*3*

The Machine that Goes “Ping”: Machine Learning and Pattern Recognition

“Ping”的机器：机器学习和模式识别

We started our discussions about data science and analytics by stating that the use of data as evidence is nothing new. Furthermore, we saw in Chapter 1 how data science is a portmanteau for a number of overlapping tasks, taking tools from empirical sciences, mathematics, business intelligence, pattern recognition and machine learning. In this chapter we will focus our attention on the latter two in order to provide context to ideas explored so far and frame the algorithms that we will discuss in the following chapters.

我们通过陈述使用数据作为证据并不是什么新鲜事来开始讨论数据科学和分析。另外，我们在第1章中看到，数据科学从经验科学，数学，商业智能，模式识别和机器学习中获取工具，如何成为许多重叠任务的重要组成部分。在本章中，我们将把注意力集中在后两者上，以便为迄今为止探索的思想提供背景，并构建我们将在后面章节中讨论的算法。

*3*.*1* Recognising Patterns

认知模式

An integral part of being human is our ability to identify structure and order in the stimuli we receive, aided by information we have acquired previously. From that point of view we are firmly in the realm of cognitive psychology, where language usage, memory, creativity and thinking are the main interests1. Needless to say, the innate ability to recognise patterns is not unique to humans, and it is general among animals. Do not worry, we are not about to enter into a discussion of perceptual processes or theories of object recognition. However, it may be useful and illustrative to briefly consider the psychological theory of feature analysis.

作为人类的一个组成部分是我们能够识别刺激中的结构和秩序，这得益于我们之前获得的信息。 从这个角度来看，人类在认知心理学的领域有很高的造诣，其中对语言的使用，记忆力，创造力和思维是人类的主要的兴趣。 毋庸置疑，识别模式的先天能力并非人类独有，在动物中也是如此。 别担心，我们不打算讨论感知过程或物体识别理论。 然而，简要地考虑特征分析的心理学理论可能是有用的和具有说服力的。

1 Eysenck, M. and M. Keane (2000). Cognitive Psychology: A Student’s Handbook. Psychology Press

1 Eysenck, M. and M. Keane (2000). 认知心理学：学生手册。 心理学出版社

In feature analysis, we are said to recognise an object by considering the constituent parts, or features, of the object. We then assemble them together to determine what the object is. For example, we know that a cat is a small fury animal with triangular ears, long whiskers and playful claws. When we see a cat we recognise it for what it is because it satisfies these (admittedly simplified) rules.

在特征分析中，我们被认为通过考虑对象的组成部分或特征来识别对象。然后我们将它们组合在一起以确定对象是什么。例如，我们知道猫是一种小型的毛茸茸的动物，有三角形的耳朵，长长的胡须和俏皮的爪子。当我们看到一只猫时，我们辨别出它是猫，是因为它满足了这些（公认的简化）规则。

If we know what a cat looks like, we can recognise other cats.

如果我们知道猫的样子，我们可以识别其他猫。

This may seem a far-fetched example, were it not for the fact that actual research work on image recognition has been demonstrated using cat faces (and human bodies)2. The field of pattern recognition is thus interested in the systematic detection of regularities in a dataset, based on the use of algorithms. We can then use these patterns to take actions such as classifying objects like cats. This does sound familiar, right?

是不是因为使用猫脸（和人体）已经证明了关于图像识别的实际研究工作，这个例子可能看起来有点牵强。因此，基于对算法的使用，模式识别领域对数据集中的规则性的系统检测感兴趣。 然后我们可以使用这些模式来执行对猫等对象进行分类的操作。这听起来很熟悉吧？

2 Le, Q. V., R. Monga, M. Devin,G. Corrado, K. Chen, M. Ranzato, J. Dean, and A. Y. Ng (2011). Building high-level features using large scale unsupervised learning. CoRR abs/*1112*.*6209*

2 Le, Q. V., R. Monga, M. Devin,G. Corrado, K. Chen, M. Ranzato, J. Dean, and A. Y. Ng (2011). 使用大规模无监督学习构建高级功能。 CoRR abs / 1112.6209

The dataset in question may not necessarily be constrained to the features of a cat, or the parts of a body. As a matter of fact, exploiting patterns has been the bread and butter of sciences such mathematics, physics or chemistry. As a brief example, think of the efforts of Tycho Brahe and Johannes Kepler: One methodically recording the positions of celestial bodies, and the other unraveling the mysteries behind these measurements and summarising them in what we now know as Kepler’s Laws.

所讨论的数据集可能不一定受限于猫的特征或身体的部分。 事实上，挖掘模式一直是科学，如数学，物理或化学的基础。举一个Tycho Brahe（第谷·布拉赫）和Johannes Kepler3（约翰尼斯·开普勒）的例子：布拉赫有条不紊地记录天体的位置，开普勒解开这些测量背后的奥秘并总结在众所周知的开普勒定律中。

3 Gilder, J. and A. Gilder (2005). Heavenly Intrigue: Johannes Kepler, Tycho Brahe, and the Murder Behind One of History’s Greatest Scientific Discoveries. Knopf Doubleday Publishing Group

3 Gilder, J. and A. Gilder (2005) 天堂阴谋：约翰尼斯·开普勒，第谷·布拉赫，以及历史上最伟大的科学发现之一的谋杀案。 Knopf Doubleday出版集团

Recognising patterns is indeed a useful thing to do, and as you may imagine, it is helpful in more than one domain of applications. As a field, pattern recognition started up as part of science and engineering, resulting in a long list of applications to very practical problems.

正如您想象的那样，识别模式确实是一件很有用的事情，它在诸多的应用领域发挥重要的作用。模式识别已经是一个学术领域了， 它是作为科学和工程的一部分逐步发展起来的，产生了一系列非常实际的应用问题。

Nonetheless, engineering was not alone in developing techniques: Other areas such as computer science have also developed capabilities to exploit regularities seen in data. It is therefore an interesting exercise to stop looking for a moment at individual spheres of knowledge and gaze at where they may end up converging. We have talked here about psychology, physics, mathematics, engineering and computer science.

尽管如此，在开发技术方面工程学并不是唯一的：计算机科学等其他领域也开发了在数据中挖掘规律的能力。因此，停止在各个独立的知识领域继续寻找，重点关注他们最终会聚的地方。我们已经讨论了心理学，物理学，数学，工程学和计算机科学。

Pattern recognition is useful in many areas such as science, engineering, computer science, etc.

模式识别在许多领域都很有用，如科学，工程，计算机科学等。

The advances of these domains of knowledge, together with the relentless curiosity of the human mind has posed questions with the aim to understand a variety of areas. In particular, self-reflection has made us turn the attention to ourselves, and has made of the brain a hot area of study. It is a topic that still has many mysteries to reveal.

这些知识领域的进步，以及对人类思维的持续的好奇，提出了旨在了解各个领域的一系列问题。特别是，自我反思使我们把注意力转向自己，并使大脑成为研究的热门领域。 这个话题仍然有很多未解之谜。

Self-reflection has made us turn attention to ourselves.

自我反省使我们把注意力转向自己。

An important aim in the efforts to understand ourselves is the explanation of how the brain works and how it is able to be the centre of complex activity. This activity manifests itself as creativity, cognition, learning or intelligence, for example.

努力了解自己的一个重要目标是解释大脑如何工作以及它如何成为复杂活动的中心。 例如，这项活动表现为创造力，认知，学习或智力。

“How does the brain work’?”A very important and difficult question.

大脑是如何工作的？这是一个非常重要且困难的问题。

Is it possible to understand this marvellous organ? And if so, can we replicate its functions? Enter the realm of Artificial Intelligence!

有可能理解这个奇妙的器官吗？ 如果可能，我们可以复制它的功能吗？ 进入人工智能领域！

*3*.*2* Artificial Intelligence and Machine Learning

3.2人工智能和机器学习

Artificial intelligence is a field that carries a number of connotations: From helpful companion androids through to sentient killer robots, and even the singularity. The key is in the allure of that second word in the compound noun: Intelligence. What is intelligence and how can we quantify it? The contempt is due to the first word: Artificial. Is it possible to recreate intelligence with the aid of a machine in such a way that the behaviour is similar to that of a person demonstrating intelligence?

人工智能是一个具有多种内涵的领域：从有用的伴侣机器人到有感知的杀手机器人，甚至是奇点。 关键在于复合名词中第二个词的吸引力：智力。 什么是智力？我们如何量化它？ 存在争议的是第一个词：人工。是否有可能借助机器重建智能，使其行为类似于人的智能行为？

The singularity is a hypothetical event where machines are capable of recursive self-improvement leading to an intelligence explosion.

奇点是一个假设的事件，机器能够持续自我改善，导致智能爆炸。

This is not a new aim: The idea of recreating a human-like being has been the inspiration of stories such as that of the Golem, Pinocchio or Frankenstein’s monster. It is therefore hardly surprising that when the genius of Alan Turing4 formulated the concept of a Universal Machine in 1936, the tantalising possibility of an intelligent machine was raised.

这并不是一个新的目标：重建像人类一样的物种的想法一直是诸如Golem，Pinocchio或Frankenstein的怪物故事的灵感来源。 因此，当天才艾伦·图灵在1936年提出通用机器的概念时，一种智能机器的诱人可能性，这很难令人吃惊。

4 Turing, A. M. (1936). On computable numbers, with an application to the Entsheidungsproblem. Proceedings of the London Mathematical Society *42*(2), 230–265

4 Turing, A. M. (1936). 将可计算数字应用于Entsheidungs问题。 伦敦数学学会会刊42（2），230-265

In the so-called Turing test, Alan Turing considered playing an “imitation game”5 where a player would have to decide which of two interlocutors is a human being and which Mind is a machine, based only on their written responses to the player’s questions.

在所谓的图灵测试中，艾伦·图灵考虑玩“模仿游戏”，玩家必须根据他们对玩家问题的书面回答，决定两个对话者中哪一个是人，哪个是机器。

5 Turing, A. M. (1950). Computing machinery and intelligence. Mind *59*, 433–460

5 Turing, A. M. (1950). 计算机器和智能。Mind *59*, 433–460

If a machine could not be distinguished from the person, then the machine could be said to be “thinking”. Indeed, if there is no way of telling what other human beings are thinking except by a process of comparison with one’s own thinking, then there is no reason to regard machines any differently.

如果无法把机器和人区分开，则机器可以说“思考”。事实上，如果除了通过与自己的思想进行比较的过程之外，没有办法说出其他人在想什么，那么没有理由将机器视为不同。

If we cannot tell what other humans are thinking, there is no reason to regard machines any different in that respect.

如果我们无法分辨其他人的想法，就没有理由认为机器在这方面有任何不同。

As a field, the aim of artificial intelligence is to make machines carry out tasks that are associated with the intellectual processing competence of humans. It is a Herculean labour, and one that not only involves advances in computer science, but also in neuroscience, psychology and even philosophy.

作为一个研究领域，人工智能的目标是使机器执行与人类智力处理能力相关联的任务。这是一项艰巨的劳动，不仅涉及计算机科学的进步，还涉及神经科学，心理学甚至哲学。

Artificial intelligence aims to make machines carry out tasks associated with intellect.

人工智能旨在使机器执行与智力相关的任务。

Among other tasks, pattern recognition, as discussed in the previous section, is an integral part of the various functions that an artificial intelligent agent would have to accomplish. The ability to recognise regularities would enable her to Or him continuously adapt to a variety of changing environmental conditions. This adaptation allows a person to act and react to their surroundings and change their behaviour through learning. The same would be thus expected of the artificial intelligent agent.

在其他任务中，模式识别（如前一节所述）是人工智能必须完成的各项功能的组成部分。 识别规律的能力将使她或他能够不断适应各种不断变化的环境条件。 这种适应性允许一个人通过学习来对周围环境采取行动并做出反应改变他们的行为。因此对人工智能代理也有相同的期望。

Or him...

或者他。。。

From that point of view, machine learning can be seen as a subfield of artificial intelligence. In that respect, machine learning has a much humbler aim than artificial intelligence: instead of aspiring to the ultimate sentient robot, machine learning is interested in studying the methods that can be used to improve the performance of an intelligent agent over time, based on stimuli from the environment.

从这个角度来看，机器学习可以看作是人工智能的一个子领域。在这方面，机器学习的目标比人工智能要简单得多：机器学习不是渴望最终的有感知机器人，而是研究基于环境刺激的可用于改善智能代理的性能的方法。

Machine learning is a subfield of artificial intelligence focussed on improving the performance of an intelligent agent.

机器学习是人工智能的一个子领域，专注于提高智能代理的性能。

Notice that although this definition uses evocative language, the stimuli may not necessarily be read in real-time or even directly by the intelligent agent, and for that matter the intelligent agent does not have to be necessarily artificial.

请注意，尽管此定义使用了令人回味的语言，但刺激可能不一定是实时读取，甚至不是由智能代理直接读取，因此智能代理不一定是人工的。

Think for example of a business manager who is interested in understanding what marketing materials have the best response in a certain sector of her online customers. The use of data about the browsing habits of her customers could play the role of the stimuli, and this understanding will help improve performance in her business. It is not hard to see why machine learning has become an essential part of data science and analytics.

举例来说，业务经理想要了解哪些营销材料是其在线客户最受欢迎的。使用其客户浏览习惯的数据可以起到作用，这种理解将有助于提高她的业务绩效。难看出为什么机器学习已经成为数据科学和分析的重要组成部分。

The use of data generated by businesses can be used to improve their performance.

通过使用企业生成的数据，可用来改善其绩效。

Machine learning has become ubiquitous in modern life, for example, every time that you check your email and identify a SPAM email in your inbox, you are providing an extra example to an algorithm (quite possibly a naïve Bayes classifier - see Section 6.4.2) that will adapt and learn in order to catch similar SPAM in the future. Similarly, online retailers are able to target products to customers based on items that other similar customers have purchased before with the aid of collaborative filtering. We mentioned image recognition earlier on, and other examples include fraud detection, advertisement placement, web search, etc.

机器学习在现代生活中已经无处不在，例如，每次检查电子邮件并在收件箱中识别垃圾邮件时，您都会为算法提供一个额外的例子（很可能是一个朴素贝叶斯分类器 - 参见第6.4.2节）以便在未来捕获类似的垃圾邮件。 类似地，在线零售商借助协同过滤，通过分析其他类似客户之前购买的商品，能够将产品定向到客户。我们之前提到了图像识别，还有其他例子包括欺诈检测，广告放置，网络搜索等。

The use of machine learning has become ubiquitous in modern life.

See Section 8.4.2 for more information on collaborative filtering.

