# Introduction to Statistical Learning and Machine Learning

## Chap 2 - Linear Regression(1)

Yanwei Fu
SDS, Fudan University

# Chap 2 - Linear Regression(1)

**Main Content**

1. Simple Linear Model
2. Least Squares;
3. The Bias-Variance tradeoff;

# Chap 2 - Linear Regression(1)

Recap

# Recap: Notations of Supervised Learning (1)

We use uppercase letters such as X, Y or G when referring to the generic aspects of a variable.

$X$ input variables , a.k.a., features, predictors, independent variables.

$Y$ output variables, a.k.a., response or dependent variable.

$Y = f(X) + \epsilon$ $\epsilon$ captures measurement errors and other discrepancies.

$l : X \to Y$ Loss function, $l\left(y, y^{'}\right)$ is the cost of predicting $y^{'}$ if $y$ is correct.

***Regression*** **when we predict quantitative outputs (infinite set);**
***Classification*** **when we predict qualitative outputs (finite set, e.g. Group labels, Ordered,)**

Training set: $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)\}$ sampled from the joint distribution (*X*, *Y*).

**i.i.d**: Independent and identically distributed random variables.
A sequence or other collection of random variables is i.i.d. if each random variable has the same probability distribution as the others and all are ***mutually*** independent.

$$\mathrm{P}(A \cap B) = \mathrm{P}(A)\mathrm{P}(B).$$

# Recap: Notations of Supervised Learning

Matrices are represented by bold uppercase letters. $\boldsymbol{X}$

Observed values are written in lowercase; hence the *i*-th observed value of $X$ is written as $x_i$

Dummy Variable：K-level qualitative variable is represented by a vector of K binary variables or bits, only one of which is “on" at a time. *One-hot vector* in Deep learning.

# Some Important Concepts

Mean squared error (MSE), $$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2,$$

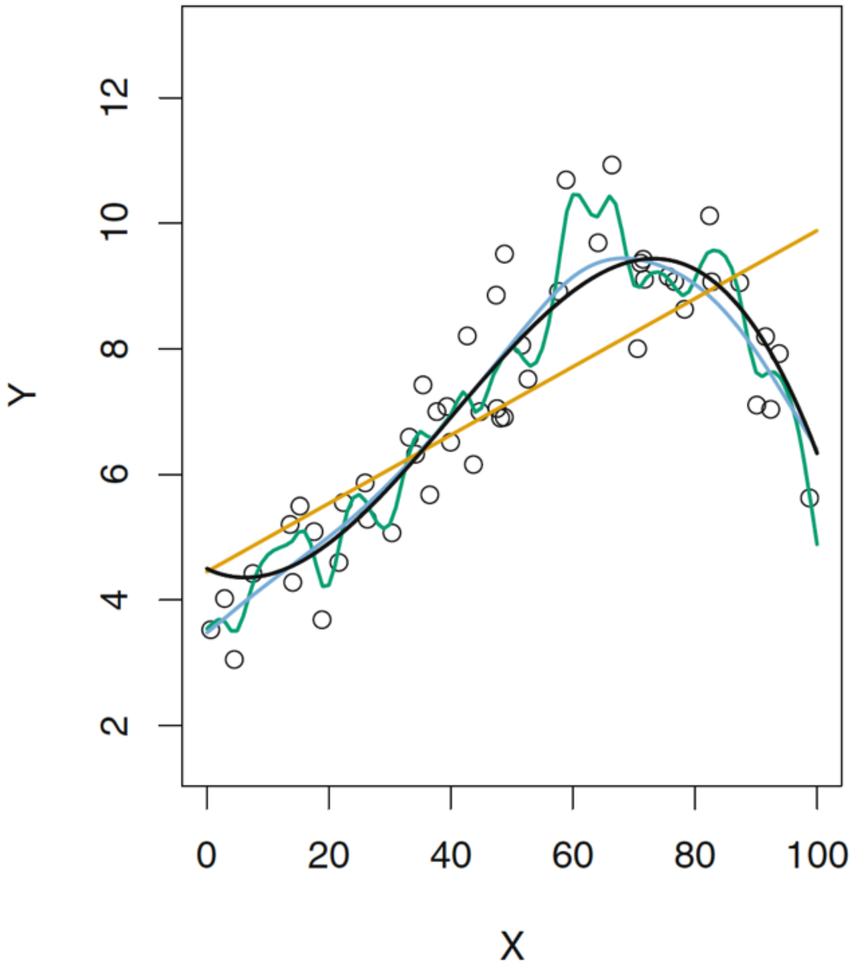
We are interested in the accuracy of the predictions that we obtain when we apply our method to previously unseen test data.

