Bias and Variance of the Estimator

PRML 3.2

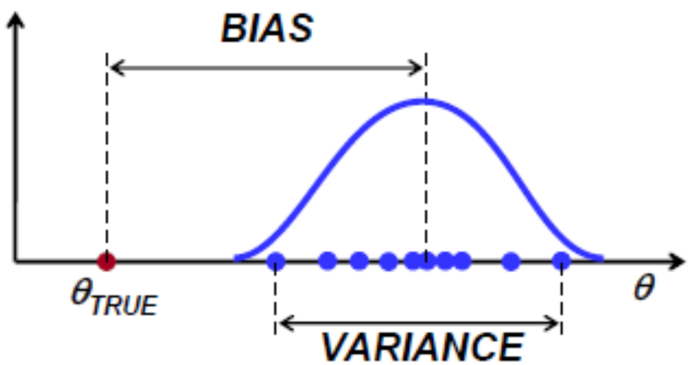
Ethem Chp. 4

- In previous lectures we showed how to build classifiers when the underlying densities are known
  - Bayesian Decision Theory introduced the general formulation

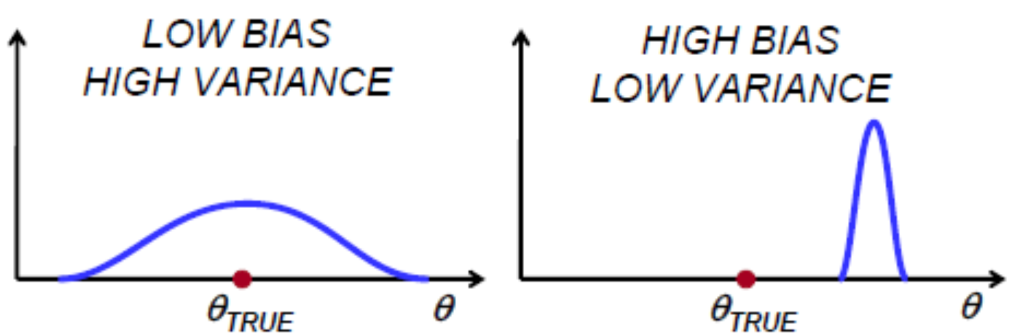
- In most situations, however, the true distributions are unknown and must be estimated from data.
  - **Parameter Estimation** (we saw the Maximum Likelihood Method)
    - Assume a particular form for the density (e.g. Gaussian), so only the parameters (e.g., mean and variance) need to be estimated
    - Maximum Likelihood
    - Bayesian Estimation
  - **Non-parametric Density Estimation** (not covered)
    - Assume NO knowledge about the density
    - Kernel Density Estimation
    - Nearest Neighbor Rule

Bias and variance (1)

- **How good are these estimates? Two measures of "goodness" are used for statistical estimates**
  - **BIAS**: how close is the estimate to the true value?
  - **VARIANCE**: how much does the estimate change for different runs (e.g. different datasets)?



- **The bias-variance tradeoff**
  - In most cases, you can only decrease one of them at the expense of the other



How Good is an Estimator

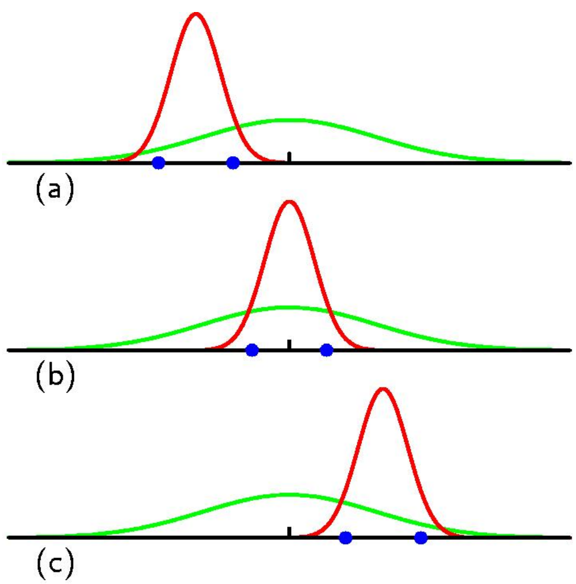
- Assume our dataset X is sampled from a population specified up to the parameter θ; how good is an estimator d(X) as an estimate for θ?

- Notice that **the estimate depends on sample set X**
- If we take an expectation of the difference over different datasets X, $E_X[(d(X)-\theta)^2]$, and expand using the simpler notation of E[d]= E[d(X)], we get:

$$E[(d(X)-\theta)^2] = E[(d(X)-E[d])^2] \quad + \quad (E[d]-\theta)^2$$

variance ... bias sq.

of the estimator



Using a simpler notation (dropping the dependence on X from the notation – but knowing it exists):

$$E[(d-\theta)^2] \quad = E[(d-E[d])^2] \quad + \quad (E[d]-\theta)^2$$

variance ... bias sq.

of the estimator

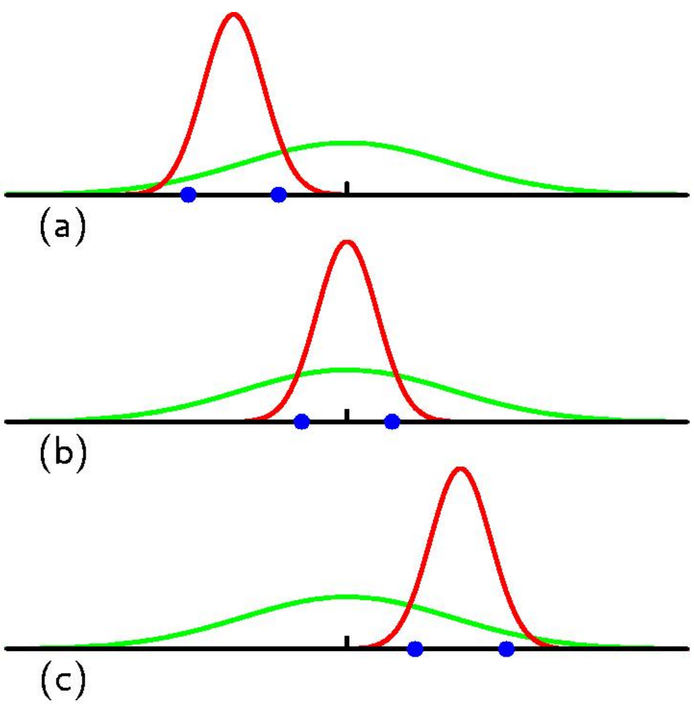
Properties of μ_{ML} *and* σ^2_{ML}

$\mathbb{E}[\mu_{\mathrm{ML}}] = \mu \longrightarrow$ μ_{ML} is an unbiased estimator

$\mathbb{E}[\sigma^2_{\mathrm{ML}}] = \left(\frac{N-1}{N}\right)\sigma^2 \longrightarrow$ σ_{ML} is biased

Use instead:

$$\begin{aligned} \widetilde{\sigma}^2 &= \frac{N}{N-1}\sigma^2_{\mathrm{ML}} \\ &= \frac{1}{N-1}\sum_{n=1}^{N}(x_n - \mu_{\mathrm{ML}})^2 \end{aligned}$$



Bias Variance Decomposition

The Bias-Variance Decomposition (1)

- Recall the *expected squared loss*,

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

Lets denote, for simplicity:

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

- We said that the second term 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*).
- Consider the error in the estimation:

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

The Bias-Variance Decomposition (3)