机器学习的使用已经在现代生活中普遍存在。

有关协同过滤的更多信息，请参见第8.4.2节。

*3*.*3* Data is Good, but other Things are also Needed

3.3数据很好，但也需要其他东西

Machine learning may offer us a substantial advantage - and insight - into the problems and questions in our business. In that respect, all we seem to need is an enormous amount of data at our disposal and the rest should follow. Data is indeed an asset in this respect and Data must be treated as an asset. given current trends, data availability may not be a problem. However, we need to assess whether the data available is indeed relevant to the questions we are seeking to answer.

机器学习可以为我们业务中遇到的问题提供重要洞察力。在这方面，似乎我们需要的是掌握大量数据。这个方面数据确实的资产，而且数据必须作为资产处理。鉴于目前的趋势，数据可用性可能不是问题。但是，我们需要评估可用数据是否确实与我们要回答的问题相关。

Data must be treated as an asset.

必须将数据视为资产。

As you can imagine, it is fairly easy to go down the route that more data is always better. Nonetheless, it may be the case that having access to better relevant data is preferable to having so-called big data. I maintain that any efforts we can spend in improving our data are worth investigating and investing in. After all, the patterns we are trying to exploit can only be as good as the data we employ.

正如您可以想象的那样，沿着更多数据的路线走下去相当容易。尽管如此，可能的情况是，访问更好的相关数据比拥有所谓的大数据更可取。我坚持认为，我们在改进数据方面所做的任何努力都值得调查和投资。毕竟，我们试图利用的模式只能与我们采用的数据一样好。

Having relevant data is preferable to having so-called big-data, particularly at early stages.

具有相关数据优于具有所谓的大数据，特别是在早期阶段。

With that in mind, it is often the case that having a wide variety of data may be more important than having a lot of it. By the same token, being able to apply a variety of clever algorithms may prove to be much more fruitful than simply having rows and rows of raw data. What is more, if the algorithms employed are scalable, adding more data may be a straightforward task.

考虑到这一点，通常情况下，拥有各种各样的数据可能比拥有大量数据更重要。出于同样的原因，能够应用各种聪明的算法可能比简单地拥有大量的原始数据更有效。更重要的是，如果所采用的算法是可扩展的，则添加更多数据可能是一项简单的任务。

Using a variety of algorithms is preferable to having lot of data that is not being used.

使用各种算法比拥有大量未使用的数据更可取。

We mentioned above the need to have relevant data, and the challenge there is to identify when we do indeed have it and when we do not. If we happen to be well-versed in the business domain where the data is being generated, we may have a good chance to decide if it is relevant or not. However, in cases where we do not have such experience, we should be able to face this challenge by enlisting the help of people with experience in the subject matter area. Having that expertise at our disposal can be as valuable as the data itself!

我们在上面提到了获得相关数据的必要性，以及判断何时确实需要相关数据以及什么时候不需要相关数据。 如果我们恰好熟悉生成数据的业务领域，我们可以判断它是否相关。 但是，如果我们没有这样的经验，我们应该通寻求相关领域有经验的人的帮助来应对这一挑战。 拥有我们掌握的专业知识可以与数据本身一样有价值！

Make sure you have access to relevant subject matter expertise too. It may prove as valuable as the data itself..

确保您也可以访问相关的主题专业知识。 它可能证明与数据本身一样有价值。

It is therefore recommended to have discussions and reviews with subject matter experts from an early stage in the process. This is particularly true in cases where the data science team may not be fortunate enough to have such expertise.

因此，建议在此过程的早期阶段与领域的专家进行讨论和审查。 在数据科学团队不拥有此类专业知识的情况下尤其如此。

Additionally, if we are indeed interested in gaining insight from the data, it is also important to discuss the results of the modelling stages with subject matter experts and decision makers. These discussions need to be organized with the understanding that not all of the people involved may be able to follow intricate and lengthy explanations about the finer points of a particular machine learning algorithm.

此外，如果我们确实有兴趣从数据中获得洞察力，那么与领域专家和决策者讨论建模阶段的结果也很重要。需要组织这些讨论，同时了解并非所有相关人员都能够对特定机器学习算法的细节进行复杂而冗长的解释。

Communicating your process and results, from the early stages is an important component to becoming a successful jackalope data scientist.

从早期阶段交流您的过程和结果是成为一名成功的jackalope数据科学家的重要组成部分。

It is therefore important to be able to communicate effectively about the main issues in the process in an inclusive manner. It is only then that the actual effectiveness of the data science process is realised.

因此，能够以包容的方式有效地就过程中的主要问题进行沟通是很重要的。只有这样才能实现数据科学过程的真实高效。

*3*.*4* Learning, Predicting and Classifying

3.4学习，预测和分类

The implementation of machine learning algorithms involves the analysis of data that could be employed in the improvement (learning) of the agent (model) and subsequently using the results to make predictions about quantities of interest or making decisions in the face of uncertainty.

机器学习算法的实现涉及分析可用于改进代理模型的数据，并使用结果来预测感兴趣的量或在面对不确定性时做出决定。

It is important to bear in mind that machine learning is interested in the regularities or patterns of the data in order to provide predictive and/or classifying power. This is not necessarily the same as causality. We would need a more thorough examination to claim causes and effects given the data we observe.

重要的是要记住，为了提供预测和/或分类能力，机器学习对数据的规律性或模式感兴趣，这不一定与因果关系相同。根据我们观察到的数据，我们需要进行更彻底的检查才能声明原因和结果。

Machine learning is interested in regularities and patterns in data.

机器学习对数据的规律性和模式感兴趣。

Machine learning tasks are traditionally divided into two camps: Predictive or supervised learning and descriptive or unsupervised learning. Let us start with supervised learning: A good example of this type of task is that of a traditional teacher-pupil situation where the teacher presents the pupil with a number of known examples to learn from.

机器学习任务传统上分为两个阵营：预测或监督学习以及描述性或无监督学习。让我们从监督学习开始：这种类型的任务的一个很好的例子是传统的师生情况，教师向学生展示一些已知的例子供学习。

We talk about two types of tasks: supervised and unsupervised ones.

我们谈论两种类型的任务：监督和无监督。

Let us return to the classification of cat faces: A teacher that knows what a cat looks like will present the pupil with several training images of cats and other animals, and the pupil is expected to use the features or attributes of the images presented to learn what a cat looks like. The teacher will have provided a label to each of the images as being of cats or not. In the testing part, the teacher will present images of various kinds of animals, and the pupil is expected to classify which ones show a friendly feline face.

让我们回到猫脸的分类：老师知道猫长什么样子，老师向学生展示猫和其他动物的几个训练图像，学生根据图像的特征或属性来学习猫看起来像什么。老师会为每个图像提供一个标签，如果是猫的标签。在测试部分，教师将呈现各种动物的图像，并且期望学生对图像进行正确的分类。

Supervised learning makes use of labelled data.

In machine learning parlance we talk about supervised learning when we are interested in learning a mapping from the input to output with the help of labelled sets of input-output pairs. Supervised learning lets us make predictions based on the data that we see and thus apply generalisations.

在机器学习中，我们说的监督学习是指当在标记的输入 - 输出对的帮助下学习从输入到输出的映射。 有监督的学习使我们能够根据我们看到的数据进行广泛的预测。

Supervised learning lets us make predictions.

监督学习让我们做出预测。

Each input has a number of features that can be represented in terms of an N-dimensional vector that will help in the task of learning the label of each of the training examples. Think of a supervised learning task as providing an annotated map to a mountaineer that is signing up to our Kilimanjaro expedition and asking her to identify similar landscape features to those marked on the map as she walks along.

每个输入都具有许多可以用N维向量表示的特征，这将有助于学习每个训练样例的标签。 将监督学习任务想象为登山者提供带注释的地图，登山者注册成为乞力马扎罗山（非洲最高的山）探险队的成员，并要求她识别与地图上的标记相似的景观特征。

Or, where there two mountaineers? Oh well...

或者，哪里有两名登山者？ 那好吧...

The other type of machine learning task is unsupervised learning. In this case, following our example of the teacher-pupil situation, the teacher takes a Montessori-style approach and lets the pupil develop, on her own, a rule about what a cat (or any other animal of the pupil’s preference) looks like, without providing any hints or labels to the learner.

另一种机器学习任务是无监督学习。 在这种情况下，按照我们的教师—学生的例子，教师采取蒙台梭利式的方法，不向学习者提供任何提示或标签，让学生自己发展一个关于猫（或学生的偏好的任何其他动物）看起来像什么的规则。

In this case, from a machine learning point of view, there are no input-output pairs. Instead, we only have the unlabelled inputs and their associated N-dimensional feature vectors, without being told the kind of pattern that we must look for. In that respect, an unsupervised learning task is less well-defined than a supervised one.

在这种情况下，从机器学习的角度来看，没有输入—输出对。相反，我们只有未标记的输入及其相关的N维特征向量，而没有被告知我们必须寻找那种模式。在这方面，无监督学习任务的定义不如监督学习任务。

In contrast, unsupervised learning does not make use of labelled data

相反，无监督学习不使用标记数据。

That does not mean that it is less useful, on the contrary, we can use unsupervised learning to gain a better understanding of the data we have acquired and it can provide us with a description or classification of the dataset as well as discovering interesting patterns in the data. In other words unsupervised learning lets us represent our data better by extracting **structure** from it.

这并不意味着它没那么有用，相反，我们可以使用无监督学习来更好地理解我们获得的数据，它可以为我们提供数据集的描述或分类，以及发现有趣的模式。换句话说，无监督学习让我们通过从数据中提取结构来更好地表示我们的数据。

Unsupervised learning can help us understand the structure of our data, providing us with ways to describe or classify data points.

无监督学习可以帮助我们理解数据结构，为我们提供描述或分类数据点的方法。

Under unsupervised learning, in the case of our Kilimanjaro mountaineers, we would ask them to go on their journey without an annotated map, and instead identify interesting areas in the landscape they are able to see from the summit. One important thing to note is that an unsupervised learning task may enable us to assign labels to those inputs and thus open the door to the use of predictive or supervised learning.

在无人监督的学习中，对应乞力马扎罗山登山者的例子，我们会要求他们在没有带注释的地图的情况下继续他们的旅行，他们可以从山顶看到的景观中找出感兴趣的区域。需要注意的事情是，无监督的学习任务可以使我们为这些输入分配标签，从而打开使用预测或监督学习的大门。

Unsupervised learning may provide us with labels to be used in a supervised learning task.

无监督学习可以为我们提供在监督学习任务中使用的标签。

We have touched upon data being labelled or not, and that has given us some clues as to the sort of problems we can tackle with each of them. Let us now turn our attention to the features and labels (if they exist). In Section 3.1 we gave an example of some of the features that would enable us to recognise a cat. Some of those features can be quantified, for example we mentioned that it must be a small furry animal: How small? We can associate a number to this measurement and then we would be talking about a numerical or continuous variable. Continuous variables are typically associated with measurement units and we can represent them with real numbers.

我们已经触及了有标记和无标记的数据，这为我们提供了一些关于我们可以解决问题的线索。 现在让我们将注意力转向特征和标签（如果他们存在的话）。在3.1节中，我们给出了一些使我们能够识别猫的特征的例子。其中一些功能可以量化，例如我们提到它必须是一个小型的毛茸茸的动物：有多小？ 我们可以将一个数字与该测量值相关联，然后我们将讨论数值或连续变量。 连续变量通常与测量单位相关联，我们可以用实数表示它们。

Numerical features are associated with measurement units and represented by numbers.

数字特征与测量单位相关联并由数字表示。

We may also have attributes that cannot be represented by a number, and instead they provide a description of the type of attribute we are referring to. In the example of the cat we mentioned triangular ears, instead of round or floppy ones. Other examples include colour (black cat, white cat), gender (male/female), etc. We refer to these attributes as categorical or nominal variables, and are typically related to a class or category.

我们也可能有一些不能用数字表示的属性，提供了属性类型的描述。 在猫的例子中，我们提到了三角形耳朵，而不是圆形或松软的耳朵。 其他示例包括颜色（黑猫，白猫），性别（男/女）等。我们将这些属性称为分类或名义变量，并且通常与类或类别相关。

Categorical features provide a description of the attributes of the data.

分类特征提供数据属性的描述。

This categorisation among the type of features and labels in a dataset may seem superfluous. However, a further bit of scrutiny will let us see that this innocuous grouping enables us to identify the type of machine learning algorithms that may be more suitable to the problems we would like to tackle. Take a look at Table 3.1, where we provide some quintessential machine learning algorithms that fall into each category.

对数据集中的要素类型和标签进行分类似乎是多余的。 然而，进一步审查将让我们看到这种无害的分组使我们能够识别可能更适合我们想要解决的问题的机器学习算法的类型。 请看表3.1，其中我们提供了属于每个类别的一些典型机器学习算法。

学习类型 分类的 连续的

监督 回归 分类

无监督 降维 聚类

Table 3.1: Machine learning algorithms can be classified by the type of learning and outcome of the algorithm

表3.1：机器学习算法可以根据学习类型和算法结果进行分类。

This distinction between feature types will make it easier for us to understand the sort of machine learning task we want to employ.

特征类型之间的这种区别将使我们更容易理解我们想要使用的机器学习任务。

*3*.*5* Machine Learning and Data Science

3.5机器学习和数据科学

Many problems that we would like to tackle using machine learning tend to have high complexity. We have to bear this in mind when trying to apply algorithms, as it is not very likely to find a perfect practical solution. Nonetheless, if the machine can learn so can we. Machine learning algorithms are suitable to the solution of problems encountered in the data science and analytics workflow where we are interested in deriving valuable insights from data.

我们希望使用机器学习解决的许多问题往往具有高度复杂性。 在尝试应用算法时我们必须牢记这一点，因为它不太可能找到一个完美的实际解决方案。 尽管如此，如果机器可以学习，我们也可以。 机器学习算法适用于数据科学和分析流程中问题的解决方案，我们有兴趣从数据中获取有价值的见解。

Remember that if the machine can learn, so can we.

请记住，如果机器可以学习，我们也可以。

Let us take for instance the case of a supervised learning task where our ultimate aim is to find a function h(x), called the hypothesis. This function enables us to predict values for the problem at hand based on the given input data x. In a practical case, the inputs in the feature vector x are varied and we would have to decide what the important features to take into account are, and then include them in our model.

让我们举一个监督学习任务的例子，我们的最终目标是找到一个函数h（x），称为假设。 此功能使我们能够根据给定的输入数据x预测问题的值。 在实际情况中，特征向量x中的输入是变化的，我们必须决定要考虑的重要特征是什么，然后将它们包含在我们的模型中。

We need to decide what are the important features to include in our models.

