Supervised Learning: Linear Regression and Classification

Yuan Yao

Department of Mathematics
Hong Kong University of Science and Technology

Most of the materials here are from Chapter 3-4 of Introduction to Statistical Learning by Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani.

Overview of Supervised Learning

- The Bias-Variance Trade-Off of Prediction Error

Linear Regression

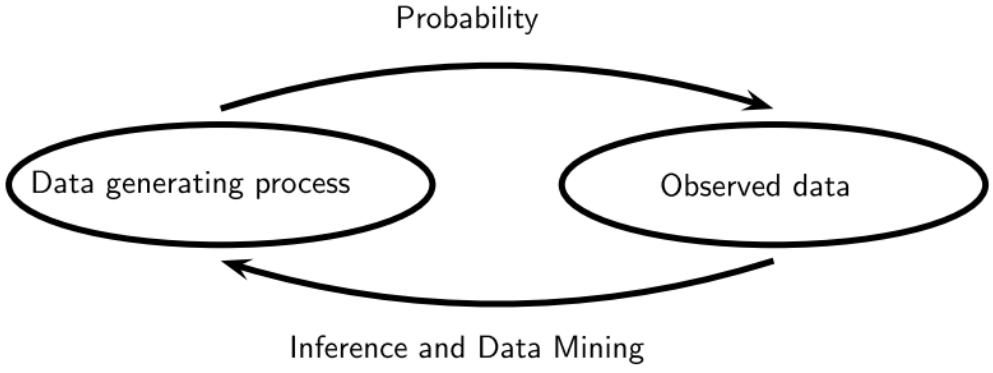
- The least squares estimation
- The statistical properties of the least squares estimates.

Linear Classification

- Logistic Regression.
- Linear Discriminant Analysis
- The ROC curve

Outline

Probability vs. Statistical Machine Learning



Forward problem: Probability is a language to quantify uncertainty.

Inverse Problem: Statistics or Machine Learning

Statistics/Data Mining Dictionary

Statisticians and computer scientists often use different language for the same thing. Here is a dictionary that the reader may want to return to throughout the course.

Statistics	Computer Science	Meaning
estimation	learning	using data to estimate an unknown quantity
classification	supervised learning	predicting a discrete Y from X
clustering	unsupervised learning	putting data into groups
data	training sample	$(X_1, Y_1), \ldots, (X_n, Y_n)$
covariates	features	the X_i's
classifier	hypothesis	a map from covariates to outcomes
hypothesis	—	subset of a parameter space Θ
confidence interval	—	interval that contains an unknown quantity with given frequency
directed acyclic graph	Bayes net	multivariate distribution with given conditional independence relations
Bayesian inference	Bayesian inference	statistical methods for using data to update beliefs
frequentist inference	—	statistical methods with guaranteed frequency behavior
large deviation bounds	PAC learning	uniform bounds on probability of errors

Figure: Larry Wasserman's classification of statistical learning vs. machine learning in Computer Science

Supervised vs. Unsupervised Learning

- Supervised Learning
  - **Data**: (x, y), where x is data and y is label
  - **Goal**: learn a function to map $f : x \rightarrow y$
  - **Examples**: classification (object detection, segmentation, image captioning), regression, etc.
  - **Golden standard**: *prediction*!
- Unsupervised Learning
  - **Data**: x, just data and no labels!
  - **Goal**: learn some hidden structure of data x
  - **Examples**: clustering (topology), dimensionality reduction (geometry), density estimation (GAN), etc.
  - **Golden standard**: Non!
- "Self-supervised Learning": cloze task in language models (autoregressive predictions, etc.)

Statistical (supervised) learning

- Suppose that we observe a quantitative response Y and p different predictors, $X_1, X_2, \ldots, X_p$. We assume that there is some mapping $f : X = (X_1, X_2, \ldots, X_p) \to Y$, written as

$$Y = f(X) + \epsilon, \tag{1}$$

  where
  - f is some fixed but unknown function to be estimated;
  - ϵ is a random *error* term, which is independent of X and has mean zero;
  - There are two main reasons that we may wish to estimate f: *prediction* and *inference*.

Prediction vs. Inference

- *Prediction* aims to minimize the gap between true value Y and predicted value $\hat{Y} = \hat{f}(X)$, usually measured by loss

$$\mathbb{E}_{(X,Y)}\mathcal{L}(Y, \hat{f}(X))$$

  - The estimation $\hat{f}$, as a mapping from X to Y with unknown parameters (e.g. linear coefficients, support vector machines, neural networks), is a random variable depending on the random data $\mathcal{D}$ for training.

Prediction vs. Inference

- *Inference* aims to estimate f and its properties, but the goal is not necessarily to make predictions for Y, e.g.
  - **Variable selection**: which predictors are associated with the response?
  - **Model selection**: can the relationship between Y and each predictor be adequately summarized using a linear equation, more complicated ones?
  - **Uncertainty**: how much is the uncertainty of your prediction or estimation given finite information?

Expected Prediction Error

- Given an estimate $\hat{f}$ and a set of predictors X, we can predict Y using

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

- Assume for a moment that both $\hat{f}$ and X are fixed. In regression setting,

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

where $\mathbb{E}(Y - \hat{Y})^2$ represents the expected squared error between the predicted and actual value of Y, and $Var(\epsilon)$ represents the variance associated with the error term ϵ. An optimal estimate is to minimize the reducible error.

Reducible vs. Irreducible Error

$$\underbrace{[f(X) - \hat{f}(X)]^2}_{\textbf{Reducible}}$$

- **Reducible error**: $\hat{f}$ will not be a perfect estimate for f, and this inaccuracy will introduce some error. This error is reducible because we can potentially improve the accuracy of $\hat{f}$ by using the most appropriate statistical learning technique to estimate f.
- For example, one may choose different model families for $\hat{f}$:
  - linear models
  - nonlinear models: splines, trees, support vector machines, neural networks

Reducible vs. Irreducible Error

$$\underbrace{Var(\epsilon)}_{\textbf{Irreducible}}$$

- **Irreducible error**: Even if it were possible to form a perfect estimate for f, so that our estimated response took the form $\hat{Y} = f(X)$, our prediction would still have some error in it! This is because Y is also a function of ϵ, which, by definition, cannot be predicted using X. The quantity ϵ may contain unmeasured variables that are useful in predicting Y: since we don't measure them, f cannot use them for its prediction. Therefore, variability associated with ϵ also affects the accuracy of our predictions. This is known as the irreducible error, because no matter how well we estimate f, we cannot reduce the error introduced by uncertainty of ϵ.

How to estimate f?

Depending on whether our ultimate goal is prediction, inference, or a combination of the two, different methods for estimating f may be appropriate.

- Assume that we have observed a set of n different data points. These observations are called the *training data* $\mathcal{D} = \{(x_i, y_i) : i = 1, \ldots, n\}$ because we will use these observations to train our method how to estimate f.
- Our goal is to apply a statistical learning method to the training data in order to estimate the unknown function f. In other words, we want to find a function $\hat{f}$ such that $Y \approx \hat{f}(X)$ for any observation (X, Y).
- Most statistical learning methods for this task can be characterized as either *parametric* or *non-parametric*.

Parametric Methods

Parametric methods are model-based approach, as it reduces the problem of estimating f down to one of estimating a set of parameters.

- First, we make an assumption about the functional form, or shape, of f.
  - For example, one very simple assumption is that f is linear in X:
$$f(X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p. \tag{3}$$
  - Once we have assumed that f is linear, the problem of estimating f is greatly simplified. Instead of having to estimate an entirely arbitrary p-dimensional function $f(X)$, one only needs to estimate the $p + 1$ coefficients $\beta_0, \beta_1, \ldots, \beta_p$.
- After a model has been selected, we need a procedure that uses the training data to fit or train the model, e.g. the most common approach as the least square method.

- **Pro**: Assuming a parametric form for f simplifies the problem of estimating f because it is generally much easier to estimate a set of parameters, such as $\beta_0, \beta_1, \ldots, \beta_p$ in the linear model (3), than it is to fit an entirely arbitrary function f.
- **Con**: The potential disadvantage of a parametric approach is that the model we choose will usually not match the true unknown form of f. If the chosen model is too far from the true f,then our estimate will be poor.
- We can try to address this problem by choosing *flexible* models that can fit many different possible functional forms flexible for f. But in general, fitting a more flexible model requires estimating a greater number of parameters. These more complex models can lead to a phenomenon known as **overfitting** the data, which essentially means they follow the errors, or noise, too closely.

