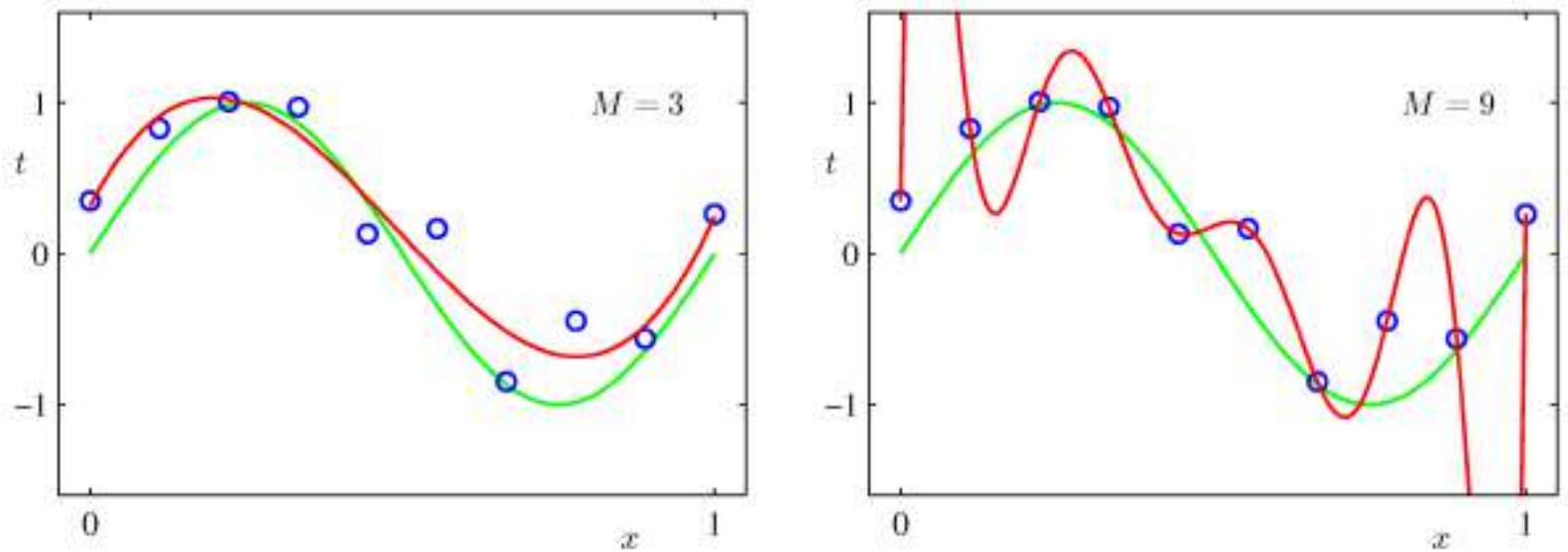
Linear Models for Regression

Stylianos Sygletos

Aston Institute of Photonic Technologies - Aston University

June 15, 2021

Regularized Least Squares



- As model complexity increases, e.g. degree of polynomial or number of basis function then we are likely to have overfitting
- We can control overfitting by adding a regularization term to the error function

$$E(\boldsymbol{w}) = E_D(\boldsymbol{w}) + \lambda E_W(\boldsymbol{w})$$

where λ is the *regularization coefficient*

Simplest regularization form (*weight-decay*)

- The simplest regularization form is the sum-of-squares of the weight vector elements:

$$E_W(\boldsymbol{w}) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}$$

then, the total error function becomes:

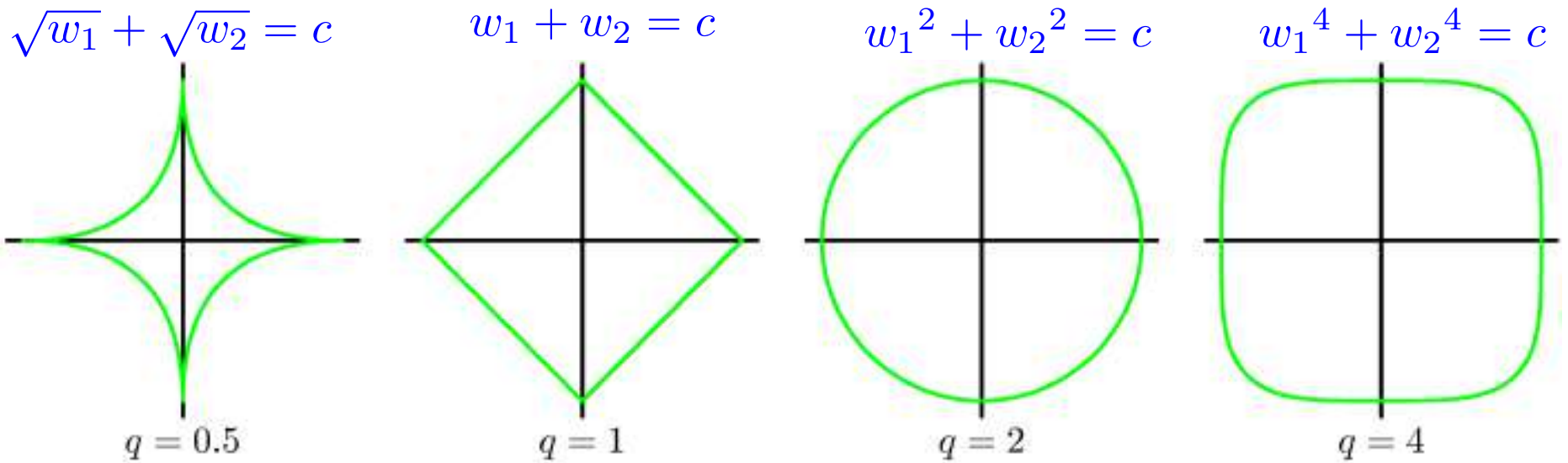
$$\frac{1}{2}\sum_{n=1}^{N}\left\{t_n - \boldsymbol{w}^T\boldsymbol{\phi}(\boldsymbol{x}_n)\right\}^2 + \frac{\lambda}{2}\boldsymbol{w}^T\boldsymbol{w}$$

- The error function remains a *quadratic function* of $\boldsymbol{w}$, so its exact minimizer can be found in analytical form:

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

which is a simple extension of the leas squared solution $\boldsymbol{w} = (\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\mathbf{t}$

A more general regularizer



- In a more general form the regularized error becomes:

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

  - $q = 2$ corresponds to the *quadratic* equalizer
  - $q = 1$ is known as *Lasso*

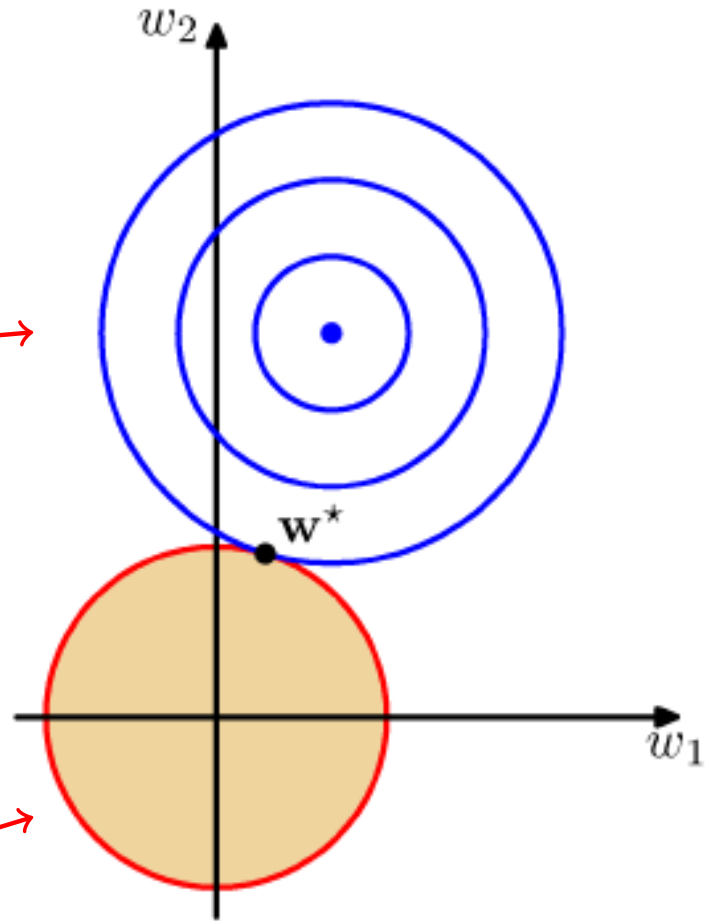
Geometric interpretation of regularizer

- **Unregularized case**
  We are tying to find $\boldsymbol{w}$ that minimizes:

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

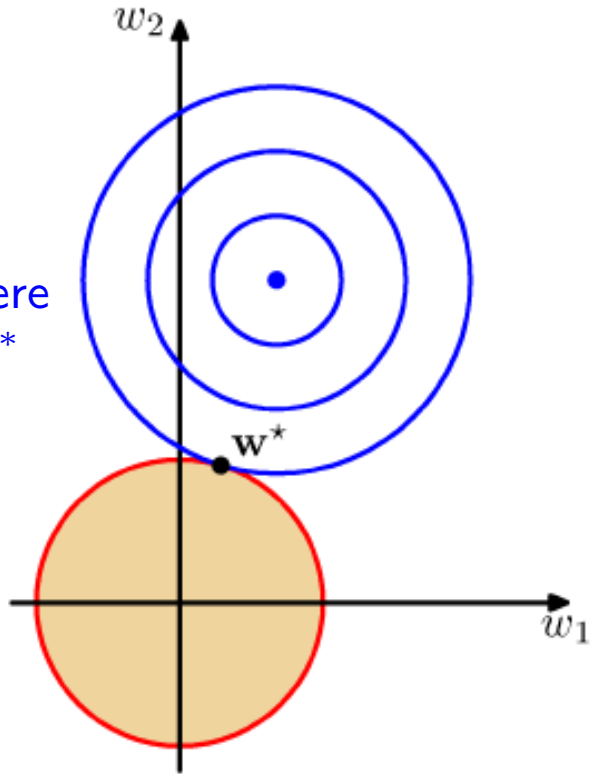
- **Regularized case**
  We choose that value of $\boldsymbol{w}$ subject to the constraint

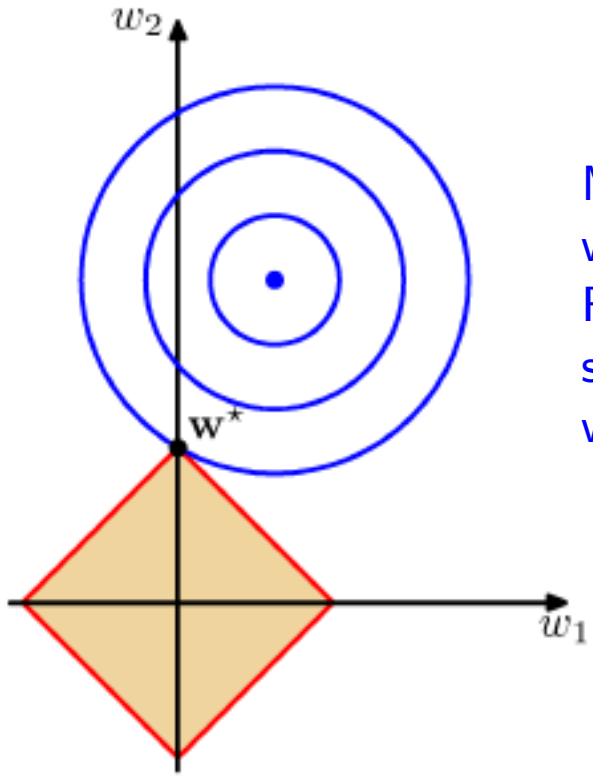
$$\sum_{j=1}^{M}|w_j|^q \leq \eta$$



Sparsity with Lasso constraint

Quadratic solution where $w_1{}^*$ and $w_2{}^*$ are nonzero





Minimization with Lasso Regularizer. A sparse solution with $w_1{}^* = 0$

- With $q = 1$ and λ is sufficiently large, some of the coefficients w_j are driven to zero.
- This leads to a sparse model where the corresponding basis functions play no role.

Regularization: Summary

- Regularization allows complex models to be trained on limited size datasets without severe overfitting
- The problem of detemining the optimal model complexity is shifted to determining a suitable λ value

Multiple Outputs: $\boldsymbol{t} = (t_1, \ldots, t_K),\ K > 1$

- It can be treated as multiple independent regression problems, introducing a different set of basis functions for each component of $\boldsymbol{t}$

- A more common approach is to introduce the same set of basis functions to model the target vector $\boldsymbol{t}$ components:

$$\boldsymbol{y}(\boldsymbol{x}, \boldsymbol{w}) = \boldsymbol{W}^T \boldsymbol{\phi}(\boldsymbol{x})$$

where $\boldsymbol{y}$ is a K-dimensional column vector; $\boldsymbol{W}$ is an $M \times K$ matrix of parameters; $\boldsymbol{\phi}(\boldsymbol{x})$ is an M-dimensional column vector with elements $\phi_j(\boldsymbol{x})$, and $\phi_0(\boldsymbol{x}) = 1$

Solution for Multiple Outputs

- We assume the target vector $\boldsymbol{t}$ has a conditional distribution of the form:

$$p(\boldsymbol{t}|\boldsymbol{x}, \boldsymbol{W}, \beta) = \mathcal{N}(\boldsymbol{t}|\boldsymbol{W}^T\boldsymbol{\phi}(\boldsymbol{x}), \beta^{-1}\boldsymbol{I})$$