我们需要确定在模型中包含哪些重要特征。

The optimisation of the predictor h(x) is done using training data points so that for each one of them we have input values xtrain that correspond to an output y which is known in advance. Learning, in this sense, is thus the effective use of data in the task of training a model in order to accomplish the job it was set to complete.

使用训练数据点完成预测器h（x）的优化，使得对于它们中的每一个，我们具有对应于预先已知的输出y的输入值xtrain。 因此，从这个意义上讲，学习是在训练模型的任务中有效地使用数据，以完成它设定完成的工作。

In supervised learning we know the output y in advance.

在有监督的学习中，我们提前知道输出y。

From this point of view, we can relate the tasks that involve that training to the data science workflow steps we listed in Section 1.4: Once we have identified the task at hand, we need to acquire relevant data, extract pertinent features and build our model. In addition to those steps, we also have to consider three important parts that will enable us to decide what sort of machine learning algorithm to choose for our problem.

从这个角度来看，我们可以将涉及该训练任务与我们在1.4节中列出的数据科学工作流程步骤联系起来：一旦我们确定了手头的任务，我们需要获取相关数据，提取相关特征并构建我们的模型。 除了这些步骤之外，我们还必须考虑三个重要部分，这些部分将使我们能够决定为我们遇到的问题选择何种机器学习算法。

The steps to follow are all framed by the data science workflow discussed in Section 1.4

要遵循的步骤都是由1.4节中讨论的数据科学工作流程构成的

With each prediction that we make, we can find the difference between that prediction and the true output value. We do this in order to assess how well out predictor is performing. An important part of the process is to be able to obtain a model that is able to perform well in a general setting, rather than memorising the intricacies of the data provided.

通过我们进行的每个预测，我们可以找到该预测与真实输出值之间的差异。 我们这样做是为了评估预测器的执行情况。该过程的一个重要部分是获得能够在一般条件下可以良好运行的模型，而不是只对所训练的数据起作用的模型。

Generalisation is an important outcome of the model.

泛化是该模型的重要结果。

For example, if we are interested in building an algorithm that is able to recognise cat faces, we would like it to perform well with new, previously unseen cats. If the algorithm is only able to recognise Bowman, the Iberian lynx, but not Mittens, the kitty, then it is not a great algorithm to be deployed. If, however, the algorithm is able to recognize that a cat is a cat even if it is a drawing, a photograph or an actual real-life cat, then the algorithm is great. We shall come back to this point when we discuss the evaluation of algorithms in Section 3.8.

例如，如果我们有兴趣构建一个能够识别猫脸的算法，我们希望它能够很好地适应新的，以前没有见过的猫。如果算法只能识别Bowman，lberian lynx，却不能识别Mittens,kity，那么它不是一个很好的部署算法。 然而，即使要识别的物体是绘图，照片或真实猫，如果算法能够识别出猫是猫，那么算法也很棒。 当我们讨论第3.8节中的算法评估时，我们将回到这一点。

Model evaluation must be part of the entire process.

模型评估必须是整个过程的一部分。

At this point, it is pertinent to make clear that there is no such thing as a perfect model, just good enough ones. The improvement in learning comes from generalising regular patterns in the training data to be able to say something about unobserved data points. We should therefore be careful not to obtain a model that “memorises” the data, also known as overfitting. We can avoid this by employing techniques such as regularisation and cross-validation as we shall see later on in this chapter.

在这一点上，明确没有完美的模型，只有足够好的模型是合适的。学习的改进来自于在训练数据中泛化常规模式，以便能够对未观察到的数据点进行说明。因此，我们应该注意不要获得“记忆”数据的模型，也称为过度拟合。 我们可以通过采用正则化和交叉验证等技术来避免这种情况，我们将在本章后面看到。

There is no such thing as a perfect model.

没有完美的模型这样的东西。

We shall discuss more about avoiding this memorisation in Sections 3.7 and 3.12.

我们将在3.7节和3.12节中讨论更多关于避免记忆的问题。

*3*.*6* Feature Selection

3.6特征选择

Machine learning can be a powerful tool under our jackalope data scientist belt. Not only can it be used together with computer science, mathematics and statistics to help us filter and prepare our data, but also to extract value out of it. It is therefore important to be able to separate the valuable relationships and patterns from any random, confounding ones. In any real application it is inevitable to have a mix of distracting noise together with the signals we want to exploit.

机器学习是一个强大的工具，它不仅可以与计算机科学，数学和统计学一起使用，以帮助我们过滤和准备我们的数据，还可以从中提取价值。 因此，能够将有价值的关系和模式与任何随机的，混杂的关系分开是很重要的。 在任何实际应用中，不可避免地会将令人分心的噪声与我们想要利用的信号混合在一起。

Unprocessed data can thus be thought of as the raw material that can be filtered and prepared to obtain the insights desired. However, as it is the case with cooking, the quality of the ingredients is as important as the steps specified in the recipe. With that analogy in mind, we need to be able to think through the available independent variables or features (ingredients) that will be included in the model (recipe).

因此，未处理的数据可以被认为是可以通过过滤以获得所需见解的原材料。然而，正如烹饪的情况一样，成分的质量与配方中指定的步骤一样重要。考虑到这个类比，我们需要能够思考可以包含在模型（配方）中的可用自变量或特征（成分）。

We have to discern between actual patterns and random noise.

我们必须辨别实际模式和随机噪声。

Like in cooking, the quality of the ingredients is as important as the steps of the recipe.

与烹饪一样，食材的质量与食谱的步骤同样重要。

In some cases using the unprocessed, raw data may be suitable. However, in many cases it is preferable to create new features that synthesise important signals spread out in the raw data. This process is known as feature selection where not only should we consider the the features readily available, but also the creation and extraction of new features and even the elimination of some variables too.

在某些情况下，使用未处理的原始数据可能是合适的。但是，在许多情况下，最好创建新的功能，以合成在原始数据中分散的重要信号。 这个过程被称为特征选择，我们不仅要考虑现有的特征，还要考虑新特征的创建和提取，甚至消除一些变量。

Feature selection considers both existing features, and the creation of new ones.

特征选择既考虑现有特征，也考虑新特征的创建。

The careful selection of the features to be used in the modelling helps with the understanding of the model outcomes. It also has a large effect in the predictions obtained from the application of machine learning algorithms. A common way to create new features is via mathematical transformations that make the variables suitable for exploitation by a particular algorithm. For instance, many algorithms rely on features having a linear relationship, and finding a transformation that renders nonlinear features to be represented as being linear in a different feature space is definitely worth considering. We will see some example of this in the next chapter and also in Section 9.1.

仔细选择要在建模中使用的特征有助于理解模型结果。 它对从机器学习算法的应用中获得的预测也有很大的影响。创建新特征的常用方法是通过数学变换使变量适合特定算法的利用。 例如，许多算法依赖于具有线性关系的特征，并且找到使非线性特征在不同特征空间中表示为线性的变换是绝对值得的。我们将在下一章和第9.1节中看到一些这样的例子。

Mathematical transformations are a typical way to create new features.

数学变换是创建新特征的典型方法。

It is true that knowing, a priori, the appropriate transformations and aggregations we should make is a hard task in and of itself. In many cases, experience with similar datasets and comparable applications is invaluable. Nonetheless, if you are starting up not all is lost. Fortunately another common way to extract features is to use machine learning itself.

确实，先验地知道我们应该做出的适当的转换和聚合本身就是一项艰巨的任务。 在许多情况下，拥有相似的数据集和类似应用的经验非常宝贵。尽管如此，如果你没有经验，也不是什么都做不了。幸运的是，还有一种通过机器学习本身来提取特征向量的常用方法。

Experience with data transformations is an invaluable asset to be used.

数据转换经验是一项非常宝贵的资产。

In this case, unsupervised learning may provide a way to find useful clusters (see Section 5.1) in the data that may point us out in the right direction. Similarly, dimensionality reduction (see Section 8.1) can help us to determine combinations of features that explain the variance shown in our dataset. We shall have an opportunity to talk about these types of algorithms later on in the book.

在这种情况下，无监督学习可以提供一种方法，可以在数据中找到有用的聚类（参见第5.1节），这可能为我们指出正确的方向。 同样，降维（见8.1节）可以帮助我们确定特征组合，解释我们数据集中显示的方差。 我们有机会将在本书的后面讨论这些类型的算法。

Unsupervised learning can also be useful in the feature selection process.

无监督学习在特征选择过程中也很有用。

3.7 Bias, Variance and Regularisation: A Balancing Act

3.7偏见，差异和正规化：平衡法

As we have mentioned in the previous section, machine learning algorithms enable us to exploit the regularities in the data. Our task is therefore to generalise those regularities and apply them to new data points that have not been observed. This is called generalisation, and we are interested in minimising the so-called generalisation error, i.e. a measure of how well our model performs against unseen data.

正如我们在上一节中提到的，机器学习算法使我们能够挖掘数据中的规律。 因此，我们的任务是概括这些规律并将其应用于尚未观察到的新数据点。这被称为泛化，我们感兴趣的是最小化所谓的泛化误差，衡量我们的模型对看不见的数据的表现。

Generalisation refers to the performance of a model against unseen data.

泛化是指模型针对看不见的数据的性能。

If we were able to create an algorithm that is able to recall the exact noise in the training data, we would be able to bring our training error down to zero. That sounds great and we would be very happy until we receive a new batch of data to test our model. It is quite likely that the performance of the model is not as good as a zero generalisation error would have us believe. We have ended up with an overfit model: We would be able to describe the noise in our data instead of uncovering a relationship, given the variance in our data.

如果我们能够创建一个能够模拟训练数据中的确切噪声的算法，我们就能够将训练误差降低到零。 这听起来不错，我们会非常开心直到我们收到一批新数据来测试我们的模型。 模型的性能很可能不如我们认为的零泛化误差那么好。 我们最终得到了一个过度拟合模型：因为我们的数据存在差异，我们能够描述数据中的噪声而不是发现关系。

In principle we can bring our training error down to zero. Unfortunately this translates into a larger generalisation error.

原则上我们可以将训练误差降至零。 不幸的是，这转化为更大的泛化错误。

The key is to maintain a balance between the propensity of our model to learn the wrong thing, i.e the bias, and the sensitivity to small fluctuations in the data, i.e. the variance. In the ideal case scenario we are interested in obtaining a model that encapsulates patterns in the training data, and that at the same time generalises well to data not yet observed. As you can imagine, the tension between both tasks means that we cannot do both equally well and a trade-off must be found in order to represent the training data well (high variance) without risking overfitting (high bias).

关键是要保持偏差和方差之间的平衡，我们模型学习错误的倾向，即偏差，对数据小波动的敏感性，即方差。在理想情况下，我们感兴趣的是获得一个模型，该模型将模式封装在训练数据中，同时很好地概括了尚未观察到的数据。可以想象，为了权衡这两个任务，意味着我们不能同时做到这两点，必须找到一个平衡点以便很好地表示训练数据（高方差）而不会有过度拟合（高偏差）的风险。

The key is to maintain a balance between bias and variance.

关键是要在偏差和方差之间保持平衡。

High-bias models typically produce simpler models that do not overfit and in those cases the danger is that of underfitting. Models with low-bias are typically more complex and that complexity enables us to represent the training data in a more accurate way. The danger here is that the flexibility provided by higher complexity may end up representing not only a relationship in the data but also the noise. Another way of portraying the bias-variance trade-off is in terms of complexity v simplicity.

高偏差模型通常会生成更简单的模型，而不会过度拟合，但这种情况会导致欠拟合。 具有低偏差的模型通常更复杂，这种复杂性使我们能够以更准确的方式表示训练数据，但是也会带来其他的问题，这种更高复杂性所提供的灵活性最终可能不仅代表数据中的关系，还代表噪声。描绘偏差-方差权衡的另一种方式是复杂性和简单性。

Another way to look at this is in terms of complexity versus simplicity of a model.

另一种看待这种情况的方法是模型的复杂性与简单性。

The tension between bias and variance, simplicity and complexity, or underfitting and overfitting is an area in the data science and analytics process that can be closer to a craft than a fixed rule. The main challenge is that not only is each dataset different, but also there are data points that we have not yet seen at the moment of constructing the model. Instead, we are interested in building a strategy that enables us to tell something about data from the sample used in building the model.

偏差和方差之间的矛盾，简单性和复杂性，或欠拟合和过度拟合是数据科学和分析过程中的一个领域，它可以更接近技巧而不是固定规则。 主要的挑战是，不仅每个数据集都不同，而且我们在构建模型时数据集并不完善。相反，我们有兴趣构建一种策略，这使我们能够从构建模型所用的样本中分辨出数据。

Keeping that balance is more an art than a science

保持这种平衡更像是一门艺术，而不是一门科学

In order to prevent overfitting it is possible to introduce ways to penalise our models for complexity by adding extra constraints such as smoothness, or requiring bounds in the norm of the vector space we are working on - more on this later on.

为了防止过度拟合，可以为模型增添一些复杂性的方法来惩罚模型，比如添加额外约束（例如平滑度）或对正在处理的向量空间增加边界- 稍后将详细介绍。

This process is known as regularisation, and the effects of the penalty introduced can be adjusted with the use of the so-called regularisation hyperparameter, λ.

这个过程称为正则化，并且可以使用所谓的正则化超参数λ来调整引入的惩罚的影响。

Regularisation is the process of introducing to our model a penalty for complexity.

正规化是向我们的模型引入复杂性惩罚的过程。

Regularisation can then be employed to fine-tune the complexity of the model in question. In a sense it is a way to introduce the Occam’s razor principle to our model.

然后可以采用正则化来微调所讨论的模型的复杂性。 从某种意义上说，这是一种将奥卡姆剃刀原理引入我们模型的方法。

Occam’s razor tells us that when we have two competing theories that make the same predictions, the simpler one is preferred.

奥卡姆剃刀原理告诉我们，当我们用两个相互竞争的理论得到相同的预测时，更简单的理论是首选。

Some typical penalty methods that are introduced for regularisation are the L1 and L2 norms that we will discuss in the following section. In Section 3.12 we will touch upon how the hyperparameter λ can be tuned with the use of cross-validation.

为正则化引入的一些典型惩罚方法是L1和L2规范，我们将在下一节中讨论。 在3.12节中，我们将讨论如何使用交叉验证来调整超参数λ。

3.8 Some Useful Measures: Distance and Similarity

3.8一些有用的措施：距离和相似性

Once we have built a set of models based on the training data we have, it is important to distinguish a good performing model against a less good one. So, how do we ascertain that a model is good enough for our purposes? The answer is that we need to evaluate the models with the aid of a scoring or objective function.

