# Lecture 7 and beginning of 8:

STAT261: Introduction to Machine Learning

Linear Regression, Model Selection, Error & Bias

# Outline: Lecture 7 April 11 (ovelap into 8)

- Linear Regression
  - MLE solution
  - Goodness of fit
  - MSE Analysis
  - Prediction error
- Nonlinear and quadratic models
- Bias and Variance
- Model Order Selection
  - Overfitting
  - Cross validation

# Lecture 7: Linear Regression

- Assume simple linear model:
  - $y \approx f(x, \beta), \;\; f(x, \beta) \coloneqq \beta_0 + \sum_{j=1}^{p} \beta_j x_j$
- Goal: Learn the parameters $\beta$ from noisy data

$$y_i = f(x_i, \beta) + w_i = \beta_0 + \sum \beta_j x_{ij} + w_i$$

  - Noise is Gaussian $w_i \sim N(0, \sigma^2)$
  - Variance may be unknown
- Can be placed in linear model matrix form:

$$y = X\beta + w$$

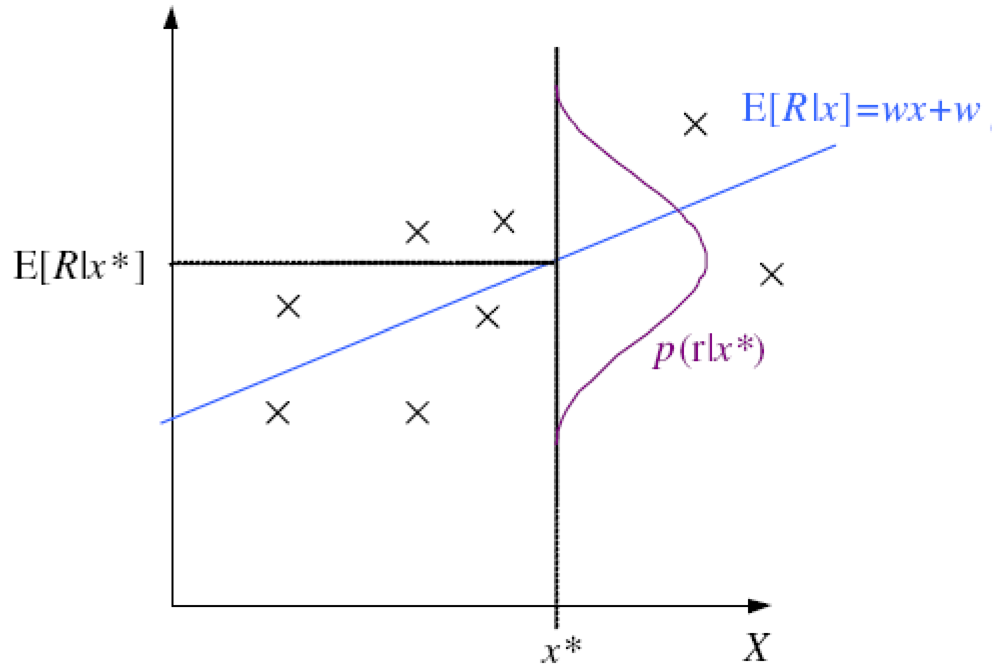
# Regression

$$y = f(x) + \varepsilon$$

$$\text{estimator}: g(x|\theta)$$

$$\varepsilon \sim \mathcal{N}(0, \sigma^2)$$

$$p(r|x) \sim \mathcal{N}(g(x|\theta), \sigma^2)$$




$$\mathcal{L}(\theta|\mathcal{X}) = \log \prod_{t=1}^{N} p(x^t, y^t)$$

$$= \log \prod_{t=1}^{N} p(y^t|x^t) + \log \prod_{t=1}^{N} p(x^t)$$

# Linear Regression

$$f\left(x^t|\beta_1,\beta_0\right)=\beta_1 x^t+\beta_0$$

$$\sum_t y^t = N\beta_0+\beta_1\sum_t x^t$$

$$\sum_t y^t x^t = \beta_0\sum_t x^t+\beta_1\sum_t \left(x^t\right)^2$$

$$\mathbf{A}=\begin{bmatrix} N & \sum_t x^t \\ \sum_t x^t & \sum_t \left(x^t\right)^2 \end{bmatrix},\ \beta=\begin{bmatrix}\beta_0\\ \beta_1\end{bmatrix},\ \mathbf{z}=\begin{bmatrix}\sum_t y^t \\ \sum_t y^t x^t\end{bmatrix}$$

$$\boldsymbol{\beta}=\mathbf{A}^{-1}\mathbf{z}$$

# MLE Solution

- Regression equivalent to a linear model:

$$y = X\beta + w$$

  - Assume $w_i \sim N(0, \sigma^2)$ iid.
- MLE given by LS solution:

$$\hat{\beta} = \arg\min_{\beta} \|y - X\beta\|^2 = \arg\min_{\beta} \; MSE(\beta)$$

