# LINEAR REGRESSION

J. Elder

CSE 4404/5327 Introduction to Machine Learning and Pattern Recognition

# Credits



- Some of these slides were sourced and/or modified from:
  - Christopher Bishop, Microsoft UK

# Linear Regression Topics



- What is linear regression?
- Example:  polynomial curve fitting
- Other basis families
- Solving linear regression problems
- Regularized regression
- Multiple linear regression
- Bayesian linear regression

# What is Linear Regression?



- In classification, we seek to identify the **categorical** class $C_k$ associate with a given input vector **x**.
- In regression, we seek to identify (or **estimate**) a **continuous** variable *y* associated with a given input vector **x**.
- *y* is called the **dependent variable.**
- **x** is called the **independent variable.**
- If *y* is a vector, we call this multiple regression.
- We will focus on the case where *y* is a scalar.
- Notation:
  - *y* will denote the continuous model of the dependent variable
  - *t* will denote discrete noisy observations of the dependent variable (sometimes called the **target variable**).

# Where is the Linear in Linear Regression?



- In regression we assume that y is a function of **x**. The exact nature of this function is governed by an unknown parameter vector **w**:

$$y = y(\mathbf{x}, \mathbf{w})$$

- The regression is linear if $y$ is linear in **w**. In other words, we can express $y$ as

$$y = \mathbf{w}^t \phi(\mathbf{x})$$

where

$\phi(\mathbf{x})$ is some (potentially nonlinear) function of **x**.

# Linear Basis Function Models



- Generally

$$y(\mathbf{x}, \mathbf{w}) = \sum_{j=0}^{M-1} w_j \phi_j(\mathbf{x}) = \mathbf{w}^{\mathrm{T}} \boldsymbol{\phi}(\mathbf{x})$$

- where $\phi_j(\mathbf{x})$ are known as *basis functions.*
- Typically, $\Phi_0(\mathbf{x}) = 1$, so that $w_0$ acts as a bias.
- In the simplest case, we use linear basis functions : $\Phi_d(\mathbf{x}) = x_d$.

# Linear Regression Topics



- What is linear regression?
- **Example: polynomial curve fitting**
- Other basis families
- Solving linear regression problems
- Regularized regression
- Multiple linear regression
- Bayesian linear regression

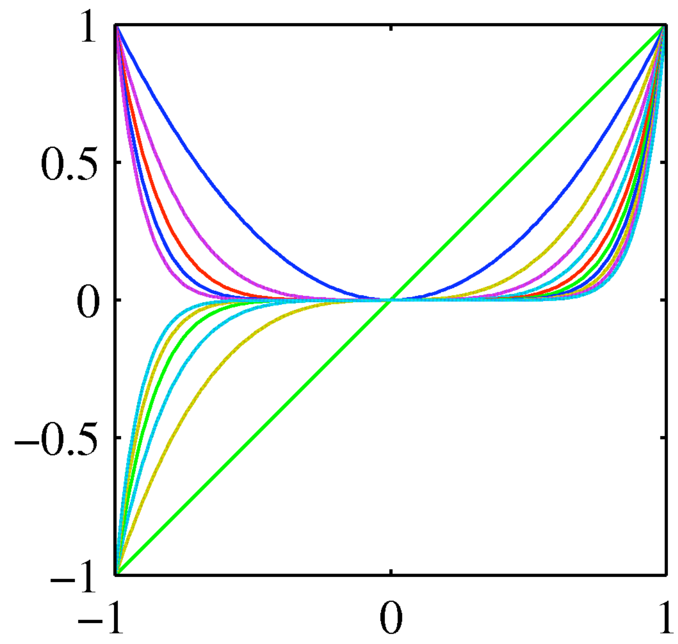
# Example: Polynomial Bases



- Polynomial basis functions:

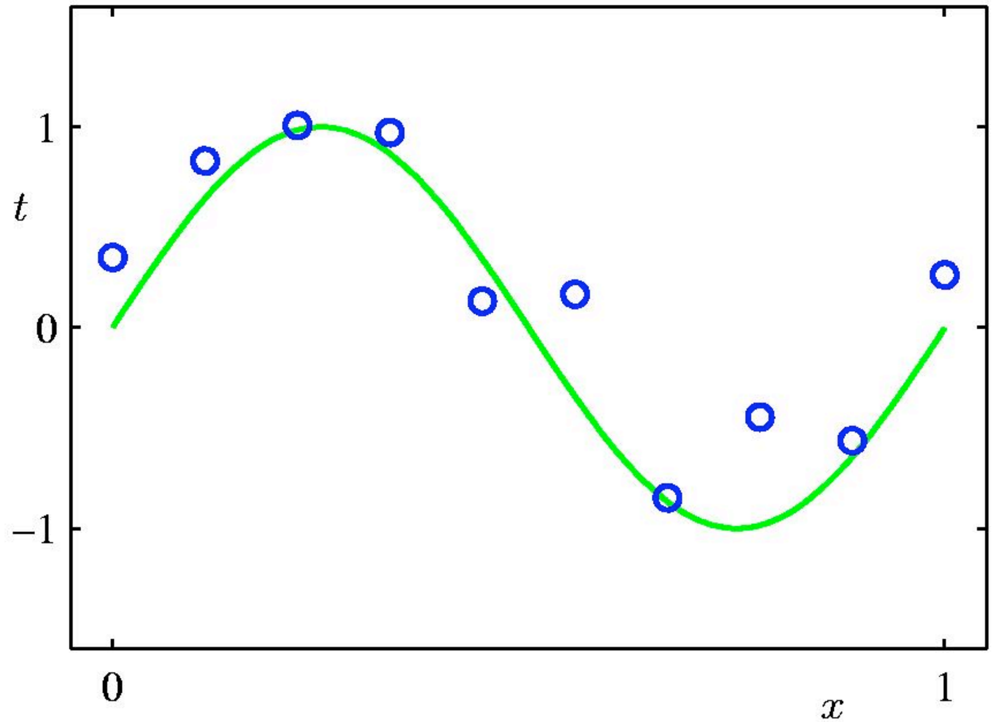
$$\phi_j(x) = x^j.$$

- These are global
  - a small change in x affects all basis functions.
  - A small change in a basis function affects y for all x.