一旦我们根据训练数据构建了一组模型，重要的是将表现良好的模型与不太好的模型区分开来。那么，我们如何确定模型对我们来说足够好呢？答案是我们需要借助评分或目标函数来评估模型。

Remember that we are working with the principle that models are good enough.

请记住，我们正在使用模型足够好的原则。

Various machine learning algorithms have appropriate ways to let us evaluate how much we can trust what had been learned and how predictive the model obtained is. The performance of a model will therefore depend on various factors such as the distribution of classes, the cost of misclassification, the size of the dataset, the sampling methods used to obtain the data, or even the range of values in the selected features. It is important to note that evaluation measures are usually specialised to the type of problem and algorithm used, and the score provided will be meaningful to the problem domain. For instance, in the case of classification problems, the classification accuracy may provide a more meaningful score than other measures.

各种机器学习算法都有适当的方法让我们评估在多大程度上可以信任机器学习学到的内容以及模型的预测性如何。因此，模型的性能将取决于各种因素，例如类的分布，错误分类的成本，数据集的大小，用于获取数据的采样方法，或甚至所选特征中的值范围。值得注意的是，评估度量通常专门针对所使用的问题类型和算法，并且所提供的分数对问题域有意义。 例如，在分类问题的情况下，分类准确性可以提供比其他度量更有意义的分数。

Evaluation measures are usually specialised to the type of algorithm used.

评估措施通常专用于所使用的算法类型。

In general model evaluation can be posed as a constrained optimisation problem given an objective function. The aim can then be presented as the problem of finding a set of parameters that minimises that objective function. This is a very useful way to tackle the problem as the evaluation measure can be included as part of the objective function itself. For example, consider the case where we are interested in finding the best line of fit given a number of data points: A perfect fit would be found in the case where the data points align flawlessly in a straight line. As you can imagine, that is very rarely the case.

通常，模型评估可以作为给定目标函数的约束优化问题。 然后可以将目标表示为找到使该目标函数最小化的一组参数的问题。这是解决问题的一种非常有用的方法，因为评估措施可以作为目标函数本身的一部分。例如，考虑我们有兴趣在给定多个数据点的情况下找到最佳拟合线的情况：在数据点拍成一条直线时，可以找到完美的拟合。可以想象，这种情况很少发生。

Model evaluation can be posed as a constrained optimization problem.

模型评估可以作为约束优化问题。

We will discuss regression in Chapter 4.

我们将在第4章讨论回归。

Instead of expecting the unexpected, we can evaluate how well a line fits the data when we take into account the difference between the location of a point and its corresponding prediction as obtained from the model. If we minimise that distance then we can evaluate and compare various calculated predictions. This particular evaluation measure used in regression analysis is known as the sum of squared residuals (SSR) and we will discuss it in more detail in Chapter 4.

通过计算点的实际位置与从模型中预测到的点的位置之间的差异，我们可以评估线与数据的拟合程度，不考虑意外发生的程度。如果我们最小化该距离，那么我们可以评估和比较各种计算到的预测。回归分析中使用的这种特殊评估指标称为残差平方和（SSR），我们将在第4章中对其进行更详细的讨论。

In regression, the minimization of the sum of squares error is a typical evaluation measure.

在回归中，平方和误差的最小化是典型的评估度量。

As we can see, the concept of distance arises naturally as a way to express the evaluation problem, and indeed a number of conventional evaluation procedures rely on measures of distance. Consider the points A and B in a two dimensional space shown in Figure 3.1. Point A has coordinates p（，） and point B has coordinates q（，）. We are interested in calculating the distance between these two points. This can be achieved in different ways and we are familiar with some of these, such as the Euclidean and the Manhattan distances.

正如我们所看到的，距离的概念作为表达评价问题的一种方式自然而然地产生了，事实上许多传统的评价程序依赖于距离的度量。考虑图3.1中所示二维空间中的点A和B。点A的坐标p（，），点B的坐标q（，）。我们感兴趣的是计算这两点之间的距离。这可以用不同的方式来实现，我们熟悉其中的一些，例如欧几里得和曼哈顿距离。

Various evaluation measures rely on measures of distance.

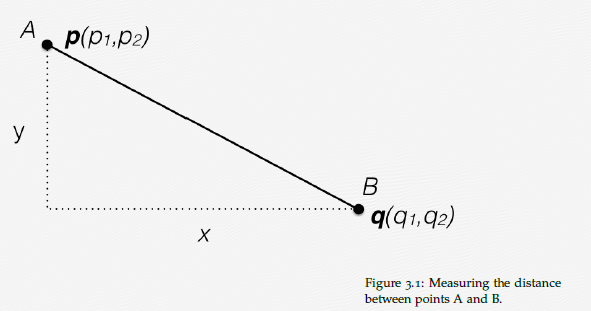
各种评估措施依赖于距离的测量。

* Euclidean distance: This corresponds to the ordinary distance calculated using the straight line that joins points A and B; in two dimensions it corresponds to the distance given by the Pythagorean theorem. Given the coordinates of each of the two points in question we can obtain the distance between A and B as:

欧几里得距离：这与用连接点A和B的直线计算的普通距离相对应；在二维上，它与毕达哥拉斯定理给出的距离相对应。问题中分别给定2个点的坐标，我们可以得到A和B之间的距离：

Remember the Pythagoras theorem?

还记得毕达哥拉斯定理吗？



= = (3.1)

where the distances x and y are shown in Figure 3.1. It is possible to extend this definition to n dimensions:

其中x和y的距离如图3.1所示。将这个定义扩展到N维：

= = (3.2)

This is the well known Euclidean distance.

这是众所周知的欧几里德距离。

where is the distance along the i-th dimension. The Euclidean distance is also known as L2-norm.

其中是沿第i维的距离。欧几里得距离也称为L2范数。

We also call it the L2-norm.

我们也称之为L2规范。

* Manhattan distance: It is easy to see why this distance measure gets this name if we think of the distance that a yellow cab would cover while travelling along the streets in Manhattan. Apart from Broadway, the cab cannot move diagonally in the street-avenue grid. Instead, it can only move North-South and East-West. In the case of points A and B in Figure 3.1, the Manhattan distance is given by

To measure the Manhattan distance think of a yellow New York taxi cab and its journey through the island.

为了衡量曼哈顿的距离，想想一辆黄色的纽约出租车及其穿越岛屿的旅程。

= |()| = |x + y| (3.3)

曼哈顿距离：如果我们想一想一辆黄色出租车在曼哈顿沿着街道行驶时，所能行驶的距离，就很容易看出这种距离度量为什么用这个名字。除了百老汇，出租车不能在街道围城的格子中沿着对角线行驶。相反，它只能在南北和东西方向行驶。在图3.1中的点A和B的情况下，曼哈顿距离由下面的公式计算：

= |()| = |x + y| (3.3)

For an n-dimensional space we can extend the above definition as:

对n维空间，我们可以把上面的定义扩展为：

= |()| = || (3.4)

The Manhattan distance is also known as L1-norm.

曼哈顿距离也被称为L1范式。

From a geometrical point of view the idea of measuring the distance between two points makes intuitive sense. Furthermore, if the distance is zero we can argue that the two points are effectively the same one, or at the very least similar to one another. This idea of similarity is therefore another useful tool in the development of evaluation measures, particularly in the case where features are not inherently amenable to being placed in a geometric space.

从几何学的角度来看，测量两点之间距离的想法非常直观。此外，如果距离为零，我们可以认为这两个点实际上是相同的，或者至少彼此相似。因此，这种相似性思想是评估计算过程中的另一个有用工具，特别是在特征本身不能被放置在几何空间中时。

Another important concept, related to that of distance, is similarity.

与距离相关的另一个重要概念是相似性。

Given two points A and B, the similarity measured must satisfy a certain number of general conditions:

1. Must be positive: d(A, B) 0

2. If the measure is zero, the points A and B are equal and vice versa: d(A, B) = 0 A = B

3. Must be symmetrical: d(A, B) = d(B, A)

4. Must satisfy the triangle inequality: d(A, B) + d(B, C) d(A, C)

给定两个点A和B，测量的相似性必须满足下列一般条件：

1．必须是正的：d(A, B) 0

2．如果测量值为零，则点A和B相等。反之亦然：d(A, B) = d(B, A)

3．必须是对称的：d(A, B) = d(B, A)

4．必须满足三角不等式：d(A, B) + d(B, C) d(A, C)

The two distance measures we discussed above can be used to gauge similarity, however there are a number of other useful ways to do this, for example the cosine and Jaccard similarities:

上面我们讨论的这两个距离测量可用于衡量相似性，但是还有许多其他有用的方法可以做到这一点，例如余弦和Jaccard相似性：

Although distance is useful, similarity can be measured in other ways too.

尽管距离很有用，但也可以通过其他方式测量相似度。

* Cosine similarity: This similarity measure is commonly used in text mining tasks, for example. In these cases the words in the documents that comprise the corpora to be mined correspond to our data features. The features can be arranged into vectors and our task is to determine if any two documents are similar or not. Cosine similarity is based on the calculation of the dot product of the feature vectors. It is effectively a measure of the angle between the vectors: If = 0, then cosis 1 and the two vectors are said to be similar. For any other value of q the cosine similarity will be less than 1. The cosine similarity of vectors and is given by:

余弦相似性：举个例子，这种相似性度量通常被用于文本挖掘任务。在这些情况下，被挖掘的语料库的文档中的单词对应于我们的数据特征。这些功能可以被安排到向量中，我们的任务是确定任何两个文件是否相似。余弦相似性基于特征向量的点积的计算。它实际上是向量构成的角度的度量：如果= 0，则cos为1，并且两个矢量被认为是相似的。对于的任何其他值，余弦相似度将小于1.向量和的余弦相似度由下面的公式给出：

(, ) = cos(θ) = , (3.5)

where ||corresponds to the usual Euclidean norm to measure the magnitude of the vector .

通常使用欧几里德范数来测量向量的大小||  
Please note that the cosine similarity is a measure of orientation and not magnitude.

请注意，余弦相似度是方向的度量，而不是幅度。

* Jaccard similarity: The Jaccard similarity measure provides us with a way to compare unordered collections of objects, i.e. sets. We define the Jaccard similarity in terms of the elements that are common to the sets in question. Consider two sets A and B with cardinalities |A| and |B| . The common elements of both sets are given by the intersection A B. In order to give us an idea of the relative size of the intersection compared to the sets, we divide the former by the union of the sets. This can be expressed as follows:

Jaccard相似性：Jaccard相似性度量为我们提供了一种比较无序对象集合（即集合）的方法。我们根据所讨论的集合共有的元素来定义Jaccard相似性。 考虑两个集合A和B，其基数分别为| A | 和| B |。 两个集合的共同元素由交集AB给出。为了让我们了解交集相对于集合的大小，我们将前者除以集合的并集。可以表示如下：

J(A, B) = = . (3.6)

With Jaccard similarity we can compare unordered collections of objects.

我们可以用Jaccard相似性比较无序的对象集合。

In the case of document similarity for example, two identical documents will have a Jaccard similarity of 1 and those completely dissimilar a value of 0. Intermediate values correspond to various degrees of similarity.

例如，在文档相似性的情况下，两个相同的文档具有的Jaccard相似性为1，完全不相似的文档的Jaccard相似性为。中间值对应于不同的相似度。

The comparison of documents is a good candidate for the use of Jaccard similarity.

文档的比较是使用Jaccard相似性很好的候选者。

There are other distance and similarity measures that can be used. The choice will depend to a great extend on the type of problem to tackle as well as the algorithms and techniques to be used to solve the problem. In the following chapters we will address specific algorithms and evaluation measures that are appropriate to each of them.

也可以使用其他距离和相似性度量。如何选择，将在很大程度上取决于要解决的问题类型以及用于解决问题的算法和技术。在接下来的章节中，我们将一一讨论适合于他们的特定算法和评估措施。

There are other ways to measure distance and similarity, these are some of the most useful/common ones.

还有其他方法可以测量距离和相似度，这些是最有用/最常用的方法。

*3*.*9* Beware the Curse of Dimensionality

3.9注意维度的诅咒

We have been referring to data features as an integral part of the ingredients we will use with our machine learning algorithms. Once we identified the features to be included in our model we can consider them as the different dimensions along which our data instances can be placed: For a single feature we have a one-dimensional space, two features can be represented in two dimensions, etc.

我们一直将数据特征称为将与机器学习算法一起使用的成分中的一部分。 一旦我们确定要包含在模型中的要素，我们就可以将它们视为可以放置数据实例的不同维度：对于单个要素，我们有一维空间，两个要素可以用二维表示，等等。

The features selected to be included in our model can be considered as the different dimensions our data inhabit.

选择那些特征包含在我们的模型中，这被称作我们的数据的不同维度。

It follows that as we increase the number of features, the number of dimensions that our model must include is increased too. Not only that, but we will also increase the amount of information required to describe the data instances, and therefore the model.

因此，随着我们增加特征数量，我们的模型包含的维数也会增加。不仅如此，我们还将增加描述数据实例所需的信息量，从而增加模型。

As the number of dimensions increases, we need to consider the fact that more data instances are required, particularly if we are to avoid overfitting. The realisation that the number of data points required to sample a space grows exponentially with the dimensionality of the space is usually called the curse of dimensionality6, a term used by Richard Bellman in the context of dynamic programming, and a great way to describe this issue!

随着维度数量的增加，我们需要考虑增加更多数据实例，特别是当我们要避免过度拟合时。 认识到采样空间所需的数据点数量随着空间的维数呈指数增长通常被称为维度的诅咒6，这是理查德·贝尔曼在动态规划环境中使用的术语，也是描述这个问题的好方法！

As the number of dimensions increases, we need to use more data points to avoid overfitting.

随着维数的增加，我们需要使用更多的数据点来避免过度拟合。

6 Bellman, R. (1961). Adaptive Control Processes: A Guided Tour. Rand Corporation. Research studies. Princeton U.P

6 Bellman，R.（1961）. 自适应控制过程：导览,兰德公司.研究性学习.普林斯顿大学

The curse of dimensionality becomes more apparent in instances where we have datasets with a number of features much larger than the number of data points. We can see why this is the case when we consider the calculation of the distance between data points in spaces with increasing dimensionality.

在我们拥有的数据集的数量远远大于数据点的数量时，维度的诅咒变得更加明显。这就是为什么当维数增加时计算空间中数据点之间的距离会使问题变得明显。

The problem becomes more apparent when we consider the distance between data points in higher dimensional spaces.

