Linear Models for Regression

CS534

Adapted from slides of C. Bishop

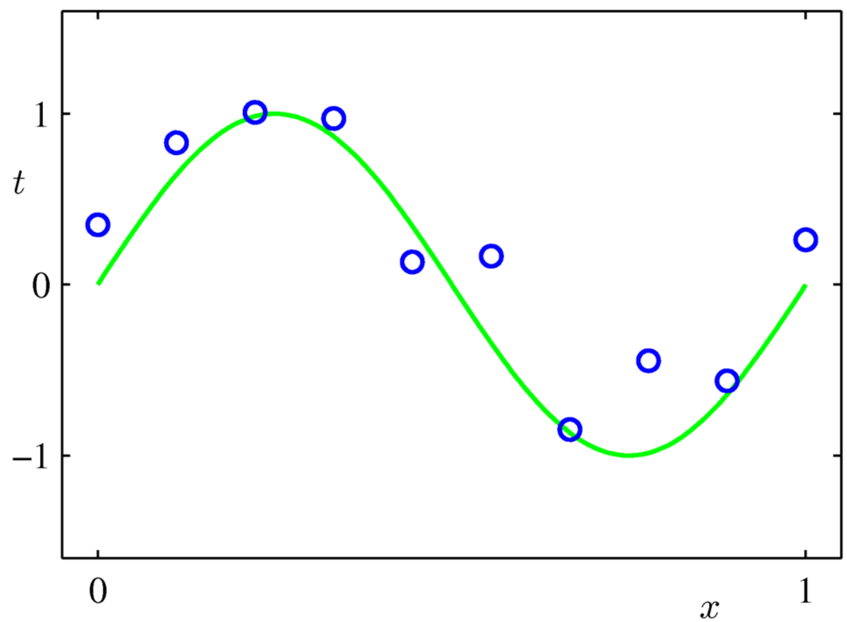
Prediction Problems

- Predict housing price based on
  - House size, lot size, Location, # of rooms …
- Predict stock price based on
  - Price history of the past month …
- Predict the abundance of a species based on
  - Environmental conditions
- General set up:

  Given a set of training examples ($\mathbf{x}_i$, t_i), i =1, …N

  Goal: learn a function $\hat{y}(\mathbf{x})$ *to minimize some loss function:* $L(\hat{y}, t)$

- 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 (1)

- Generally

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

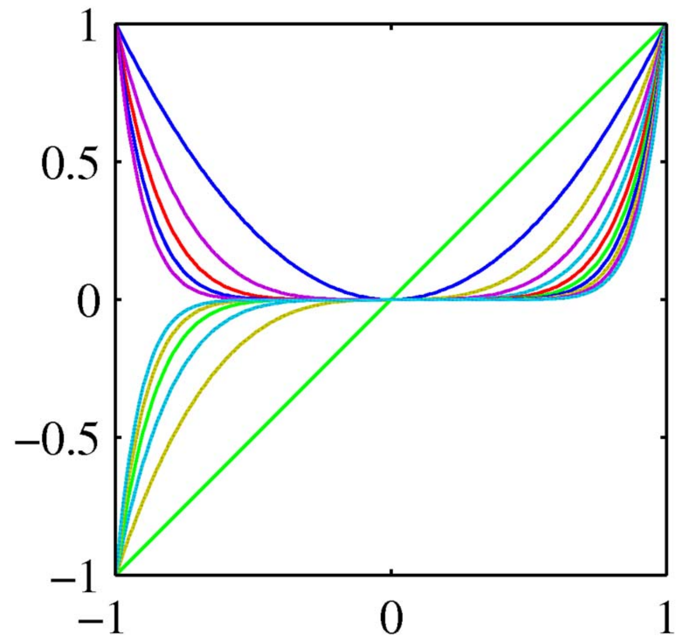
- where ϕ_j's are known as ***basis functions***.
- Typically $\phi_0 = 1$ so that w_0 acts as a ***bias***.
- In the simplest case, we use linear basis functions : $\phi_i(\mathbf{x}) = x_i$
  - Multiple linear regression

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.