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

- 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**

- **where**

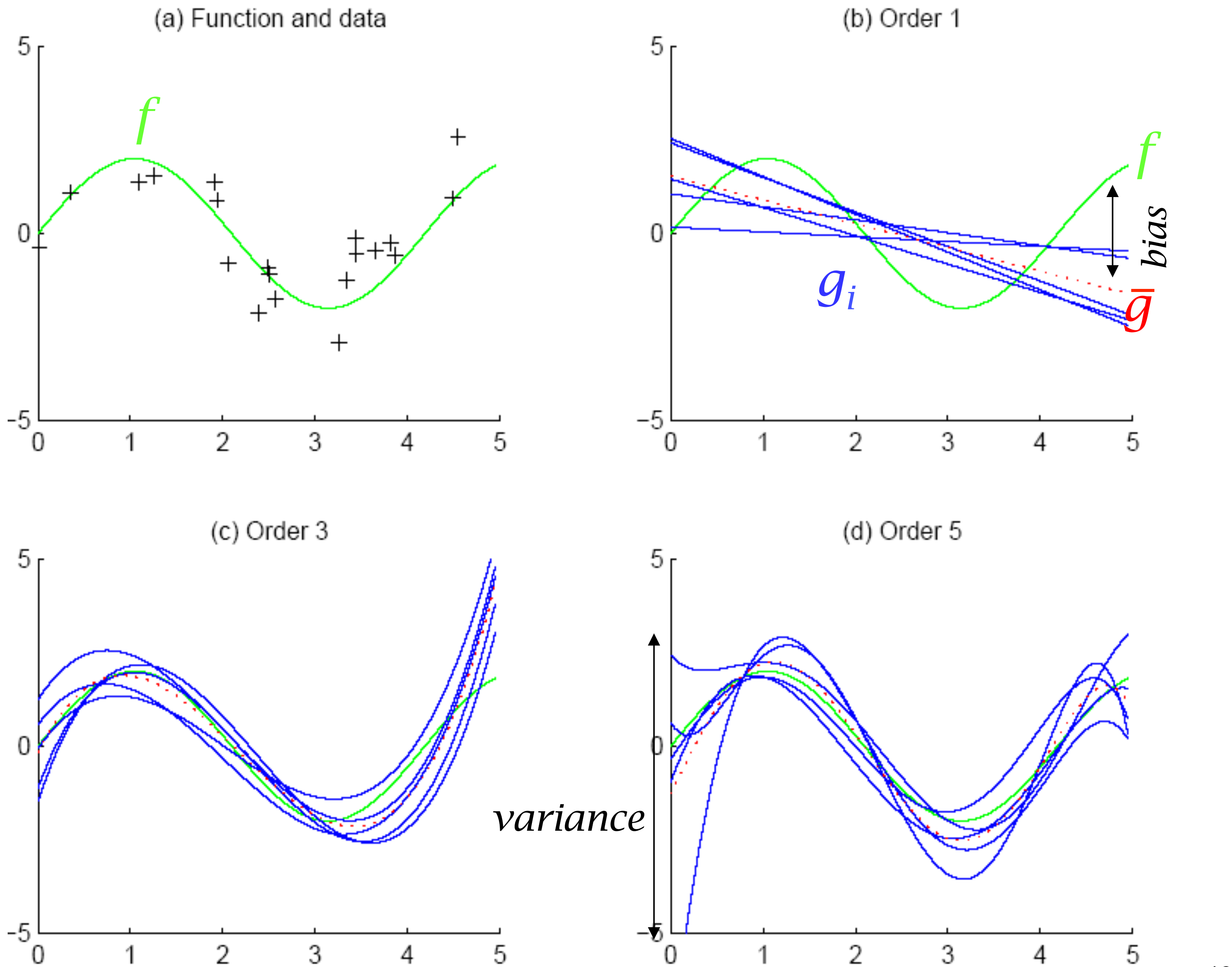
$$\text{expected loss} = (\text{bias})^2 + \text{variance} + \text{noise}$$

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

- **Bias** measures how much the prediction (averaged over all data sets) differs from the desired regression function.

- **Variance** measures how much the predictions for individual data sets vary around their average.

- There is a trade-off between bias and variance
- As we increase **model complexity**,
- bias decreases (a better fit to data) and
- variance increases (fit varies more with data)


(a) Function and data
f
(b) Order 1
f
bias
g_i
$\bar{g}$
(c) Order 3
(d) Order 5
variance

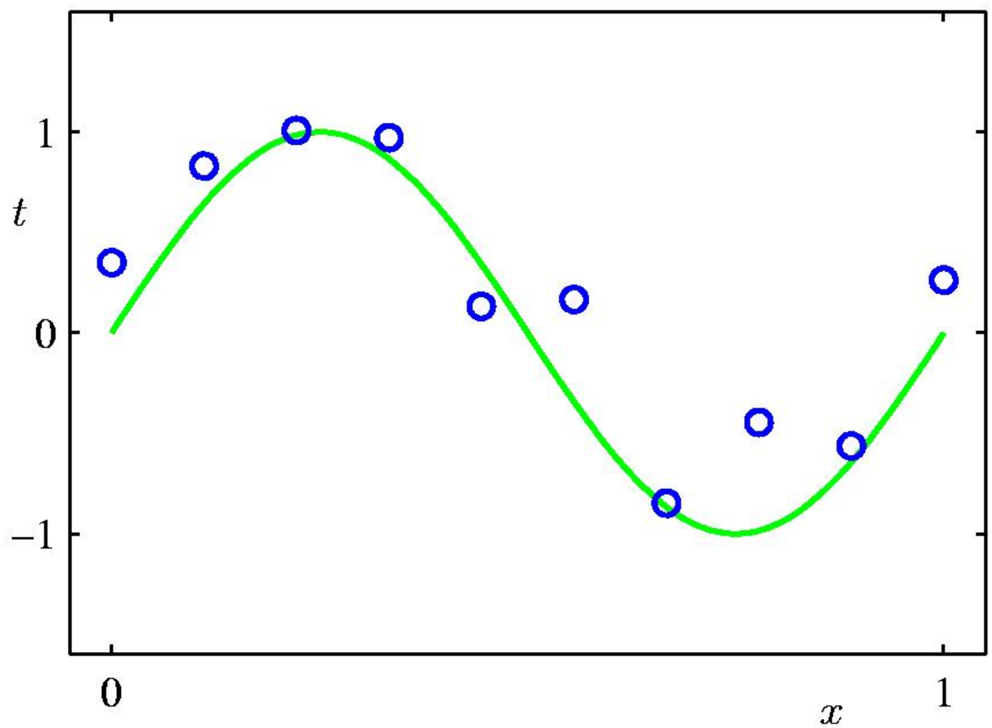
Model Selection Procedures

1. **Regularization** (Breiman 1998): Penalize the augmented error:
    1. error on data + λ.model complexity
    1. If λ is too large, we risk introducing bias
    2. Use cross validation to optimize for λ
2. **Structural Risk Minimization** (Vapnik 1995):
    1. Use a set of models ordered in terms of their complexities
        1. Number of free parameters
        2. VC dimension,…
    2. Find the best model w.r.t empirical error and model complexity.
3. **Minimum Description Length Principle**
4. **Bayesian Model Selection:** If we have some prior knowledge about the approximating function, it can be incorporated into the Bayesian approach in the form of p(model).