当我们考虑高维空间中的数据点之间的距离时，问题变得更加明显。

Let us consider, without loss of generality, that we have a set of M = 10 data instances belonging to three different classes. We are interested in finding the closest neighbour to each of the data points and, in this case, assess if they belong to the same class or not. This is a very simple classification task. We can simplify the discussion by considering the use of a unit length measurement and counting the number of data points that fall within it. We are depicting this situation in Figure 3.2 where we show the ten data instances, represented with a triangle, an open circle and a plus sign, in 1, 2 and 3 dimensions.

让我们在不失一般性的前提下考虑，我们有一组属于三个不同类的数据实例，M = 10。 我们感兴趣的是找到每个数据点的最近邻居，在这种情况下，评估它们是否属于同一个类。 这是一个非常简单的分类任务。我们可以通过考虑使用单位长度测量并计算其中的数据点数来简化讨论。我们在图3.2中描述了这种情况，其中我们显示了十个数据实例，用三角形，空心圆和加号表示，分别为1,2和3维。

Let us count the number of data points that fall within one unit length measurement.

让我们计算一个单位长度内的数据点数

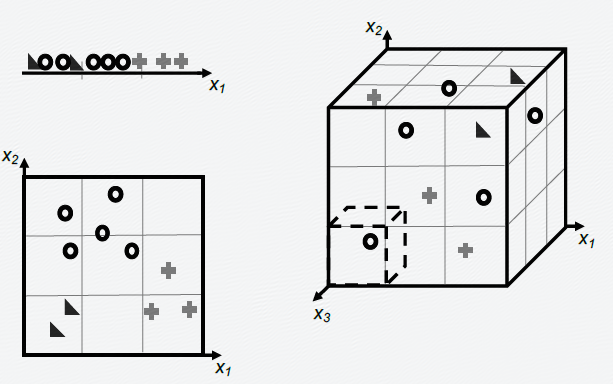


Figure 3.2: The curse of dimensionality. Ten data instances placed in spaces of increased dimensionality, from 1 dimension to 3. Sparsity increases with the number of dimensions.

图3.2：维数的诅咒。 十个数据实例放置在维度增加的空间中，从1维到3。稀疏度随着维数的增加而增加。

In a one-dimensional space, the unit measurement is given by a line. In the example shown in Figure 3.2 we can see a space with three unit intervals, so the sample density is . In other words we have about 3.333... data points per interval and thus finding a near neighbour and assessing its class is certainly possible.

在一维空间中，单位测量由线给出。在图3.2所示的示例中，我们可以看到一个具有三个单位间隔的空间，因此样本密度为10/3。 换句话说，我们每个区间有大约3.333 ......数据点，因此找到一个近邻并且评估它所属的类别肯定是可能的。

In the two-dimensional case, we would have to search an area of 3 3 = 9 unit squares and in this case the sample density is , or an average of 1.111... data points per square. In this case it is less likely to find a neighbour of a given data instance within the unit square where the data point of interest is located. Finally, in the 3D case we have a feature space of 3 3 3 = 27 unit cubes and the sample density is 0.111... on average. In this case our search for a neighbour within a cubic square becomes more difficult as most of the feature space is effectively empty. We say that the space is sparse.

在二维情况下，我们必须搜索3×3 = 9个单位正方形的区域，在这种情况下，样本密度为10/9，或平均每平方1.111 ...个数据点。 在这种情况下，不太可能在感兴趣的数据点所在的单位平方内找到给定数据实例的邻居。最后，在3D情况下，我们的特征空间为3×3×3 = 27个单位立方体，样本平均密度为10/27= 0.111 ... 在这种情况下，我们在立方体内搜索邻居变得更加困难，因为大多数特征空间实际上是空的。我们说这个空间很稀疏。

As the number of dimensions increases, the distance between data points becomes larger and larger. We end up with very sparse spaces.

随着纬度数量的增加，数据点之间的距离变得越来越大。 我们最终得到了非常稀疏的空间。

It is easy to see that as we keep adding features (dimensions), our space becomes sparser. It is due to this sparsity that we require a larger and larger number of data instances when dealing with higher-dimensions. For example, if we were interested in carrying out our classification task using one feature with a range of values between 0 and 1, and we wanted our dataset to cover 25% of this range, we would need 25% of the complete population. By adding another feature we would require 50% of the population (= 0.25) and with three features we would need 63% of the population (0.25).

很容易看出，当我们不断添加特征（维度），我们的空间变得更加稀疏。正是由于这种稀疏性，我们在处理更高维度时需要越来越多的数据实例。例如，如果我们有兴趣使用一个介于0和1之间的值的特征来执行我们的分类任务，同时希望我们的数据集覆盖此范围的25％，那么我们将需要完整总体的25％。通过添加另一个特征，我们需要总体的50％（(= 0.25)），有三个特征时，我们需要总体的63％。（0.25）。

It is due to this sparsity that we require a larger and larger number of data instances.

正是由于这种稀疏性，我们需要越来越多的数据实例。

We may think that simply adding more data is the appropriate solution to dispel the curse of dimensionality. However, as we saw above, it is important to remember that the number of data instances needed grows exponentially with the number of dimensions. In practice we very rarely

have access to an infinite amount of data. Furthermore, using too many features actually results in overfitting.

我们可能认为简单地添加更多数据是消除维度诅咒的合适的解决方案。但是，正如我们上面所看到的，重要的是要记住所需的数据实例数量随着维度的数量呈指数增长。在实践中我们很少可以访问无限量的数据。此外，使用太多特征实际上会导致过度拟合。

The number of data instances needed grows exponentially with the number of dimensions.

所需的数据实例数量随着维度的数量呈指数增长

Following our classification example above with three categories (triangle, open circle and plus sign), it is much easier to find various ways to classify the data instances into separate classes when considering higher number of features. This is a great thing to start with, but we must be careful to avoid overfitting, or even getting carried away with false patterns. Furthermore, the sparsity in a higher dimensional space is not homogeneous, and it turns out that the space around the origin is much more sparse than in the corners of the hyper-space.

按照上面的分类示例，有三个类别（三角形，空心圆和加号），在考虑更多数量的特征时，更容易找到将数据实例分类为单独类的各种方法。 这是一个很好的开始，但我们必须小心避免过度拟合，或者甚至被错误的模式带偏了方向。 此外，较高维空间中的稀疏性不均匀，并且发现原点周围的空间比超空间角落的更稀疏。

Sparsity in higher dimensional space is not homogeneous.

高维空间中的稀疏性不均匀。

In order to understand this issue let us consider a 2D space to start with. The mean of the feature space is at the centre of a unit square. If we wanted to search the space within one unit distance from the centre, we would be searching in the area given by a circle of unit radius (circumscribed by the square). Any data instances that fall outside the area of this circle turn out to be closer to the edges of the square and become more difficult to classify as their feature values are more distant to those in the centre (the mean). Let us now consider this situation in N dimensions:

为了理解这个问题，让我们考虑一下2D空间。 特征空间的平均值位于单位正方形的中心。如果我们想要在距离中心一个单位距离内搜索空间，我们将在由单位半径圆（由正方形限定）给出的区域内搜索。 落在该圆的区域之外的任何数据实例都变得更接近正方形的边缘并且变得更难以分类，因为它们的特征值与中心的那些（平均值）更远。 现在让我们考虑N维的这种情况：

Data instances further away from the centre of a unit circle are more difficult to classify. This is aggravated in higher dimensions.

远离单位圆中心的数据实例更难以分类。 这在更高的维度上更加严重。

* The volume of a unit hypercube of N dimensions is = 1

N维单位超立方体的体积为 = 1

* The volume of a unit hypersphere of N dimensions7 is:

V(N) = , (3.7)

where the radius r = 1 for the unit hypersphere and is the gamma function.

N维的单位超球的体积是：

V(N) = , (3.7)

其中半径r = 1表示单位超球面，是伽马函数。

7 DLMF (2015). NIST Digital Library of Mathematical Functions. http://dlmf.nist.gov/, Release 1.0.10 of 2015-08-07

7 DLMF（2015年）。 NIST数学函数数字图书馆。 http://dlmf.nist.gov/,2015-08-07版本1.0.10

In Figure 3.3 we show how the volume of the hypersphere tends to zero as the dimensionality N increases. Nonetheless, the volume of the hypercube remains fixed. This means that in higher-dimensional spaces, most of the data is actually in the corners of the hypercube that defines

the feature space, making the classification task more difficult to achieve.

在图3.3中，我们展示了随着维数N增加，超球体的体积趋于零。尽管如此，超立方体的体积仍然是固定的。这意味着在高维空间中，大多数数据实际上位于定义的超立方体的角落特征空间，使分类任务更难实现。

In higher-dimensional spaces, most of the data is in the corners of the hypercube.

在高维空间中，大多数数据位于超立方体的角落。

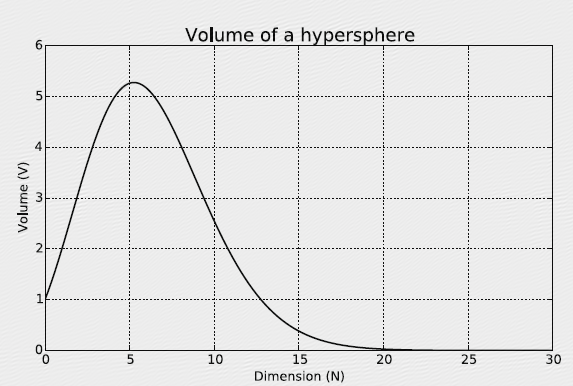


Figure 3.3: Volume of a hypersphere as a function of the dimensionality N. As the number of dimensions increases, the volume of the hypersphere tends to zero.

图3.3：超球面的体积与维数N的函数关系。随着维数的增加，超球体的体积趋于零。

The curse of dimensionality is a very real thing and there is not much that can be done to eliminate it completely. However, it is possible to minimise it, for instance by carefully checking that low-dimensional methods are effective in higher dimensions. Avoiding the curse of dimensionality can be done by increasing the amount of data, but even before going down that route it is worth considering if the features used are indeed a suitable collection.

维度的诅咒真实存在，并没有太多可以完全消除它的方法， 但是，可以将其最小化，例如通过仔细检查低维方法在更高维度上是否有效。避免维度诅咒可以通过增加数据量来完成，但即便考虑用这种方法，还是要考虑所使用的特征数据是否确实是合适的数据集合。

Eliminating the curse of dimensionality completely is not an easy task.

完全消除维度的诅咒并非易事。

In that respect, apart from a careful feature selection process, we can also reduce the dimensionality of the problem by transforming the data from a higher-dimensional space into a space with fewer dimensions as it is the case with Principal Component Analysis (PCA). We will discuss this technique in Chapter 8. As for avoiding overfitting, in Section 3.12 we will discuss the ideas behind cross-validation. But first we need to make a stop to talk about Scikit-learn.

在这方面，除了仔细的选择特征外，我们还可以通过将数据从较高维空间转换为较少维度的空间来减少问题的维数，如主成分分析（PCA）的情况。我们将在第8章讨论这种技术。至于避免过度拟合，在3.12节我们将讨论交叉验证背后的想法。 但首先我们需要停下来谈谈Scikit-learn。

Dimensionality reduction aims to transform data from a higherdimensional space into one with fewer dimensions.

降维旨在将数据从较高维空间转换为具有较小维度的空间。

See Section 8.2 for a discussion on Principal Component Analysis (PCA).

有关主成分分析（PCA）的讨论，请参见第8.2节。

*3*.*10* Scikit-Learn is our Friend

3.10 Scikit-Learn是我们的朋友

The broad aim of machine learning, as we have seen, is the development and application of algorithms that can learn from data. This goal can be accomplished in a variety of ways and in recent times, with the advancement of computer power, together with the availability of useful *learning within our reach. software tools this is a goal that can be within the reach of many of us.*

正如我们所看到的，机器学习的广泛目标是开发和应用从数据中学习的算法。最近随着计算机能力的提高，以及我们掌握的有用的学习越来越方便被人使用，这个目标可以通过各种方式实现，软件工具这是我们许多人都可以接触到的目标。

The advancement of computer power and the availability of useful software tools put machine learning within our reach.

计算机能力的提高和有用的软件工具的可用性使机器学习成为可能。

Among the many programming languages and tools that are at hand, in this book we have chosen to use Python. Within the libraries and packages at our disposal we concentrate mainly on Scikit-learn as it contains a wide-range of machine learning algorithms. Scikit-learn builds upon libraries that we have already seen in Chapter 2 such as NumPy, SciPy and matplotlib. Scikit-learn is able to interact with Pandas dataframes and other objects in Python. It is worth mentioning that the focus of Scikit-learn is the modelling part of the data science workflow, rather than the manipulation of data.

在本书中在众多编程语言和工具中我们选择使用Python。 在我们可以使用的库和包中，我们主要关注Scikit-learn，因为它包含广泛的机器学习算法。 Scikit-lear构建于我们在第2章中已经看过的库，例如NumPy，SciPy和matplotlib。 Scikit-learn能够与Pandas数据帧和Python中的其他对象进行交互。 值得一提的是，Scikit-learn的重点是数据科学工作流的建模部分，而不是数据的操纵。

Scikit-learn contains a wide-range of machine learning algorithms and builds upon libraries that we have discussed.

Scikit-learn包含各种机器学习算法，并以我们讨论过的库为基础。

Scikit-learn enables us to run popular models and techniques such as:

* Regression
* Clustering
* Feature selection
* Dimensionality reduction
* Classification
* Cross-validation
* etc.

Scikit-learn使我们能够运行流行的模型和技术，例如：

* 回归
* 聚类
* 特征选择
* 降维
* 分类
* 交叉验证
* 等等

Some popular models and techniques available in Scikit- learn.

Scikit-learn中提供了一些流行的模型和技术。

In the following chapters we shall have opportunity to explore some of the implementations of these models. Scikit-learn also comes packaged with some test datasets that can be used for investigating the usage of the various models.

在接下来的章节中，我们将有机会探索这些模型的一些实现。Scikit-learn还附带了一些测试数据集，可用于调研各种模型的用法。

Scikit-learn comes packaged with some test datasets ready to play with.

Scikit-learn随附了一些可供使用的测试数据集。

Given that we will be using this library extensively, it is worth mentioning the typical data representation expected by the models in Scikit-learn. As explained in Section 2.4, matrices and vectors are a favourable data representation, in particular for mathematical calculations and manipulations. Scikit-learn bears that in mind and it expects data to be represented in the form of two-dimensional numeric matrices with M data instances (rows) and N distinct features (columns).

鉴于我们将广泛使用该库，很值得在这里谈一下Scikit-learn中的模型所期望的典型数据表示。 如2.4节所述，矩阵和向量是一种受欢迎的数据表示，特别是对于数学计算和操作。 Scikit-learn继承了这一点，并且它期望数据以具有M个数据实例（行）和N个不同特征（列）的二维数字矩阵的形式表示。

Scikit-learn expects data in a matrix representation.