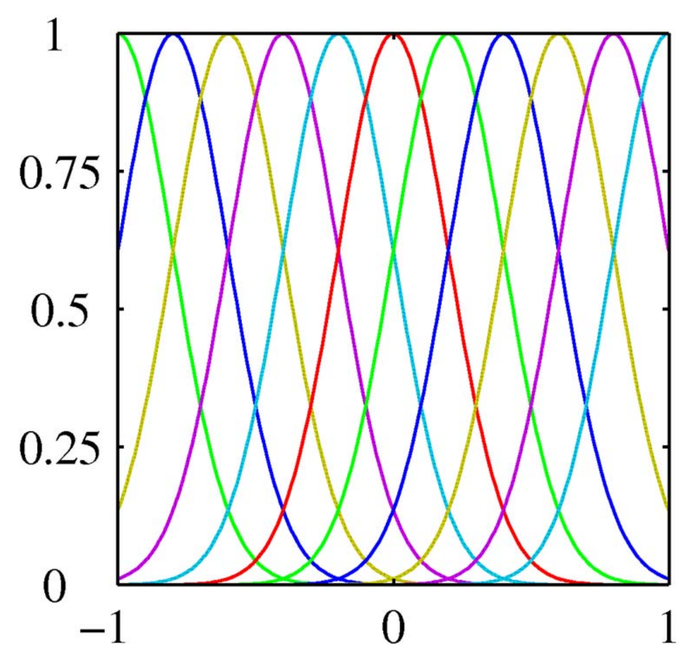


Linear Basis Function Models (3)

- 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. μ_j and s control the location and scale (width).



Linear Basis Function Models (4)

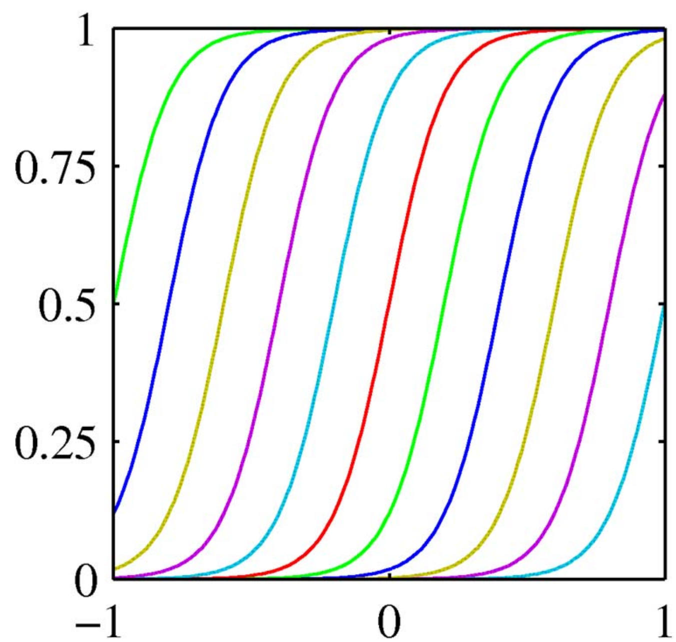
- Sigmoidal basis functions:

$$\phi_j(x) = \sigma\left(\frac{x - \mu_j}{s}\right)$$

- where

$$\sigma(a) = \frac{1}{1 + \exp(-a)}.$$

- Also these are local; a small change in x only affect nearby basis functions. μ_j and s control location and scale (slope).



Maximum Likelihood Estimation of **w**

- **Assumption:** observations drawn 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,

$$p(t|\mathbf{x}, \mathbf{w}, \beta) = \mathcal{N}(t|y(\mathbf{x}, \mathbf{w}), \beta^{-1}) = \sqrt{\frac{\beta}{2\pi}} \exp\left\{-\frac{1}{2}\beta(t - y(\mathbf{x}, \mathbf{w}))^2\right\}$$

- Given a set of observed inputs, $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, and their corresponding targets, $\mathbf{t} = [t_1, \ldots, t_N]^{\mathrm{T}}$, if we assume that the parameters take specific values **w** and β, the likelihood of observing the data is:s