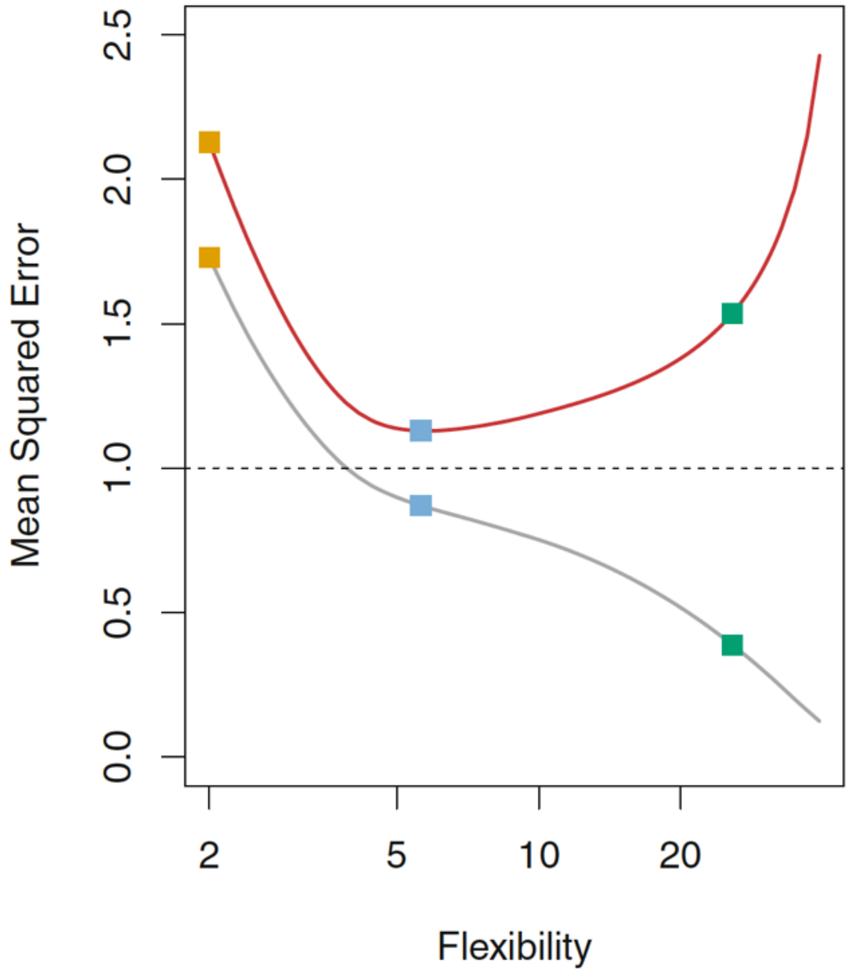
Test MSE $\text{Ave}(y_0 - \hat{f}(x_0))^2$, $(x_0, y_0)$ is a previously unseen test observation.

**Overfitting**: a method yields a small training MSE but a large test MSE, we are said to be overfitting the data

This happens because our statistical learning procedure is working too hard to find patterns in the training data, and may be picking up some patterns that are just caused by random chance rather than by true properties of the unknown function $f$.

**Underfitting**: a method function (polynomial with degree 1) is not sufficient to fit the training samples. (Not small enough MSE on training data).




Left: Data simulated from $f$, shown in black. Three estimates of $f$ are shown: the linear regression line (orange curve), and two smoothing spline fits (blue and green curves). Right: Training MSE (grey curve), test MSE (red curve), and minimum possible test MSE over all methods (dashed line). Squares represent the training and test MSEs for the three fits shown in the left-hand panel.

The number of **degrees of freedom** (flexibility) is the number of values in the final calculation of a statistic that are free to vary.

Linear regression with two degrees of freedom.

**Expectation operator:** $\mathbf{E}[\cdot]$ Constants, Monotonicity, Linearity.

$$\mathbf{E}[c] = c.$$

$X \leq Y$ Almost surely $\mathbf{E}[X] \leq \mathbf{E}[Y]$

$$\mathbf{E}[X + c] = \mathbf{E}[X] + c$$
$$\mathbf{E}[X + Y] = \mathbf{E}[X] + \mathbf{E}[Y]$$
$$\mathbf{E}[aX] = a\,\mathbf{E}[X]$$

**Conditional expectation,** *For any two discrete random variables X, Y.*

$$\mathbf{E}[X \mid Y = y] = \sum_x x \cdot \mathrm{P}(X = x \mid Y = y), \qquad f : y \mapsto \mathbf{E}(X \mid Y = y).$$

We call it *conditional expectation of X with respect to Y.* $\mathbf{E}[X] = \mathbf{E}[\mathbf{E}[X \mid Y]].$

# Chap 2 - Linear Regression(1)

Non-parametric methods.
Vs. Parametric methods

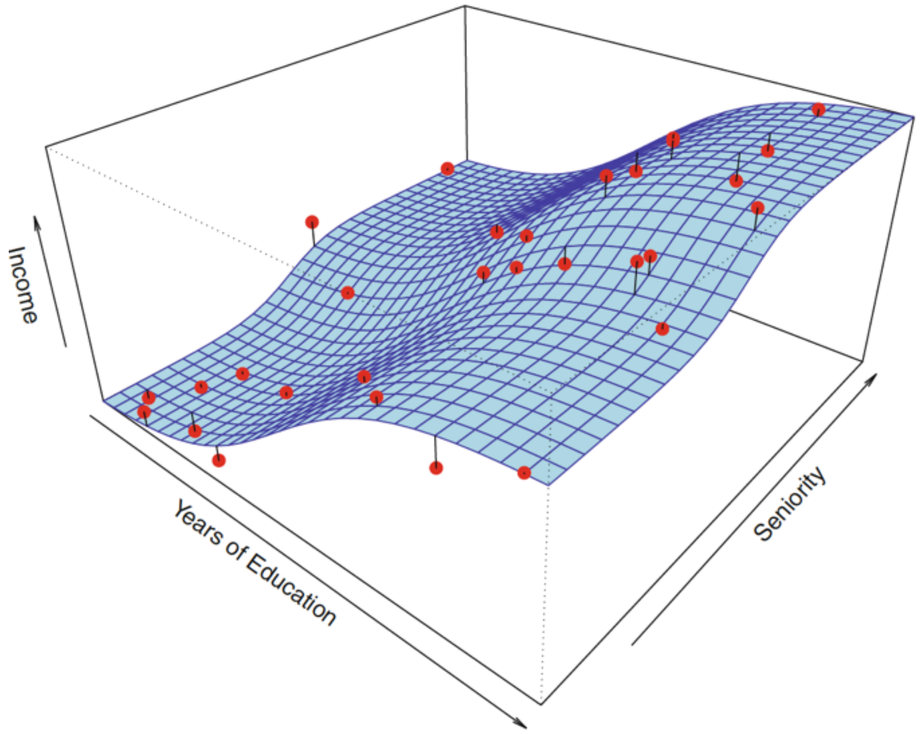
# Two basic ideas of *How Do We Estimate $f$?*