- Log-Likelihood

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

where

- $\boldsymbol{T}$ is $N \times K$ matrix combining the $\boldsymbol{t}_1, \ldots, \boldsymbol{t}_N$ observation set
- $\boldsymbol{X}$ combines the input vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$

Solution for Multiple Outputs

- The solution that maximized the log-likelihood is:

$$\boldsymbol{W}_{ML} = (\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\boldsymbol{T}$$

- For each target variable t_k, we have:

$$\boldsymbol{w}_{ML} = (\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\mathbf{t}_k = \boldsymbol{\Phi}^\dagger\mathbf{t}_k$$

Conclusion:
The solution decouples between the target variables, thus we need only to compute a single pseudo-inverse matrix $\boldsymbol{\Phi}^\dagger$, shared by all of the vectors $\boldsymbol{w}_k$

The Bias-Variance Decomposition

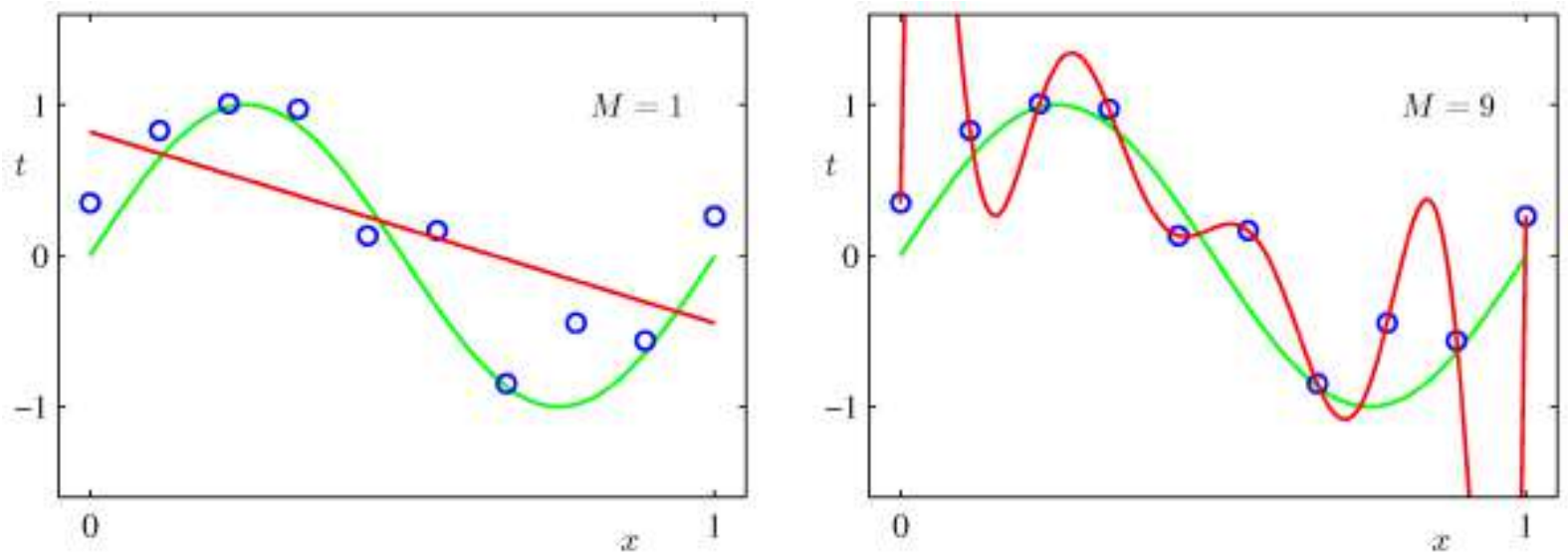
Model Complexity in Linear Regression

- Least squares method can lead to over-fitting if complex models are trained on limited size data sets.
- Limiting the number (M) of the basis function, will reduce the flexibility of the model to capture important data trends.
- Regularization can control overfitting, however, what is the value of the optimum parameter λ?
- Seeking to minimize the regularized error function with respect to both weight $\boldsymbol{w}$ and parameter λ leads to an unregularized solution with $\lambda = 0$

Overfitting is a property of Maximum Likelihood

- Overfitting in an unfortunate property of maximum likelihood. It does not arise in a Bayessian approach.
- Before considering Bayesian view, it is instructive to consider frequentist's viewpoint of model complexity
- This is called *Bias-Variance trade-off*

Bias-Variance Tradeoff in Regression



- Low degree polynomial has high bias (fits poorly) but has low variance with different data sets
- High degree polynomial has low bias (fits well) but has high variance with different data sets