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

i.i.d. assumption

Maximum Likelihood Estimation

- Taking the logarithm of the likelihood ftn, we get

$$\ln p(\mathbf{t}\,|\mathbf{w},\beta) = \sum_{n=1}^{N} \ln N(t_n|\mathbf{w}^{\boldsymbol{T}}\,\phi(\mathrm{x_n}), \boldsymbol{\beta}^{-\mathbf{1}})$$

$$= \frac{N}{2}\ln\frac{\beta}{2\pi} - \frac{\beta}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^{\boldsymbol{T}}\boldsymbol{\phi}(\boldsymbol{x_n})\}^2$$

- Note that

$$\underset{\mathbf{w}}{\operatorname{argmax}} \ln p(\mathbf{t}\,|\mathbf{w},\beta) = \underset{\mathbf{w}}{\operatorname{argmin}}\, E_D(\mathbf{w})$$

$$E_D(\mathbf{w}) = \frac{\mathbf{1}}{\mathbf{2}}\sum_{\boldsymbol{n}=\mathbf{1}}^{\boldsymbol{N}}\left(\boldsymbol{t_n} - \boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{\phi}(\boldsymbol{x_n})\right)^{\mathbf{2}}$$

- $E_D(\mathbf{w})$ is called the least -square objective.
- Maximizing likelihood = least squares

Maximum Likelihood Estimation

- 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

$$\mathbf{w}_{\mathrm{ML}} = \overbrace{\left(\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{\Phi} \right)^{-1} \boldsymbol{\Phi}^{\mathrm{T}}}^{\text{The Moore-Penrose pseudo-inverse, } \boldsymbol{\Phi}^{\dagger}.} \mathbf{t}$$

- where

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

A single example

N x M

A basis function

Maximum Likelihood and Least Squares (4)

- Maximizing with respect to the bias, w_0, alone, we see that

$$\begin{aligned} w_0 &= \bar{t} - \sum_{j=1}^{M-1} w_j \overline{\phi_j} \\ &= \frac{1}{N}\sum_{n=1}^{N} t_n - \sum_{j=1}^{M-1} w_j \frac{1}{N}\sum_{n=1}^{N} \phi_j(\mathbf{x}_n). \end{aligned}$$

- We can also maximize with respect to β, giving

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

System Equation View of Linear Regression

$$\boldsymbol{t} = \begin{bmatrix} t_1 \\ \dots \\ t_N \end{bmatrix} \quad \boldsymbol{\phi} = \begin{bmatrix} \phi_0(\mathbf{x_1}) & \dots & \phi_{m-1}(\mathbf{x_1}) \\ & \dots & \\ \phi_0(\mathbf{x}_N) & \dots & \phi_{m-1}(\mathbf{x}_N) \end{bmatrix} \quad \mathbf{w} = \begin{bmatrix} w_0 \\ \dots \\ w_{m-1} \end{bmatrix}$$

$$\boldsymbol{t} = \boldsymbol{\phi}\boldsymbol{w}$$

- Over-constrained system of equations
- There exists no solution
- Maximum likelihood and Least squared solution

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

Geometry of Least Squares

- Consider

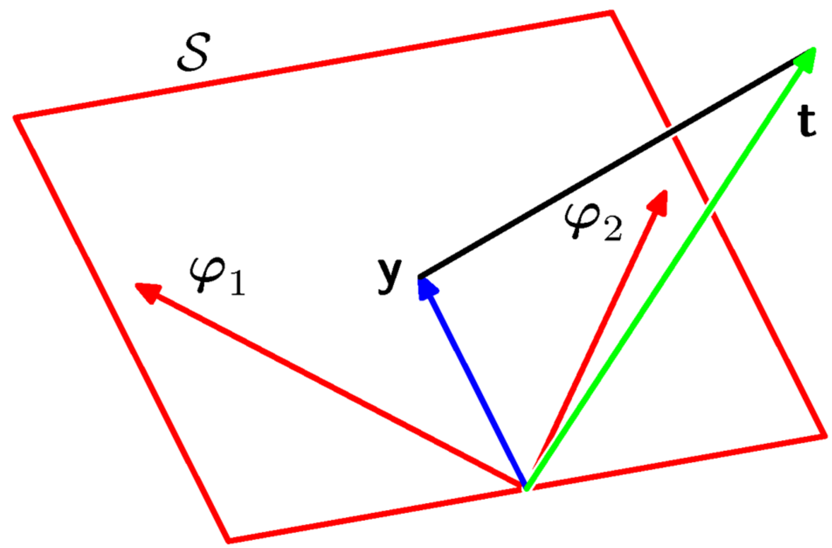
$$\boldsymbol{y} = \boldsymbol{\phi} \boldsymbol{w_{ML}} = [\phi_0 \quad \cdots \quad \boldsymbol{\phi_{m-1}}]\boldsymbol{w_{ML}} = \boldsymbol{w_0}\boldsymbol{\phi_0} + \cdots + \boldsymbol{w_{m-1}}\boldsymbol{\phi_{m-1}}$$