Reminder: Introduction to Overfitting
PRML 1.1

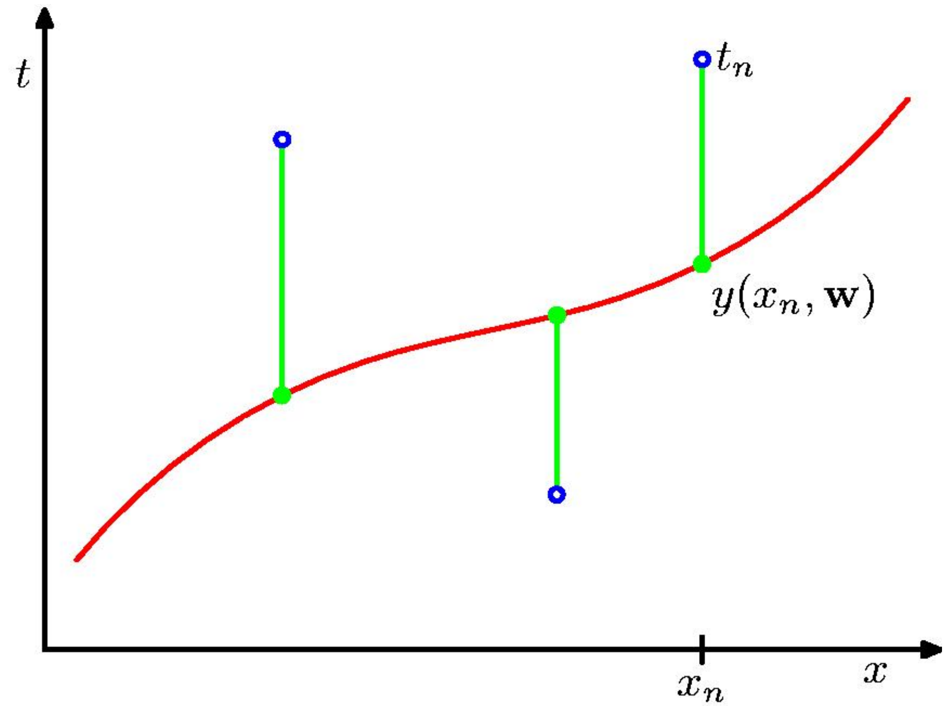
Concepts: Polynomial curve fitting,
overfitting, regularization,
training set size vs model complexity

Polynomial Curve Fitting



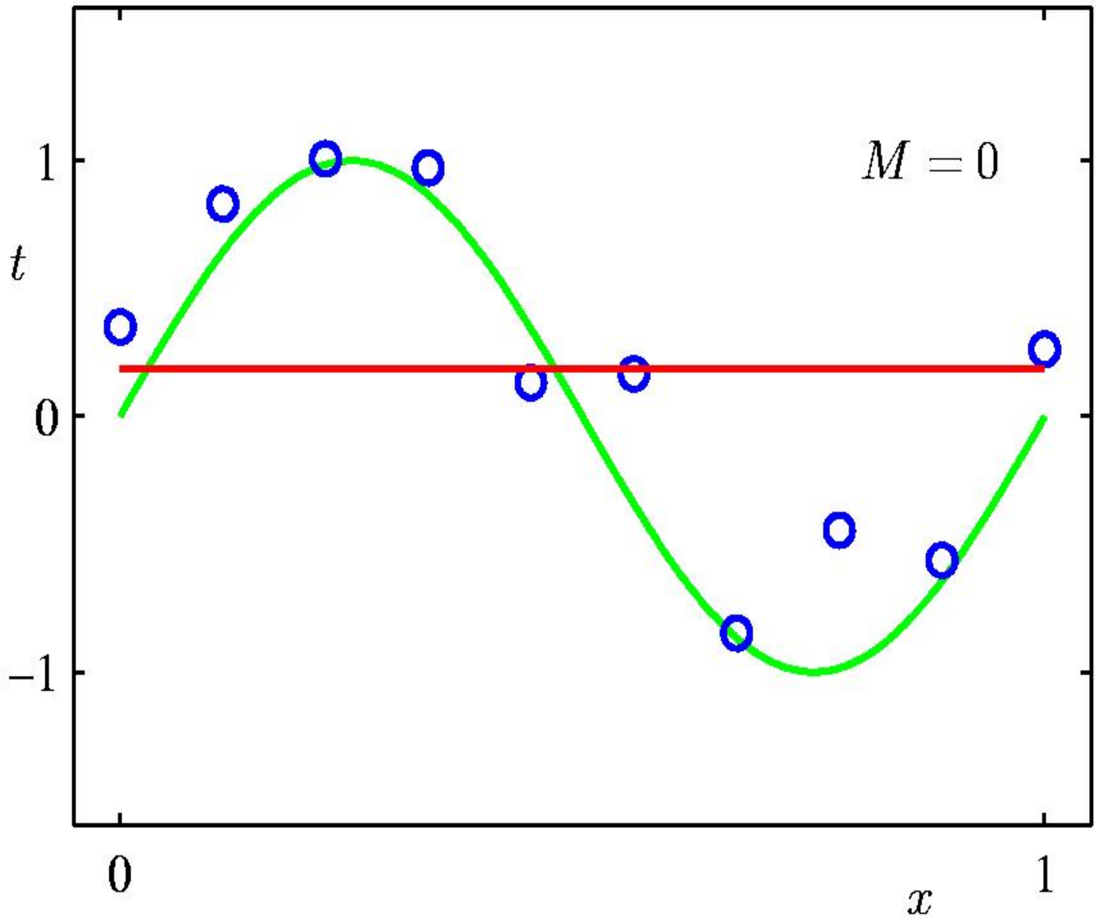
$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

