Lecture Notes on Statistical Modelling

Qingyuan Zhao

December 1, 2025

Website for this course: http://www.statslab.cam.ac.uk/~qz280/teaching/modelling-2025/.

This document contains lecture notes and practical sheets for the *Statistical Modelling* course in the Cambridge Mathematics Tripos Part IIC. The materials covered here are influenced by previous lecturers. It should be used for educational purposes only. Please contact me if you find any mistakes or have any comments.

Contents

Chapter 1

Introduction

1.1 This course

This course requires a good understanding of the Part IB course *Statistics*. (Don't worry if you are rusty; we will review all we need from IB *Statistics* in the first two lectures.) This course complements the Part II courses *Principles of Statistics* and *Mathematics of Machine Learning* by providing a more applied and computational perspective. You will learn to write some `R` code, use statistical models to analyze some real datasets, prove a few results that are not too technically challenging but very insightful, and, most importantly, think like a statistician.

On the course website you will find the lecture notes from 2019 and 2024. This year we have a slightly revised schedule which brings in a few "modern" elements. Additionally, you might find the following books useful:

- A. Agresti. *Foundations of Linear and Generalized Linear Models*. Wiley 2015. (Especially Chapters 2, 3, 4, 7.)
- D. Freedman. *Statistical Models: Theory and Practice*. Cambridge University Press 2009. (Provides perspectives from causal inference and social science.)
- G. James, D. Witten, T. Hastie, R. Tibshirani. *An Introduction to Statistical Learning (with Applications in R)*. Springer 2013. (Provides perspectives from machine learning.)
- B. Efron, T. Hastie. Computer Age Statistical Inference: Algorithms, Evidence and Data Science. Cambridge University Press 2016. (Not in the schedules but expand course materials in many ways.)

We will make the distinction between lectures and practical sessions (in which you will need to bring in a laptop with R installed) less pronounced.

For Cambridge mathematics students, it might not be obvious that *statistics is not a branch of mathematics*.[1] There is no consensus on the definition of statistics (especially with the rise of machine learning and data science), but the following definition in Wikipedia cannot be too wrong:

- *Statistics is the discipline that concerns the collection, organization, analysis, interpretation, and presentation of data.*

Compare this with the following definition of mathematical statistics:

- *Mathematical statistics is the study of statistics from a mathematical standpoint, using probability theory as well as other branches of mathematics such as linear algebra and analysis.*

This is similar in many ways to physics versus theoretical physics. Another way to think about it is that mathematics is mostly about deductive reasoning from a set of axioms and assumptions, while statistics is mostly concerned with inductive reasoning from empirical data.[2] Through exploring different statistical models and learning R, a great programming language for statistical computing, you will be exposed to both the mathematical and non-mathematical elements of statistics.

To understand how statistics are used in practice, the following quote by George Box[3] may be illuminating:

> Scientific research is usually an iterative process. The cycle: conjecture–design–experiment–analysis leads to a new cycle of conjecture–design–experiment–analysis and so on.... The experimental environment ... and techniques appropriate for design and analysis tend to change as the investigation proceeds.

At one point, the dominant view was that statistical modelling is a critical step of "analysis" and the model is built after data are collected. However, modern statisticians (and in fact, many pioneers like Box and Fisher) view statistical model as an essential component of the scientific process that guides all steps of the cycle and is being continuously updated. And others like John Tukey envisioned a more dynamic/less rigorous process. My personal take is a tree—"statistical theory" is the root, "statistical methodology" is the trunk, "statistical principles" are the branches, and "statistical practices" are the leaves and fruits. In any case, the title of this course is of great taste: we will learn about "statistical modelling" instead of just "statistical models".

In the broad sense, a statistical model is a mathematical representation of some real-world process that generate data. With the rise of data mining/machine learning/big data/data science/artificial intelligence, you will notice a divide between "two cultures" of statistical modelling: the "data modelling" culture and the "algorithmic modelling" culture in the language used by Leo Breiman a quarter of a century ago.[4] This is about two aspects of any data analyses:

Computing Numbers (or other types of data) go into a computer, and some numbers (or other types of data) come out.

Inference What the input means, and what the output tells us about our problem.

While statistics are mainly concerned with inference in the past, computing is playing a bigger and bigger role. We will try to maintain a neutral position and introduce both perspectives in this course.

We will primary use "statistical model" in the narrow sense to refer to a collection of probability distributions $\mathbb{P} = \{\mathsf{P}_\theta : \theta \in \Theta\}$.[5] Statistical inference in the narrow sense is about learning the unknown parameter θ or the distribution P_θ using a realization from P_θ (this is why statistics was sometimes called "inverse probability" in very old times). Actually most of the time we will specify statistical models by making restrictions/assumptions on (the probability distribution of) the random variables. Although we often do not write down the statistical model explicitly, you should always be prepared to accept that challenge.

A slightly more technical point is that statistical models can be defined at different levels:

(i) Models for conditional moments. For example, a *linear model for conditional expectation* assumes $\mathsf{E}[Y \mid X = x] = x^T\beta$.

(ii) Models for joint or conditional distributions. For example, the classical *normal linear model* assumes $Y = X^T\beta + \epsilon$ where the noise variable $\epsilon \perp\!\!\!\perp X$ and $\epsilon \sim \mathrm{N}(0, \sigma^2)$.

(iii) Structural or causal models that not only describe (associational) relationship for the data at hand but also (causal) relationship under counterfactual interventions. For example, the *linear structural equation model* assumes $Y^{(x)} = x^T\beta + \epsilon$, where $Y^{(x)}$ is the counterfactual value of Y under the intervention that sets X to x and ϵ is an independent noise variable.

This course used to be mainly about models of the second kind, but we will try to cover other kinds of models as part of the modernization effort.

1.2 Basic probability and statistics

1.2.1 Notation

Upper-case letters indicate matrices or random variables. Lower-case letters indicate fixed quantities. We use I_p to denote the $p \times p$ identity matrix, 1_p to denote the p-vector of ones, and 0_p the p-vector of zeros. Independent random variables (or vectors) X and Y are denoted as $X \perp\!\!\!\perp Y$. As a convention, we usually use subscript $i \in \{1, \dots, n\}$ to index observations and $j \in \{1, \dots, p\}$ to index variables. "Independent and identically distributed" is abbreviated as "i.i.d.". The Euclidean norm of a vector Y is denoted as $\|Y\|$. Convergence in probability is denoted as $\overset{p}{\to}$. Convergence in distribution (weak convergence) is denoted as $\overset{d}{\to}$. We use sans serif font P to denote a probability distribution and $\mathbb{P}$ to denote a statistical model—a collection of probability distributions.

1.2.2 Basic probability

A *probability space* $(\Omega, \mathcal{F}, \mathsf{P})$ is a triple where Ω is a set of outcomes (the *sample space*), $\mathcal{F}$ is a collection of subsets of Ω that is a σ-algebra (the *event space*), and $\mathsf{P} : \mathcal{F} \to [0, 1]$ is a function that satisfies the standard Kolmogorov axioms (non-negativity, unit measure, and countable additivity). A *(real-valued) random variable* X is a (Borel measurable) function from the probability space to $\mathbb{R}$. A *random vector* X is a (Borel measurable) function from the probability space to $\mathbb{R}, d \geq 2$. These abstract definitions will never be used and best forgetten in this course. Instead, what is important for us is the probability measure on $\mathbb{R}$, $\mathbb{R}^d$, or a more general space that the data live in that is *induced by the random variable*. So the basic objects for us are the data—modelled as a random variable/vector—and how they are generated—modelled by their probability distribution, which we also denote by P.

There are many ways to characterize the probability distribution of a real-valued random variable X. The *cumulative distribution function* (CDF) of X is defined as[6]

$$F(x) = \mathsf{P}(X \leq x).$$

The *quantile function* is the inverse of the CDF:

$$F^{-1}(u) = \inf\{x \in \mathbb{R} : F(x) \geq u\}.$$

If X is (aboslutely) continuous, we can describe the distribution of X using its *probability density function*

$$f(x) = \frac{\mathrm{d}}{\mathrm{d}x} F(x).$$

If X is discrete, we can use its probability mass function

$$f(x) = \mathsf{P}(X = x).$$

If you know the Radon-Nikodym theorem, you will know these are basically the same thing (so we use the same notation). The *moment generating function* is defined as

$$M(t) = \mathsf{E}(e^{tX}),\ t \in \mathbb{R}\,.$$

We know $M(0) = 1$, but it can be infinite elsewhere. The expectation E here is over the probability distribution P and you should know how that is defined.[7] All these definitions, besides the quantile function, can be naturally extended to multivariate X.

At this point, you should check if you can confidently answer the following questions:

(i) What does it mean for random variables $X_1, \dots, X_n$ to be *independent*? What is the implication of independence on the *joint density function* of $X = (X_1, \dots, X_n)$?

(ii) How do you compute the expectation and variance of a random variable? If $X = (X_1, \dots, X_n)$ is a random vector, $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$ are fixed, what is the expectation and covariance matrix of $AX + b$?

(iii) What is the law of large numbers? What is the central limit theorem?

(iv) What is conditional expectation? What is the law of total expectation?

If you are unsure about anything above, take a look at the notes for IB Statistics lecture 1.

We now review the normal distribution and related distributions. A d-dimensional random vector Z is said to follow the *multivariate normal* distribution with mean $\mu \in \mathbb{R}^d$ and positive definite[8] covariance matrix $\Sigma \in \mathbb{R}^{d \times d}$, written as $Z \sim \mathrm{N}_d(\mu, \Sigma)$, if its probability density function is given by

$$f(z) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} e^{-(z-\mu)^T \Sigma^{-1}(z-\mu)/2}.$$

The multivariate normal distribution has two important properties:

(i) If $Z \sim \mathrm{N}_d(\mu, \Sigma)$, then for any fixed matrix $A \in R^{k \times d}$ and vector $b \in \mathbb{R}^k$, $AZ + b \sim \mathrm{N}_k(A\mu + b, A\Sigma A^T)$.

(ii) If Z_1 and Z_2 are two random vectors and $\begin{pmatrix} Z_1 \\ Z_2 \end{pmatrix}$ follows a multivariate normal distribution, then $Z_1 \perp\!\!\!\perp Z_2$ if and only if $\mathsf{Cov}(Z_1, Z_2) = 0$.

We often omit the subscript d in $\mathrm{N}_d(\mu, \Sigma)$ if the dimension is clear from the context.

Let $Z \sim \mathrm{N}_d(0, I)$. Then we say

$$\|Z\|^2 = \sum_{i=1}^{d} Z_i^2 \sim \chi_d^2$$

follows the *chi-square distribution* with d degrees of freedom. Suppose $Z \sim \mathrm{N}(0, 1)$, $S \sim \chi_d^2$, and $Z \perp\!\!\!\perp S$. Then we say

$$\frac{Z}{\sqrt{S/d}} \sim t_d$$

follows the *(Student's) t-distribution*[9] with d degrees of freedom. Suppose $S_1 \sim \chi^2_{d_1}$, $S_2 \sim \chi^2_{d_2}$, and $S_1 \perp\!\!\!\perp S_2$. Then we say

$$\frac{S_1/d_1}{S_2/d_2} \sim F_{d_1,d_2}$$

follows the *F-distribution* with degrees of freedom d_1 and d_2.

As a memory aid, the above definitions can be summarized as follows:

$$\chi^2_d = \underbrace{\mathrm{N}(0,1)^2 + \cdots + \mathrm{N}(0,1)^2}_{d \text{ times}}, \quad t_d = \frac{\mathrm{N}(0,1)}{\sqrt{\chi^2_d/d}}, \quad F_{d_1,d_2} = \frac{\chi^2_{d_1}/d_1}{\chi^2_{d_2}/d_2}.$$

In this informal description, two random variables (as indicated by their distributions) are independent whenever they appear in the same expression. It is obvious that $t_d^2 = F_{1,d}$.

1.2.3 Basic statistics

A *statistical model* is a collection of probability distributions $\mathbb{P} = \{\mathsf{P}_\theta : \theta \in \Theta\}$ indexed by a unknown *parameter* θ. This notation comes with an implicit bias that the interesting object is the parameter θ, but a true statistician is really interested in knowing the probability distribution P_θ. This motivates the concept of identifiability: we say a statistical model is *identifiable* if $\mathsf{P}_{\theta_1} = \mathsf{P}_{\theta_2}$ implies $\theta_1 = \theta_2$. Most of the models in this course are *parametric*, meaning Θ is an (usually open) subset of $\mathbb{R}^p, p \geq 1$. However, nonparametric and semiparametric models, in which Θ is infinite dimensional, do creep in. The properties of any statistical procedure (how we interpret the "numbers" coming out of a computer) depend on the statistical model $\mathbb{P}$. When doing math we assume the statistical model $\mathbb{P}$ is given, but that is usually not the case when we analyze data.

We learned in IB *Statistics* about ways to obtain point/interval estimators of θ and test a hypothesis like $\theta \in \Theta_0$. In parametric models, the center pillar for this the *likelihood function*:

$$L(\theta) = f(X; \theta),$$

where the data $X \sim \mathsf{P}_\theta$ with density function $f(\cdot; \theta)$. In words, the likelihood function is the density function evaluated at the data and viewed as a function of the parameter. The *maximum likelihood estimator* (MLE) is defined as

$$\hat{\theta}_{\mathrm{MLE}} = \arg\max_\theta L(\theta).$$

It is often easier to do this maximization using the log-likelihood function, $l(\theta) = \log L(\theta)$. The MLE is a *statistic* (a function of the data), because the likelihood function depends on the data.

We can characterize how good a point estimator $\hat{\theta}$ is using its *bias*:

$$\mathsf{Bias}_\theta(\hat{\theta}) = \mathsf{E}_\theta(\hat{\theta}) - \theta$$

and its *mean squared error* (MSE):

$$\mathsf{MSE}_\theta(\hat{\theta}) = \mathsf{E}_\theta\{(\hat{\theta} - \theta)^2\}.$$

These are functions of the unknown parameter θ, which is important because we care about how good $\hat{\theta}$ is over the entire statistical model. We say $\hat{\theta}$ is *unbiased* if $\mathsf{Bias}(\hat{\theta};\theta) = 0$ for all θ. We say a statistic $T = T(X)$ is *sufficient* if the conditional distribution of X given T does not depend on θ. The *factorization theorem* says that T is sufficient if and only if $f(x;\theta) = g(T(x),\theta)h(x)$ for some suitable functions g and h. The *Rao-Blackwell theorem* says that if T is a sufficient statistic, then the MSE of $\mathsf{E}(\tilde{\theta} \mid T)$ is no larger than that of $\tilde{\theta}$ for any estimator $\tilde{\theta}$ at any θ. The mean squared error admits a *bias-variance decomposition*:

$$\mathsf{MSE}_\theta(\hat{\theta}) = \mathsf{Bias}_\theta(\hat{\theta})^2 + \mathsf{Var}_\theta(\hat{\theta}).$$

So it is possible for biased estimators to have smaller MSE than unbiased estimators.

We say $S(X) \subseteq \Theta$ is a $(1-\alpha)$-*confidence set* of θ if

$$\mathsf{P}_\theta\{\theta \in S(X)\} = 1 - \alpha, \text{ for all } \theta \in \Theta.$$

In IB *Statistics* we only looked at the case where $S(X) = [L(X), U(X)]$ is an interval. The key to construct confidence intervals/sets is usually a *pivotal quantity* $R(X,\theta)$ whose distribution under P_θ does not depend on θ.

Hypothesis testing is often formulated as deciding between $H_0 : \theta \in \Theta_0$ and $H_1 : \theta \in \Theta_1$ (usually $\Theta_1 = \Theta \backslash \Theta_0$), and the decision can be represented by a test function $T(X) \in \{0,1\}$. The key concept is the *power function* $\beta(T;\theta) = \mathsf{P}_\theta(T(X) = 1)$. The *size* of the test is $\sup_{\theta\in\Theta_0} \beta(T,\theta)$. The Neyman-Pearson theory formulates hypothesis testing as maximizing $\beta(T;\theta)$ for $\theta \in \Theta_1$ while controlling its size at some prespecified level α. For simple versus simple ($\Theta_0 = \{\theta_0\}$ and $\Theta_1 = \{\theta_1\}$), the most powerful test rejects H_0 if the following *likelihood ratio statistic* is larger than some threshold

$$\Lambda(X) = \frac{L(\theta_1)}{L(\theta_0)} = \frac{f(X;\theta_1)}{f(X;\theta_0)}.$$

This is a special case of the *generalized likelihood ratio statistic*

$$\Lambda(X) = \frac{\sup_{\theta\in\Theta_0\cup\Theta_1} L(\theta)}{\sup_{\theta\in\Theta_0} L(\theta)}, \tag{1.1}$$

which can be used for composite hypotheses. It is often easier to work with them on the logarithmic scale. For the multinomial model $(N_1,\dots,N_k) \sim \text{Multi}(n; p_1,\dots,p_k)$ where n is given and $\theta = (p_1,\dots,p_k)$ is any vector in the standard $(k-1)$-simple, we saw in IB *Statistics* that the generalized log-likelihood ratio statistic for testing $H_0 : p_i = p_{0i}, i = 1,\dots,k$ for some given $\theta_0 = (p_{01},\dots,p_{0k})$ is given by

$$2\log\Lambda = 2\sum_{i=1}^{k} O_i \log\left(\frac{O_i}{E_i}\right) \approx \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i},$$

where $O_i = N_i$ is the "observed count", $E_i = np_{0i}$ is the "expected count", and the approximation on the right hand side is called Pearson's χ^2-statistic because it (and also the other expression) converge in distribution to χ^2_{k-1} as $n \to \infty$ by Wilks' theorem.

There is a useful duality between simple hypothesis tests and confidence sets. Consider

$$T(\theta, X) = 1_{\{\theta \notin I(X)\}} \text{ and } S(X) = \{\theta : T(\theta, X) = 0\}.$$

Then $T(\theta_0, X)$ is a size-α test of $H_0 : \theta = \theta_0$ for all $\theta_0 \in \Theta$ if and only if $S(X)$ is a $(1-\alpha)$-confidence set for θ.

For hypothesis tests, it is often easier to report the *P-value*:

$$P = \sup_{\theta \in \Theta_0} P(\theta, X), \text{ where } P(\theta, X) = \mathsf{P}_\theta(\Lambda(\tilde{X}) \geq \Lambda(X) \mid X),$$

and $\tilde{X}$ is an i.i.d. copy of X ($\tilde{X} \sim \mathsf{P}_\theta$ and $\tilde{X}$ is independent of X). So $P(\theta, X)$ measures how "extreme" $\Lambda(X)$ is if the data X are generated from P_θ. Rejecting H_0 when $P \leq \alpha$ (so $T = 1_{\{P \leq \alpha\}}$) ensures that the test has size α. This corresponds to the $(1-\alpha)$-confidence set

$$S(X) = \{\theta \in \Theta : P(\theta, X) > \alpha\},$$

which contains all θ not rejected at level α.

Thus far we have discussed frequentist inference that treats probability as long-term frequency and the parameter θ as a fixed quantity describing a natural law. An alternative viewpoint is Bayesian inference that treats probability as subjective plausibility and the parameter θ as a random variable, with *prior distribution* π. By using the Bayes formula, the *posterior distribution* of θ is given by

$$\pi(\theta \mid X) = \frac{\pi(\theta) f(x \mid \theta)}{f(x)},$$

where $f(x) = \int f(x \mid \theta)\pi(\theta) d\theta$ is the marginal density of X. We often ignore the normalizing constant and write $\pi(\theta \mid X) \propto \pi(\theta) f(x \mid \theta)$, or in other words,

$$\text{posterior} \propto \text{prior} \cdot \text{likelihood}.$$

So we use information in the likelihood to update our belief about θ.

In Bayesian statistics, all we know about the parameter is contained in the posterior distribution $\theta(\theta \mid X)$. We can use the *Bayes risk* to evaluate an estimator $\tilde{\theta} = \tilde{\theta}(X)$:

$$R(\tilde{\theta}) = \mathsf{E}_\pi(L(\theta, \tilde{\theta}) \mid X) = \int_\Theta L(\theta, \tilde{\theta}(X)) \pi(\theta \mid X) d\theta,$$

where $L : \Theta \times \Theta \to \mathbb{R}$ is a *loss function.* The *Bayes estimator* is given by $\hat{\theta} = \arg\min_{\tilde{\theta}} R(\tilde{\theta})$. We saw in IB *Statistics* that the Bayes estimator under the quadratic loss $L(\theta, \tilde{\theta}) = (\theta - \tilde{\theta})^2$ is the posterior mean $\mathsf{E}_\pi(\theta \mid X)$.

Exercise 1.1. Suppose X is a random variable with a continuous cumulative distribution function $F(x)$. Without assuming $F(x)$ is strictly increasing, show that $F(X) \sim \text{Unif}[0, 1]$. Use this to show that $P(\theta, X)$ is a pivotal quantity if the CDF of $\Lambda(X)$ is continuous for all $\theta \in \Theta$.

1.3 Practical 1: R basics

R is a free and open-source programming language for statistical computing and data visualization. Please download and install the appropriate distribution of R from https://cran.r-project.org/. Many students will find it easier to use an integrated development environment (IDE) such as RStudio, which can be downloaded from https://posit.co/download/rstudio-desktop/.

1.3.1 Arithmetics

R can be used as a calculator:

```
> (9.1^3) * sqrt(14) * exp(-5) / log(4)
```

Help on any R function can be found by typing a question mark followed by the function.

```
> ?exp
> help(exp)
> ??exp
```

You will need to use this `help` facility extensively (and get used to skim-reading to find the relevant bit!). Note that R is case-sensitive.

The `<-` symbol is the usual assignment operator in R (the `=` symbol can also be used, but it has slightly different purposes so you are recommended to stick to `<-`). For instance, we can assign the value `3` to the variable `x`, and then perform operations on `x`. Anything which appears after the hash symbol `#` is a comment and need not be typed.

```
> x <- 3
> round(x^2 + log10(x), 3)  # try ?round to see what it does
[1] 9.477
> 37 %/% 3  # try ?‘%/%’
[1] 12
> 37 %% 3
[1] 1
```

1.3.2 Vectors, matrices, lists

Creating vectors

The `c` function (for "concatenate") combines values into a vector.

```
> x <- c(3, 6, 4, 2)
> x
[1] 3 6 4 2
> length(x)
[1] 4
```

There is no such thing as a scalar in R; what one might think of as a scalar is treated as a vector of length 1. Note that, unlike Matlab, R does not distinguish between row and column vectors.

A sequence of equally spaced numbers can be created using the `seq` function. The `rep` function provides different ways of repeating vectors.

The base R does many things in a “smart” way. This is often convenient but can sometimes be perilous. For example, the default `seq` method takes the following input:

```
seq(from = 1, to = 1, by = ((to - from)/(length.out - 1)),
    length.out = NULL, along.with = NULL, ...)

> seq(1, 10)
 [1]  1  2  3  4  5  6  7  8  9 10
> seq(10)
 [1]  1  2  3  4  5  6  7  8  9 10
> seq(1, 10, 2)
[1] 1 3 5 7 9
> seq(1, by = 2, length = 5)
[1] 1 3 5 7 9
> seq(1, 10, length = 5)
[1]  1.00  3.25  5.50  7.75 10.00
> seq(10, length = 5)
[1] 10 11 12 13 14
> seq(to = 10, length = 5)
[1]  6  7  8  9 10
```

The `seq` function is a bit extreme, but it is generally a good idea to try things out when you are not sure.

Operations on vectors

Operations on vectors in R are performed component by component. For example

```
> x + x
[1]  6 12  8  4
> x*x
[1]  9 36 16  4
> exp(x)
[1]  20.085537 403.428793  54.598150   7.389056
```

Another “smart” thing in R: when operations are performed on vectors of different lengths, the shorter vector is cycled until it is the same length as the longer vector.

```
> x <- c(3, 6, 4, 2)
> y <- c(1, 2)
> x + y
[1] 4 8 5 4
```

```
> x*y
[1]  3 12  4  4
> x^y
[1]  3 36  4  4
> y <- 1:3  # same as y <- c(1, 2, 3) or y <- seq(1, 3)
> x + y
[1] 4 8 7 3
Warning message:
In x + y : longer object length is not a multiple of shorter object length
```

What are the values of `x + 2`, `3*x` and `(2 + x)^3`?

Indexing vectors

```
> x <- c(3, 6, 4, 2)
> x[2]  # 2nd component of x
[1] 6
> x[c(1, 3)]  # 1st and 3rd components of x
[1] 3 4
> x[-1]  # All of x except the 1st component
[1] 6 4 2
> x[-(1:2)]  # All of x except the 1st two components
[1] 4 2
> x[1:2] <- c(7.1, 3.4)  # We can assign values to components
> x
[1] 7.1 3.4 4.0 2.0
```

Note that after the final command, `x` has automatically transformed from a vector of integers to a vector of *floating point numbers* (these are a way of representing real numbers on computers, though of course only to a certain degree of accuracy).

We can also index components of a vector using a `TRUE / FALSE` (logical) vector.

```
> index_vec <- c(TRUE, TRUE, FALSE, TRUE)
> x[index_vec]
[1] 7.1 3.4 2.0
```

Logical vectors can also be created using the binary operator `<` which performs componentwise comparisons.

```
> x > 3.6
[1]  TRUE FALSE  TRUE FALSE
> x[x > 3.6]
[1] 7.1 4.0
```

Matrices

We can create a matrix using the `matrix` function.

```
> A <- matrix(1:8, 2, 4)
> A
     [,1] [,2] [,3] [,4]
[1,]    1    3    5    7
[2,]    2    4    6    8
```

Can you enter the terms by row instead?

Rows and columns of matrices can be extracted in the following way:

```
> A[1, ]
[1] 1 3 5 7
> A[, 3]
[1] 5 6
```

Note that the rows and columns thus formed are now vectors. We can check this using the very helpful `str` (for 'structure') function.

```
> str(A[1, ])
 int [1:4] 1 3 5 7
```

Here we see that `A[1, ]` is an integer vector of length 4. To keep the 2-by-1-matrix structure-type, we use

```
> A[, 2, drop = FALSE]
     [,1]
[1,]    3
[2,]    4
```

An alternative is to do

```
> matrix(A[, 2])
     [,1]
[1,]    3
[2,]    4
```

Submatrices can be formed by e.g. `A[, 1:3]`. The diagonal can be extracted using `diag`. We can perform many standard operations on matrices.

```
> A %*% x  # matrix vector multiplication
     [,1]
[1,] 51.3
[2,] 67.8
> A*A  # componentwise multiplication
     [,1] [,2] [,3] [,4]
[1,]    1    9   25   49
```

```
[2,]    4   16   36   64
> t(A)
     [,1] [,2]
[1,]    1    2
[2,]    3    4
[3,]    5    6
[4,]    7    8
> A %*% t(A)  # matrix matrix multiplication
     [,1] [,2]
[1,]   84  100
[2,]  100  120
```

The `solve` function can be used to solve linear systems like $Ax = b$. If b is missing in the input, it will invert the matrix A.

```
> solve(A %*% t(A))
      [,1]  [,2]
[1,]  1.50 -1.25
[2,] -1.25  1.05
```

Lists

Lists collect together items of different types, e.g.

```
> Empl <- list(employee="Eve", spouse = "Adam",
               children=2, child.ages = c(4, 7))
```

Elements of a list need not be of the same length, but its components are numbered. Thus `Empl` is a list of length 4, and its components are referred to as `Empl[[1]]` etc.. Notice that `Empl[[4]]` is a vector, so `Empl[[4]][1]` is its first entry. Names of components can also be used:

```
> Empl$employee
> Empl[["employee"]]
> Empl$child.ages[2]
```

The different components of a list can be almost anything, even functions or other lists.

1.3.3 Functions

A few important functions

```
> x <- c(3, 6, 4, 2)
> sum(x)
[1] 15
> sum(x > 3)  # TRUE is treated as 1 and FALSE, 0
[1] 2
> mean(x)
```

```
[1] 3.75
> sort(x)
[1] 2 3 4 6
> sd(x)  # standard deviation
[1] 1.707825
```

How is the standard deviation being calculated? Functions for matrices:

```
> mean(A)  # mean treats A as a vector
[1] 4.5
> colMeans(A)
[1] 1.5 3.5 5.5 7.5
> rowSums(A)
[1] 16 20
```

The function `cbind` 'glues' columns of matrices together.

```
> cbind(1, A)
     [,1] [,2] [,3] [,4] [,5]
[1,]    1    1    3    5    7
[2,]    1    2    4    6    8
```

Writing your own functions

When writing anything more than a few lines, it is useful to edit the commands from a script file that ends with `.R`. Create a script file (in Rstudio you can use Ctrl+Shift+N), then copy and paste the following code:

```
f <- function (x, y) {
  z <- x^2 + y^2
  return(c(cos(z), sin(z)))
}
```

Once you have written the algorithm and saved the file as `Rubbish.R`, say, in the current working directory, you can execute the commands by typing

```
> source("Rubbish.R")
```

in the console. Typing `f` in the console will now echo the code of your function, and you can run your function by giving it the right arguments e.g. `f(2, 3)`.

Generating (pseudo) random numbers

(Pseudo) independent and identically distributed sequences of random numbers are generated with commands like `rnorm, runif, rchisq` etc. (normal, uniform, χ^2). The corresponding density, cumulative distribution and quantile functions are, e.g. `dnorm, pnorm, qnorm`.

```
> x <- rnorm(1000)
> hist(x, freq = FALSE)
> x_vec <- seq(-3, 3, by = 0.1)
> lines(x_vec, dnorm(x_vec), col = "red")  # adds lines to an existing plot
```

What does the following code do?

```
> X <- matrix(runif(50*1000, min=-1, max=1), 50, 1000)
> hist(sqrt(50) * colMeans(X) / sd(X), freq = FALSE)  # sd treats X as a vector
> lines(x_vec, dnorm(x_vec), col = "red")
```

`lines` plots on top of the current plot, but if you wish to use superposition to histograms (or plots) you can use the following instruction (equivalent to to "hold on" in MATLAB).

```
> par(new=TRUE)
```

Experiment with other distributions and other sample sizes.

1.3.4 Loops and vectorization

`for` Loops and `while` Loops

Often we will like to run the same code many times, for instance if we using simulated data to examine a statistical methodology. The prime tools to do this are `for` and `while` loops in the code. These loops repeat the code inside the { } blocks then iterates until the for loop runs out of indices or the while loop condition is broken. These have similar syntax:

```
for (i in 1:B) { doSomething() }

counter <- 0
while (counter < B) {
   doSomething()
   counter <- counter + 1
}
```

both of which accomplish the same goal of calling the `doSomething` function B times.

This approach applies the code blocks in sequence, which is useful if accessing the same memory block and adjusting it multiple times (which is what happened to the memory storing i in the while loop). However often the code will access new memory for each iteration, and so we could instead run these in parallel. For further reading on this, search for the documentation on the `parallel` package in R.

Vectorization of Code

Consider the computing the column means of the elements in a large matrix X using a for loop:

```
n <- 100
d <- 3000

X <- matrix(rnorm(n * d), nrow = n, ncol = d)
X_means <- rep(0, d)
for (i in 1:d) {
  X_means[i] <- mean(X[,i])
}
```

we can use the `system.time` function to test how long this computation takes:

```
system.time(for (i in X) {X_means[i] <- mean(X[,i])})
   user  system elapsed
 43.518   7.317  51.124
```

In contrast, we can compute this much more quickly by using the `apply` function to compute these column means in parallel.

```
system.time(apply(X, 1, mean))
   user  system elapsed
  0.002   0.000   0.002
```

This is much faster because vectorized computation in R directly calls functions written in C. Vectorizing the for loops also helps you focus on the statistical operations.

1.3.5 Exercises

You are recommended to discuss computing exercises in this course with a learning partner during and/or after the practical sessions.

Exercise 1.2. Use R to solve the following linear equations:

$$\begin{aligned} 3a + 4b - 2c + d &= 9 \\ 2a - b + 7c - 2d &= 13 \\ 6a + 2b - c + d &= 11 \\ a + 6b - 2c + 5d &= 27. \end{aligned}$$

Exercise 1.3. Estimate the upper 5% quantile of a χ^2_6 distribution using the `dchisq` and `quantile` functions, then validate your answer using `qchisq`.

Exercise 1.4. Consider a real valued random variable X with distribution function given by:

$$F_X(x) = \tfrac{1}{\pi}\text{arctan}(x) + \tfrac{1}{2}.$$

Write a function `rmydist(n)` that generates n independent samples of X. Use this function to estimate $\mathrm{Var}(X)$. [*Hint:* Exercise 1.1]

Exercise 1.5. Two lecturers mark the same Tripos question for two randomly selected, disjoint sets of students. To make sure that neither of them is more lenient than the other, they want to test whether their average marks are equal. But they are afraid the sample size is too small to apply the Central Limit Theorem, so one of them writes the following code.

```
grades1 <- c(10,11,14.5,15,15,18,12,19,18.5,19,20,13)
grades2 <- c(12,11,14.5,13,12,11,12,14.5,20,17)
(diff <- mean(grades1) - mean(grades2))
grades.all <- c(grades1,grades2)
n1 <- length(grades1); n2 <- length(grades2);

one.sim <- function(iter) {
    grades.perm <- sample(grades.all);
    mean(grades.perm[1:n1]) - mean(grades.perm[n1 + 1:n2])
}

system.time(diff.perms <- replicate(100000, one.sim(0)))
typeof(diff.perms)
mean(abs(diff.perms) > diff)

system.time(diff.perms <-
                parallel::mclapply(1:100000, one.sim, mc.cores = 2))
typeof(diff.perms)
diff.perms <- unlist(diff.perms)
mean(abs(diff.perms) > diff)

hist(abs(diff.perms))
abline(v = abs(diff), col = "red", lty = "dashed")
```

Execute the code in your R session and explain what it is doing. Search for the documentation of any function you have not encountered before.

1.4 Normal linear models

1.4.1 Model and likelihood

We now review the normal linear model introduced in IB *Statistics*. This model assumes that $(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n) \in \mathbb{R}^p \times \mathbb{R}^1$ are independent and the conditional distribution of Y_i given X_i satisfies[10]

$$Y_i = \sum_{j=1}^{p} X_{ij}\beta_j + \epsilon_i,\ \epsilon_i \perp\!\!\!\perp X_i,\ \epsilon_i \sim \mathrm{N}(0, \sigma^2),\ i = 1, \ldots, n. \tag{1.2}$$

Equation (1.2) is rather cumbersome and can be simplified by introducing the following vector/matrix notation:

$$Y = \begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix},\ X = \begin{pmatrix} X_1^T \\ \vdots \\ X_n^T \end{pmatrix},\ \beta = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_p \end{pmatrix},\ \text{and } \epsilon = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{pmatrix}.$$

Now we can rewrite (1.2) as

$$Y = X\beta + \epsilon,\ \epsilon \mid X \sim \mathrm{N}(0, \sigma^2 I_p). \tag{1.3}$$

This is also known as *linear regression*, due to early applications of such models by Francis Galton that identified a "regression toward the mean" phenomenon in the inheritance of human traits (such as height).

Example 1.6 (Normal measurements)**.** This model assumes $Y_i \overset{i.i.d.}{\sim} \mathrm{N}(\mu, \sigma^2),\ i = 1, \ldots, n$. The model matrix $X = 1_n$ is a matrix with just one column and the regression coefficient $\beta = \mu$ is one-dimensional.

Example 1.7 (ANOVA)**.** Let $F_i \in \{1, \ldots, l\}$ be a *categorical variable* with l levels (also called a *factor*). The classical ANalysis Of VAriance (ANOVA) assumes $Y_i = \beta_{F_i} + \epsilon_i, \epsilon_i \overset{i.i.d.}{\sim} \mathrm{N}(0, \sigma^2)$ where β is a l-dimensional parameter vector. The ith row X_i of the corresponding model matrix X is an indicator vector whose F_ith entry is 1 and all other entries are 0.

The matrix X is known as the *design matrix* or *model matrix*. The former terminology was derived from the classical setting in experimental design in which X is chosen by the experimenter. This is rarely the case in modern applications. For this reason, we will refer to X as the model matrix in this course.

To distinguish (1.2) and (1.3) with models for the conditional expectation, let $\mu_i = \mathsf{E}[Y_i \mid X_i]$. Then (1.3) contains three different types of assumptions:

(i) The conditional expectation satisfies

$$\mu = \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_n \end{pmatrix} = X\beta;$$

(ii) The noise $\epsilon = Y - \mu$ satisfies $\epsilon \perp\!\!\!\perp X$;

(iii) The noise ϵ is distributed as $\mathrm{N}(0, \sigma^2 I_n)$.

Note that (1.2) does not make any assumptions on the distribution of the regressors X. This is dangerous, because the distribution of X should really be regarded as a parameter in the model and be included as an argument of the likelihood function in (1.4). In other words, although the normal linear model is often regarded as the most basic parametric model, it is actually semiparametric (involving infinite-dimensional parameters) in general. Of course, this remark no longer applies if one assumes X is fixed or the distribution of X is unknown, but in most applications the distribution of X is unknown.

In any case, the distribution of X does not really matter in the classical normal linear model because the assumption $\epsilon \perp\!\!\!\perp X$ allows us to factorize the likelihood function as

$$L(\beta, \sigma^2) = f(x_1, \dots, x_n, y_1, \dots, y_n; \beta, \sigma^2) = f(x_1, \dots, x_n) \cdot \prod_{i=1}^{n} f(y_i \mid x_i; \beta, \sigma^2), \tag{1.4}$$

where f is a generic symbol for density functions and $f(y_i \mid x_i; \beta, \sigma^2)$ is the density function of a normal random variable:

$$f(y \mid x; \beta, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(y - x^T\beta)^2/(2\sigma^2)}.$$

Because the marginal density function $f(x_1, \dots, x_n)$ of X does not depend on β, whether the distribution of X is known or not does not affect the inference for β following the likelihood principle.[11]

1.4.2 Ordinary least squares and its geometry

Derivation of ordinary least squares

Following (1.4), the log-likelihood function is given by

$$\begin{aligned} l(\beta, \sigma^2) &= \log \prod_{i=1}^{n} f(Y_i \mid X_i; \beta) + \text{constant} \\ &= -\frac{n}{2} \log \sigma^2 - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (Y_i - X_i^T \beta)^2 + \text{constant} \\ &= -\frac{n}{2} \log \sigma^2 - \frac{1}{2\sigma^2} \|Y - X\beta\|^2 + \text{constant} \end{aligned}$$

Therefore, the maximum likelihood estimator (MLE) of β is given by the solution of the *ordinary least squares* (OLS) problem

$$\hat{\beta} = \arg\min_{\beta} \|Y - X\beta\|^2. \tag{1.5}$$

Notice that this holds regardless of whether σ^2 is known or not.