$$\mathbf{y} \in \mathcal{S} \subseteq \mathcal{T} \qquad \mathbf{t} \in \mathcal{T}$$

N-dimensional

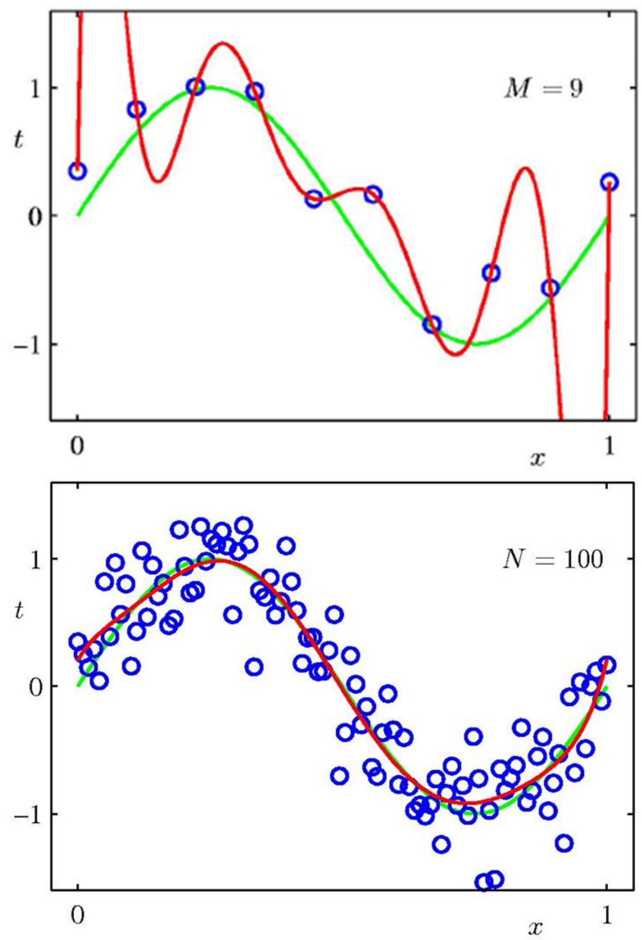
M-dimensional

- **t** is a n-d vector
- S is the space spanned by $\phi_i's$ and $t \notin S$
- w_{ML} minimizes the distance between ***t*** and S by finding the projection of ***t*** onto S



$$\boldsymbol{y} = \boldsymbol{\phi}\left(\boldsymbol{\phi}^T\boldsymbol{\phi}\right)^{-1}\boldsymbol{\phi}^T\boldsymbol{t}$$

Over-fitting issue



- What can we do to curb overfitting
  - Use less complex model
  - Use more training examples
  - Regularization

Regularized Least Squares (1)

- Consider the error function:

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

Data term + Regularization term (penalize complex models)

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

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

Encourage small weight values

λ is called the regularization coefficient.