Sum-of-Squares Error Function

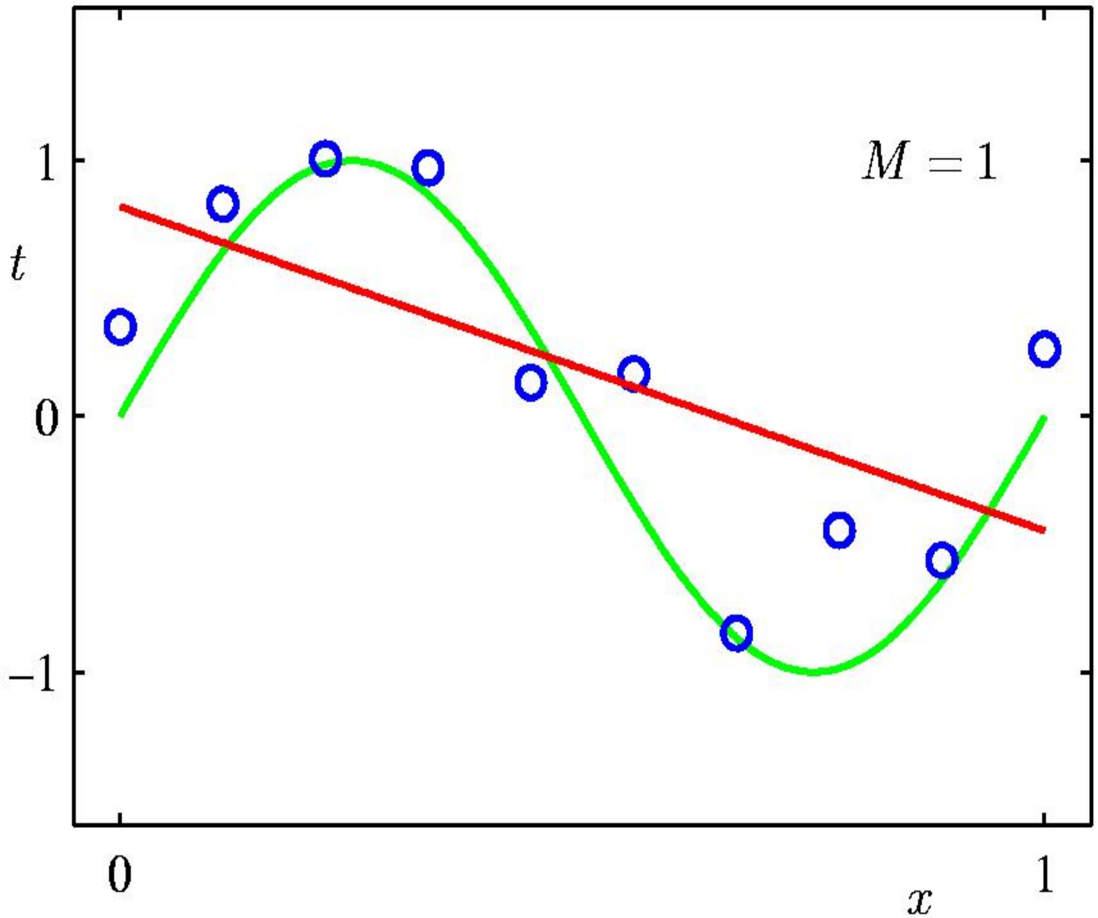


$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\}^2$$

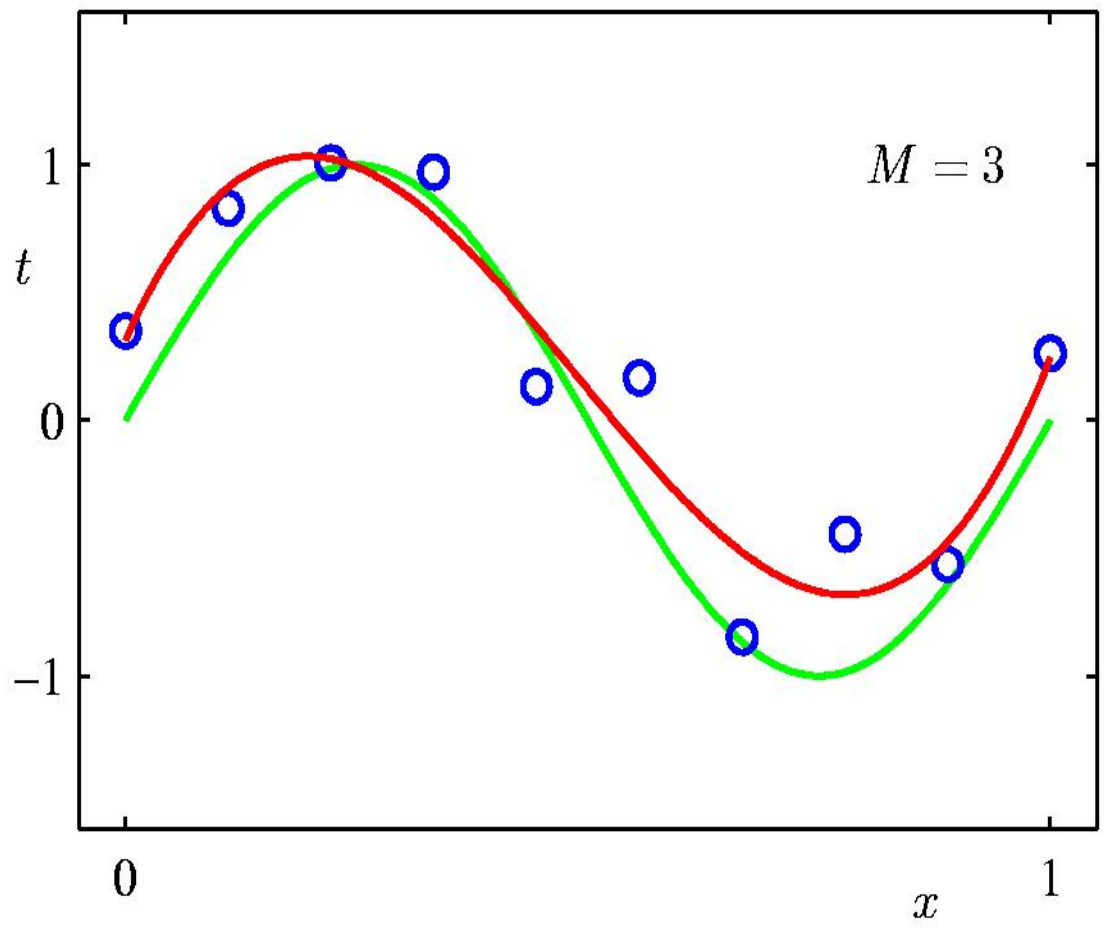
0^{th} Order Polynomial



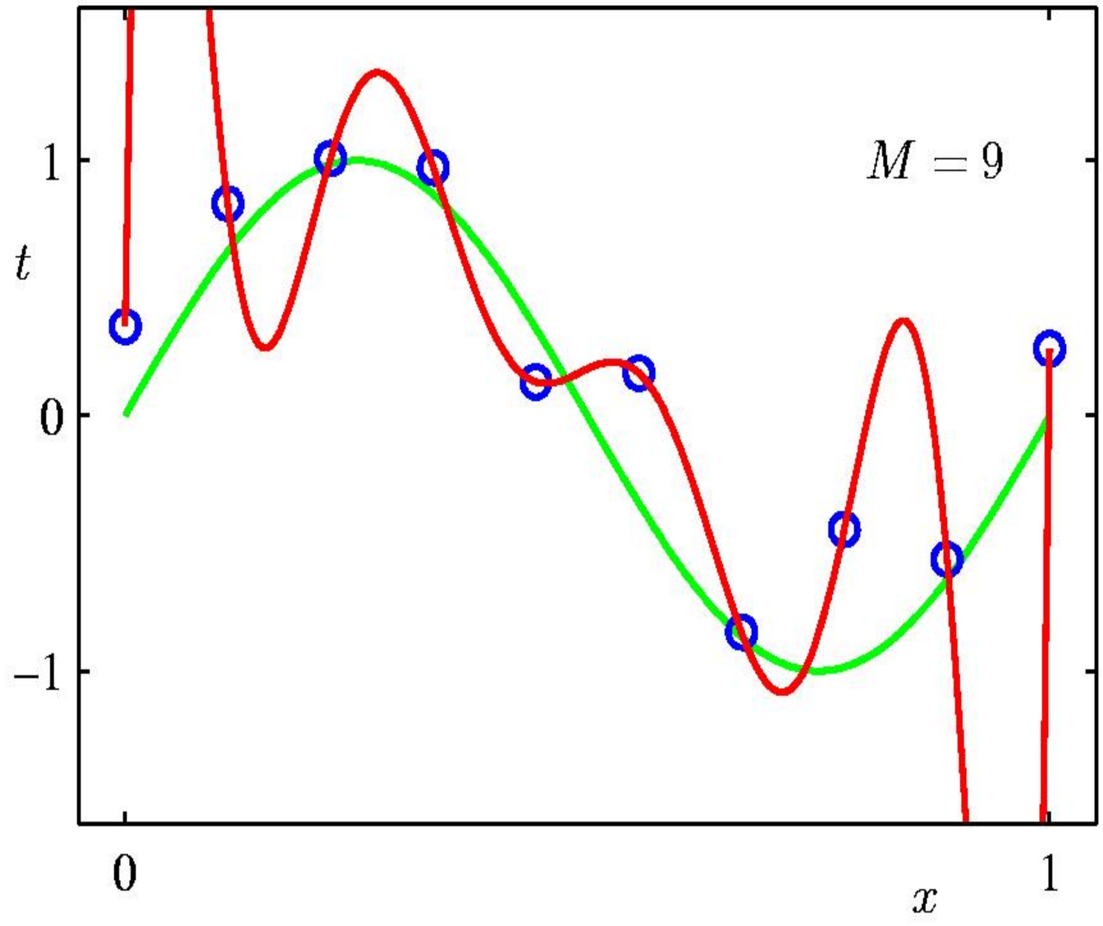
1^{st} Order Polynomial



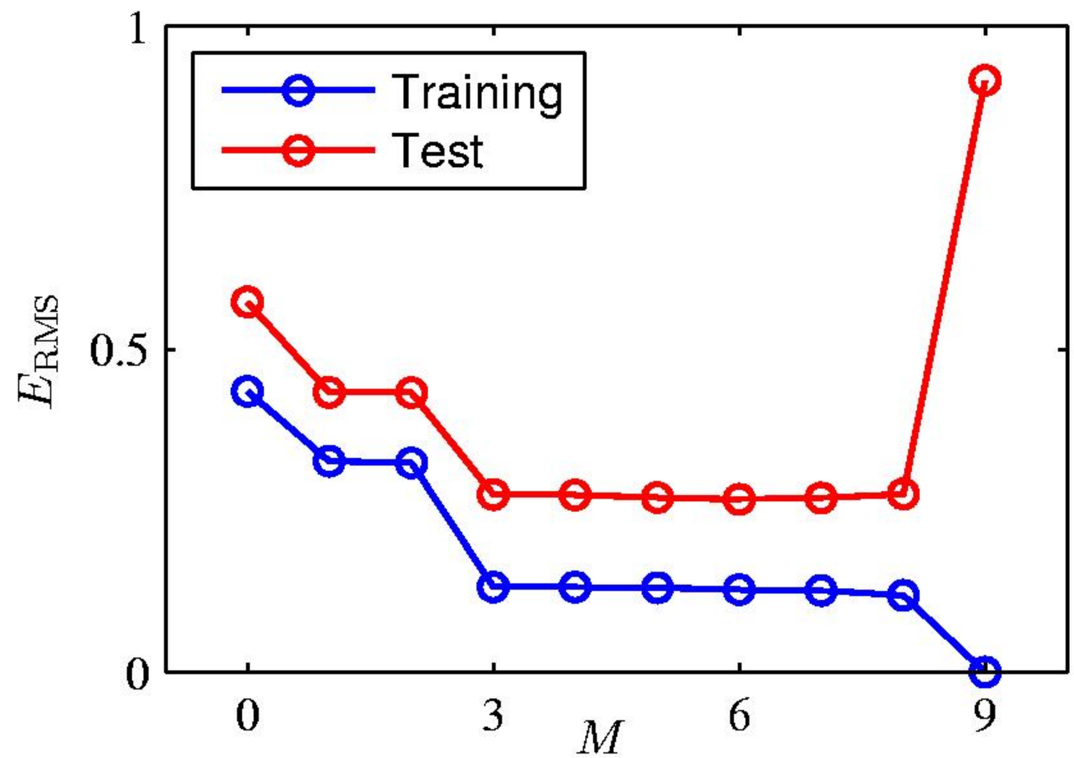
3rd Order Polynomial