  - $MSE(\beta) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$
  - $\hat{y}_i = x_i^T\beta$ = "predicted" value of $y_i$
- MLE solution: $\hat{\beta} = \left(X^TX\right)^{-1}X^Ty$
- Error variance: $var\left(\hat{\beta}\middle|\beta\right) = \sigma^2\left(X^TX\right)^{-1}$

# Regression: From LogL to Error

$$\mathcal{L}(\theta|\mathcal{X}) = \log \prod_{t=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{\left[y^t - f\left(x^t|\theta\right)\right]^2}{2\sigma^2}\right]$$

$$= -N \log\sqrt{2\pi}\sigma - \frac{1}{2\sigma^2}\sum_{t=1}^{N}\left[y^t - f\left(x^t|\theta\right)\right]^2$$

$$E(\theta|\mathcal{X}) = \frac{1}{2}\sum_{t=1}^{N}\left[y^t - f\left(x^t|\theta\right)\right]^2$$

$$\theta = [\beta_0\ \beta_1 \ldots\ldots]'$$

# Goodness of Fit

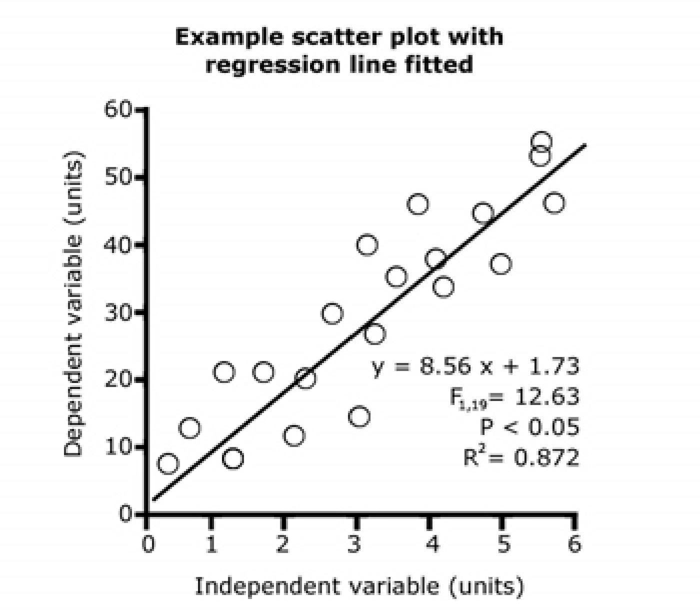
- $\hat{y}_i = x_i^T \beta$ = "predicted" value of $y_i$
- Average relative prediction error

$$R^2 := 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} y_i^2}$$

  - Value between 0 and 1
  - $R^2 = 1 \Rightarrow$ Perfect fit
  - $R^2 = 0 \Rightarrow$ No relation between variables
- Interpretation: $R^2 =$ fraction of variance not explained by the predictors

# Graphical Interpretation for $p = 1$

- Consider case when $p = 1$.
- Display data points $(x_i, y_i), i = 1, \dots, n$ in a scatter plot.
- Regression is fitting a line



# MLE for $p = 1$

- Write in matrix form: $y = X\beta + w$

$$X = \begin{bmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} = [\mathbf{1} \quad \boldsymbol{x}], \qquad \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}$$

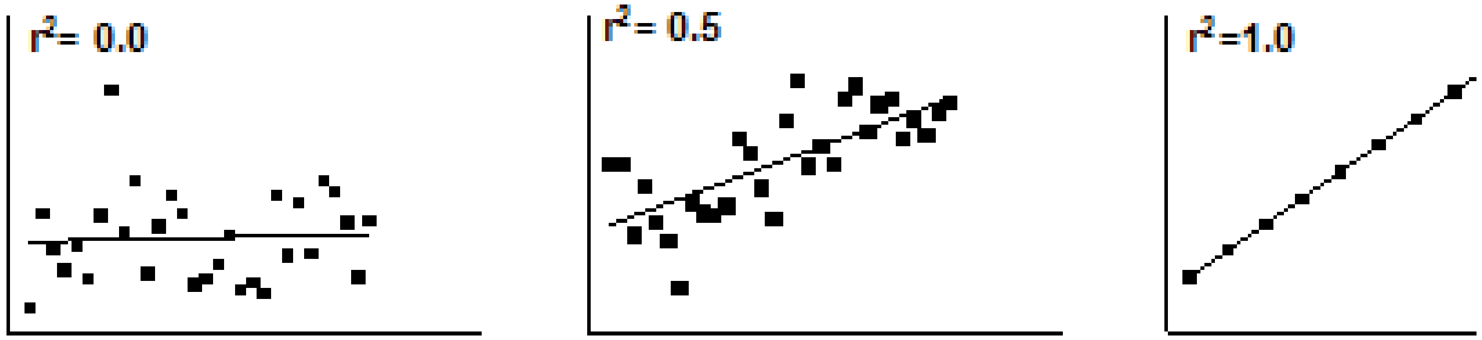
- Compute sample statistics:
  - $\mu_x = \frac{1}{n}\sum x_j,\ \mu_y = \frac{1}{n}\sum y_j$
  - $\sigma_x^2 = \frac{1}{n}\sum_j \left(x_j - \mu_x\right)^2, \sigma_y^2 = \frac{1}{n}\sum_j \left(y_j - \mu_y\right)^2$
  - $\sigma_{xy} = \frac{1}{n}\sum_j \left(x_j - \mu_x\right)\left(y_j - \mu_y\right)$