- **Parametric Methods**: Linear Least Square -> generalized linear models
  1. we make an assumption about the functional form, or shape, of $f$ $f(X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p.$
  2. we use the training data to fit the model (parameters); $Y \approx \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p.$

- **Non-parametric Methods**: Nearest Neighbors -> kernel method and SVM
  1. We do not make explicit assumptions about the functional form of $f$. Instead they seek an estimate of $f$ that gets as close to the data points as possible without being too rough or wiggly.
  2. Not make explicit assumptions about the functional form of $f$.

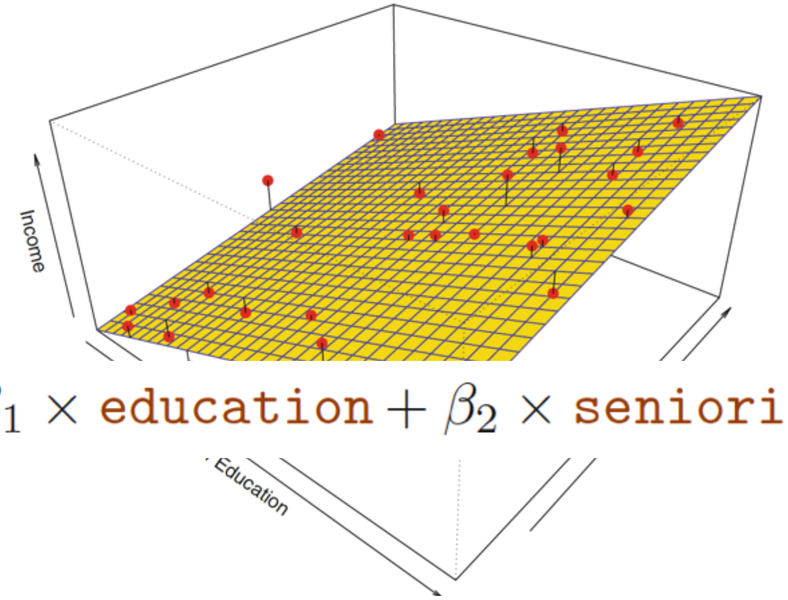
# An example of Parametric Vs. Non-parametric methods

The observations are displayed in red; the yellow plane indicates the fitted model;




The plot displays income as a function of years of education and seniority in the Income data set. The blue surface represents the true underlying relationship between income and years of education and seniority, which is known since the data are simulated. The red dots indicate the observed values of these quantities for 30 individuals.

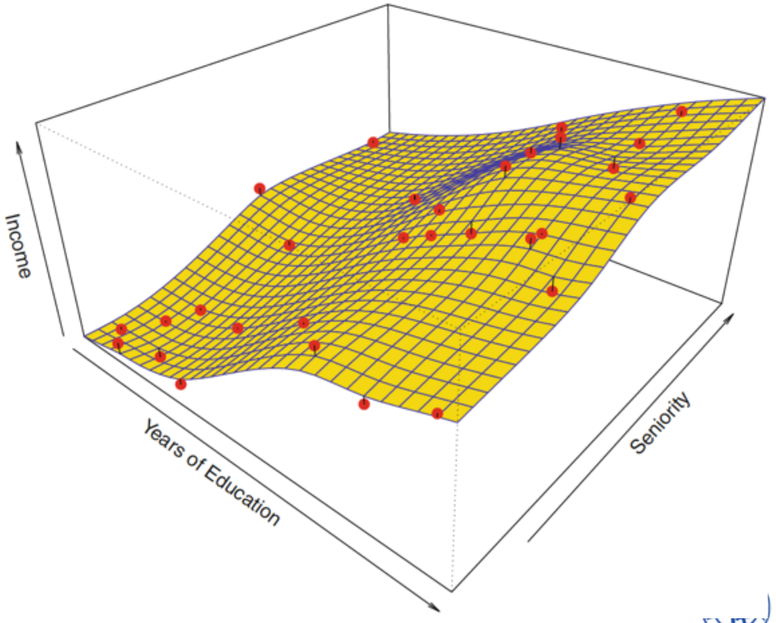
A linear model fit by least squares to the Income data




$$\texttt{income} \approx \beta_0 + \beta_1 \times \texttt{education} + \beta_2 \times \texttt{seniority}$$

A smooth thin-plate spline fit to the Income data.