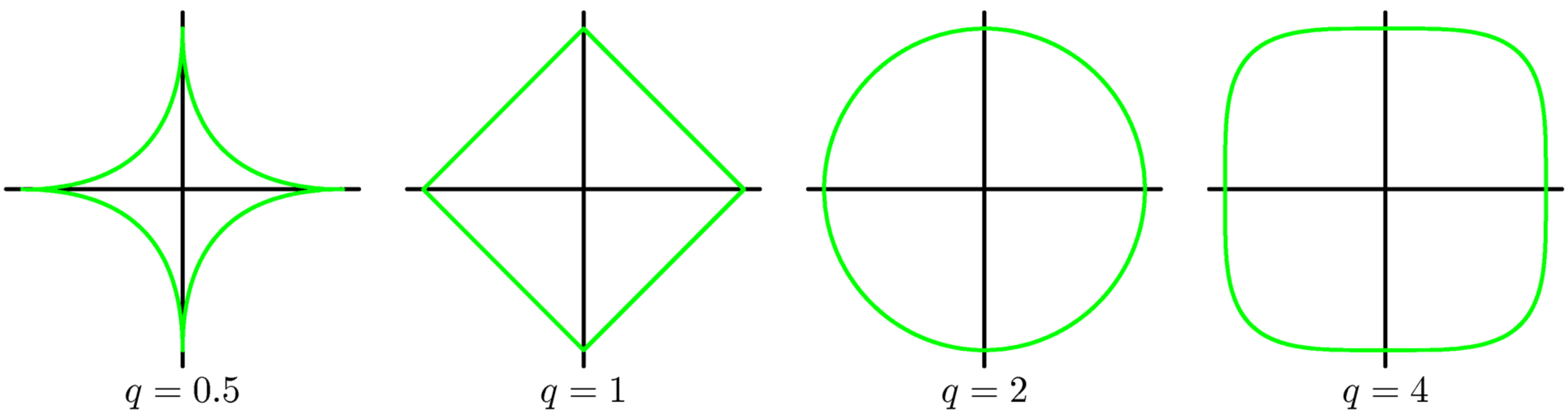
- which is minimized by

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

Regularized Least Squares (2)