# MLE Solution for $p = 1$

- MLE is:
  - $\beta_1 = \sigma_{xy}/\sigma_x^2 = \rho_{xy}\sigma_y/\sigma_x$
  - $\beta_0 = \mu_x - \beta_1\mu_y$
- Prediction error: $R^2 = \rho_{xy}^2$
- Linear model:
  - $y = \sigma_{xy}/\sigma_x^2\,(x - \mu_x) + \mu_y$
  - Similar to the linear MMSE estimate

- Cross correlation $\rho_{xy}$ determines goodness of fit and slope

# Graphical Interpretation of $R^2$



# Outline

- Linear Regression
  - MLE solution
  - Goodness of fit
- MSE Analysis
  - Prediction error
- Nonlinear models
- Bias and Variance
- Model Order Selection
  - Overfitting
  - Cross validation

# Regression: Mean Squared Error with Independent Data

- Linear model $y = X\beta + w$
- Suppose that $X = [x_1 \cdots x_n]^T$ where
  - $x_i$ are iid random vectors $x_i \sim x$. Assume $E(x) = 0$
- Sample covariance is

$$\frac{1}{n}\sum_{i=1}^{n} x_i x_i^T = \frac{1}{n} X^T X$$

- Variance for large $n$,

$$var(\hat{\beta}) = \sigma_w^2 \left(X^T X\right)^{-1} \approx \frac{\sigma_w^2}{n} var(x)^{-1}$$

# MSE: Implications

- Error variance $var(\hat{\beta}) = \frac{\sigma_w^2}{n} var(x)^{-1}$
- Decreases linearly in number of samples $n$
- Increases linearly in $\sigma_w^2$
- If $var(x) = \sigma_x^2 I$ then $var(\hat{\beta}) = \frac{\sigma_w^2}{n\sigma_x^2} I$
  - Independent errors inversely proportional to SNR
- If $var(x) = diag(\sigma_1^2, \cdots, \sigma_p^2)$, then $var(\widehat{\beta_j}) = \frac{\sigma_w^2}{n\sigma_j^2}$
  - Error is lower in "directions" where variance of $x$ is larger.

# Regression: Prediction Error

- Train a model with $n$ samples
- Let $(x, y)$ be a new sample
  - Drawn from the same distribution as training data
  - But independent.
- $\hat{y} = x^T \beta$ = "predicted" value of $y$
- Prediction error is (proof on board):

$$E(y - \hat{y})^2 = E\left(x^T\left(\beta - \hat{\beta}\right) + w\right)^2 = \left(1 + \frac{p}{n}\right)\sigma_w^2$$

# Implications

- Prediction error: $E(y-\hat{y})^2 = \left(1+\frac{p}{n}\right)\sigma_w^2$
- Unpredictable term $\sigma_w^2$
- "Learnable" term: $p\sigma_w^2/n$
  - Decreases with samples $n$
  - Increases with number of parameters $p$
  - Does not depend on distribution of $x$
  - Assumes training data and test data have same statistics

# Outline

- Linear Regression
  - MLE solution
  - Goodness of fit
- Linear MSE Analysis
  - Prediction error
- Nonlinear and quadratic models
- Bias and Variance
- Model Order Selection
  - Overfitting
  - Cross validation

# Nonlinear Models

- What if we want a nonlinear fit?
- Polynomial fit: $y_i = \sum_{j=0}^{p} \beta_j x_i^j + w_i$
- Define new data matrix:

$$H = \begin{bmatrix} 1 & x_1 & \cdots & x_1^p \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \cdots & x_n^p \end{bmatrix}$$

- Problem has $p + 1$ parameters

# Polynomial Regression

$$f\left(x^t \middle| \beta_k, \ldots, \beta_2, \beta_1, \beta_0\right) = \beta_k \left(x^t\right)^k + \cdots + \beta_2 \left(x^t\right)^2 + \beta_1 x^t + \beta_0$$

$$\mathbf{H} = \begin{bmatrix} 1 & x^1 & \left(x^1\right)^2 & \cdots & \left(x^1\right)^k \\ 1 & x^2 & \left(x^2\right)^2 & \cdots & \left(x^2\right)^k \\ \vdots & & & & \\ 1 & x^N & \left(x^N\right)^2 & \cdots & \left(x^N\right)^2 \end{bmatrix} \quad \boldsymbol{y} = \begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^N \end{bmatrix}$$

$$\boldsymbol{\beta} = \left(\mathbf{H}^T \mathbf{H}\right)^{-1} \mathbf{H}^T \boldsymbol{y}$$

