: Parametric versus nonparametric ***generalized linear models***



Linearly Separable Data

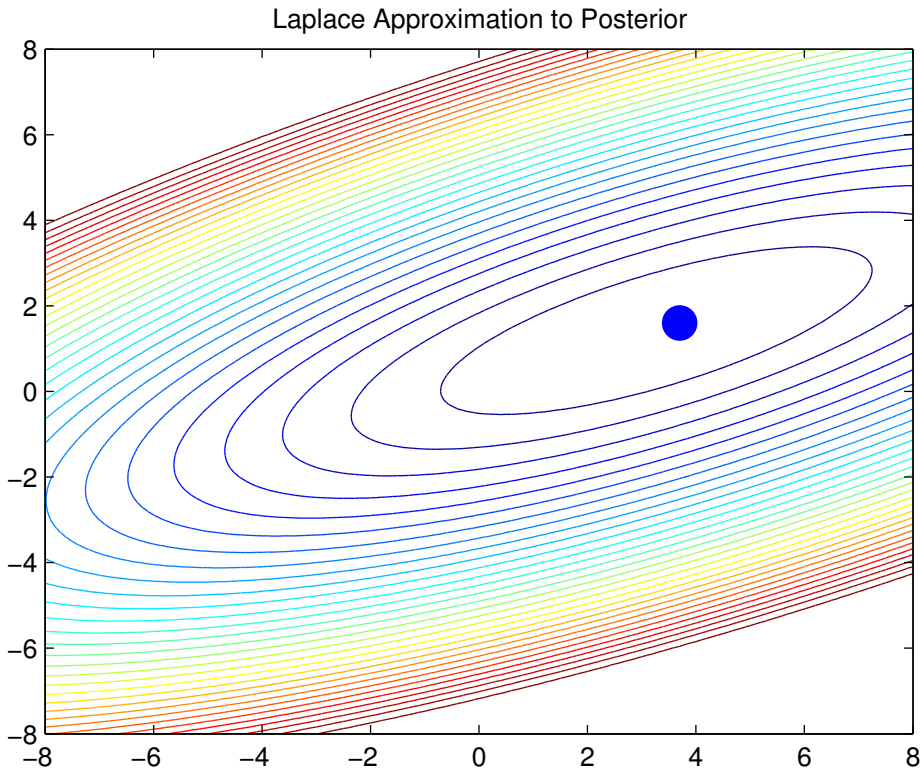


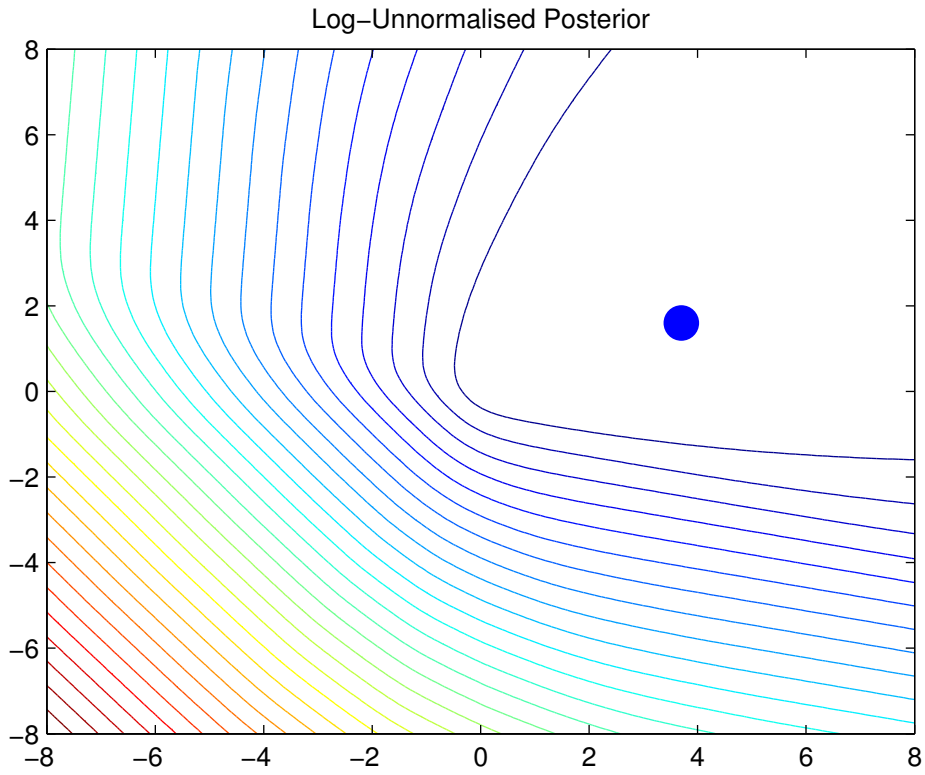
Log-likelihood Function

$$p(\mathbf{y}|\mathbf{X},\mathbf{w}) = \prod_{i:y_i=1} \frac{e^{\mathbf{w}^T\mathbf{x}_i}}{1+e^{\mathbf{w}^T\mathbf{x}_i}} \prod_{i:y_i=0} \frac{1}{1+e^{\mathbf{w}^T\mathbf{x}_i}} = \exp(\mathbf{w}^T \sum_i y_i\mathbf{x}_i) \prod_{i=1}^{N} (1+e^{\mathbf{w}^T\mathbf{x}_i})^{-1}$$

Linear Regression ⟺ ***Logistic Regression*** ***as*** ***GP Regression*** ⟺ ***GP Classification***

Laplace Approximation of LR Posterior





Laplace Approximation

$$p(\mathbf{w}|\mathcal{D}) \approx \mathcal{N}(\mathbf{w}|\hat{\mathbf{w}}, \mathbf{H}^{-1})$$

$$\mathbf{H} = -\nabla^2 E(\mathbf{w})|_{\hat{\mathbf{w}}}$$

Log Posterior Distribution

$$E(\mathbf{w}) = -(\log p(\mathcal{D}|\mathbf{w}) + \log p(\mathbf{w}))$$

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} E(\mathbf{w})$$

Losses for Binary Classification