The matrices can be NumPy or SciPy arrays for instance.

Scikit-learn期望以矩阵形式的表示数据。  
例如，矩阵可以是NumPy或SciPy数组。

A canonical example in data science and analytics is the Iris dataset, and as you can imagine it is included with Scikitlearn. The set was first used by Ronald Fisher8, and has a total of M = 150 samples of three species of iris flowers: Setosa (50), Virginica (50) and Versicolor (50). For each specimen we are provided with N = 4 feature measures (in centimetres): sepal length, sepal width, petal length and petal width.

数据科学和分析中的典型示例是Iris数据集，您可以想象它包含在Scikitlearn中。 该数据集最初由Ronald Fisher使用，并且总共有M = 150个样品 ，其中包含三种鸢尾花：Setosa（50），Virginica（50）和Versicolor（50）。 对于每个样本，我们提供N = 4个特征量度（以厘米为单位）：萼片长度，萼片宽度，花瓣长度和花瓣宽度。

8 Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. Annals of Eugenics *7*(2), 179–188

8 Fisher，R。A.（1936）。 在分类学问题中使用多次测量。 优生志年鉴7（2），179-188

We can load this dataset by importing it directly from Scikitlearn as follows:

我们可以通过直接从Scikitlearn导入来加载此数据集，如下所示：

from sklearn.datasets import load\_iris

iris = load\_iris()

With the two lines of code above we have imported the dataset and loaded it into an object named iris. We can now inspect the matrix that contains the feature data. For instance the size of the matrix must be M = 150 and N = 4, and we can verify this with the shape method as follows:

通过上面的两行代码，我们导入了数据集并将其加载到名为iris的对象中。我们现在可以检查包含特征数据的矩阵。 例如，矩阵的大小必须是M = 150和N = 4，我们可以使用shape方法验证这一点，如下所示：

The Iris dataset is represented by a 150 4 matrix.

Iris数据集由150×4矩阵表示。

> iris.data.shape

(150L, 4L)

Let us see the first six data instances:

让我们先看看这六个数据实例：

> iris.data[0:6,0:4]

array([[ 5.1, 3.5, 1.4, 0.2],

[ 4.9, 3. , 1.4, 0.2],

[ 4.7, 3.2, 1.3, 0.2],

[ 4.6, 3.1, 1.5, 0.2],

[ 5. , 3.6, 1.4, 0.2],

[ 5.4, 3.9, 1.7, 0.4]]) [ 54, 3.9, 1.7, 0.4]])

We can use slicing and dicing to see the contents of the dataset.

我们可以使用切片和切块来查看数据集的内容。

The dataset also contains the class to which each data instance belongs, i.e. setosa, versicolor or virginica. The information can be obtained by looking at the target\_names of the iris object:

数据集还包含每个数据实例所属的类，即setosa，versicolor或virginica。 可以通过查看iris对象的target\_names来获取信息：

> print(iris.target\_names)

[’setosa’ ’versicolor’ ’virginica’]

The class names of the iris flowers.

鸢尾花的类名。

Remember that Scikit-learn expects data in numeric format, so using strings to represent the classes is not suitable. Instead, each of the three categories has been encoded with numbers corresponding to the position of the names in the list above:

请记住，Scikit-learn需要数字格式的数据，因此使用字符串来表示类是不合适的。相反，三个类别中的每一个都使用与上面列表中名称的位置相对应的数字进行编码：

> iris.target[0:151:50]

array([0, 1, 2])

We will come back to this dataset every so often in the rest of the book.

我们将在本书的其余部分经常回到这个数据集。

*3*.*11* Training and Testing

3.11训练和测试

The models we use for gaining insight into our business or research questions require data. With data as a resource, we need to be mindful of how, when and where it is used. Let us imagine that we are tasked with running a classification model based on the Iris dataset we saw in the previous section. We can consider using all 150 records for this purpose and base our model on the 4 features provided.

我们用于深入了解业务或研究问题的模型需要数据。以数据为资源，我们需要注意它的使用方式，时间和地点。假设我们的任务是运行我们在上一节中看到的Iris数据集的分类模型。我们可以考虑基于所提供的4个特征值，使用全部150条记录，来建立我们的模型。

We need to be mindful of how, when and where data is used.

我们需要注意数据的使用方式，时间和地点。

We carry out our modelling and the result can be used to classify any new iris flower we encounter based on the 4 measurements (features) used. However, how do we know how well (or how badly) our model performs? We would have to wait until we get new data not seen by the model. This may be an issue as we do not necessarily have a Mark Watney9 - Martian botanist extraordinaire - or any other (fictional or real) botanist at hand to obtain new iris specimens, whether they are grown on Earth, Mars or elsewhere.

我们进行的模型，根据所使用的4种测量（特征），建模结果可以对我们遇到的任何新鸢尾花进行分类。 但是，我们如何知道我们的模型表现得有多好（或多么糟糕）？ 我们必须等到获得在模型训练阶段没有得到的新数据的出现。这可能是一个问题，因为我们不一定有Mark Watney 9 - 火星植物学家 - 或任何其他（虚构的或真实的）植物学家手头获得新的iris标本，无论它们是生长在地球，火星还是其他地方。

We are interested in the performance against unseen data

我们对未见数据的表现感兴趣。.

9Weir, A. (2014). The Martian: A Novel. Crown/Archetype

9Weir，A。（2014）。 火星人：一部小说。皇冠/原型

What is more, we must remember that we build a model because we are interested in using it effectively. This means that we should care about its performance with new unseen data and therefore a way to assess this is with error rates. If we use our entire dataset to train the model, determining our training error becomes impossible. Not to mention the fact that the model will be built to account for the training data only and may thus overfit it, i.e. it will not generalize to new data instances.

更重要的是，我们必须记住，我们建立一个模型，因为我们有兴趣有效地使用它。这意味着我们应该关注其新的看不见的数据的性能，因此评估这种情况的方法是错误率。如果我们使用整个数据集来训练模型，则无法确定我们的训练误差。更不用说该模型将被构建为仅考虑训练数据并因此可能过度拟合，即它不会推广到新的数据实例。

We have discussed some causes of overfitting earlier in this chapter.

我们在本章前面讨论了过度拟合的一些原因。

One way to tackle this problem is to prepare two independent datasets from the original one:

* **Training set**: This is the data that the model will see and it is used to determine the parameters of the model
* **Testing set**: We can think of this as “new” data. The model has not encountered it yet and it will enable us to measure the performance of the model built with the training set .

解决此问题的一种方法是从原始数据集中准备两个独立的数据集：

* 训练集：这是模型将看到的数据,它用于确定模型的参数
* 测试集：我们可以将其视为“新”数据。该模型尚未遇到它，它将使我们能够测量使用训练集构建的模型的性能。

In some cases instead of partitioning the data into two sets, it is divided into three. The third component is called the **validation set** and it is used for tuning the model. All three parts need to be representative of the data that will be used with the model. It is important to clarify that the testing data must not be used for training the model, and that the validation set must not be applied for testing.

在某些情况下，它不是将数据分成两组，而是分为三组。第三组称为验证集，用于调整模型。 所有这三个部分都需要代表将与模型一起使用的数据。重要的是要澄清测试数据不得用于训练模型，并且验证集不得用于测试。

The testing set is sometimes also called the holding set.

Sometimes a third partition is used to validate the model.

The testing set must never be used for training the model.

测试集有时也称为保留集。  
有时，第三个分区用于验证模型。  
绝不能将测试集用于训练模型。

We can see a schematic representation of this situation in Figure 3.4. Notice that the use of the training set in the modelling provides us with a measure of the training error. In turn, when applying the testing dataset we can get a measure of how well the model performs, i.e. we obtain a measure of the generalisation error. Finally, when using unseen new data we are getting a measure of the so-called out-of-sample error.

A measure of generalisation error is obtained from the testing set.

我们可以在图3.4中看到这种情况的示意图。 请注意，在建模中使用的训练集为我们提供了训练误差的度量。反过来，当应用测试数据集时，我们可以测量模型的执行情况，即我们获得了泛化误差的度量。 最后，当使用看不见的新数据时，我们获得所谓的样本外错误的度量。  
从测试集获得泛化误差的度量。

From the discussion we had about the curse of dimensionality, we know that the more data instances we have for modelling the better. On the other hand, the more test data we have the more accurate will be our error estimate. A common way to split the data set into training and testing is 80/20% , and typically between one third and one tenth of the data is held out for testing. Other combinations are possible such as 70/30% for example.

从我们对维度诅咒的讨论中，我们知道我们用于建模的数据实例越多越好。另一方面，我们得到的测试数据越多，我们的误差估计就越准确。将数据集拆分为训练和测试的常用方法是80/20％，通常将三分之一到十分之一的数据用于测试。 其他组合也是可能的，例如70/30％。

In cases when a straightforward splitting leaves us with datasets that may not be representative of the data, we can consider using a stratification method, for instance in situations where a particular class in the data is not represented in the training set. Stratification will aim at splitting the dataset so that each class is represented in both the testing and training sets.

There are a number of considerations that must be made when splitting a given dataset.

如果直接拆分为我们留下可能无法代表数据的数据集，我们可以考虑使用分层方法，例如在数据中的特定类未在训练集中表示的情况下。分层将旨在分割数据集，以便在测试和训练集中表示每个类。  
拆分给定数据集时必须考虑许多因素。

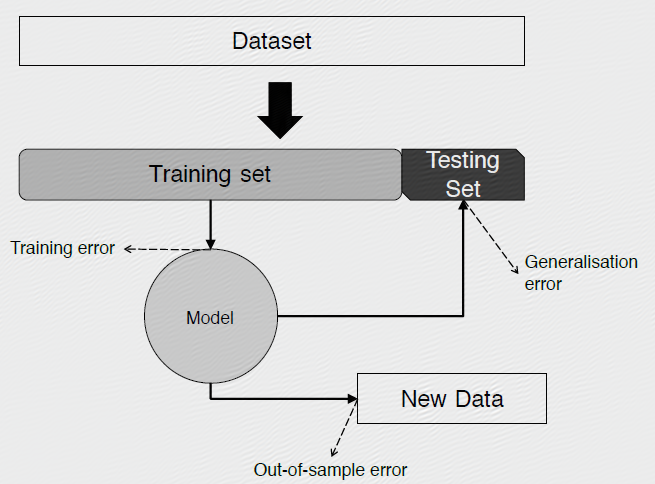


Figure 3.4: A dataset is split into training and testing sets. The training set is used in the modelling phase and the testing set is held for validating the model.

图3.4：数据集分为训练集和测试集。 训练集用于建模阶段，测试集用于验证模型。

Splitting the data can be done by partitioning the set into two (or three) sections without altering the order of the data. However, in cases where the dataset has been ordered, this naïve procedure may lead to unbalanced training and testing sets. A better approach is to randomly select the data instances that will form each of the two partitions. It makes sense thus to carry out the randomisation as part of your workflow, regardless of the ordering of the data.

Splitting the dataset into training and testing is better done with random selection.

拆分数据可以通过将集合划分为两个（或三个）部分而不改变数据的顺序来完成。但是，如果订好了数据集，这种朴素的程序可能会导致不培训集和测试集的不平衡。更好的方法是随机选择每个数据集的数据实例。因此，无论数据的顺序如何，都可以将随机化作为工作流程的一部分进行。

Sckit-learn is able to assist with the splitting of the dataset into random train and test subsets with the model\_selection module and its train\_test\_split function following this syntax:

For versions of Scikit-learn earlier than 0.18 this functionality was in the cross\_validation module

Sckit-learn能够使用model\_selection模块及其train\_test\_split函数按照以下语法协助将数据集拆分为随机序列和测试子集：

对于早于0.18的Scikit-learn版本，此功能位于cross\_validation模块中

model\_selection.train\_test\_split( \

\*arrays, \

test\_size, \

train\_size, \

random\_state)

Training and testing datasets can be obtained with the train\_test\_split function.

可以使用train\_test\_split函数获取训练和测试数据集。

where \*arrays are the datasets that will be split, test\_size accepts a value between 0 and 1 representing the proportion of the dataset to be included in the testing set, train\_size can be left out, and if so its value is automatically set to be the complement of the test size. Finally, random\_state initialises the random number generator for sampling.

The train size parameter is optional.

其中\*数组是要拆分的数据集，test\_size接受0到1之间的值，表示要包含在测试集中的数据集的比例，train\_size可以省略，如果省略，其值会自动设置并填充。最后，random\_state初始化随机数生成器以进行采样。  
train大小参数是可选的。

We can see how this can be applied to the case of the Iris dataset. We would like to store the features in arrays called X\_train and X\_test and the corresponding targets in Y\_train and Y\_test. We would like to hold 20% of the data for testing:

Let us split the Iris dataset with Scikit-learn.

我们可以看到这如何应用于Iris数据集的情况。我们希望将这些特征存储在名为X\_train和X\_test的数组中，将对应的目标存储在Y\_train和Y\_test中。 我们希望持有20％的数据用于测试：  
让我们用Scikit-learn分割Iris数据集。

from sklearn import model\_selection

X\_train, X\_test,\

Y\_train, Y\_test =\

model\_selection.train\_test\_split(\

iris.data, iris.target,\

test\_size=0.2, random\_state=0)

We are using the pythonic style of multiple assignation.

我们正在使用pythonic风格的多重分配。

We can check the sizes of the newly created sets:

我们可以检查新创建的集合的大小：

> print(X\_train.shape, Y\_train.shape)

((120, 4) (120,))

> print(X\_test.shape, Y\_test.shape)

((30, 4) (30,))

We can verify the size of the matrices we have after the split.

我们可以验证拆分后矩阵的大小。

The modelling task will then be done with the training dataset X\_train and Y\_train. We can assess how well our model works by measuring the error against the testing dataset X\_test and Y\_test.

然后，将使用训练数据集X\_train和Y\_train完成建模任务。我们可以通过测量测试数据集X\_test和Y\_test的误差来评估模型的工作情况。

*3*.*12* Cross-Validation

3.12交叉验证

Since we are interested in making accurate and useful predictions, we need to ensure that any models we create generalise well to unseen data. We have discussed how a training and testing dataset split can help us with this goal. Nonetheless, the parameters that we obtain with the use of a single training dataset may end up reflecting the particular way in which the data split was performed.

In other words, we avoid overfitting.