# Outline

- Linear Regression
  - MLE solution
  - Goodness of fit
- Linear MSE Analysis
  - Prediction error
- Nonlinear and quadratic models
- Model Order Selection:
  - Bias and Variance
  - Overfitting
  - Cross validation

# Bayesian Model Selection

- Prior on models, $p(\text{model})$

$$p(\text{model} \mid \text{data}) = \frac{p(\text{data} \mid \text{model})\, p(\text{model})}{p(\text{data})}$$

- Regularization, when prior favors simpler models
- Bayes, MAP of the posterior, $p(\text{model} \mid \text{data})$
- Could average over a number of models with high posterior

# Model Selection: Motivating Example

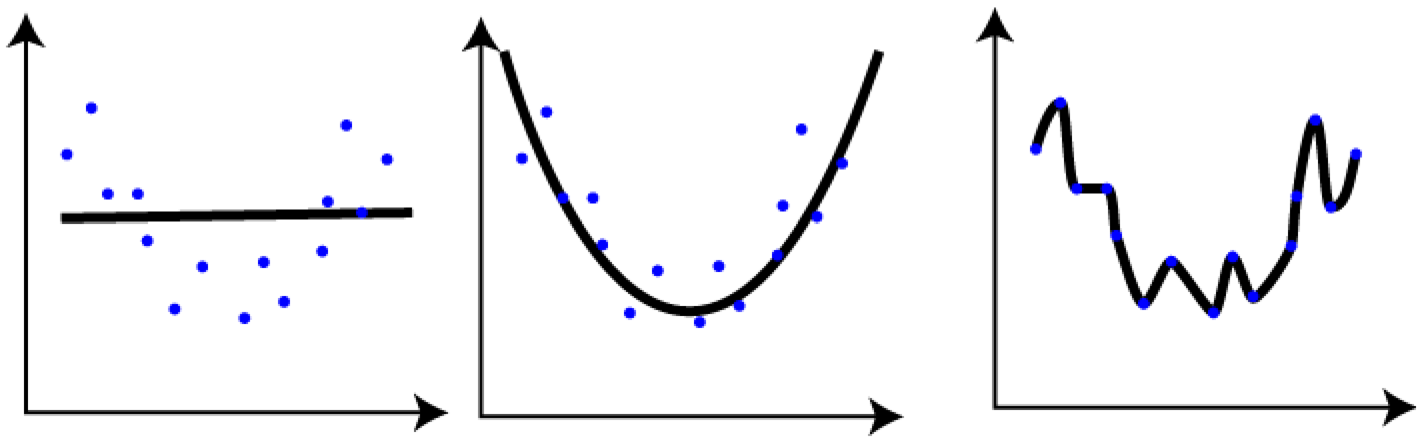
- Consider polynomial model: $y_i = \sum_{j=0}^{p} \beta_j x_i^j + w_i$
- $p$ is called the model order.
- What if $p$ is unknown?
- Can we learn it?
- Simple idea:
  - Try using different values of $p$
  - Measure prediction error (e.g. $R^2$) for each model order.
  - Select one with lowest prediction error
- This doesn't work. Why?

# Feature space

- Often map data to a "feature space"
- Then perform linear fit on features
- Can incorporate much larger set of models
- But, number of parameters grows
- For example: A quadratic function of $k$ variables requires:
  - $1$ constant term
  - $k$ linear terms
  - $k(k+1)/2$ quadratic terms (all the $x_i x_j$ pairs)
  - Total of $O(k^2)$ terms

# Overfitting of Data

- Polynomial model: $y_i = \sum_{j=0}^{p} \beta_j x_i^j + w_i$
- Prediction error will always decrease with $p$
- But, start to "fit noise"



$p = 0$

# Overfitting: What went wrong?

- Given data $(x_i, y_i)$ and model $y \approx f(x, \beta)$
  - Example: $f(x, \beta)$ is a polynomial model for some degree
- Find $\hat{\beta}$ from data.
- Now suppose data samples are iid $(x_i, y_i) \sim (x, y)$
- Want to evaluate: $MSE(\hat{\beta}) := E\left(y - f(x, \hat{\beta})\right)^2$
  - Don't know true distribution
- Use training error: $MSE_{train}(\hat{\beta}) := \frac{1}{n} \sum \left(y_i - f(x_i, \hat{\beta})\right)^2$
  - Assume $MSE_{train}(\hat{\beta}) \to MSE(\hat{\beta})$
  - Not valid since $\hat{\beta}$ depends on data $(x_i, y_i)$