Non-parametric Methods

- Non-parametric methods 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.
- Such approaches can have a major advantage over parametric approaches: by avoiding the assumption of a particular functional form for f, they have the potential to accurately fit a wider range of possible shapes for f.
  - Any parametric approach brings with it the possibility that the functional form used to estimate f is very different from the true f, in which case the resulting model will not fit the data well.
  - In contrast, non-parametric approaches completely avoid this danger, since essentially no assumption about the form of f is made.

Non-parametric Methods

- But non-parametric approaches do suffer from a major **disadvantage**: since 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.

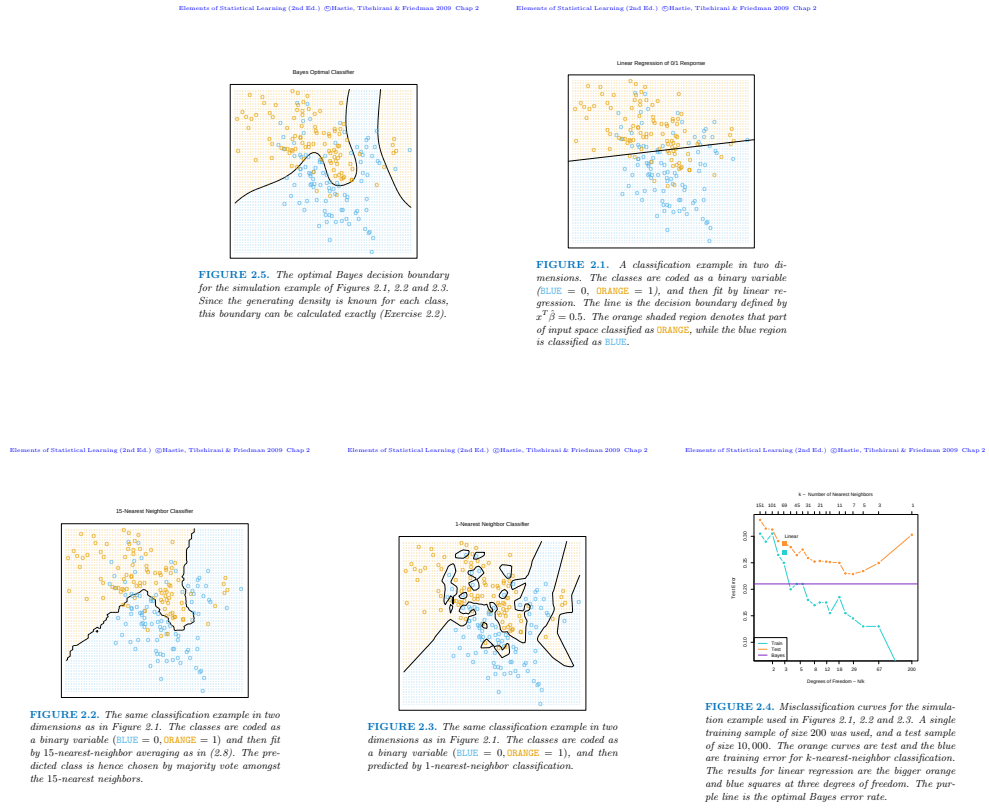


FIGURE 2.5. *The optimal Bayes decision boundary for the simulation example of Figures 2.1, 2.2 and 2.3. Since the generating density is known for each class, this boundary can be calculated exactly (Exercise 2.2).*

FIGURE 2.1. *A classification example in two dimensions. The classes are coded as a binary variable (BLUE = 0, ORANGE = 1), and then fit by linear regression. The line is the decision boundary defined by $x^T\hat{\beta} = 0.5$. The orange shaded region denotes that part of input space classified as ORANGE, while the blue region is classified as BLUE.*

FIGURE 2.2. *The same classification example in two dimensions as in Figure 2.1. The classes are coded as a binary variable (BLUE = 0, ORANGE = 1) and then fit by 15-nearest-neighbor averaging as in (2.8). The predicted class is hence chosen by majority vote amongst the 15-nearest neighbors.*

FIGURE 2.3. *The same classification example in two dimensions as in Figure 2.1. The classes are coded as a binary variable (BLUE = 0, ORANGE = 1), and then predicted by 1-nearest-neighbor classification.*

FIGURE 2.4. *Misclassification curves for the simulation example used in Figures 2.1, 2.2 and 2.3. A single training sample of size 200 was used, and a test sample of size 10,000. The orange curves are test and the blue are training error for k-nearest-neighbor classification. The results for linear regression are the bigger orange and blue squares at three degrees of freedom. The purple line is the optimal Bayes error rate.*

Figure: An illustrative example. Upper Left: A simulated binary classification data with its true generative model (Bayes decision boundary). Upper Right: A fitted Linear model (parametric). Lower Left: Two fitted k-nn models (non-parametric). Lower Right: Test errors.

"No free lunch in statistics"

- 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.
- Hence it is an important task to decide for any given set of data which method produces the best results. Selecting the best approach can be one of the most challenging parts of performing statistical learning in practice.
- In this section, we discuss some of the most important concepts that arise in selecting a statistical learning procedure for a specific data set.

The Bias-Variance Trade-Off

- Let $f(X)$ be the true function which we aim at estimating from a training data set $\mathcal{D} = \{(x_i, y_i) : i = 1, \ldots, n\}$.
- Let $\hat{f}(X; \mathcal{D})$ be the estimated function from the training data set $\mathcal{D}$.
- **Fisher's view**: data set $\mathcal{D}$ is a **random selection** from the set of all possible measurements which form the true distribution!
- Expected prediction error

$$\min_{\hat{f}} \mathbb{E}_{\mathcal{D}} \left[f(X) - \hat{f}(X; \mathcal{D}) \right]^2, \tag{4}$$

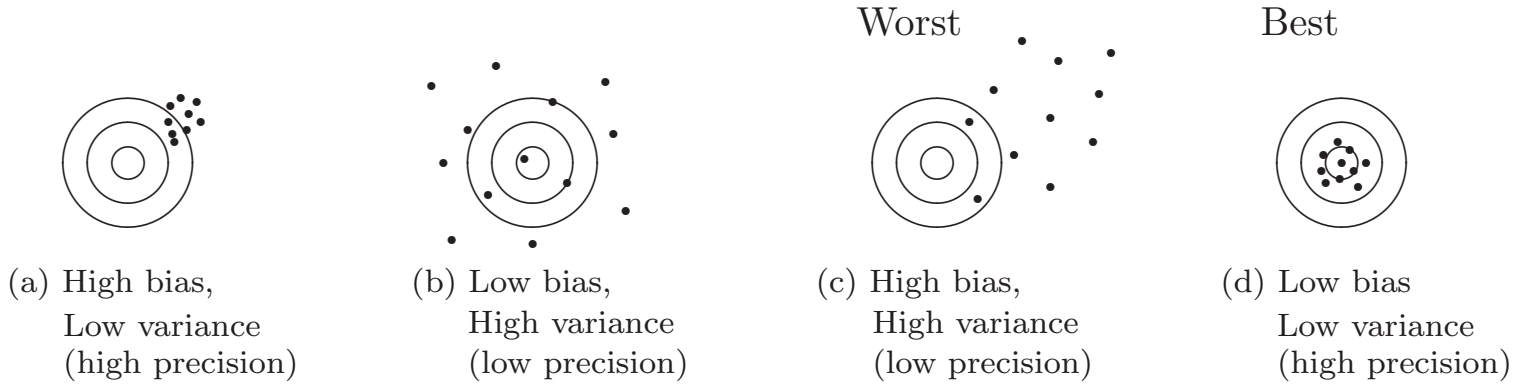
where randomness caused by **random selection** has been taken into account.