薄板样条函数



# Parametric Method Vs. Non-parametric Methods

| | Advantages | Disadvantages |
|---|---|---|
| Parametric method | • Reducing the *hard* problem down to estimating a set of parameters (*easy*);<br>• Low variance; | • the model we choose will usually not match the true unknown form of $f$.<br>• These more complex models can lead to a phenomenon known as overfitting the data, which means they follow the errors, or noise, too closely. |
| Non-Parametric method | • Avoiding the assumption of a particular functional form for $f$. | • they do not reduce the problem of estimating $f$ to a small number of parameters, a very large number of observations (far more than is typically needed for a parametric approach) is required in order to obtain an accurate estimate for $f$. |

Why is it necessary to introduce so many different statistical learning approaches, rather than just a single best method? *There is no free lunch in statistics*: no one method dominates all others over all possible data sets. On a particular data set, one specific method may work best, but some other method may work better on a similar but different data set.

# Chap 2 - Linear Regression(1)

- Simple Linear Regression;
- Bias-Variance Trade-off in a Nutshell;
- Key concepts of Statistics in Linear Regression;

# Simple Linear Regression

Parametric method

- ***Simple Linear Regression***: *Y* is is **quantitative** (e.g price, blood pressure);on the basis of a single predictor variable *X.*

$$Y \approx \beta_0 + \beta_1 X.$$

*Symbols explanations:*

- You might read "≈" as "*is approximately modeled as*";
- $\beta_0$ and $\beta_1$ are two unknown constants that represent the intercept and slope terms;
- saying that we are regressing *Y* on *X* (or *Y* onto *X*).
- hat symbol, ˆ , to denote the estimated value for an unknown parameter or coefficient, or to denote the predicted value of the response.

So how to estimate the Coefficients?

# Estimating the Coefficients of Simple Linear Regression

Simple Linear Regression

$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$ be the prediction for *Y* based on the *i*-th value of *X*.

$e_i = y_i - \hat{y}_i$ represents the *i*-th residual —this is the difference between the *i*-th observed response value and the *i*-th response value that is predicted by our linear model.

Residual sum of squares: $\mathrm{RSS} = e_1^2 + e_2^2 + \cdots + e_n^2$

Least squares coefficient estimators:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2},$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x},$$

How to compute the minimizer?

Homework: prove it.

$$\bar{y} \equiv \frac{1}{n}\sum_{i=1}^{n} y_i \qquad \bar{x} \equiv \frac{1}{n}\sum_{i=1}^{n} x_i$$

# Assessing the Accuracy of the Coefficient Estimates

Simple Linear Regression

Population regression line $Y = \beta_0 + \beta_1 X + \epsilon.$ mean-zero random error term.

$\beta_0$ is the intercept term—that is, the expected value of *Y* when *X* = 0,

$\beta_1$ is the slope—the average increase in Y associated with a one-unit increase in *X.*

Suppose we annotate $\mu$ as the population mean of random variable $Y$

A reasonable estimate $\hat{\mu} = \bar{y}, \quad \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$

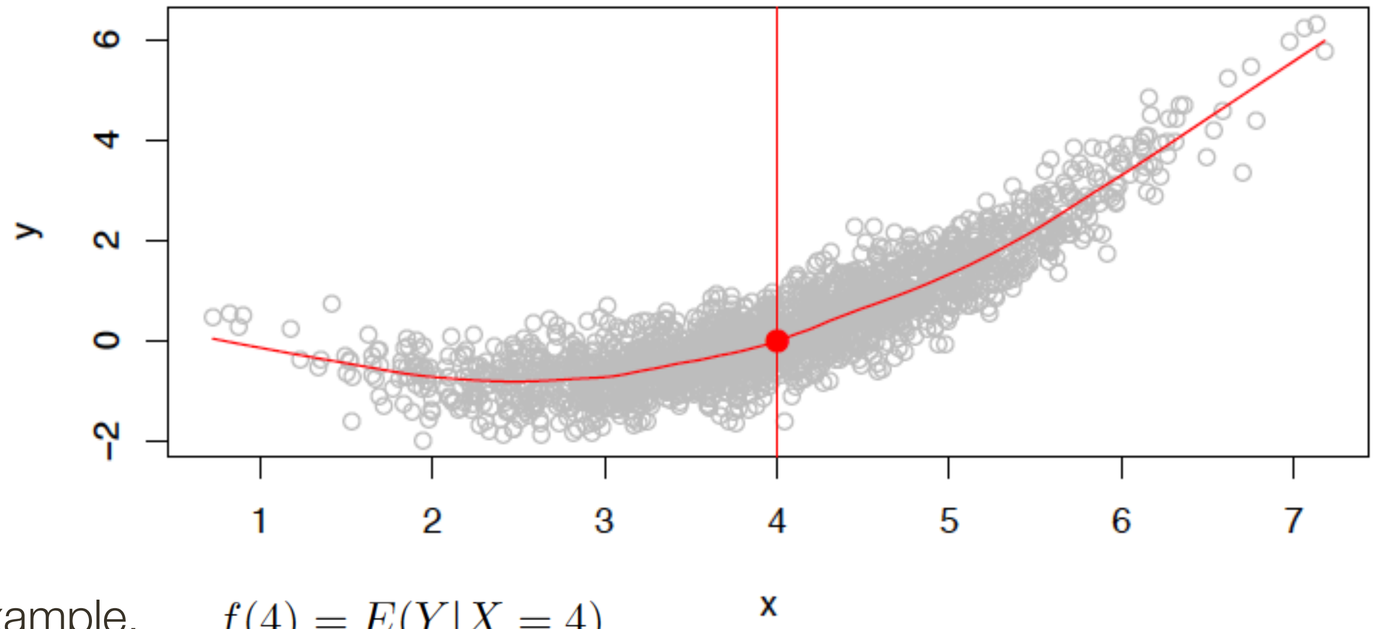
If we use the sample mean $\hat{\mu}$ to estimate $\mu$, this estimate is unbiased.