- With a more general regularizer, we have

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

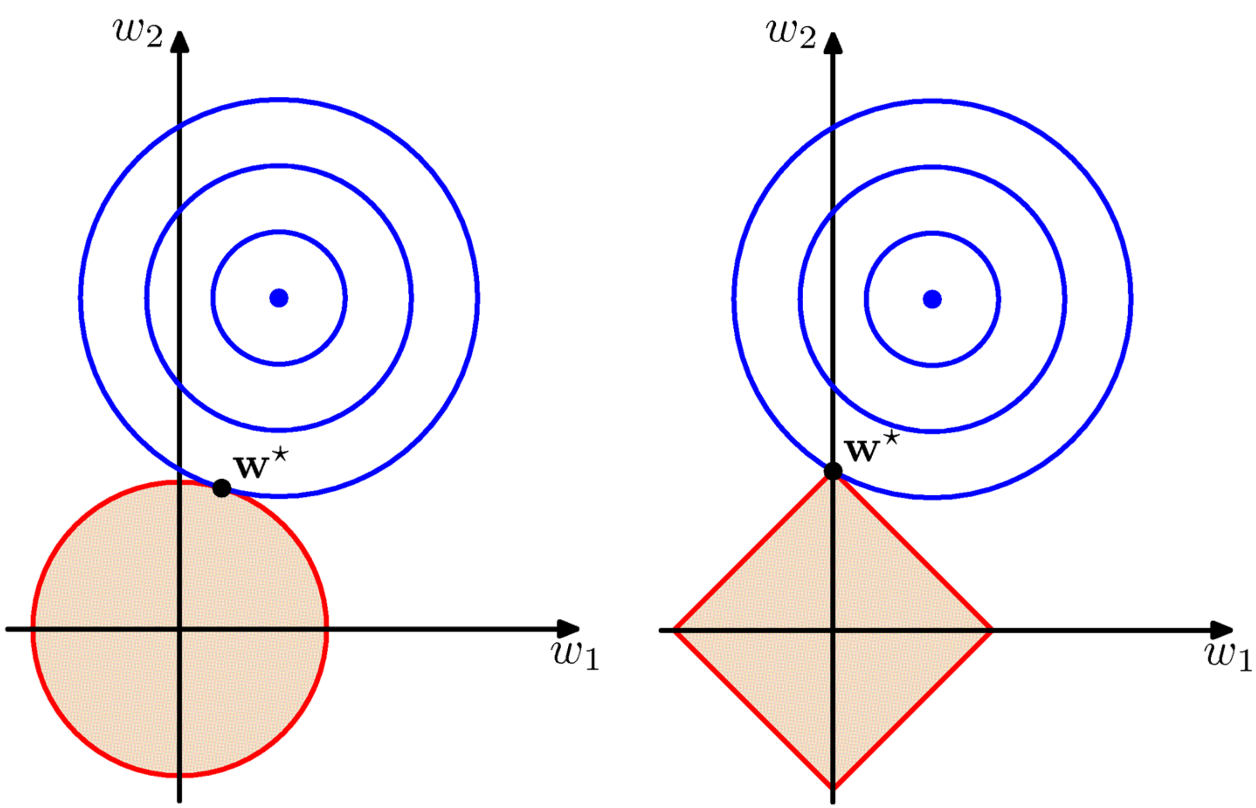


Lasso

Quadratic

Regularized Least Squares (3)

- Lasso tends to generate sparser solutions (majority of the weights shrink to zero) than a quadratic regularizer.



The Bias-Variance Decomposition (1)

- Consider the *expected squared loss*,

$$\mathbb{E}[L] = \int \{y(\mathbf{x}) - h(\mathbf{x})\}^2 p(\mathbf{x})\,\mathrm{d}\mathbf{x} + \iint \{h(\mathbf{x}) - t\}^2 p(\mathbf{x}, t)\,\mathrm{d}\mathbf{x}\,\mathrm{d}t$$

- where

$$h(\mathbf{x}) = \mathbb{E}[t|\mathbf{x}] = \int t p(t|\mathbf{x})\,\mathrm{d}t.$$

- The second term of E[L] corresponds to the noise inherent in the random variable t.

- What about the first term?

The Bias-Variance Decomposition (2)

- Suppose we were given multiple data sets, each of size N. Any particular data set, D, will give a particular function y(x;D). We then have

$$
\begin{aligned}
&\{y(\mathbf{x};\mathcal{D}) - h(\mathbf{x})\}^2 \\
&\quad = \quad \{y(\mathbf{x};\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})] + \mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})] - h(\mathbf{x})\}^2 \\
&\quad = \quad \{y(\mathbf{x};\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})]\}^2 + \{\mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})] - h(\mathbf{x})\}^2 \\
&\qquad\quad +2\{y(\mathbf{x};\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})]\}\{\mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})] - h(\mathbf{x})\}.
\end{aligned}
$$

The Bias-Variance Decomposition (3)

- Taking the expectation over D yields

$$\mathbb{E}_{\mathcal{D}}\left[\{y(\mathbf{x};\mathcal{D}) - h(\mathbf{x})\}^2\right]$$
$$= \underbrace{\{\mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})] - h(\mathbf{x})\}^2}_{(\text{bias})^2} + \underbrace{\mathbb{E}_{\mathcal{D}}\left[\{y(\mathbf{x};\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})]\}^2\right]}_{\text{variance}}.$$

The Bias-Variance Decomposition (4)