Prediction in Linear Regression

- Regression desision is to choose a specific estimate $y(\boldsymbol{x})$ of the output value t for each input $\boldsymbol{x}$
- In doing so, we incur loss $L(t, y(\boldsymbol{x}))$ function, whereas the average expected loss is :

$$E[L] = \int \int L(t, y(\boldsymbol{x}))p(\boldsymbol{x}, t)d\boldsymbol{x}dt$$

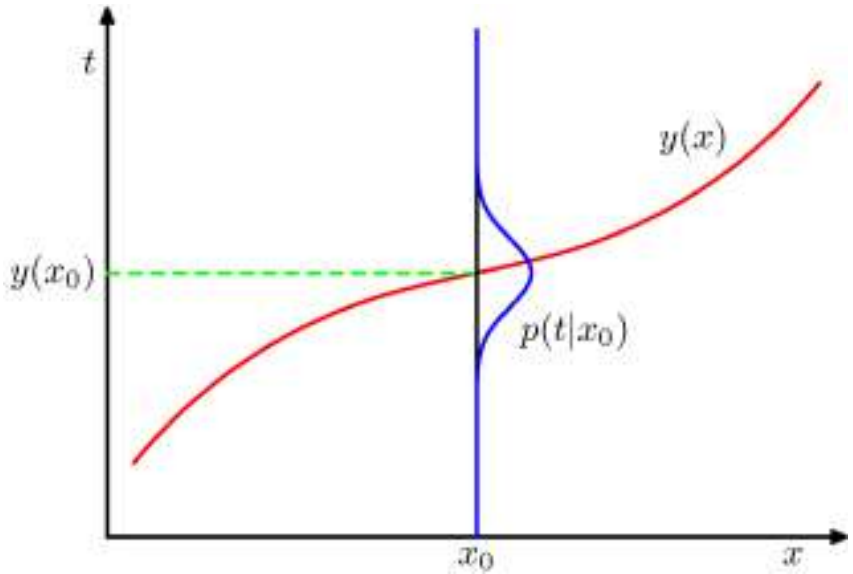
- Squared losss function is a common choice in regression problems:

$$E[L] = \int \int \{y(\boldsymbol{x}) - t\}^2 p(\boldsymbol{x}, t)d\boldsymbol{x}dt$$

- We take the derivative of E w.r.t $y(\boldsymbol{x})$, using calculus of variations:

$$\frac{\delta E[L]}{\delta y(\boldsymbol{x})} = 2 \int (y(\boldsymbol{x}) - t)p(\boldsymbol{x}, t)dt$$

Prediction in Linear Regression



- Setting equal to zero and solving for $y(\boldsymbol{x})$ we get

$$\boxed{y(\boldsymbol{x}) = \int t p(t|\boldsymbol{x}) dt = E_t[t|\boldsymbol{x}] = h(\boldsymbol{x})}$$

- Regression function $y(\boldsymbol{x})$ which minimizes the expected squared loss is given by the mean of the conditional distribution $p(t|\boldsymbol{x})$

Alternative Derivation

- We can show that the optimal prediction is equal to the conditional mean in another way. First we have:

$$\{y(\boldsymbol{x}) - t\}^2 = \{y(\boldsymbol{x}) - E[t|\boldsymbol{x}] + E[t|\boldsymbol{x}] - t\}^2$$

- Substituting into the loss function, we obtain the expression for the loss function as:

$$E[L] = \underbrace{\int \{y(\boldsymbol{x}) - E[t|\boldsymbol{x}]\}^2 p(\boldsymbol{x}) d\boldsymbol{x}}_{\text{Term-A}} + \underbrace{\int \text{var}(t|\boldsymbol{x}) p(\boldsymbol{x}) d\boldsymbol{x}}_{\text{Term-B}}$$

- Term-B : arises from the intrinsic noise on the data and it is independent of $y(\boldsymbol{x})$,
- Term-A : depends on our choice for the function $y(\boldsymbol{x})$ and becomes minimum when: $y(\boldsymbol{x}) = E[t|\boldsymbol{x}] = h(\boldsymbol{x})$