We may obtain a closed-form solutionto (1.5) by using the following identities for matrix calculus:

$$\frac{\partial}{\partial\beta}(a^T\beta) = a, \text{ and } \frac{\partial}{\partial\beta}(\beta^T A\beta) = (A + A^T)\beta.$$

Therefore the OLS estimator satisfies

$$X^T(Y - X\hat{\beta}) = 0. \tag{1.6}$$

Equation (1.6) is called the *normal equations* because it requires the vector of *residuals* $R = Y - X\hat{\beta}$ to be orthogonal to X.

The linear equations (1.6) have an unique solution if X^TX is invertible (or equivalently if X has full column rank, which requries $n \geq p$). In this case, we have

$$\hat{\beta} = (X^TX)^{-1}X^TY.$$

We maintain the assumption that X has rank p in this Chapter.

The maximum likelihood estimator of σ^2 can be obtained by differentiating $l(\beta, \sigma^2)$ with respect to σ^2:

$$\frac{\partial}{\partial\sigma^2} l(\beta, \sigma^2) = -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4}\|Y - X\beta\|^2.$$

By solving $l(\hat{\beta}, \sigma^2) = 0$, we obtain

$$\hat{\sigma}^2_{\text{MLE}} = \frac{1}{n}\|Y - X\hat{\beta}\|^2 = \frac{1}{n}\|R\|^2.$$

The quantity $\|R\|^2$ is often referred to as the *residual sum of squares* (RSS). Because $\hat{\sigma}^2_{\text{MLE}}$ is biased (see Section 1.4.3), it is more common to use the following unbiased estimator of σ^2:

$$\hat{\sigma}^2 = \frac{n}{n-p}\hat{\sigma}^2_{\text{MLE}} = \frac{1}{n-p}\|Y - X\hat{\beta}\|^2 = \frac{1}{n-p}\|R\|^2.$$

Orthogonal projections

Before discussing the statistical properties of the OLS estimator, it is useful to get a geometric understanding of what it does. By definition, the *fitted values* in the linear model are given by

$$\hat{\mu} = X\hat{\beta} = X(X^TX)^{-1}X^TY,$$

which is a linear transformation of the original response vector Y. Let $P = X(X^TX)^{-1}X^T$, which is sometimes called the *hat matrix* in statistics literature for the obvious reason. Geometrically, the least squares problem (1.5) implies that the vector of fitted values $\hat{\mu} = PY$ is the projection of the response vector Y onto the column space of X.

The matrix P is an *orthogonal projection matrix*, meaning it is symmetric ($P^T = P$) and idempotent ($P^2 = P$). Some useful properties include:

(i) $P = UU^T$, where columns of $U \in \mathbb{R}^{n \times p}$ form an orthonormal basis for the column space of X.

(ii) Eigen-values of P are either 0 or 1. In consequence, $\mathrm{tr}(P) = \mathrm{rank}(P) = p$.

(iii) $I - P$ is also an orthogonal projection matrix, and

$$\|Y\|^2 = \|PY\|^2 + \|(I-P)Y\|^2 = \|\hat{\mu}\|^2 + \|R\|^2.$$

The following result will be useful.

Lemma 1.8. *Let $Z \sim N(0, I_n)$ be a standard normal vector. Then $\|PZ\|^2 \sim \chi^2_p$.*

Exercise 1.9. Prove the last result.

Projection onto nested models

Consider a partition of the regressors:

$$X = (X_0\ X_1),\ \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix},$$

where $X_0 \in \mathbb{R}^{n \times p_0}$, $X_1 \in \mathbb{R}^{n \times (p-p_0)}$, $\beta_0 \in \mathbb{R}^{p_0 \times 1}$, and $\beta_1 \in \mathbb{R}^{(p-p_0)\times 1}$. Recall that $\mu_i = \mathsf{E}(Y_i \mid X_i)$ and $\mu = (\mu_1, \dots, \mu_n)^T$. It useful to think about this in terms of defining different statistical models:

(i) The *saturated model* $\mu \in \mathbb{R}^n$;

(ii) The *full model* $\mu \in \{X\beta : \beta \in \mathbb{R}^p\}$;

(iii) The *submodel* $\mu \in \{X_0\beta_0 : \beta_0 \in \mathbb{R}^{p_0}\}$.

To formally define the model we have to further specify the distribution of Y. In the normal linear model, this is $Y \sim \mathrm{N}(\mu, \sigma^2 I_n)$.

Let P (P_0) denote the projection matrix onto the column space of X (X_0). They satisfy two important properties:

(i) $PP_0 = P_0P = P_0$; see Figure 1.1.

(ii) $P - P_0$ is also a projection matrix.

Exercise 1.10. Prove the second property. Which subspace does $P - P_0$ project onto?

These geometric properties imply the following useful result that generalizes the Gram-Schmidt process in linear algebra.[12]

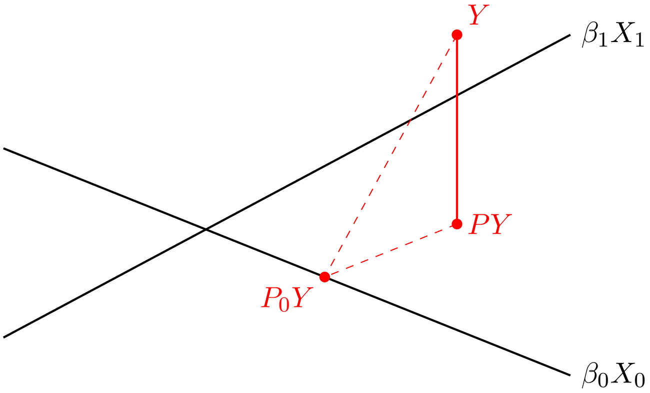


Figure 1.1: Nested model projections.

Proposition 1.11 (Partial regression characterization). *Consider the partition of X as described above and let*

$$\hat{\beta} = \begin{pmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \end{pmatrix}$$

be the corresponding partition of the OLS estimator $\hat{\beta}$. Let $\tilde{X}_1 = (I - P_0)X_1$ and $\tilde{Y} = (I - P_0)Y$ be the residuals of X_1 and Y after projecting onto the column space of X_0. Then $\hat{\beta}_1$ is equal to the OLS estimator for a linear regression of $\tilde{Y}$ on $\tilde{X}_1$:

$$\hat{\beta}_1 = (\tilde{X}_1^T \tilde{X}_1)^{-1} \tilde{X}_1^T \tilde{Y}. \tag{1.7}$$

To prove this, consider any $X_2 \in \mathbb{R}^{n \times (n-p)}$ such that $(X_0 \; X_1 \; X_2)$ is a full-rank $n \times n$ matrix. By applying Gram-Schmidt, we obtain matrices X_0, $\tilde{X}_1 = (P - P_0)X_1$, and $\tilde{X}_2 = (I - P)X_2$ that are orthogonal to each other. In consequence, $P - P_0 = P_{\tilde{X}_1} = \tilde{X}_1(\tilde{X}_1^T \tilde{X}_1)^{-1}\tilde{X}_1^T$ is the projection matrix onto the column space of $\tilde{X}_1$. Correspondingly, Y can be decomposed as

$$\begin{aligned} Y &= X_0\hat{\beta}_0 + X_1\hat{\beta}_1 + R \\ &= \underbrace{(X_0\hat{\beta}_0 + P_0 X_1 \hat{\beta}_1)}_{P_0 Y} + \underbrace{(I - P_0)X_1\hat{\beta}_1}_{(P-P_0)Y} + \underbrace{R}_{(I-P)Y} . \end{aligned}$$

Therefore,

$$\tilde{X}_1 \hat{\beta}_1 = (P - P_0)Y = P_{\tilde{X}_1} Y.$$

Because $\tilde{X}_1$ has full column rank, this establishes (1.7).

An important special case is $p_0 = p - 1$, where X_1 is a single regressor. In this case, some authors refer to $\hat{\beta}_1$ as the *partial regression coefficient* to distinguish from the *marginal regression coefficient* in a regression of Y on just X_1.

Example 1.12 (Simple linear regression)**.** When $p = 1$, the OLS estimator is given by the simple formula:

$$\hat{\beta} = \frac{X^T Y}{X^T X}.$$

When $p = 2$ and the model matrix is

$$X = \begin{pmatrix} 1 & X_1 \\ \vdots & \vdots \\ 1 & X_n \end{pmatrix},$$

the coefficient β_1 is called the *intercept* and β_2 is called the *slope*. By treating the first column of X as X_0 in the above partition, we obtain

$$\hat{\beta}_2 = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^n (X_i - \bar{X})^2},$$

where $\bar{X} = \sum_{i=1}^n X_i/n$ and $\bar{Y} = \sum_{i=1}^n Y_i/n$.

1.4.3 Exact inference for the normal linear model

Besides motivating the OLS problem (1.5) as finding the MLE of β in the normal linear model, the rest of Section 1.4.2 was entirely algebraic. Next, we discuss statistical properties of the OLS estimator and how to use it to make exact inference under the normal linear model (1.2).

Distribution of $\hat{\beta}$ and $\hat{\sigma}^2$

Since $\hat{\beta} = (X^T X)^{-1} X^T Y$ is a linear transformation of Y, it has a multivariate normal distribution conditional on X:

$$\begin{aligned} \hat{\beta} \mid X &\sim \mathrm{N}\left((X^T X)^{-1} X^T \mu, (X^T X)^{-1} X^T \sigma^2 I_n X (X^T X)^{-1}\right) \\ &= \mathrm{N}\left(\beta, \sigma^2 (X^T X)^{-1}\right). \end{aligned}$$

This shows that $\hat{\beta}$ is unbiased. Unbiasedness depends on linearity of μ but not the assumption of normal distribution. The Gauss-Markov theorem says that $\hat{\beta}$ is the *best linear unbiased estimator* (BLUE), that is, it has the smallest variance among all unbiased estimators of β that is linear in Y.

The estimator $\hat{\sigma}^2$ of the noise variance σ^2 can be written as

$$\hat{\sigma}^2 = \frac{1}{n-p} \|Y - X\hat{\beta}\|^2 = \frac{1}{n-p} \|(I_n - P)Y\|^2.$$

Because $I_n - P$ is also a projection matrix, this implies that

$$\hat{\sigma}^2 \mid X \sim \sigma^2 \chi^2_{n-p}/(n-p).$$

This shows that $\mathsf{E}(\hat{\sigma}^2) = \mathsf{E}[\mathsf{E}(\hat{\sigma}^2 \mid X)] = \sigma^2$ and thus $\hat{\sigma}^2$ is unbiased.[13]

Exercise 1.13. Show $\hat{\beta}$ and $\hat{\sigma}^2$ are still unbiased without the normality assumption, that is, by only assuming $\mathsf{E}(\epsilon \mid X) = 0$ and $\mathsf{Var}(\epsilon \mid X) = \sigma^2 I_n$.

Finally, under the normal linear model $\hat{\beta}$ and $\hat{\sigma}^2$ are independent (more precisely, conditionally independent given X) because $(X^T X)^{-1} X^T Y$ and $(I_n - P)Y$ are linear transforms of Y that is multivariate normal and

$$\mathsf{Cov}\left((I_n - P)Y, (X^T X)^{-1} X^T Y \mid X\right) = (I_n - P)\sigma^2 I_n X (X^T X)^{-1} = 0.$$

Confidence sets

We have been very careful about when X is conditioned on. This makes the statements precise but clumsy. For the rest of this section, we view X as fixed. You should know how to translate the statements if X is not fixed but conditioned on.

The key to exact inference is to find *pivotal quantities* whose distribution does not depend on unknown parameters. For example,

$$\frac{(n-p)\hat{\sigma}^2}{\sigma^2} \sim \chi^2_{n-p} \tag{1.8}$$

is pivotal, but

$$\hat{\beta} - \beta \sim \mathrm{N}\left(0, \sigma^2 (X^T X)^{-1}\right) \tag{1.9}$$

is not pivotal because the distribution depends on σ^2. Instead, we can use the following pitoval quantity

$$\frac{\hat{\beta} - \beta}{\hat{\sigma}} \sim \frac{\mathrm{N}\left(0, (X^T X)^{-1}\right)}{\sqrt{\chi^2_{n-p}/(n-p)}}.$$

Element-wise, we have

$$\frac{\hat{\beta}_j - \beta_j}{\hat{\sigma}} \sim \frac{\mathrm{N}\left(0, (X^T X)^{-1}_{jj}\right)}{\sqrt{\chi^2_{n-p}/(n-p)}} = \sqrt{(X^T X)^{-1}_{jj}} \cdot t_{n-p},\ j = 1, \dots, p. \tag{1.10}$$

By using (1.10), we can immediately construct a $(1-\alpha)$-confidence interval for β_j:

$$\mathcal{CI}_j(\alpha) = \left[\hat{\beta}_j - \hat{\sigma}\sqrt{(X^T X)^{-1}_{jj}} t_{n-p}(\alpha/2), \hat{\beta}_j + \hat{\sigma}\sqrt{(X^T X)^{-1}_{jj}} t_{n-p}(\alpha/2)\right],$$

where $t_{n-p}(\alpha/2)$ is the upper $(\alpha/2)$-quantile of t_{n-p}. By $(1-\alpha)$-confidence interval, we mean the following probabilistic statement is true:

$$\mathsf{P}\left(\beta_j \in \mathcal{CI}_j(\alpha)\right) = 1 - \alpha.$$

Indeed, this is true given any realization of X, and conditional coverage is stronger than unconditional coverage because $\mathsf{P}\left(\beta_j \in \mathcal{CI}_j(\alpha)\right) = \mathsf{E}\{\mathsf{P}\left(\beta_j \in \mathcal{CI}_j(\alpha) \mid X\right)\}$ by the law of total probability.

To construct a confidence region for the p-dimensional vector β, a simple approach is to take the product of univariate confidence intervals $\prod_{j=1}^p \mathcal{CI}_j(\alpha/p)$. (*Exercise:* Show that this set covers β with probability at least $1-\alpha$.) However, this product set is usually quite conservative because it does not take into account the dependence between the entries of $\hat{\beta}$. A better solution is to use the following pivotal quantity

$$\frac{(\hat{\beta}-\beta)^T(X^TX)(\hat{\beta}-\beta)}{p\hat{\sigma}^2} \sim F_{p,n-p}. \tag{1.11}$$

So the following ellipsoid is a $(1-\alpha)$-confidence set of β:

$$\mathcal{CS}(\alpha) = \left\{ \beta \in \mathbb{R}^p \;\middle|\; \frac{(\hat{\beta}-\beta)^T(X^TX)(\hat{\beta}-\beta)}{p\hat{\sigma}^2} \leq F_{p,n-p}(\alpha) \right\},$$

where $F_{p,n-p}(\alpha)$ is the upper α-quantile of $F_{p,n-p}$.

Exercise 1.14. Use (1.8) to construct a $(1-\alpha)$-confidence interval for σ^2.

Exercise 1.15. Let $(X^*, Y^*) \in \mathbb{R}^p \times \mathbb{R}$ be a new observation of the normal linear model. That is, suppose $Y^* = (X^*)^T\beta + \epsilon^*$ where $\epsilon^* \perp\!\!\!\perp (X, \epsilon, X^*)$ and $\epsilon^* \sim \mathrm{N}(0, \sigma^2)$. Construct a $(1-\alpha)$-confidence interval for $(X^*)^T\beta$ and Y^*. (The latter is called a $(1-\alpha)$-*prediction interval.*)

Hypothesis tests and analysis of variance

By using the duality between hypothesis testing and confidence interval, we can easily construct level-α tests for

$$H_0 : \beta_j = 0 \text{ vs. } H_1 : \beta_j \neq 0 \quad \text{and} \quad H_0 : \beta = 0 \text{ vs. } H_1 : \beta \neq 0.$$

That is, we reject $\beta_j = 0$ if $0 \notin \mathcal{CI}_j(\alpha)$ and reject $\beta = 0$ if $0 \notin \mathcal{CI}(\alpha)$.

More generally, we may be interested in comparing nested linear models. As before, consider the following partition

$$X = (X_0 \; X_1) \text{ and } \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix},$$

where $X_0 \in \mathbb{R}^{n \times p_0}$ and $\beta_0 \in \mathbb{R}^{p_0}$. We are interested in comparing the full model $\mu = X\beta$ with the submodel $\mu = X_0\beta_0$, which amounts to testing $H_0 : \beta_1 = 0$ vs. $H_1 : \beta_1 \neq 0$. The *(generalized) likelihood ratio statistic* is given by

$$\frac{\sup_{\beta \in \mathbb{R}^p} L(\beta, \sigma^2)}{\sup_{\beta_0 \in \mathbb{R}^{p_0}, \beta_1 = 0} L(\beta, \sigma^2)} = \left\{ 1 + \frac{\|(P - P_0)Y\|^2}{\|(I-P)Y\|^2} \right\}^{n/2}$$

Exercise 1.16. Prove the above equality.

Thus, the likelihood ratio test rejects $H_0 : \beta_1 = 0$ if $\|(P-P_0)Y\|^2 / \|(I-P)Y\|^2$ is large. To determine the critical value, we need to derive the distribution of the test statistic under $H_0 : \beta_1 = 0$, which is given by

$$F = \frac{\|(P-P_0)Y\|^2/(p-p_0)}{\|(I-P)Y\|^2/(n-p)} \sim F_{p-p_0, n-p} \text{ under } H_0.$$

So the level-α likelihood ratio test rejects $H_0 : \beta_1 = 0$ when $F > F_{p-p_0,n-p}(\alpha)$.

Exercise 1.17. Show that the null distribution of the F-statistic above in the normal linear model is $F_{p-p_0,n-p}$. [*Hint*: Show that $\|(P - P_0)Y\|^2 = \|(P - P_0)\epsilon\|^2$, then use Lemma 1.8.]

Note that $\|(I - P)Y\|^2$ is the residual sum of squares (RSS) of the full model, while $\|(P - P_0)Y\|^2$ is the reduction of RSS when we enlarge the submodel to the full model. This ratio has obvious geometric interpretations; see Figure 1.1.

Exercise 1.18. Show that the t-test and F-test for $H_0 : \beta_j = 0$ vs. $H_0 : \beta_j = 0$ are equivalent.

1.5 Practical 2: Data and normal linear models

1.5.1 *Data manipulation

Many functions in R expect that the dataset is stored in a `data.frame` object, which is essentially a `list` of equal-length vectors (possibly of heterogeneous types). We will learn about a package called `data.table` which provides a fast and modern implementation of the legacy `data.frame` data structure in R. To this end, you will first need to install and load the `data.table` package.

```
install.packages("data.table")
library(data.table)
```

One important advantage of `data.table` is that it is much faster.

```
> path <- "https://raw.githubusercontent.com/Rdatatable/data.table/"
> file <- "master/vignettes/flights14.csv"
> input <- paste0(path, file)
> system.time(flights <- read.csv(input))
   user  system elapsed
  0.657   0.023   1.143
> class(flights)
[1] "data.frame"
> system.time(flights <- fread(input))
 [100%] Downloaded 2193882 bytes...
   user  system elapsed
  0.127   0.028   0.340
> class(flights)
[1] "data.table" "data.frame"
```

The last line shows that `flights` is an object of class “data.table”, which inherits the “data.frame” class.

The `flights` dataset contains all flights that departed from New York City airports in January to October, 2014. You can get a sense about this dataset in various ways.

```
> dim(flights)
[1] 253316     11
> print(flights, 2)
        year month day dep_delay arr_delay carrier origin dest air_time
     1: 2014     1   1        14        13      AA    JFK  LAX      359
     2: 2014     1   1        -3        13      AA    JFK  LAX      363
    ---
253315: 2014    10  31        -4        15      MQ    LGA  DTW       75
253316: 2014    10  31        -5         1      MQ    LGA  SDF      110
        distance hour
     1:     2475    9
     2:     2475   11
    ---
253315:      502   11
253316:      659    8
> summary(flights)
      year          month             day          dep_delay
 Min.   :2014   Min.   : 1.000   Min.   : 1.00   Min.   :-112.00
 1st Qu.:2014   1st Qu.: 3.000   1st Qu.: 8.00   1st Qu.:   -5.00
 Median :2014   Median : 6.000   Median :16.00   Median :   -1.00
 Mean   :2014   Mean   : 5.639   Mean   :15.89   Mean   :  12.47
 3rd Qu.:2014   3rd Qu.: 8.000   3rd Qu.:23.00   3rd Qu.:  11.00
 Max.   :2014   Max.   :10.000   Max.   :31.00   Max.   :1498.00
   arr_delay          carrier             origin              dest
 Min.   :-112.000   Length:253316      Length:253316      Length:253316
 1st Qu.: -15.000   Class :character   Class :character   Class :character
 Median :  -4.000   Mode  :character   Mode  :character   Mode  :character
 Mean   :   8.147
 3rd Qu.:  15.000
 Max.   :1494.000
    air_time        distance         hour
 Min.   : 20.0   Min.   :  80   Min.   : 0.00
 1st Qu.: 86.0   1st Qu.: 533   1st Qu.: 9.00
 Median :134.0   Median : 944   Median :13.00
 Mean   :156.7   Mean   :1099   Mean   :13.06
 3rd Qu.:199.0   3rd Qu.:1416   3rd Qu.:17.00
 Max.   :706.0   Max.   :4983   Max.   :24.00
```

The `data.table` class implements an elegant syntax of data maniputation. The general form is `DT[i, j, by]`, which means "take `DT`, subset/reorder rows using `i`, then calculate `j`, grouped by `by`". For example, we can easily count the number of departure delays longer than 2 hours for each carrier in June, 2014 by

```
> flights[dep_delay > 120 & month == 6, table(carrier)]
carrier
 AA  AS  B6  DL  EV  F9  FL  MQ  UA  US  VX  WN
 76   2  80 109 213   2   4  24 139  26   4  63
```

We can find the five most delayed carrier-destination pairs that have at least 30 flights in total by

```
> delay_summary <- flights[, .(total = .N,
                               delayed = sum(dep_delay > 120),
                               percent = mean(dep_delay > 120),
                               mean = mean(dep_delay)),
                             by = .(carrier, dest)]
> delay_summary[total > 30][order(-percent)][1:5]
```

```
   carrier dest total delayed    percent     mean
1:      AA  EGE    85      11 0.12941176 43.32941
2:      WN  MSY   284      28 0.09859155 25.68662
3:      EV  CHO   124      12 0.09677419 26.16129
4:      WN  CAK   365      33 0.09041096 31.77808
5:      EV  BGR   274      22 0.08029197 22.54380
```

This should be enough for us for now, but you are encouraged to follow the official introduction to data.table (https://cran.r-project.org/web/packages/data.table/vignettes/datatable-intro.html) to learn more about it.

1.5.2 Data visualization

For simple and quick visualization, you can use the plot, hist, and boxplot functions in base R. Try the following

```
x <- rnorm(100); y <- x + rnorm(100); plot(x, y)
hist(flights$air_time)
boxplot(air_time ~ carrier, flights)
```

The last command requires a bit of explanation. The boxplot function can take in a vector or a "formula" (such as y $\sim$ grp where the numeric vector y is split into groups according to grp). Boxplot is a powerful way to visualize the distribution of a continuous variable, see Figure 1.2.[14]

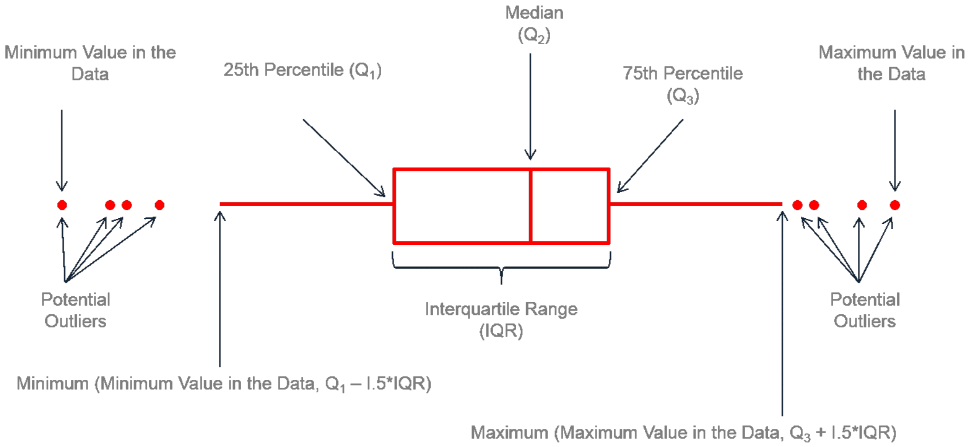

Median
(Q_2)
Minimum Value in the Data
25th Percentile (Q_1)
75th Percentile
(Q_3)
Maximum Value in the Data
Potential Outliers
Interquartile Range
(IQR)
Potential Outliers
Minimum (Minimum Value in the Data, Q_1 – 1.5*IQR)
Maximum (Maximum Value in the Data, Q_3 + 1.5*IQR)

Figure 1.2: A boxplot.

The ggplot2 package and extensions provide tools to generate production-quality figures using human-friendly syntax. We provide one example here to give you a taste; there are many amazing tutorials online and you can also easily get help from Google/Stack Overflow/LLMs.

```
> # install.packages("ggplot2") # uncomment if ggplot2 is not installed yet
> library(ggplot2)
> pl <- ggplot(delay_summary[total > 30],
               aes(x = percent, y = mean, col = carrier, size = total))
> pl <- pl + geom_point(alpha = 0.5) + theme_minimal(base_size = 15)
> pl <- pl + scale_x_continuous(labels = scales::percent)
> pl <- pl + xlab("Delay > 2 hours") + ylab("Average delay (in minutes)")
> pl
```

The output can be found in Figure 1.3.

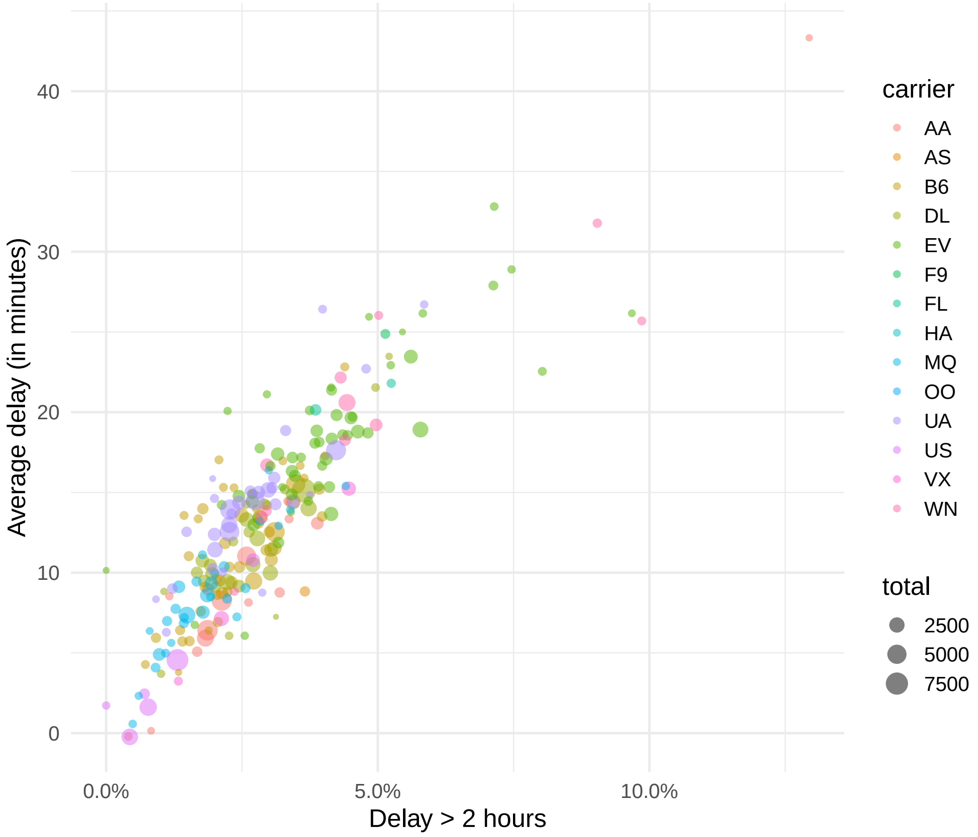


Figure 1.3: Flight delay summary.

1.5.3 Normal linear models

Let us learn about fitting normal linear models in R using some simulated data.

```
set.seed(42)
n <- 50
a <- rnorm(n)
```

```
b <- rnorm(n)
y <- a - b + 2 * rnorm(n)
data <- data.frame(y = y, x1 = a, x2 = b)
```

The main function for normal linear model in R is `lm`.

```
lm(formula, data, subset, weights, na.action,
   method = "qr", model = TRUE, x = FALSE, y = FALSE, qr = TRUE,
   singular.ok = TRUE, contrasts = NULL, offset, ...)
```

You can directly use `lm` on the global variables `a`, `b`, `y`, but you are recommended to use the `data` argument which needs to be a `data.frame` object. The following function fits a normal linear model and outputs some important summary statistics.

```
> fit1 <- lm(y ~ x1 + x2, data)
> summary(fit1)

Call:
lm(formula = y ~ x1 + x2, data = data)

Residuals:
    Min      1Q  Median      3Q     Max
-3.4578 -0.8958 -0.1260  0.5328  6.1281

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.3005     0.2680  -1.122   0.2677
x1            1.1764     0.2406   4.888 1.23e-05 ***
x2           -0.9570     0.2996  -3.194   0.0025 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.883 on 47 degrees of freedom
Multiple R-squared:  0.484,     Adjusted R-squared:  0.462
F-statistic: 22.04 on 2 and 47 DF,  p-value: 1.767e-07
```

The `summary` function is very useful and adapts to the object class. (When applying the `summary` method to `fit1` which is a `lm` object, R automatically calls `summary.lm`; try `?summary.lm`.)

Here are what the numbers in the output mean:

- Below `Residuals:` are the five-number summary of the residuals R.
- The `Estimate` column contains the OLS estimator $\hat{\beta}_j, j = 1, \ldots, p$.
- The `Std. Error` column contains the estimated standard errors $\hat{\sigma}\sqrt{(X^T X)^{-1}_{jj}}$.

- The `t value` column contains the t-statistics, which is simply the ratio between the previous columns.
- The `Pr(>|t|)` column contains the p-values for the two-sided t-tests.
- The `Residual standard error` is simply $\hat{\sigma}$, with degrees of freedom $n - p$.
- The `Multiple R-squared` refers to the so-called "coefficient of determination", which is defined as the variance explained by the regressors

$$R^2 = \frac{\|\hat{\mu} - \bar{Y}1\|^2}{\|Y - \bar{Y}1\|^2} = 1 - \frac{\|Y - \hat{\mu}\|^2}{\|Y - \bar{Y}1\|^2},$$

  where $\bar{Y} = \sum_{i=1}^{n} Y_i/n$. The `Adjusted R-squared` is a less biased estimator of the population R^2 and is defined as (you don't need to remember this)

$$R^2_{\text{adj}} = 1 - (1 - R^2)\frac{n-1}{n-p-1}.$$

- The `F-statistic` is the F-statistic with respect to the submodel in which the coefficients of all non-intercept terms are 0, which has degrees of freedom $p - 1$ and $n - p$. The `p-value` is the p-value for the corresponding F-test.