# Example: Polynomial Curve Fitting

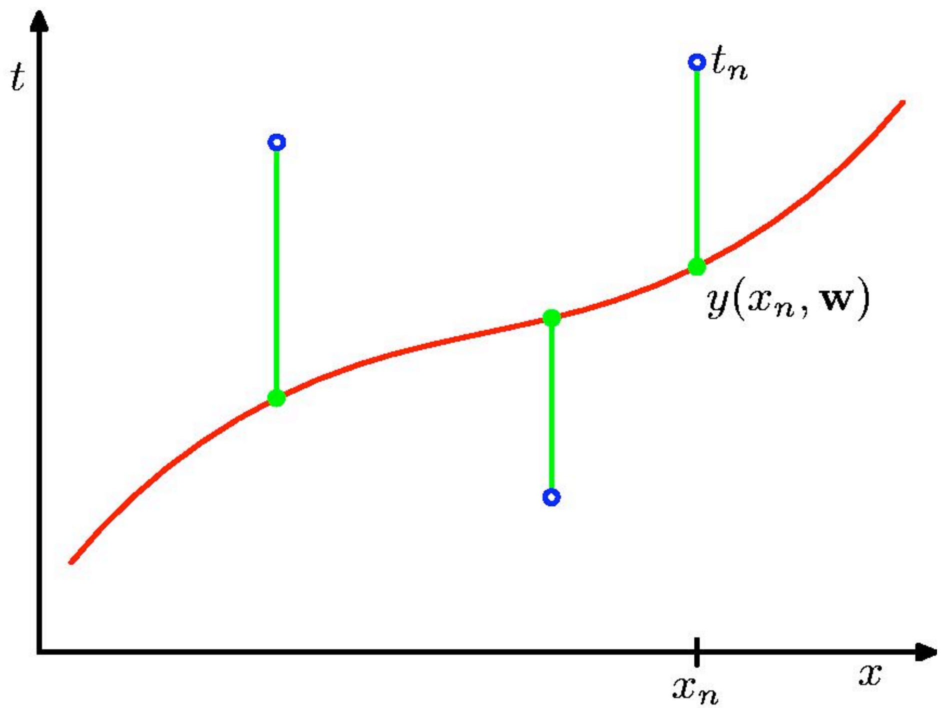





$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

# Sum-of-Squares Error Function

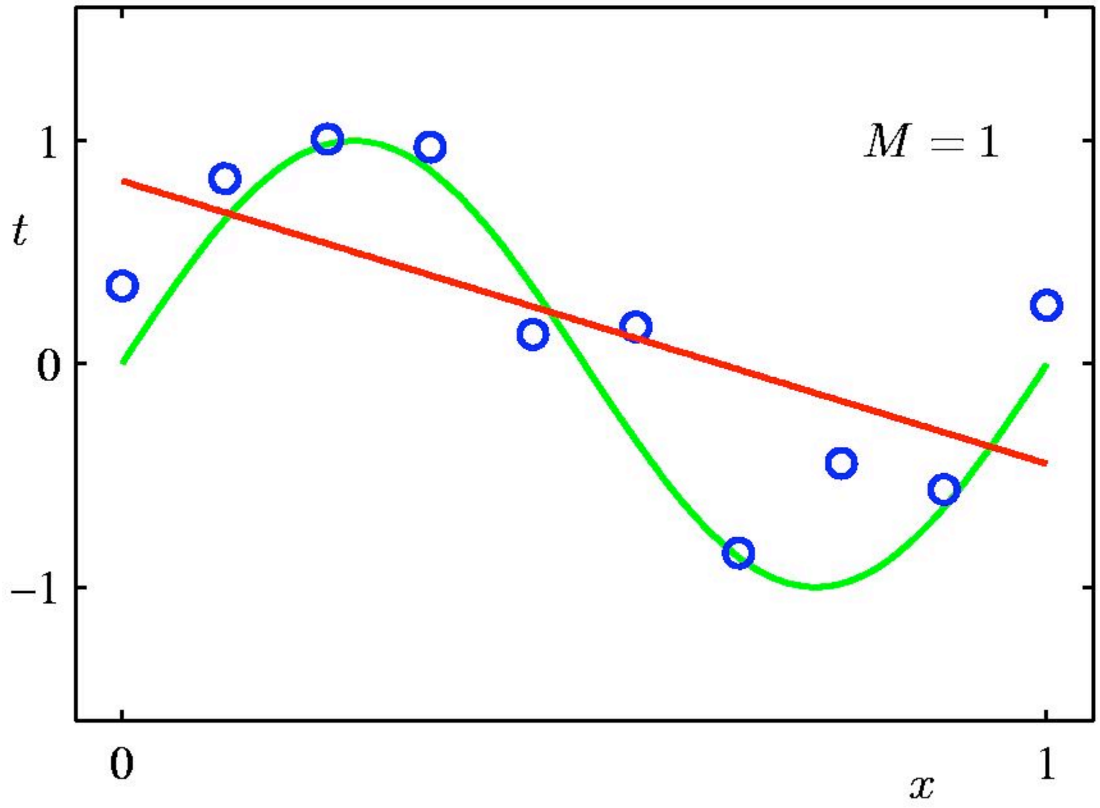





$$E(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n, \mathbf{w}) - t_n\}^2$$