由于我们有兴趣进行准确和有用的预测，我们需要确保我们创建的任何模型都能很好地概括看不见的数据。 我们已经讨论了训练和测试数据集的拆分如何帮助我们实现这一目标。尽管如此，我们使用单个训练数据集获得的参数可能最终反映了特定方式，这种方式是数据拆分所导致的结果。

换句话说，我们避免过度拟合。

The solution to this problem is straightforward: We can use statistical sampling to get more accurate measurements. This process is usually referred to as cross-validation. Cross-validation improves statistical efficiency by performing repeated splitting of data into training and validation sets, and re-performing model training and evaluation every time. The aim of cross-validation is to use a dataset to validate the model during the training phase.

Cross-validation improves statistical efficiency by performing repeated splitting of data.

这个问题的解决方案很简单：我们可以使用统计抽样来获得更准确的测量结果。此过程通常称为交叉验证。交叉验证通过将数据重复分成训练和验证集，并且每次都重新执行模型训练和评估来提高统计效率。交叉验证的目的是使用数据集在训练阶段验证模型。  
交叉验证通过重复分割数据来提高统计效率。

Let us see why this helps by considering the following scenario: We have carried out an initial training/testing split. The training set is used for modelling, and we perform the evaluation with the testing set. Imagine now that a different random state had been used to split the data. We would expect the model to see different data points during training. The generalisation error obtained with the second split would be different from the first. We can reduce variability by repeating this scenario over and over again, using different partitions and averaging the validation results over the rounds. Moreover, cross-validation is a great tool when we have a large, but limited amount of data points. Let us see how this can be done with k-fold cross-validation.

Cross-validation can help reduce variability by repeated use of training and testing splits.

让我们看看交叉验证为什么能够帮助我们，考虑以下情况：我们已经进行了初步的训练/测试拆分。训练集用于建模，我们使用测试集执行评估。想象一下，现在已经使用了一种不同的随机状态来分割数据。 我们希望模型在训练期间看到不同的数据点。 使用第二次拆分获得的泛化误差将与第一次划分时不同。我们可以通过一遍又一遍地重复这种情况来减少可变性，使用不同的分区并对轮次中的验证结果进行平均。此外，当我们拥有大量但有限的数据点时，交叉验证是一个很好的工具。让我们看看如何通过k-折交叉验证来完成。

交叉验证可以通过重复使用训练和测试拆分来帮助减少可变性

*3*.*12*.*1* k-fold Cross-Validation

A common cross-validation technique is the k-fold procedure: The original data is divided into k equal sets. From the k subsets, a single partition is kept for validating the model, and k 1 subsets are used for training. The process is then repeated k times, using one by one each of the k subsets for validation. We will therefore have a total of k trained models. The results of each of the folds can be combined, for instance by averaging, to obtain a single estimation of the out-of-sample error. We can see a schematic representation of the k-fold cross-validation

procedure for the case k = 4 in Figure 3.5.

We need to split the dataset into k partitions.

3.12.1 k-折交叉验证  
常见的交叉验证技术是k-折程序：原始数据被分成k个相等的集合。从k个子集中，保留单个分区用于验证模型，另外的k-1个子集用于训练。然后将该过程重复k次，逐个使用k个子集中的每个子集进行验证。 因此，我们将共有k个训练模型。 可以组合每个折叠的结果，例如通过平均，以获得样本外误差的单个估计。我们可以看到k折交叉验证的示意图，图3.5中k = 4的过程。  
我们需要将数据集拆分为k个分区。

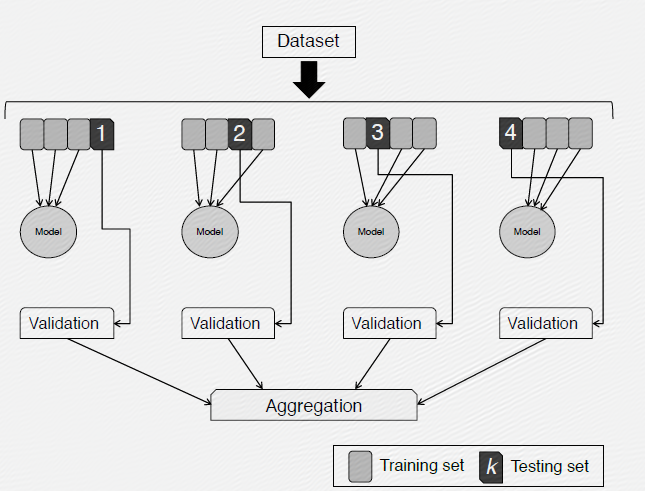


Figure 3.5: For k = 4, we split the original dataset into 4 and use each of the partitions in turn as the testing set. The result of each fold is aggregated (averaged) in the final stage.

图3.5：对于k = 4，我们将原始数据集拆分为4个，并依次使用每个分区作为测试集。 每个折叠的结果在最后阶段汇总（平均）。

There are other procedures of cross-validation such as Leave-One-Out, (LOO) where one single data sample is taken for validation and the rest of the M1 data points are used for training. If we were to remove p samples from the complete set we would be implementing the so-called

Leave-P-Out (LPO) procedure.

Other cross-validation procedures are also available.

还有其他交叉验证程序，例如留一法Leave-One-Out（LOO），其中一个数据样本用于验证，其余的M-1数据点用于训练。如果我们要从整个数据集中删除p个样本，我们将实现所谓的Leave-P-Out（LPO）程序。

还可以使用其他交叉验证程序。

Scikit-learn enables us to carry out cross-validation splits with the aid of functions such as KFold, LeaveOneOut and LeavePOut. The idea behind these functions is to generate k lists of indices that can be used to select the appropriate data points for each fold. For instance, we can create 10 folds for the Iris dataset as follows:

Cross-validation with the k-fold method is implemented with the KFold function.

Scikit-learn使我们能够借助KFold，LeaveOneOut和LeavePOut等功能进行交叉验证分割。 这些函数背后的想法是生成k个索引列表，可用于为每个折叠选择适当的数据点。例如，我们可以为Iris数据集创建10个折叠，如下所示：  
使用K-fold函数实现k折方法的交叉验证。

kfindex = cross\_validation.KFold(n\_splits=10,\

shuffle=True,\

random\_state=0)

for train\_ix, test\_ix in kfindex.split(iris.data):

X\_train, X\_test =\

iris.data[train\_ix], iris.data[test\_ix]

Y\_train, Y\_test =\

iris.target[train\_ix], iris.target[test\_ix]

KFold effectively maintains an index that keeps track of the data instances that go into each of the training and testing sets.

KFold有效地维护一个索引，该索引跟踪进入每个训练和测试集的数据实例。

Cross-validation is a useful and straightforward way to get a more accurate estimate of the out-of-sample error, and at the same time a more efficient use of data than a single training/testing split. This is because each record in the dataset is used in both training and validating.

Cross-validation provides a more efficient use of our data.

交叉验证是一种有用且直接的方法，可以更准确地估计样本外错误，同时比单个训练/测试拆分可以更有效地使用数据。这是因为数据集中的每条记录都用于训练和验证。  
交叉验证可以更有效地使用我们的数据。

Cross-validation can also be useful in feature and model selection procedures. For example, it can be used for tuning the regularisation parameter introduced in Section 3.7: We split out training data and train a model for a fixed value of. We can then test it on the remaining subsets and repeat this procedure while varying . Finally, we select the best that minimises our measure of error.

Cross-validation can be very useful in tuning the hyperparameter in regularisation.

交叉验证在特征和模型选择过程中也很有用。例如，它可以用于调整3.7节中引入的正则化参数λ：我们分出训练数据并训练固定值为λ的模型。然后我们可以在剩余的子集上测试它，并在改变λ的同时重复这个过程。最后，我们选择最小的λ来最小化我们的误差测量。

Despite these advantages we must bear in mind that cross-validation increases the computational work that needs to be done, and if overused, can lead to overfitting. I urge you to use cross-validation given the advantages mentioned above. In a true case of “do as I say, not as I do”, there will be examples in the book where I will not perform the cross-validation step as part of the explanations of the various models we will tackle. However, it is worth remembering that cross-validation is an integral part of the modelling part of the data science workflow.

Cross-validation increased the computational work that needs to be done.

尽管有这些优点，但我们必须记住，交叉验证会增加需要完成的计算工作，如果过度使用，可能会导致过度拟合。 鉴于上述优点，我建议您使用交叉验证。 作为“像我说的那样，不像我一样”的真实案例，在本书中的一些例子中，我们不会把交叉验证步骤作为各种模型要解决的问题来介绍。 但是，值得记住的是，交叉验证是数据科学工作流程建模部分的一个组成部分。

交叉验证增加了需要完成的计算工作。

*3*.*13* Summary

I would like to finish this chapter with a few thoughts that we must always keep in mind during our work in data science and analytics:

• If the machine can learn, so can we!

• Machine learning and data science are not not focused on causality, but in prediction, insight and knowledge

• All models are wrong: There is no such thing as a perfect model, just good enough ones

• The data science and analytics workflow is a balancing act:

**–** Bias v variance

**–** Complexity v simplicity

**–** Overfitting v regularisation

**–** More data v cunning algorithm and resources

**–** Accuracy v insight

**–** Effort saved v computational cost

**–** Jackalopes v unicorns

• Having a lot of data (even big data) is good, and being able to construct models is a great skill. Nonetheless, they are not magic wands

• Beware the curse of dimensionality

• Splitting our data into training and testing not only is good practice but a must

• An important part of the modelling phase in data science is the use of cross validation. Remember that the testing set must never be used for training

3.13摘要

在本章结束时，我们在数据科学和分析工作中必须始终牢记这些想法：

•如果机器可以学习，我们也可以！

•机器学习和数据科学不是关注因果关系，而是关注预测，洞察力和知识

•所有模型都是错误的：没有完美的模型，只有足够好的模型

•数据科学和分析工作流程是一种平衡行为：

- 偏差 v 变化

- 复杂性 v 简单性

- 过度拟合v正则化

- 更多数据v cunning算法和资源

- 准确性 v 洞察力

- 节省的努力 v 计算成本

- Jackalopes v unicorns

•拥有大量数据（甚至是大数据）是好的，能够构建模型是一项很好的技能。尽管如此，它们并不是魔杖

•注意维度的诅咒

•将我们的数据分成训练和测试不仅是良好的做法，而且是必须的

•数据科学建模阶段的一个重要部分是使用交叉验证。请记住，测试数据绝不能用于训练

*4*

The Relationship Conundrum: Regression

关系难题：回归

Regression analysis is one of the most widely used tools in statistical analysis. Most of us may have come across it at some point either by employing it or interpreting it. It is a powerful technique due to both its ease of calculation and simplicity of assumptions. However, it is due to these attributes that sometimes regression is misapplied or misinterpreted.

Regression is a well-known and widely used machine learning tool.

回归分析是统计分析中使用最广泛的工具之一。我们大多数人可能在某个时候通过使用它或解释它已经和它有过交集。 由于其易于计算和假设的简单性，使它成为一个强大的技术。 但是，也正是由于这些特征，有时回归被误用或引起误解。

回归是一种众所周知且广泛使用的机器学习工具

In this chapter we will cover the main aspects of regression analysis starting up with a motivation to the problem and covering both linear and polynomial regression techniques. Similarly, we will see how feature selection can be done with the help of appropriate regularisation techniques.

We will cover here some of its most important aspects.

在本章中，我们将介绍回归分析的主要方面，从问题的动机开始，涵盖线性和多项式回归技术。同样，我们将看到如何在恰当的正则化技术的帮助下完成特征选择。

我们将在这里介绍一些最重要的方面。

*4*.*1* Relationships between Variables: Regression

4.1变量之间的关系：回归

Consider a situation where you are interested to determine the association between two (or more) pieces of information; say for example the relation of the height of a child compared to that of her parents, or ice cream sales and temperature, or even the body mass of an animal and the mass of its brain. We can collect data for these events and use it for constructing a model that enables us to explore the relationship between the variables in question. Ultimately, our goal is to use our model to predict the outcome of the variable of interest given the values of the other variable(s).

All these are actual, well documented examples of regression usage.

考虑一种情况，您想确定两条（或更多条）信息之间的关联; 例如，孩子的身高与父母身高的关系，或者冰淇淋的销售和温度的关系，甚至动物的体重和大脑的重量之间的关系。我们可以收集这些事件的数据并构建模型，模型使我们能够探索相关变量之间的关系。最终，在给定其他变量的情况下，我们的目标是使用我们的模型来预测感兴趣的变量。

所有这些都是回归使用的实际且有据可查的例子。

We usually call the quantity of interest the **response** or **dependent** variable and denote it with the variable y. The other quantities are called **predictors**, **explanatory** or **independent** variables and denote them as x. Intuitively, we know that two quantities are correlated if there is a relationship between the two variables, i.e. the value of one tells us something about the value of the other one.

The dependent variable is the quantity we want to predict.

The independent variables are also called regressors.

我们通常将响应量或因变量称为感兴趣的量，并用变量y表示它。 其他变量被称为预测变量，解释变量或自变量，并将它们表示为x。直观的讲，我们知道如果两个变量之间存在关系，则两个量是相关的，即其中一个的值会告诉我们另一个的值。

因变量是我们想要预测的数量。  
自变量也称为回归量。

In a correlation analysis we estimate a value bounded between -1 and 1 and we call it the correlation coefficient. This coefficient tells us the strength of the linear association between the two variables. If the two quantities vary intandem (if one increases/decreases, the other one does too) the correlation coefficient is positive, whereas it is negative when the two quantities vary out of sync (if one decreases, the other one increases).

The correlation coefficient measures the degree of linear relationship among variables.

在相关分析中，我们估计一个在-1和1之间的值，我们称之为相关系数。该系数告诉我们两个变量之间线性关联的强度。如果两个量在一定程度上变化（如果一个增加/减少，另一个增加/减少），则相关系数为正，而当两个量变化不同步时（如果一个减少，另一个增加）则相关系数为负。

相关系数测量变量之间的线性相关程度。

It is important to remember that the correlation coefficient measures the strength of linear relationship between the variables and therefore a value of zero does not mean that there is no relationship at all. It simply indicates that there is no linear relation between the variables in question.

A zero correlation coefficient simply indicates no linear relationship, but other types are available!

重要的是要记住，相关系数测量变量之间的线性关联的强度，因此零值并不意味着根本没有关系。它只是表明所讨论的变量之间没有线性关系。

零相关系数仅表示没有线性关系，但其他类型可用！

Determining the strength of the relationship provides us with some clues towards answering our original question. Although we can tell whether the relationship is strong (1) or not (0), the correlation coefficient does not tell us how. A regression analysis does allow us to start seeing how.

Regression analysis lets us explore the relationships among variables.