So how accurate is the estimation?

Standard error of $\hat{\mu}$ $\mathrm{Var}(\hat{\mu}) = \mathrm{SE}(\hat{\mu})^2 = \frac{\sigma^2}{n},$ $\sigma$ is the standard deviation of each of the realizations $y_i$ for uncorrelated observations.

# Bias-Variance Trade-off(1)

Is there an ideal $f(X)$?




Take $X$=4 as and example, $f(4) = E(Y|X = 4)$

$f(x) = E(Y|X = x)$ is called the regression function.

We minimise least square errors over all points $X$=$x$

$$E[(Y - \hat{f}(X))^2|X = x] = \underbrace{[f(x) - \hat{f}(x)]^2}_{Reducible} + \underbrace{\mathrm{Var}(\epsilon)}_{Irreducible}$$

# Bias-Variance Trade-off(2)

$$E[(Y - \hat{f}(X))^2 | X = x] = \underbrace{[f(x) - \hat{f}(x)]^2}_{Reducible} + \underbrace{\mathrm{Var}(\epsilon)}_{Irreducible}$$

$$\hat{Y} = \hat{f}(X),$$

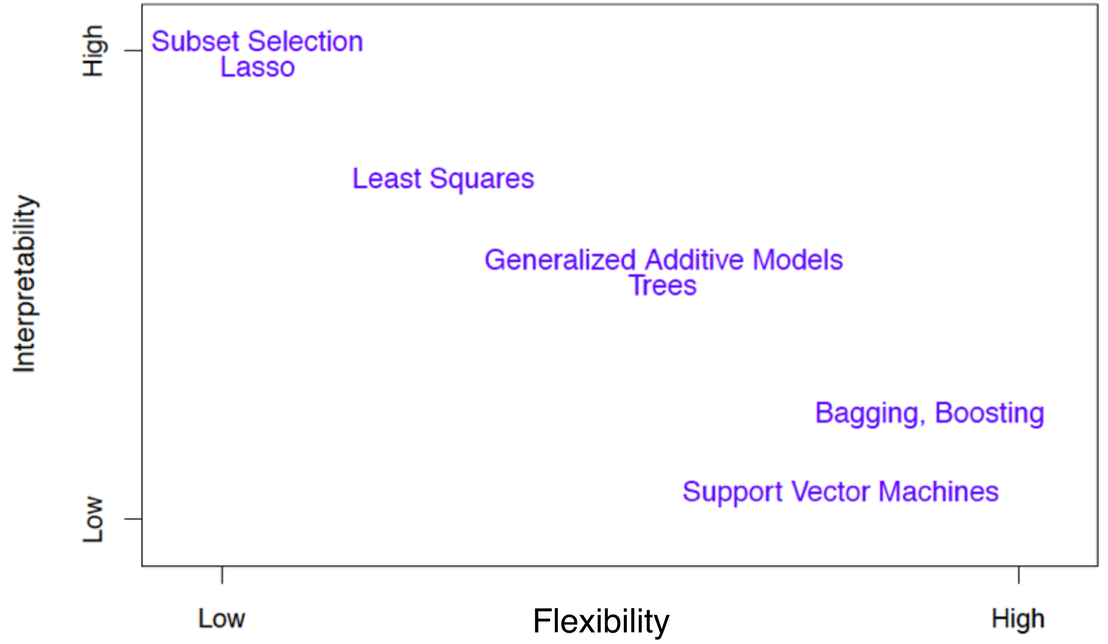
$$\begin{aligned} E(Y - \hat{Y})^2 &= E[f(X) + \epsilon - \hat{f}(X)]^2 \\ &= \underbrace{[f(X) - \hat{f}(X)]^2}_{\text{Reducible}} + \underbrace{\mathrm{Var}(\epsilon)}_{\text{Irreducible}}, \end{aligned}$$

$E(Y - \hat{Y})^2$ represents the average, or expected value, of the squared difference between the predicted and actual value of *Y.*

$\mathrm{Var}(\epsilon)$ represents the variance associated with the error term $\epsilon$.

# Some Trade-off

- Prediction accuracy versus interpretability.
  - Linear models are easy to interpret; thin-plate splines(薄板样条插值) are not.
- Good fit versus over-fit or under-fit.
  - How do we know when the fit is just right?
- Parsimony versus black-box.
  - We often prefer a simpler model involving fewer variables over a black-box predictor involving them all.