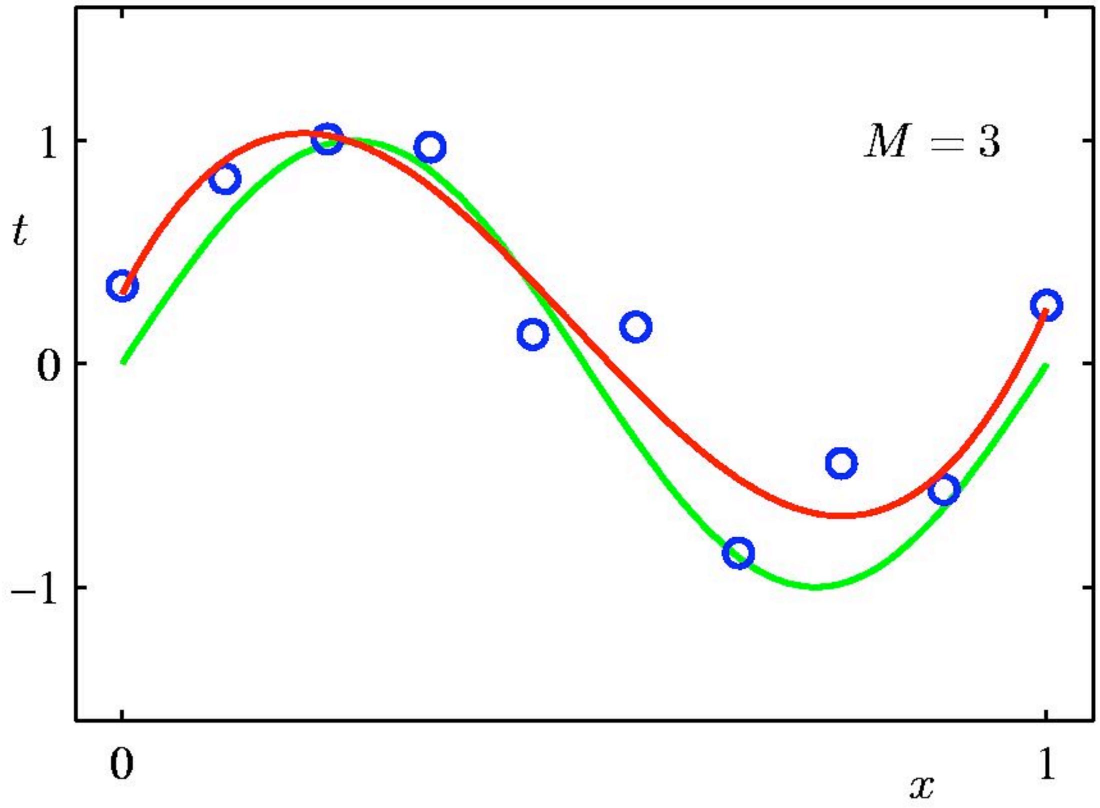
# 1st Order Polynomial





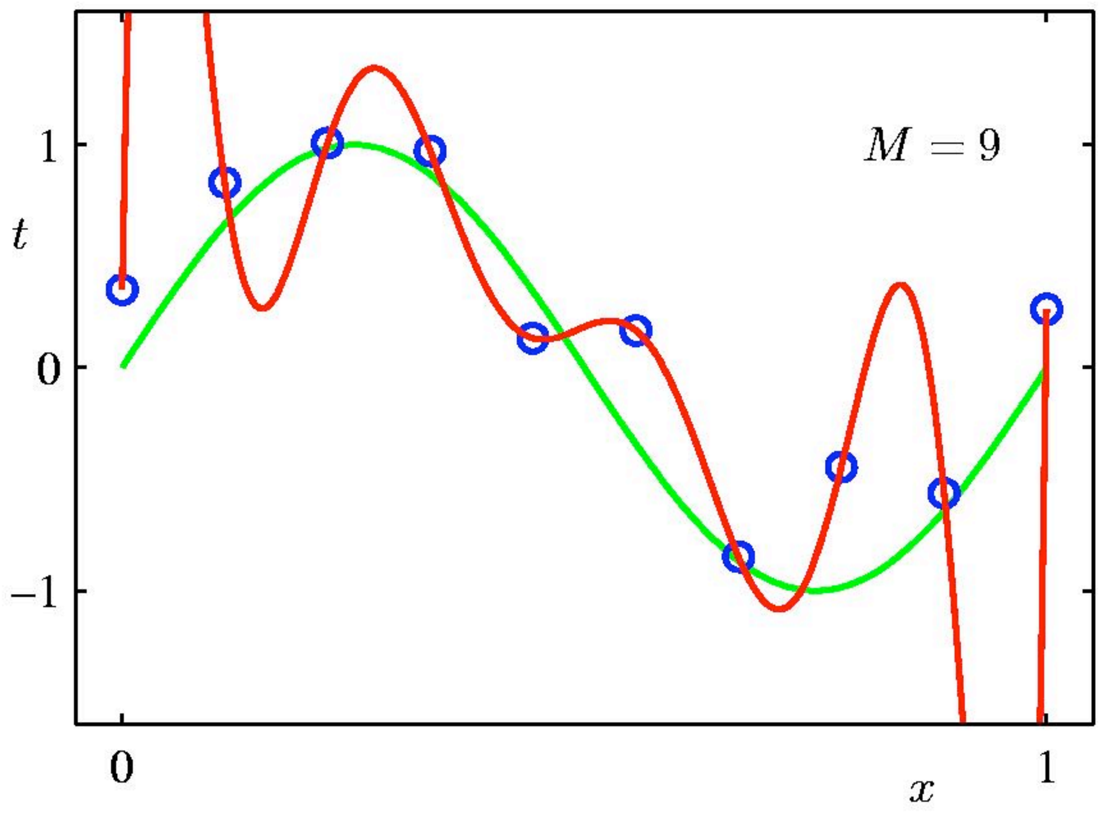
# 3rd Order Polynomial





# $9^{th}$ Order Polynomial





# Regularization



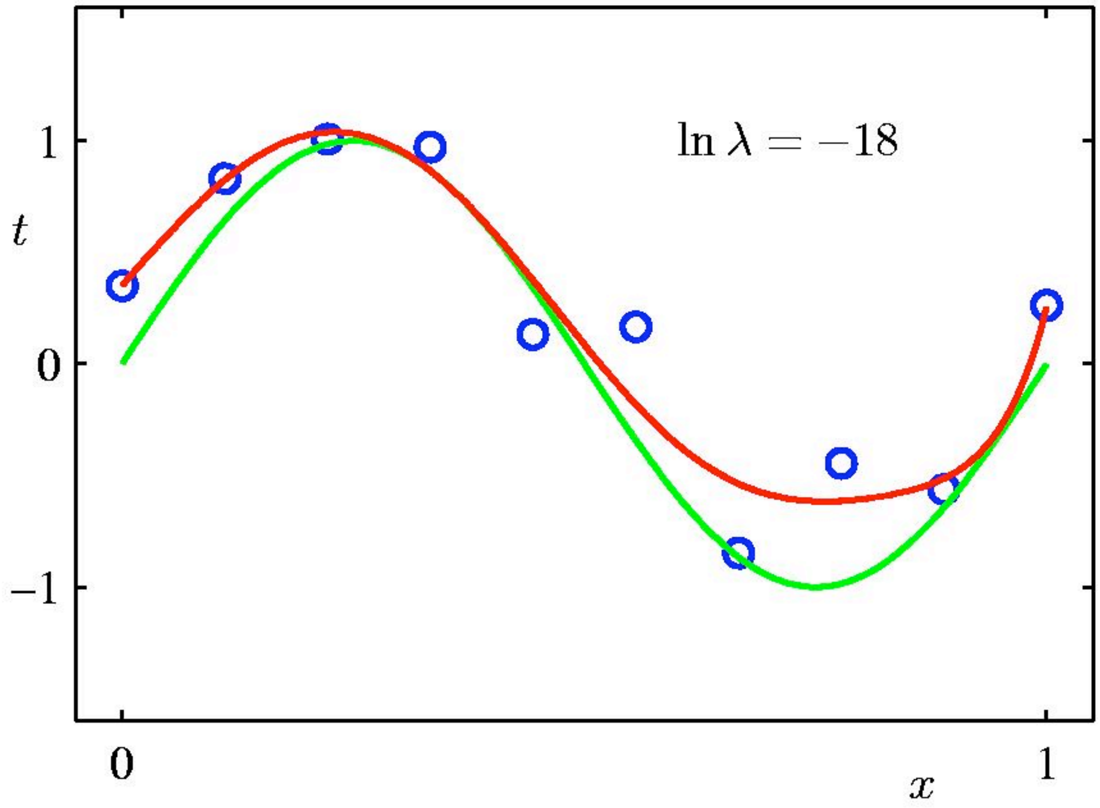
- Penalize large coefficient values

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

# Regularization



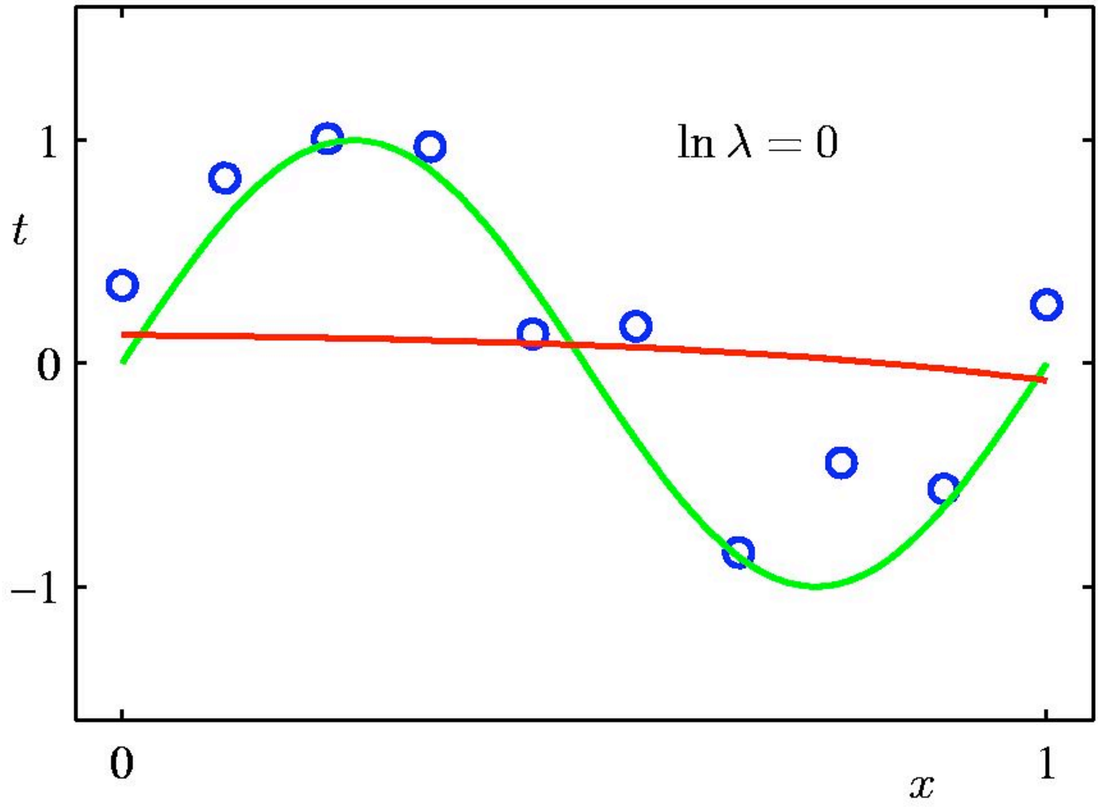
$9^{th}$ Order Polynomial



# Regularization



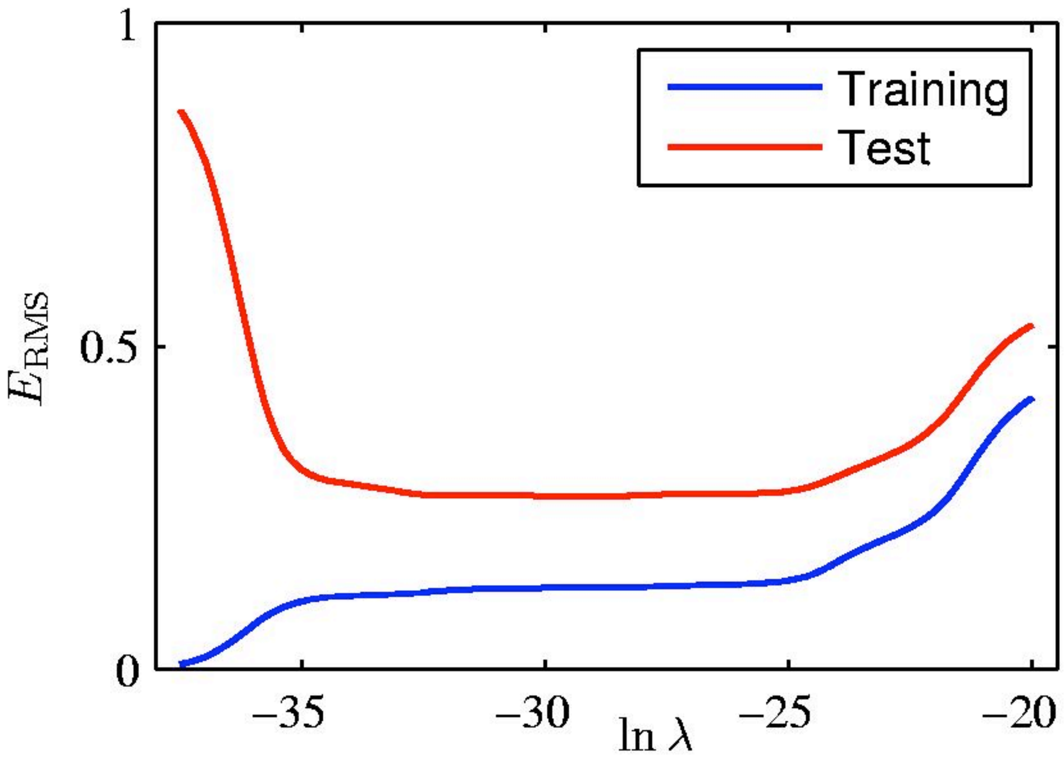
## 9th Order Polynomial



# Regularization



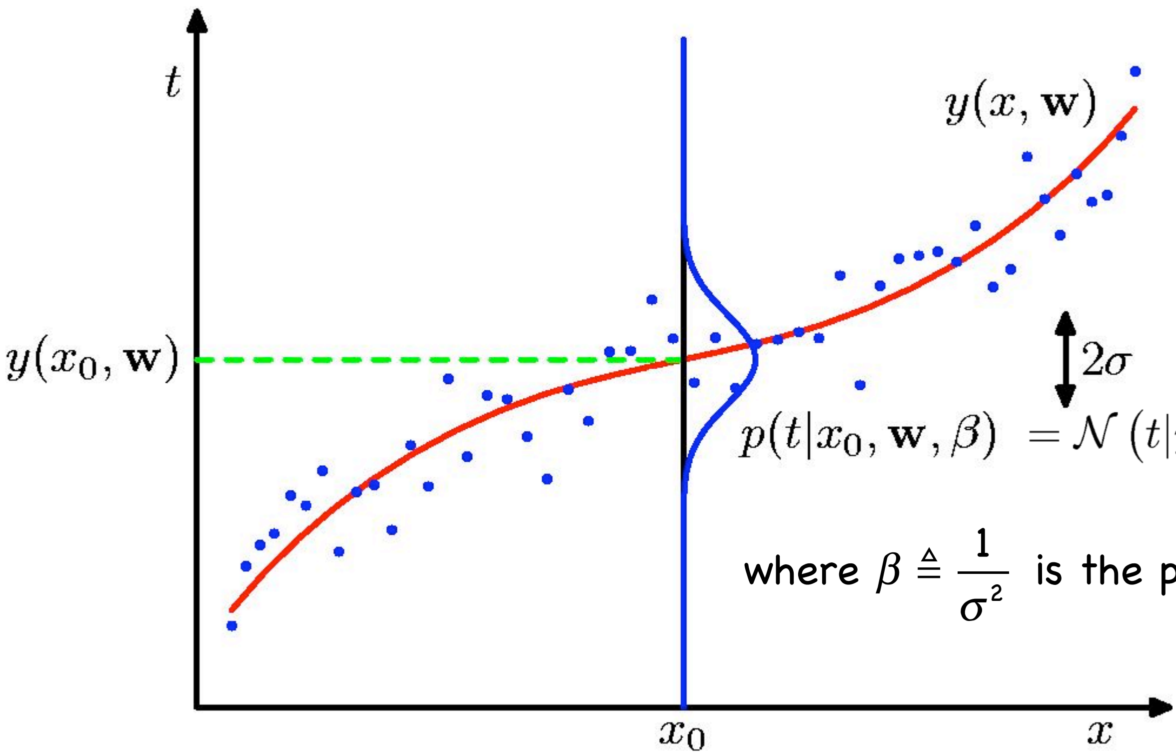
$9^{th}$ Order Polynomial



# Probabilistic View of Curve Fitting



- Why least squares?
- Model noise (deviation of data from model) as Gaussian i.i.d.




$$p(t|x_0, \mathbf{w}, \beta) = \mathcal{N}\left(t|y(x_0, \mathbf{w}), \beta^{-1}\right)$$

where $\beta \triangleq \dfrac{1}{\sigma^2}$ is the precision of the noise.