You can use `anova` for analysis of variance. Here is a simple example.

```
> fit2 <- lm(y ~ x1, data)
> anova(fit2, fit1)
Analysis of Variance Table

Model 1: y ~ x1
Model 2: y ~ x1 + x2
  Res.Df    RSS Df Sum of Sq      F   Pr(>F)
1     48 202.93
2     47 166.73  1      36.2 10.205   0.0025 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

1.5.4 Exercises

Exercise 1.19. Explain why the p-value in the `anova` output (0.002502) is exactly the same as the p-value in the `summary.lm` output for `x2`.

Exercise 1.20. You can fit an intercept-only model using `fit3 <- lm(y ~ 1, data)`. Without executing `anova(fit3, fit1)`, can you guess what its output will be based on the output `summary.lm`?

Exercise 1.21. Use `R` to validate Proposition 1.11.

Exercise 1.22. Without calling `lm` or using an LLM,[15] write a function called `mylm` with arguments `X` ($n \times p$ model matrix) and `Y` (n-vector) that outputs all the numbers reported by `summary.lm`. Test your code using the example above.

Exercise 1.23. Modify your `mylm` function so that it also accepts a logical vector `S0` of length p and performs the analysis of variance test for the sub-model that only uses the regressors `X0 <- X[, S0]`. Compare your output with `anova`.

1.6 Basic asymptotic statistics

The theory of normal linear model is beautiful but relies quite heavily on properties of the multivariate normal distribution. We will introduce many relaxations of the normal linear model in this course. In general, we cannot obtain exact inference for parameters in those models, but it is often possible to obtain approximate inference when the sample size n is large. As the focus of this course is on statistical modelling instead of statistical theory, we will only state the relevant results in asymptotic statistics (the proofs are covered in *Principles of Statistics*) and discuss how to use them in statistical modelling.

Asymptotic statistics (or large-sample theory) is primarily based on two fundamental results in probability theory: the law of large numbers and the central limit theorem. We will state the versions in IA *Probability*. Let $X_1, X_2, \dots, X_n, \dots$ be a sequence of i.i.d. random variables and let $\bar{X}_n = (X_1 + \dots X_n)/n$. The *weak law of large numbers* says that if $\mu = \mathsf{E}(X_1)$ exists, then as $n \to \infty$,

$$\bar{X}_n \xrightarrow{p} \mu,$$

where $\xrightarrow{p}$ means "convergence in probability", that is, $\mathsf{P}(|\bar{X}_n - \mu| > \epsilon) \to 0$ for all $\epsilon > 0$. The *central limit theorem* says that if in addition $\sigma^2 = \mathsf{Var}(X_1)$ exists, then as $n \to \infty$,

$$\frac{\sqrt{n}(\bar{X}_n - \mu)}{\sigma} \xrightarrow{d} \mathrm{N}(0,1),$$

where $\xrightarrow{d}$ means "convergence in distribution", that is, $\mathsf{P}(\sqrt{n}(\bar{X}_n - \mu)/\sigma \leq z) \to \Phi(z)$ for all $z \in \mathbb{R}$ where Φ is the CDF of $\mathrm{N}(0,1)$. A less precise but more intuitive way to state this is $\bar{X}_n \overset{\cdot}{\sim} \mathrm{N}(\mu, \sigma^2/n)$ where $\overset{\cdot}{\sim}$ means "approximate distribution".

One of the most important results in asymptotic statistics is the following.

Theorem 1.24 (Informal). *If $\mathbb{P}$ is a "regular" parametric model and the data $X_1, \dots, X_n$ are generated i.i.d. from P_θ, then the maximum likelihood estimator*

(i) exists;

(ii) is consistent in the sense that $\hat{\theta}_{MLE} \to \theta$ in probability as $n \to \infty$;

(iii) is asymptotically normal in the sense that

$$\sqrt{n}(\hat{\theta}_{MLE} - \theta) \xrightarrow{d} N(0, \{I(\theta)\}^{-1}),$$

where $I(\theta) = \mathsf{Var}_\theta(\nabla_\theta l(\theta))$ is called the Fisher information matrix *and $l(\theta) = \log L(\theta)$ is the log-likelihood function of a single observation X_1. Moreover, $I(\theta)^{-1}$ is the "best possible asymptotic variance" for any "regular" estimator of θ.*

There is a fanscinating history about how this result was conceived, formalized, and proved.[16] We will not attempt to prove it, but our investigation of exponential family in Chapter 3 will provide some insights about this deep result.

The next two lemmas about convergence of random variables are often useful.

Lemma 1.25 (Slutsky's lemma)**.** *Suppose $A_n \xrightarrow{d} A$ and $B_n \xrightarrow{p} b$ where b is a constant. Then*

(i) $A_n + B_n \xrightarrow{d} A + b$*;*

(ii) $A_n B_n \xrightarrow{d} Ab$*;*

(iii) $A_n / B_n \xrightarrow{d} A/b$ *if* $b \neq 0$.

This can be easily extended to multivariate settings. This result is a corollary of the *continuous mapping theorem*, which says if g is a continuous function, then $X_n \to X$ implies $g(X_n) \to g(X)$ where $\to$ can be almost sure convergence, convergence in probability, or convergence in distribution (but the two $\to$ need to be the same).

Lemma 1.26 (Delta method)**.** *Suppose*

$$\sqrt{n}(\hat{\eta} - \eta) \xrightarrow{d} N(0, \tau^2),$$

where τ^2 may depend on η and $g(\eta)$ is continuously differentiable at η. Then

$$\sqrt{n}\left\{g(\hat{\eta}) - g(\eta)\right\} \xrightarrow{d} N\left(0, \tau^2 g'(\eta)^2\right).$$

This result can be proved by considering the Taylor expansion of $g(\hat{\eta})$ at η and applying Slutsky's lemma. It can be easily extended to multivariate settings.

For this course, you only need to know how to apply Slutsky's lemma and the delta method. An immediate application of Slutsky's lemma at this point is that

$$\sqrt{n} I(\hat{\theta})^{1/2}(\hat{\theta}_{\mathrm{MLE}} - \theta) \xrightarrow{d} \mathrm{N}(0, I).$$

This allows us to use

$$\hat{\theta}_{\mathrm{MLE}} \dot{\sim} \mathrm{N}(\theta, \{I(\hat{\theta})\}^{-1}/n)$$

to construct asymptotic tests and confidence intervals. For example, a (pointwise[17]) asymptotic $(1-\alpha)$-confidence interval for θ_j is

$$\mathcal{CI}_j(\alpha) = \left[\hat{\theta}_j - \hat{\sigma}\sqrt{(X^T X)^{-1}_{jj}} z(\alpha/2), \hat{\theta}_j + \hat{\sigma}\sqrt{(X^T X)^{-1}_{jj}} z(\alpha/2)\right],$$

where $z(\alpha/2)$ is the upper $(\alpha/2)$-quantile of $\mathrm{N}(0,1)$, that is,

$$\lim_{n\to\infty} \mathsf{P}_\theta\left(\theta_j \in \mathcal{CI}_j(\alpha)\right) \to 1 - \alpha \quad \text{as } n \to \infty \text{ for all } \theta \in \Theta. \tag{1.12}$$

The last useful asymptotic result for us is Wilks' theorem.

Theorem 1.27 (Informal)**.** *Consider testing* $H_0 : \theta \in \Theta_0$ *vs.* $H_1 : \theta \in \Theta \setminus \Theta_0$ *in parametric models using the generalized likelihood ratio statistic* Λ *in* (1.1)*. Under regularity conditions (smoothness of the likelihood function), if* $\Theta_0 \subset \Theta$ *are nested linear spaces, then*

$$2 \log \Lambda \xrightarrow{d} \chi^2_{\dim(\Theta)-\dim(\Theta_0)} \quad \textit{as } n \to \infty,$$

under P_θ *for any* $\theta \in \Theta_0$*.*

Notes

[1] Perhaps this is so obvious because for most people mathematics and statistics are both about "numbers". When I told my neighbours that I am a statistician, most of their first reaction is that I do mathematics.

[2] Mathematics also involves induction, see G. Pólya's book *Mathematics and Plausible Reasoning*, but mathematical induction is a deductive method. Statistics also involves deductive reasoning (which is basically mathematical statistics).

[3] Box, G. E. P. (1957). Abstracts. *Biometrics*, *13*(2), 238–246.

[4] Breiman, L. (2001). Statistical modeling: The two cultures. *Statistical Science*, *16*(3), 199–231.

[5] The notation here has the risk of being eccentric because many authors use $\mathbb{P}$ to denote a probability measure. The problem with that is it encourages one to think the probability measure as something fixed, which is very dangerous for statisticians. The blackboard bold font is commonly used to denote "special sets", such as $\mathbb{R}$, $\mathbb{C}$, etc. In statistics, the special set is often the statistical model—a collection of probability distributions. So in this sense it is appropriate to denote a statistical model by $\mathbb{P}$, and it is even better when someone is confused about what $\mathbb{P}$ means because that forces them to think about what model is being used.

[6] I received complaints about also using P to denote CDF in IB *Statistics*, so I am switching to the conventional notation in this course.

[7] In IA *Probability*, expectation is defined as a unique linear functional on random variables on a probability space that satisfies certain properties. Some people actually prefer to define probability via expectation, with good reasons; see Whittle, P. (2000). *Probability via expectation.* Springer. De Finetti would even use P to also denote the expectation operator. We will not be a zealot about this.

[8] It is possible to define multivariate normal with a degenerate covariance matrix; see Rao, C. R. (1973). *Linear statistical inference and its applications* (2nd ed.). Wiley, Chapter 8.

[9] Named after the statistician W. S. Gosset who used the pseudonym "Student" to publish his method.

[10] Equation (1.2) does not necessarily describe the causal relationship between X_i and Y_i. That is, if an external force sets X_i to x_i (instead of its "natural" value), (1.2) does not make any assumptions on what the resulting Y_i would become. In contrast, a linear structural equation model assumes that the counterfactual value of Y_i, often denoted as $Y_i^{(x_i)}$, is $x_i^T \beta + \epsilon_i$. What has confused generations of statisticians and scientists is that many people also use (1.2) to indicate a linear structural equation model. For more discussion on the distinction between regression and causation, see Section 6.4 of Freedman's book.

[11] This is why some texts assume X is "fixed". A better way to think about this is that the inference for the normal linear model (1.2) is conditional on the model matrix X. This is an instance of the *conditionality principle*, which says that the unconditional distribution and the conditional distribution given an ancillary statistic (X in this case) carry the same information for statistical inference.

[12] In econometrics, this result is known as the Frisch–Waugh–Lovell theorem.

[13] If we are more rigorous, we should write the estimator of σ^2 as $\widehat{\sigma^2}$ instead of $\hat{\sigma}^2$. But it is widely understood that we estimate σ by first estimating σ^2 and $\hat{\sigma}^2$ does not mean "$\hat{\sigma}$ square". Notice that unbiasedness of $\hat{\sigma}^2$ does not translate to unbiasedness of $\sqrt{\hat{\sigma}^2}$ as an estimator of σ.

[14] Image source: https://leansigmacorporation.com/box-plot-with-minitab/.

[15] Do we still need to learn programming with the rise of LLMs? I asked the same question to an AI and got the following code https://g.co/gemini/share/add2a2a9cad6. There are at least two substantial mistakes in the examples that accompany the funtion written by the AI. Can you find them?

[16] Stigler, S. M. (2007). The epic story of maximum likelihood. *Statistical Science*, *22*, 598–620.

[17] Results like (1.12) are sometimes unsatisfactory in practice because it does not tell us how large the sample size n needs to be for the coverage probability to be sufficiently close to $1-\alpha$. Moreover, (1.12) does not exclude the possibility that the confidence interval may have bad coverage probability (e.g. less than $1-2\alpha$) at *some* θ for any sample size n. This is why some people prefer *uniform* asymptotic confidence sets S that satisfy

$$\liminf_{n\to\infty} \inf_{\theta\in\Theta} \mathsf{P}_\theta\left(\theta \in S\right) \geq 1-\alpha \quad \text{as } n \to \infty.$$

Chapter 2

Advanced linear models

2.1 Linear model diagnostics

Our setting in this Chapter is similar to that in Section 1.4: we observe some data $(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n) \in \mathbb{R}^p \times \mathbb{R}^1$ and would like to model $\mu_i = \mathsf{E}(Y_i \mid X_i)$ as a linear function of X_i. As discussed in Section 1.4, the normal linear model (1.3) contains three assumptions:

(i) the conditional expectation follows a linear model $\mu = X\beta$;

(ii) the noise ϵ is independent of X;

(iii) the noise $\epsilon \sim \mathrm{N}(0, \sigma^2 I_n)$.

These assumptions are essential in the exact statistical inference discussed in Section 1.4.3 but are often too restrictive in practice. One nice thing about making restrictive assumptions is that we can often check them empirically. Here we provide some useful diagnostic quantities and plots for the normal linear model.

Let $P = X(X^T X)^{-1} X^T$ denote the projection matrix onto the column space of X. The *leverage* of the ith observation is defined as P_{ii}, the ith diagonal element of the hat matrix. Recall that the fitted value for Y_i is

$$\hat{\mu}_i = (PY)_i = P_{ii} Y_i + \sum_{k \neq i} P_{ik} Y_k.$$

So the leverage P_{ii} measures how much the fitted value $\hat{\mu}_i$ is determined by the observed value Y_i. Another motivation for leverage is the following result (recall $R = Y - \hat{\mu}$ is the vector of residuals)

$$\mathsf{Var}(R_i \mid X) = \sigma^2 (1 - P_{ii}). \tag{2.1}$$

So the residual R_i is close to 0 if the leverage P_{ii} is close to 1. Motivated by (2.1), the *studentized* or *standardized residual* of the ith observation is defined as

$$\tilde{R}_i = \frac{R_i}{\hat{\sigma}\sqrt{1 - P_{ii}}}.$$

Exercise 2.1. Prove (2.1).

Exercise 2.2. In the normal linear model, show that $\tilde{R}_i \sim t_{n-p-1}$ if we replace $\hat{\sigma}$ in the definition of $\tilde{R}_i$ by $\hat{\sigma}_{(-i)}$, which is the estimator of σ using all observations besides (X_i, Y_i).

Next we describe the diagnostic plots produced by the R function `plot.lm` by default.

The first is the *residual vs. fitted* plot, which plots the studentized residual $\tilde{R}_i$ against the predicted value $\hat{\mu}_i$. We can visually assess the linearity assumption by looking for apparent trends (e.g. a quadratic trend) in the plot.

The second is the *quantile-quantile (Q-Q) plot*, which is used to visually check normality of the noise ϵ_i. If the normal linear model is correct, $\tilde{R}_i$ should be close to ϵ_i/σ, which follows a standard normal distribution. We may check this assumption by plotting the sample quantiles of $(\tilde{R}_1, \ldots, \tilde{R}_n)$ against the theoretical quantiles of $\mathrm{N}(0, 1)$; see Figure 2.1 for an illustration.

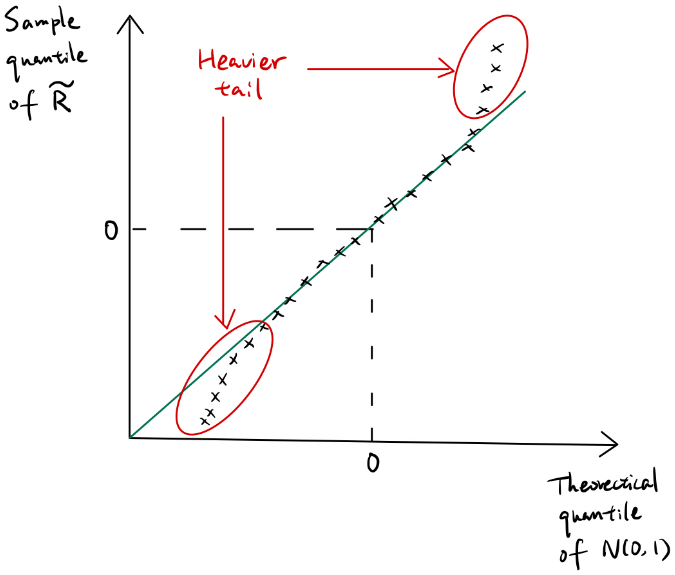


Figure 2.1: Quantile-quantile (Q-Q) plot.

The third diagnostic plot is the *scale-location plot*, which shows the square root of the absolute value of the standardized residual $\sqrt{|\tilde{R}_i|}$ against the fitted value $\hat{\mu}_i$. This plot is used to check the homoscedasticity assumption $\mathsf{Var}(\epsilon_i \mid X_i) = \sigma^2$ (see Section 2.2.2 below), under which $\sqrt{|\tilde{R}_i|}$ should have an average value around 1.

The fourth and final one is a plot of *residuals vs. leverage*. More precisely, this plot shows $\tilde{R}_i$ against P_{ii} and is used to identify outliers with a large leverage. We say an observation (X_i, Y_i) is an *outlier* if $|R_i|$ is much larger than what is expected if $\epsilon_i \sim \mathrm{N}(0, \sigma^2)$. In other words, these observations differ substantially from model-predicted values. Especially of concern are outliers with a high leverage, because just one or a few of them can severely bias a regression model. (Note that the definition of "outlier" depends on the model.) It is not rare to have one observation that is not an outlier originally become an outlier when some other apparently outlying observations are removed. See Figure 2.2 for an illustration.

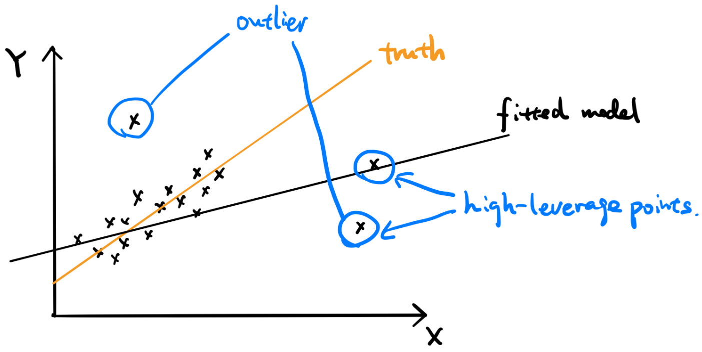


Figure 2.2: Outliers with a high leverage can severely bias a regression model.

A useful quantity for outlier detection is Cook's distance:

$$D_i = \frac{\|X(\hat{\beta} - \hat{\beta}_{(-i)})\|^2}{p\hat{\sigma}^2} = \frac{1}{p}\frac{P_{ii}}{1 - P_{ii}}\tilde{R}_i^2, \tag{2.2}$$

where $\hat{\beta}_{(-i)}$ is the "leave-one-out" OLS estimator of β when (X_i, Y_i) is removed from the dataset. By definition, D_i is a standardized change of the fitted values when the ith observation is removed. Therefore, a large value of D_i indicates that the ith observation have a large influence on the fitted values. Some clever algebra produces the second equality in (2.2), so in order to compute Cook's distance, it is unnecessary to repeatedly solve least squares problems.

Exercise 2.3. Prove the second equality in (2.2) using the Sherman-Morrison formula

$$(A + uv^T)^{-1} = A^{-1} - \frac{A^{-1}uv^TA^{-1}}{1 + v^TA^{-1}u},$$

which holds for any invertible $A \in \mathbb{R}^{p\times p}$ and $u, v \in \mathbb{R}^p$ such that $v^TA^{-1}u \neq -1$.

Recall from Section 1.4.3 that a $(1-\alpha)$-confidence ellipsoid for β in the normal linear model is given by

$$\mathcal{CS}(\alpha) = \left\{ \beta \in \mathbb{R}^p \;\middle|\; \frac{\|X(\hat{\beta} - \beta)\|^2}{p\hat{\sigma}^2} \leq F_{p,n-p}(\alpha) \right\},$$

Motivated by this, a rule of thumb is that a Cook's distance $D_i > F_{p,n-p}(0.5)$ indicates an outlier of concern.

As a final remark on model diagnostics, the above quantities and plots should be regarded as visual “falsification tests” of the various assumptions made by the normal linear model. This means that even when all the diagnostic plots look exactly like what are expected, we cannot conclude that the linear model must be correct. These tools depart from rigorous theorems in mathematical statistics but are immensely useful in practice. They provide empirical evidence to improve a statistical model and fit in nicely with Box’s cycle of scientific research discussed in Section 1.1.

2.2 Linear conditional expectation models

We next discuss several possible relaxations of the standard normal linear model.

2.2.1 Generalized least squares

One possible relaxation is to assume ϵ follows an non-isotropic normal distribution:

$$Y = X\beta + \epsilon, \ \epsilon \mid X \sim \mathrm{N}(0, \sigma^2 \Sigma), \tag{2.3}$$

where $\sigma^2 \in \mathbb{R}$ is unknown and $\Sigma \in \mathbb{R}^{n \times n}$ is a known positive-definite matrix that may depend on X.

Theoretical results in Sections 1.4.2 and 1.4.3 can be easily extended to this model by using the transformation

$$(X, Y) \to (\Sigma^{-1/2} X, \Sigma^{-1/2} Y).$$

The maximum likelihood estimator of β in this model is given by the *generalized least squares* (GLS) estimator:

$$\hat{\beta}_{\mathrm{GLS}} = (X^T \Sigma^{-1} X)^{-1} X^T \Sigma^{-1} Y.$$

Here we assume $X^T \Sigma^{-1} X$ is invertible.

Exercise 2.4. Derive the above formula for $\hat{\beta}_{\mathrm{GLS}}$ by maximizing the likelihood function for the model (2.3). Then derive it again using the transformation above.

An important special case of GLS is the *weighted least squares* (WLS). Given a vector of weights $w = (w_1, \dots, w_n)$, the WLS estimator is given by

$$\hat{\beta}_{\mathrm{WLS}} = \arg\min_{\beta} \sum_{i=1}^{n} w_i (Y_i - X_i^T \beta)^2.$$

This is equivalent to choosing $\Sigma = \mathrm{diag}(w_1^{-1}, \dots, w_n^{-1})$ in GLS. In R, you can obtain the WLS estimator by using the `weights` argument in the `lm` method.

2.2.2 Heteroscedasticity

Consider the following less restrictive linear model:

$$Y_i = X_i^T \beta + \epsilon_i, \ i = 1, \dots, n,$$

where

- $(\epsilon_i, X_i), \ i = 1, \dots, n,$ are i.i.d.;
- $\mathsf{E}(\epsilon_i \mid X_i) = 0$;
- $\mathsf{Var}(\epsilon_i \mid X_i) = \sigma^2(X_i)$.

Exercise 2.5. Write down this model as a collection of probability distributions of $(X_1, Y_1), \dots, (X_n, Y_n)$.

Compared to the model in (2.3), it no longer assumes $\epsilon_i \perp\!\!\!\perp X_i$, the distribution of ϵ_i is normal, or the variance of ϵ_i is known (up to a multiplicative constant). When $\sigma^2(X_i) = \sigma^2$ is a constant, we say the noise is *homoscedastic*; otherwise, we say the noise is *heteroscedastic*.

Due to the lack of distributional assumptions, exact statistical inference is no longer possible. However, we can rely on asymptotic arguments. Let $\hat{\beta} = (X^T X)^{-1} X^T Y$ be the OLS estimator, then

$$\begin{aligned}\sqrt{n}(\hat{\beta} - \beta) &= \sqrt{n}\left\{(X^T X)^{-1} X^T Y - \beta\right\} \\ &= \sqrt{n}\left\{(X^T X)^{-1} X^T (X\beta + \epsilon) - \beta\right\} \\ &= \sqrt{n}(X^T X)^{-1} X^T \epsilon \\ &= \left(\frac{1}{n}\sum_{i=1}^n X_i X_i^T\right)^{-1} \left(\frac{1}{\sqrt{n}}\sum_{i=1}^n X_i \epsilon_i\right).\end{aligned}$$

Under suitable regularity conditions, the first term converges in probability to $\Sigma_X = \mathsf{E}[X_i X_i^T]$ by the weak law of large numbers, and the second term converges in distribution to $\mathrm{N}(0, \Omega)$, where $\Omega = \mathsf{E}(X_i X_i^T \epsilon_i^2) = \mathsf{E}\{\sigma^2(X_i) X_i X_i^T\}$. Therefore, by Slutsky's lemma,

$$\sqrt{n}(\hat{\beta} - \beta) \xrightarrow{d} \mathrm{N}\left(0, \Sigma_X^{-1} \Omega \Sigma_X^{-1}\right), \text{ as } n \to \infty. \tag{2.4}$$

The form of matrix $\Sigma_X^{-1} \Omega \Sigma_X^{-1}$ is common in misspecified maximum likelihood and is often called the *sandwich variance* (for the obvious reason) or the inverse *Godambe information matrix*. When the noise is homoscedastic, i.e. $\sigma^2(X_i) = \sigma^2$, (2.4) reduces to

$$\sqrt{n}(\hat{\beta} - \beta) \xrightarrow{d} \mathrm{N}(0, \sigma^2 \Sigma_X^{-1}),$$

which is consistent with the exact distribution (1.9) obtained under normality after taking $n \to \infty$.

Equation (2.4) is not an (asymptotic) pivotal quantity yet because the distribution depends on Σ_X and Ω. These unknown quantities can be estimated by

$$\hat{\Sigma}_X = \frac{1}{n}\sum_{i=1}^n X_i X_i^T \quad \text{and} \quad \hat{\Omega} = \frac{1}{n}\sum_{i=1}^n X_i X_i^T R_i^2, \text{ where } R_i = Y_i - X_i^T \hat{\beta}.$$

Under suitable regularity conditions, they converge to Σ_X and Ω in probability. By Slutsky's lemma,

$$\sqrt{n}\hat{\Sigma}_X \hat{\Omega}^{-1/2}(\hat{\beta} - \beta) \xrightarrow{d} \mathrm{N}(0, I_p), \text{ as } n \to \infty.$$

It is then straightforward to construct confidence intervals or hypothesis tests for β.

Exercise 2.6. Suppose $X_i \in \mathbb{R}$ and we know $\sigma^2(X_i) = \sigma^2(1 + \eta X_i^2)$ for some unknown $\sigma^2, \eta > 0$. Could you find a more efficient estimator of β by first giving an estimator of (σ^2, η)?

2.2.3 Misspecified conditional expectation

One may further question the validity of the linear model $\mu = X\beta$ itself. To emphasize that the linear model could be misspecified, we sometimes call $\mu = X\beta$ a linear *working model.* But what does β mean when the working model is incorrect?

Suppose $(X_1, Y_1), \dots, (X_n, Y_n) \in \mathbb{R}^p \times \mathbb{R}$ are i.i.d. Recall that the OLS estimator $\hat{\beta}$ minimizes $(1/n)\sum_{i=1}^n (Y_i - X_i^T\beta)^2$. Therefore, it is expected that $\hat{\beta}$ will converge (e.g. in probability), as $n \to \infty$, to

$$\beta_{\text{OLS}} = \arg\min_{\beta} \mathsf{E}\left\{(Y_i - X_i^T\beta)^2\right\} = \{\mathsf{E}(X_iX_i^T)\}^{-1}\,\mathsf{E}(X_iY_i). \tag{2.5}$$

Exercise 2.7. Prove the second equality in (2.5) by assuming that you can exchange derivative and expectation.

Here is another way to understand β_{OLS}. Let $\mu(X_i) = \mathsf{E}(Y_i \mid X_i)$ and $\epsilon_i = Y_i - \mu(X_i)$, so $\mathsf{E}(\epsilon_i \mid X_i) = 0$. Then

$$\begin{aligned}\beta_{\text{OLS}} &= \arg\min_{\beta} \mathsf{E}\left\{(Y_i - X_i^T\beta)^2\right\}\\ &= \arg\min_{\beta} \mathsf{E}\left\{(\mu(X_i) - X_i^T\beta + \epsilon_i)^2\right\}\\ &= \arg\min_{\beta} \mathsf{E}\left\{(\mu(X_i) - X_i^T\beta)^2\right\} + \underbrace{\mathsf{E}\left\{(\mu(X_i) - X_i^T\beta)\epsilon_i\right\}}_{\text{=0 (by Law of Total Expectation)}} + \underbrace{\mathsf{E}(\epsilon_i^2)}_{\text{=constant}}\\ &= \arg\min_{\beta} \mathsf{E}\left\{(\mu(X_i) - X_i^T\beta)^2\right\}.\end{aligned}$$

Therefore, $X_i^T\beta_{\text{OLS}}$ may be viewed as the projection of $\mu(X_i)$ onto the space of linear functions of X_i.

We make two remarks on misspecified linear models. First, the “best approximation” parameter β_{OLS} depends on the distribution of X; see Figure 2.3 for an illustration. Second, the definition of the population regression coefficient β also generally depends on the estimator we use. For example, the *least absolute deviation* (LAD) estimator[1]

$$\hat{\beta}_{\text{LAD}} = \arg\min_{\beta} \sum_{i=1}^n |Y_i - X_i^T\beta|$$

does not converge to β_{OLS} in general.

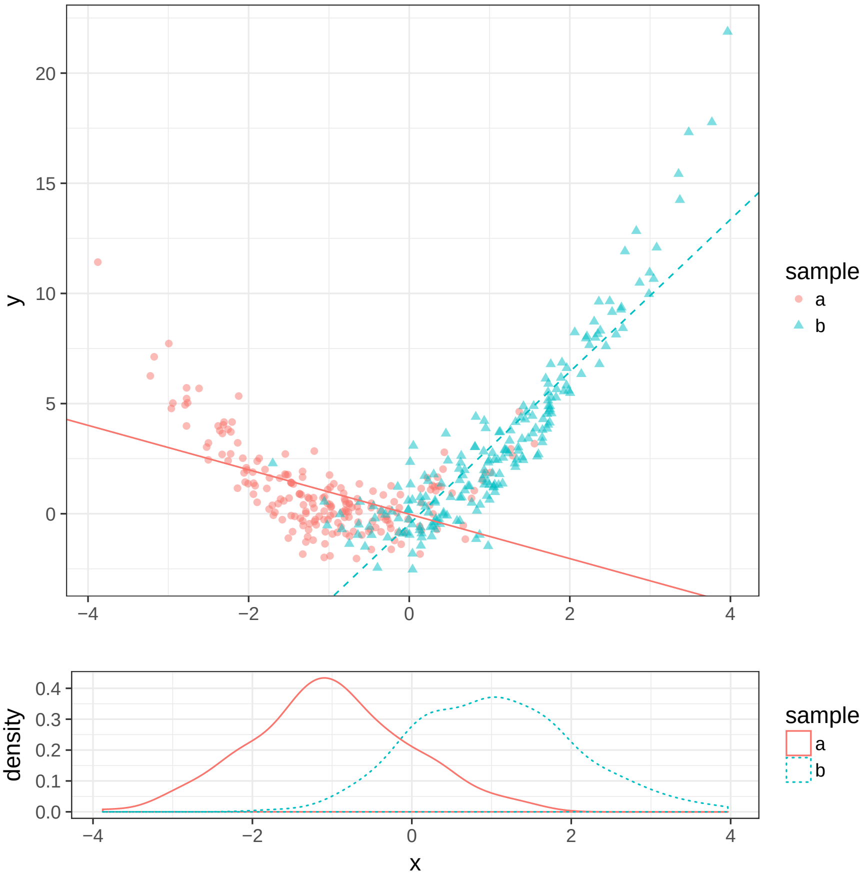


Figure 2.3: The "true" value of β_{OLS} depends on the distribution of the regressors. This figure shows two samples generated from the same conditional distribution of Y_i given X_i but different marginal distributions of X_i. In both samples, $Y_i = X_i^2 + X_i + \epsilon_i$ where $\epsilon_i \sim \mathrm{N}(0, 1)$. In sample a, $X_i \sim \mathrm{N}(-1, 1)$; in sample b, $X_i \sim \mathrm{N}(1, 1)$. The value of β_{OLS} is negative in sample a but positive in sample b.

2.3 Model selection

Although the normal linear model makes several restrictive assumptions, it remains the default choice for many applications used due to its simplicity. In practice, a common task is to select a linear working model according to one or some of the following criteria:

(i) Does the model appears to provide a good fit to the observed data?

(ii) How interpretable is the model?

(iii) How large is the model's prediction error?

(iv) How likely is the true model covered, assuming the data are indeed generated from it?

Although answers to the first two questions are going to be subjective, we can gain considerable insights about statistical modelling by trying to answer the other two questions.

2.3.1 The bias-variance decomposition

We first consider the prediction error of a linear working model when it is possibly misspecified. Consider the so-called *nonparametric regression* model:

$$Y_i = \mu(X_i)+\epsilon_i, \quad (X_i, \epsilon_i) \text{ are } \textit{i.i.d.},\ \epsilon_i \perp\!\!\!\perp X_i,\ \mathsf{E}(\epsilon_i) = 0,\ \mathsf{Var}(\epsilon_i) = \sigma^2,\ i = 1, \ldots, n. \tag{2.6}$$

In Section 2.2.3, we saw that the OLS estimator $\hat{\beta}_{\text{OLS}}$ estimates β_{OLS}, the projection of $\mu(X_i)$ onto the space of linear functions of X_i in the population. Here we additionally assume $\epsilon_i = Y_i - \mu(X_i)$ is independent of X_i to simplify the calculations below.

Let $\beta_n = \mathsf{E}(\hat{\beta} \mid X_1, \ldots, X_n)$ be the expected value of any estimator $\hat{\beta}$. Notice that β_n depends on the model matrix X, so β_n is generally a random quantity and this is why a subscript n is included. But we will treat $X_1, \ldots, X_n$ (and thus β_n) as fixed in the calculations below.

Let (X_{n+1}, Y_{n+1}) be a new independent observation from the same distribution. The *mean squared prediction error* (MSPE) at a fixed value $x \in \mathbb{R}^n$ of the regressors is defined as[2]

$$\begin{aligned}
\mathrm{MSPE}(x) &= \mathsf{E}\left[\{Y_{n+1} - X_{n+1}^T\hat{\beta}\}^2 \mid X_{n+1} = x\right] \\
&= \mathsf{E}\left[\{\mu(x) - x^T\hat{\beta} + \epsilon_{n+1}\}^2\right] \\
&= \mathsf{E}\left[\{\mu(x) - x^T\hat{\beta}\}^2\right] + \underbrace{\mathsf{E}\left[\{\mu(x) - x^T\hat{\beta}\}\epsilon_{n+1}\right]}_{=0 \text{ because } \epsilon_{n+1} \perp\!\!\!\perp \hat{\beta} \text{ and } \mathsf{E}[\epsilon_{n+1}]=0.} + \mathsf{E}(\epsilon_{n+1}^2) \\
&= \mathsf{E}\left[\{\mu(x) - x^T\beta_n + x^T\beta_n - x^T\hat{\beta}\}^2\right] + \mathsf{E}(\epsilon_{n+1}^2) \\
&= \mathsf{E}\left[\{\mu(x) - x^T\beta_n\}^2\right] + \underbrace{\mathsf{E}\left[\{\mu(x) - x^T\beta_n\}\{x^T\beta_n - x^T\hat{\beta}\}\right]}_{=0 \text{ because } \mathsf{E}[\hat{\beta}-\beta_n]=0.} \\
&\quad + \mathsf{E}\left[\{x^T\beta_n - x^T\hat{\beta}\}^2\right] + \mathsf{E}(\epsilon_{n+1}^2) \\
&= \mathsf{E}\left[\{\mu(x) - x^T\beta_n\}^2\right] + \mathsf{E}\left[\{x^T\beta_n - x^T\hat{\beta}\}^2\right] + \mathsf{E}(\epsilon_{n+1}^2).
\end{aligned}$$

To summarize, we have obtained the following *bias-variance* decomposition of MSPE:

$$\mathrm{MSPE}(x) = \underbrace{\mathsf{E}\left[\{\mu(x) - x^T\beta_n\}^2\right]}_{\text{bias}^2} + \underbrace{\mathsf{Var}\left(x^T\hat{\beta}\right)}_{\text{variance}} + \underbrace{\sigma^2}_{\text{irreducible}}. \tag{2.7}$$

Equation (2.7) plays a central role in understanding the predictive behaviour of regression models, as its derivation does not rely on how $\hat{\beta}$ is obtained. For the OLS estimator $\hat{\beta}_{\mathrm{OLS}}$, it can be shown that

$$\sum_{i=1}^n \mathsf{Var}(X_i^T\hat{\beta}_{\mathrm{OLS}} \mid X) = p\sigma^2. \tag{2.8}$$

Therefore, the average MSPE over the observed regressors is given by

$$\frac{1}{n}\sum_{i=1}^n \mathrm{MSPE}(X_i) = \frac{1}{n}\sum_{i=1}^n \{\mu(X_i) - X_i^T\beta_n\}^2 + \frac{p\sigma^2}{n} + \sigma^2. \tag{2.9}$$

Exercise 2.8. Prove (2.8).

Equation (2.9) illustrates a fundamental phenomenon called the *bias-variance trade-off*. In order to make the bias term $\sum_{i=1}^n \{\mu(X_i) - X_i^T\beta_n\}^2$ smaller, we can increase model complexity and include more regressors in the linear model. However, this comes at a price: the variance term $p\sigma^2/n$ will become larger. This trade-off of bias and variance applies to not only the least squares estimator but also many other statistical tasks;[3] see Figure 2.4 for a nice schematic illustration.

In linear models, the complexity of the least squares estimator is measured by p, which coincides with the degrees of freedom. In more complex models, it is not always straightforward to come up with a good measure of model complexity.

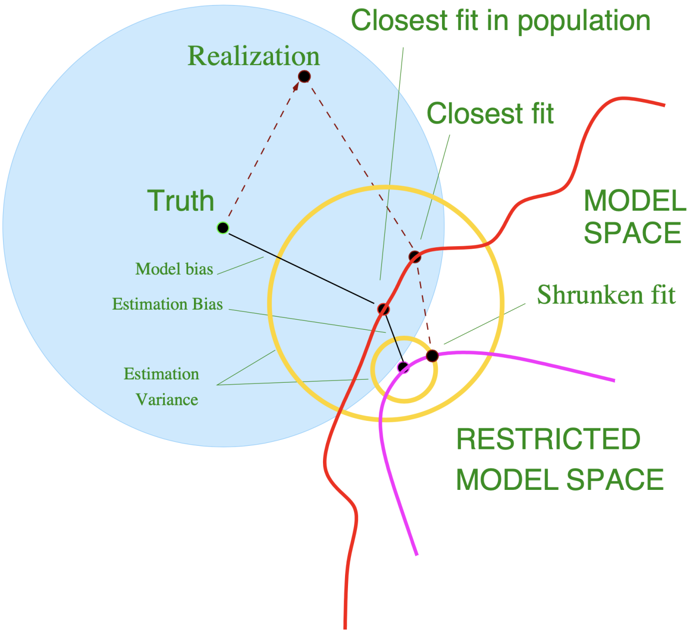


Figure 2.4: Schematic of the behavior of bias and variance.[4]

2.3.2 Quantitative criteria for model selection

Next, we review some commonly used criteria for model selection. A better idea is to estimate the prediction error of the working model.

The first criterion is *Mallows' C_p*, which is an unbiased estimator of the average MSPE in (2.9) (up to a constant scaling). To derive C_p, we first compute the expected value of the RSS under the nonparametric regression model (2.6):

$$\begin{aligned} \mathsf{E}\left(\|Y-\hat{\mu}\|^2 \mid X\right) &= \mathsf{E}\left\{\|(I-P)Y\|^2 \mid X\right\} \\ &= \mathsf{E}\left\{\|(I-P)(\mu+\epsilon)\|^2 \mid X\right\} \\ &= \|(I-P)\mu\|^2 + \mathsf{E}\left\{\|(I-P)\epsilon\|^2 \mid X\right\} \\ &= \|(I-P)\mu\|^2 + (n-p)\sigma^2. \end{aligned}$$

Notice that for the OLS estimator $\hat{\beta}_{\text{OLS}}$,

$$X\beta_n = X\,\mathsf{E}(\hat{\beta}_{\text{OLS}} \mid X) = X(X^TX)^{-1}X\,\mathsf{E}(Y \mid X) = P\mu.$$

Therefore, by comparing with (2.9), we see that

$$C_p = \|Y-\hat{\mu}\|^2 + 2p\sigma^2 \tag{2.10}$$

is an unbiased estimator of $\sum_{i=1}^n \text{MSPE}(X_i)$.

In practice, in order to use Mallows' C_p the noise variance σ^2 needs to be estimated. One common choice is to use the $\hat{\sigma}^2$ obtained from the full working model that uses all the regressors.

Heuristically, because the *training error rate* $\|Y-\hat{\mu}\|^2/n$ evaluates the predictive performance of the model using the same data that the model is fitted to, it underestimates the *prediction error rate* $\sum_{i=1}^n \text{MSPE}(X_i)/n$. The gap between the training error and prediction error is sometimes referred to as the *optimism* of the training error rate. In the case of OLS, the amount of optimism is $2p\sigma^2/n$, which is proportional to the degrees of freedom p, a measure of model complexity. In general, the optimism tends to become larger when the working model becomes more complex.

Exercise 2.9. Consider any linear estimator $\hat{\mu} = MY$ of μ where $M \in \mathbb{R}^{n\times n}$ only depends on the data through X. Show that $\text{tr}(M)$ is a "generalized degrees of freedom" in the sense that

$$C_M = \|Y-\hat{\mu}\|^2 + 2\sigma^2\text{tr}(M)$$

is an unibased estimator of $\sum_{i=1}^n \text{MSPE}(X_i)$ for $\hat{\mu}$.

Our second criteria is *leave-one-out cross-validation (LOO-CV)*, which is defined as

$$\text{LOO-CV} = \sum_{i=1}^n (Y_i - \hat{\mu}_{(-i)})^2,\ \hat{\mu}_{(-i)} = X_i^T\hat{\beta}_{(-i)},$$

where $\hat{\beta}_{(-i)}$ is the leave-one-out OLS estimator that is computed using all observations besides (X_i, Y_i). The idea of leave-one-out was used in the definition of Cook's distance (2.2) previously. Likewise, it is not necessary to actually compute the LOO OLS estimators repeatedly. Indeed, it can be shown that

$$\hat{\mu}_i = P_{ii}Y_i + (1-P_{ii})\hat{\mu}_{(-i)}.$$

Therefore, we have the following simple formula

$$\text{LOO-CV} = \sum_{i=1}^n \left(Y_i - \frac{\hat{\mu}_i - P_{ii}Y_i}{1-P_{ii}}\right)^2 = \sum_{i=1}^n \frac{(Y_i-\hat{\mu}_i)^2}{(1-P_{ii})^2}.$$

For linear models, Mallows' C_p and LOO-CV often lead to very similar estimates of the prediction error. The advantage of cross-validation is that it can still be used in more complex problems when there is no closed-form formula.

Another two commonly used criteria for model selection are *Akaike's information criterion* (AIC) and the *Bayesian information criterion* (BIC). Because these information criteria are based on the likelihood function, they can be applied to a wide range of statistical problems. To illustrate this flexibility, we describe these criteria in more general setups.

Suppose $Y_i \overset{\text{i.i.d.}}{\sim} f(y), i = 1,\ldots,n$, but a parametric model $Y_i \overset{\text{i.i.d.}}{\sim} f(y;\theta)$ is fitted instead over an Euclidean model space Θ. Under suitable regularity conditions, the MLE is expected to converge to

$$
\begin{aligned}
\hat{\theta} &= \arg\max_{\theta \in \Theta} \frac{1}{n} \sum_{i=1}^{n} \log f(Y_i; \theta) \\
&= \arg\max_{\theta \in \Theta} \frac{1}{n} \sum_{i=1}^{n} \log \frac{f(Y_i; \theta)}{f(Y_i)} \\
&\xrightarrow{p} \arg\max_{\theta \in \Theta} \mathsf{E}_f \left\{ \log \frac{f(Y_i; \theta)}{f(Y_i)} \right\} \\
&= \arg\min_{\theta \in \Theta} \mathsf{E}_f \left\{ -\log f(Y_i; \theta) \right\},
\end{aligned}
$$

where the subscript f under E means that the expectation is computed over the density $f(\cdot)$. To select an approriate model space Θ, AIC attempts to estimate

$$
\mathsf{E}_f \left\{ -2 \log f(Y_{n+1}; \hat{\theta}) \mid \hat{\theta} \right\},
$$

where the expectation is taken over a new observation $Y_{n+1} \sim f$. It does so by making a correction to the log-likelihood of the observed data at the MLE $\hat{\theta}$:

$$
\mathrm{AIC} = -2 \sum_{i=1}^{n} \log f(Y_i; \hat{\theta}) + 2 \dim(\Theta). \tag{2.11}
$$

The correction term $2\dim(\Theta)$ penalizes the log-likelihood function evaluated at the same data used to fit the model. This closely resembles the idea of estimating the optimism of the training error in Mallows' C_p. It can be shown that AIC (divided by n) is a consistent estimator of its target as $n \to \infty$, but that proof is beyond the scope of this course.

To simplify the presentation, we assumed above that the data are i.i.d.. The same idea can be easily extended to regression problems by replacing $f(Y_i; \hat{\theta})$ with the conditional likelihood given X_i.

Exercise 2.10. Show that for the normal linear model with known σ^2, AIC concides with Mallows' C_p.

Let $\{\Theta_1, \dots, \Theta_m\}$ be a collection of Euclidean model spaces. Then BIC is defined as

$$
\mathrm{BIC}(\Theta_k) = -2 \sum_{i=1}^{n} \log f(Y_i; \hat{\theta}_k) + \dim(\Theta_k) \, \log n,
$$

where $\hat{\theta}_k$ is the MLE over Θ_k. BIC, as its name indicates is motivated by the Bayesian perspective on model selection. If we assign a uniform prior on the model spaces,

$$
\pi(\Theta_k) = \frac{1}{m}, \ k = 1, \dots, m,
$$

Then it can be shown that, as $n \to \infty$, the posterior probability for a model is approximately given by

$$
\pi(\Theta_k \mid \mathrm{Data}) \propto e^{-\mathrm{BIC}(\Theta_k)/2}.
$$

Compared with AIC, BIC puts a larger penalty on model complexity and thus selects a sparser model. In practice, a rule of thumb is that AIC is more suitable for predictions and BIC is more suitable for selecting the "correct" model.[5]

2.3.3 Algorithms for model selection

Besides the statistical considerations discussed above, there are also computational challenges in model selection, as the number of submodels grows exponentially as the number of regressors increases. This section describes some algorithms that explore a large numberof of models more efficiently.

Our discussion thus far provides two useful insights for model selection. First, the RSS $\|Y - X\hat{\mu}\|^2$ concides with the (negative) log-likelihood function in the classical normal linear model and is indicative of predictive performance. Thus, it is reasonable to compare different models (especially those with the same complexity) by their RSS. Second, a good measure of model complexity is the degrees of freedom, i.e. the number of parameters in the model that are allowed to vary freely. These insights motivate the *best subset* algorithm, which selects the submodel with the smallest RSS for every degrees of freedom k.

However, the best subset algorithm is still computationally intensive for large p because it requires us to compute the RSS for all 2^p submodel. Two greedy algorithms are commonly used to reduce the number of search paths. The first is the *forward stepwise* algorithm, which starts from the null model and greedily adds one unselected regressor at a time that reduce RSS the most. The second is the *backward stepwise* algorithm, which starts from the full model and greedily removes one unselected regressor at a time that increases the RSS the least. There is of course no guarantee that these greedy algorithms will select the aboslute best submodel for each degrees of freedom k. But they often select a reasonably good submodel by examining only $O(p^2)$ submodels.

Example 2.11. Consider Figure 2.5, which shows the RSS for every submodel represented by a set of indicies of regressors for $p = 3$. For $k = 0, 1, 2$, and 3,

- The best subset algorithm selects $\emptyset$, $\{3\}$, $\{1, 2\}$, and $\{1, 2, 3\}$;
- The forward stepwise algorithm selects $\emptyset$, $\{3\}$, $\{2, 3\}$, and $\{1, 2, 3\}$;
- The backward stepwise algorithm selects $\emptyset$, $\{2\}$, $\{1, 2\}$, and $\{1, 2, 3\}$.

A common feature of these algorithms is that they produce a path of solutions that is indexed by model complexity. Once such a path is obtained, we can then select a single model by using one of the quantitative criteria introduced above. We can also resort to the model diagnostics and select a model that passes the visual checks or add new regressors to the search space (e.g. add a quadratic term when the residual vs. fitted plot shows a quadratic trend). There is no need to feel too uncomfortable about the ad hoc nature of model selection. As G. Box summarized in a famous aphorism, “All models are wrong, but some are useful.”

2.3.4 *Regularization

Thus far, our discussion on model selection has been fairly “discrete”. A single subset of regressors is selected, and model complexity is measured by an integer (the number of selected regressors). It is possible and in fact often desirable to “smoothen” this process

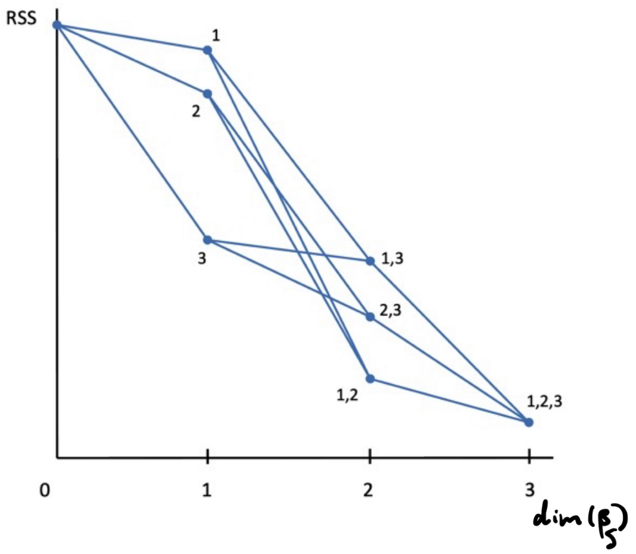


Figure 2.5: An illustration of model selection.

via an important idea called *regularization*. Briefly speaking, regularization tries to stablize the fitted model (or in statistical terms, reduce the variance of the estimator) by penalizing model complexity.[6]

Our first example is the best subset algorithm, which can be rewritten as the solution to the following optimization problem

$$\begin{aligned} \text{minimize} \quad & \|Y - X\beta\|^2 \\ \text{subject to} \quad & \|\beta\|_0 \le k, \end{aligned}$$

where $\|\beta\|_0 = |\{j \mid \beta_j \neq 0\}|$ is the number of non-zero entries in β (the ℓ_0-"norm"). Because the minimal value of this problem is decreasing in k, the solution path for $k = 0, \ldots, p$ can be reconstructed by the solution to the following unconstrained optimization problem

$$\text{minimize} \quad \|Y - X\beta\|^2 + \lambda\|\beta\|_0,$$

where $\lambda\|\beta\|_0$ is the regularzing penalty to the least squares objective $\|Y - X\beta\|^2$ and $\lambda \ge 0$ is a tuning parameter that controls the amount of regularization.

However, the l_0-"norm" $\|\beta\|_0$ is a difficult penalty to work with computationally. The most widely used alternatives are the *ridge regression* (l_2-norm penalty) that solves

$$\min_{\beta} \quad \|Y - X\beta\|^2 + \lambda\|\beta\|_2^2, \tag{2.12}$$

and the *lasso* (l_1-norm penalty) that solves

$$\min_{\beta} \quad \|Y - X\beta\|^2 + \lambda\|\beta\|_1. \tag{2.13}$$

One can also combine the l_1 and l_2 penalties and arrive at the *elastic net* regularization

$$\min_{\beta} \quad \|Y - X\beta\|^2 + \lambda\{(1-\alpha)\|\beta\|^2/2 + \alpha\|\beta\|_1\}, \tag{2.14}$$

which often works well for correlated regressors ($0 \leq \alpha \leq 1$ is another tuning parameter).

Exercise 2.12. Show that the ridge regression estimator is given by

$$\hat{\beta}_\lambda = (X^T X + \lambda I)^{-1} X^T Y$$

using the following two methods:

(i) Matrix calculus (see Section 1.4.2); and

(ii) Transforming (2.12) to an ordinary least squares problem of the form in (1.5).

Exercise 2.13. Suppose someone gave you a computationally efficient algorithm that can solve the lasso problem (2.13). Describe how you can modify it to also solve the elastic net problem (2.14).

2.3.5 **Inference after model selection

After a model is selected (e.g. by any of the algorithms described in Section 2.3.3), a common pitfall is to pretend that the selected regressors are determined a priori and apply the standard inference procedures (e.g. those described in Section 1.4.3). This is problematic because the selected subset of regressors is not fixed; in fact, it depends on the realized Y and incurs selection bias.

There are two common solutions to inference after model-selection:

(i) One can split the sample and use some observations for model selection and the others for statistical inference;

(ii) One can try to account for model selection, by excluding the information used by model selection from statistical inference.

The second solution is indeed an active research area in statistics.

2.4 Practical 3: Linear model diagnostics and selection

2.4.1 Model diagnostics

Applying the plot function to the output of `lm` gives four useful diagnostic plots (see Section 2.1 and `plot.lm` for further details). Here we demonstrate these plots using the `palmerpenguins` dataset, which contain measurements for 344 penguins. First, execute the code below in R and comment on the output.