- Add and subtract $\mathbb{E}_{\mathcal{D}}(\hat{f}(X;\mathcal{D}))$ inside the braces, then expand,

$$\begin{aligned}
&\left[f(X) - \hat{f}(X;\mathcal{D})\right]^2 \\
&= \left[f(X) - \mathbb{E}_{\mathcal{D}}(\hat{f}(X;\mathcal{D})) + \mathbb{E}_{\mathcal{D}}(\hat{f}(X;\mathcal{D})) - \hat{f}(X;\mathcal{D})\right]^2 \\
&= \left[f(X) - \mathbb{E}_{\mathcal{D}}(\hat{f}(X;\mathcal{D}))\right]^2 + \left[\mathbb{E}_{\mathcal{D}}(\hat{f}(X;\mathcal{D})) - \hat{f}(X;\mathcal{D})\right]^2 \\
&\quad + 2\left[f(X) - \mathbb{E}_{\mathcal{D}}[\hat{f}(X;\mathcal{D})]\right]\left[\mathbb{E}_{\mathcal{D}}[\hat{f}(X;\mathcal{D})] - \hat{f}(X;\mathcal{D})\right].
\end{aligned}$$

- Take the expectation with respect to $\mathcal{D}$,

$$\begin{aligned}
&\mathbb{E}_{\mathcal{D}}\left[f(X) - \hat{f}(X;\mathcal{D})\right]^2 \\
&= \underbrace{\left[f(X) - \mathbb{E}_{\mathcal{D}}(\hat{f}(X;\mathcal{D}))\right]^2}_{\textit{Bias}^2} + \underbrace{\mathbb{E}_{\mathcal{D}}\left[\left[\mathbb{E}_{\mathcal{D}}(\hat{f}(X;\mathcal{D})) - \hat{f}(X;\mathcal{D})\right]^2\right]}_{\textit{Variance}}
\end{aligned}$$



(a) High bias, Low variance (high precision)

(b) Low bias, High variance (low precision)

(c) High bias, High variance (low precision)

(d) Low bias Low variance (high precision)

- **Bias** refers to the error that is introduced by approximating a real-life problem, which may be extremely complicated, by a much simpler model.
- **Variance** refers to the amount by which $\hat{f}_{\mathcal{D}}$ would change if we estimated it using a different training data set $\mathcal{D}$.
- **Bias and variance trade-off**: The optimal predictive capability is the one that leads to balance between bias and variance.

Bias-variance tradeoff

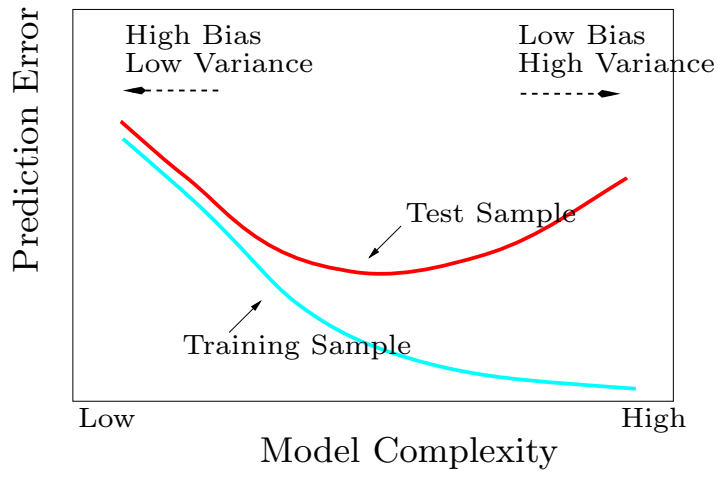
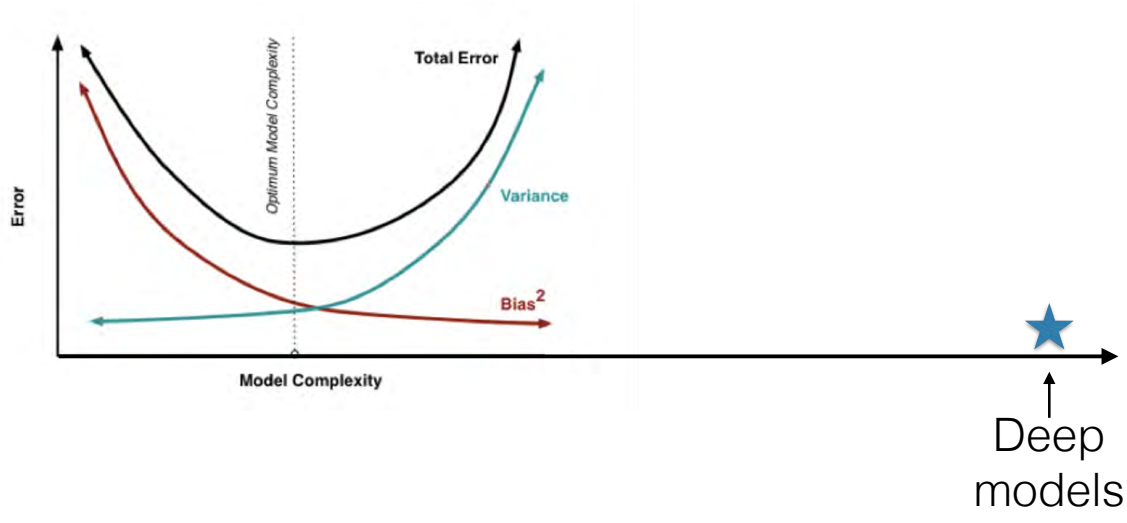


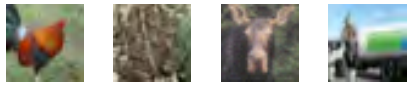
FIGURE 2.11. *Test and training error as a function of model complexity.*

Deep learning: Bias-variance tradeoff?



Models where p>20n are common

Why Big Models Generalize Well?



CIFAR10

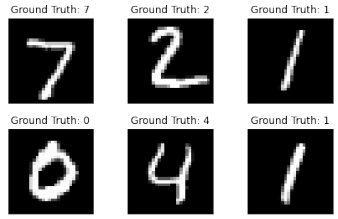
n=50,000
d=3,072
k=10

What happens when I turn off the regularizers?

Model	parameters	p/n	Train ***loss***	Test ***error***
CudaConvNet	145,578	2.9	0	23%
CudaConvNet (with regularization)	145,578	2.9	0.34	18%
MicroInception	1,649,402	33	0	14%
ResNet	2,401,440	48	0	13%

Figure: Why overparameterized deep neural networks do not overfit? Ben Recht, FoCM 2017.

MNIST experiments with Neural Networks



- MNIST: $(x_i, y_i) \in \mathbb{R}^{784 \times [10]}$, $i \in [50,000]$.
- Two-layers neural networks f_N :

$$f_N(x;\theta) = \sum_{j=1}^{N} a_j \sigma\left(\langle w_j, x\rangle\right).$$

- Square loss without regularization.
- Find a local minimizer, report training and test error.
- Perform a sequence of experiments for different N.
- Plot training and test error vs N.

Increasing # parameters

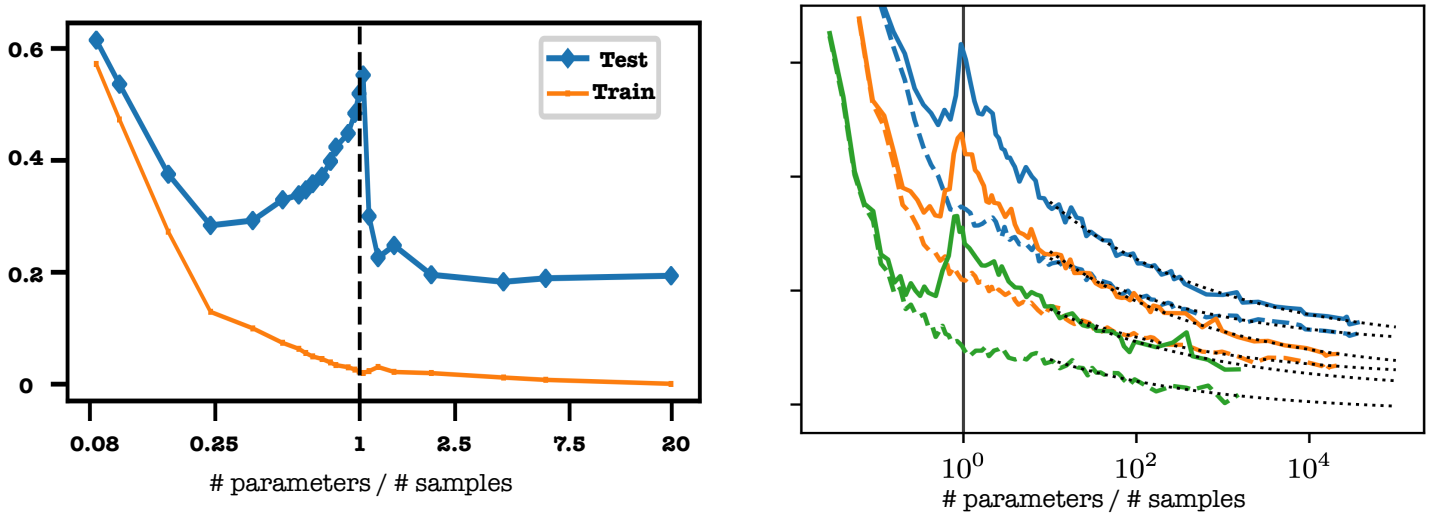


Figure: Experiments on MNIST. Left: [Belkin, Hsu, Ma, Mandal, 2018]. Right: [Spigler, Geiger, Ascoli, Sagun, Biroli, Wyart, 2018].