# Maximum Likelihood



$$p(\mathbf{t}|\mathbf{x},\mathbf{w},\beta) = \prod_{n=1}^{N} \mathcal{N}\left(t_n|y(x_n,\mathbf{w}),\beta^{-1}\right)$$

- We determine $\boldsymbol{w}_{ML}$ by minimizing the squared error $E(\boldsymbol{w})$.

$$\ln p(\mathbf{t}|\mathbf{x},\mathbf{w},\beta) = \underbrace{-\frac{\beta}{2}\sum_{n=1}^{N}\{y(x_n,\mathbf{w}) - t_n\}^2}_{\beta E(\mathbf{w})} + \frac{N}{2}\ln\beta - \frac{N}{2}\ln(2\pi)$$

- **Thus least-squares regression reflects an assumption that the noise is i.i.d. Gaussian.**

# Maximum Likelihood



$$p(\mathbf{t}|\mathbf{x},\mathbf{w},\beta) = \prod_{n=1}^{N} \mathcal{N}\left(t_n|y(x_n,\mathbf{w}),\beta^{-1}\right)$$

- We determine $\boldsymbol{w}_{ML}$ by minimizing the squared error $E(\boldsymbol{w})$.

$$\ln p(\mathbf{t}|\mathbf{x},\mathbf{w},\beta) = \underbrace{-\frac{\beta}{2}\sum_{n=1}^{N}\{y(x_n,\mathbf{w})-t_n\}^2}_{\beta E(\mathbf{w})} + \frac{N}{2}\ln\beta - \frac{N}{2}\ln(2\pi)$$

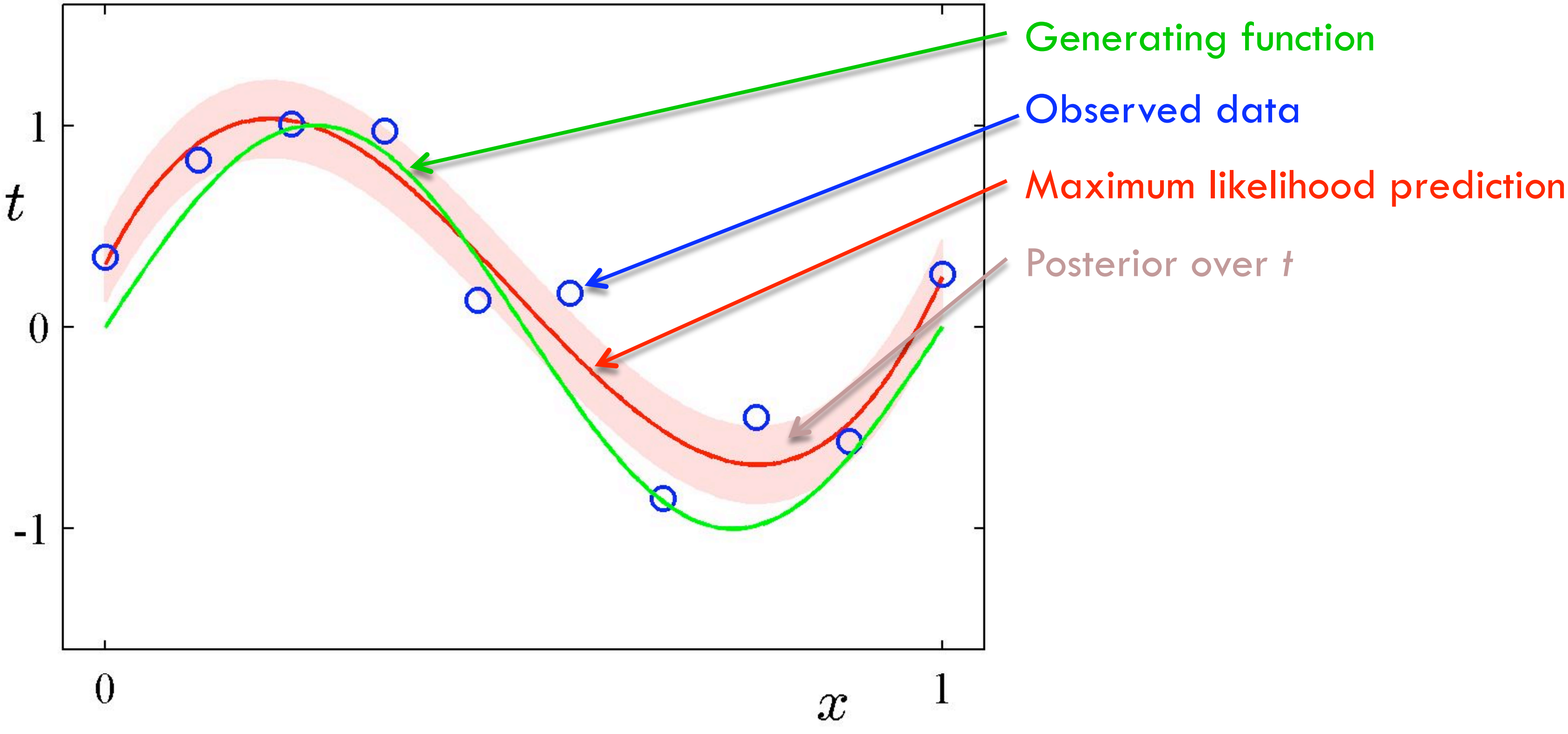
- Now given $\boldsymbol{w}_{ML}$, we can estimate the variance of the noise:

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

# Predictive Distribution



$$p(t|x, \mathbf{w}_{\mathrm{ML}}, \beta_{\mathrm{ML}}) = \mathcal{N}\left(t|y(x, \mathbf{w}_{\mathrm{ML}}), \beta_{\mathrm{ML}}^{-1}\right)$$



# MAP: A Step towards Bayes



- Prior knowledge about probable values of **w** can be incorporated into the regression:

$$p(\mathbf{w}|\alpha) = \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha^{-1}\mathbf{I}) = \left(\frac{\alpha}{2\pi}\right)^{(M+1)/2} \exp\left\{-\frac{\alpha}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w}\right\}$$

- Now the posterior over **w** is proportional to the product of the likelihood times the prior:

$$p(\mathbf{w}|\mathbf{x}, \mathbf{t}, \alpha, \beta) \propto p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta)p(\mathbf{w}|\alpha)$$

- The result is to introduce a new quadratic term in **w** into the error function to be minimized:

$$\beta\widetilde{E}(\mathbf{w}) = \frac{\beta}{2}\sum_{n=1}^{N}\{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\alpha}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w}$$

- **Thus regularized (ridge) regression reflects a 0-mean isotropic Gaussian prior on the weights.**

# Linear Regression Topics



- What is linear regression?
- Example: polynomial curve fitting
- **Other basis families**
- Solving linear regression problems
- Regularized regression
- Multiple linear regression
- Bayesian linear regression

# Gaussian Bases

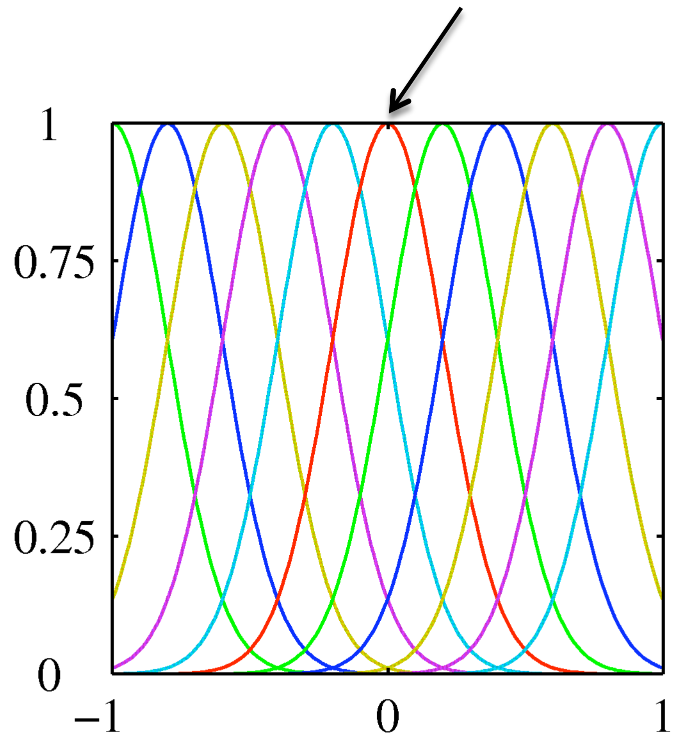


- Gaussian basis functions:

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

- These are local:
  - a small change in x affects only nearby basis functions.
  - a small change in a basis function affects y only for nearby x.
  - $\mu_j$ and s control location and scale (width).

Think of these as interpolation functions.



# Linear Regression Topics



- What is linear regression?
- Example: polynomial curve fitting
- Other basis families
- **Solving linear regression problems**
- Regularized regression
- Multiple linear regression
- Bayesian linear regression

# Maximum Likelihood and Linear Least Squares



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

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

- 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 and Linear Least Squares



- 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 and Least Squares



- 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

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

The Moore-Penrose pseudo-inverse, $\boldsymbol{\Phi}^{\dagger}$.

- where

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

# End of Lecture 8

# Linear Regression Topics



- What is linear regression?
- Example: polynomial curve fitting
- Other basis families
- Solving linear regression problems
- **Regularized regression**
- Multiple linear regression
- Bayesian linear regression

# Regularized Least Squares



- 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

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

- which is minimized by

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

$\lambda$ is called the regularization coefficient.