# Under Modeling

- True relation: $y = f_0(x) + w$
  - $f_0(x) =$ "true" function
  - May not be exactly polynomial or within model class.
- Model: $\hat{y} = f(x, \beta) + w$
  - Example: Polynomial fit
- For a given $\beta$, prediction error is:

$$MSE(\beta) = E(y - \hat{y})^2 = \Delta(\beta) + \sigma^2$$

  - Undermodeling error: $\Delta(\beta) = E\left((f_0(x) - f(x, \beta)\right)^2$
  - Noise: $\sigma^2$

# Effect of Model Order

- Consider minimum model error:

$$\bar{\Delta} \coloneqq \min_{\beta} \Delta(\beta) = \min_{\beta} E\Big(\big(f_0(x) - f(x,\beta)\big)\Big)^2$$

- $\bar{\Delta}$ decreases with model order
  - Higher number of terms allows better approximation.
  - But, requires knowledge of $f_0(x)$ to find optimal $\beta$
- But, higher model order means:
  - More parameters to estimate
  - Higher variance in $\hat{\beta}$
  - Higher variance in $E\left(\left(f_0(x) - f\left(x,\hat{\beta}\right)\right)\right)^2$

# Bias-Variance Tradeoff

- Suppose we use a linear model: $f(x, \beta) = x^T \beta$
- Then: MSE is given by

$$MSE(\hat{\beta}) = \Delta(\hat{\beta}) + \sigma^2$$

$$= E\left(f_0(x) - x^T \hat{\beta}\right)^2 + \sigma^2$$

$$= E(f_0(x) - x^T E\hat{\beta})^2 + var\left(x^T \hat{\beta}\right) + \sigma^2$$

Unlearnable component

Bias term
Decreases with model order

Variance term
Increases with model order

- Model order selection trades off between bias & variance

# Regression: Bias and Variance

$$E\left[\left(y-g\left(x\right)\right)^2 | x\right] = E\left[\left(y-E\left[\mathrm{y}|x\right]\right)^2 | x\right] + \left(E\left[\mathrm{y}|x\right] - f\left(x\right)\right)^2$$

*noise* *squared error*

$$E_X\left[\left(E\left[\mathrm{y}|x\right] - f\left(x\right)\right)^2 | x\right] =$$

$$\left(E\left[\mathrm{y}|x\right] - E_X\left[f\left(x\right)\right]\right)^2 + E_X\left[\left(f\left(x\right) - E_X\left[f\left(x\right)\right]\right)^2\right]$$