- Thus we can write

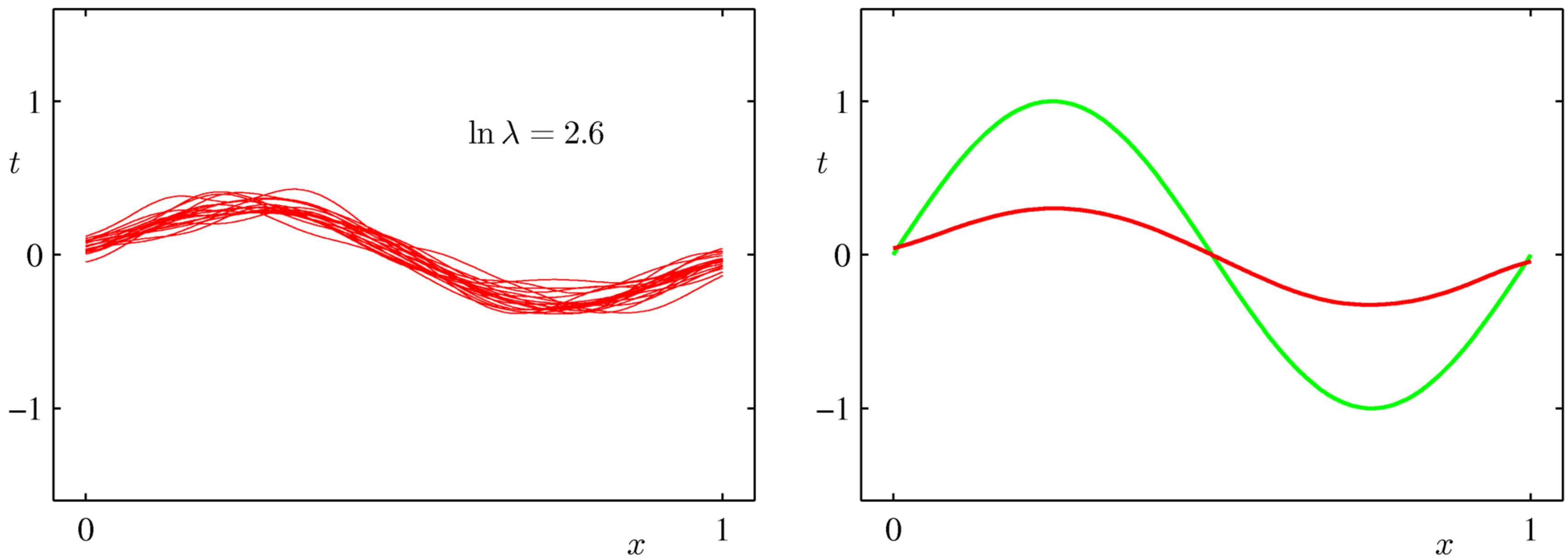
$$\text{expected loss} = (\text{bias})^2 + \text{variance} + \text{noise}$$

- where

$$\begin{aligned}
(\text{bias})^2 &= \int \{\mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})] - h(\mathbf{x})\}^2 p(\mathbf{x})\,\mathrm{d}\mathbf{x} \\
\text{variance} &= \int \mathbb{E}_{\mathcal{D}}\left[\{y(\mathbf{x};\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})]\}^2\right] p(\mathbf{x})\,\mathrm{d}\mathbf{x} \\
\text{noise} &= \iint \{h(\mathbf{x}) - t\}^2 p(\mathbf{x},t)\,\mathrm{d}\mathbf{x}\,\mathrm{d}t
\end{aligned}$$

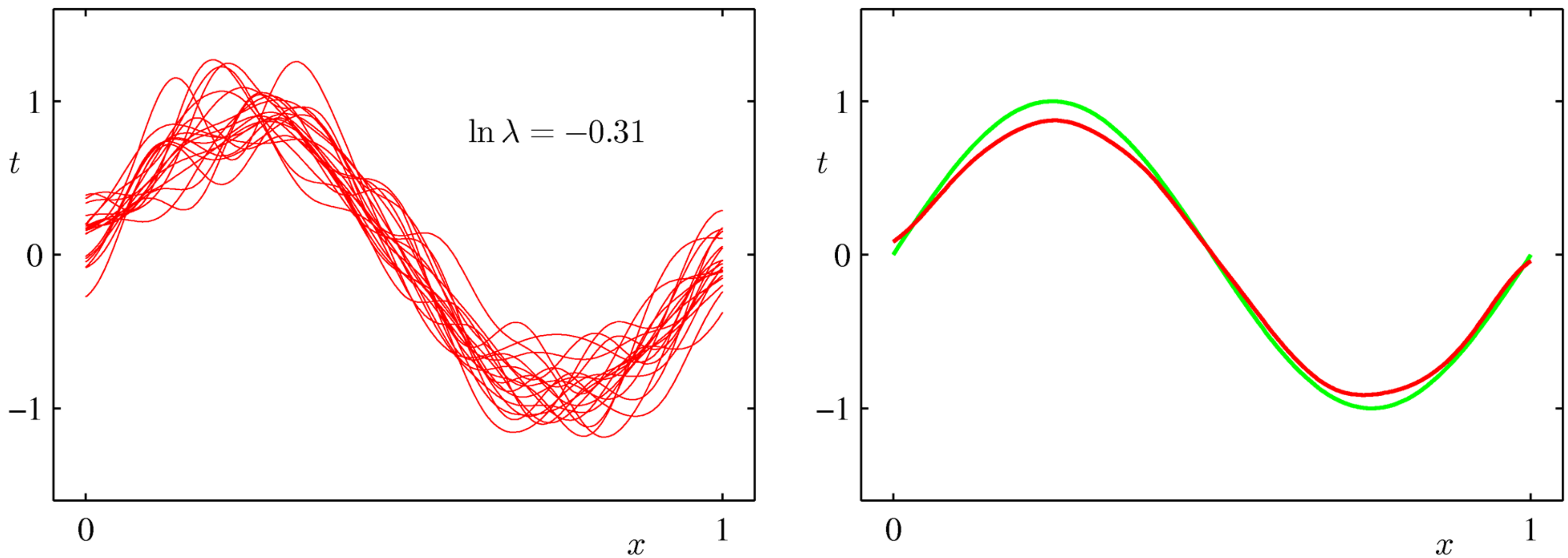
The Bias-Variance Decomposition (5)