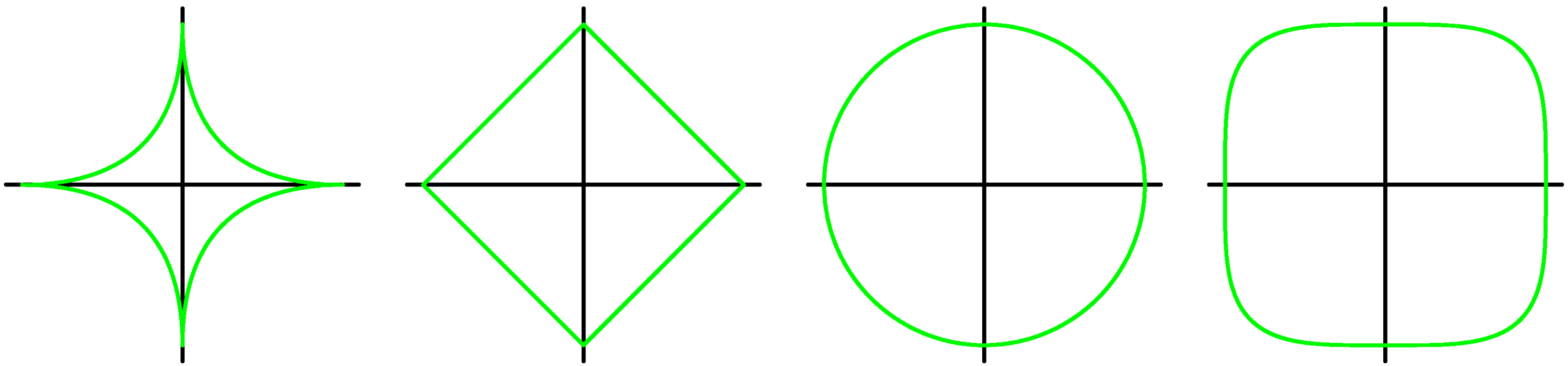
**Thus the name 'ridge regression'**

# Regularized Least Squares



- With a more general regularizer, we have

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



$q = 0.5$ $q = 1$ $q = 2$ $q = 4$

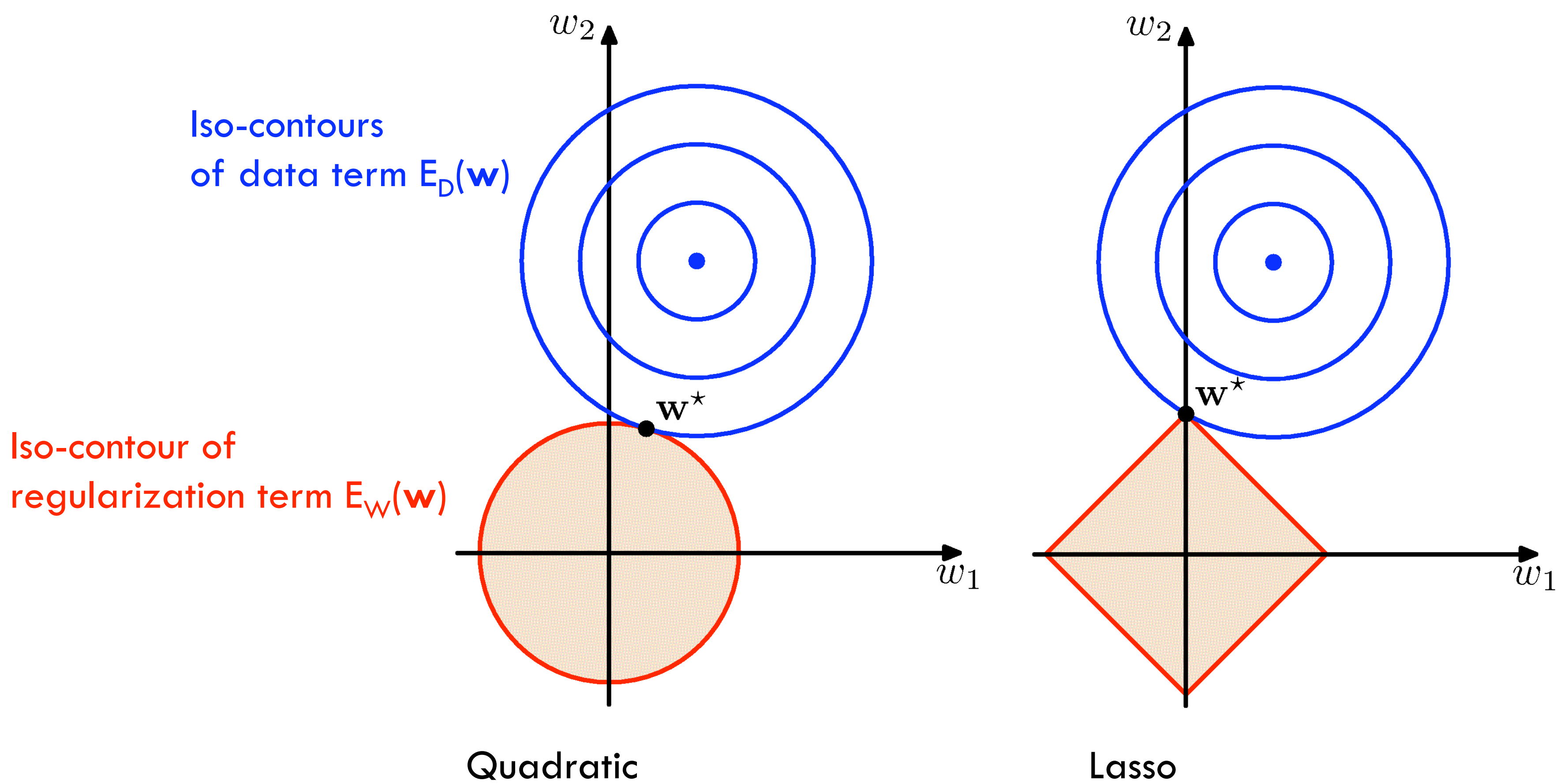
Lasso Quadratic

(Least absolute shrinkage and selection operator)

# Regularized Least Squares



- Lasso generates sparse solutions.



# Solving Regularized Systems



- Quadratic regularization has the advantage that the solution is closed form.
- Non-quadratic regularizers generally do not have closed form solutions
- Lasso can be framed as minimizing a quadratic error with linear constraints, and thus represents a convex optimization problem that can be solved by quadratic programming or other convex optimization methods.
- We will discuss quadratic programming when we cover SVMs

# Linear Regression Topics



- What is linear regression?
- Example: polynomial curve fitting
- Other basis families
- Solving linear regression problems
- Regularized regression
- **Multiple linear regression**
- Bayesian linear regression

# Multiple Outputs



- Analogous to the single output case we have:

$$\begin{aligned} p(\mathbf{t}|\mathbf{x}, \mathbf{W}, \beta) &= \mathcal{N}(\mathbf{t}|\mathbf{y}(\mathbf{W}, \mathbf{x}), \beta^{-1}\mathbf{I}) \\ &= \mathcal{N}(\mathbf{t}|\mathbf{W}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}), \beta^{-1}\mathbf{I}). \end{aligned}$$

- Given observed inputs $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ , and targets $\mathbf{T} = [\mathbf{t}_1, \ldots, \mathbf{t}_N]^{\mathrm{T}}$

  we obtain the log likelihood function

$$\begin{aligned} \ln p(\mathbf{T}|\mathbf{X}, \mathbf{W}, \beta) &= \sum_{n=1}^{N} \ln \mathcal{N}(\mathbf{t}_n|\mathbf{W}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n), \beta^{-1}\mathbf{I}) \\ &= \frac{NK}{2} \ln\left(\frac{\beta}{2\pi}\right) - \frac{\beta}{2} \sum_{n=1}^{N} \left\| \mathbf{t}_n - \mathbf{W}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n) \right\|^2. \end{aligned}$$

# Multiple Outputs



- Maximizing with respect to W, we obtain

$$\mathbf{W}_{\mathrm{ML}} = \left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{T}.$$

- If we consider a single target variable, $t_k$, we see that

$$\mathbf{w}_k = \left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}\mathbf{t}_k = \mathbf{\Phi}^{\dagger}\mathbf{t}_k$$

- where $\mathbf{t}_k = [t_{1k}, \ldots, t_{Nk}]^{\mathrm{T}}$, which is identical with the single output case.

# Some Useful MATLAB Functions



- polyfit
  - Least-squares fit of a polynomial of specified order to given data
- regress
  - More general function that computes linear weights for least-squares fit

# Linear Regression Topics



- What is linear regression?
- Example:  polynomial curve fitting
- Other basis families
- Solving linear regression problems
- Regularized regression
- Multiple linear regression
- **Bayesian linear regression**