*bias* *variance*

# Simple Regression: Bias/Variance Example

- Example: $g_i(x)=2$ has no variance and high bias

  $g_i(x)= \sum_t r^t_i/N$ has lower bias with variance

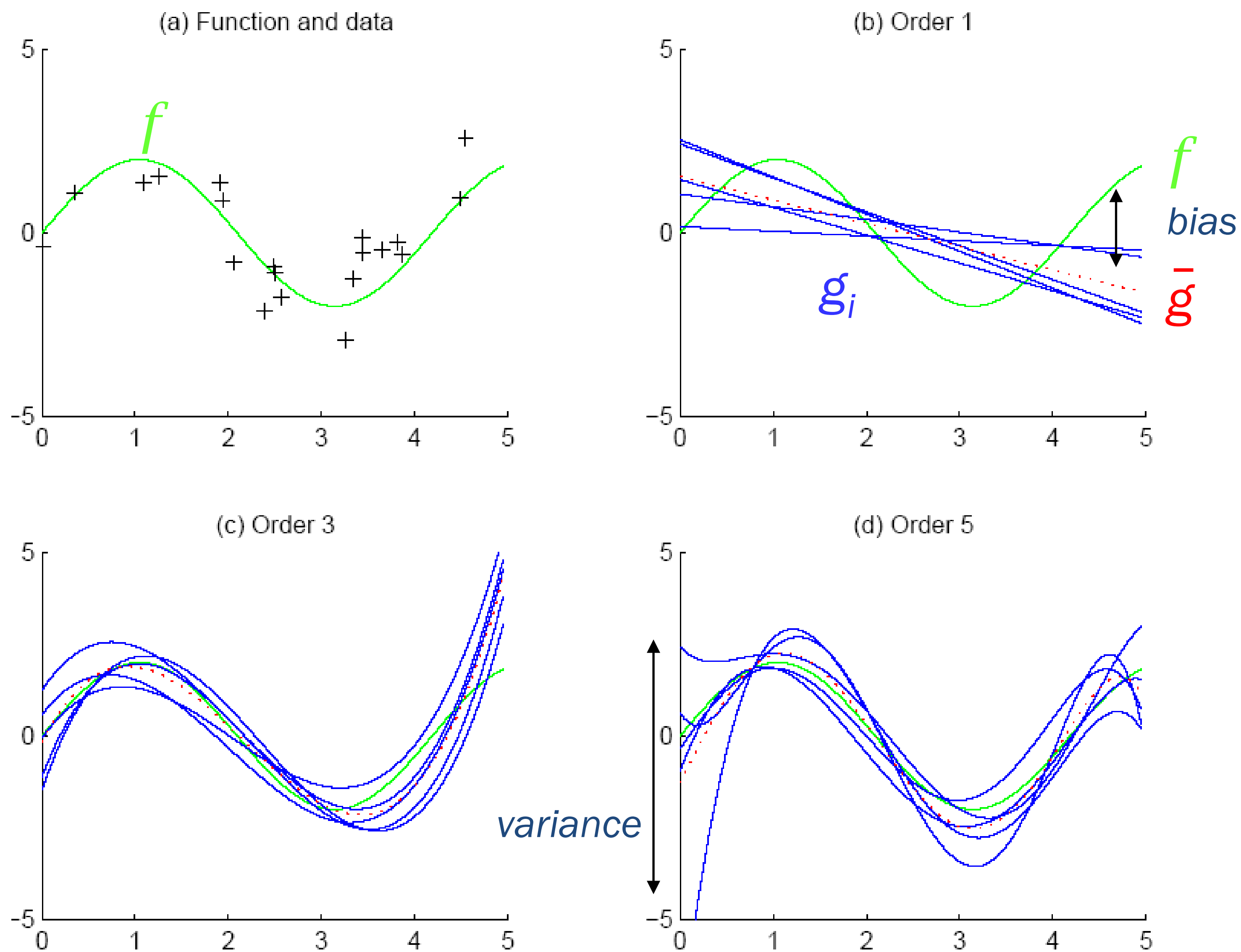
- As we increase complexity,

  bias decreases (a better fit to data) and

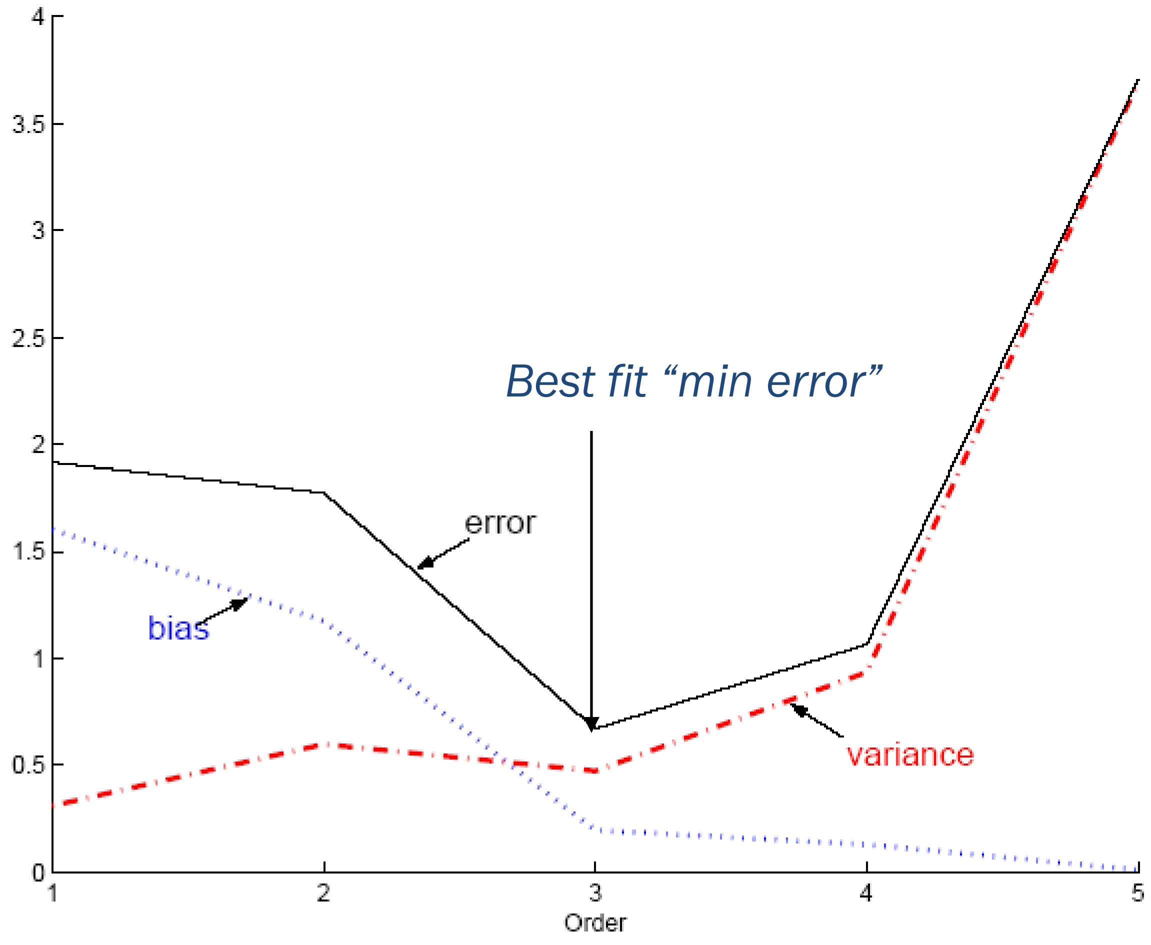
  variance increases (fit varies more with data)