```
# install.packages("palmerpenguins") # uncomment if not installed yet
library(palmerpenguins)
summary(penguins)
table(penguins$species, penguins$sex)
boxplot(body_mass_g ~ species * sex, penguins)
plot(body_mass_g ~ flipper_length_mm, penguins)
summary(fit1 <- lm(body_mass_g ~ flipper_length_mm, penguins))
abline(fit1, col = "red")
```

The `par` function is used to set plotting parameters. Here we set up the display so that four plots shown by `plot.lm` are produced on the same screen, saving the old parameters in `old_par`. We then reinstate the old parameters after the plot has been produced.

```
> old_par <- par(mfrow = c(2, 2))
> plot(fit1)
> par(old_par)
```

The output can be found in Figure 2.6. There is nothing extraordinary in these plots. The only noteworthy observations are a weak quadratic trend in the “Residual vs. Fitted” plot and a weak decreasing trend in the “Scale-Location” plot. The first observation might motivate us to include a quadratic term in the linear model. This can be done by modifying the model formula as in the code below. Do you see an improvement using the updated model?

```
fit2 <- lm(body_mass_g ~ flipper_length_mm + I(flipper_length_mm^2), penguins)
anova(fit, fit2)
plot(fit2)
```

2.4.2 Model selection and *regularization

Let us remove the rows with `NA` entries and split the dataset into two.

```
set.seed(42)
data <- na.omit(penguins)
n <- nrow(data)
training_index <- sample(n, 200)
training_data <- data[training_index, ]
test_data <- data[-training_index, ]
```

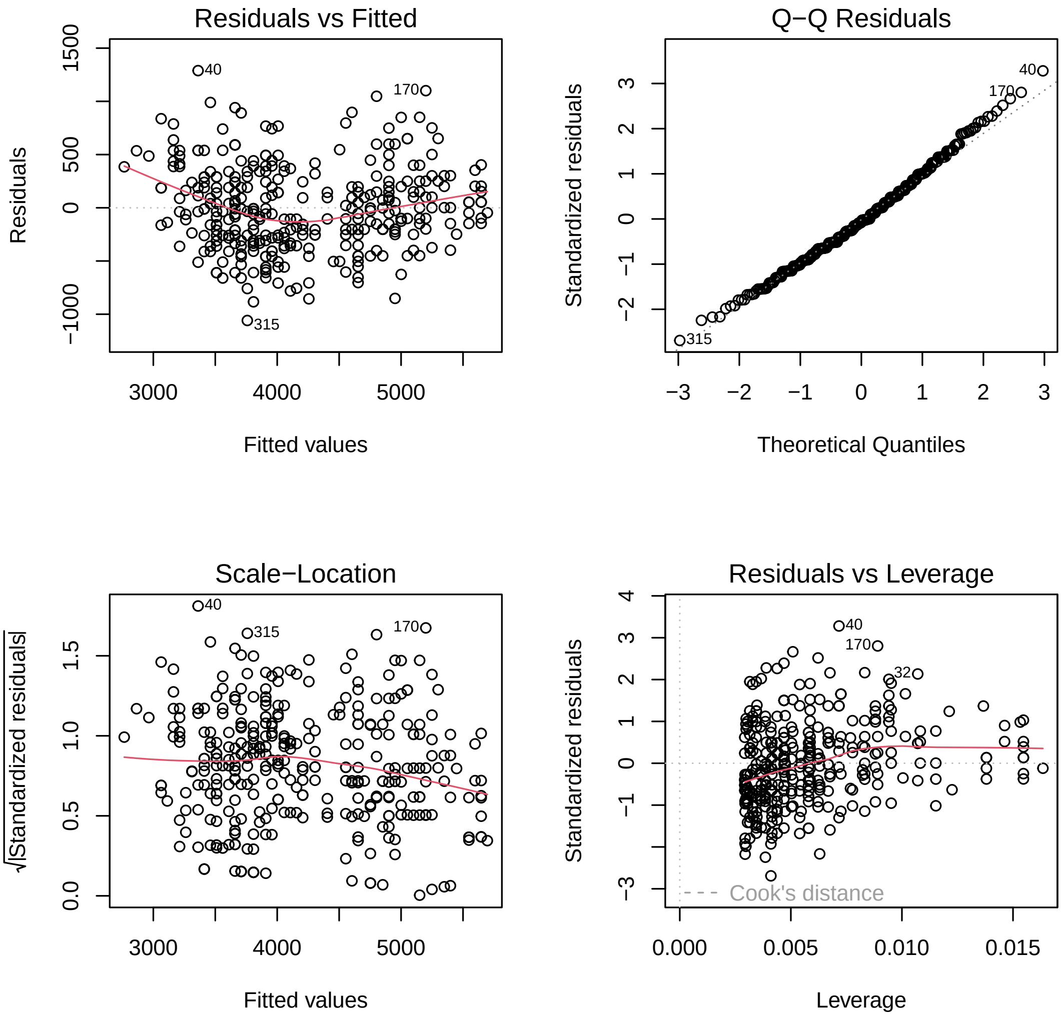


Figure 2.6: Diagnostic plots for a linear model.

We will first fit a “full” linear model with all main effects and interaction terms.

```
formula <- body_mass_g ~ (species + island + bill_length_mm +
                          bill_depth_mm + flipper_length_mm)^2
fit_full <- lm(formula, training_data, x = TRUE)
```

The `stepAIC` function in the `MASS` package does forward/backward model selection using AIC as the criterion. Run the code yourself and make sense of what it does from the output.

```
library(MASS)
```

```
fit_backward <- stepAIC(fit_full, direction = "backward")
coef(fit_backward)
```

We will use the `glmnet` package to demonstrate model regularization.

```
install.packages("glmnet")
library(glmnet)

fit_lasso <- cv.glmnet(fit_full$x[, -1],
                        training_data$body_mass_g,
                        alpha = 1)
plot(fit_lasso)
coef(fit_lasso, s = "lambda.min")

fit_ridge <- cv.glmnet(fit_full$x[, -1],
                        training_data$body_mass_g,
                        alpha = 0)
coef(fit_ridge, s = "lambda.min")

fit_elasticnet <- cv.glmnet(fit_full$x[, -1],
                             training_data$body_mass_g,
                             alpha = 0.2)
coef(fit_elasticnet, s = "lambda.min")
```

Which method returns the sparsest coefficients?

To compare the performance of these models, we can use the `test_data` that was held out in training the models. For each model, we make the predictions and calculate the root mean square error using `test_data`.

```
pred_full <- predict(fit_full, test_data)
RMSE_full <- sqrt(mean((pred_full - test_data$body_mass_g)^2))

pred_backward <- predict(fit_backward, test_data)
RMSE_backward <- sqrt(mean((pred_backward - test_data$body_mass_g)^2))

x_test <- model.matrix(formula, test_data)[, -1]

pred_ridge <- predict(fit_ridge, x_test, s = "lambda.min")
RMSE_ridge <- sqrt(mean((pred_ridge - test_data$body_mass_g)^2))

pred_lasso <- predict(fit_lasso, x_test, s = "lambda.min")
RMSE_lasso <- sqrt(mean((pred_lasso - test_data$body_mass_g)^2))

pred_elasticnet <- predict(fit_elasticnet, x_test, s = "lambda.min")
RMSE_elasticnet <- sqrt(mean((pred_elasticnet - test_data$body_mass_g)^2))
```

```
> c(RMSE_full, RMSE_backward, RMSE_ridge, RMSE_lasso, RMSE_elasticnet)
[1] 340.3524 337.8536 337.0227 337.3427 336.7763
```

Which method gives the best predictions?

2.4.3 Exercises

Exercise 2.14. There are some NA (Not Available) entries in the penguins dataset. Find out how lm handles the missing values.

Exercise 2.15. Color the body mass vs. flipper length scatterplot using species by

```
plot(body_mass_g ~ flipper_length_mm, penguins, col = species)
```

What do you observe? Given these observations, how would you update your linear model? Discuss your choice with your partner.

Exercise 2.16. Consider Anscombe's datasets, available in the R package datasets:

```
library(datasets)
anscombe
```

The next chunk of code shows four simple linear regression. The output can be found in Figure 2.7. What do you notice from the plots?

```
par(mfrow = c(2, 2))
for (i in 1:4) {
    x <- anscombe[[paste0("x", i)]]
    y <- anscombe[[paste0("y", i)]]
    fit <- lm(y~x)
    plot(x, y, type = "p", pch = 16, xlim = c(3,20), ylim = c(3,13),
         xlab = paste0("x", i), ylab = paste0("y", i))
    title(paste("y ~ ",
                signif(coef(fit)[1], 5), " + ",
                signif(coef(fit)[2], 5), "*x"))
    abline(fit, col = "red")
}
```

These synthetic datasets demonstrate some of the ways in which linear models can fail, and why we might need the diagnostic plots above. Run these diagnostics on these datasets and discuss what, if anything, can be done to fix these issues with your partner.

Exercise 2.17. Let us now briefly look at another dataset on house prices.

```
file_path <- "https://raw.githubusercontent.com/AJCoca/SM19/master/"
HousePrices <- read.csv(paste0(file_path, "HousePrices.csv"))
```

This data gives the sale prices (in US dollars) of houses in New York along with various factors that are thought to be relevant for predicting sale price. To fit a model of Sale.price against all other variables in HousePrices, we can do the following.

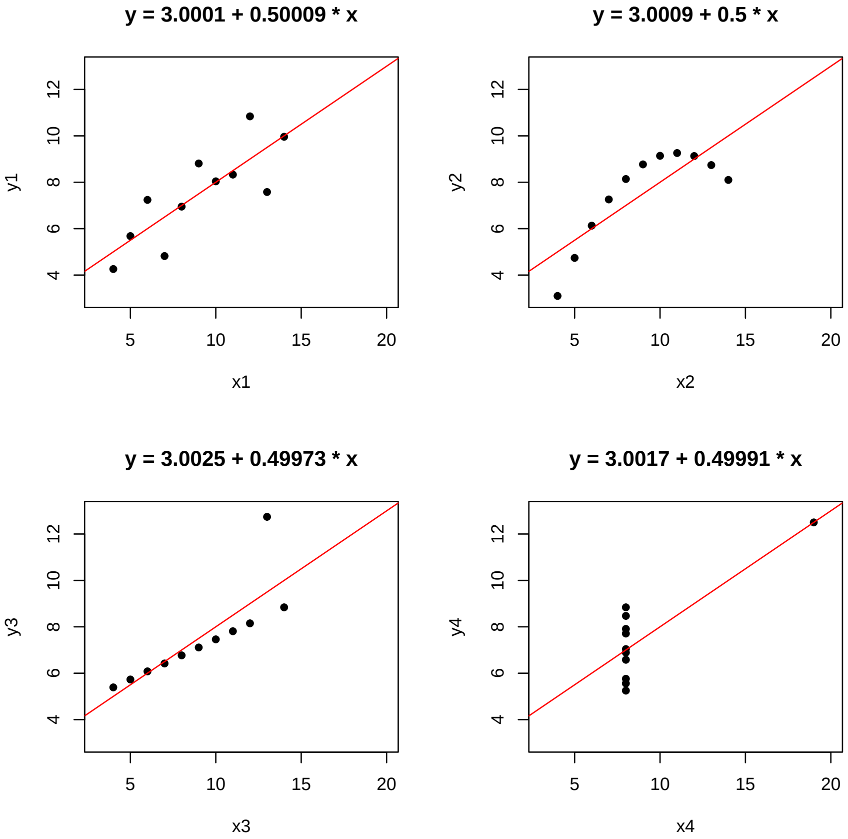


Figure 2.7: Anscombe's quartet.

```
HousePricesLM <- lm(Sale.price ~ ., data = HousePrices)
```

The `predict` function (try `?predict.lm`) can be used to give $x^{*T}\hat{\beta}$ for a new observation x^*. The observation x^* must be supplied as a data frame:

```
newdata <- data.frame("Bedrooms"=5, "Bathrooms"=2, "Living.area"=1400,
                      "Lot.size"=7000, "Year.built"=1950, "Property.tax"=9000)
```

(i) Apply the `confint` function to `HousePricesLM` to obtain confidence intervals for the coefficients.

(ii) Use the `interval` option of the `predict` function to get confidence intervals for $x^{*T}\beta$ and a prediction interval for $Y^* \sim N(x^{*T}\beta, \sigma^2)$.

(iii) Which model is selected using backward stepwise selection with the AIC?

2.5 Confounding and causality

2.5.1 Omitted-variables bias and the Yule-Simpson paradox

Misspecified models may also arise if some covariates are omitted in the regression. Recall the partial regression representation of the OLS estimator in Proposition 1.11. For the partition $X = (X_0\, X_1)$, let the correspoding OLS estimator be

$$\hat{\beta} = \begin{pmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \end{pmatrix} = (X^T X)^{-1} X^T Y.$$

By Proposition 1.11, we have

$$\hat{\beta}_0 = (\tilde{X}_0^T \tilde{X}_0)^{-1} \tilde{X}_0^T Y,$$

where $\tilde{X}_0 = (I - P_1)X_0$ where P_1 is the orthogonal projection matrix onto X_1. Now if we ignore the predictors X_1 and only use X_0 in the "short" linear regression, we would have obtained

$$\hat{\beta}_{0,\text{short}} = (X_0^T X_0)^{-1} X_0^T Y.$$

In general, $\hat{\beta}_0 \neq \hat{\beta}_{0,\text{short}}$ and this phenomenon is often known as *omitted variable bias* or *confounding*. To understand why confounding occurs, let's look at the following decomposition of Y in our proof of Proposition 1.11:

$$Y = \underbrace{(X_0\hat{\beta}_0 + P_0 X_1 \hat{\beta}_1)}_{P_0 Y} + \underbrace{(I - P_0) X_1 \hat{\beta}_1}_{(P-P_0)Y} + \underbrace{R}_{(I-P)Y}.$$

We see that

$$X_0 \hat{\beta}_{0,\text{short}} = P_0 Y = X_0 \hat{\beta}_0 + P_0 X_1 \hat{\beta}_1 \neq X_0 \hat{\beta}_0,$$

and the last inequality is true if $P_0 X_1 \neq 0$ and $\hat{\beta}_1 \neq 0$ (what do these mean geometrically?).

In other words, confounding means that marginal and conditional associations are generally different. In extreme cases, $\hat{\beta}_0$ and $\hat{\beta}_{0,\text{short}}$ can have different signs. This is known as the *Yule-Simpson paradox*. One of the best-known examples is the 1973 Berkeley admission data; see Table 2.1. Examining the university-wide statistics, men appear to be more likely to be admitted than women. However, if we use the department-level statistics, for most departments women had a higher admission rate. This apparent paradox can be explained by the observation that there appear to be more men applying to departments with a higher admission rate.

Fundamentally, the reason behind Simpson's paradox is that a regression coefficient only measures (conditional) correlation and does not necessarily indicate causation; see Figure 2.8 for an amusing illustration.[8]

2.5.2 Instrumental variables and two-stage least squares

Instrumental variables method is an approach to overcome unmeasured confounding (the relevant predictors are not just omitted but cannot be observed). To introduce the

Department	Men		Women	
	Applicants	Admitted	Applicants	Admitted
A	825	62%	108	82%
B	560	63%	25	68%
C	325	37%	593	34%
D	417	33%	375	35%
E	191	28%	393	24%
F	373	6%	341	7%
⋮	⋮	⋮	⋮	⋮
Total	8442	44%	4321	35%

Table 2.1: Berkeley admission data.[7]

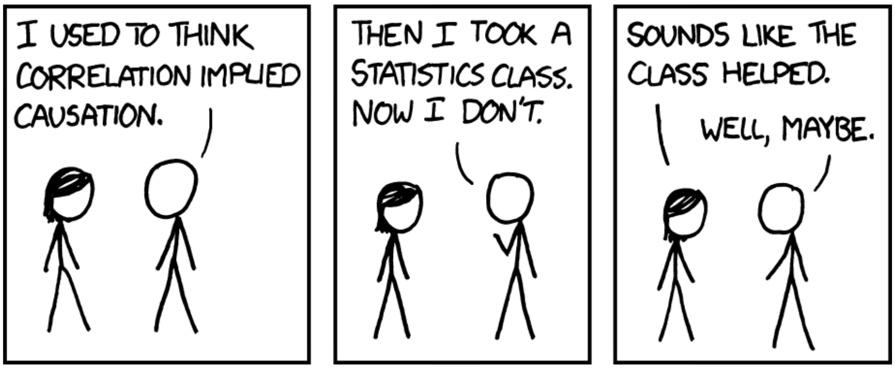

I USED TO THINK CORRELATION IMPLIED CAUSATION.
THEN I TOOK A STATISTICS CLASS. NOW I DON'T.
SOUNDS LIKE THE CLASS HELPED.
WELL, MAYBE.

Figure 2.8: Correlation does not imply causation.

basic idea, we assume the observations $(X_1, Y_1), \ldots, (X_n, Y_n) \in \mathbb{R}^p \times \mathbb{R}$ are i.i.d.. We can concisely write a confounded linear model as

$$Y_i = X_i^T\beta + \epsilon_i, \ \mathsf{E}(\epsilon_i X_i) \neq 0, \ i = 1, \ldots, n, \tag{2.15}$$

As discussed above, the OLS estimator will typically not estimate β.

To address this, we can use some *instrument variables* $Z_i \in \mathbb{R}^q$ such that $(X_i, Y_i, Z_i), i = 1, \ldots, n$ are i.i.d. and $\mathsf{E}(Z_i\epsilon_i) = 0$. Thus β satisfies

$$\mathsf{E}\{Z_i(Y_i - X_i^T\beta)\} = 0.$$

This gives us q linear equations with p unknowns. If $q = p$ (this is called the *just identified* case), we have

$$\beta = \{\mathsf{E}(Z_i X_i^T)\}^{-1}\,\mathsf{E}(Z_i Y_i),$$

which can be estimated using the data by

$$\hat{\beta} = \left\{\frac{1}{n}\sum_{i=1}^n Z_i X_i^T\right\}^{-1} \frac{1}{n}\sum_{i=1}^n Z_i Y_i = \left\{\sum_{i=1}^n Z_i X_i^T\right\}^{-1} \sum_{i=1}^n Z_i Y_i,$$

provided that $\mathsf{E}(Z_i X_i^T)$ and $\sum_{i=1}^n Z_i X_i^T$ are non-singular. If $q > p$ (this is called the *over-identified* case), we can use a function $g : \mathbb{R}^q \to \mathbb{R}^p$ and solve

$$\mathsf{E}\{g(Z_i)(Y_i - X_i^T \beta)\} = 0.$$

The corresponding estimator is given by

$$\hat{\beta} = \left\{ \sum_{i=1}^n g(Z_i) X_i^T \right\}^{-1} \sum_{i=1}^n g(Z_i) Y_i. \tag{2.16}$$

Thus, the just identified case corresponds to using the identity function as g. Under mild regularity conditions, it can be shown that $\hat{\beta}$ is consistent ($\hat{\beta} \xrightarrow{p} \beta$) and asymptotically normal ($\sqrt{n}(\hat{\beta} - \beta)$ converges to a normal distribution with mean 0) as long as $\mathsf{E}(g(Z_i) X_i^T)$ is non-singular.

In practice one has to choose the function g. It can be shown that the limiting variance of $\sqrt{n}(\hat{\beta} - \beta)$ is minimized by choosing $g(Z_i) = \mathsf{E}(X_i \mid Z_i)$. This is unknown but can be approximated using the data. A common choice is to use a linear working model for $\mathsf{E}(X_i \mid Z_i)$ and use

$$g(Z_i) = \hat{\gamma}^T Z_i, \quad \text{where } \hat{\gamma} = \operatorname*{arg\,min}_{\gamma \in \mathbb{R}^{q \times p}} \sum_{i=1}^n \|X_i - \hat{\gamma}^T Z_i\|^2 = \operatorname*{arg\,min}_{\gamma_1, \dots, \gamma_p \in \mathbb{R}^q} \sum_{i=1}^n \sum_{j=1}^p (X_{ij} - \hat{\gamma}_j^T Z_i)^2.$$

In matrix form ($X \in \mathbb{R}^{n \times p}$, $Z \in \mathbb{R}^{n \times q}$), we can write this as

$$\hat{\gamma} = (Z^T Z)^{-1} Z^T X,$$

assuming Z has rank q. Denote $P_Z = Z(Z^T Z)^{-1} Z$ as the projection matrix onto the column space of Z (so $P_Z = P_Z^2 = P_Z^T$) and $\hat{X} = P_Z X$ as the fitted values in the X on Z regression (so $\hat{X}_i = \hat{\gamma}^T Z_i$). We can write (2.16) with the above choice of g as

$$\begin{aligned} \hat{\beta}_{\text{TSLS}} = & \left(\sum_{i=1}^n \hat{X}_i X_i^T \right)^{-1} \sum_{i=1}^n \hat{X}_i Y_i \\ = & (\hat{X}^T \hat{X})^{-1} \hat{X}^T Y \\ = & (X^T P_Z X)^{-1} X^T P_Z Y \\ = & (\hat{X}^T \hat{X})^{-1} \hat{X}^T \hat{Y}, \end{aligned} \tag{2.17}$$

where $\hat{Y} = P_Z Y$. The expression in (2.17) is why this $\hat{\beta}_{\text{TSLS}}$ is called the *two-stage least squares* estimator: we first regress X on Z and then regress Y on the fitted values of X from the first regression. This has a clear geometric interpretation: we project X and Y onto the column space of Z before calculating the OLS estimator. From a statistial standpoint, we obtain an (asymptotically) unbiased estimator of β by using the randomness of X generated by the intrumental variable Z, which is not confounded with Y. This can be contrasted with the partial regression characterization in Proposition 1.11: in an linear regression of Y on X and Z, the OLS coefficient of X is given by

$$(\tilde{X}^T \tilde{X})^{-1} \tilde{X}^T \tilde{Y} = (X^T (I - P_Z) X)^{-1} X^T (I - P_Z) Y,$$

where $\tilde{X} = (I - P_Z)X$ and $\tilde{Y} = (I - P_Y)Y$. So in ordinary regression analysis that "adjusts for" Z (instead of using Z as instrumental variables), we project X and Y onto the *orthogonal complement* of the column space of Z before calculating the OLS estimator.

Exercise 2.18. Prove the equivalent expressions of the two-stage least squares estimator $\hat{\beta}_{\text{TSLS}}$ above.

2.5.3 **Linear structural equation models

The confounded linear model in (2.15) can be confusing if you are not used to think about causality. From a mathematical standpoint, (2.15) is satisfied with any choice of β. What (2.15) is really meant to describe is a structural equation, that is, if we can manipulate the system and set X_i to some value, then the response variable Y_i will be generated according to (2.15). To formally describe this, we need to think beyond the observations. If we let $Y_i(x_i)$ denote the *potential outcome* of Y_i under an intervention that sets X_i to x_i, then by calling (2.15) a *structural equation* we mean

$$Y_i(x_i) = x_i^T \beta + \epsilon_i \text{ for all } x_i,\ i = 1, \dots, n.$$

Again, the key point is that this equation needs to hold for *all* interventions.

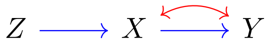


Figure 2.9: Instrumental variable graph.

We may use several structural equations to describe the causal relationship between a number of variables. In this case, it is useful to visualize the causal dependence using a directed (mixed) graph. The standard instrumental variable graph can be found in Figure 2.9. For simplicity, let us assume X and Z are univariate and mean 0 (so we do not need to include an intercept term). The corresponding linear structural equation model is given by

$$\begin{aligned} X(z) &= \gamma z + \epsilon_X, \\ Y(z, x) &= \beta x + \epsilon_Y, \end{aligned} \tag{2.18}$$

and ϵ_Z is independent of (ϵ_X, ϵ_Y). Directed edges in Figure 2.9 indicate direct causal dependence, and the lack of the $Z \to Y$ edge means Z can only influence Y indirectly through X (thus $Y(z, x)$ does not depend on z). The bidirected edge $X \leftrightarrow Y$ in Figure 2.9 means that X and Y are confounded by exogenous sources, possibly due to a unobserved common cause. Mathematically speaking, this means that ϵ_X and ϵ_Y may be correlated.

The equations in (2.18) describe the causal dependence between Z, X, and Y. To obtain the equations, we first plug in a realization of Z to the first equation to obtain the realized X, and then plug in the realized (Z, X) to the second equation to obtain Y. This is called *recursive substitution* and results in the equations

$$\begin{aligned} X &= \gamma Z + \epsilon_X, \\ Y &= \beta X + \epsilon_Y, \end{aligned} \tag{2.19}$$

which look like a regression model. It is clear that ϵ_Y and X may be correlated because ϵ_X and ϵ_Y are allowed to be correlated (due to the $X \leftrightarrow Y$ edge in Figure 2.9). This motivates the confounded linear model in (2.15), in which β can be interpreted as the causal effect of X on Y.

2.6 Practical 4: Interpreting linear models

2.6.1 Yule-Simpson paradox

We first give a demonstration of the Yule-Simpson paradox using `palmerpenguins` dataset. The correlation between body mass and bill depth is negative. This surprising phenomenon disappears when we condition on species.

```
library(palmerpenguins)
plot(body_mass_g ~ bill_depth_mm, penguins, col = species)
abline(lm(body_mass_g ~ bill_depth_mm, penguins), col = "blue")
fit <- lm(body_mass_g ~ 0 + species + bill_depth_mm, penguins)
abline(fit$coef[1], fit$coef[4], col = "black")
abline(fit$coef[2], fit$coef[4], col = "red")
abline(fit$coef[3], fit$coef[4], col = "green")
```

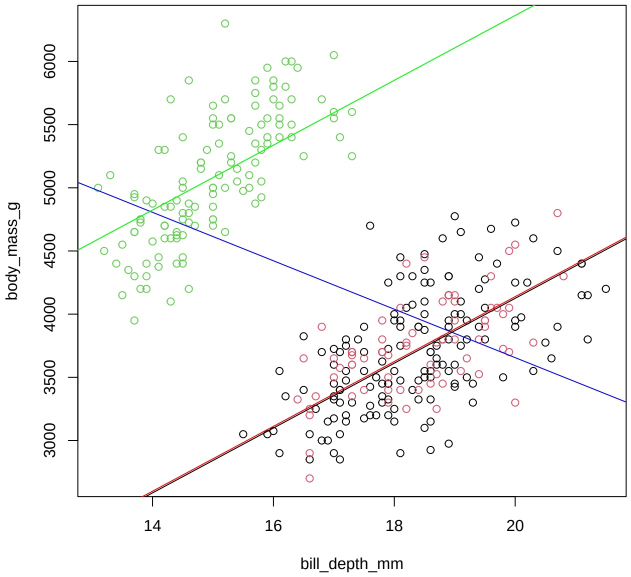


Figure 2.10: Yule-Simpson paradox.

2.6.2 Yule on the causes of poverty

Legendre (1805) and Gauss (1809) developed regression techniques to fit data on orbits of astronomical objects. The relevant variables were known from Newtonian mechanics, and so were the functional forms of the equations connecting them. Measurement could be done with high precision. Much was known about the nature of the errors in the measurements and equations. Furthermore, there was ample opportunity for comparing predictions to reality. Later on, Quetelet (1835) wanted to uncover “social physics”—the laws of human behaviour—by using statistical technique. Investigators were using regression on social science data where the conditions of a normal linear model did not hold, even to a rough approximation—with consequences that need to be explored.

Much of this example is taken from Freedman (Section 1.4), who re-investigated the dataset and argument in the following paper.

- Yule, G. U. (1899). An investigation into the causes of changes in pauperism in England, chiefly during the last two intercensal decades. (part i.) *Journal of the Royal Statistical Society*, *62*(2), 249–286. doi:10.1111/j.2397-2335.1899.tb03709.x.

```
> path <- "https://www.statslab.cam.ac.uk/~qz280/teaching/modelling-2025/"
> data <- read.table(file.path(path, "yule.csv"))
> data
                    paup outrelief old pop
Kensington           -73       -95   4  36
Paddington           -53       -88  15  11
Fulham               -69       -79 -15  74
Chelsea              -36       -79 -19  24
St. George's         -54       -82  13  -4
Westminster          -48       -73   5  -9
Marylebone           -19       -64   0  -3
St. John, Hampstead  -39       -61   3  41
St. Pancras          -39       -65   1   7
Islington            -41       -65   1  32
Hackney              -67       -78  -9  50
St. Giles'           -24       -70   3 -15
Strand               -36       -73  -3 -19
Holborn              -21       -67  -5  -7
City                 -21       -36  13 -32
Shoreditch           -48       -79   8   0
Bethnal Green        -54       -81   2   6
Whitechapel          -65       -94  -7  -7
St. George's East    -63       -94  -2  -2
Stepney              -66       -90 -13   1
Mile End             -57       -85   2  13
Poplar               -63       -80   2  35
St. Saviour's        -48       -78   0  11
St. Olave's          -43       -68   2  10
Lambeth              -43       -62  -1  22
Wandsworth           -77       -82  -9  68
Camberwell           -70       -86 -17  68
Greenwich            -45       -63  -6  31
Lewisham             -59       -76   0  42
Woolwich             -24       -80  19  10
```

```
Croydon                  -62       -71   1  42
West Ham                 -62       -51 -14 103
```

The columns represent percentage changes in pauperism (the state of being supported at public expense), out-relief (supported outside "poorhouses"), population aged over 65, and the population from 1871 to 1881 in different areas. Yule used a linear model to explain the changes in pauperism and got the following equation from ordinary least squares:

$$\Delta\text{paup} = 13.19 + 0.755\Delta\text{outrelief} - 0.022\Delta\text{old} - 0.322\Delta\text{pop} + \text{error}.$$

The coefficient of Δoutrelief being relatively large and positive, Yule concludes that out-relief causes poverty.

The causal interpretation of the coefficient 0.755 is the following. Other things being equal, if Δoutrelief increased by 1%—the adminstrative district supports more people outside the poorhouses—then Δpaup will go up by 0.755%. This is a *quantitative inference*. Out-relief causes an increase in pauperism—a *qualitative inference*. The point of introducing Δold and Δpop into the quation is to control for possible confounders, implementing the idea of "other things being equal". For Yule's argument, it is important that the coefficient of Δout be significantly positive. Qualitative inferences are often the important ones; with regression, the two aspects are woven together.

The exact inference for normal linear models had not been developed in Yule's time.[9] Using modern software we can easily obtain, for example, confidence intervals of the coefficients.