Similar phenomenon appeared in the literature [LeCun, Kanter, and Solla, 1991], [Krogh and Hertz, 1992], [Opper and Kinzel, 1995], [Neyshabur, Tomioka, Srebro, 2014], [Advani and Saxe, 2017].

Double Descent for Overparameterized Models?

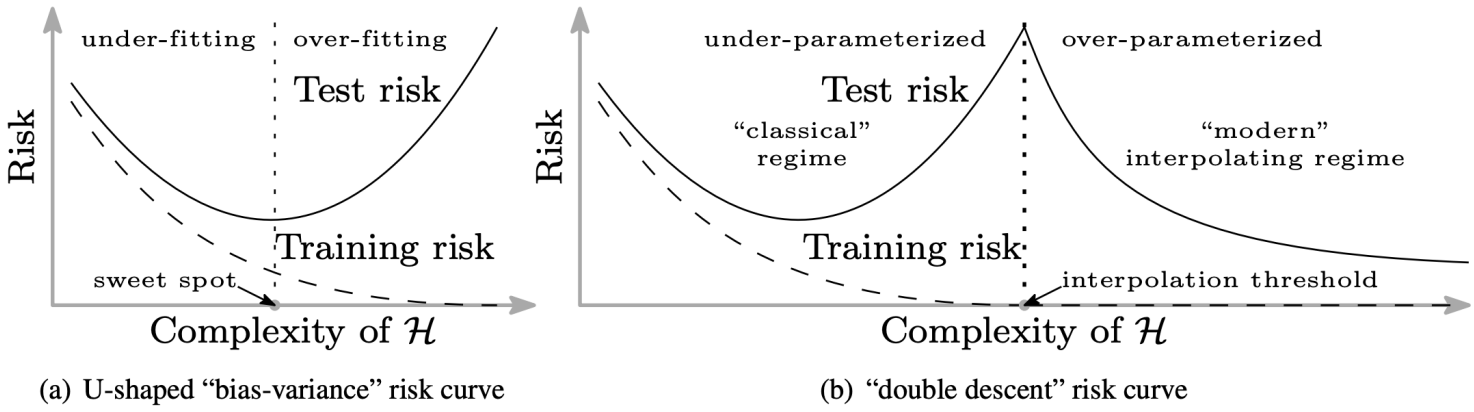


Figure: A cartoon by [Belkin, Hsu, Ma, Mandal, 2018].

Figure: Double descent: 1) Peak at the interpolation threshold; 2) monotone decreasing in the overparameterized region; 3) Global minimum when the number of parameters is infinity.

All models are wrong, but some are useful.

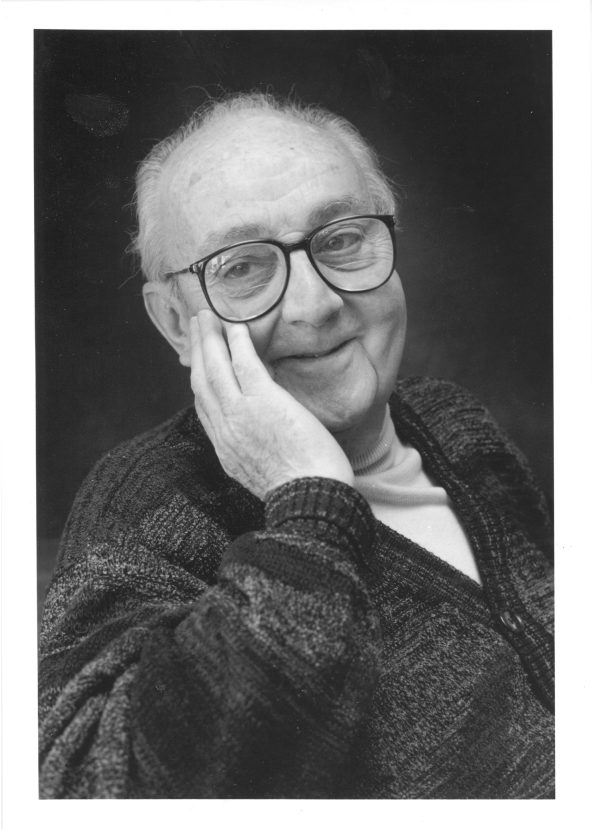


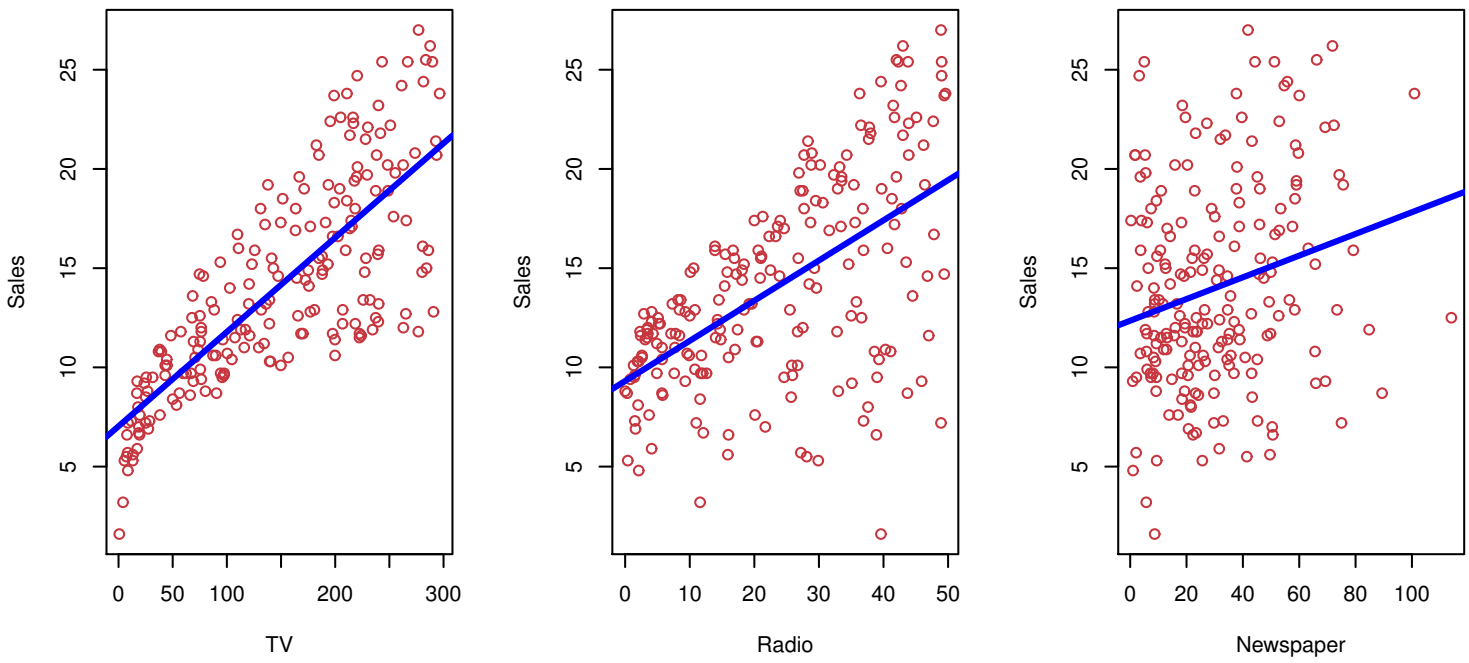
Figure: George Box: "Essentially, all models are wrong, but some are useful."

Outline

Example: Advertising data

The data contains 200 observations.
Sample size: $n = 200$.
Sales: y_i, $i = 1..., n$.
TV (bugdets): $x_{i1}, i = 1, ..., n$.
Radio (budgets): $x_{i2}, i = 1, ..., n$.
Newspaper (budgets): $x_{i3}, i = 1, ..., n$.
Dimensionality: $p = 3$.