- Example: 25 data sets from the sinusoidal, varying the degree of regularization, λ.



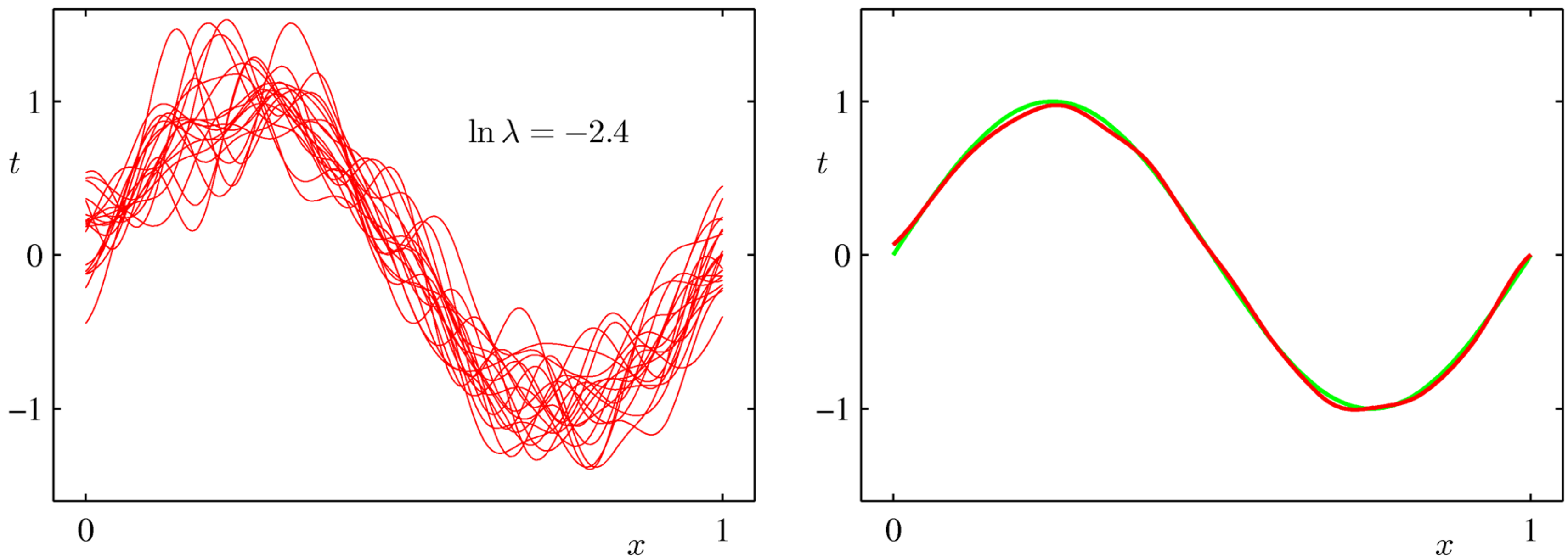
The Bias-Variance Decomposition (6)

- Example: 25 data sets from the sinusoidal, varying the degree of regularization, λ.



The Bias-Variance Decomposition (7)

- Example: 25 data sets from the sinusoidal, varying the degree of regularization, λ.



The Bias-Variance Trade-off

•From these plots, we note that an over-regularized model (large λ) will have a high bias, while an under-regularized model (small λ) will have a high variance.