```
> fit <- lm(paup ~ outrelief + old + pop, data)
> summary(fit)

Call:
lm(formula = paup ~ outrelief + old + pop, data = data)

Residuals:
    Min      1Q  Median      3Q     Max 
-17.475  -5.311  -1.829   3.132  25.335 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 12.88356   10.36722   1.243    0.224    
outrelief    0.75209    0.13499   5.572 5.83e-06 ***
old          0.05560    0.22336   0.249    0.805    
pop         -0.31074    0.06685  -4.648 7.25e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 9.547 on 28 degrees of freedom
Multiple R-squared:  0.6972,    Adjusted R-squared:  0.6647
```

```
F-statistic: 21.49 on 3 and 28 DF,  p-value: 2.001e-07

> confint(fit)
                 2.5 %     97.5 %
(Intercept) -8.3527245 34.1198351
outrelief    0.4755856  1.0286034
old         -0.4019236  0.5131276
pop         -0.4476771 -0.1737995
```

Yule's method has a number of potential pitfalls:

(i) Districts with more efficient administrations were building poorhouses and reducing poverty. So efficiency of adminstration is then a confounder, influencing both the presumed causes and its effect. Economics may be another confounder.

(ii) Yule's results are not consistent across time and geography. For example, the same equation for the 1881-1891 period is

$$\Delta\text{paup} = 1.36 + 0.324\Delta\text{outrelief} + 1.37\Delta\text{old} - 0.369\Delta\text{pop} + \text{error}.$$

(iii) Yule has established association: conditional on the covariates, there is a positive association between Δpaup and Δoutrelief. Is this association causal? If so, which way do the causal arrows point? For instance, a parish may choose not to build poorhouses in response to a short-term increase in the number of paupers, in which case pauperism causes out-relief. Likewise, the number of paupers in one area may well be affected by relief policy in neighbouring areas.

Yule was aware of the problems and indeed withdrew all causal claims in a foonote: "Strictly speaking, for 'due to' read 'associated with"'.

Statistical inference could have been made at different levels for this dataset:

(i) *Descriptive* inference tells us about the data that we happen to have. For example, we can say pauperism on average reduced by 49.7% across the 32 areas from 1871 to 1881.

(ii) *Predictive* inference approximate the value of Δpaup. For example, we can use Yule's equation and the values of Δoutrelief, Δold, Δpop from 1891 to 1901 to predict further changes in pauperism in these areas.

(iii) *Causal* inference claims to tell us what will happen to some of the numbers if you intervene to change other numbers. For example, if Yule could have used his findings to convince the administrators of Islington to build 10% more poorhouses, how much would that change pauperism there?

2.6.3 Economic return to schooling

Statistical methods for social science data have seen much development after Yule. We next investigate a dataset in the following paper that was used to estimate the economic effect of education. The dataset was compiled from the National Longitudinal Survey of Young Men (NLSYM) and contains a sample of 3010 young men at the age of 14 to 24 in 1966 who were followed up until 1981.

- Card, D. (1995). Using geographic variations in college proximity to estimate the return to schooling. In Christofides, L. N., Grant, E. K., and Swidinsky, R., editors, *Aspects of Labor Market Behaviour: Essays in Honour of John Vanderkamp.* University of Toronto Press.

```
> # install.packages("ivmodel")
> library(ivmodel)
> vars <- c("lwage", "educ", "exper", "expersq", "black", "south", "smsa", "nearc4")
> data <- card.data[, vars]
> head(data)
     lwage educ exper expersq black south smsa nearc4
1 6.306275    7    16     256     1     0    1      0
2 6.175867   12     9      81     0     0    1      0
3 6.580639   12    16     256     0     0    1      0
4 5.521461   11    10     100     0     0    1      1
5 6.591674   12    16     256     0     0    1      1
6 6.214608   12     8      64     0     0    1      1
```

We focus on the variables used in the original analysis by Card, including

- `lwage`: log wage in 1976.
- `educ`: years of education.
- `exper`: years of labor force experience in 1976.
- `expersq`: square of `exper`.
- `black`: race is black.
- `south`: lived in the South in 1976.
- `smsa`: lived in SMSA (Standard Metropolitan Statistical Area) in 1976.
- `nearc4`: grew up near a four-year college.

Other variables in the dataset can be found in `?card.data`.

Worried about potential unmeasured confounders between education and income (such as motivation of the young men), Card used proximity to a four-year college as an instrumental variable. His main analysis can be reproduced by

```
> fit <- ivmodelFormula(lwage ~ educ + exper + expersq + black + south + smsa |
                        nearc4 + exper + expersq + black + south + smsa, data)
> fit
First Stage Regression Result:

F=16.71759, df1=1, df2=3003, p-value is 4.4515e-05
R-squared=0.005536144,   Adjusted R-squared=0.005204987
Residual standard error: 1.942531 on 3004 degrees of freedom
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

Coefficients of k-Class Estimators:

              k Estimate Std. Error t value Pr(>|t|)
OLS    0.000000 0.074009   0.003505  21.113  < 2e-16 ***
Fuller 0.999667 0.128981   0.047601   2.710  0.00677 **
TSLS   1.000000 0.132289   0.049233   2.687  0.00725 **
LIML   1.000000 0.132289   0.049233   2.687  0.00725 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Some of the output is truncated. The first thing to see is the F-statistic of the first-stage regression, which indicates the "strength" of the instrumental variable (recall that we require $\mathsf{E}(g(Z_i)X_i)$ to be non-singular in Section 2.5.2). When the instrument is too weak, the asymptotic inference can be inaccurate. The `ivmodel` package uses several instrumental variables methods (`Fuller`, `TSLS`, `LIML`); they are asymptotically equivalent in the standard regime (fixed p, fixed instrument strength, $n \to \infty$). We see from the output that Card's instrumental variable analysis supports the hypothesis that more education led to increased income in a causal way. In fact, in this case the TSLS estimate is not too different from the OLS estimate.

We can also obtain the two-stage least squares estimate by solving two least-squares problem as shown below. The estimated causal effect 0.1322888 (i.e. one more year of education increases income by about 14%) is identical to that returned by the `ivmodel` function.

```
> data$educ.fitted <-
     fitted(lm(educ ~ nearc4 + exper + expersq + black + south + smsa))
> fit2 <- lm(lwage ~ educ.fitted + exper + expersq + black + south + smsa, data)
> summary(fit2)
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.7527813  0.8495104   4.418 1.03e-05 ***
educ.fitted  0.1322888  0.0504306   2.623  0.00876 **
exper        0.1074980  0.0218186   4.927 8.81e-07 ***
expersq     -0.0022841  0.0003423  -6.674 2.96e-11 ***
black       -0.1308019  0.0541582  -2.415  0.01579 *
```

```
south       -0.1049005  0.0236342  -4.438 9.39e-06 ***
smsa         0.1313237  0.0308626   4.255 2.15e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4005 on 3003 degrees of freedom
Multiple R-squared:  0.1871,    Adjusted R-squared:  0.1854
F-statistic: 115.2 on 6 and 3003 DF,  p-value: < 2.2e-16
```

2.6.4 Exercises

Exercise 2.19. The green line in Figure 2.10 appears to provide a poor fit to the cloud of green points. What happened and how can you fix it?

Exercise 2.20. Yule presented his paper at a meeting of the Royal Statistical Society on 21 March 1899. There was a lively discussion.[10]

(i) According to Professor FY Edgeworth, if one diverged much from the law of normal errors, “one was on an ocean without rudder or compass”; this normal law of error “was perhaps more universal than the law of gravity.” Do you agree? Discuss briefly.

(ii) According to Sir Robert Griffen, practical men who were concerned with poor-law administration knew that “if the strings were drawn tightly in the matter of out-door relief, they could immediately observe a reduction of pauperism itself.” Yule replied,

> “he was aware that the paper in general only bore out conclusions which had been reached before... but he did not think that lessened the interest of getting an independent test of the theories of practical men, purely from statistics. It was an absolutely unbiased test, and it was always an advantage in a method that it was unbiased.”

What do you think of this reply? Is Yule’s test “purely from statistics”? Is it Yule’s methods that are “unbiased,” or his estimates of the parameters given his model?

Discuss the following questions further with your partner:

(iii) What are the real-world assumptions in using Yule’s equation for prediction?

(iv) How can you validate Yule’s equation as predictive inference and as causal inference?

(v) How can you use mathematics to formally distinguish a causal model from a predictive model?

Exercise 2.21. Do you think proximity to four-year college is a good instrumental variable for estimating the economic return to schooling? Discuss with your partner.

Exercise 2.22. In our analysis of Card’s dataset, the standard error of the causal effect of education in `fit2`, obtained by using `lm` twice, is different from that in `fit` obtained by using `ivmodel`. Can you give an explanation to this observation? *[Hint: Try calculating the variance of $\hat{\beta}_{TSLS}$ by pretending $Z^T X$ is a constant.]*

Notes

[1] The LAD estimator is a *robust regression* method that tries to limit the influence of outliers.

[2] Some other texts define MSPE as $\mathsf{E}[\{\mu(x) - x^T\hat{\beta}\}]$, which we shall refer to as the *mean squared error.*

[3] One simple instance is *Stein's* paradox, which is discussed in the *Principles of Statistics* course in detail.

[4] Taken from Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning* (2nd ed.). New York: Springer, Figure 7.2.

[5] In a Bayesian setup, it is not necessary to select a single model. An alternative and perhaps more desirable approach is called *Bayesian model averaging.*

[6] The same idea is frequently used to solve ill-posed inverse problems in applied mathematics and engineering, often under the name *Tikhonov regularization.*

[7] Freedman, D., Pisani, R., & Purves, R. (2007). *Statistics.* New York: W W Norton, p.18.

[8] https://xkcd.com/552.

[9] In fact, Yule calculated the least squares estimate with two slide rules and the "Brunsviga Arithmometer"—a pin-wheel calculating machine that could add, substract, multiply, and divide.

[10] The discussion is published in *Journal of the Royal Statistical Society*, Volume 62, Issue 2, June 1899, Pages 287–295, https://doi.org/10.1111/j.2397-2335.1899.tb03710.x. The first two questions are rephrased from Freedman Section 7.5, Question 20.

Chapter 3

Exponential families

3.1 Definition and examples

This Chapter provides an introduction to the theory of expoential families, which expands the classical statistical theory based on normality. Exponential families are basic building blocks of the generalized linear models discussed in the next Chapters and more complex statistical models.

3.1.1 Exponential tilting

Exponential families are obtained by exponentially "tilting" any density function. Suppose $f_0(y),\ y \in \mathcal{Y} \subseteq \mathbb{R}^d$ is a density function with respect to a dominating measure $m(dy)$. By *exponential tilting*, we mean a collection of density functions given by

$$f(y;\theta) \propto e^{\theta^T T(y)} f_0(y).$$

By normalizing the density functions, we obtain

$$f(y;\theta) = e^{\theta^T T(y) - K(\theta)} f_0(y), \tag{3.1}$$

where

$$K(\theta) = \log \int_{\mathcal{Y}} e^{\theta^T T(y)} f_0(y) m(dy).$$

Some terminologies for the terms in (3.1):

- $\theta \in \Theta \subseteq \mathbb{R}^p$ is called the *natural parameter* or *canonical parameter*.
- $T(y) \in \mathbb{R}^p$ is called the *sufficient statistic*.
- $f_0(y)$ is called the *carrying density*.
- $K(\theta)$ is called the *cumulant function*.
- $\Theta = \left\{\theta \in \mathbb{R}^p \mid \int_{\mathcal{Y}} e^{\theta^T T(y)} f_0(y) m(dy) < \infty\right\}$ is called the *natural parameter space*.

Obviously, $f(y;0) = f_0(y)$, so $0 \in \Theta$. Furthermore, for any $\theta_0 \in \Theta$, we may rewrite the density function as

$$f(y;\theta) = e^{(\theta-\theta_0)^T T(y) - \{K(\theta)-K(\theta_0)\}} f(y;\theta_0),$$

So comparing to (3.1), we see that the same exponential family can be obtained by exponentially tilting any density function $f(y;\theta_0)$ within it.

Exercise 3.1. Show that Θ is a convex set and $K(\theta)$ is a convex function on Θ. [*Hint:* Use Hölder's inequality.]

3.1.2 Examples

The first motivation to study exponential families is that they contain many important probability distributions. Next we go through some examples.

Example 3.2 (Normal distribution)**.** The density function of $\mathrm{N}(\mu, 1)$ is given by

$$\begin{aligned} f(y;\mu) &= \frac{1}{\sqrt{2\pi}} e^{-(y-\mu)^2/2} \\ &= \exp\big(\underbrace{\mu}_{\theta} \underbrace{y}_{T(y)} - \underbrace{\mu^2/2}_{K(\theta)}\big) \underbrace{\frac{1}{\sqrt{2\pi}} e^{-y^2/2}}_{f(y;0)}. \end{aligned}$$

Example 3.3 (Poisson distribution)**.** The Poisson distribution with rate λ can be obtained by exponentially tilting the probability mass function of Poisson(1):

$$f_0(y) = e^{-1}\frac{1}{y!},\ y = 0, 1, \dots .$$

We can first compute the cumulant function

$$K(\theta) = \log \sum_{y=0}^{\infty} e^{\theta y} e^{-1} \frac{1}{y!} = -1 + \log \sum_{y=0}^{\infty} \left(e^{\theta}\right)^y \frac{1}{y!} = e^{\theta} - 1$$

The exponentially tilted density is then given by

$$\begin{aligned} f(y;\theta) &= e^{\theta y - K(\theta)} f_0(y) \\ &= e^{\theta y - e^{\theta}} \frac{1}{y!} \\ &= \lambda^y e^{-\lambda} \frac{1}{y!}, \quad \text{for } \lambda = e^{\theta}. \end{aligned}$$

Thus, the natural parameter θ is related to the mean parameter λ via $\theta = \log \lambda$.

Example 3.4 (Binomial distribution)**.** The probability mass function of a Binomial(n, π) with fixed n is given by

$$f(y;\pi) = \binom{n}{y} \pi^y (1-\pi)^{n-y} = e^{y \log \frac{\pi}{1-\pi} + n \log(1-\pi)} \binom{n}{y},\ y = 0, 1, \dots, n.$$

So the natural parameter is the so-called *logit function* or *log odds*

$$\theta(\pi) = \log \frac{\pi}{1-\pi}$$

that maps $(0,1)$ to $\mathbb{R}$. By inversing this, we can obtain the usual parameter for binomial by the *expit function*

$$\pi(\theta) = \frac{e^\theta}{1+e^\theta}.$$

The cumulant function is given by

$$K(\theta) = -n \log(1-\pi) = n \log(1+e^\theta).$$

More rigorously, we should further normalize the $\binom{n}{y}$ term as it does not add up to 1. But for all practical purposes, it is enough to obtain a cumulant function $K(\theta)$ up to a constant difference.

Exercise 3.5. Show the following distributions are exponential families and find their natural parameter, sufficient statistic, and cumulant function:

(i) The normal distribution

$$f(y;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-\mu)^2}{2\sigma^2}}, y \in \mathbb{R}.$$

(ii) The Gamma distribution

$$f(y;\alpha,\beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} y^{\alpha-1} e^{-\beta y},\ y > 0.$$

(iii) The negative binomial distribution with fixed k

$$f(y;\pi) = \binom{y+k-1}{y} \pi^k (1-\pi)^y,\ y = 0,1,2,\dots.$$

Exercise 3.6. Consider the multinomial distribution Multinomial(n,π) with probability mass function given by

$$f(y;\pi) = \frac{n!}{y_1! \cdots y_L!} \pi_1^{y_1} \cdots \pi_L^{y_L}.$$

Suppose n is known but $\pi = (\pi_1,\dots,\pi_L)$ is unknown.

(i) Write this as an exponential family.

(ii) We say an exponential family is *minimal* if the sufficient statistics are linearly independent. Is your answer in (i) minimal? If not, can you write it as an alternative exponential family that is minimal and give its cumulant function? *[Hint: Use the natural parameters* $\log(\pi_1/\pi_L),\dots,\log(\pi_{L-1}/\pi_L)$*.]*

3.2 Properties of exponential families

Next, we introduce some important properties about one-parameter exponential families. Many of the results below can be readily extended to multi-parameter exponential families.

3.2.1 Cumulants

The moments of an exponential family distribution (3.1) can be easily computed by its cumulant function. Recall that for a random variable Y, its *moment generating function* is given by

$$M(t) = \mathsf{E}\left(e^{tY}\right),$$

and its *cumulant generating function* is defined as

$$K(t) = \log M(t).$$

Suppose $M(t)$ is infinitely differentiable at 0 (which requires $M(t)$ to be well defined in a neighbourhood of 0). Then we have the following Maclaurin expansions

$$\begin{aligned} M(t) &= \sum_{r=0}^{\infty} \mathsf{E}(Y^r)\frac{t^r}{r!}, \\ K(t) &= \sum_{r=0}^{\infty} \kappa_r \frac{t^r}{r!}, \end{aligned}$$

where $\mathsf{E}(Y^r) = M^{(r)}(0)$ and $\kappa_r = K^{(r)}(0)$. The values $\kappa_1, \kappa_2, \dots$ are called *cumulants* of the probability distribution and are closely related to the moments. Cumulants are useful because they are the only "summaries" of the probability distribution that are additive (with respect to sum of i.i.d. variables). In particular, the first two cumulants are the mean and variance respectively. The third and fourth cumulants, after normalization, are the called the *skewness* and *kurtosis* of this distribution respectively.

Exercise 3.7. Verify that $\kappa_1 = \mathsf{E}(Y)$, $\kappa_2 = \mathsf{Var}(Y)$, $\kappa_3 = \mathsf{E}(Y-\kappa_1)^3$, and $\kappa_4 = \mathsf{E}(Y-\kappa_1)^4 - 3\kappa_2^2$.

The exponential family $\{f(y;\theta) \mid \theta \in \Theta\}$ is called *regular* if Θ is an open set. Nearly all exponential families and certainly all the exponential families we will consider are regular. For a regular exponential family with a one-parameter natural parameter θ, the moment generating function (for small enough t) is given by

$$\begin{aligned} M_\theta(t) &= \mathsf{E}_\theta(e^{tY}) \\ &= \int e^{ty} e^{\theta y - K(\theta)} f_0(y) m(dy) \\ &= e^{K(\theta+t)-K(\theta)} \int e^{(t+\theta)y - K(\theta+t)} f_0(y) m(dy) \\ &= e^{K(\theta+t)-K(\theta)}. \end{aligned}$$

So the cumulant generating function is given by

$$K_\theta(t) = \log M_\theta(t) = K(\theta + t) - K(\theta).$$

Therefore, the mean and variance are given by the first two derivates of the cumulant function $K(\cdot)$ at θ:

$$\mu(\theta) = \mathsf{E}_\theta(Y) = \frac{d}{dt} K_\theta(t)\bigg|_{t=0} = K'(\theta), \tag{3.2}$$

$$V(\theta) = \mathsf{Var}_\theta(Y) = \frac{d^2}{dt^2} K_\theta(t)\bigg|_{t=0} = K''(\theta). \tag{3.3}$$

This is why we often only need to determine $K(\theta)$ up to an additive constant. These formulas can be readily generalized to multi-parameter families (with multi-variate sufficient statistics), for which derivatives are simply replaced by gradients.

We refer to $\mu(\theta)$ as the *mean function* and $V(\theta)$ the *variance function*. The above derivation shows that they are related through the following key identity

$$\mu'(\theta) = K''(\theta) = V(\theta) \geq 0. \tag{3.4}$$

This shows that, apart from pathological cases with zero variance, the mean function $\mu(\theta)$ is strictly increasing and the cumulant function $K(\theta)$ is strictly convex.

3.2.2 Mean value parametrization

Because $\mu(\theta)$ is strictly increasing in θ, we can also parameterize a univariate exponential family by its mean value. Suppose the inverse function of $\mu(\theta)$ is $\theta(\mu)$. By the inverse function theorem,

$$\theta'(\mu) = \frac{1}{V(\theta)}.$$

The exponential family can be alternatively written as

$$f(y; \mu) = e^{\theta(\mu)y - K(\theta(\mu))} f_0(y)$$

for $\mu \in \mathcal{M} = \{\mu(\theta) \mid \theta \in \Theta\}$. The set $\mathcal{M}$ is usually referred to as the *mean space*. When using the mean-value parameterization, we often write the variance function as $V(\mu)$.

Example 3.8. Continuing from Examples 3.2 to 3.4, the natural parameter of $N(\mu, 1)$ is $\theta(\mu) = \mu$ and the cumulant function is $K(\theta) = \theta^2/2$. Therefore, the mean and variance functions are

$$\mu(\theta) = \theta, \ V(\theta) = 1.$$

For Poisson(λ), the natural parameter is $\theta = \log \lambda$ and $K(\theta) = e^\theta - 1$. Therefore, its mean and variance functions are given by

$$\mu(\theta) = V(\theta) = e^\theta = \lambda.$$

For Bernoulli(π) = Binomial($1, \pi$), the natural parameter is $\theta(\mu) = \log\{\pi/(1-\pi)\}$ and the cumulant function is $K(\theta) = \log(1 + e^{\theta})$. Therefore, its mean and variance functions are given by

$$\begin{aligned} \mu(\theta) &= \frac{e^{\theta}}{1+e^{\theta}} = \frac{1}{1+e^{-\theta}} = \pi, \\ V(\theta) &= \frac{e^{\theta}}{(1+e^{\theta})^2} = \pi(1-\pi). \end{aligned}$$

Exercise 3.9. Derive the mean and variance of the negative binomial distribution.

3.2.3 *Bayesian posterior distribution

Because the exponential family density function (3.1) is symmetric in the natural parameter and sufficient statistic, one can carefully choose a prior distribution so that the posterior update is simple. We demonstrate this with an example.

Example 3.10. Suppose $Y \sim \text{Binomial}(n, \pi)$. We saw in Example 3.4 that

$$f(y;\theta) \propto \pi^{y}(1-\pi)^{n-y} = e^{\theta y - n\log(1+e^{\theta})},$$

where $\theta = \log\{\pi/(1-\pi)\}$ is the natural parameter. The key idea is keep the same form in the prior:

$$\pi(\theta) \propto \pi^{\alpha_1}(1-\pi)^{\alpha_2} = e^{\alpha_1\theta - (\alpha_1+\alpha_2)\log(1+e^{\theta})}.$$

We recognize that this can be normalized if $\alpha_1 > -1$ and $\alpha_2 > -1$. In fact, $\pi(\theta)$ is the Beta($\alpha_1 - 1, \alpha_2 - 1$) distribution, and the posterior distribution is simply $\pi(\theta \mid Y) = \text{Beta}(\alpha_1 - 1 + Y, \alpha_2 - 1 + n - Y)$.

The posterior update in this example is simple because the prior and posterior distributions belong to the same exponential family (Beta distribution). family is the primary example in which such *conjugate priors* exist.

Building further on this symmetry, we can get a simple formula for the posterior mean of θ. Suppose we observe $Y \in \mathbb{R}$ from a one-parameter exponential family

$$f(y;\theta) = e^{\theta y - K(\theta)} f_0(y),$$

and $\theta \in \Theta \subseteq \mathbb{R}$ itself has a prior density $\theta \sim \pi(\theta)$. Let $f(y)$ be the marginal density

$$f(y) = \int_{\Theta} \pi(\theta) f(y;\theta)\, d\theta.$$

By using the Bayes formula, the posterior distribution of θ is given by

$$\begin{aligned} \pi(\theta \mid Y = y) &= \frac{\pi(\theta) f(y;\theta)}{f(y)} \\ &= \frac{\pi(\theta) e^{\theta y - K(\theta)} f_0(y)}{f(y)} \\ &= e^{y\theta - \log\{f(y)/f_0(y)\}} \pi(\theta) e^{-K(\theta)}. \end{aligned}$$

This is an exponential family with natural parameter y, sufficient statistic θ, and cumulant function $\log\{f(y)/f_0(y)\}$ (up to a constant). Thus, the poterior mean of θ is given by

$$\mathsf{E}(\theta \mid Y = y) = \frac{d}{dy}\log\{f(y)/f_0(y)\} = \frac{f'(y)}{f(y)} - \frac{f_0'(y)}{f_0(y)}.$$

As an application of this, suppose $Y \sim \mathrm{N}(\mu, \sigma^2)$, where σ^2 is known and μ has a prior density $\pi(\mu)$. By plugging in the natural parameter $\theta = \mu/\sigma^2$ and

$$f_0(y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{y^2}{2\sigma^2}},$$

we obtain *Tweedie's formula*:

$$\mathsf{E}(\mu \mid Y) = Y + \sigma^2 \frac{f'(Y)}{f(Y)}. \tag{3.5}$$

3.2.4 *Empirical Bayes

Suppose we observe independent variables $Y_i \sim \mathrm{N}(\mu_i, \sigma^2), i = 1, \dots, n$, where the mean parameters are generated by $\mu_i \overset{\text{i.i.d.}}{\sim} \pi(\mu), i = 1, \dots, n$ but the density $\pi(\mu)$ is unknown. Suppose σ^2 is known and let $Y = (Y_1, \dots, Y_n)$ and $\mu = (\mu_1, \dots, \mu_n)$. In this model, the MLE of μ is given by $\hat{\mu} = Y$, with risk

$$\mathsf{E}\left(\|\hat{\mu} - \mu\|^2\right) = \mathsf{E}\left(\|Y - \mu\|^2\right) = n\sigma^2.$$

But we cannot apply the usual asymptotic efficiency theory for MLE here because the dimension of the parameter is not fixed. In fact, the James-Stein estimator

$$\hat{\mu}_{\mathrm{JS}} = \left(1 - \frac{(p-2)\sigma^2}{\|Y\|^2}\right) Y.$$

has a strictly smaller mean squared error than the MLE $\hat{\mu}$ for all values of μ, a result that should be proved in *Principles of Statistics*.

This phenomenon can be best understood in the empirical Bayes framework. If the prior $\pi(\mu)$ is known, the optimal estimator of μ under the mean squared error is given by the posterior mean (the "Bayes estimator"):

$$\hat{\mu}_{\mathrm{Bayes},i} = \mathsf{E}(\mu_i \mid Y_i) = Y_i + \sigma^2 \frac{f'(Y_i)}{f(Y_i)}.$$

This is not a real (frequentist) estimator because f is unknown. Nonetheless, in this case we can plug in an estimator of the marginal density f using the data:

$$\hat{\mu}_{\mathrm{EB},i} = \mathsf{E}(\mu_i \mid Y_i) = Y_i + \sigma^2 \frac{\hat{f}'(Y_i)}{\hat{f}(Y_i)}.$$

This is an instance of *empirical Bayes* estimator, which uses a "prior distribution" estimated from the data. So empirical Bayes is a frequentist method motivated by Bayesian considerations.

Exercise 3.11. Show that the James-Stein estimator is the empirical Bayes estimator that assumes a normal prior: $\mu_i \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0, \tau^2)$. [*Hint:* use the fact that $\mathsf{E}\{(p-2)/\chi^2_p\} = 1$.]

What is remarkable about the James-Stein estimator is that it dominates the MLE even though the normal prior on μ may be wrong. It also provides a motivation for regularization (especially ridge regression, see Section 2.3.4).

Tweedie's formula (3.5) demonstrates a statistical concept called *shrinkage*, which is also closely related to regularization. The posterior mean $E(\mu \mid Y)$ is given by the MLE Y (the optimal unbiased estimator) plus a correction term $\sigma^2 f'(Y)/f(Y)$ which increases bias but decreases variance. When $f(\cdot)$ is unimodal, this correction term can be seen as a kind of "regression toward the mean" or a correction to "winner's curse"; see Figure 3.1 for an illustration.

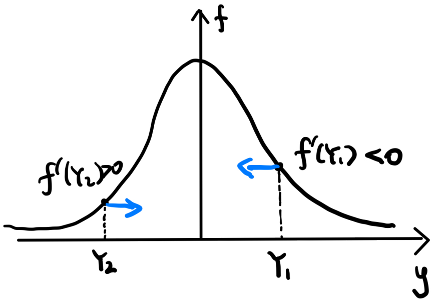


Figure 3.1: Tweedie's formula and shrinkage.

The James-Stein estimator and the idea of shrinkage were popularized by a baseball dataset, which contain the batting averages of 18 Major League players have been observed over the 1970 season. We would like to use the observed averages over the players' first 90 at bats to predict the average over the remainder of the season (370 further at bats on average). Figure 3.2[1] shows that the James-Stein estimator provides much more accurate predictions than the MLE.

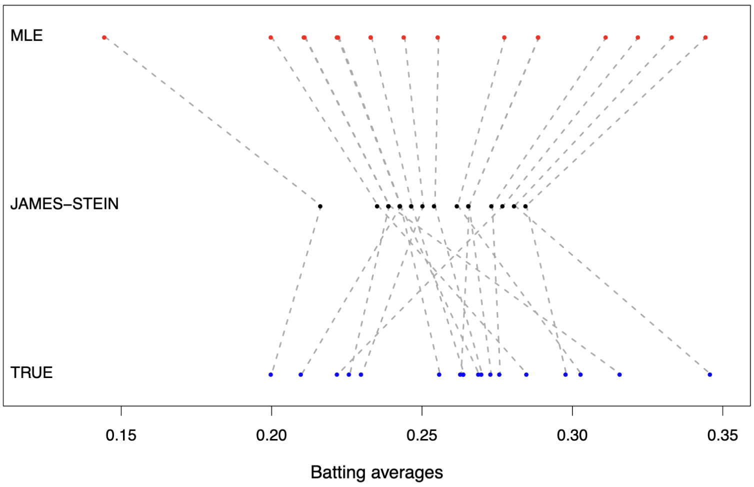


Figure 3.2: Application of the James-Stein estimator to a baseball dataset.

3.3 Likelihood inference

3.3.1 i.i.d. sampling

Suppose $Y_1, \ldots, Y_n \overset{\text{i.i.d.}}{\sim} f(y;\theta)$ where $f(y;\theta)$ is a one-parameter exponential family with sufficient statistic Y. Then their joint density is given by

$$\begin{aligned} f(y_1, \ldots, y_n; \theta) &= \prod_{i=1}^{n} f(y_i;\theta) \\ &= \prod_{i=1}^{n} e^{\theta y_i - K(\theta)} f_0(y_i) \\ &= e^{n\{\theta \bar{y} - K(\theta)\}} \prod_{i=1}^{n} f_0(y_i). \end{aligned}$$

This is a new exponential family with

- Natural parameter $\theta^{(n)} = n\theta$;
- Sufficient statistic $\bar{Y} = \frac{1}{n}\sum_{i=1} Y_i$;
- Cumulant function $K^{(n)}(\theta^{(n)}) = nK(\theta) = nK(\theta^{(n)}/n)$;
- Carrying density $\prod_{i=1}^{n} f_0(y_i)$.

This property allows us to easily extend results for a single random variable from exponential families to i.i.d. sampling. In fact, exponential families are the only statistical models that have this "sufficient dimension reduction" property.[2]

Exercise 3.12. Use the cumulant function above to show that $\mu^{(n)} = \mu$ and $V^{(n)} = V/n$.

3.3.2 Maximum likelihood estimator

Consider the setting of i.i.d. sampling above. The log-likelihood function is given by

$$l(\theta) = n\left\{\theta\bar{Y} - K(\theta)\right\} + \text{constant}. \tag{3.6}$$

The *score* is defined as the gradient of the log-likelihood, which in this case is given by

$$U(\theta) = l'(\theta) = n\{\bar{Y} - K'(\theta)\} = n\{\bar{Y} - \mu(\theta)\}. \tag{3.7}$$

The last equality uses (3.2), i.e. the first cumulant of a distribution is its mean.

The MLE $\hat{\theta} = \arg\max_\theta l(\theta)$ should satisfy the first-order condition $U(\hat{\theta}) = 0$, which means that

$$\hat{\mu} = \mu(\hat{\theta}) = \bar{Y}, \text{ or equivalent } \hat{\theta} = \theta(\bar{Y}).$$

In other words, the MLE simply matches the theoretical mean $\mu(\theta)$ with the observed mean $\bar{Y}$.

Example 3.13. For Poisson(μ), $\hat{\theta} = \log(\hat{\mu}) = \log(\bar{Y})$. For Binomial($n, \pi$) with fixed n, $\hat{\theta} = \log\{\hat{\pi}/(1-\hat{\pi})\}$ where $\hat{\pi} = \hat{\mu}/n = \bar{Y}/n$.

3.3.3 Asymptotic inference

The large-sample distribution of $\hat{\theta}$ can be obtained by the standard asymptotic theory for MLE. By (3.7), we obtain the following formula for the *Fisher information*

$$i^{(n)}(\theta) = \mathsf{Var}(U(\theta)) = nV(\theta).$$

Thus in exponentialy families, the Fisher information is simply the variance times n. Because $V(\theta) = K''(\theta)$, the above equation implies

$$i^{(n)}(\theta) = \mathsf{Var}\{l'(\theta)\} = \mathsf{E}\{-l''(\theta)\}, \tag{3.8}$$

This is called the *second Bartlett identity* and is true for all regular paramateric models.

Exercise 3.14. Prove (3.8) by differentiating the identity $\int f(y;\theta)\,dy = 1$ with respect to θ and interchanging differentiation and integral.

By the general asymptotic theory for MLE (Theorem 1.24), we have

$$\hat{\theta} \,\dot{\sim}\, \mathrm{N}\left(\theta, \frac{1}{i^{(n)}(\theta)}\right).$$

We sketch a proof of this result in exponential families. The standard approach is to take the first-order Taylor expansion of the score equation at $\hat{\theta} = \theta$:

$$0 = U(\hat{\theta}) \approx U(\theta) + U'(\theta)(\hat{\theta} - \theta). \tag{3.9}$$

By using (3.7) and the central limit theorem, we have

$$\frac{U(\theta)}{\sqrt{n}} = \sqrt{n}\{\bar{Y} - \mu(\theta)\} \overset{d}{\to} \mathrm{N}(0, V(\theta)).$$

Moreover, (3.7) implies that $U'(\theta) = nK''(\theta) = nV(\theta)$. Thus,

$$\sqrt{n}(\hat{\theta} - \theta) \approx -\frac{U(\theta)/\sqrt{n}}{U'(\theta)/n} \overset{d}{\to} -\frac{\mathrm{N}(0, V(\theta))}{V(\theta)} = \mathrm{N}\left(0, \frac{1}{V(\theta)}\right). \tag{3.10}$$

The calculations above are simplified by the fact that $U'(\theta)$ is a constant for exponential families. In the more general case, one can invoke the law of large numbers for $U'(\theta)$ and Slutsky's lemma. A rigorous proof will require bounding the error term in (3.9). This can be done through a "first-order analysis" that uses continuity of U' or a "higher-order analysis" that consider higher-order derivatives of U which shows the accuracy of the normal approximation depends on the skewness of the distribution (why?).

Exercise 3.15. Prove (3.10) by applying the delta method (Lemma 1.26) to $\hat{\theta} = \theta(\bar{Y})$.

3.3.4 Hypothesis testing

Consider testing a simple null hypothesis $H_0 : \theta = \theta_0$ against a simple alternative hypothesis $H_1 : \theta = \theta_1$ for some $\theta_1 > \theta_0$. By (3.6), the likelihood-ratio statistic is given by

$$l(\theta_1) - l(\theta_0) = n\left\{(\theta_1 - \theta_0)\bar{Y} - K(\theta_1) + K(\theta_0)\right\},$$

which is increasing in $\bar{Y}$. Thus, by the Neyman-Pearson Lemma, the most powerful level-α test rejects H_0 if $\bar{Y} > C_{1-\alpha}$, where $C_{1-\alpha}$ is the $(1-\alpha)$-quantile of $\bar{Y}$ under $\theta = \theta_0$. Because this test does not depend on θ_1 and controls the type I error for any null parameter value smaller than θ_0, it is indeed the uniformly most powerful test for $H_0 : \theta \leq \theta_0$ versus $H_1 : \theta > \theta_0$.

To test $H_0 : \theta = \theta_0$ against $H_1 : \theta \neq \theta_0$, one can resort to asymptotic arguments. The likelihood-ratio statistic is given by $l(\hat{\theta}) - l(\theta_0)$, which converges in distribution to $\chi^2_1/2$ as $n \to \infty$ by Wilks' theorem. It is not possible to obtain a uniformly most powerful test for this problem because a test based on the likelihood ratio cannot be simultaneous most powerful at $\theta_1 > \theta_0$ and $\theta_2 < \theta_0$. One solution is to restrict to *unbiased tests*, which has power $\geq$ size at any parameter value in the alternative hypothesis. It can be shown that the uniformly most powerful unbiased test exists for the one-parameter exponential family problem and rejects H_0 if $\bar{Y} < c_1$ or $\bar{Y} > c_2$, where c_1 and c_2 are determined by the condition that the power function reaches its minimum α at θ_0.

3.3.5 Deviance

Deviance is a measure of how one distribution in an exponential family differs from another:

$$\begin{aligned} D(\theta_1, \theta_2) &= 2\,\mathsf{E}_{\theta_1}\left\{\log \frac{f(Y;\theta_1)}{f(Y;\theta_2)}\right\} \\ &= 2\,\mathsf{E}_{\theta_1}\{(\theta_1 - \theta_2)Y - K(\theta_1) + K(\theta_2)\} \\ &= 2\{(\theta_1 - \theta_2)\mu_1 - K(\theta_1) + K(\theta_2)\}. \end{aligned} \tag{3.11}$$

If you are familiar with information theory, deviance is simply twice the Kullback-Leibler divergence.

Example 3.16 (Continuing Example 3.2)**.** For the family of normal distributions $N(\mu, 1)$, the natural parameter is $\theta = \mu$ and the cumulant function is $K(\theta) = \theta^2/2$. Therefore,

$$D(\mu_1, \mu_2) = 2\left\{(\mu_1 - \mu_2)\mu_1 - \frac{\mu_1^2}{2} + \frac{\mu_2^2}{2}\right\} = (\mu_1 - \mu_2)^2$$

coincides with squared Euclidean distance.