# Bayesian Linear Regression

Rev. Thomas Bayes, 1702 - 1761

# Bayesian Linear Regression



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

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

# Bayesian Linear Regression



- A common choice for the prior is

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

- for which

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

- Thus $\boldsymbol{m}_N$ represents the ridge regression solution with

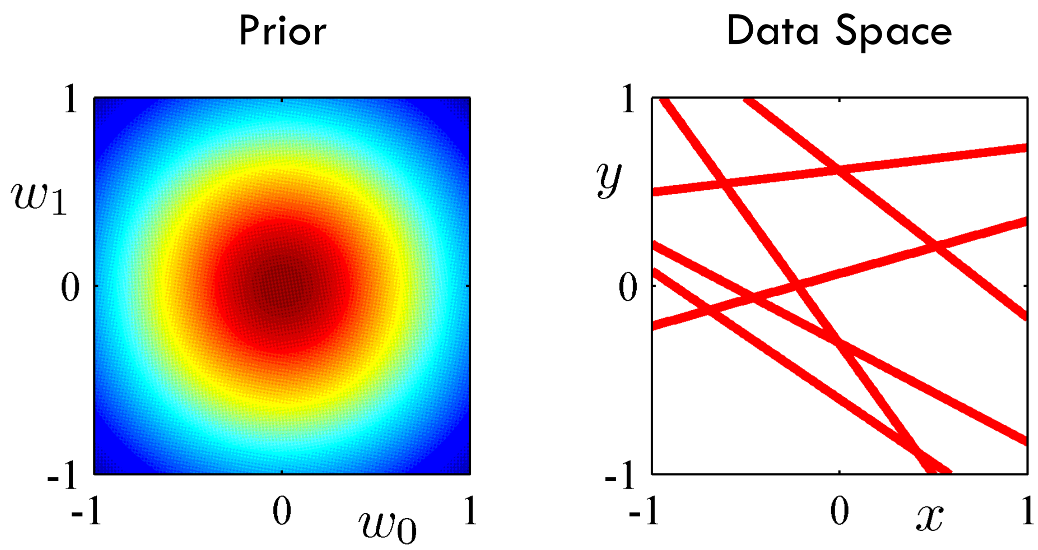
$$\lambda = \alpha / \beta$$

- Next we consider an example …

# Bayesian Linear Regression



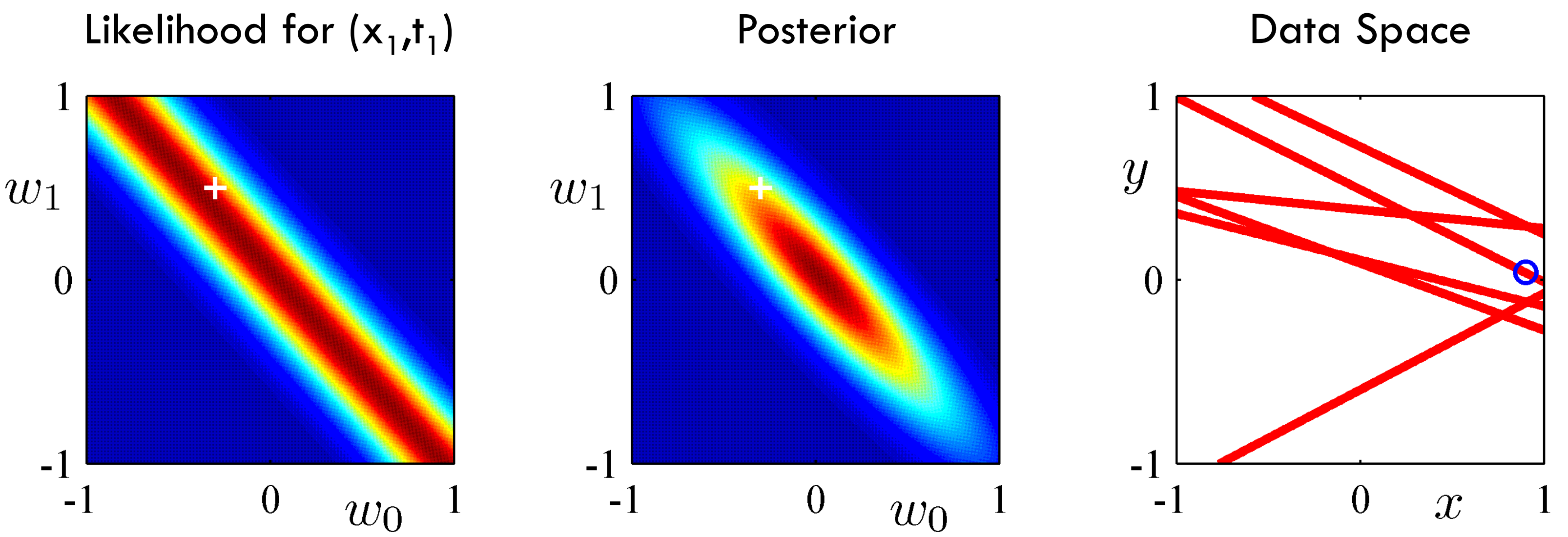
## 0 data points observed



# Bayesian Linear Regression

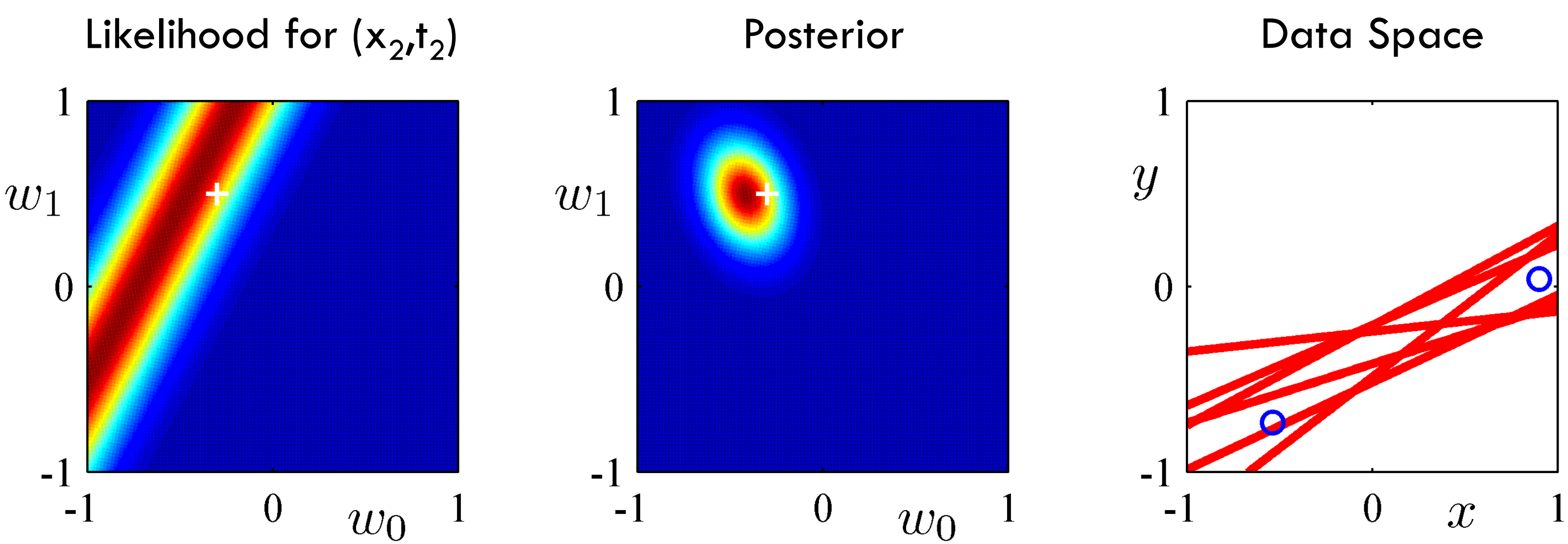


## 1 data point observed



# Bayesian Linear Regression

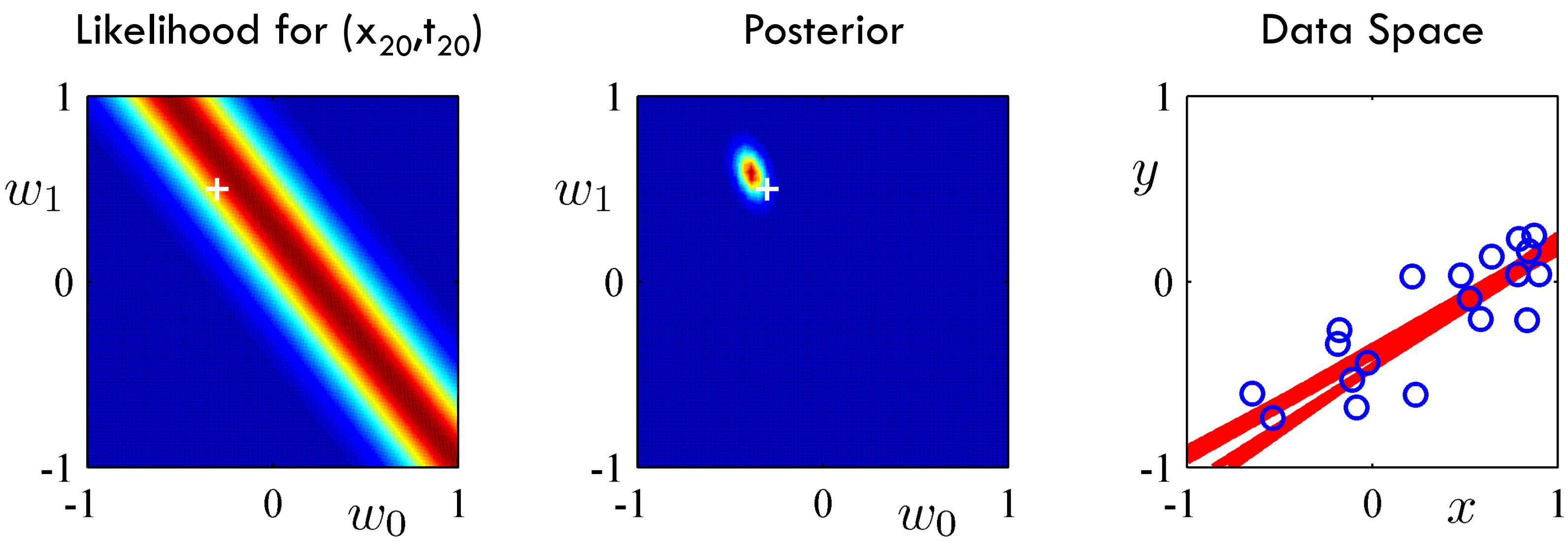
## 2 data points observed



# Bayesian Linear Regression



## 20 data points observed



# Predictive Distribution




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

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

- where

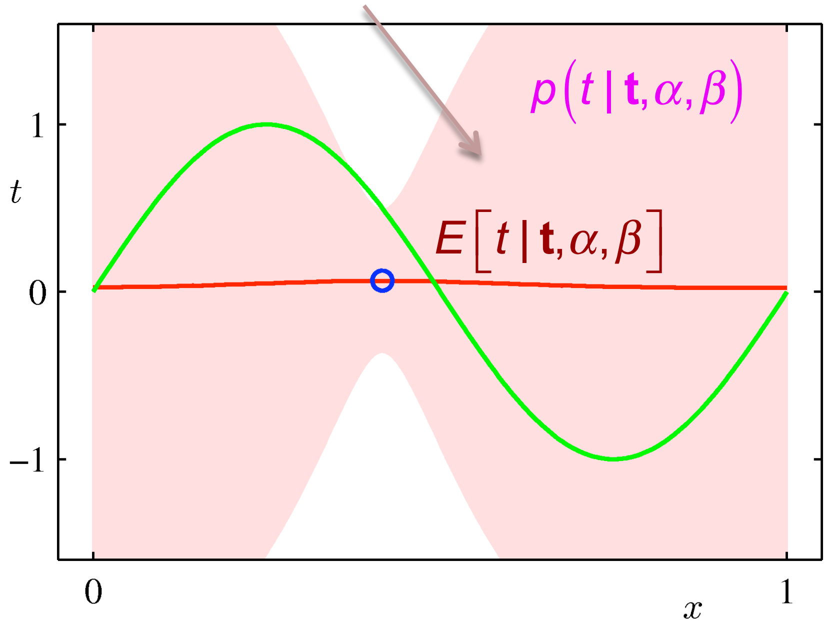
$$\sigma_N^2(\mathbf{x}) = \frac{1}{\beta} + \boldsymbol{\phi}(\mathbf{x})^{\mathrm{T}}\mathbf{S}_N\boldsymbol{\phi}(\mathbf{x}).$$