9^{th} 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

Regularization

One solution to control complexity is to penalize complex models -> regularization.

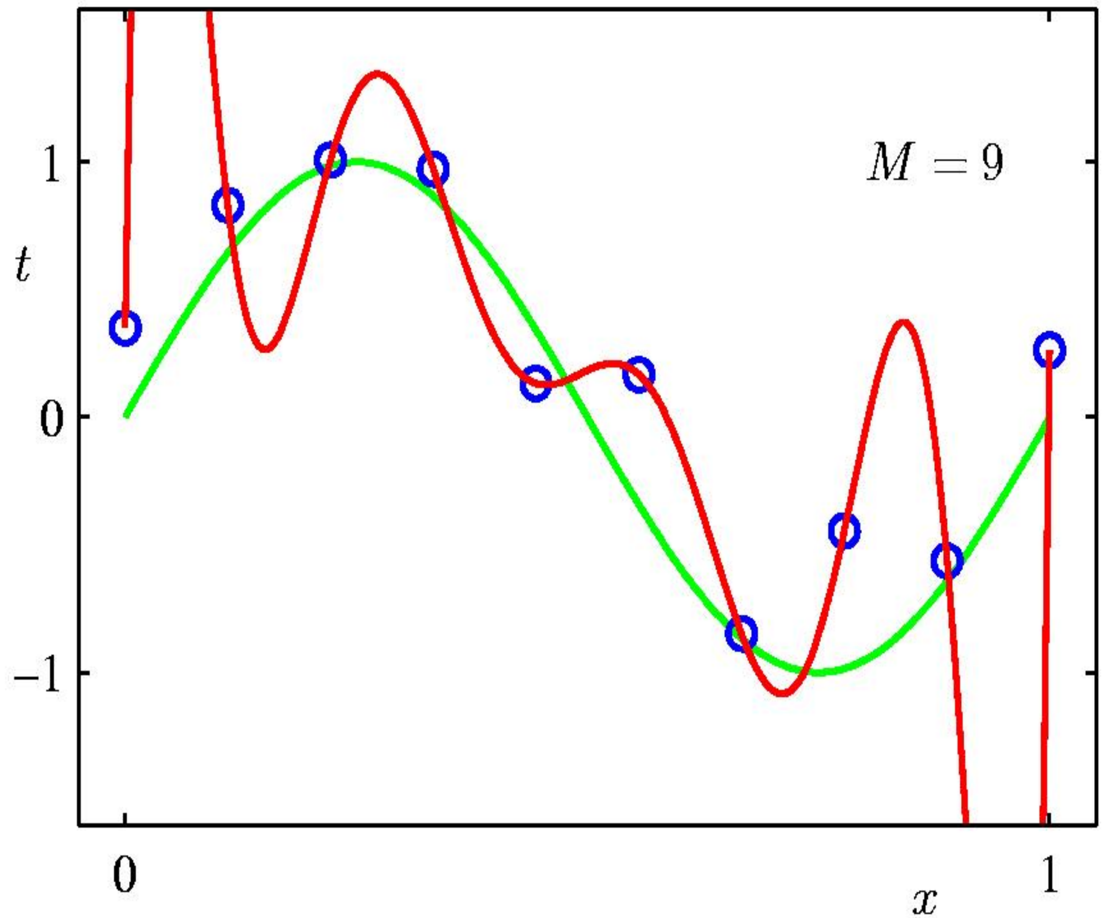
Regularization

- Use complex models, but 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 on 9^{th} Order Polynomial