# Standard Error and Confidence Intervals

Simple Linear Regression

Standard Errors $\hat{\beta}_0$ and $\hat{\beta}_1$ $\mathrm{SE}(\hat{\beta}_0)^2 = \sigma^2 \left[ \frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \right]$, $\mathrm{SE}(\hat{\beta}_1)^2 = \frac{\sigma^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$, $\sigma^2 = \mathrm{Var}(\epsilon)$.

$\epsilon_i$ for each observation are uncorrelated with common variance $\sigma^2$

The estimate of $\sigma$ residual standard error is known as the *residual standard error.*

$$\mathrm{RSE} = \sqrt{\mathrm{RSS}/(n-2)}.$$

1, Standard errors can be used to compute *confidence intervals.* A 95% confidence interval is defined as a range of values such that with 95% probability, the range will contain the true unknown value of the parameter:

$$\hat{\beta}_1 \pm 2 \cdot \mathrm{SE}(\hat{\beta}_1).$$

There is approximately a 95% chance that the interval, (assume Gaussian Errors here).

$$\left[ \hat{\beta}_1 - 2 \cdot \mathrm{SE}(\hat{\beta}_1),\ \hat{\beta}_1 + 2 \cdot \mathrm{SE}(\hat{\beta}_1) \right]$$

will contain the true value of $\beta_1$

# Hypothesis Testing

Simple Linear Regression

2, Standard errors can also be used to perform hypothesis tests on the coefficients. The most common hypothesis test involves testing the null hypothesis of

$H_0$ : There is no relationship between *X* and *Y* versus the alternative hypothesis.

$H_A$ : There is some relationship between *X* and *Y* :

Mathematically, this corresponds to testing

$$H_0 : \beta_1 = 0 \quad \text{versus} \quad H_A : \beta_1 \neq 0,$$

since if $\beta_1 = 0$ then the model reduces to $Y = \beta_0 + \epsilon$ , and *X* is not associated with *Y* .

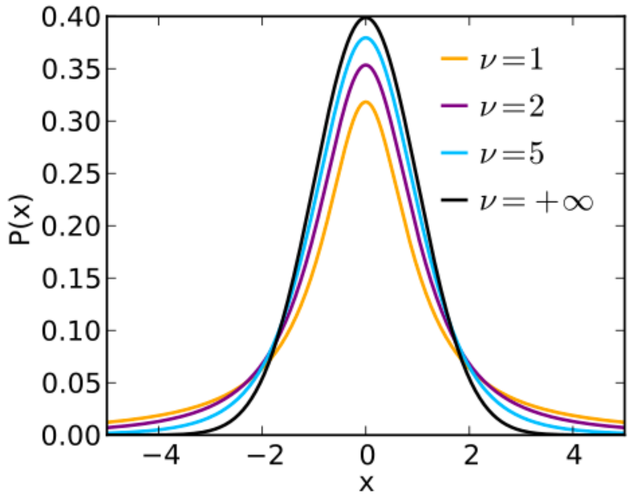
To test the null hypothesis, we compute a t-statistic, given by

$$t = \frac{\hat{\beta}_1 - 0}{\text{SE}(\hat{\beta}_1)},$$

This will have a *t*-distribution with n-2 degrees of freedom, assuming $\beta_1 = 0$

Using statistical software, it is easy to compute the probability of observing any value equal to $|t|$ or larger. We call this probability the p-value.

Probability density function of Student's t-distribution

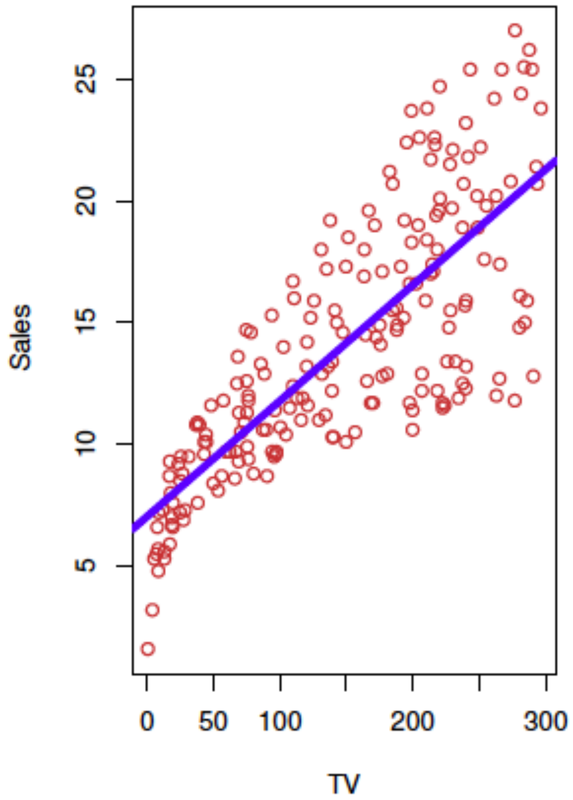


# Results for the Advertising Data

Simple Linear Regression

Typical p-value cutoffs for rejecting the null hypothesis are 5 or 1%. When n = 30, these correspond to t-statistics of around 2 and 2.75.

| | Coefficient | Std. error | t-statistic | p-value |
|---|---|---|---|---|
| Intercept | 7.0325 | 0.4578 | 15.36 | $< 0.0001$ |
| TV | 0.0475 | 0.0027 | 17.67 | $< 0.0001$ |