# Predictive Distribution

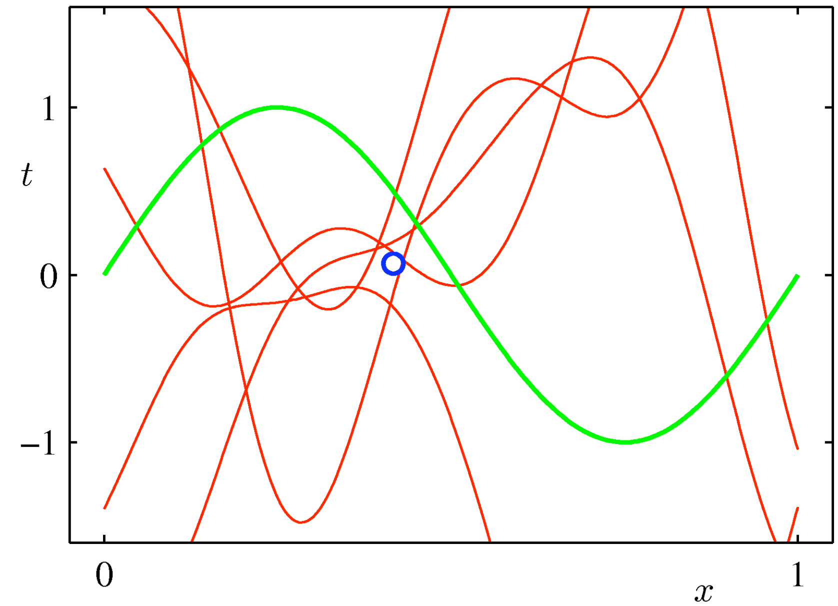


- Example: Sinusoidal data, 9 Gaussian basis functions, 1 data point

**Notice how much bigger our uncertainty is relative to the ML method!!**



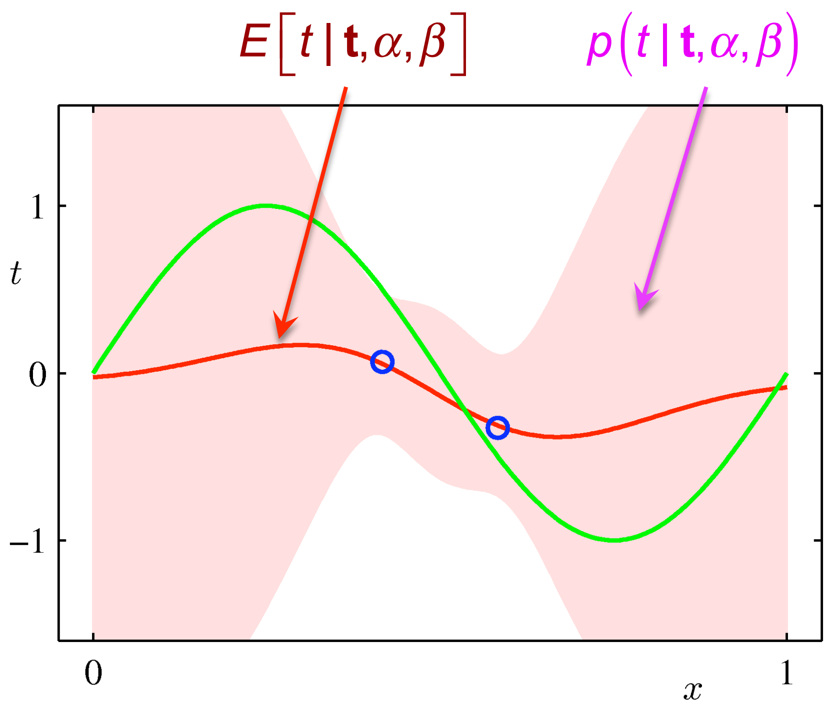

Samples of y(x,**w**)