Decomposition into Data Sets

- In practice, calculation of $y(\boldsymbol{x}) = E[t|\boldsymbol{x}]$ cannot be accurate, since our dataset D is of limited size
- For a given data set D, our learning algorithm gives only a prediction function $y(\boldsymbol{x}; D)$, which differs from the ideal $h(\boldsymbol{x})$
- Corresponding squared loss is:

$$\{y(\boldsymbol{x}; D) - h(\boldsymbol{x})\}^2$$

- The performance of our learning algorithm is assessed by taking the average of an ensemble of data sets:

$$\begin{aligned}
&\big\{y(\boldsymbol{x}; D) - h(\boldsymbol{x})\big\}^2 \\
&= \big\{y(\boldsymbol{x}; D) - E_D[y(\boldsymbol{x}; D)] + E_D[y(\boldsymbol{x}; D)] - h(\boldsymbol{x})\big\}^2 \\
&= \big\{y(\boldsymbol{x}; D) - E_D[y(\boldsymbol{x}; D)]\big\}^2 + \big\{E_D[y(\boldsymbol{x}; D)] - h(\boldsymbol{x})\big\}^2 \\
&+ 2\big\{y(\boldsymbol{x}; D) - E_D[y(\boldsymbol{x}; D)]\big\}\big\{E_D[y(\boldsymbol{x}; D)] - h(\boldsymbol{x})\big\}
\end{aligned}$$

Decomposition into Data Sets

- We take the expectation with respect to D:

$$E_D[\{y(\boldsymbol{x}; D) - h(\boldsymbol{x})\}^2]$$
$$= \underbrace{\{E_D[y(\boldsymbol{x}; D)] - h(\boldsymbol{x})\}^2}_{(\text{bias})^2} + \underbrace{E[\{y(\boldsymbol{x}; D) - E_D[y(\boldsymbol{x}; D)\}^2]}_{\text{variance}}$$

- $(\text{bias})^2$: represents the extend to which average prediction from all data sets differs from the desired regression function
- variance : measures the extend to which solutions for individual data sets vary around their average

Bias-Variance in Regression

- For a particular data set the squared loss function $E[L]$ can be written as

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

where

$$(\text{bias})^2 = \int \{E_D[y(\boldsymbol{x}; D)] - h(\boldsymbol{x})\}^2 p(\boldsymbol{x}) d\boldsymbol{x}$$

$$\text{variance} = \int E_D\left[\{y(\boldsymbol{x}; D) - E_D[y(\boldsymbol{x}; D)]\}^2\right] p(\boldsymbol{x}) d\boldsymbol{x}$$

$$\text{noise} = \int \{h(\boldsymbol{x}) - t\}^2 d\boldsymbol{x} dt$$

- There is a trade-off between bias and variance
  - Very flexible models have low bias and high variance
  - Rigid models have high bias and low variance
  - Optimal models have the best balance

Dependence of Bias-Variance on Model Complexity

Problem Definition:

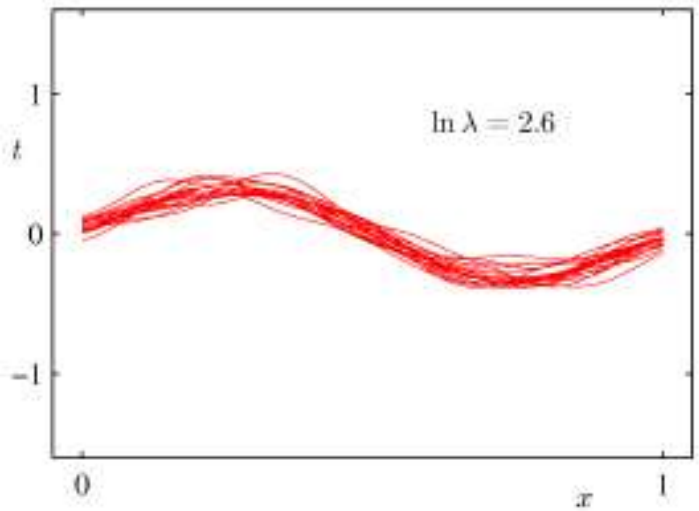
- $h(x) = \sin(2\pi x)$
- $L = 100$ data sets
- each with $N = 25$ points
- regularization parameter λ

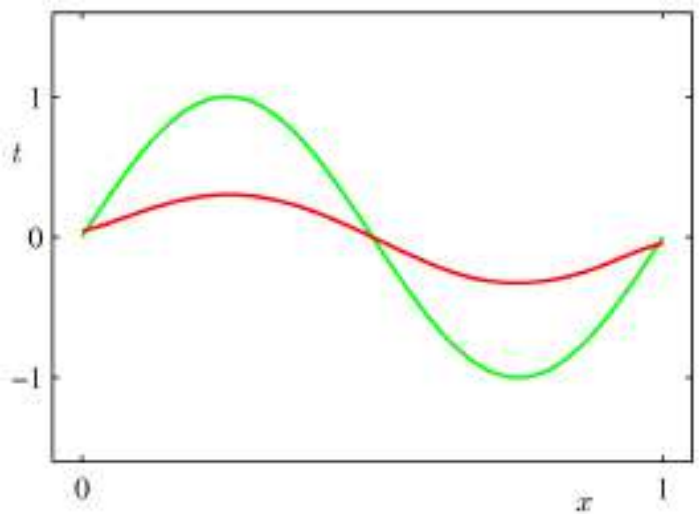
Total Error Function:

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

where $\boldsymbol{\phi}$ is the vector of the basis function

Low Variance, High Bias



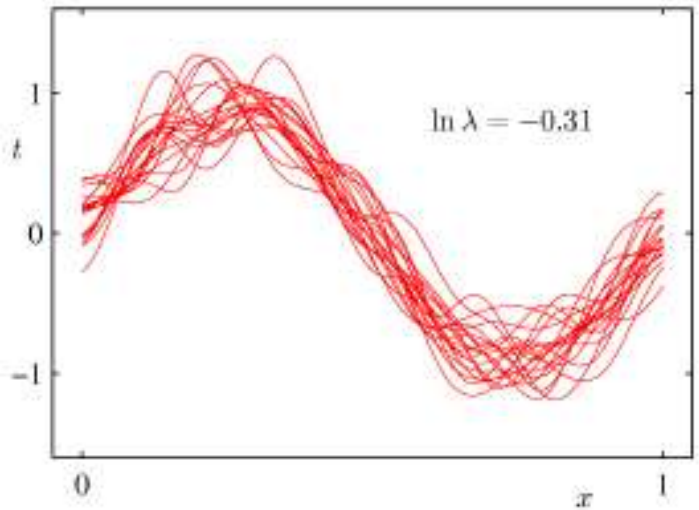


High λ

Dependence of Bias-Variance on Model Complexity

Problem Definition:

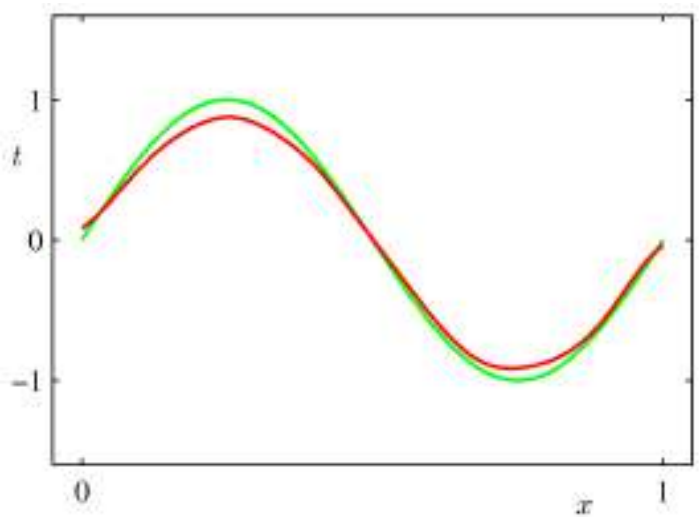
- $h(x) = \sin(2\pi x)$
- $L = 100$ data sets
- each with $N = 25$ points
- regularization parameter λ



Total Error Function:

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

where $\boldsymbol{\phi}$ is the vector of the basis function



Dependence of Bias-Variance on Model Complexity

Problem Definition:

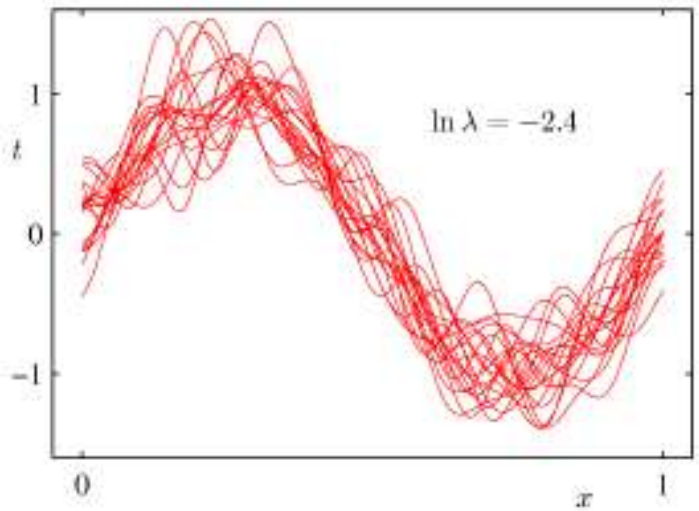
- $h(x) = \sin(2\pi x)$
- $L = 100$ data sets
- each with $N = 25$ points
- regularization parameter λ