Heuristically, deviance can be thought of as an extension of the Euclidean geometry to exponential families, although generally it is not a distance metric (it is not symmetric and does not obey the triangle inequality). By rewriting μ_1 as $K'(\theta_1)$, we have

$$\frac{D(\theta_1, \theta_2)}{2} = K(\theta_2) - K(\theta_1) - (\theta_2 - \theta_1)K'(\theta_1).$$

Recall that the cumulant function $K(\theta)$ is convex. Thus, the last identity can be informatively represented by the picture in Figure 3.3, which is closely related to duality theory in convex analysis. In particular, this picture shows that the deviance can be locally approximated by the squared Euclidean distance times the Fisher information $i(\theta_1) = i^{(1)}(\theta_1) = V(\theta_1)$, which is curvature of the cumulant function $K(\theta)$ at θ_1:

$$D(\theta_1, \theta_2) \approx i(\theta_1)(\theta_2 - \theta_1)^2 \text{ for } \theta_2 \approx \theta_1. \tag{3.12}$$

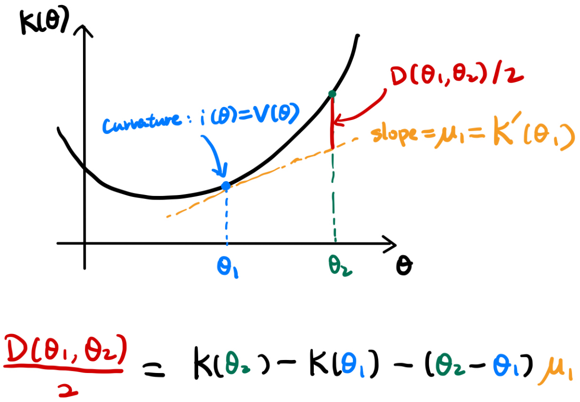

$K(\theta)$
Curvature : $i(\theta)=V(\theta)$
$D(\theta_1,\theta_2)/2$
slope $=\mu_1 = K'(\theta_1)$
θ_1
θ_2
θ
$\frac{D(\theta_1,\theta_2)}{2} = K(\theta_2) - K(\theta_1) - (\theta_2 - \theta_1)\mu_1$

Figure 3.3: An informative picture about the deviance in exponential families.

Exercise 3.17. Verify the following formulae:

- For Poisson(λ), the deviance is given by

$$D(\lambda_1, \lambda_2) = 2\left\{\lambda_1 \log \frac{\lambda_1}{\lambda_2} - \lambda_1 + \lambda_2\right\}.$$

- For Binomial(n, π) with fixed n, the deviance is given by

$$D(\pi_1, \pi_2) = 2n\left\{\pi_1 \log \frac{\pi_1}{\pi_2} + (1 - \pi_1) \log \frac{1 - \pi_1}{1 - \pi_2}\right\}.$$

Deviance also behaves nicely under i.i.d. sampling:

$$\begin{aligned} D^{(n)}(\theta_1, \theta_2) &= 2\,\mathsf{E}_{\theta_1}\left\{\log \prod_{i=1}^{n} \frac{f(Y_i;\theta_1)}{f(Y_i;\theta_2)}\right\} \\ &= \sum_{i=1}^{n} 2\,\mathsf{E}_{\theta_1}\left\{\log \frac{f(Y_i;\theta_1)}{f(Y_i;\theta_2)}\right\} \\ &= nD(\theta_1, \theta_2). \end{aligned}$$

Exercise 3.18. Show that, for one-parameter exponential family, the likelihood-ratio statistic for testing $H_0 : \theta = \theta_0$ versus $H_1 : \theta \neq \theta_0$ is given by $D^{(n)}(\hat{\theta}, \theta)$. Show that this statistic has a χ^2_1 asymptotic distribution under the null by using (3.12).

3.3.6 Deviance residual

Because deviance can be viewed as an extension to Euclidean distance, it allows us to extend the definition of residuals to exponential families. With an abuse of notation, we use $D(\mu_1, \mu_2)$ to denote the deviance between two distributions in the exponential family with mean μ_1 and μ_2.

In the normal linear model, $D(y, \mu) = (y - \mu)^2$ is the squared residual. On the other hand, the exponential family analogue of $y - \mu$ is given by

$$\operatorname{sign}(y - \mu)\sqrt{D(y, \mu)}.$$

With i.i.d. sampling, the total deviance is given by $D^{(n)}(\hat{\mu}, \mu) = nD(\bar{Y}, \mu)$. This motivates us to define the *deviance residual* by

$$R = \operatorname{sign}(\bar{Y} - \mu)\sqrt{D^{(n)}(\bar{Y}, \mu)}.$$

Exercise 3.19. Use Wilks' theorem to show that $R^2 \overset{d}{\to} \chi^2_1$ as $n \to \infty$.

In practice, deviance residual R is generally preferred over the Pearson residual

$$R_P = \frac{\bar{Y} - \mu}{\sqrt{V(\mu)/n}},$$

because the distribution of R is much less skewed and closer to the standard normal distribution.

3.4 Practical 5: Exponential family

3.4.1 Overdispersion due to clustering

In practice, it is not unusual that the empirical variance of a variable Y is larger than what is expected from a theoretical model. A common mechanism for overdispersion and underdispersion is unaccounted structure in the sample, as illustrated by the next example.

Example 3.20. Suppose a sample of size n has n/k clusters, each of size k. The observations are distributed as $Z_{ij} \sim \text{Bernoulli}(\pi_i), i = 1, \ldots, n/k, j = 1, \ldots, k$. The response Y is the total $Y = \sum_{i=1}^{n/k} \sum_{j=1}^{k} Z_{ij}$, which is often modelled by a binomial distribution. This is reasonable when $\pi_i = \pi$ for all i, as Y then follows a Binomial(n, π) distribution with

$$\mathsf{E}(Y) = n\pi, \ \mathsf{Var}(Y) = n\pi(1 - \pi).$$

However, if the probabilities π_i themselves are IID and

$$\mathsf{E}(\pi_i) = \pi, \ \mathsf{Var}(\pi_i) = \tau^2\pi(1 - \pi),$$

it can be shown by using the laws of total expectation and total variance that

$$\mathsf{E}(Y) = n\pi, \ \mathsf{Var}(Y) = \sigma^2 n\pi(1 - \pi), \ \text{where } \sigma^2 = 1 + \tau^2(k - 1).$$

That is, the mean of Y is unchanged but the variance is increased by a factor of σ^2.

Here is some R code that demonstrates this example

```
set.seed(42)
n <- 10000
k <- 50
one.sim <- function() {
    pi <- rbeta(n / k, 2, 2)
    Z <- rbinom(n, 1, pi)
    sum(Z)
}
Y <- replicate(1000, one.sim())
```

```
> c(mean(Y) / n, var(Y) / n)
[1] 0.499479 2.666703
> (sigma2.hat <- var(Y) / n / 0.5^2 - 1)
[1] 9.666813
> (sigma2.theory <- (k - 1) * 0.05 / 0.5^2)
[1] 9.8
```

To address the overdispersion problem, we may want to include a second parameter to model the variance of the distribution. A potential solution to this is the exponential dispersion family, with density function given by

$$f(y; \theta, \sigma^2) = e^{\{\theta y - K(\theta)\}/\sigma^2} f_0(y; \sigma^2), \tag{3.13}$$

where $f_0(y;\sigma^2)$ is some density function and $\sigma^2 > 0$ is called the *dispersion parameter.*

Exercise 3.21. Suppose the distribution of Y is given by (3.13). Show that $\mathsf{E}(Y_1) = K'(\theta)$ and $\mathsf{Var}(Y_1) = \sigma^2 K''(\theta)$.

3.4.2 *Bartlett correction

It is possible to give a better approximation to the distribution of the deviance residual R. The skewness and (excess) kurtosis of a probability distribution are defined as, respective,

$$\gamma = \kappa_3/\kappa_2^{3/2} = \frac{\mathsf{E}\{(Y - E(Y))^3\}}{\mathsf{Var}(Y)^{3/2}}, \quad \delta = \kappa_4/\kappa_2^2 = \frac{\mathsf{E}\{(Y - E(Y))^4\}}{\mathsf{Var}(Y)^2} - 3,$$

where κ_r is the rth cumulant of the distribution and Y is a random variable that follows that distribution. *Bartlett's correction* refers to the following approximation of the deviance residual[3]

$$R = \mathrm{N}\left(-a_n, (1 + b_n)^2\right) + O_p(n^{-3/2}),$$

where

$$a_n = \frac{\gamma}{6\sqrt{n}} \quad \text{and} \quad b_n = \frac{14\gamma^2 - 9\delta}{72n},$$

and γ and δ, just like μ in our notation, are the skewness and kurtosis of a single observation, respectively. The $O_p(n^{-3/2})$ error term means that

$$\mathsf{P}\left(\frac{R + a_n}{1 + b_n} > z_\alpha\right) = \alpha + O(n^{-3/2}),$$

where z_α is the upper-α quantile of $\mathrm{N}(0,1)$. Therefore, it is possible to obtain inexact but very accurate inference for small n by staying within the exponential family framework.

We demonstrate this correction using the Gamma family, with density function

$$f(y;\alpha,\lambda) = \frac{\lambda^\alpha}{\Gamma(\alpha)} y^{\alpha-1} e^{-\lambda y}, \ y > 0.$$

Using properties of the exponential family or otherwise, it can be shown that

(i) If $\alpha > 0$ is fixed, this is an exponential family with natural parameter $\theta = -\lambda$ and cumulant function $K(\theta) = -\alpha \log(-\theta)$.

(ii) The mean, variance, skewness, and kurtosis are given by

$$\mu = \kappa_1 = \alpha/\lambda, \ \kappa_2 = \alpha/\lambda^2, \ \gamma = 2/\sqrt{\alpha}, \ \delta = 6/\alpha.$$

(iii) If α is a positive integer and $Y_1, \dots, Y_\alpha \sim \mathrm{Gamma}(1,\lambda)$ are i.i.d., then $\sum_{i=1}^{\alpha} Y_i \sim \mathrm{Gamma}(\alpha,\lambda)$.

(iv) The deviance between $\mathrm{Gamma}(\alpha,\lambda_1)$ and $\mathrm{Gamma}(\alpha,\lambda_2)$ is given by

$$D(\mu_1,\mu_2) = 2\alpha\{\log(\mu_2/\mu_1) + (\mu_1/\mu_2 - 1)\},$$

where $\mu_1 = \alpha/\lambda_1$ and $\mu_2 = \alpha/\lambda_2$.

The next chunk of code computes the Pearson and deviance residuals for Gamma$(5, 1)$:

```
alpha <- 5
lambda <- 1
mu <- alpha / lambda
deviance.gamma <- function(mu1, mu2, alpha) {
  2 * alpha * (log(mu2 / mu1) + mu1 / mu2 - 1)
}
Y <- rgamma(1000000, alpha, 1)
dev <- deviance.gamma(Y, mu, alpha)
resid.dev <- sign(Y - mu) * sqrt(dev)
resid.pearson <- (Y - mu) / sqrt(alpha/lambda^2)
```

We now use the Bartlett correction and compare the residuals. The result can be found in Figure 3.4.

```
gamma <- 2 / sqrt(alpha)
delta <- 6 / alpha
a <- 2 / sqrt(alpha) / 6
b <- (14 * gamma^2 - 9 * delta) / 72
resid.bartlett <- (resid.dev + a) / (1 + b)

s <- 1:1000 # too many points to show
pl <- qqnorm(resid.dev[s], col = scales::alpha("red", 0.4))
points(pl$x, resid.pearson[s], col = scales::alpha("blue", 0.4))
abline(0, 1, lty = "dashed")
points(pl$x, resid.bartlett[s], col = scales::alpha("green", 0.4))
legend("topleft", c("Pearson", "Deviance", "Bartlett"),
       fill = c("blue", "red", "green"))
```

It is obvious from the Q-Q plot that the distribution of Pearson's residuals is far from normal. The deviance residuals have a distribution that is already quite close to the standard normal, and Barlett's correction mainly removes the small amount of bias.

```
> c(mean(resid.dev), -a)
[1] -0.1505926 -0.1490712
> c(sd(resid.dev) - 1, b)
[1] 0.005485308 0.005555556
```

3.4.3 Exercises

Exercise 3.22. Suppose we are given some data $X_1, \dots, X_n \sim$ Gamma(α, λ), where $n = 30$, $\alpha = 2$, and λ is unknown.

(i) Write a function gammaMLE in R that returns the maximum likelihood estimator $\hat{\lambda}$ of the rate parameter λ. Check your code by using simulations to verify that $\hat{\lambda}$ is consistent.

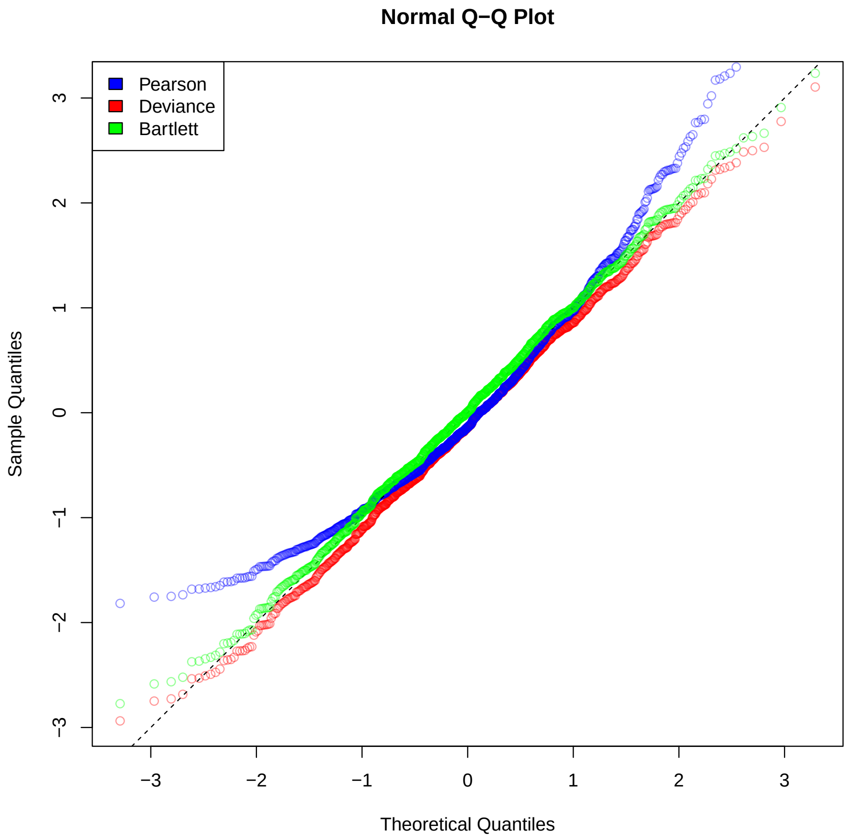


Figure 3.4: Demonstration of different residuals.

(ii) What is the variance of $\sqrt{n}(\hat{\lambda} - \lambda)$ based on your simulations, and does it match the theoretical value based on an asymptotic normal approximation? Augment your `gammaMLE` function with an option `interval=TRUE` in which case it will also return an 95% asymptotic confidence interval of λ. Calculate the proportion for which this interval covers the true parameter in 1000 simulations.

(iii) Find the exact distribution of $\hat{\lambda}$. Check your answer by comparing your simulated MLEs with the theoretical distribution using a Q-Q plot. *Hint: If* $X \sim$ *Gamma*(α, λ)*, then* $1/X \sim$ *InvGamma*(α, λ)*.*

(iv) Augment your `gammaMLE` function using the R function `optim` so it can return the MLE of (α, λ) when α is unknown. Apply this to your simulated data from earlier and plot these as pairs of points $(\hat{\alpha}, \hat{\lambda})$.

(v) The `optim` function has an option to return the observed information matrix by setting `Hessian = TRUE`. Use this to write a function that draws a confidence ellipse for the parameters and test their empirical coverage. *Hint: the* R *function* `eigen` *may be helpful for this.*

Chapter 4

Generalized linear models

4.1 Canonical GLMs and extensions

As discussed in Section 1.4, the normal linear model assumes that Y given X has linear conditional expectation and and is normally distributed. Several relaxations of this model are introduced in Chapter 2. All of them assume the linear model is at least a reasonable approximation to $\mathsf{E}(Y \mid X)$. In this Chapter we will introduce generalized linear models (GLMs) that expand the classical normal linear models in a different way.

To simplify the exposition, we will adopt the same vector/matrix notation as in linear models:

$$Y = \begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix}, \; X = \begin{pmatrix} X_1^T \\ \vdots \\ X_n^T \end{pmatrix}, \; \beta = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}, \; \mu = \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_n \end{pmatrix}.$$

Unless noted otherwise, we will assume the generalized linear model is correctly specified and treat the regressors X as fixed. In other words, the inference for GLMs will be conditioned on X.

4.1.1 The canonical form

Let $\{f(y;\theta) \mid \theta \in \Theta\}$ be a one-parameter exponential family. A *canonical form GLM* assumes that the responses $Y_1, \dots, Y_n$ are independent and

$$Y_i \mid X_i \sim f(y;\theta_i),\; i = 1,\dots,n,$$

where the natural parameter is given by

$$\theta_i = X_i^T\beta.$$

In other words, this model simply sets the natural parameter to be a linear function of X. An immediate consequence is that the mean parameter is given by

$$\mu_i = \mathsf{E}(Y_i \mid X_i) = \mu(\theta_i).$$

The joint density of Y is given by

$$\begin{aligned} f(y;\beta) &= \prod_{i=1}^{n} f(y_i;\theta_i) \\ &= e^{\sum_{i=1}^{n} \theta_i y_i - K(\theta_i)} \prod_{i=1}^{n} f_0(y_i) \\ &= e^{\beta^T X^T y - \sum_{i=1}^{n} K(X_i^T\beta)} \prod_{i=1}^{n} f_0(y_i). \end{aligned}$$

This is a p-parameter exponential family, with

- Natural parameter β;
- Sufficient statistic $Z = X^T Y$; and
- Cumulant function $\phi(\beta) = \sum_{i=1}^{n} K(X_i^T\beta)$.

Therefore, the canonical form GLMs can be studied using the theory for multi-parameter exponential families and have many nice properties that generalize the theory in Section 3.2. For example, it can be shown that the expectation and covariance matrix of Z are given by the gradient and Hessian matrix of the cumulant function:

$$\begin{aligned} \mathsf{E}_\beta(Z) &= \nabla\phi(\beta) = \sum_{i=1}^{n} K'(X_i^T\beta)X_i = X^T\mu(\beta), \\ \mathsf{Cov}_\beta(Z) &= \nabla^2\phi(\beta) = \sum_{i=1}^{n} K''(X_i^T\beta)X_iX_i^T = X^TV(\beta)X, \end{aligned}$$

where $\mu(\beta) = (\mu_1(\beta), \ldots, \mu_n(\beta))^T$ and

$$V(\beta) = \mathrm{diag}(K''(X_1^T\beta), \ldots, K''(X_n^T\beta)) = \mathrm{diag}(\mathsf{Var}(Y_1), \ldots, \mathsf{Var}(Y_n)).$$

The log-likelihood function of β is given by

$$l(\beta) = \beta^T X^T Y - \sum_{i=1}^{n} K(X_i^T\beta) + \text{ constant }.$$

The score function is given by

$$U(\beta) = \nabla l(\beta) = X^T Y - \sum_{i=1}^{n} K'(X_i^T\beta)X_i = X^T\{Y - \mu(\beta)\}.$$

Thus, the MLE $\hat{\beta}$ satisfies the normal equations

$$X^T\{Y - \mu(\hat{\beta})\} = 0. \tag{4.1}$$

Geometrically, the MLE is obtained by projecting Y onto $\{\mu(\beta) \mid \beta \in R^p\}$, a p-dimensional manifold in $\mathbb{R}^n$. Of course, the GLM normal equations (4.1) reduce to the normal equations (1.6) for the normal location family.

For asymptotic inference of GLMs, the one-dimensional theory in Section 3.3.3 can be extended in a straightforward manner. The *Fisher information matrix* for β is defined as

$$I^{(n)}(\beta) = \mathsf{Cov}\{U(\beta)\} = \mathsf{E}\{-\nabla^2 l(\beta)\} = \sum_{i=1}^{n} K''(X_i^T\beta)X_iX_i^T = X^TV(\beta)X.$$

The asymptotic theory for the MLE suggests that, under suitable regularity conditions,

$$\hat{\beta} \dot{\sim} \mathrm{N}(\beta, I^{(n)}(\beta)^{-1}).$$

This is an informal way of writing the convergence in distribution

$$\sqrt{n}(\hat{\beta} - \beta) \overset{d}{\to} \mathrm{N}(0, I(\beta)^{-1}), \text{ as } n \to \infty,$$

where $I(\beta) = \lim_{n\to\infty} I^{(n)}(\beta)/n$ is assumed to exist.

4.1.2 Analysis of deviance

The deviance extends the RSS/variance in normal linear models as a way to measure the goodness-of-fit of a GLM. Recall that in a one-parameter exponential family $\{f(y;\theta) \mid \theta \in \Theta\}$, the deviance between $f(y;\theta_1)$ and $f(y;\theta_2)$ is defined as

$$\begin{aligned} D(\theta_1,\theta_2) &= 2\,\mathsf{E}_{\theta_1}\{\log f(Y;\theta_1) - \log f(Y;\theta_2)\} \\ &= 2\{(\theta_1 - \theta_2)\mu_1 - K(\theta_1) + K(\theta_2)\}. \end{aligned}$$

As discussed in Section 3.3.5, deviance extends the Euclidean geometry to exponential families. With an abuse of notation, it is often convenient to parameterize an exponential family distribution by its mean and write the deviance as $D(\mu_1, \mu_2)$. For two n-vectors of mean-value parameters μ_1 and μ_2, their *total deviance* is defined as

$$D^{(n)}(\mu_1,\mu_2) = \sum_{i=1}^{n} D(\mu_{1i}, \mu_{2i}).$$

Nested models

The *full model* is given by

$$\theta = \eta = X\beta,$$

where $X \in \mathbb{R}^{n\times p}$ and $\beta \in \mathbb{R}^p$. As in Section 1.4.2, suppose the design matrix X and coefficient vector β are partitioned as

$$X = (X_0\ X_1),\ \beta = \begin{pmatrix}\beta_0 \\ \beta_1\end{pmatrix},$$

where $X_0 \in \mathbb{R}^{n\times p_0}$, $X_1 \in \mathbb{R}^{n\times(p-p_0)}$, $\beta_0 \in \mathbb{R}^{p_0\times 1}$, and $\beta_1 \in \mathbb{R}^{(p-p_0)\times 1}$. The *submodel* or *null model* we consider is

$$\theta = \eta = X_0\beta_0.$$

In other words, we are interested in testing the hypothesis $H_0 : \beta_1 = 0$ against $H_1 : \beta_1 \neq 0$.

According to the GLM normal equations (4.1), the full model MLE $\hat\beta \in \mathbb{R}^p$ and submodel MLE $\hat\beta \in \mathbb{R}^{p_0}$ satisfy

$$X^T\{Y - \hat\mu)\} = 0, \text{ where } \hat\mu = \mu(X\hat\beta) = \begin{pmatrix}\mu(X_1^T\hat\beta)\\ \vdots \\ \mu(X_n^T\hat\beta)\end{pmatrix}; \text{ and}$$

$$X_0^T\{Y - \hat\mu_0)\} = 0, \text{ where } \hat\mu_0 = \mu(X_0\hat\beta_0) = \begin{pmatrix}\mu(X_1^T\hat\beta_0)\\ \vdots \\ \mu(X_n^T\hat\beta_0)\end{pmatrix}.$$

See Figure 4.1 for an geometric illustration of nested GLM fits.

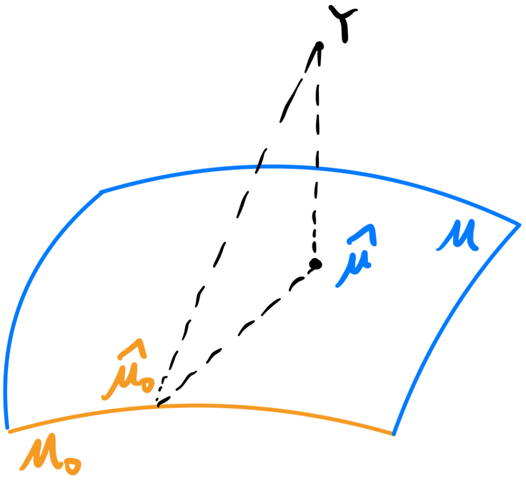


Figure 4.1: Illustration of nested GLMs. The full model space is given by $\mathcal{M} = \{\mu(X\beta) \mid \beta \in \mathbb{R}^p\}$ and the submodel space is given by $\mathcal{M}_0 = \{\mu(X_0\beta_0) \mid \beta_0 \in \mathbb{R}^{p_0}\}$.

A GLM is called *saturated* if X has rank n (which implies $p \geq n$). In this case, $\hat\mu = Y$. Assuming the intercept is also included in the model, the smallest GLM is given by $p = 1$ and $X = 1_n$. In this case, $\hat\mu = \bar{Y}1_n$.

Deviance additivity

The *deviance additivity theorem* says that the deviance between the observations (or equivalently the saturated model) and the submodel can be decomposed as

$$D^{(n)}(Y, \hat\mu_0) = D^{(n)}(Y, \hat\mu) + D^{(n)}(\hat\mu, \hat\mu_0). \tag{4.2}$$

This equation follows immediately from the following relation between the deviance and log-likelihood:

$$D^{(n)}(\hat{\mu}, \hat{\mu}_0) = 2\left\{l(\hat{\mu}) - l(\hat{\mu}_0)\right\}, \tag{4.3}$$

where $l(\mu) = \sum_{i=1}^n \log f(Y_i; \theta(\mu_i))$ is the log-likelihood function. In Example 3.16, we saw that in normal linear models with $\sigma^2 = 1$, we have $D^{(n)}(\mu_1, \mu_2) = \|\mu_1 - \mu_2\|^2$. So in this case the deviance additivity theorem reduces to Pythagoras' theorem.

Exercise 4.1. Show (4.3), then use it to prove (4.2).

By Wilks' theorem and (4.3), we have $D^{(n)}(\hat{\mu}, \hat{\mu}_0) \overset{d}{\to} \chi^2_{p-p_0}$ as $n \to \infty$ under the null $H_0 : \beta_1 = 0$. So we reject H_0 if

$$D^{(n)}(\hat{\mu}, \hat{\mu}_0) > \chi^2_{p-p_0}(\alpha).$$

With a sequence of nested GLMs, one can further perform a chain of analyses of deviance.

4.1.3 Linkage and over-dispersion

The canonical form GLM can be extended in several ways:

(i) Use a dispersion parameter σ^2 to model the variance of Y.

(ii) Use a different *link function* g to relate μ_i to the *linear preidctor* $\eta_i = \beta^T X_i$ by $\eta_i = g(\mu_i)$.

In this more general setup, it is assumed that $Y_1, \dots, Y_n$ are independent and $Y_i \sim f(y; \theta_i, \sigma_i^2)$ follows a distribution from a exponential dispersion family

$$f(y; \theta, \sigma^2) = e^{\{\theta y - K(\theta)\}/\sigma^2} f_0(y; \sigma^2),$$

with the natural and dispersion parameters modelled by

$$\theta_i = \theta(\mu_i) = \theta(g^{-1}(\eta_i)) = \theta(g^{-1}(X_i^T \beta)) \quad \text{and} \quad \sigma_i^2 = \sigma^2 / w_i,$$

where g is a strictly increasing and twice differentiable function, $\sigma^2 > 0$ is possibly unknown, and $w_i, i = 1, \dots, n$ are some known weights.

Example 4.2. The familiar normal linear model corresponds to assuming $Y_i \sim \mathrm{N}(\theta_i, \sigma_i^2)$, $w_i = 1$, and $\theta_i = \mu_i = \eta_i$. So g is the identity function.

The canonical form GLM corresponds to equating θ_i with η_i, so it uses the *canonical link* $g(\mu) = \theta(\mu)$. Non-canonical link functions can be useful in some latent variable models. They break some useful geometrical properties of exponential families, but much of the (first-order) asymptotic theory still goes through. It can be shown that the β-score is given by

$$U_\beta(\beta, \sigma^2) = \nabla_\beta\, l(\beta, \sigma^2) = \frac{1}{\sigma^2} X^T W R \tag{4.4}$$

where

$$W = W(\beta) = \operatorname{diag}\left(\frac{w_i}{V(\mu_i)\{g'(\mu_i)\}^2}\right), \ R = \begin{pmatrix} R_1 \\ \vdots \\ R_n \end{pmatrix}, \ R_i = (Y_i - \mu_i)g'(\mu_i), \tag{4.5}$$

and the Fisher information matrix for (β, σ^2) is block-diagonal:

$$I^{(n)}(\beta, \sigma^2) = \begin{pmatrix} I^{(n)}_{\beta\beta}(\beta, \sigma^2) & 0 \\ 0 & I^{(n)}_{\sigma^2\sigma^2}(\beta, \sigma^2) \end{pmatrix}, \tag{4.6}$$

where

$$I^{(n)}_{\beta\beta}(\beta, \sigma^2) = \frac{1}{\sigma^2} X^T W X. \tag{4.7}$$

Exercise 4.3. Consider the general GLM introduced above.

(i) Derive (4.4), (4.6), and (4.7).

(ii) When σ^2 is unknown, show that a consistent estimator is

$$\hat{\sigma}^2 = \frac{1}{n-p} \sum_{i=1}^{n} w_i \frac{(Y_i - \hat{\mu}_i)^2}{V(\hat{\mu}_i)}.$$

(iii) Use the results above to construct an asymptotic $(1 - \alpha)$ confidence interval for $\beta_j, j \in \{1, \dots, p\}$.

4.2 Numerical computation and model selection

Up till now, we have not said anything about how the MLE $\hat{\beta}$ can be computed. Unlike in the normal linear model where $\hat{\beta}$ can be found by solving some linear equations, the score equations (4.1) for GLMs are not linear in β. Thus, some iterative algorithms are needed.

4.2.1 Newton-Raphson

The Newton-Raphson algorithm is a general algorithm for optimization or root finding problems. We illustrate this with a classical problem in statistics—finding the MLE. Consider the optimization problem

$$\underset{\beta \in \mathbb{R}^p}{\text{maximize}} \quad l(\beta),$$

where $l(\beta)$ is the log-likelihood function for some statistics problem. Let $U(\beta)$ and $H(\beta)$ be the gradient and Hessian matrix of $l(\beta)$ at β. That is,

$$\begin{aligned} U_k(\beta) &= \frac{\partial}{\partial \beta_k} l(\beta),\ k = 1, \dots, p, \\ H_{jk}(\beta) &= \frac{\partial^2}{\partial \beta_j \partial \beta_k} l(\beta),\ j, k = 1, \dots, p. \end{aligned}$$

The key idea of the Netwon-Raphson algorithm is that the objective function $l(\beta)$ can be locally approximate near $\beta^* \in \mathbb{R}^p$ by its second-order Taylor expansion (assuming the function is sufficiently smooth):

$$l(\beta) \approx l(\beta^*) + (\beta - \beta^*)^T U(\beta^*) + \frac{1}{2}(\beta - \beta^*)^T H(\beta^*)(\beta - \beta^*).$$

Because the local approximation is a quadratic function of β, we can easily find its maximizer. By differentiating with respect to β, the maximizer should satisfy

$$U(\beta^*) + H(\beta^*)(\beta - \beta^*) = 0.$$

This motivates the following iterative algorithm (see Figure 4.2):

(i) Start at an initial parameter value $\beta^{(0)}$.

(ii) For $t = 1, 2, \dots$, update the parameter by

$$\beta^{(t)} = \beta^{(t-1)} - \left\{ H(\beta^{(t-1)}) \right\}^{-1} U(\beta^{(t-1)}).$$

(iii) Stop the algorithm until the sequence $\beta^{(t)}$ converges in a numerical sense (e.g. if $l(\beta^{(t)}) - l(\beta^{(t-1)}) < \tau$ where τ is some tolerance level).

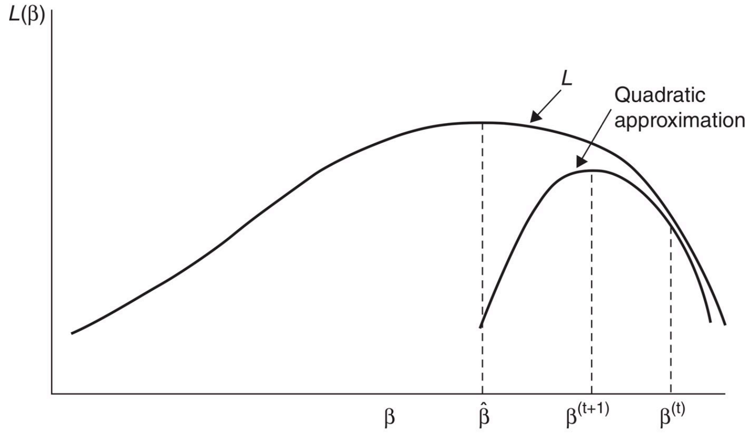


Figure 4.2: An illustration of the Newton-Raphson algorithm.[4]

4.2.2 Fisher scoring

A drawback of the Newton-Raphson algorithm is that the Hessian matrix $H(\beta^{(t-1)})$ is sometimes close to singularity, making its inverse numerically unstable.

When the objective function $l(\beta)$ is the log-likelihood of some IID data, the Fisher information matrix is the expectation of the negative Hessian matrix (which is sometimes called the observed information):

$$I(\beta) = \mathsf{E}_\beta\{-H(\beta)\}.$$

The Fisher information matrix is guaranteed to be positive definite. *Fisher scoring* refers to the modification of the Newton-Raphson algorithm where $-H(\beta^{(t-1)})$ is replaced by $I(\beta^{(t-1)})$. In machine learning, this technique is known as the *natural gradient* method.

4.2.3 Iteratively reweighted least squares

Let us now apply the general algorithms above to GLMs. For the most general form of GLM described in Section 4.1.3, the log-likelihood function is given by

$$l(\beta, \sigma^2) = \sum_{i=1}^{n} \frac{1}{\sigma_i^2}\{\theta_i Y_i - K(\theta_i)\} + \log f_0(Y_i; \sigma_i^2),$$

where $\theta_i = \theta(g^{-1}(X_i^T\beta))$ and $\sigma_i^2 = \sigma^2/w_i$. The Hessian matrix can be obtained by differentiating the β-score given in (4.4) and is rather complicated. The calculations greatly simplify if $g(\mu)$ is the canonical link, i.e. $g(\mu) = \theta(\mu)$. In this case, the negative Hessian matrix (a.k.a. the observed information matrix) is indeed equal to the Fisher information matrix:

$$-H_{\beta,\beta}(\beta, \sigma^2) = I^{(n)}_{\beta,\beta}(\beta, \sigma^2) = \frac{1}{\sigma^2}X^T W X, \tag{4.8}$$

where W is defined in (4.5). So for GLMs using a canonical link function, the Newton-Raphson algorithm coincides with the Fisher scoring algorithm.

Exercise 4.4. Prove (4.8) when $g(\mu)$ is the canonical link function.

The Fisher scoring algorithm admits a nice representation in the GLM problem. Recall that the β-score is given by

$$U_\beta(\beta, \sigma^2) = \frac{1}{\sigma^2} X^T W R,$$

where the "weights" W and "residuals" R depend on β. Let $\eta_i^{(t)} = X_i^T \beta^{(t)}$, $\mu_i^{(t)} = g^{-1}(\eta_i^{(t)})$, and similarly define $W^{(t)}$ and $R^{(t)}$. The Fisher scoring update is then given by

$$\begin{aligned}
\beta^{(t)} &= \beta^{(t-1)} + \{I_{\beta\beta}^{(n)}(\beta^{(t-1)}, \sigma^2)\}^{-1} U_\beta(\beta^{(t-1)}, \sigma^2) \\
&= \beta^{(t-1)} + \left(X^T W^{(t-1)} X\right)^{-1} X^T W^{(t-1)} R^{(t-1)} \\
&= \left(X^T W^{(t-1)} X\right)^{-1} X^T W^{(t-1)} \left(\eta^{(t-1)} + R^{(t-1)}\right).
\end{aligned}$$

The last expression is the solution to a weighted least squares problem (Section 2.2.1). Therefore, the Fisher scoring algorithm for GLMs is also known as the *iteratively reweighted least squares* that updates the model parameters as follows

$$\beta^{(0)} \to \eta^{(0)}, \mu^{(0)} \to W^{(0)}, R^{(0)} \overset{\text{WLS}}{\to} \beta^{(1)} \to \eta^{(1)}, \mu^{(1)} \to \cdots$$

To initiate the algorithm, it is common to choose $\beta^{(0)} = 0$ or $\mu^{(0)} = Y$.

4.2.4 Model diagnostics

The diagnosis of GLMs generalizes that of linear models and is built on the iteratively reweighted least squares formulation of the MLE. The key idea is to define the pseudo-response:

$$Z^{(t)} = \eta^{(t)} + R^{(t)}$$

So the Fisher scoring update can be written as

$$\beta^{(t)} = \left(X^T W^{(t-1)} X\right)^{-1} X^T W^{(t-1)} Z^{(t-1)}.$$

By letting $\hat{Z} = \lim_{t\to\infty} Z^{(t)}$ and $\hat{R} = \lim_{t\to\infty} R^{(t)}$, we obtain

$$\hat{Z} = X\hat{\beta} + \hat{R}.$$

The GLM diagnosis basically proceeds by treating $\hat{W}^{1/2}\hat{Z}$, $\hat{W}^{1/2}\hat{\eta}$, and $\hat{W}^{1/2}\hat{R}$ as the "adjusted" responses, fitted values, and residuals. One can then apply the diagnostic plots in Section 2.2.1 after suitably modifying the definitions of leverage, residual, etc.

4.2.5 Model selection

To select a GLM, we cannot apply Mallows' C_p criterion because it relies on mean squared error. However, we can still use cross-validation by replacing the squared error with the deviance. In other words, we seek a model that minimizes

$$\mathrm{CV}(\text{model}) = \sum_{i=1}^{n} D(Y_i, \hat{\mu}_{-i}),$$

where $\hat{\mu}_{-i}$ is the leave-one-out fitted value for the ith observation. AIC and BIC can be applied in the same way to GLMs by using the corresponding log-likelihood function.