Example: Advertising data



Linear models formulation

- Consider linear regression model:

$$y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \epsilon_i, \qquad i = 1, ..., n \qquad (2.1)$$

  where $y_i, x_i = (x_{i1}, ..., x_{ip})$ are the i-th observation of the response and covariates.
- Responses are sometimes called dependent variables or outputs;
- Covariates called independent variables, or predictors or features or inputs or regressors.
- Noise ϵ_i is independent, of zero mean, and fixed but unknown variance, e.g. Gaussian noise $\mathcal{N}(0, \sigma^2)$

Example: Advertising data

Now, we consider three covariates: TV, radio and newspapers.
The number of covariates (predictors, or features): $p = 3$.
The linear regression model is

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i3} + \epsilon_i, \qquad i = 1, ..., n$$

Estimating the coefficient by the least squares

Minimizing the sum of squares of error (Gauss'1795) [essentially as $-\log$ **likelihood** of normal distribution]:

$$\min_{\beta_0,\ldots,\beta_3} \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_{i1} - \beta_2 x_{i2} - \beta_3 x_{i3})^2.$$

Figure: Carl Friedrich Gauss

Notations

With slight abuse of notation, in this chapter, we use

$$\begin{aligned}\mathbf{X} &= \begin{pmatrix} 1 & x_{11} & x_{12} & \dots & x_{1p} \\ 1 & x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{1} \vdots \mathbf{x}_1 \vdots \mathbf{x}_2 \vdots \cdots \vdots \mathbf{x}_p \end{pmatrix}.\end{aligned}$$

Here a column of ones, $\mathbf{1}$, is added, which corresponds to the intercept β_0. Then $\mathbf{X}$ is a n by $p+1$ matrix.

Recall that

$$\mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \quad \beta = \begin{pmatrix} \beta_0 \\ \vdots \\ \beta_p \end{pmatrix}, \quad \epsilon = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{pmatrix}, \quad \mathbf{x}_j = \begin{pmatrix} x_{1j} \\ \vdots \\ x_{nj} \end{pmatrix}$$

The least squares criterion

The least squares criterion is try to minimize the sum of squares:

$$\sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_{i1} - \cdots - \beta_p x_{ip})^2.$$

Using matrix algebra, the above sum of squares is

$$\|\mathbf{y} - \mathbf{X}\beta\|^2 = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta).$$

The LSE, fitted values and residuals

By some linear algebra calcuation, the least squares estimator of β is then

$$\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}.$$

Then

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\beta}$$

is called the fitted values; viewed as the predicted values of the responses based on the linear model.

$$\hat{\epsilon} = \mathbf{y} - \hat{\mathbf{y}}$$

are called residuals. The sum of squares of these residuals

$$RSS = \sum_{i=1}^{n} \hat{\epsilon}_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \|\mathbf{y} - \hat{\mathbf{y}}\|^2.$$

Orthogonality

- The residual $\hat{\epsilon}$ is orthogonal to all columns of $\mathbf{X}$, i.e, all $\mathbf{1}, \mathbf{x}_1, ..., \mathbf{x}_p$. This can be seen by

$$\begin{aligned} & \mathbf{X}^T \hat{\epsilon} = \mathbf{X}^T \mathbf{y} - \mathbf{X}^T \mathbf{X} \hat{\beta} \\ = \quad & \mathbf{X}^T \mathbf{y} - \mathbf{X}^\mathbf{T} \mathbf{H} \mathbf{y} = 0. \end{aligned}$$

  where $\mathbf{H} = \mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T = \mathbf{H}^2$ is the projection.
- The residual vector $\hat{\epsilon}$ is orthogonal to the hyperplane formed by vectors $\mathbf{1}, \mathbf{x}_1, ..., \mathbf{x}_p$ in n dimensional real space.

The least squares projection

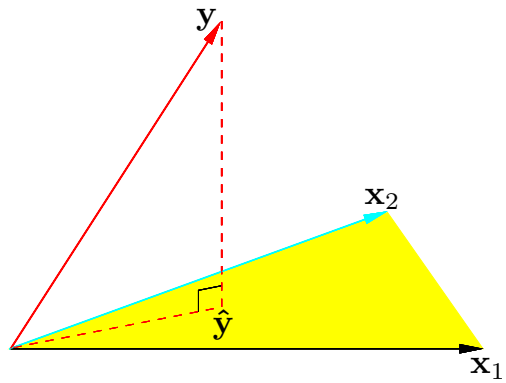


FIGURE 3.2. *The N-dimensional geometry of least squares regression with two predictors. The outcome vector $\mathbf{y}$ is orthogonally projected onto the hyperplane spanned by the input vectors $\mathbf{x}_1$ and $\mathbf{x}_2$. The projection $\hat{\mathbf{y}}$ represents the vector of the least squares predictions*

Gauss-Markov Theorem

Theorem (Gauss-Markov Theorem). Assume the noise ϵ_i is of

- mean zero, i.e. $\mathbb{E}(\epsilon_i) = 0$,
- homoscedastic in variance, i.e. $\mathbb{E}(\epsilon_i^2) = \sigma^2 < \infty$),
- uncorrelated, i.e. $\mathbb{E}(\epsilon_i \epsilon_j) = 0$).

Then the least square estimate is the *Best Linear Unbiased Estimate (BLUE)*, i.e. among all linear unbiased estimates, the least squares estimate has the smallest variance, thus smallest mean squared error.

- Note: no Gaussian distribution assumption on noise!


Figure: Left: Carl Friedrich Gauss; Right: Andrey Andreyevich Markov

Proof of Gauss-Markov Theorem

- Let $\mathbf{A} = (\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_n)$. Then, linear unbiased estimate of β is

$$\mathbf{Ay}$$

  with
  - mean $\mathbb{E}[\mathbf{Ay}] = \mathbf{AX}\beta = \beta$ by the unbiasedness,
  - and variance matrix $\mathbf{AA}^T$.
- Write $\mathbf{A} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T + \mathbf{D}$. Then, $\mathbf{DX} = 0$.
- Then,

$$\mathbf{AA}^T = (\mathbf{X}^T\mathbf{X})^{-1} + \mathbf{DD}^T \succeq (\mathbf{X}^T\mathbf{X})^{-1}.$$

  Here the inequality is for symmetric matrices, i.e., $\mathbf{A} \succeq \mathbf{B}$ is defined as $\mathbf{A} - \mathbf{B}$ is nonnegative definite. □

Result of the estimation

TABLE 3.9. (from ISLR) The advertising data: coefficients of the LSE for the regression on number of units sold on TV, radio and newspaper advertising budgets.

	Coefficient	Std.error	t-statistic	p-value
Intercept	2.939	0.3119	9.42	< 0.0001
TV	0.046	0.0014	32.81	< 0.0001
radio	0.189	0.0086	21.89	< 0.0001
newspaper	-0.001	0.0059	-0.18	0.8599

Statistical properties of LSE

Assume Gaussian noise $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$,

$$\hat{\beta} \sim \mathcal{N}(\beta, \sigma^2(\mathbf{X}^T\mathbf{X})^{-1});$$

$$\text{RSS} = \sum_{i=1}^{n} \hat{\epsilon}_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \sim \sigma^2 \chi^2_{n-p-1}$$

$\hat{\beta}$ and RSS are independent

$s^2 = \text{RSS}/(n-p-1)$ unbiased estimate of σ^2

$$\frac{\hat{\beta}_j - \beta_j}{s\sqrt{c_{jj}}} \sim t_{n-p-1}$$

$$\frac{(\hat{\beta}-\beta)^T(\mathbf{X}^T\mathbf{X})(\hat{\beta}-\beta)/p}{s^2} \sim F_{p+1,n-p-1}$$

where $c_{00}, c_{11}, ..., c_{pp}$ are the diagonal elements of $(\mathbf{X}^T\mathbf{X})^{-1}$.

Confidence intervals

For example,

$$\hat{\beta}_j \pm t_{n-p-1}(\alpha/2) s \sqrt{c_{jj}}$$

is a confidence interval for β_j at confidence level $1 - \alpha$. Here $t_{n-p-1}(\alpha/2)$ is the $1 - \alpha/2$ percentile of the t-distribution with degree of freedom $n - p - 1$.

- the j-th p-value tells the probability of seeing $\hat{\beta}_j$ with assumption $\beta_j = 0$, useful for variable selection or elimination

Prediction interval

For a given value of input $\mathbf{x}$ which is a $p+1$ vector (the first component is constant 1), its mean response is $\beta^T\mathbf{x}$. The confidence interval for this mean response, also called prediction interval, is

$$\hat{\beta}^T\mathbf{x} \pm t_{n-p-1}(\alpha/2)s\sqrt{\mathbf{x}^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{x}}$$

- The confidence interval for β_j is a special case of the above formula by taking $\mathbf{x}$ as a vector that all zero except the $(j+1)$ entry corresponding β_j. (Because of β_0, β_j is at the $j+1$th position of $\hat{\beta}$.)