Advertising Data

Once we have rejected the null hypothesis in favor of the alternative hypothesis, it is natural to want to quantify the extent to which the model fits the data. The quality of a linear regression fit is typically assessed using two related quantities: the residual standard error (RSE) and the $R^2$ statistic.

provides details of
model for the regre
units sold on TV ac
the Advertising da
coefficients for $\hat{\beta}0$
large relative to the
the t-statistics are
probabilities of see
is true are virtually
conclude that $\beta 0 =$

# Simple Linear Regression

Residual Standard Error $$\mathrm{RSE} = \sqrt{\frac{1}{n-2}\mathrm{RSS}} = \sqrt{\frac{1}{n-2}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2},$$

Residual sum-of-squares $RSS = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2.$

R-squared or fraction of variance explained is

$$R^2 = \frac{\mathrm{TSS} - \mathrm{RSS}}{\mathrm{TSS}} = 1 - \frac{\mathrm{RSS}}{\mathrm{TSS}}$$

$\mathrm{TSS} = \sum_{i=1}^{n}(y_i - \bar{y})^2$ is the total sum of squares.

| Quantity | Value |
|---|---|
| Residual Standard Error | 3.26 |
| $R^2$ | 0.612 |
| F-statistic | 312.1 |

**Advertising data results.**

In this simple linear regression setting, we have $R^2 = r^2$ where *r* is the correlation between *X* and *Y*:

$$r = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \overline{y})^2}}.$$

# Chap 2 - Linear Regression(1)

Linear Regression from Probabilistic Perspective

[1] Chap 3, Bishop 2006

# Maximum Likelihood and Least Squares (1)

Optional subtitle

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 Least Squares (2)

Optional subtitle

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 (3)

Optional subtitle

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

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

# Geometry of Least Squares

Optional subtitle

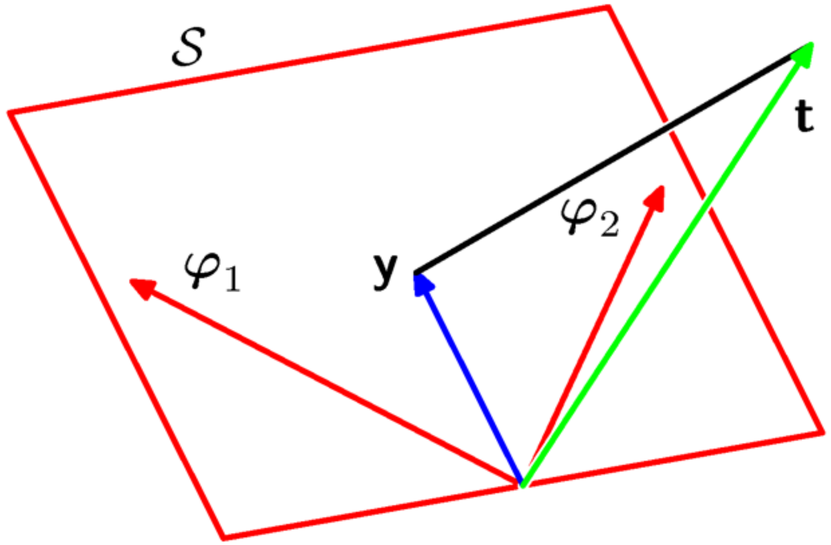
Consider $\mathbf{y} = \mathbf{\Phi}\mathbf{w}_{\mathrm{ML}} = [\boldsymbol{\varphi}_1, \ldots, \boldsymbol{\varphi}_M]\,\mathbf{w}_{\mathrm{ML}}.$

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

N-dimensional

M-dimensional

$\boldsymbol{\varphi}_1, \ldots, \boldsymbol{\varphi}_M$




S is spanned by .

$w_{ML}$ minimizes the distance between t and its orthogonal projection on S, i.e. y.

# Sequential Learning

Optional subtitle

Big Data Problem? Lots of training data. Hard to load them all together.

Data items considered one at a time (a.k.a. online learning); use stochastic (sequential) gradient descent:

$$\begin{aligned}\mathbf{w}^{(\tau+1)} &= \mathbf{w}^{(\tau)} - \eta \nabla E_n \\ &= \mathbf{w}^{(\tau)} + \eta(t_n - \mathbf{w}^{(\tau)\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n))\boldsymbol{\phi}(\mathbf{x}_n).\end{aligned}$$

This is known as the *least-mean-squares (LMS) algorithm*.

# Regularized Least Squares (1)

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

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

$\lambda$ is called the regularization coefficient.