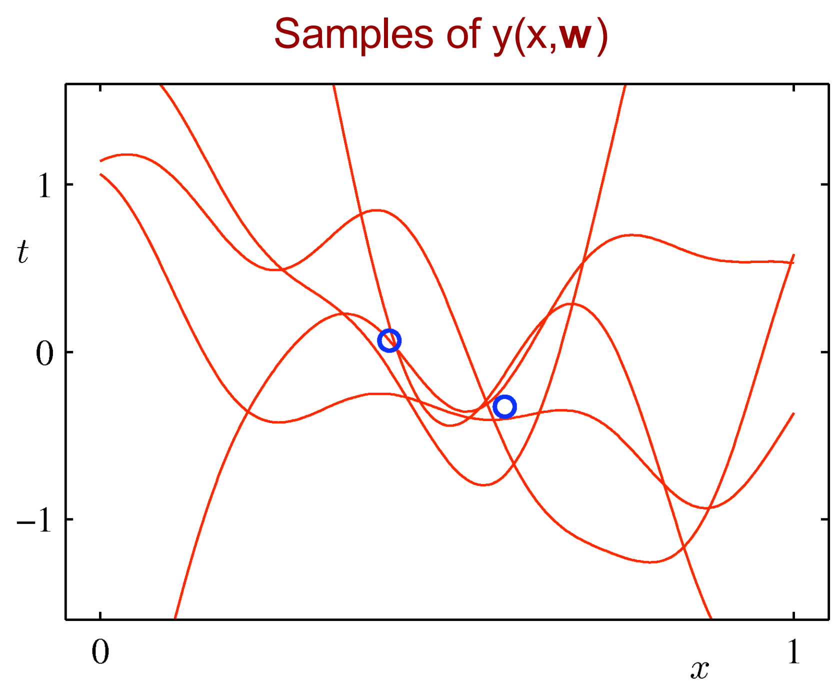


# Predictive Distribution




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

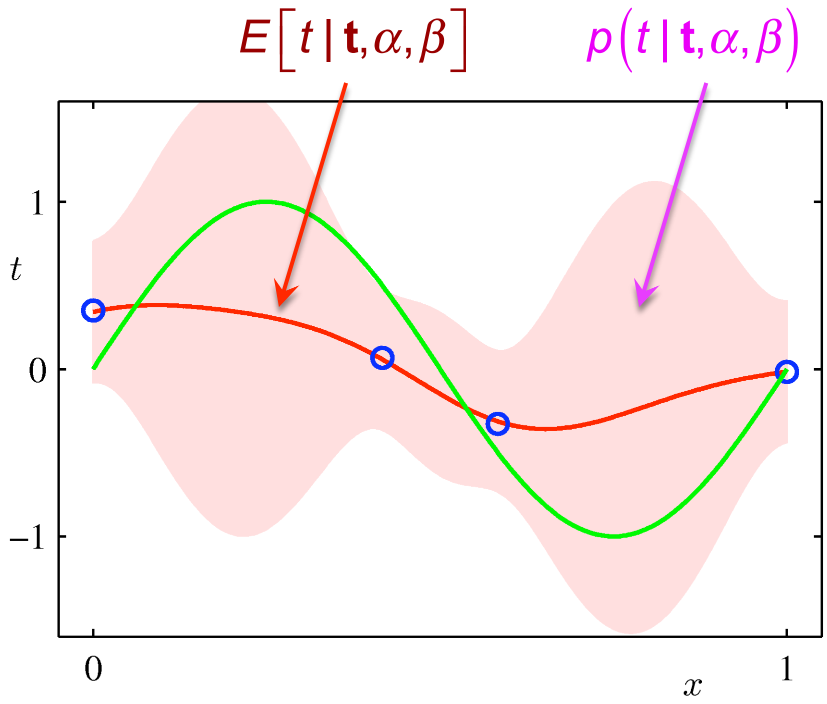


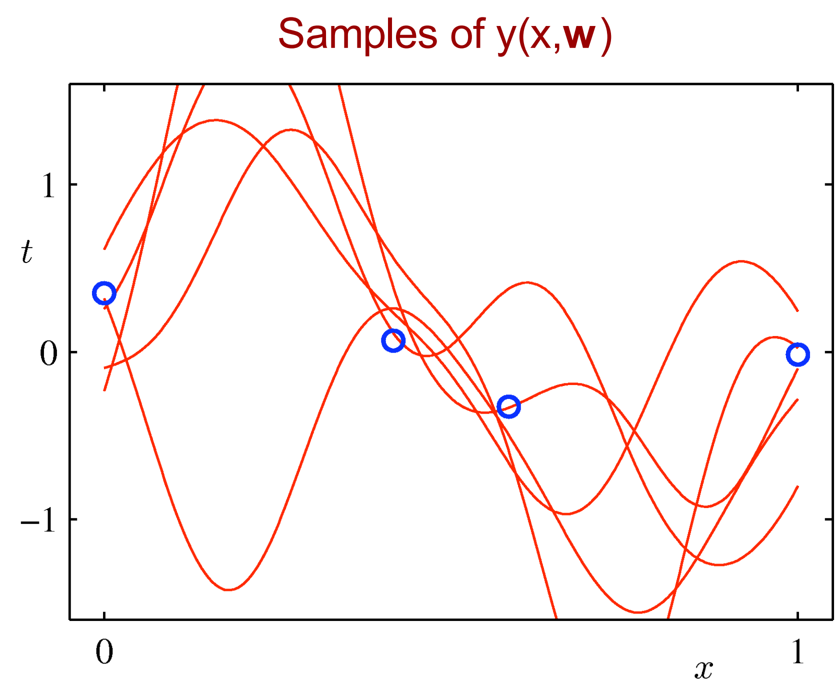



# Predictive Distribution



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

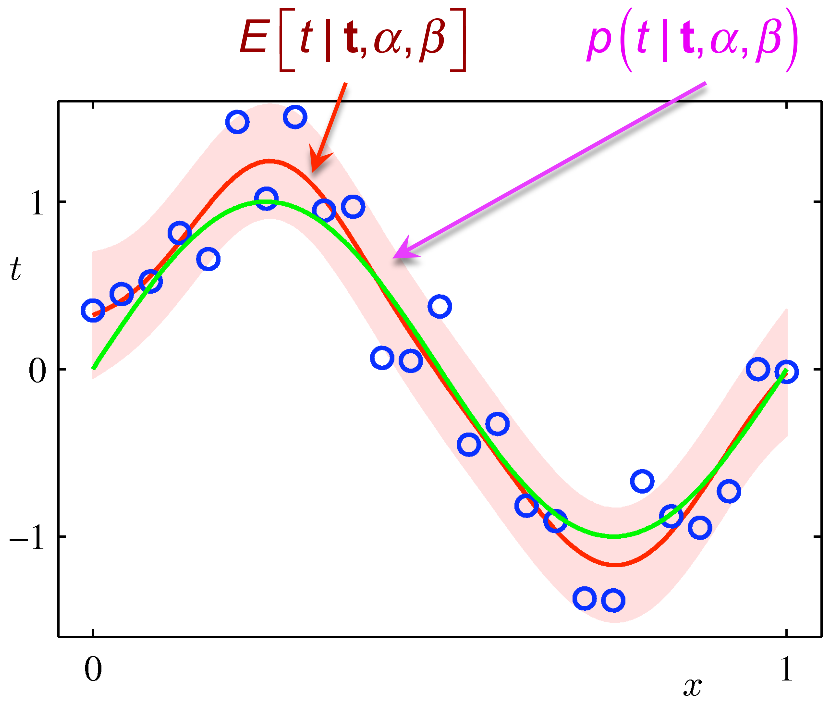


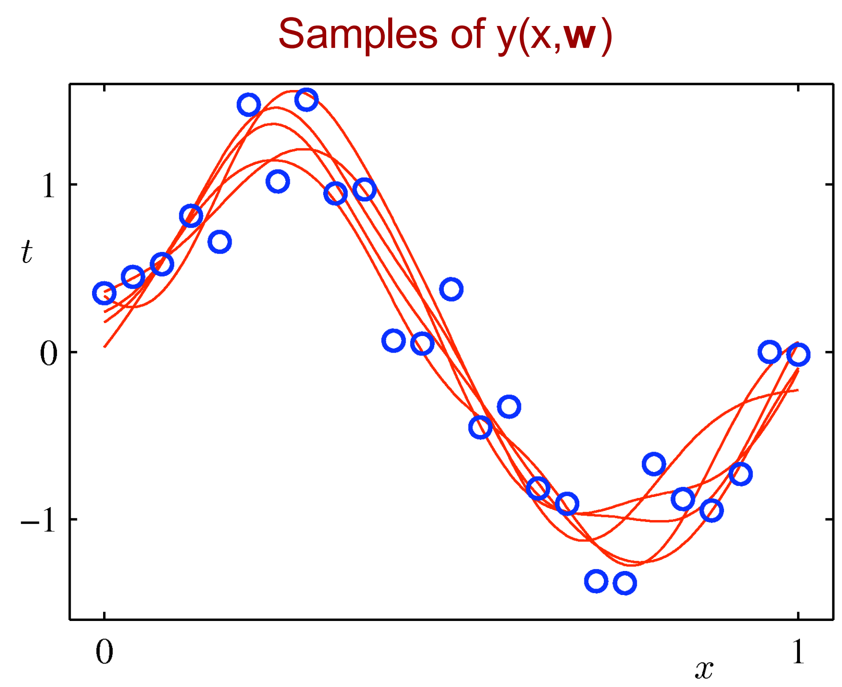



# Predictive Distribution



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



# Equivalent Kernel



- The predictive mean can be written

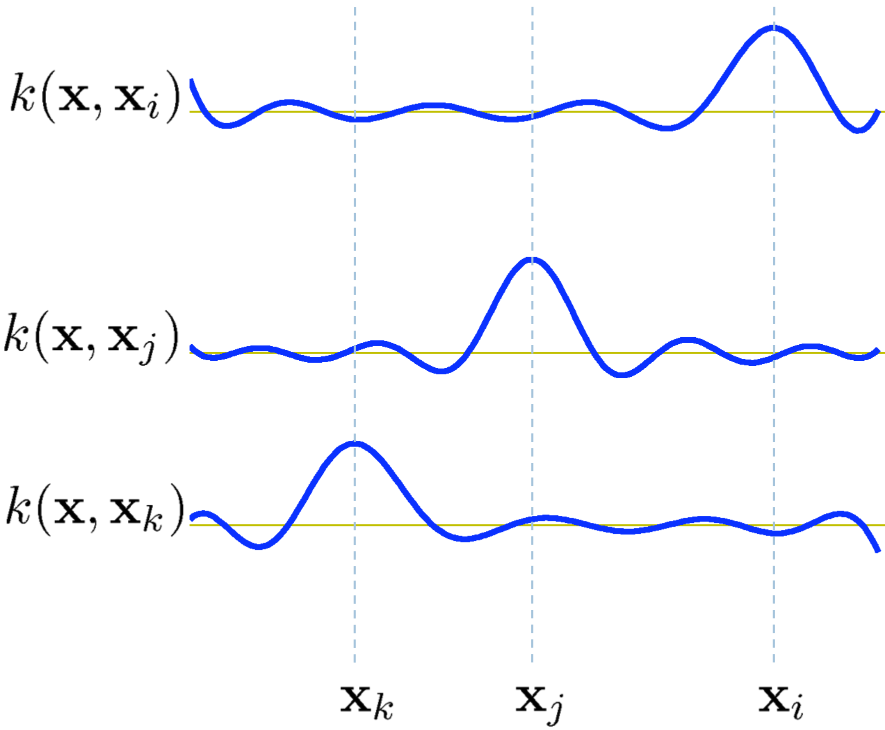
$$\begin{aligned} y(\mathbf{x}, \mathbf{m}_N) &= \mathbf{m}_N^{\mathrm{T}} \boldsymbol{\phi}(\mathbf{x}) = \beta \boldsymbol{\phi}(\mathbf{x})^{\mathrm{T}} \mathbf{S}_N \boldsymbol{\Phi}^{\mathrm{T}} \mathbf{t} \\ &= \sum_{n=1}^{N} \beta \boldsymbol{\phi}(\mathbf{x})^{\mathrm{T}} \mathbf{S}_N \boldsymbol{\phi}(\mathbf{x}_n) t_n \\ &= \sum_{n=1}^{N} k(\mathbf{x}, \mathbf{x}_n) t_n . \end{aligned}$$

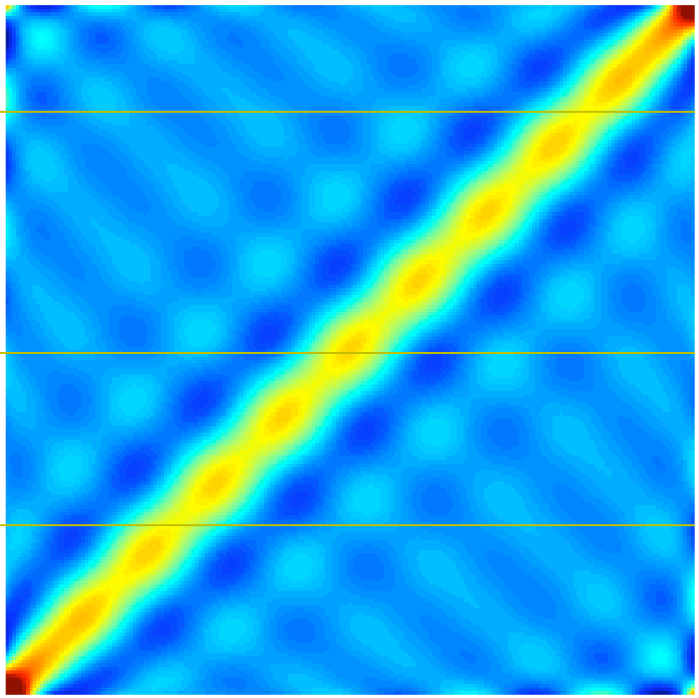
*Equivalent kernel or smoother matrix.*

- This is a weighted sum of the training data target values, $t_n$.

# Equivalent Kernel





Weight of $t_n$ depends on distance between $x$ and $x_n$; nearby $x_n$ carry more weight.

# Linear Regression Topics



- What is linear regression?
- Example: polynomial curve fitting
- Other basis families
- Solving linear regression problems
- Regularized regression
- Multiple linear regression
- Bayesian linear regression