Regarding algorithms for model selection, the stepwise methods and the best subset method can be applied in the same way as before. Regularization can be achieved by adding the same penalty on certain complexity measure of β as in the linear model.

4.3 Binomial regression

In the rest of this Chapter, we discuss two of the most widely used families of GLMs: binomial regression and Poisson regression.

In a binomial regression, it is assumed that the responses $Y_1, \ldots, Y_n$ are independent and

$$Y_i \sim \frac{1}{n_i}\text{Binomial}(n_i, \mu_i),\ i = 1, \ldots, n,$$

where n_i is known but μ_i is unknown. It is not difficult to shown that Binomial(n, μ) is an exponential dispersion family:

$$\begin{aligned} f(y; n, \mu) &= \binom{n}{ny} \mu^{ny} (1-\mu)^{n(1-y)} \\ &= \exp\left\{ \frac{1}{n^{-1}} \left(y \log \frac{\mu}{1-\mu} + \log(1-\mu) \right) \right\} \binom{n}{ny}. \end{aligned}$$

The natural parameter is $\theta = \log\{\mu/(1-\mu)\}$, and the cumulant function is given by $K(\theta) = \log(1 + e^\theta)$. In the GLM context, we can fix dispersion parameter at $\sigma^2 = 1$ and use the dispersion weight $w = n$.

4.3.1 Common link functions

Recall that the link function relates the linear predictor with the mean value. Specifically, $g(\mu_i) = \eta_i = X_i^T \beta$. The canonical link makes η_i equal to the natural parameter θ_i, so for binomial regression the canonical link is given by the logit function

$$g(\mu) = \theta(\mu) = \log \frac{\mu}{1-\mu}.$$

More generally, we can choose $g(\mu)$ to be any strictly increasing function from $(0, 1)$ to $\mathbb{R}$.[5] In other words, we can let g to be the quantile function (inverse of the CDF) of any continuous random variable ϵ. The logit link corresponds to the *logistic distribution*, whose distribution function is simply the expit function:

$$F(\eta) = \frac{e^\eta}{1 + e^\eta}.$$

Another commonly used link is the *probit link* $g(\mu) = \Phi^{-1}(\mu)$, which corresponds to letting $\epsilon \sim \text{N}(0, 1)$. Some less common link functions include the *identity link* $g(\mu) = \mu$ and the *complementary log-log (cloglog) link* $g(\mu) = \log\{-\log(1-\mu)\}$.

4.3.2 Latent variable interpretation

The above quantile function viewpoint provides an interesting interpretation of the link functions for the binomial regression. To illustrate this, suppose $n_i = 1,\ i = 1, \ldots, n$. Let

$$Y^* = \eta + \epsilon,\ \epsilon \sim F(\cdot),\ Y = 1_{\{Y^* > 0\}},$$

where $F(\cdot)$ is the CDF of some continuous probability distribution. Then the mean value of Y can be given by

$$\mu = \mathsf{E}(Y) = \mathsf{P}(Y^* > 0) = \mathsf{P}(\epsilon > \eta) = 1 - F(-\eta).$$

Thus, if the distribution is symmetric about 0,

$$\eta = -F^{-1}(1-\mu) = F^{-1}(\mu).$$

This formulation is quite useful because it allows us to fit a linear model to the latent variable Y^* using just the sign of Y^*, as long as the noise distribution is known. For example, this allows us to learn about some latent "disease liability" Y^* given just an indicator of the diesease Y.

The cloglog link arises from a similar model in which the latent variable is distributed as

$$Y^* \sim \text{Poisson}(e^{\eta}).$$

Note that $\mu = e^{\eta}$ is in fact the canonical link for Poisson regression (see Section 4.5). Suppose the observation is still given by $Y = 1_{\{Y^*>0\}}$, then

$$1 - \mu = \mathsf{P}(Y = 0) = \mathsf{P}(Y^* = 0) = e^{-e^{\eta}},$$

which results in the cloglog link $\eta = \log(-\log(1-\mu))$.

4.3.3 Logistic regression and odds ratio

The logit link (aka the *logistic regression*) is by far the most popular for binomial regression. Beyond the fact that it enjoys some nice properties being the caonical link, it has some other advanrages. First, in logistic regression the odds of an observation is given by

$$\frac{\mathsf{P}(Y_i = 1)}{\mathsf{P}(Y_i = 0)} = \frac{\mu_i}{1-\mu_i} = e^{\eta_i} = e^{X_i^T\beta} = \prod_{j=1}^{p} \left(e^{\beta_j}\right)^{X_{ij}}.$$

Therefore, e^{β_j} represents a multiplicative change to the odds per nuit change of the jth regressor.[6]

Moreover, when we just have a single binary regressor, consider the saturated model

$$\log \frac{\mu}{1-\mu} = \eta = \beta_0 + \beta_1 X,$$

where $\mu = \mathsf{E}(Y \mid X) = \mathsf{P}(Y = 1 \mid X)$. Then the difference in odds ratio for different levels of X is given by

$$\log \frac{\mathsf{P}(Y = 1 \mid X = 1)}{\mathsf{P}(Y = 0 \mid X = 1)} - \log \frac{\mathsf{P}(Y = 1 \mid X = 0)}{\mathsf{P}(Y = 0 \mid X = 0)} = (\beta_0 + \beta_1) - \beta_0 = \beta_1.$$

Therefore, the *odds ratio* is given by

$$\frac{\mathsf{P}(Y = 1 \mid X = 1)/\,\mathsf{P}(Y = 0 \mid X = 1)}{\mathsf{P}(Y = 1 \mid X = 0)/\,\mathsf{P}(Y = 0 \mid X = 0)} = e^{\beta_1}.$$

The odds ratio is a useful quantity because it enjoys a symmetry:

$$\frac{\mathsf{P}(Y=1 \mid X=1)/\,\mathsf{P}(Y=0 \mid X=1)}{\mathsf{P}(Y=1 \mid X=0)/\,\mathsf{P}(Y=0 \mid X=0)} = \frac{\mathsf{P}(X=1 \mid Y=1)/\,\mathsf{P}(X=0 \mid Y=1)}{\mathsf{P}(X=1 \mid Y=0)/\,\mathsf{P}(X=0 \mid Y=0)}. \tag{4.9}$$

This neat property implies that we can sample from a population according to Y (suppose $Y = 1$ means a case), and it does not bias the odds ratio. For example, in case-control studies for rare diseases, we can pair each case (e.g. a patient suffering from the disease) with a control (e.g. a healthy individual). This is much more efficient than a random sample from the population, which may contain very few cases. For rare diseases, the odds ratio offers a good approximation to the more interpretable *risk ratio*, defined as $\mathsf{P}(Y = 1 \mid X = 1)/\,\mathsf{P}(Y = 1 \mid X = 0)$, because $\mathsf{P}(Y = 0)$ is very close to 1.

4.4 Practical 6: Binomial GLM

4.4.1 Binary responses

A dataset from Hosmer, D.W., Lemeshow, S. and Sturdivant, R.X. (2013) *Applied Logistic Regression* concerns factors affecting the development of myopia (short-sightedness) in children. Our response is binary, 1 indicating that the child developed myopia within the period of study, and 0 indicating they did not. The explanatory variables available to us are the gender of the child, the myopia status of each of their parents, and the number of hours per week spent by the child outside school on various activities: sport, reading for pleasure, playing computer games, studying for school assignments and watching television.

```
path <- "http://www.statslab.cam.ac.uk/~rds37/teaching/statistical_modelling/"
Myopia <- read.csv(file.path(path, "Myopia.csv"))
head(Myopia)
```

To perform logistic regression (i.e. regression with binomial responses and the logit link function, so the canonical link in this case), or to fit any generalised linear model, we can use the `glm` function (as usual with a new function, do `?glm` to find out more). This works rather like the `lm` in terms of how arguments must be specified and also in terms of procedures for accessing details of the fitted model. We may fit a binomial logistic regression model to our data with `myopic` as the response with the following code:

```
> fit1 <- glm(myopic ~ ., data = Myopia, family = binomial)
> summary(fit1)

Call:
glm(formula = myopic ~ ., family = binomial, data = Myopia)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.0202  -0.5760  -0.4252  -0.2894   2.5828

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  -2.670761   0.409491  -6.522 6.93e-11 ***
gendermale   -0.365467   0.260908  -1.401 0.161288
sportHR      -0.042545   0.018199  -2.338 0.019397 *
readHR        0.095053   0.038217   2.487 0.012875 *
compHR        0.030289   0.037993   0.797 0.425326
studyHR      -0.064828   0.064246  -1.009 0.312949
TVHR          0.008328   0.022853   0.364 0.715548
mumMyopicYes  0.868446   0.261778   3.317 0.000908 ***
dadMyopicYes  0.988224   0.262875   3.759 0.000170 ***
---
```

```
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 480.08  on 617  degrees of freedom
Residual deviance: 439.60  on 609  degrees of freedom
AIC: 457.6

Number of Fisher Scoring iterations: 5
```

At the bottom of the output, we are told the dispersion parameter σ^2 for the binomial family is taken to be 1, as we expect. We have seen that the residual sum of squares in the normal linear model can be generalised to GLMs by the *deviance*, and the residuals can be generalized in several ways, one of them being the *deviance residuals.* The null/residual deviance is the deviance of an intercept only model/the fitted model compared to a saturated model.

To test the null hypothesis that the intercept only model is correct, against the alternative of the model we have fitted, we can perform a likelihood ratio test with the following:

```
> pchisq(480.08-439.60, df = 617-609, lower.tail = FALSE)
[1] 2.607117e-06
```

This can also be done using the `anova` function. We have already seen it in action for `lm` objects. With `glm` objects it performs the analysis of deviance.

```
> fit2 <- glm(myopic ~ 1, data = Myopia, family = binomial)
> anova(fit2, fit1, test = "LRT")
Analysis of Deviance Table

Model 1: myopic ~ 1
Model 2: myopic ~ gender + sportHR + readHR + compHR + studyHR + TVHR +
    mumMyopic + dadMyopic
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1       617     480.08
2       609     439.60  8   40.478 2.61e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

In contrast to the `lm` case, here we need to specify that we want to perform a likelihood ratio test (`test = "LRT"`) in order for the final column containing the p-value to be given.

4.4.2 Binomial responses

Download the `Smoking` data with

```
Smoking <- read.csv(paste0(path, "Smoking.csv"))
```

In 1972-1974 a survey was taken; twenty years later a follow-up study was conducted, and it was determined if each interviewee was still alive. Among the information obtained originally was whether a person was a smoker or not and their age; the latter was divided into seven categories represented by the `Age.group` variable which gives the mid-points of these categories. Thus, the interviewees were divided into 14 groups, the information of each corresponding to a data point. Let us begin by plotting the data.

```
Smoking$total <- Smoking$Survived + Smoking$Died
Smoking$propDied <- Smoking$Died / Smoking$total

par(mfrow = c(1, 2))
plot(propDied ~ Age.group, Smoking, subset = (Smoker == "Yes"))
points(propDied ~ Age.group, Smoking, subset = (Smoker == "No"), pch = 4)
plot(qlogis(propDied) ~ Age.group, Smoking, subset = (Smoker == "Yes"))
points(qlogis(propDied) ~ Age.group, Smoking, subset = (Smoker == "No"), pch = 4)
```

The `qlogis` function returns the quantile of the "logistic distribution" (defined through the latent variable interpretation in Section 4.3.2), which is given by

$$g(\mu) = \log \frac{\mu}{1-\mu}.$$

This is the canonical link for binomial regression and is also known as the logit or the log odds function. If a logistic regression model taking as response the proportion of interviewees who died and as explanatory variables the age group and smoking status is correct, we should see that the points fall close to two parallel lines. If there was an interaction between smoking status and age, we would expect two non-parallel lines. Why, and which one of theses model is fitted below?

```
> fit1 <- glm(propDied ~ Age.group + Smoker, data = Smoking,
              family = binomial, weights = total)
> summary(fit1)

Call:
glm(formula = propDied ~ Age.group + Smoker, family = binomial,
    data = Smoking, weights = total)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3947  -0.3841   0.6008   1.3799   3.4824

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -7.687751   0.447646 -17.174   <2e-16 ***
Age.group    0.124957   0.007274  17.178   <2e-16 ***
SmokerYes    0.266053   0.168702   1.577    0.115
```

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 641.496  on 13  degrees of freedom
Residual deviance:  32.572  on 11  degrees of freedom
AIC: 85.568

Number of Fisher Scoring iterations: 5
```

Note we now must specify the `weights` for the GLM. Why should we use `total` as the weights?

From the null deviance reported by the summary output, we can easily reject the intercept only model against the model we have fitted. The residual deviance suggests that there is some room to improve the model (or some overdispersion). By examining the GLM diagnostic plots using the code below, we see a clear quadratic trend in the residuals vs. fitted plot. So perhaps we can add a quadratic age term.

```
par(mfrow = c(2,2)); plot(fit1)

> fit2 <- glm(propDied ~ Age.group + I(Age.group^2) + Smoker, data = Smoking,
            family = binomial, weights = total)
> summary(fit2)

Call:
glm(formula = propDied ~ Age.group + I(Age.group^2) + Smoker,
    family = binomial, data = Smoking, weights = total)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.3185  -0.5402   0.1183   0.5725   1.9612

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)    -2.8658262  1.0224461  -2.803  0.00506 **
Age.group      -0.0772521  0.0426067  -1.813  0.06981 .
I(Age.group^2)  0.0019587  0.0004256   4.602 4.18e-06 ***
SmokerYes       0.4306573  0.1764364   2.441  0.01465 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 641.496  on 13  degrees of freedom
```

```
Residual deviance:  11.625  on 10  degrees of freedom
AIC: 66.621

Number of Fisher Scoring iterations: 4
```

We can also treat the age groups as a categorical variable.

```
> fit3 <- glm(propDied ~ factor(Age.group) + Smoker, data = Smoking,
+             family = binomial, weights = total)
> summary(fit3)

Call:
glm(formula = propDied ~ factor(Age.group) + Smoker, family = binomial,
    data = Smoking, weights = total)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.72545  -0.22836   0.00005   0.19146   0.68162

Coefficients:
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)              -3.8601     0.5939  -6.500 8.05e-11 ***
factor(Age.group)29.5     0.1201     0.6865   0.175 0.861178
factor(Age.group)39.5     1.3411     0.6286   2.134 0.032874 *
factor(Age.group)49.5     2.1134     0.6121   3.453 0.000555 ***
factor(Age.group)59.5     3.1808     0.6006   5.296 1.18e-07 ***
factor(Age.group)69.5     5.0880     0.6195   8.213  < 2e-16 ***
factor(Age.group)80      27.8073 11293.1430   0.002 0.998035
SmokerYes                 0.4274     0.1770   2.414 0.015762 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 641.4963  on 13  degrees of freedom
Residual deviance:   2.3809  on  6  degrees of freedom
AIC: 65.377

Number of Fisher Scoring iterations: 20
```

If you have to choose a model fit, which one would you pick?

4.4.3 Exercises

Exercise 4.5. Using the Myopia dataset, test the null hypothesis that the hours spent watching TV and hours spent playing computer games collectively have no effect on myopia.

Exercise 4.6. Test the hypothesis that the coefficients for `mumMyopic` and `dadMyopic` are the same in the model

```
fit1 <- glm(myopic ~ ., data = Myopia, family = binomial)
```

Hint: Create a new variable by

```
Myopia$parentMyopic <- (Myopia$dadMyopic == "Yes") + (Myopia$mumMyopic == "Yes")
```

Exercise 4.7. Fit a model in which the effects of each parent having myopia are equal, but there may be an interaction, i.e. the effect of both parents having myopia is not necessarily the sum of the effect of each parent having myopia.

Exercise 4.8. Using the `Smoking` dataset and your preferred model, estimate the multiplicative change in odds of dying for an increase in age of one year.

4.5 Poisson regression

4.5.1 Models for count data

Poisson regression is used to model count data: $Y_i \in \{0, 1, 2, \dots\}$, $i = 1, \dots, n$. It is common to model counts by a Poisson distribution, $Y_i \sim \text{Poisson}(\mu_i)$. One rationale for this is the following *law of small numbers.* Consider a triangular array $\{\mu_{n,j} > 0 \mid 1 \le j \le n\}$ such that $\sum_{j=1}^{n} \mu_{n,j} = \mu$. Then under the assumption that $\max_j \mu_{n,j} \to 0$ as $n \to \infty$, we have

$$\sum_{j=1}^{n} \text{Bernoulli}(\mu_{n,j}) \to \text{Poisson}(\mu) \text{ as } n \to \infty.$$

In words, if Y is the total count of many small probability events, then Y approximately follows a Poisson distribution.

The Poisson distribution $Y \sim \text{Poisson}(\mu)$ has the mean-variance relation (Example 3.8)

$$\mathsf{Var}(Y) = \mathsf{E}(Y) = \mu.$$

In practice, the data are sometimes overdispersed compared to the theoretical relationship above due to clustering or other reasons (Section 3.4.1).

4.5.2 *Variance stabilizing transform

To deal with overdispersion, one can use the *variance stabilizing transform* that maps Y to $g(Y)$. By the delta method, $\mathsf{Var}(g(Y)) \approx \{g'(\mu)\}^2 \,\mathsf{Var}(Y)$. Thus, if we take

$$g'(\mu) = \frac{1}{\sqrt{\mathsf{Var}(Y)}},$$

then $\mathsf{Var}(g(Y)) \approx 1$. For Poisson, this is $g(Y) = 2\sqrt{Y}$. We can then fit a linear model for $\mathsf{E}(2\sqrt{Y} \mid X)$ and use the linear model noise variance to probe overdispersion. The drawback of this approach is that $\sqrt{Y}$ might not be the scale we would like to investigate. We can also use a dispersion parameter σ^2 in the (quasi-)Poisson GLM to model overdispersion, which will be discussed in more detail next.[7]

4.5.3 Poisson regression

Recall that the probability mass function of $\text{Poisson}(\mu_i)$ is given by

$$f(y_i; \mu_i) = e^{-\mu_i} \frac{\mu^{y_i}}{y_i!} = e^{y_i \log \mu_i - \mu_i} \frac{1}{y_i!}, \; y_i = 0, 1, \dots .$$

With the expersion parameter σ^2 included, the probability mass function bcomes

$$f(y_i; \mu_i, \sigma^2) = e^{\frac{1}{\sigma^2}\{y_i \log \mu_i - \mu_i\}} f_0(y_i; \sigma^2).$$

So the natural parameter is $\theta_i = \log(\mu_i)$ and the cumulant function is $K(\theta) = e^{\theta}$.

In Poisson regression, the most common choice of the link function is the canonical log link $g(\mu) = \theta(\mu) = \log(\mu)$, so the model is

$$\log \mu_i = X_i^T \beta.$$

This is often referred to as the *log-linear model.* This model is straightforward to interpret, as

$$\mu_i = e^{X_i^T \beta} = \prod_{j=1}^{p} (e^{\beta_j})^{X_{ij}},$$

So e^{β_j} represents a multiplicative change to the predicted mean value per nuit change of the jth regressor.

Other common link functions for Poisson regression including the identity link and the square root link.[8] Notice that the square root link assumes that $\sqrt{\mathsf{E}(Y_i \mid X_i)} = X_i^T \beta$, which is different from fitting a linear model after the square root variance stabilizing transform which assumes that $\mathsf{E}(\sqrt{Y_i} \mid X_i) = X_i^T \beta$.

The deviance in Poisson regression ($\sigma^2 = 1$) is given by (Exercise 3.17)

$$D(Y_i, \hat{\mu}_i) = 2 \left\{ Y_i \log \frac{Y_i}{\hat{\mu}_i} - Y_i + \hat{\mu}_i \right\}.$$

If X includes intercept (a column 1) and the canonical log link is used, the score equation $X^T(Y - \hat{\mu}) = 0$ implies that

$$\sum_{i=1}^{n} \hat{\mu}_i = \sum_{i=1}^{n} Y_i.$$

Therefore, by letting $\delta_i = Y_i - \hat{\mu}_i$ and assuming $|\delta_i| \ll \hat{\mu}_i$, the total deviance for the Poisson regression is approximately given by

$$\begin{aligned} D^{(n)}(Y, \hat{\mu}) &= 2 \sum_{i=1}^{n} Y_i \log \frac{Y_i}{\hat{\mu}_i} \\ &= 2 \sum_{i=1}^{n} (\hat{\mu}_i + \delta_i) \log \left(1 + \frac{\delta_i}{\hat{\mu}_i} \right) \\ &\approx 2 \sum_{i=1}^{n} (\hat{\mu}_i + \delta_i) \left(\frac{\delta_i}{\hat{\mu}_i} - \frac{1}{2} \frac{\delta_i^2}{\hat{\mu}_i^2} \right) \\ &\approx 2 \sum_{i=1}^{n} \delta_i + \frac{1}{2} \frac{\delta_i^2}{\hat{\mu}_i} \\ &= \sum_{i=1}^{n} \frac{(Y_i - \hat{\mu}_i)^2}{\hat{\mu}_i}. \end{aligned}$$

The last expression is precisely the Pearson χ^2-statistic from *IB Statistics*:

$$\chi^2 = \sum \frac{(\text{observed} - \text{fitted})^2}{\text{fitted}}.$$

For Poisson regression, Pearson's residual is given by

$$R_{P,i} = \frac{Y_i - \hat{\mu}_i}{\sqrt{V(\hat{\mu}_i)}} = \frac{Y_i - \hat{\mu}_i}{\hat{\mu}_i},$$

so Pearson's χ^2-statistic is given by $\chi^2 = \sum_{i=1}^n R_{P,i}^2$ and converges to χ^2_{n-p} if the Poisson regression is correctly specified. Note that this convergence does not require n to converge to infinity; in fact, convergence to χ^2_{n-p} would be ill-defined if n increases and p is fixed. The crucial assumption is that $\min_i \mu_i \to \infty$ (which can be seen from the assumption that $\delta_i \ll \hat{\mu}_i$). This is the so-called *small dispersion asymptotics.*

4.5.4 Multinomial models and the Poisson trick

Poisson regression can also be used to analyze multinomial data. Suppose $(Y_1, \dots, Y_L) \sim \text{Multinomial}(n, \pi)$, where n is known but $\pi = (\pi_1, \dots, \pi_L)$ is unknown. The probability mass function is given by

$$f(y; \pi) = \frac{n!}{y_1! \cdots y_L!} \pi_1^{y_1} \cdots \pi_L^{y_L}, \text{ for } \sum_{i=1}^L Y_i = n.$$

This is not a minimal exponential family because of the constraint on Y, which also implies that $Y_1, \dots, Y_L$ are not independent. A potential solution is to set one level as the reference and obtain a $(L-1)$-parameter exponential family (Exercise 3.6). However, the symmetry in the parameters is broken.

A more elegant solution is the *Poisson trick*, which refers to the following probabilistic result. Suppose $Y_i \sim \text{Poisson}(\mu_i),\ i = 1, \dots, L$ independently. Let $Y_+ = \sum_{i=1}^L Y_i$, then

$$Y_+ \sim \text{Poisson}(\mu_+) \quad \text{and} \quad Y_1, \dots, Y_L \mid Y_+ \sim \text{Multinomial}(Y_+; \pi),$$

where $\pi_i = \mu_i / \mu_+,\ i = 1, \dots, L$ and $\mu_+ = \sum_{i=1}^L \mu_i$.[9]

Exercise 4.9. Verify the Poisson trick.

Consider the log-linear Poisson model

$$Y_i \sim \text{Poisson}(\mu_i) \text{ independently, and } \log \mu_i = \alpha + X_i^T \beta,$$

where the intercept α is distinguished from the rest of the coefficients. Then by the Poisson trick,

$$Y_+ = \sum_{i=1}^L Y_i \sim \text{Poisson}(\mu_+) \quad \text{and} \quad Y \mid Y_+ \sim \text{Multinomial}(Y_+, \pi),$$

where

$$\mu_+ = \sum_{i=1}^n \mu_i = e^\alpha \sum_{i=1}^n e^{X_i^T \beta}, \text{ and}$$

$$\pi_i = \frac{\mu_i}{\mu_+} = \frac{e^{X_i^T\beta}}{\sum_{i=1}^n e^{X_i^T\beta}},\ i = 1, \ldots, n. \tag{4.10}$$

Importantly, π does not depend on the intercept α in the Poisson model. In consequence, the likelihood function for the Poisson model factorizes as

$$\begin{aligned} L_P(\alpha, \beta) &= \prod_{i=1}^n f(Y_i; \mu_i) \\ &= f(Y_1, \ldots, Y_L \mid Y_+; \beta) f(Y_+; \alpha, \beta) \\ &= L_M(\beta) f(Y_+; \alpha, \beta), \end{aligned}$$

where $L_M(\beta)$ denotes the likelihood for the multinomial model (4.10). Alternatively, because given β, μ_+ is uniquely determined by α and vice versa, we can write this more concisely as

$$L_P(\mu_+, \beta) = L_M(\beta) f(Y_+; \mu_+).$$

The above likelihood factorization implies that we can fit the multinomial model (4.10) using the Poisson log-linear model with an intercept, and the likelihood inference for β in the two models will be equivalent. To see this, the MLE $\hat{\beta}$ for the Poisson likelihood $L_P(\mu_+, \beta)$ will also maximize the multinomial likelihood $L_M(\beta)$. The Fisher information matrix for the Poisson model is block-diagonal:

$$I^{(n)}(\mu_+, \beta) = \begin{pmatrix} I^{(n)}_{\mu_+\mu_+}(\mu_+) & 0^T \\ 0 & I^{(n)}_{\beta\beta}(\beta) \end{pmatrix}.$$

Deviance in the Poisson model is the same as deviance in the multinomial model, because $\hat{\mu}_+ = Y_+$.

4.6 Contingency tables

Next we apply the Poisson and multinomial models to analyze contingency tables that display empirical frequencies of random variables.

4.6.1 Two-way contingency tables

Example 4.10. The following contingency table was constructed from a interim release of a Phase-III trial for the Moderna COVID-19 vaccine in November, 2020.[10] The * cells were not reported, but they are presumably very large because the total number of participants was about 30,000. The press release claims that the vaccine efficacy is about $1-(5+0)/(79+11) = 94.5\%$ and the p-vlaue (for no efficacy) is less than 0.0001.

	Not a case	Non-severe case	Severe case
Placebo	*	79	11
Vaccine	*	5	0

There are two ways to think about the data in contingency tables:

- We observe counts $Y_{jk},\ j=1,\dots,J,\ k=1,\dots,K$. In the previous example, $J=2$ and $K=3$.
- The table is an aggregation of individual observations $(A_i,B_i),\ i=1,\dots,n$. In the previous example, A_i is the treatment received (placebo or vaccine), B_i is the outcome (not a case, non-severe case, or severe case), and $n \approx 30,000$. The observed counts are given by

$$Y_{jk} = \sum_{i=1}^{n} 1_{\{A_i=j,B_i=k\}},\ j=1,\dots,J,\ k=1,\dots,K.$$

A common question in two-way contingency tables is testing the null hypothesis that the rows and columns are independent, $H_0: A_i \perp\!\!\!\perp B_i$. In the vaccine trial example, this amounts to testing the hypothesis that the vaccine has no effect at all.

Suppose $(A_i,B_i),\ i=1,\dots,n$ are IID. Let $\pi_{jk} = \mathsf{P}(A_i=j,B_i=k),\ j=1,\dots,J,k=1,\dots,K$, so the counts follow a multinomial distribution $Y \sim \text{Multinomial}(n,\pi)$. The null hypothesis can be expressed in terms of π as

$$H_0:\ \pi_{jk} = \pi_j^A \pi_k^B,\ \text{for all } j,k,$$

where $\pi_j^A = \sum_{k=1}^K \pi_{jk}$ and $\pi_k^B = \sum_{j=1}^J \pi_{jk}$ are the marginal distributions of A and B. In the surrogate Poisson model, this can be expressed as

$$H_0:\ \mu_{jk} = \mu_+ \pi_j^A \pi_k^B \text{ for all } j,k,$$

which is equivalent to the log-linear model

$$H_0:\ \log \mu_{jk} = \alpha + \beta_j^A + \beta_k^B,\ \text{for all } j,k, \tag{4.11}$$

This is a submodel of the saturated model that places no restrictions on μ_{jk}:

$$H_1: \log \mu_{jk} = \alpha + \beta_j^A + \beta_k^B + \beta_{jk}^{AB}, \text{ for all } j, k. \tag{4.12}$$

Therefore, testing independence in contingency tables is equivalent to testing nested models in Poisson regression.

Notice that (4.11) and (4.12) are overparametrized. For identifiability, it is necessary to set some levels as the reference. For example, we may set $\beta_1^A = \beta_1^B = \beta_{1k}^{AB} = \beta_{j1}^{AB} = 0$ for all j, k.

Example 4.11. For a 2 × 2 table ($J = K = 2$), the null/independence log-linear model assumes

$$\log \mu = \begin{pmatrix} \log \mu_{11} \\ \log \mu_{12} \\ \log \mu_{21} \\ \log \mu_{22} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}}_{X_0} \begin{pmatrix} \alpha \\ \beta_2^A \\ \beta_2^B \end{pmatrix} = \begin{pmatrix} \alpha \\ \alpha + \beta_2^B \\ \alpha + \beta_2^A \\ \alpha + \beta_2^A + \beta_2^B \end{pmatrix},$$

and the saturated log-linear model assumes

$$\log \mu = \begin{pmatrix} \log \mu_{11} \\ \log \mu_{12} \\ \log \mu_{21} \\ \log \mu_{22} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}}_{X} \begin{pmatrix} \alpha \\ \beta_2^A \\ \beta_2^B \\ \beta_{22}^{AB} \end{pmatrix} = \begin{pmatrix} \alpha \\ \alpha + \beta_2^B \\ \alpha + \beta_2^A \\ \alpha + \beta_2^A + \beta_2^B + \beta_{22}^{AB} \end{pmatrix}.$$

The degrees of freedom of the sub/independence model is $1+(J-1)+(K-1) = J+K-1$ and the degrees of freedom of the saturated model is JK. By Wilks' theorem, under H_0 and as $n \to \infty$, the deviance between the two models or equivalently Pearson's χ^2-statistic converges in distribution to $\chi^2_{JK-(J+K-1)} = \chi^2_{(J-1)(K-1)}$. This provides an asymptotic test for the independence hypothesis.

4.6.2 Three-way contingency tables

The discussion above can be extended to three-way contingency tables, although there are more independence and conditional independene hypotheses that can be tested. Individually, the observations come as IID triplets $(A_i, B_i, C_i), i = 1, \ldots, n$, which can be aggregated by a three-way table:

$$Y_{jkl} = \sum_{i=1}^n 1_{\{A_i=j, B_i=k, C_i=l\}}, \ j = 1, \ldots, J, \ k = 1, \ldots, K, \ l = 1, \ldots, L.$$

There are several possible models for the joint probability mass $\pi_{jkl} = \mathsf{P}(A_i = j, B_i = k, C_i = l)$. The first model assumes

$$H_1: \pi_{jkl} = \pi_j^A \pi_k^B \pi_l^C \text{ for all } j, k, l,$$

where $\pi_j^A, \pi_k^B, \pi_l^C$ are the corresponding marginal probabilities (similar conventions are used below). This is equivalent to assuming

$$H_1: \ A_i \perp\!\!\!\perp B_i \perp\!\!\!\perp C_i.$$

The second model assumes

$$H_2: \ \pi_{jkl} = \pi_j^A \pi_{kl}^{BC} \text{ for all } j, k, l,$$

which amounts to the independence

$$H_2: A_i \perp\!\!\!\perp (B_i, C_i).$$

The third model assumes

$$H_3: \ \pi_{jkl} = \gamma \pi_{jk}^{AB} \pi_{kl}^{BC} \text{ for all } j, k, l,$$

where γ is a normalizing constant. It can be shown that this implies

$$\mathsf{P}(A_i = j, C_i = l \mid B_i = k) = \mathsf{P}(A_i = j \mid B_i = k)\, \mathsf{P}(C_i = l \mid B_i = k). \tag{4.13}$$

So this model amounts to the conditional independence

$$H_3: \ A_i \perp\!\!\!\perp C_i \mid B_i.$$

Exercise 4.12. Verify (4.13).

The fourth model assumes

$$H_4: \ \pi_{jkl} = \gamma \pi_{jk}^{AB} \pi_{kl}^{BC} \pi_{jl}^{AC}.$$

This model does not imply any independence or conditional independence, but it assumes that there is no three-way interaction in the joint distribution.

Finally, the fifth and saturated model assumes

$$H_5: \ \pi_{jkl} = \pi_{jkl}^{ABC},$$

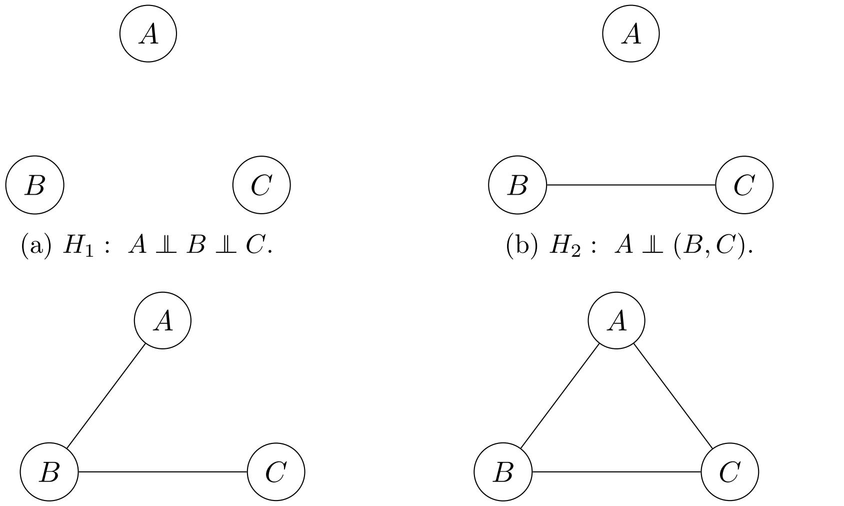
where π_{jkl}^{ABC} is completely unrestricted besides the constraints that the marginals need to sum up to 1. Of course, this model also makes no independence or conditional independence assumptions.

See Table 4.1 for a comparison of the models above.

The five models above for the three-way contingency table are nested and can be tested using the deviance or Pearson's χ^2 of the corresponding surrogate Poisson models. These Poisson log-linear models differ in whether certain two-way and three-way interaction terms are included.

`glm` formula	Poisson model	Multinomial model	Independence
`Y~A+B+C`	$\log \mu_{jkl} = \alpha + \beta_j^A + \beta_k^B + \beta_l^C$	$\pi_{jkl} = \pi_j^A \pi_k^B \pi_l^C$	$A \perp\!\!\!\perp B \perp\!\!\!\perp C$
`Y~A+B*C`	$\log \mu_{jkl} = \alpha + \beta_j^A + \beta_{kl}^{BC}$	$\pi_{jkl} = \pi_j^A \pi_{kl}^{BC}$	$A \perp\!\!\!\perp (B, C)$
`Y~A*B+B*C`	$\log \mu_{jkl} = \alpha + \beta_{jk}^{AB} + \beta_{kl}^{BC}$	$\pi_{jkl} \propto \pi_{jk}^{AB} \pi_{kl}^{BC}$	$A \perp\!\!\!\perp C \mid B$
`Y~A*B+B*C+C*A`	$\log \mu_{jkl} = \alpha + \beta_{jk}^{AB} + \beta_{kl}^{BC} + \beta_{jl}^{AC}$	$\pi_{jkl} \propto \pi_{jk}^{AB} \pi_{kl}^{BC} \pi_{jl}^{AC}$	No
`Y~A*B*C`	$\log \mu_{jkl} = \alpha + \beta_{jkl}$	$\pi_{jkl} = \pi_{jkl}^{ABC}$	No

Table 4.1: Different models for three-way contigency tables.



(a) $H_1: \ A \perp\!\!\!\perp B \perp\!\!\!\perp C$.

(b) $H_2: \ A \perp\!\!\!\perp (B, C)$.

(c) $H_3: \ A \perp\!\!\!\perp C \mid B$.

(d) H_4 and H_5: no independence or conditional independence.

Figure 4.3: Graphical models for three-way contingency tables.

4.6.3 *Graphical models

With more variables, it is more convenient to represent independence and conditional independence relationship using a graph.

Consider an undirected graph $(V, \mathcal{E})$ where $V = (V_1, \dots, V_p)$ is a discrete random vector and $E \subseteq \{V_1, \dots, V_p\}^2$ is the edge set. We say the distribution of V *factorizes according to this graph* if the probability mass function of V can be written as

$$\mathsf{P}(V = v) = \prod_{\substack{\mathcal{C} \subseteq \{V_1, \dots, V_p\} \\ (A_1, A_2) \in \mathcal{E} \text{ for all } A_1, A_2 \in \mathcal{C}}} \pi^{\mathcal{C}}(v_{\mathcal{C}}).$$

Such subset of vertices $\mathcal{C}$ is called a *clique* or *complete subgraph*. Thus, graphical factorization means that the distribution can be decomposed according to the cliques in the graph.