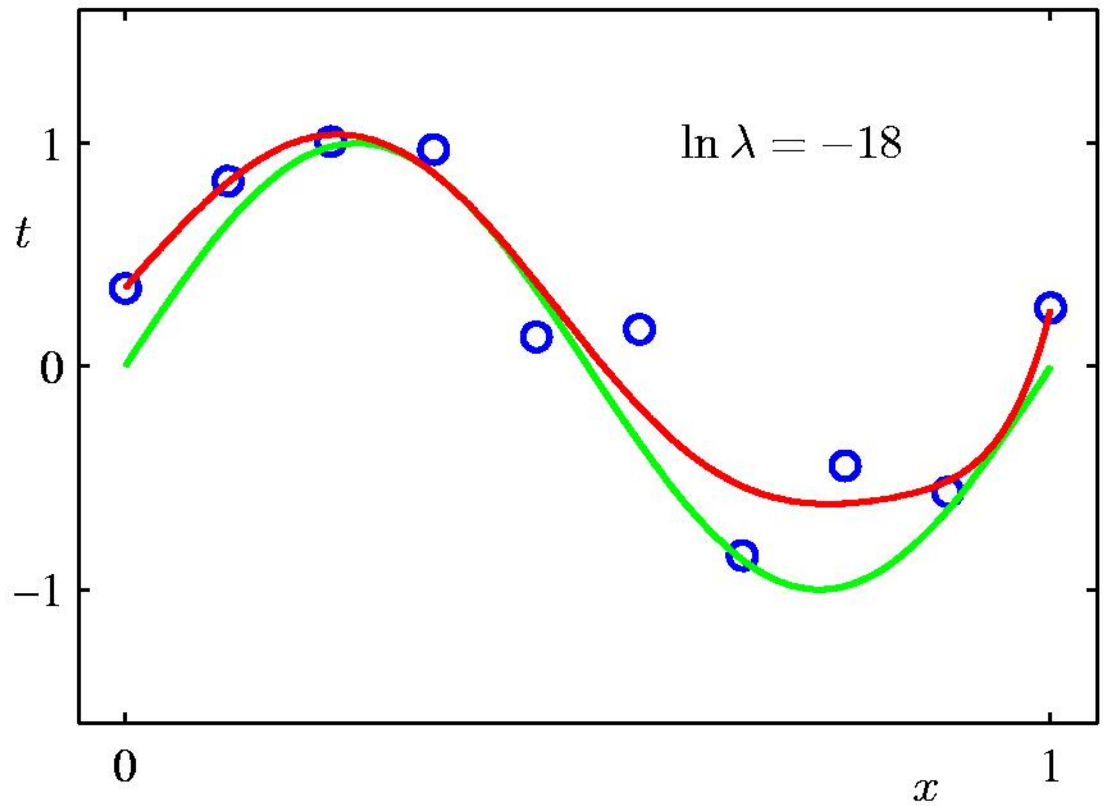
$\ln \lambda = -\inf$



Too small λ – no regularization effect

Regularization on 9th degree polynomial:

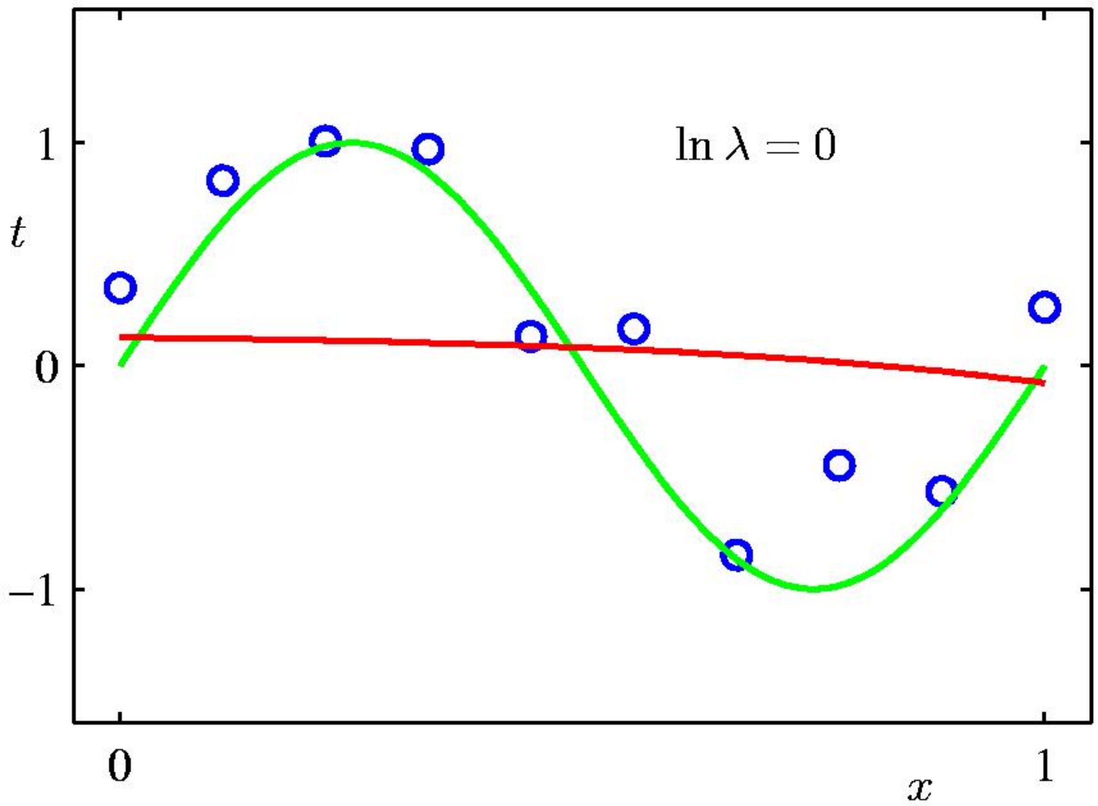
$$\ln \lambda = -18$$



Right λ –good fit

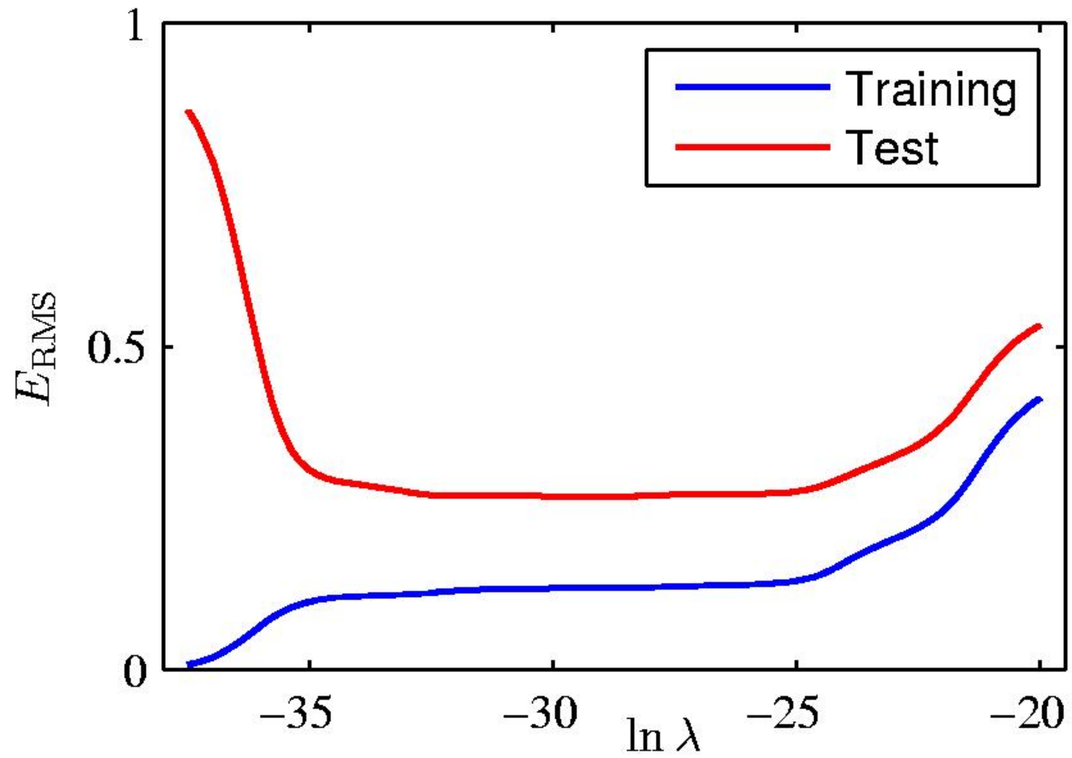
Regularization:

$$\ln \lambda = 0$$



Large λ –regularization dominates

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

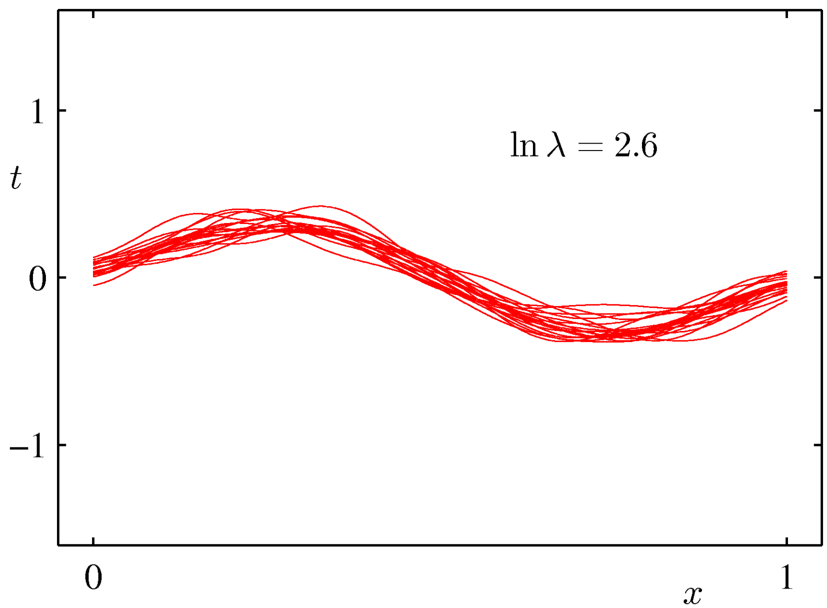


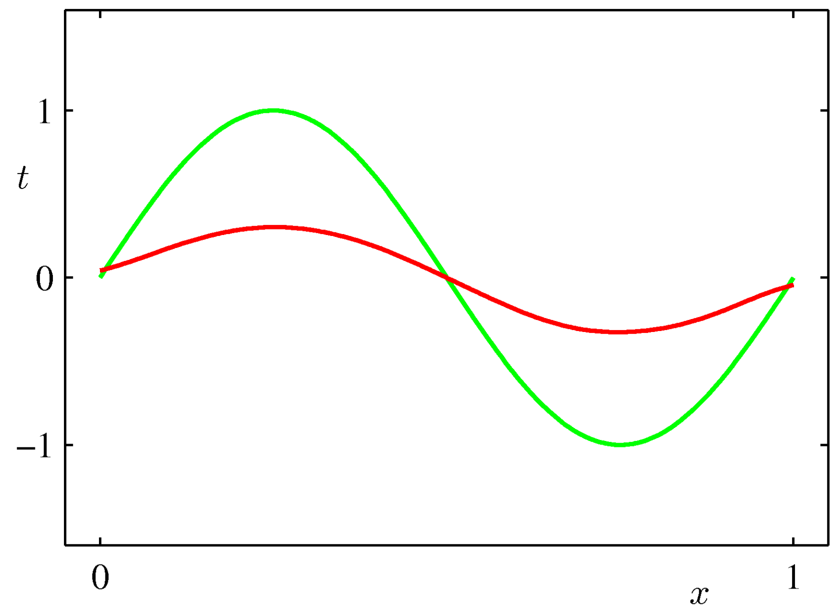
Polynomial Coefficients

	$\ln \lambda = -\infty$	$\ln \lambda = -18$	$\ln \lambda = 0$
$w_0^\star$	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 Bias-Variance Decomposition (5)

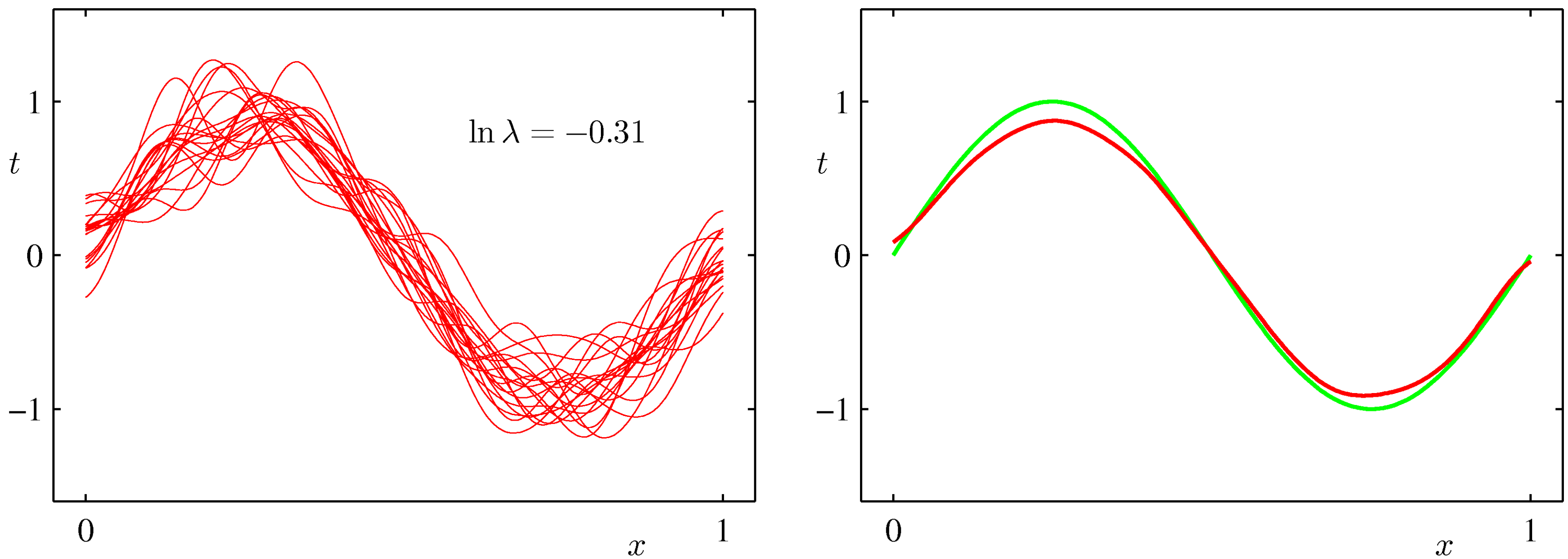
- Example: 100 data sets, each with 25 data points from the sinusoidal h(x) = sin(2px), varying the degree of regularization, λ.





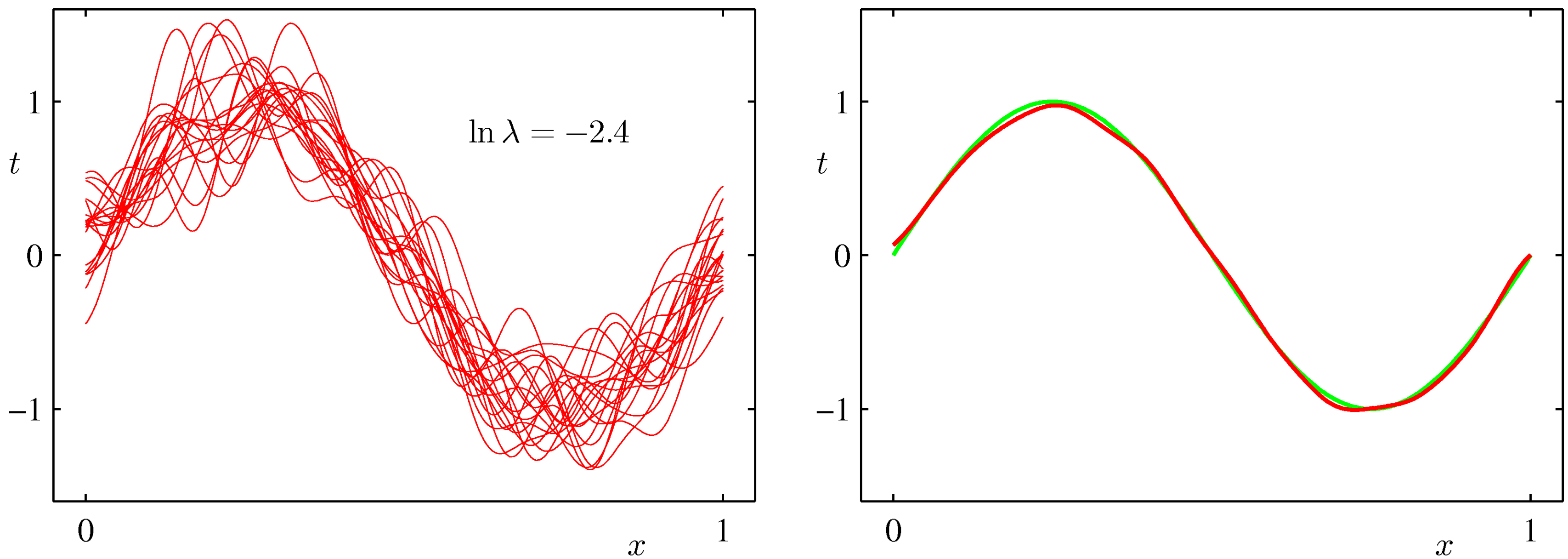
The Bias-Variance Decomposition (6)