  (in this constant case, variance increase A LOT)

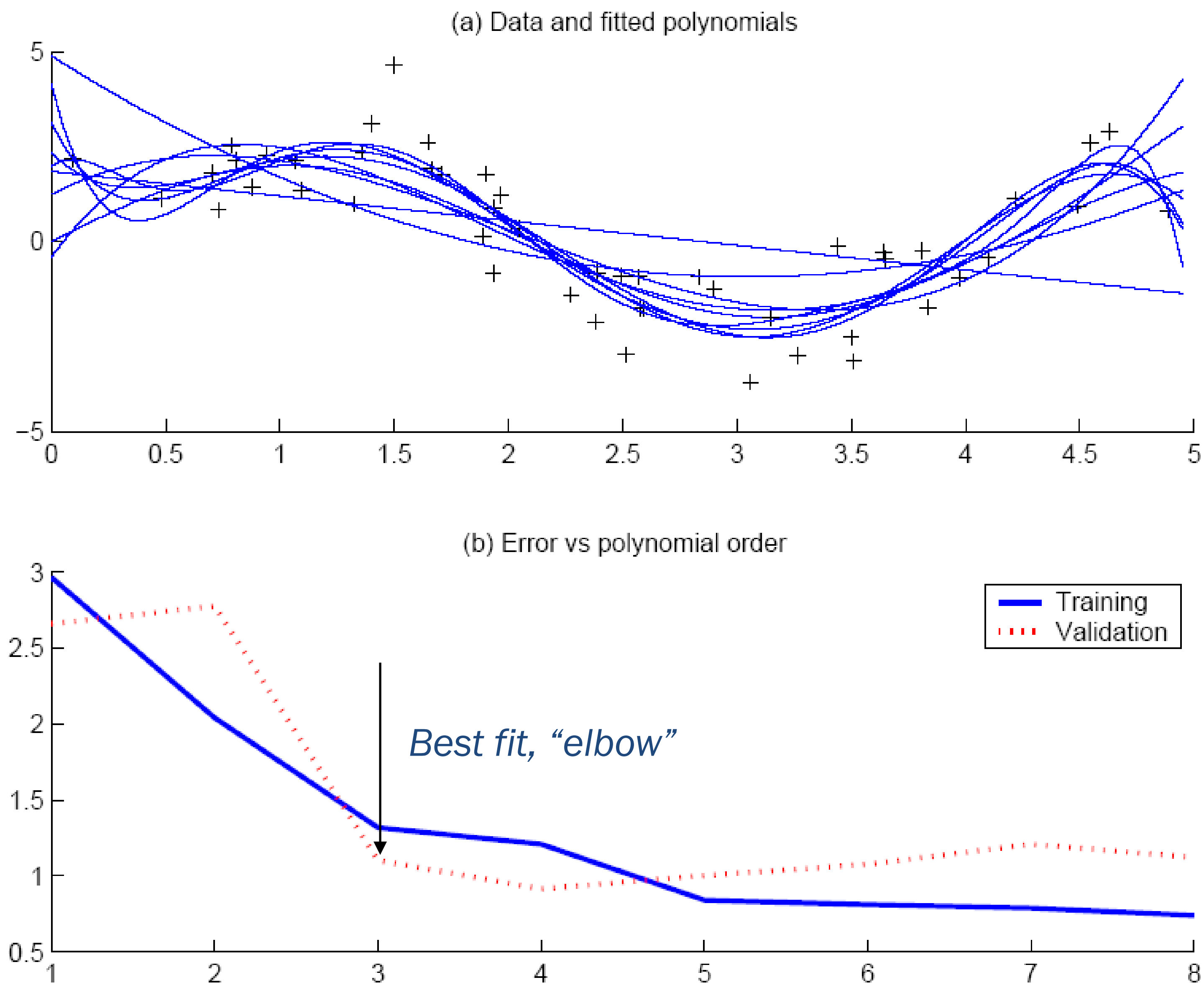
# Fitting Higher Order Polynomials: Regression



# Polynomial Regression



# Polynomial Regression



# Other Error Measures

- Square Error: $$E(\theta|\mathcal{X}) = \frac{1}{2}\sum_{t=1}^{N}\left[y^t - f\left(x^t|\theta\right)\right]^2$$
- Relative Square Error: $$E(\theta|\mathcal{X}) = \frac{\sum_{t=1}^{N}\left[y^t - f\left(x^t|\theta\right)\right]^2}{\sum_{t=1}^{N}\left[y^t - \overline{y}\right]^2}$$
- Absolute Error: $E(\theta|X) = \sum_t |y^t - f(x^t|\theta)|$
- ε-sensitive Error:

$$E(\theta|X) = \sum_t 1(|y^t - g(x^t|\theta)| > \varepsilon)(|y^t - g(x^t|\theta)| - \varepsilon)$$