See Figure 4.3 for the graphical models corresponding to the five models for the three-way contingency table. There is a deep connection between graph theory and conditional independence: in an undirected graphical model, if the probability distribution factorizes according to the graph and a subset of variables B "blocks" all paths between two other non-overlapping subsets A and C, then $A \perp\!\!\!\perp C \mid B$.[11]

4.7 Practical 7: Poisson GLM

4.7.1 Poisson regression

Download the English Premiership data from 2014 with

```
path <- "http://www.statslab.cam.ac.uk/~rds37/teaching/statistical_modelling/"
football2014 <- read.csv(file.path(path, "football2014.csv"), stringsAsFactors = TRUE)
```

There are 20 teams in the Premier League and each team plays every other team twice, once at home and once away. Thus 380 matches are played in total. Our dataset here has 536 rows as the goals scored by the home and away teams are recorded separately, and the data were collected before the season ended.

```
> football2014[1:3, ]
  GoalsScored         By        Against HomeAway
1           2    Arsenal Crystal Palace     Home
2           2  Leicester        Everton     Home
3           1 Man United        Swansea     Home
> football2014[269:271, ]
    GoalsScored             By    Against HomeAway
269           1 Crystal Palace    Arsenal     Away
270           2        Everton  Leicester     Away
271           2        Swansea Man United     Away
```

The 1st and 269th rows say that the Arsenal (Home) vs. Crystal Palace (Away) game ended with a score 2:1. You can use the `relevel` function to treat your favourite team as the baseline in the models below, e.g.,

```
> football2014$By <- relevel(football2014$By, "Leicester")
> football2014$Against <- relevel(football2014$Against, "Leicester")

> summary(fit.football <- glm(GoalsScored ~ HomeAway + By + Against,
                      data = football2014, family = poisson))

Call:
glm(formula = GoalsScored ~ HomeAway + By + Against, family = poisson,
    data = football2014)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1477  -1.0860  -0.0816   0.5266   3.5319

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)            0.35052    0.24139   1.452 0.146479
```

```
HomeAwayHome          0.23510    0.07711   3.049 0.002298 **
ByAston Villa        -1.35321    0.31169  -4.342 1.41e-05 ***
ByBurnley            -0.71171    0.24557  -2.898 0.003753 **
ByChelsea             0.09854    0.19453   0.507 0.612476

[omitted output...]

---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 634.32  on 535  degrees of freedom
Residual deviance: 521.89  on 496  degrees of freedom
AIC: 1527

Number of Fisher Scoring iterations: 5
```

Figure 4.4 is generated by the next chunk of code and shows the estimated attack and denfese strengths of all the teams (relative to Leicester).

```
team <- names(coef(fit.football))[3:21]
team <- substr(team, 3, nchar(team))
team_strength <- data.frame(team = team,
                            attack_rating = coef(fit.football)[3:21],
                            defense_rating = - coef(fit.football)[22:40])
team_strength$team <- with(team_strength,
                           reorder(team, attack_rating + defense_rating))
library(ggplot2)
ggplot(reshape2::melt(team_strength), aes(x = team, fill = variable)) +
    geom_bar(aes(y = value), stat = "identity", position = "dodge") +
    theme_minimal(base_size = 25) + coord_flip()
```

*Advanced analysis

We will attempt to find for each team, the probability that at the end of the season they are in a particular position based on our fitted model. Download the remaining fixtures using

```
football2014_rem <- read.csv(file.path(path, "fixtures_remaining.csv"))
```

According to our model, the number of goals scored by each team in each match are independent Poisson random variables. Our estimates of the means of these Poisson random variables can be obtained using the `predict` function:

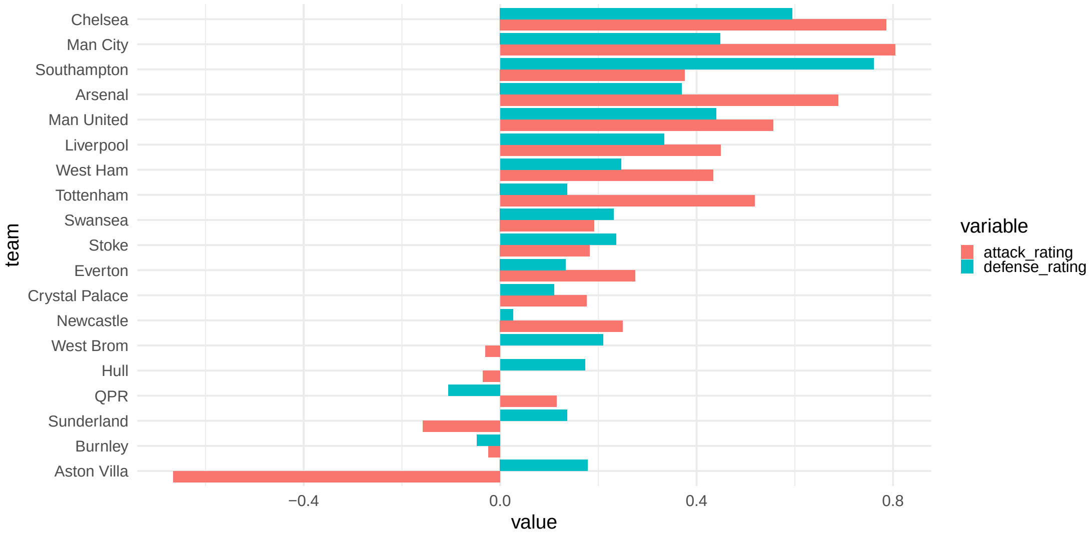


Figure 4.4: Estimated strengths of Premier League teams.

```
> pred <- predict(fit.football, newdata=football2014_rem, type="response")
> head(cbind(football2014_rem, GoalsPred = pred), 3)
           By   Against HomeAway GoalsPred
1   Leicester   Chelsea     Home 0.7209788
2         QPR Tottenham     Home 1.2784597
3 Aston Villa West Brom     Home 0.5449875
```

Using these means, we can simulate the scores of the remaining matches in the premiership. In each match, a team is awarded 3 points if it wins, 1 if it draws and 0 if it loses. The final positions of the teams at the end of the season are based on the total number of points accrued. The next function computes the standing of the teams given the match results.

```
standing <- function(data) {
    library(data.table)
    data <- data.table(data)
    out <- data[1:(nrow(data)/2), c("By", "Against", "GoalsScored")]
    names(out) <- c("HomeTeam", "AwayTeam", "GoalsHome")
    out$GoalsAway <- data[- (1:(nrow(data)/2)), ]$GoalsScored
    out[, HomeScore := 3 * (GoalsHome > GoalsAway) + (GoalsHome == GoalsAway)]
    out[, AwayScore := 3 * (GoalsAway > GoalsHome) + (GoalsHome == GoalsAway)]
    out1 <- out[, .(HomeScore = sum(HomeScore)), by = HomeTeam]
    out2 <- out[, .(AwayScore = sum(AwayScore)), by = AwayTeam]
    out <- merge(out1, out2, by.x = "HomeTeam", by.y = "AwayTeam")
    names(out)[1] <- "Team"
    out[, Score := HomeScore + AwayScore]
    out[order(- Score)]
}
```

The current standing is

```
> standing(football2014)
               Team HomeScore AwayScore Score
             <fctr>     <num>     <num> <num>
 1:         Chelsea        35        25    60
 2:        Man City        27        28    55
 3:         Arsenal        30        21    51
 4:      Man United        34        16    50
 5:       Liverpool        26        22    48
 6:     Southampton        24        22    46
 7:       Tottenham        24        20    44
 8:         Swansea        25        15    40
 9:           Stoke        20        19    39
10:        West Ham        24        15    39
11:       Newcastle        22        13    35
12: Crystal Palace        12        18    30
13:       West Brom        18        12    30
14:         Everton        16        12    28
15:            Hull        15        11    26
16:      Sunderland        13        12    25
17:     Aston Villa        11        11    22
18:         Burnley        14         8    22
19:             QPR        19         3    22
20:       Leicester        10         8    18
               Team HomeScore AwayScore Score
```

We next simulate the results of the rest of the matches based on the model predictions

```
one.sim <- function(sim) {
    football2014_rem$GoalsScored <- rpois(nrow(football2014_rem), pred)
    out <- merge(standing(football2014), standing(football2014_rem), by = "Team")
    out <- out[, .(Score = Score.x + Score.y), by = "Team"]
    out <- out[order(- Score)]
    out$Ranking <- 1:nrow(out)
    out
}
out <- do.call(rbind, lapply(1:1000, one.sim))

> out[, .(Score.med = median(Score),
+         Ranking.med = median(Ranking),
+         Champion.prob = mean(Ranking == 1),
+         Top4.prob = mean(Ranking <= 4),
+         Bottom3.prob = mean(Ranking >= 18)),
+     by = Team][order(- Score.med)]
+
               Team Score.med Ranking.med Champion.prob Top4.prob Bottom3.prob
             <char>     <num>       <num>         <num>     <num>        <num>
 1:         Chelsea        86           1         0.906     1.000        0.000
 2:        Man City        79           2         0.083     0.993        0.000
 3:         Arsenal        72           3         0.008     0.862        0.000
 4:      Man United        69           4         0.002     0.546        0.000
 5:     Southampton        67           5         0.000     0.249        0.000
 6:       Liverpool        66           6         0.000     0.270        0.000
 7:       Tottenham        63           7         0.001     0.079        0.000
 8:        West Ham        56           9         0.000     0.001        0.000
 9:         Swansea        54           9         0.000     0.000        0.000
10:           Stoke        53           9         0.000     0.000        0.000
11:       Newcastle        48          11         0.000     0.000        0.000
12:         Everton        43          13         0.000     0.000        0.004
13:       West Brom        42          14         0.000     0.000        0.008
14: Crystal Palace        42          13         0.000     0.000        0.006
15:            Hull        37          15         0.000     0.000        0.118
16:      Sunderland        36          16         0.000     0.000        0.255
17:             QPR        33          17         0.000     0.000        0.477
18:         Burnley        31          18         0.000     0.000        0.632
19:     Aston Villa        30          19         0.000     0.000        0.748
20:       Leicester        30          19         0.000     0.000        0.752
               Team Score.med Ranking.med Champion.prob Top4.prob Bottom3.prob
```

If you are a serious football fan, you will know that there is a more complicated tie-breaking rule that is not considered above. Anyway, you can find out how good these predictions are following this link: https://en.wikipedia.org/wiki/2014%E2%80%9315_Premier_League#League_table.

Note that the "training data" corresponded the first $\sim 70\%$ of the season. Thus, we do not expect the predictions to be very accurate but rather be guidelines. They succeed in this. Furthermore, it would be reasonable to think that the coefficients in the model are actually time-dependent (e.g., they depend on the motivation of the team, on injuries, on whether the team is at a critical stage at another competition and not at the league, etc.). Hence, there is considerable room to improve the model (and the predictions). Even with this, perhaps no "reasonable" model could have anticipated the glorious rise of "The Unbelievables" to the first position two years after!

4.7.2 Contingency tables

Fisman et al. conducted a study on dating behaviour using data from a Speed Dating experiment at Columbia University. Pairs of men and women students interacted for four minutes and then each filled out a form which recorded whether or not they wanted to receive the other's email address, and also various details about themselves including their chosen subject of study, what motivated them to sign up for the speed dating experiment, how often they go out and how often then go on dates. If both individuals wanted each others email addresses, it is considered a match. The original data are available at http://andrewgelman.com/2008/01/21/the_speeddating_1/. Download an edited version of the data with

```
> SD_data <- read.csv(file.path(path, "SD_match.csv"), stringsAsFactors=T)
> head(SD_data, 3)
  match subject_m goal_m     date_m go_out_m subject_f   goal_f       date_f
1     0      Econ    fun several/yr > 2/week       Law meet ppl almost never
2     0      Econ    fun    1/month  2/month       Law meet ppl almost never
3     1      Econ   date   > 2/week > 2/week       Law meet ppl almost never
  go_out_f
1 > 2/week
2 > 2/week
3 > 2/week
```

The first row records the meeting of a male Economics student and a female Law student, which did not result in a match (`match` is 0). The goals of the man and woman were to have fun and to mee people, respectively. We also have the frequencies with which the individuals go out and go out on dates. Let us first focus on the relationship between match and the subjects of the individuals. The `table` function converts the data into contingency table format (3-way in this case).

```
> (SD_table <- table(SD_data[, c("match", "subject_m", "subject_f")]))
, , subject_f = Arts+Humanities

     subject_m
match Arts+Humanities Econ Law Sciences
    0             357  683 126      518
    1              73  132  35       87

, , subject_f = Econ

     subject_m
match Arts+Humanities Econ Law Sciences
    0              71  108  22      132
    1              21   33   2       26

, , subject_f = Law

     subject_m
match Arts+Humanities Econ Law Sciences
    0              30   87  43       52
    1                5   19  24        6

, , subject_f = Sciences

     subject_m
match Arts+Humanities Econ Law Sciences
    0             175  210  41      272
    1              24   45   8       65
```

In order to fit models to the data, the `glm` function automatically applies `as.data.frame` to the contingency table `SD_table`. This produces a data frame where each each row gives the number of original observations (`Freq`) that fall into each of the possible categories given by each pair of subject types and match category.

```
> head(as.data.frame(SD_table), 3)
  match       subject_m       subject_f Freq
1     0 Arts+Humanities Arts+Humanities  357
2     1 Arts+Humanities Arts+Humanities   73
3     0            Econ Arts+Humanities  683
```

The next chunk of code tests the hypothesis that `subject_m`, `subject_f`, and `match` are three independent variables in a random speed date. This hypothesis is easily rejected (Deviance = 144.89 on 15 degrees of freedom).

```
fit1 <- glm(Freq ~ subject_m + subject_f + match,
            data = SD_table, family = poisson)
fit2 <- glm(Freq ~ (subject_m + subject_f + match)^2,
            data = SD_table, family = poisson)

> anova(fit1, fit2, test = "LR")
Analysis of Deviance Table

Model 1: Freq ~ subject_m + subject_f + match
Model 2: Freq ~ (subject_m + subject_f + match)^2
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1       504     4925.3
2       489     4780.4 15   144.89 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Let us look at the `summary` of `fit2`:

```
> summary(fit2)

Call:
glm(formula = Freq ~ (subject_m + subject_f + match)^2, family = poisson,
    data = SD_table)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.00130  -0.44272   0.00984   0.53082   1.65076

[Coefficients omitted...]

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 4587.508  on 31  degrees of freedom
Residual deviance:   21.707  on  9  degrees of freedom
AIC: 253.05

Number of Fisher Scoring iterations: 4
```

We see evidence of overdispersion (residual deviance = 21.707 on 9 degrees of freedom). We can handle this using the `quasipoisson` family:

```
fit3 <- glm(Freq ~ subject_m + subject_f + match,
            data = SD_table, family = quasipoisson)
fit4 <- glm(Freq ~ (subject_m + subject_f + match)^2,
            data = SD_table, family = quasipoisson)
> anova(fit3, fit4, test = "LR")
Analysis of Deviance Table

Model 1: Freq ~ subject_m + subject_f + match
Model 2: Freq ~ (subject_m + subject_f + match)^2
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1        24    166.137
2         9     21.707 15   144.43 7.308e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

The independence null hypothesis is still rejected.

4.7.3 Exercises

Exercise 4.13. Consider the football example and the model fit `fit.football`. How can you explain the estimated “home advantage” by this model to a football fan who does not know A-level mathematics? And the `(Intercept)` term?

Exercise 4.14. How can you modify the formula for `fit.football` and use analysis of deviance to test the hypothesis that the “home advantage” is the same for all the teams? What is your conclusion using the `football2014` dataset?

Exercise 4.15. In the analysis of the speed dating experiment above, find out how the p-value in the analysis of deviance table reported by `anova(fit3, fit4, test = "LRT")` is computed. [*Hint:* You can obtain the estimated dispersion parameter by `summary(fit4)$dispersion`.]

Exercise 4.16. Let's also consider the motivation for signing up for the speed dating experiment. Which model is selected by the stepwise AIC procedure below? Which undirected graph does it correspond to?

```
SD_table <- table(SD_data[, c("match", "subject_m", "subject_f", "goal_m", "goal_f")])
stepAIC(glm(Freq ~ .^3, data = SD_table, family = poisson))
```

Notes

[1] Efron and Hastie. *Computer-Age Statistical Inference.* Figure 7.1.

[2] This is known as the Pitman–Koopman–Darmois theorem and is in fact how exponential families were originally motivated.

[3] See McCullagh and Nelder (1989). *Generalized Linear Models*, Chapman & Hall, Appendix C.

[4] Taken from Agresti, A. (2015). *Foundations of linear and generalized linear models.* John Wiley & Sons, Figure 4.2.

[5] We may need $g(\mu)$ to be sufficiently smooth (e.g. twice differentiable) for the asymptotic theory to go through.

[6] This is not necessarily a causal effect. See Section 2.5.1.

[7] One can also use GLMs with other discrete distributions such as the negative binomial. However, this is beyond the scope of this course.

[8] See `?family` in `R`.

[9] This is in fact a special instance of a more general result for exponential families. See Brown, L. D. (1986). *Fundamentals of statistical exponential families: With applications in statistical decision theory.* Institute of Mathematical Statistics, Theorem 1.15.

[10] https://investors.modernatx.com/news-releases/news-release-details/modernas-covid-19-vaccine-candidate-meets-its-primary-efficacy.

[11] This is one direction of the Hammersley-Clifford theorem.

Chapter 5

Binary classification

5.1 Binary classification

In binary classification, our goal is to use some "features" $X \in \mathcal{X}$ to predict a binary "label" $Y \in \{0, 1\}$. Some examples include classifying an email as spam or not spam, a patient as having a specific disease or healthy, and a loan application as good or likely to default.

5.1.1 Loss function and Bayes classifier

Classification is a very common problem in practice and there are many powerful approaches. The statistical (inference-centered) approach is based on assuming a probability distribution on the joint distribution of (X, Y), so the goal is estimating the conditional probability function

$$\pi(X) = \mathsf{P}(Y = 1 \mid X).$$

The algorithmic (computing-centered) point view is that we would like to learn a decision rule that maps X to a label:

$$\delta : \mathcal{X} \to \{0, 1\}.$$

These approaches can be linked by a *loss function*

$$l : \{0, 1\} \times \{0, 1\} \to \mathbb{R}\,.$$

The best classifier then minimizes the *classification risk* which is defined as

$$R(\delta) = \mathsf{E}\{l(Y, \delta(X))\}.$$

The most widely used loss function is the zero-one loss:

$$l(Y, \delta(X)) = \begin{cases} 1, & \text{if } Y \neq \delta(X), \\ 0, & \text{if } Y = \delta(X). \end{cases}$$

In words, we incur a loss of 1 if the predicted label is wrong and no loss otherwise. The corresponding classification risk is then given by

$$\begin{aligned} R(\delta) &= \mathsf{P}(Y = 1, \delta(X) = 0) + \mathsf{P}(Y = 0, \delta(X) = 1) \\ &= \mathsf{E}\{\mathsf{P}(Y = 1, \delta(X) = 0 \mid X) + \mathsf{P}(Y = 0, \delta(X) = 1 \mid X)\} \\ &= \mathsf{E}\{\pi(X) 1_{\{\delta(X)=0\}} + (1 - \pi(X)) 1_{\{\delta(X)=1\}}\}. \end{aligned}$$

From here, we see that the optimal classifier δ^* satisfies

$$\delta^*(x) = \begin{cases} 1, & \text{if } \pi(x) > 1/2, \\ 0, & \text{if } \pi(x) < 1/2. \end{cases}$$

The classifier is free to make any prediction when $\pi(x) = 1/2$. This is often called the *Bayes classifier* in the literature because it simply compares the conditional probability of the class label, which can be computed using the Bayes formula by

$$\pi(X) = \mathsf{P}(Y = 1 \mid X) = \frac{\pi_1 f_1(X)}{\pi_0 f_0(X) + \pi_1 f_1(X)},$$

where $f_y(x)$ is the conditional density function of X given $Y = y$ and $\pi_y = \mathsf{P}(Y = y)$ is the marginal probability of class $y = 0, 1$.

Although the zero-one loss is most commonly used to compare classifiers, it is not convenient for training a classifier because it is not differentiable. Furthermore, instead of directly training the binary classifier δ, it is common to learn a "soft classifier"

$$\theta : \mathcal{X} \to \mathbb{R}$$

which can be used to obtain a "hard classifier" by

$$\delta(X) = 1_{\{\theta(X)>0\}}.$$

In practice, a *convex surrogate* loss

$$l(y, \theta(x)) = \phi((2y - 1)\theta(x)),\ \ \phi : \mathbb{R} \to [0, \infty) \text{ is convex.}$$

is often used to estimate a soft classifier by *empirical risk minimization* with a sample $(X_1, Y_1), \dots, (X_n, Y_n)$:

$$\hat{\theta}(X) = \arg\min_{\theta} \sum_{i=1}^{n} l(Y_i, \theta(X_i)).$$

Commonly used surrogate loss functions include the *logistic loss* $\phi(s) = \log_2(1 + e^{-s})$ and the *hinge loss* $\phi(s) = \max(1 - s, 0)$. They belong to a larger class of loss functions for which the Bayes classifier remains optimal; see *Mathematics of Machine Learning* for further details.

Exercise 5.1. Let

$$\theta(X) = \log \frac{\pi(X)}{1 - \pi(X)}$$

be the log-odds function. Show that finding the maximum likelihood estimator for θ is equivalent to empirical risk minimization with the logistic surrogate loss function.

Exercise 5.2. Plot the logistic and hinge loss functions and compare them to the zero-one loss function $\phi(s) = 1_{\{s \le 0\}}$. What do you observe?

5.1.2 Linear discriminant analysis

Suppose $X \in \mathbb{R}^p$ and consider the following model for the conditional distribution of X given Y:

$$X \mid Y = 0 \sim \mathrm{N}(\mu_0, \Sigma), \quad X \mid Y = 1 \sim \mathrm{N}(\mu_1, \Sigma). \tag{5.1}$$

This is often called "generative model" because it tells us how the features are generated given the class label.[1] By using the Bayes formula, we obtain

$$\begin{aligned}
\theta(X) =& \log \frac{\pi(X)}{1 - \pi(X)} \\
=& \log \frac{\mathsf{P}(Y = 1 \mid X)}{\mathsf{P}(Y = 0 \mid X)} \\
=& \log \frac{\pi_1 f_1(X)}{\pi_0 f_0(X)} \\
=& \log(\pi_1/\pi_0) - \frac{1}{2}\left\{(X - \mu_1)^T \Sigma^{-1}(X - \mu_1) - (X - \mu_0)^T \Sigma^{-1}(X - \mu_0)\right\} \\
=& \log(\pi_1/\pi_0) + \frac{1}{2}(\mu_0^T \Sigma^{-1}\mu_0 - \mu_1^T \Sigma^{-1}\mu_1) + \{\Sigma^{-1}(\mu_1 - \mu_0)\}^T X.
\end{aligned}$$

So the Bayes rule $\delta^*(X) = 1_{\{\pi(X) > 1/2\}} = 1_{\{\theta(X) > 0\}}$ corresponds to using a separating hyperplane determined by μ_0, μ_1, Σ that can be easily estimated using empirical data. This is known as *linear discriminant analysis.*

The fact that $\theta(X)$ is an affine function of X suggests that this model is a special case of logistic regression (binomial GLM with canonical link and an intercept term) which assumes

$$\log \frac{\mathsf{P}(Y = 1 \mid X)}{\mathsf{P}(Y = 0 \mid X)} = \alpha + \beta^T X.$$

Binomial GLM is more general because it does not model the distribution of X (the inference is conditional on X). This means that although linear discriminant analysis will be more efficient when the normal model (5.1) is correct, it will be less robust when the data are not normal within each class. We will further explore this in the next practical.

5.1.3 k-nearest neighbour

The k-nearest neighbour (k-NN) classifier simply uses the majority vote from nearby data points. Statistically speaking, k-NN corresponds to using a non-parametric estimator of the conditional probability function:

$$\hat{\pi}_{k\text{-nn}}(x) = \frac{1}{k} \sum_{i \in \mathcal{N}(x)} Y_i,$$

where $\mathcal{N}(x)$ contains the indicies of the K data points among $(X_1, Y_1), \ldots, (X_n, Y_n)$ that are closest to $x \in \mathcal{X}$. The k-NN classifier then plug this into the Bayes classifier

$$\hat{\delta}_{k\text{-nn}}(x) = 1_{\{\hat{\pi}_{k\text{-nn}}(x) > 0.5\}}.$$

See Figure 5.1 for a demonstration.[2]

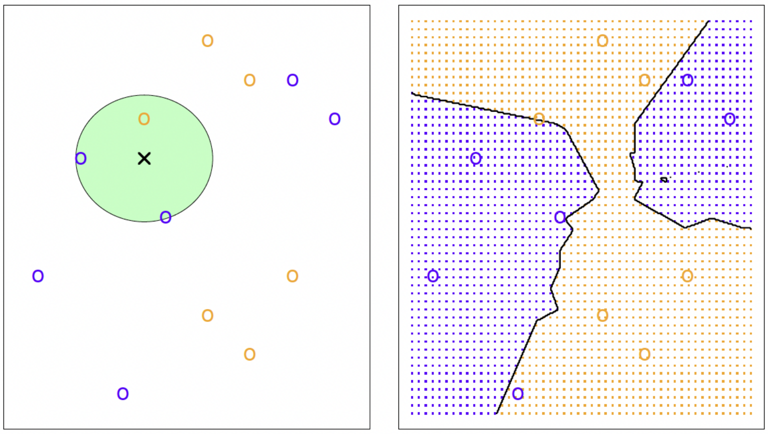


Figure 5.1: Demonstration of the k-NN classifier. Left: the 3-nearest neighbours of the black point. Right: k-NN classifier for this example.

5.2 Practical 8: Binary classification

5.2.1 Decision boundaries

I prompted an AI with the following text:

```
I'm teaching my students about logistic regression, linear
discriminant analysis, and k-nearest neighbour classifiers. Could you
design an R simulation example to demonstrate the relative merits and
limitations of these approaches? Please write the code using base R
functions if possible and keep the syntax simple.
```

The returned R code can be found at https://gemini.google.com/share/6fd29dcf119b. Executing the R code gives Figure 5.2.

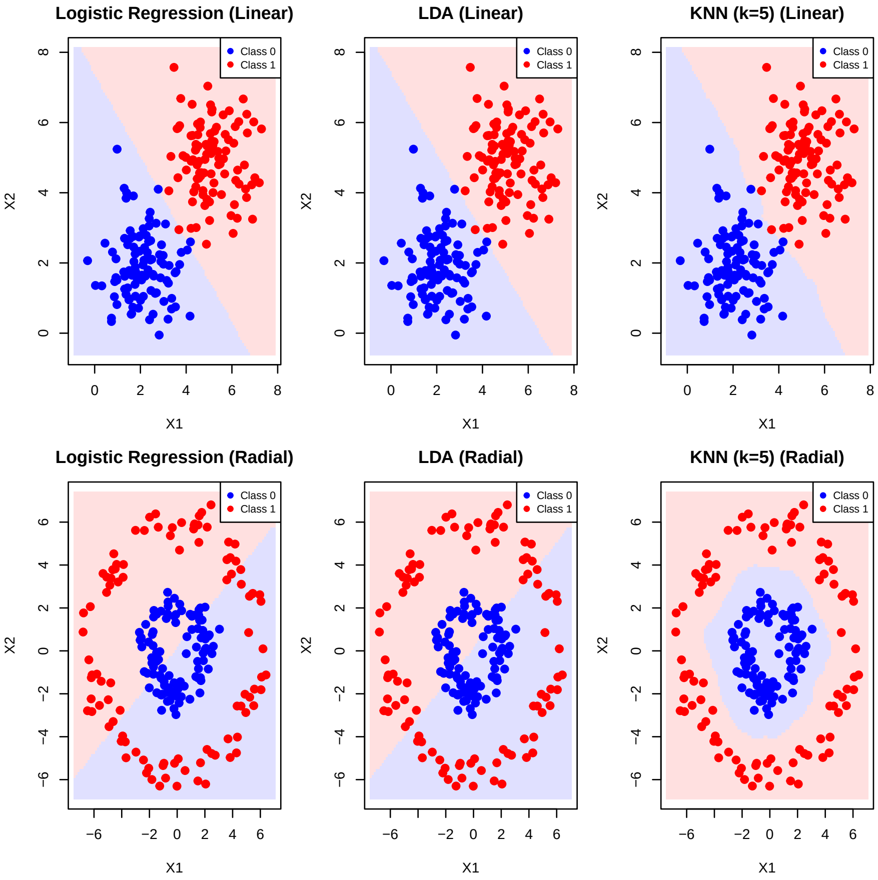


Figure 5.2: Demonstration of the decision boundaries of different binary classifiers.

I followed up with the following prompt.

```
I would like to demonstrate the efficiency-vs-robustness trade-off
between logistic regression and LDA. Please create a simulation
```

```
example in R and show two plots: in the first plot, show 200 data
points sampled from the LDA model (data points in each class follows a
Gaussian distribution with the same covariance matrix) and compare the
fitted logistic regression and LDA classifiers with the Bayes
classifier; in the second plot, show 200 data points sampled from the
logistic regression model (data are not normally distributed, but the
labels are generated according to the logistic regression model), then
compare the fitted logistic regression and LDA classifiers again with
the Bayes classifier.
```

I obtained the plots in Figure 5.3.

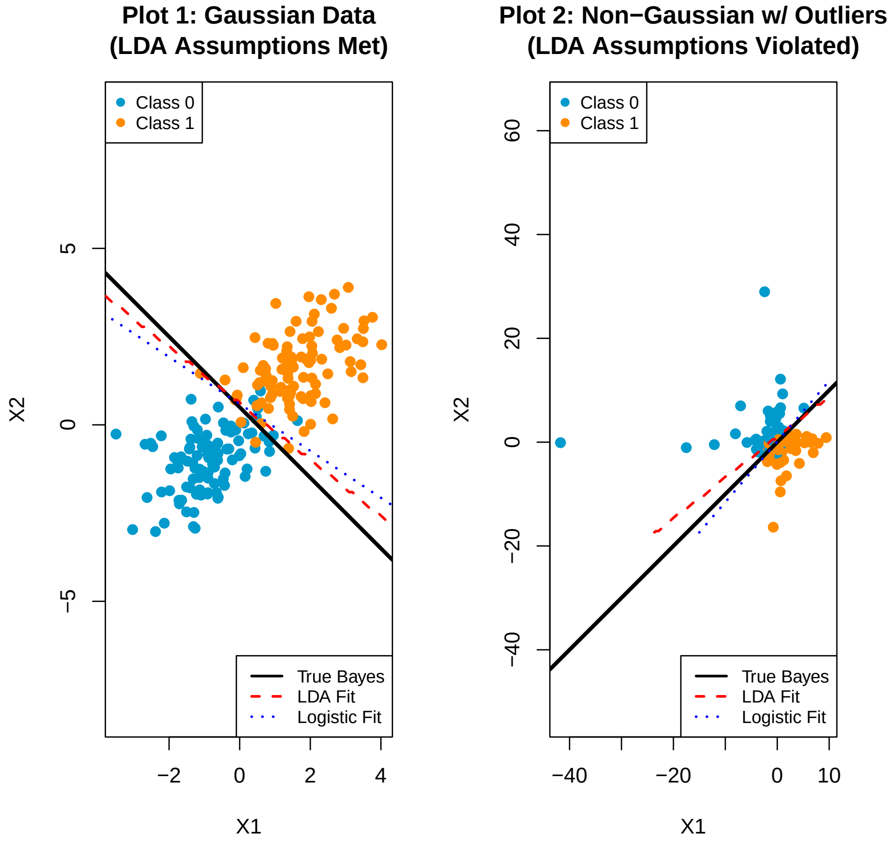


Figure 5.3: Demonstration of the efficiency-robustness trade-off for linear discriminant analysis versus logistic regression.

After some back-and-forth with the AI, I eventually learned about an old paper[3] that shows logistic regression is typically about $1/2$ to $2/3$ as efficient as linear discriminant analysis (meaning logistic regression requires 50% to 100% more data to achieve the same variance as the LDA) when the Gaussian assumption is met.

5.2.2 Cross-validation

Read the R code below which generates the plot in Figure 5.4. Then answer the questions below. You can of course use `R` to explore the dataset and any modifications to the code.

(i) Which response variable is being predicted? How are the data prepared before a classifier is trained?

(ii) Which classifier is used, and how is the tuning parameter selected?

(iii) Why are the two curves in Figure 5.4 U-shaped? Are you surprised that the “Test Set Error” curve is lower than the “Cross-Validation Error” curve (for the most part)?

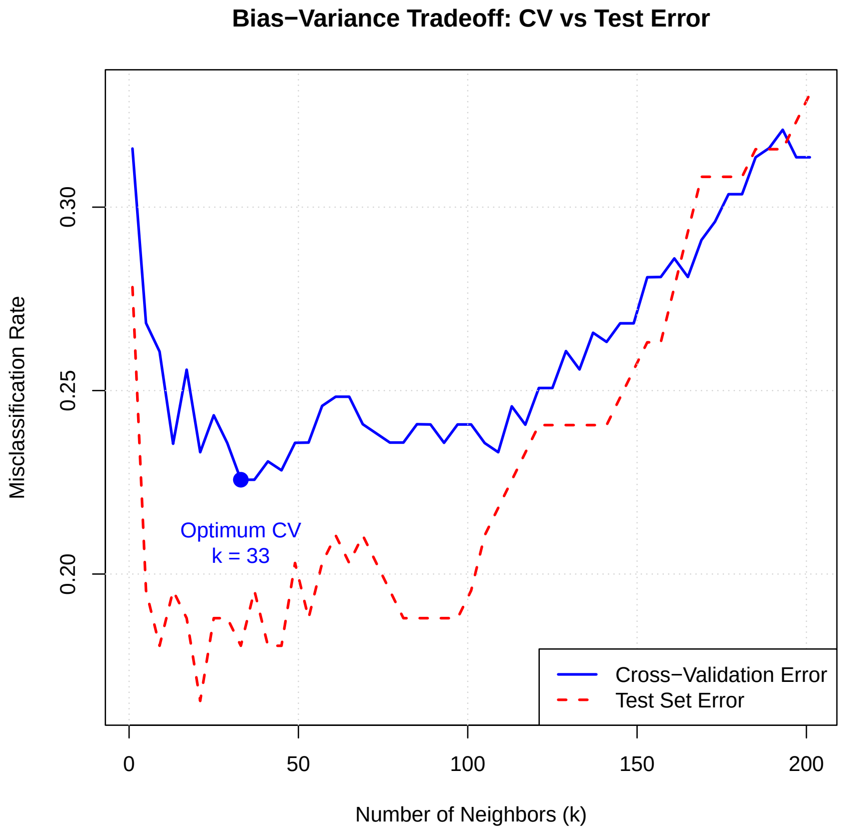


Figure 5.4: k-nearest neighbour classification with cross-validation.

```
library(class)
library(MASS)
```

```
set.seed(42)

# 1. PREPARE DATA
data(Pima.tr)
data(Pima.te)
full_data <- rbind(Pima.tr, Pima.te)

normalize <- function(x) {
  return ((x - min(x)) / (max(x) - min(x)))
}
data_norm <- as.data.frame(lapply(full_data[, 1:7], normalize))
data_norm$type <- full_data$type

train_idx <- sample(1:nrow(data_norm), 0.75 * nrow(data_norm))
train_data <- data_norm[train_idx, ]
test_data  <- data_norm[-train_idx, ]

train_X <- train_data[, 1:7]
train_y <- train_data$type
test_X  <- test_data[, 1:7]
test_y  <- test_data$type

# 2. LOOP FOR CV AND TEST ERROR
k_values <- seq(1, 201, by = 4)
cv_error_rates <- numeric(length(k_values))
test_error_rates <- numeric(length(k_values))

folds <- cut(seq(1, nrow(train_data)), breaks = 10, labels = FALSE)

for(i in 1:length(k_values)) {
  k <- k_values[i]

  fold_errors <- numeric(10)
  for(j in 1:10) {
    val_idx <- which(folds == j, arr.ind = TRUE)
    cv_val_X <- train_X[val_idx, ]
    cv_val_y <- train_y[val_idx]
    cv_train_X <- train_X[-val_idx, ]
    cv_train_y <- train_y[-val_idx]

    pred_cv <- knn(train = cv_train_X, test = cv_val_X, cl = cv_train_y, k = k)
    fold_errors[j] <- mean(pred_cv != cv_val_y)
  }
  cv_error_rates[i] <- mean(fold_errors)
  pred_test <- knn(train = train_X, test = test_X, cl = train_y, k = k)
  test_error_rates[i] <- mean(pred_test != test_y)
}

# 3. PLOT BOTH CURVES
y_min <- min(c(cv_error_rates, test_error_rates))
y_max <- max(c(cv_error_rates, test_error_rates))

plot(k_values, cv_error_rates, type = "l",
     col = "blue", lwd = 2,
     main = "Bias-Variance Tradeoff: CV vs Test Error",
     xlab = "Number of Neighbors (k)",
     ylab = "Misclassification Rate",
     ylim = c(y_min, y_max))

lines(k_values, test_error_rates, col = "red", lwd = 2, lty = 2)

grid()

legend("bottomright",
       legend = c("Cross-Validation Error", "Test Set Error"),
       col = c("blue", "red"),
       lty = c(1, 2), lwd = 2, bg="white")

optimal_idx <- which.min(cv_error_rates)
optimal_k <- k_values[optimal_idx]
min_cv_error <- cv_error_rates[optimal_idx]
test_error_at_opt <- test_error_rates[optimal_idx]

points(optimal_k, min_cv_error, pch=19, col="blue", cex=1.5)
text(optimal_k, min_cv_error - 0.01,
     paste("Optimum CV\nk =", optimal_k), pos=1, col="blue")
```

Notes

[1] In machine linear/AI, the term “generative model” is more broadly used to refer to a computer program from which we can draw a sample that exhibit desirable patterns.

[2] *Introduction to Statistical Learning with R, Figure 2.14.*

[3] Efron, B. (1975). The Efficiency of Logistic Regression Compared to Normal Discriminant Analysis. Journal of the American Statistical Association, 70(352), 892–898. https://doi.org/10.2307/2285453

Chapter 6

Look forward

Some more sophisticated statistical models and where you can find them.

(i) Mixed effect models: Assume some elements of (high-dimensional) β are random. (Part III, *Statistical Learning in Practice.*)

(ii) Generalized additive models and kernel machines: Replace $X_i^T\beta$ by some basis function expansion. (Part III, *Modern Statistical Methods.*)

(iii) Regularization for sparsity: lasso and beyond. (Part III, *Modern Statistical Methods.*)

(iv) Trees, random forests, and boosting: Replace $X_i^T\beta$ by a (complicated) step function. (Part II, *Mathematics of Machine Learning*; Part III, *Statistical Learning in Practice.*)

(v) Neural networks: Replace $X_i^T\beta$ by a composition of simple nonlinear functions (essentially GLMs). (Part II, *Mathematics of Machine Learning*; Part III, *Statistical Learning in Practice.*)

(vi) Graphical models. (Part III, *Bayesian Statistics*, *Causal Inference.*)

(vii) Distinguishing correlation from causation, more on instrumental variables. (Part III, *Causal Inference.*)