Variable selection

- We may be concerned with a subset of the p variables are irrelevant with the response.
- Let the subset be denoted as $A = \{i_1, ..., i_r\}$, where $r \leq p$. Then, the null hypothesis is

$$H_0 : \beta_{i_1} = \beta_{i_2} = \cdots = \beta_{i_r} = 0,$$

which again is equivalent to

$$H_0 : \mathbf{P}(\mathbf{y}) \in \mathcal{L}(A^c),$$

where $\mathcal{L}(A)$ is the linear space in R^n spanned by $\mathbf{x}_{i_1}, ..., \mathbf{x}_{i_r}$, which is of r dimension.

The F-test

$$F = \frac{\|\hat{\mathbf{y}} - \hat{\mathbf{y}}_0\|^2/(p+1-r)}{\|\mathbf{y} - \hat{\mathbf{y}}\|^2/(n-p-1)} \sim F_{p+1-r,n-p-r}$$

This F-statistic is used to test the hypothesis that $H_0 : \mathbf{P}(\mathbf{y}) \in \mathcal{L}(A^c)$, against the alternative H_a : otherwise.

- The smaller $Prob(F)$, the more significant of H_a
- The commonly considered hypothesis, $H_0 : \beta_1 = \cdots = \beta_p = 0$ can be formulated as $H_0 : \mathbf{P}(\mathbf{y}) \in \mathcal{L}(\mathbf{1})$, where $\mathcal{L}(\mathbf{1})$ represent the linear space of a single vector $\mathbf{1}$.

Outline

Examples: The default data

- Simulated data: 10000 individuals.
- Three predictors: income, balance (monthly), and student (Yes or No)
- One output: Default (Yes or No).
- Judge if one fails to pay the credit card debt (default).

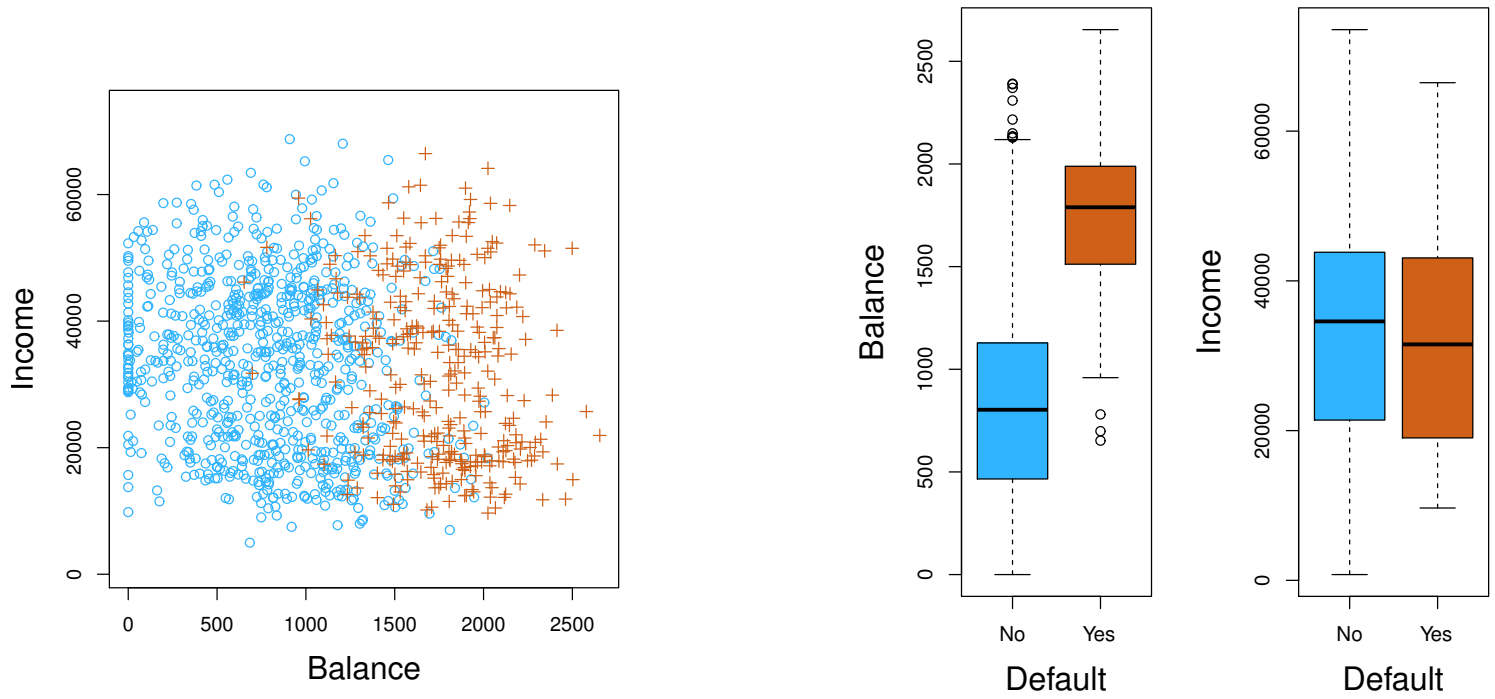


Figure: FIGURE 4.1. The Default data set. Left: The annual incomes and monthly credit card balances of a number of individuals. The individuals who defaulted on their credit card payments are shown in orange, and those who did not are shown in blue. Strong relation between balance and default while a weaker relation between income and default.

The special case of binary response

- Consider the output is binary: two class,
- Code the response into 0 and 1 and apply linear regression produce the same result as linear discriminant anlaysis.
- Not the case for output with more than two classes.

Example: The default data

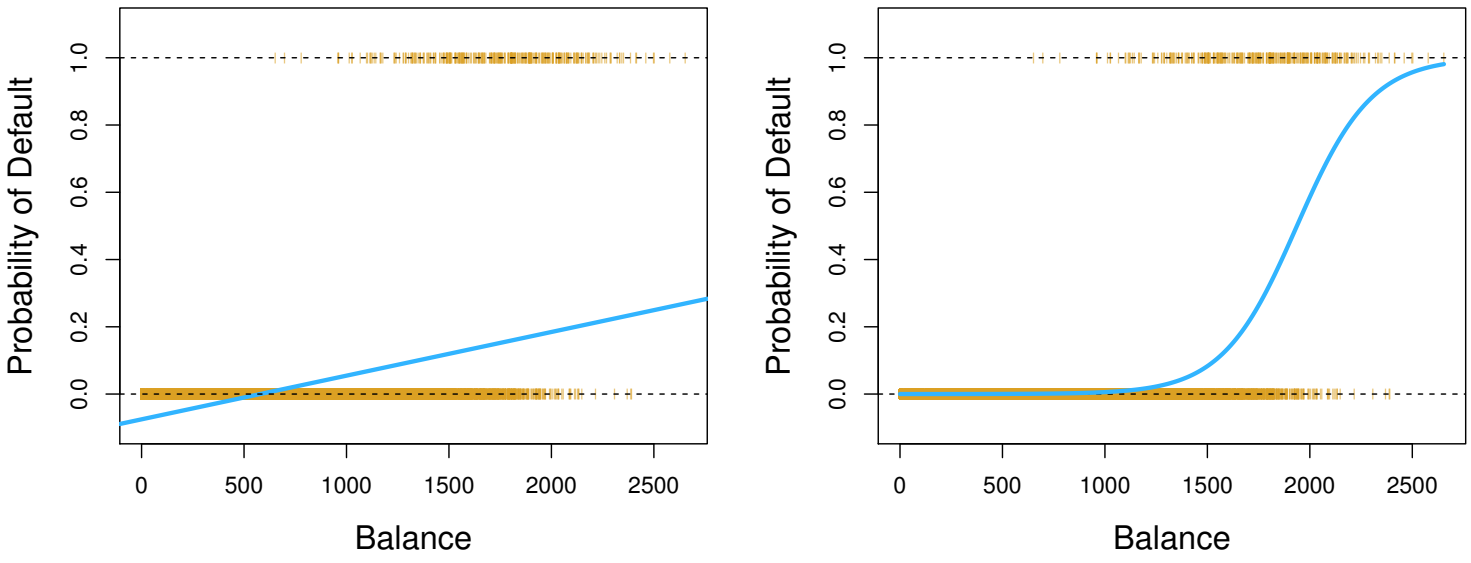


Figure: Left: linear regresion; Right: logistic regression

The setup for binary output

- The training data: $(\mathbf{x}_i, y_i)$: $i = 1, ..., n$.
- $y_i = 1$ for class 1 and $y_i = 0$ for class 0.
- $\mathbf{x}_i = (1, x_{i1}, ..., x_{ip})$ are $p + 1$ vectors with actually p inputs.
- If instead consider linear regression model is

$$y_i = \beta^T \mathbf{x}_i + \epsilon_i$$

  β can be estimated by the least squares, and $\hat{\beta}^T \mathbf{x}_i$ is the predictor of y_i.
- Key idea: should focus on predicting the probability of the classes.
- Using $P(y = 1|\mathbf{x}) = \beta^T \mathbf{x}$ is not appropriate.