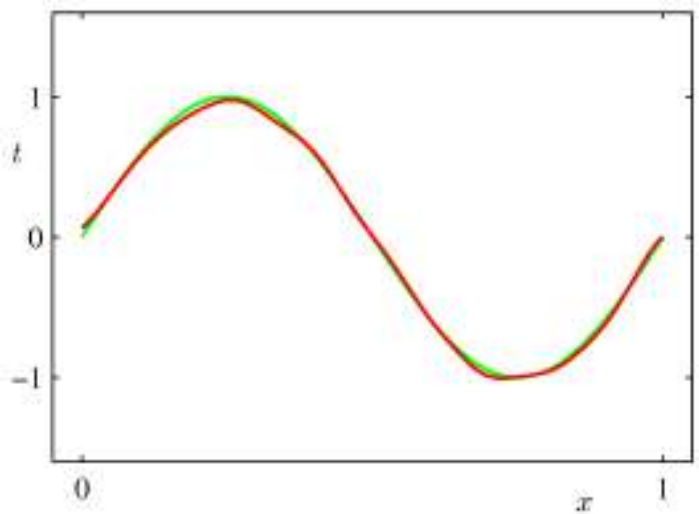
Total Error Function:

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

where $\boldsymbol{\phi}$ is the vector of the basis function

High Variance, Low Bias





Low λ

Determining optimal λ

- Average Prediction

$$\bar{y}(x) = \frac{1}{L}\sum_{l=1}^{L} y^{(l)}(x)$$

- Squared Bias

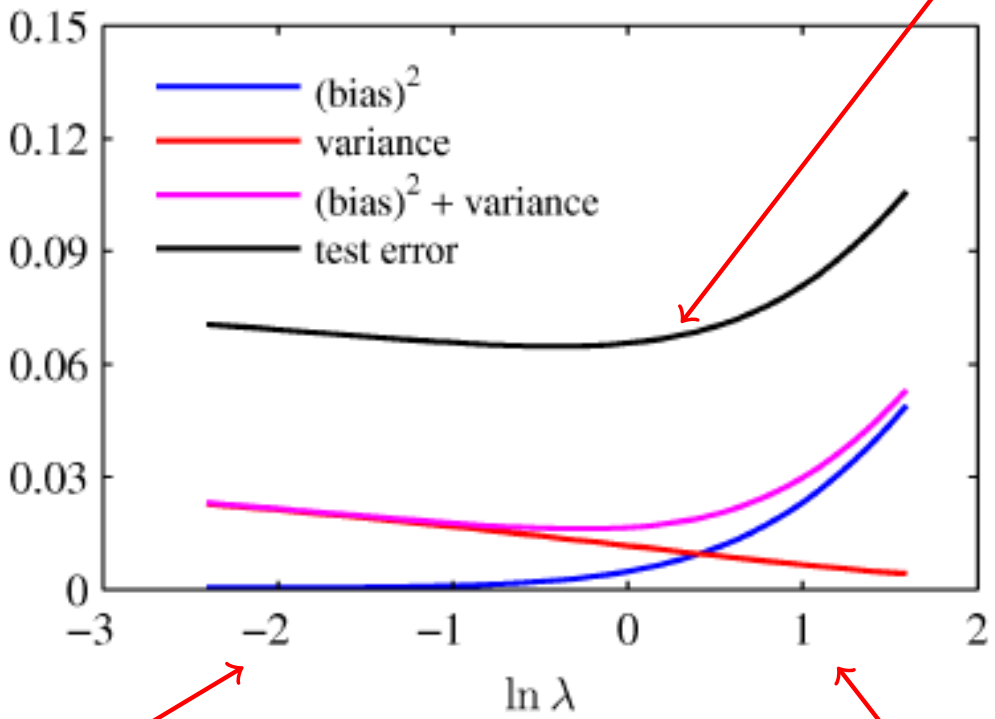
$$(\text{bias})^2 = \frac{1}{N}\sum_{n=1}^{N}\{\bar{y}(x_n) - h(x_n)\}^2$$

- Variance

$$\text{variance} = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{L}\sum_{l=1}^{L}\{y^{(l)}(x_n) - \bar{y}(x_n)\}^2$$

Squared Bias and Variance versus λ

Test error minimum occurs close to minimum of bias^2 and variance



Small values of λ allow model to become finely tuned to noise leading to large variance

Large values of λ pull weight parameters to zero leading to large bias

Bias-Variance vs Bayesian Approach

- Bias-Variance decomposition provides insight into model complexity from a frequentist perspective
- However it is of limited practical value since it is based on averages with respect to ensembles of data set
  - in practice there is only a single observed data set
  - if there are many training sets, better combine them into a single large training set to reduce overfitting
- Bayesian approach gives useful insights into overfitting and it is also practical