which is minimized by

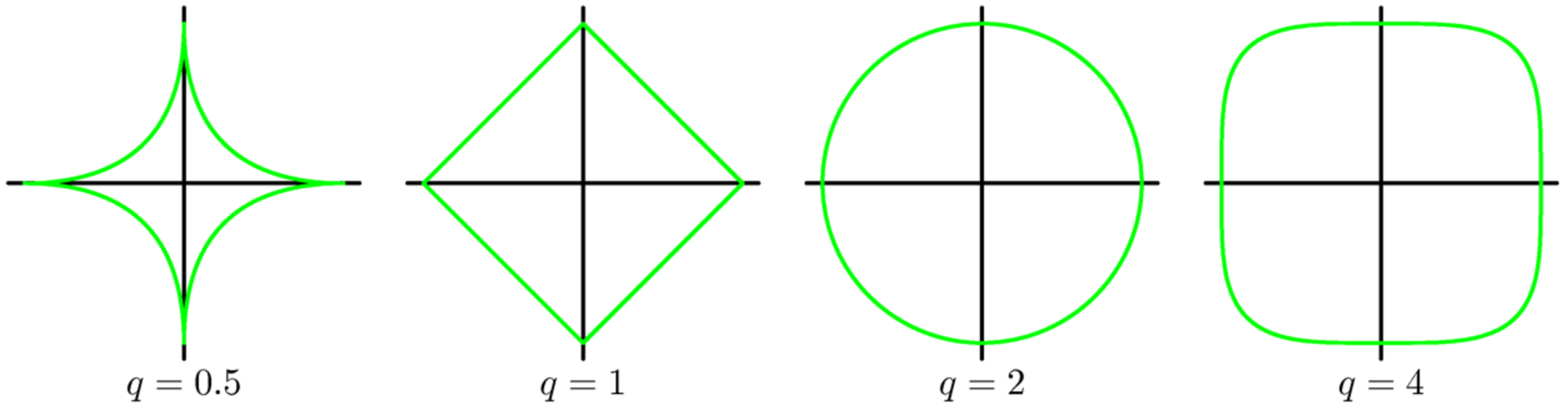
$$\mathbf{w} = \left(\lambda\mathbf{I} + \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}.$$

# Regularized Least Squares (2)

Optional subtitle

With a more general regularizer, we have

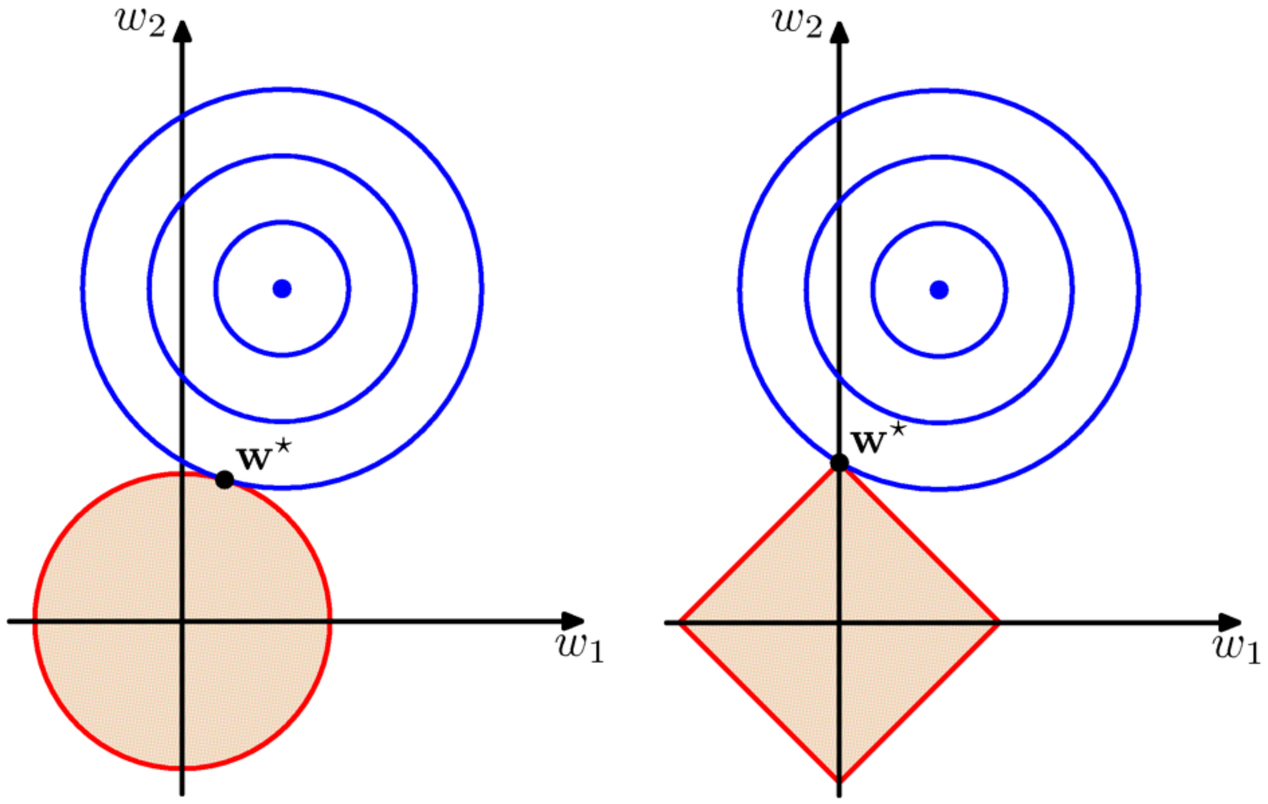
$$\frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n)\}^2 + \frac{\lambda}{2}\sum_{j=1}^{M}|w_j|^q$$




Lasso Quadratic

# Regularized Least Squares (3)

Lasso tends to generate sparser solutions than a quadratic regularizer.



# The Bias-Variance Decomposition (1)

Optional subtitle

Recall 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)

Optional subtitle

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

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

# The Bias-Variance Decomposition (3)

Optional subtitle

Taking the expectation over D yields

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

# The Bias-Variance Decomposition (4)

Optional subtitle

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 Trade-off

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