The logistic regression model

- Assume

$$P(y = 1|\mathbf{x}) = \frac{1}{1 + \exp(-\beta^T \mathbf{x})}$$

  As a result,

$$P(y = 0|\mathbf{x}) = 1 - P(y = 1|\mathbf{x}) = \frac{1}{1 + \exp(\beta^T \mathbf{x})}$$

- $$\log(\frac{P(y = 1|\mathbf{x})}{P(y = 0|\mathbf{x})}) = \beta^T \mathbf{x}$$

  This is called log-odds or logit. And

$$\frac{P(y = 1|\mathbf{x})}{P(y = 0|\mathbf{x})}$$

  is called odds.
- Interpretation: one unit increase in variable x_j, increases the log-odds of class 1 by β_j.

The maximum likelihood estimation

- Recall that, the likelihood is the joint probability function of joint density function of the data.
- Here, we have independent observations $(\mathbf{x}_i, y_i)$, $i = 1, ..., n$, each follow the (conditional) distribution

$$P(y_i = 1|\mathbf{x}_i) = \frac{1}{1 + \exp(-\beta^T \mathbf{x}_i)} = 1 - P(y_i = 0|\mathbf{x}_i).$$

- So, the joint probability function is

$$\prod_{i=1,...,n;y_i=1} p(y_i = 1|\mathbf{x}_i) \prod_{i=1,...,n;y_i=0} p(y_i = 0|\mathbf{x}_i)$$

which can be conveniently written as

$$\prod_{i=1}^{n} \frac{\exp(y_i \beta^T \mathbf{x}_i)}{1 + \exp(\beta^T \mathbf{x}_i)}.$$

The likelihood and log-likelihood

- The likelihood function is the same as the joint probability function, but viewed as a function of β.
- The log-likelihood function is

$$\sum_{i=1}^{n}[y_i\beta^T x_i - \log(1+\exp(\beta^T \mathbf{x}_i))]$$

- The maximizor is denoted as $\hat{\beta}$, which is the MLE of β based on logistic model.

Example: the default data with all three inputs: balance, income and student

TABLE 4.3. For the Default data, estimated coefficients of the logistic regression model that predicts the probability of default using balance, income, and student status. Student status is encoded as a dummy variable student[Yes], with a value of 1 for a student and a value of 0 for a non-student. In fitting this model, income was measured in thousands of dollars.

	Coefficient	Std.error	t-statistic	p-value
Intercept	-10.8690	0.4923	-22.08	< 0.0001
Balance	0.0057	0.0002	24.74	< 0.0001
Income	0.0030	0.0082	0.37	0.7115
Student[Yes]	-0.6468	0.2362	-2.74	0.0062

The Bayes Theorem

- Suppose there are K, denoted as $1, 2, ..., K$, for the output Y.
- X is the input of p-dimension. Both Y and X are random variables.
- Let $\pi_k = P(Y = k)$.
- Let $f_k(x) = f(x|Y = k)$ be the conditional density function of X given $Y = k$.
- Then, Bayes theorem[1] implies

$$p_k(x) = P(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{j=1}^{K} \pi_j f_j(x)}$$

- We classify a subject with input x into class k, if its $p_k(x)$ is the largest, for $k = 1, ..., K$.

[1] General Bayes theorem: for $A_i \cap A_j = \emptyset$, $P(\cup_i A_i) = 1$,

$$P(A_k|B) = \frac{P(B|A_k)P(A_k)}{\sum_{j=1}^{K} P(B|A_j)P(A_j)}.$$

Model Assumptions of LDA

X is p-dimensional. $Y = 1, ..., K$, totally K classes. Assume, for $k = 1, ..., K$,

$$X|Y = k \sim \mathcal{N}(\mu_k, \Sigma),$$

where μ_k is p-vector and Σ is p-by-p covariance matrix. i.e.,

$$f_k(x) = \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} \exp\Big(-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)\Big)$$

Note that we assumed the same Σ for all classes $k = 1, ..., K$.

Computing $p_k(x)$ for LDA

- We aim to maximize over k the following

$$p_k(x) = \frac{\pi_k \exp[(-1/2)(x-\mu_k)^T \Sigma^{-1}(x-\mu_k)]}{\sum_{l=1}^{K} \pi_l \exp[(-1/2)(x-\mu_l)^T \Sigma^{-1}(x-\mu_l)]}$$

- Comparing $p_k(x)$ is the same as comparing the numerator. Set the k-th score, linear in x,

$$\delta_k(x) = \mu_k^{-1}\Sigma^{-1}x - \frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k + \log \pi_k.$$

  where the Bayesian classifier is the k with the largest δ_k.

- A practical problem: parameter Σ and μ_j and π_j, $j = 1, ..., K$ are usually unknown?

Fisher's Linear Discriminant Analysis

Choose the class to maximize the following *linear* score function:

$$\max_k \hat{\delta}_k(x) = \hat{\mu}_k^T \hat{\Sigma}^{-1} x - \frac{1}{2} \hat{\mu}_k^T \hat{\Sigma}^{-1} \hat{\mu}_k + \log \hat{\pi}_k, \tag{5}$$

where given data $(x_i, y_i), i = 1, ..., n$,

- $\hat{\mu}_k$ is the sample mean of class k

$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} x_i;$$

- $\hat{\Sigma}$ is the pooled (overall) sample covariance

$$\hat{\Sigma} = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T,$$

- $\hat{\pi}_k = n_k/n$ is the sample proportion of class k where n_k is the number of subjects in class k

Example: the default data, a Rare Event problem

- Fit the LDA model to 10,000 training samples gives training error rate of **2.75%**. That is, 275 samples are misclassified.
- Is it good?
- But, actually only **3.33%** of the training samples default, meaning that, a naive classifier that classifies every sample as non-default only has 3.33% error rate.
- So, minimizing misclassification error is misleading here!
- Instead, let's look at the *confusion matrix.*

The Confusion matrix.

TABLE 4.4. A confusion matrix compares the LDA predictions to the true default statuses for the 10, 000 training observations in the Default data set. Elements on the diagonal of the matrix represent individuals whose default status were correctly predicted, while off-diagonal elements represent individuals that were misclassified. LDA made incorrect predictions for 23 individuals who did not default and for 252 individuals who did default.

		True default status		
		No	Yes	Total
Predicted	No	9644	252	9896
default status	Yes	23	81	104
	Total	9667	333	10000

Class-specific performance

- Overall misclassification error rate: $(252 + 23)/10000 = 2.75\%$.
- Error rate with default people: $252/333 = 75.7\%$, losing a lot!!
- Sensitivity (TPR): $1 - 75.7\% = 24.3\%$
- Error rate within people no-default (FPR): $23/9667 = 0.24\%$.
- Specificity (1-FPR) : $1 - 0.24\% = 99.8\%$

A modification

- The Bayes classifier classifies a subject into class k, if the posterior probability $p_k(x)$ is the largest.
- In this case with two classes (Yes/No), i.e., $K = 2$. Bayes classifier classifies into *default* class if

$$Pr(default = Yes|X = x) > 0.5.$$

- A modification is classifies into *default* class if

$$Pr(default = Yes|X = x) > 0.2.$$

"I would rather kill a thousand by mistake than let one go..."

The classification result of the modification

TABLE 4.5. A confusion matrix compares the LDA predictions to the true default statuses for the 10, 000 training observations in the Default data set, using a modified threshold value that predicts default for any individuals whose posterior default probability exceeds 20%.