# Outline

- Linear Regression
- Linear MSE Analysis
  - Prediction error
- Nonlinear and quadratic models
- Bias and Variance
- Model Order Selection
  - Overfitting and Underfitting
  - Cross validation

# Cross validation

- Divide data into two components:
  - Training data ⇒ Learn parameters from first $m$ samples
  - Test data ⇒ Test model on remaining $n - m$ samples
- Model training: Take first $m < n$ samples: $\hat{\beta} = \arg\min_{\beta} MSE_{train}(\beta)$,

$$MSE_{train}(\beta) = \frac{1}{m}\sum_{i=1}^{m}\left(y_i - f(x_i, \beta)\right)^2$$

- Test model on remaining $n - m$ samples:

$$MSE_{test}(\beta) = \frac{1}{n-m}\sum_{i=m+1}^{n}\left(y_i - f(x_i, \beta)\right)^2$$

- Select model with lowest $MSE_{test}(\hat{\beta})$

# Why Does Cross-Validation Work?

- Assume data $(x_i, y_i)$, $\mathrm{i} = 1, \ldots, n$ is iid
- MLE $\hat{\beta}$ depends on first $m$ samples.
- Thus, $\hat{\beta}$ is independent of test data (remaining $n - m$ samples)
- Therefore, if $n - m \to \infty$,

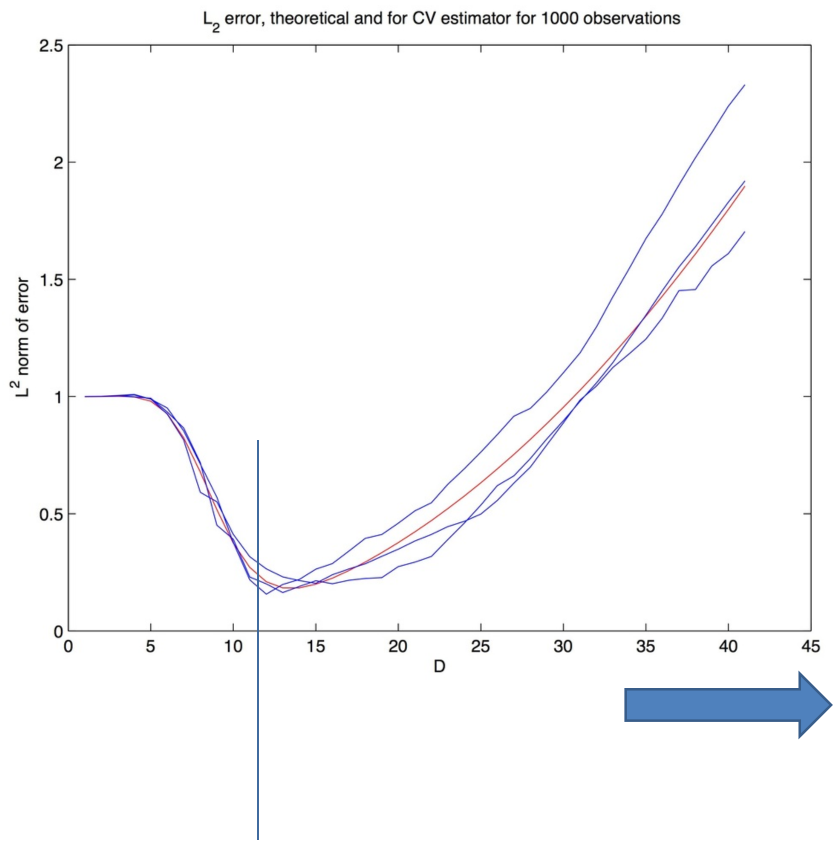
$$MSE_{test}(\beta) = \frac{1}{n-m} \sum_{i=m+1}^{n} \left(y_i - f(x_i, \beta)\right)^2 \to MSE_{true}(\beta)$$

- Asymptotically correct estimate of true MSE:

$$MSE_{true}(\beta) = E\left(y - f(x, \beta)\right)^2$$

# Cross validation for model selection

MSE




Model order

Optimal model order

- From https://www.maths.nottingham.ac.uk/personal/pmzmig/research-topics.html

# Variants on Cross Validation

- Hold out: Divide data into two parts: Train & test
  - Can optimize relative fractions of data
- K-fold validation
  - Divide data into K parts.
  - Use K-1 parts for training. Use remaining for test.
  - Average over the K test choices
  - More accurate, but requires K fits of parameters
- Leave one out cross validation (LOOCV)
  - Take K = N so one sample is left out.
  - Most accurate, but requires N model fittings

# Model Selection: Other factors

- Cross-validation: Measure generalization accuracy by testing on data unused during training
- Regularization: Penalize complex models

  E'=error on data + λ model complexity

  Akaike's information criterion (AIC), Bayesian information criterion (BIC)
- Minimum description length (MDL): Kolmogorov complexity, shortest description of data
- Structural risk minimization (SRM)