- Regularization constant $\lambda = \exp\{-0.31\}$.



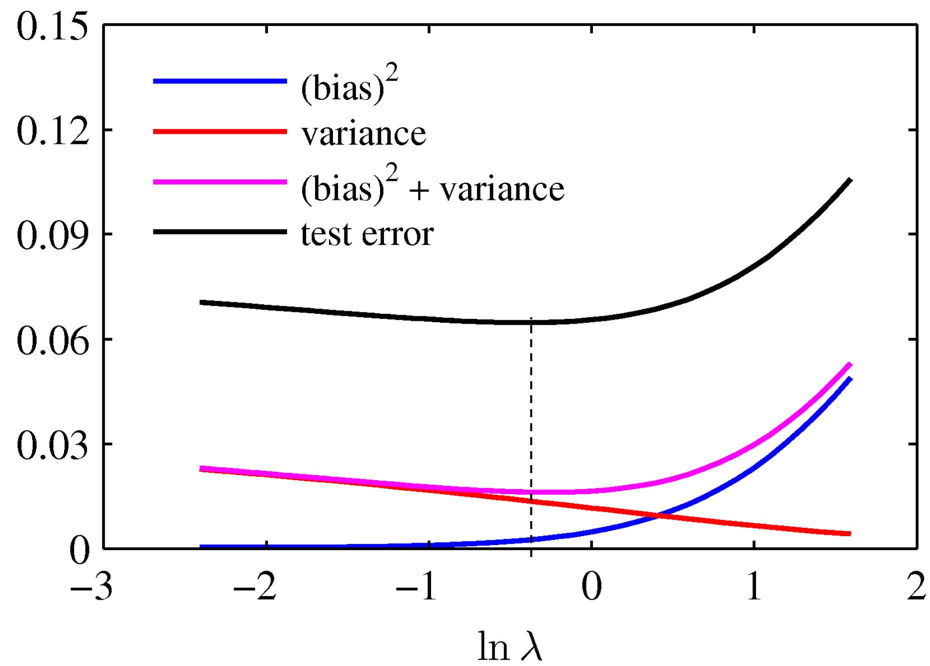
The Bias-Variance Decomposition (7)

- **Regularization constant λ = exp{-2.4}.**

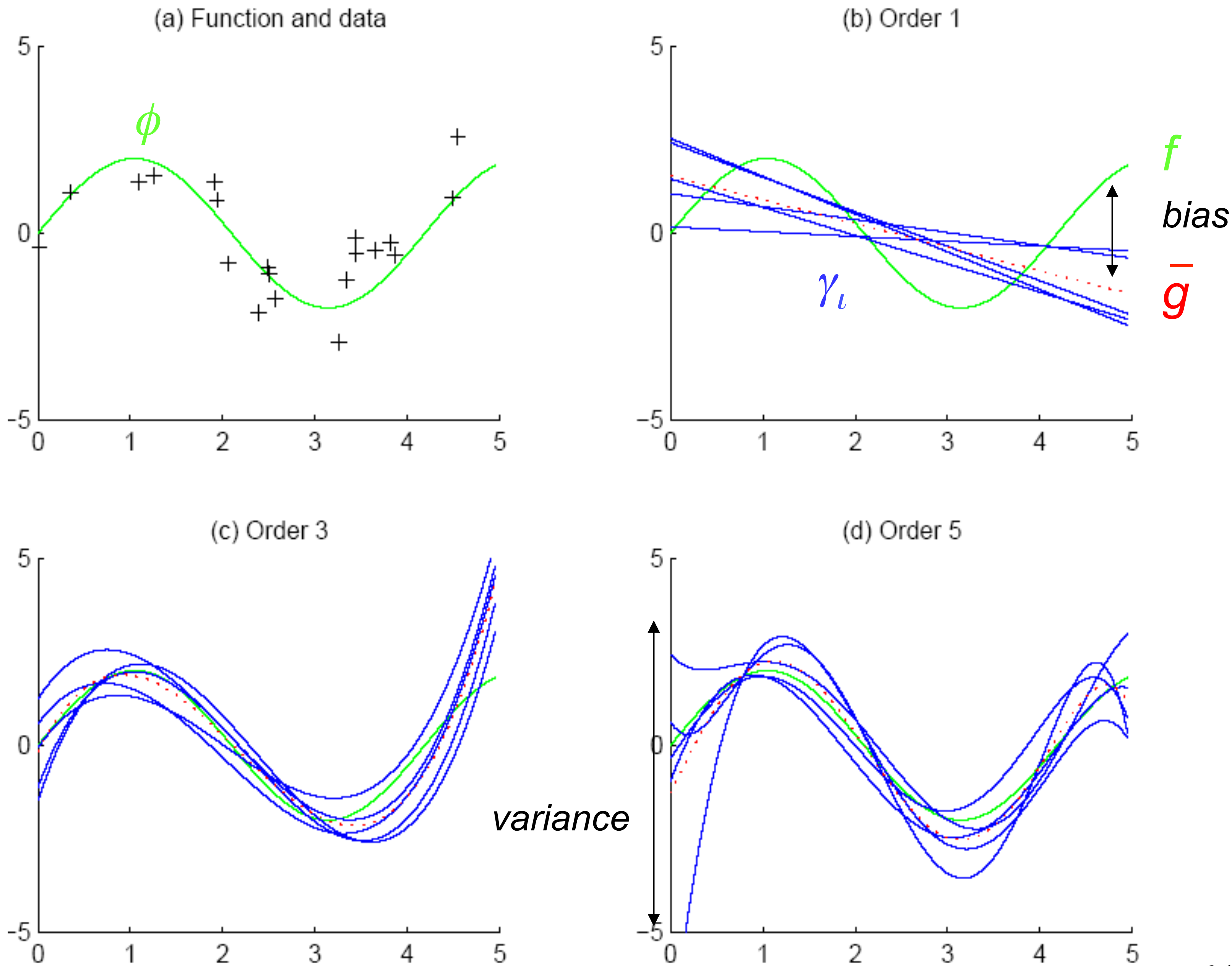


The Bias-Variance Trade-off

- 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.**



Minimum value of bias^2+variance is around λ=-0.31
This is close to the value that gives the minimum error on the test data.


(a) Function and data
(b) Order 1
(c) Order 3
(d) Order 5
φ
f
bias
γ
g
variance
5
0
-5
0
1
2
3
4
5

Model Selection Procedures

Cross validation: Measure the total error, rather than bias/variance, on a validation set.

- Train/Validation sets
- K-fold cross validation
- Leave-One-Out
- No prior assumption about the models