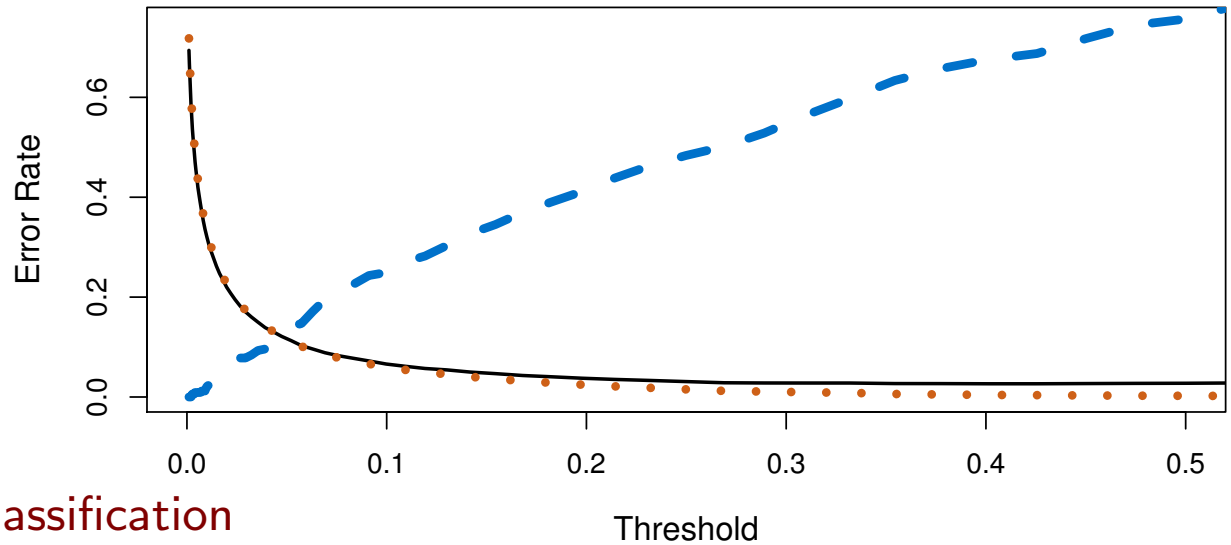
		True default status		
		No	Yes	Total
Predicted	No	9432	138	9570
default status	Yes	235	195	430
	Total	9667	333	10000

Class-specific performance

- Overall error rate: $(235 + 138)/10000 = 3.73\%$. (increased)
- Error rate with default people: $138/333 = 41.4\%$ (lower after modification)
- Sensitivity (TPR): $1 - 41.4\% = \mathbf{58.6\%}$ (Bayes 24.3%)
- Error rate within people no-default (FPR): $235/9667 = 2.43\%$. (increased from 0.24%)
- Specificity: $1 - 2.43\% = 97.57\%$ (Bayes $\mathbf{99.8\%}$)
- Identification of defaulter (sensitivity) is more important to credit card company!
- This modification may be helpful to the company. A tradeoff of specificity for sensitivity.

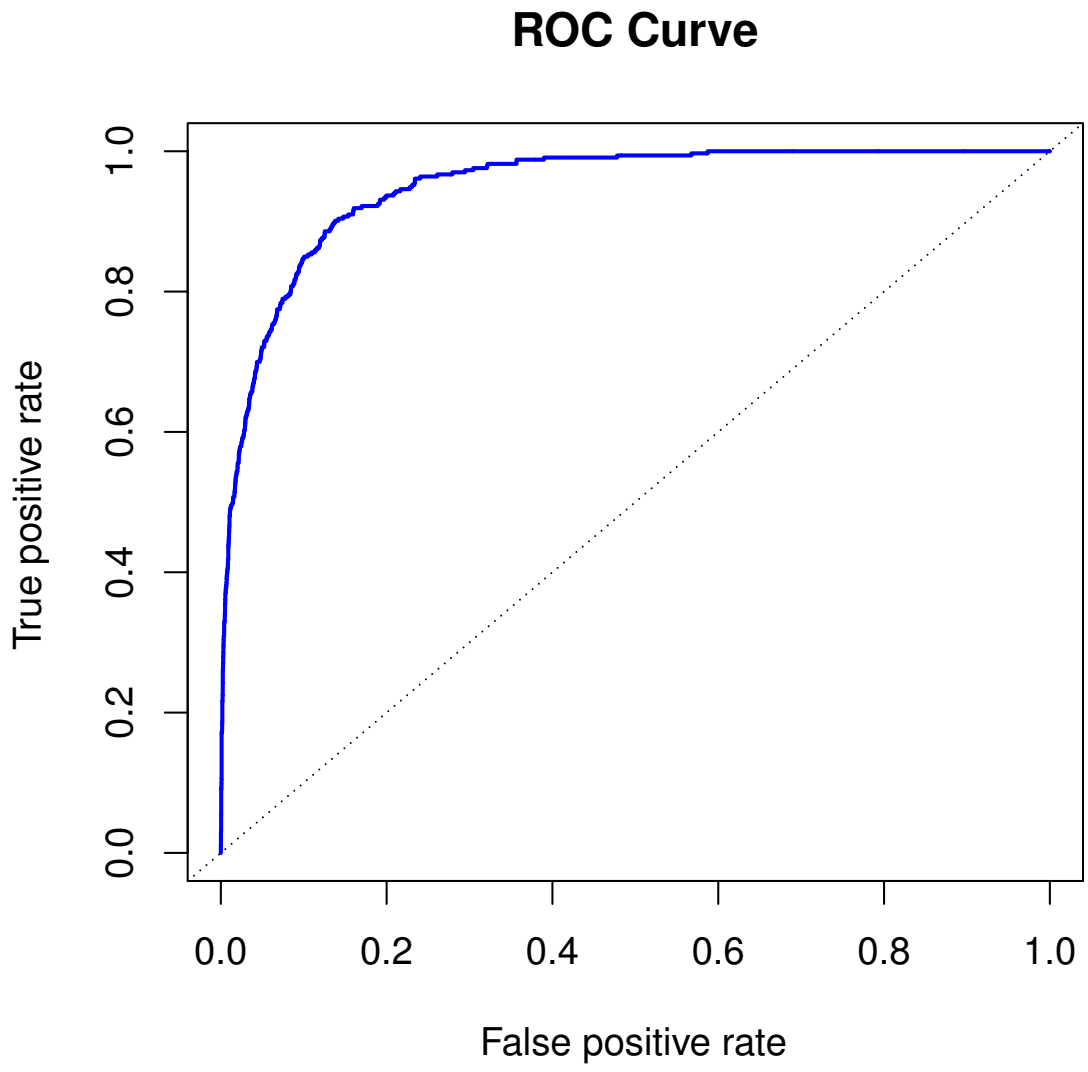
The tradeoff

FIGURE 4.7. For the Default data set, error rates are shown as a function of the threshold value for the posterior probability that is used to perform the assignment. The black solid line displays the overall error rate. The blue dashed line represents the fraction of defaulting customers that are incorrectly classified (Type II error), and the orange dotted line indicates the fraction of errors among the non-defaulting customers (Type I error: FPR=1-Specificity).



The ROC curve (Receiver Operation Characteristic)

FIGURE 4.8. A ROC curve for the LDA classifier on the Default data. It traces out two types of error as we **vary the threshold value** for the posterior probability of default. The actual thresholds are not shown. The true positive rate is the sensitivity: the fraction of defaulters that are correctly identified, using a given threshold value. The false positive rate is 1-specificity: the fraction of non-defaulters that we classify incorrectly as defaulters, using that same threshold value. The ideal ROC curve hugs the top left corner, indicating a high true positive rate and a low false positive rate. The dotted line represents the ROC of random classifier; this is what we would expect if student status and credit card balance are not associated with probability of default.


ROC Curve
True positive rate
False positive rate
0.0
0.2
0.4
0.6
0.8
1.0

General confusion matrix

TABLE 4.6. Possible results when applying a classifier or diagnostic test to a population.

		TRUE class		
		- or Null	+ or Non-null	Total
Predicted status	- or Null	True Neg. (TN)	False Neg. (FN)	N*
	+ or Non-null	False Pos. (FP)	True Pos. (TP)	P*
	Total	N	P	

Clarifying the terminology

TABLE 4.7. Important measures for classification and diagnostic testing, derived from quantities in Table 4.6.

Name	Definition	Synonyms
False Pos. rate	FP/N	Type I error, 1 - specificity
True Pos. rate	TP/P	1- Type II error, power, sensitivity, recall
Pos. Pred.value	TP/P*	Precision, 1- false discovery rate
Neg.Pred.value	TN/N*	

- F_1-score is the harmonic mean of precision and recall:

$$F_1 = \frac{2}{\text{precision}^{-1} + \text{recall}^{-1}} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}.$$