确定关系的强弱为我们提供了回答最初问题的一些线索。虽然我们可以判断关系是强（±1）还是没有线性关系（0），但相关系数并没有告诉我们如何去确定关系。 回归分析帮助我们了解如何去确定关系的强弱。

回归分析让我们探索变量之间的关系。

Before we continue, a word of caution: Just because we measure a correlation between two variables, it does not mean that there is a causal relationship between them. In other words, the fact that people use umbrellas when it rains does not mean that umbrellas cause rain to fall. We are better off avoiding Sir Bedevere’s type of reasoning: “If you weigh the same as a duck, then, you’re made of wood and must be a witch”.

You may have heard the age old aphorism: “Correlation does not imply causation’’.

在我们继续之前，请注意：仅仅因为我们测量两个变量之间的相关性，并不意味着它们之间存在因果关系。 换句话说，人们在下雨时使用遮阳伞这一事实并不意味着雨伞会导致降雨。 我们最好避开贝德维尔爵士的推理类型：“如果你的重量与鸭子一样，那么，你就是用木头做的，必须是一个女巫”。

你可能听过古老的格言：“相关并不意味着因果关系”。

Similarly, we must be careful when considering relationships between variables as they may be related to a third, confounding, variable. Take for example the relationship between ice cream sales and temperature we mentioned earlier on: As Summer approaches, the ice cream van is busy selling more ice cones. A similar trend has been noted for the murder rates, as the heat rises, the number of killings do too1. In a simplistic analysis one may risk looking at the relationship between ice cream sales and murder, and concluding that one causes the other, without taking into account the weather. Always be on the lookout for confounding variables.

1 Lehren, A. W. and Baker, A. (2009, Jun 18th). In New York, Number of Killings Rises With Heat. The New York Times

同样，在考虑变量之间的关系时我们必须要小心，因为它们可能与第三个混淆变量有关。 以我们前面提到的冰淇淋销售与温度之间的关系为例：随着夏季临近，冰淇淋车正在忙着销售更多的冰淇淋。谋杀率也有类似的趋势，因为热量上升，杀戮的数量也是如此。在一个简单的分析中，人们可能会冒险查看冰淇淋销售和谋杀之间的关系，并得出结论，不考虑天气因素，冰淇淋销售和谋杀有因果关系。始终注意混淆变量。

1 Lehren，A. W.和Baker, A.（2009年6月18日）。 在纽约，杀戮数量随着热量上升。 纽约时报

Nonetheless, trying to figure out the existence of these relationships and explaining them is by no means something new. As a matter of fact even the name of the technique carries some historical connotations: Sir Francis Galton, a 19th century polymath and first cousin of Charles Darwin, effectively coined the term. Galton was interested in a variety of subjects, from psychology to astronomy as well as statistics. The acceptance of fingerprints as evidence in court was advanced thanks to Galton’s studies2, including estimating the probability that two people have the same fingerprints.

2 Cole, S. (2004). History of fingerprint pattern recognition. In N. Ratha and R. Bolle (Eds.), Automatic Fingerprint Recognition

Systems, pp. 1–25. Springer New York

尽管如此，试图找出这些关系的存在并解释它们并不是什么新鲜事。事实上，甚至这项技术的名称也带有一些历史内涵：19世纪的博学家弗朗西斯.高尔顿爵士，同时也是查尔斯.达尔文的第一代堂兄，有效地创造了这个词。 高尔顿对各种学科感兴趣，从心理学到天文学以及统计学。因此在法庭上接受指纹作为证据也是由于高尔顿的研究2，包括估计两个人有相同指纹的可能性。

2 Cole，S。（2004）。 指纹模式识别的历史。 在N.Ratha和R. Bolle（Eds.），自动指纹识别系统，第1-25页。 施普林格 纽约

Back to our subject of interest, Galton pioneered the application of statistical methods to many of his scientific interests. For instance, he indeed was interested in the relative size/height of children and their parents3 (both in animals and plants). Among his observations he noticed that a tall parent is likely to have a child that is taller than average. However, the child is likely to be less tall than the parent. Similarly, a parent that is shorter than average would have children taller than the parent, but still below the average. In other words, the difference in height between parent and offspring is proportional to the parent’s deviation from the typical population. He described this by saying that the height of the offspring regresses towards a mediocre point.

3 Galton, F. (1886). Regression Towards Mediocrity in Hereditary Stature. The Journal of the Anthropological Institute of Great

Britain and Ireland *15*, 246–263

回到我们感兴趣的主题，高尔顿率先将统计方法应用于他感性的许多科学领域。例如，他确实对儿童及其父母3（动物和植物）的相对胖瘦/身高感兴趣。在他的观察中，他注意到一个高大的父母可能会有一个高于平均水平的孩子。然而，孩子可能不如父母高。同样，比平均身高矮的父母也会有比父母更高的孩子，但仍然低于平均水平。换句话说，父母和子女之间的身高差异与父母与典型人口身高的偏差成正比。 他通过说明后代的高度向均值方向回归来描述这一点。

3 Galton，F。（1886）。身材遗传回归均值. 大学人类学研究所 英国和爱尔兰15,246-263

The height difference in proportional to the parent’s deviation from the typical population.

We would call this the mean in modern, politically-correct terms.

父母与子女之间的高度差与父母与典型人口的偏差成正比。  
我们用现代，政治上正确的术语，均值来称呼它。

Regression towards the mean is a purely statistical phenomenon and can be seen as a fact of life, if you will. The key to the matter is the expectation value of the measured mean: A sprinter that breaks the world record in a race is expected to run to her average time in the next one, or the score in a mid-term exam can be expected to be less bad than the score in the final.

Regression to the mean is an inescapable fact of life.

回归均值是一种纯粹的统计现象，也被看作生活的真理。问题的关键在于测量平均值的期望值：在比赛中打破世界纪录的短跑运动员预计会在下一个比赛中达到的平均时间，或者期中考试的分数可能会比期末考试的分数差

回归均值是生活中不可避免的事实。

All in all, regression is thus the mean value of a response variable as a function of one or more explanatory variables. A regression model is an approximate to it. As a first attempt to determining the dependence among the variables, the simplest thing we can do is check if the relationship follows a straight line.

In that sense, a linear regression model assumes (among other things) that the response can be described by a linear function. Even if it is not, we can at least approximate linearly over a range of values or carry out transformations to linearise relationships.

总而言之，回归是响应变量的平均值，相应变量是一个函数，具有一个或多个解释变量。 回归模型与相应变量函数近似的。作为确定变量之间依赖关系的第一次尝试，我们能做的最简单的事情是检查关系是否遵循直线。

在这个意义上，线性回归模型假设（除其他事项外）响应可以由线性函数描述。即使不是这样，我们也可以至少在一系列值上线性近似，或者通过变换以线性化关系。

Linear regression assumes:

• Linear relationship.

• Multivariate normality.

• No or little multicollinearity.

• No auto-correlation.

• Homoskedasticity.

线性回归假定：  
•线性关系。

•多元正态性。

•没有或很少多重共线性。

•无自相关。

•同质性

The veracity of a linear model may or may not reflect the actual relationship among the variables in question, and we should remind ourselves that there is no such a thing as a perfect model!

The linear regression model has therefore the following form:

线性模型的准确性可能会或可能不会反映相关变量之间的实际关系，我们应该提醒自己，没有一个完美的模型！  
因此，线性回归模型具有以下形式：

**y** = f (**x**) +,

= + **x** + , (4.1)

where is the intercept of the line, is the slope of the line, anddenotes a vector of random deviations or residuals assumed to be independent and identically normally distributed. We refer to and as the regression coefficients. In the next section we will extend the model to more than one independent variable and will see how to implement the model using matrix notation.

The intercept is the point where the line crosses the y-axis.

其中是线的截距，是线的斜率，并且ε被假定为独立且服从正态分布的随机偏差或残差的向量。 我们将和 称为回归系数。在下一节中，我们将模型扩展为多个独立变量，并将看到如何使用矩阵表示法实现模型。

截距是直线与y轴交叉的点。

*4*.*2* Multivariate Linear Regression

4.2多元线性回归

In the previous discussion we have only taken into account the relationship between the dependent variable and a single independent one. We can extend the model to include many more variables, for example let us consider N observations on the response with i = 1, 2, 3, . . . , N; and with M regressors with j = 1, 2, 3, . . . ,M. The multivariate linear regression model is written as:

在前面的讨论中，我们只考虑了因变量和自变量之间的关系。我们可以扩展模型以包含更多变量，例如让我们观察N个响应，其中i = 1,2,3，....，N; 并且使用M个回归量，其中j = 1,2,3，...，M。 多元线性回归模型写成：

Remember that are vectors that contain the various data points we will use in our model.

请记住，是包含我们将在模型中使用的各种数据点的向量。

= + + (4.2)

We would like to express the linear regression model in terms of matrices. As such, we can write the independent variable as an N M matrix:

我们想用矩阵表示线性回归模型。因此，我们可以将自变量写为N×M矩阵：

1

1

：： ： ： ：

. . . . .

1 …

X = , 4.3)

The matrix **X** provides us with a compact representation of the collection of the different M features for each of our N data points.

whereas the independent variable is a column vector with N elements:

Similarly, the vector **Y** allows us to collate all the various responses

矩阵X为我们提供了N个数据点中每个数据点的M个不同特征集合的紧凑表示。

而自变量是具有N个元素的列向量：

类似地，向量Y允许我们整理所有的响应

：

.

Y = (4.4)

Finally, the regression coefficients and the residuals are given by:

最后，回归系数和残差由下式给出：

：

.

= (4.5)

The intercept has been included in the regression coefficient vector.

截距已经包括在回归系数向量β中。

：

.

= (4.5)

Note that we have included the intercept in the expression of the regression coefficients. This is why we have a column of ones in the matrix shown in expression (4.3) in the previous page.

In that manner, we end up with the following form for the regression model:

注意，我们在回归系数的表达式中包括了截距。 这就是为什么我们在上一页表达式（4.3）显示的矩阵中有1这一列。以这种方式，我们最终得到以下回归模型的形式：

**Y** =  **X** + (4.7)

This is the expression for a multivariate linear regression model.

这是多元线性回归模型的表达式。

Our task is therefore to find the regression coefficients in the vector. The simplicity of Equation (4.7) is provided by the use of matrices. Furthermore, it is also their use that will make all the calculations and manipulations to find the regression coefficients much easier as we shall see in the next section.

因此，我们的任务是找到向量β中的回归系数。 通过使用矩阵，等式（4.7）看着非常简单。 此外，通过使用这种方式，正如我们将在下一节中看到的那样，将使所有计算和操作更容易找到回归系数。

*4*.*3* Ordinary Least Squares

4.3普通最小二乘法

We are tasked with finding the regression coefficientsin the multivariate regression model given by Equation (4.7).Let us recall that we are interested in predicting the value ofthe dependent variable given the values of the explanatoryvariables. If we were able to craft a perfect linear model,the actual value of y would match exactly the prediction f (). This implies that the residuals are zero.

我们的任务是在由方程（4.7）给出的多元回归模型中找到回归系数β。 让我们回想一下，我们有兴趣根据解释变量的值来预测因变量的值。如果我们能够做出完美的线性模型，y的实际值将与预测函数f ()的值完全匹配。这意味着残差ε为零。

A perfect prediction would make the residuals equal to zero.

完美的预测会使残差等于零。

In a more realistic scenario, we would find a pretty good line of best fit to the data by minimising the error. One way to implement a suitable objective function for this purpose is to minimise the sum of squared residuals as follows:

In linear regression we are thus interested in minimising the sum of the square residuals.

更为现实的情况是，我们会通过最小化错误找到最合适的数据线。为了达到这个目的，实现目标函数的一种方法是最小化残差平方和，如下所示：

因此，在线性回归中，我们对最小化平方残差的总和感兴趣。

SSR =, (4.8)

= ,

=,

SSR = **Y**   (4.9)

Notice that the third term in the last expression is actually a scalar: =

请注意，最后一个表达式中的第三个术语实际上是标量：=

Since we require the minimum of the SSR quantity above, we take its derivative with respect to each of the parameters. This leads us to the following expression:

请注意，最后一个表达式中的第三个术语实际上是标量：

由于我们需要上述SSR数量的最小值，因此我们对每个参数求导。我们得到以下表达式：

**Y**

**Y + (X)**  (4.10)

We need to take the first derivative to calculate the minimum.

我们需要采用一阶导数来计算最小值。

We can now equate the above expression to zero, leading us to the solution of the matrix equation as:

我们现在可以将上面的表达式设为零，从而使我们得到矩阵方程的解：

(4.11)

This is the solution to the linear model given in Equation (4.7).

这是公式（4.7）中给出的线性模型的解。

We refer to Equation (4.11) as the the normal equation associated with the regression model.

We have already encountered this calculation in Section 2.4.1 where we used Python to demonstrate the use of linear algebra operations such as transposition, inversion and multiplication. Let us go through the calculations one step at a time.

We have actually already implemented this calculation in Section 2.4.1.

我们将等式（4.11）称为与回归模型相关的正规方程。

我们已经在2.4.1节中遇到过这种计算，我们使用Python来演示线性代数运算的使用，例如转置，求逆运算和相乘。让我们一步一步地完成计算。

我们实际上已经在2.4.1节中实现了这个计算。

*4*.*3*.*1* The Maths Way

4.3.1数学方法

We can use the normal equation given in expression (4.11) to solve the linear system given by the linear model in Equation (4.7). Let us see how this is done for the same data used in Section 2.4.1. For the independent variable we have:

我们可以使用表达式（4.11）中给出的正规方程来求解由公式（4.7）中的线性模型给出的线性系统。 让我们看看如何对第2.4.1节中使用的相同数据进行此操作。对于自变量，我们有：

1

2

3

4

X = (4.12)

and for the dependent variable:

对于因变量：

1

2

3

4

Y = (4.13)

We have a single feature and four records.

我们有一个特征和四个记录。

We have a very succinct dataset with only four observations and one single feature. In other words, we have an N = 4 by M = 1 system. Let us express Equation (4.7) as = .

We can now start by calculating as follows:

我们有一个非常简洁的数据集，一个特征只有四个观察量。换句话说，我们有一个M = 1 乘以 N = 4的系统。让我们将等式（4.7）表示为= 。

我们现在可以通过计算开始如下：

In this case we have a 41 linear system.

在这种情况下，我们有一个41线性系统。

= (4.14)

=

=

= (4.15)

We calculate the first part of the solution as =

我们计算的第一部分=

Whereas the second part is given by:

第二部分由下式给出**：**

The second part is given by=

第二部分由＝

=  (4.16)

=

= (4.17)

Finally, the regression coefficients are given by:

最后，回归系数由下式给出：

= ,

=

= (4.18)

Finally the multiplication　 gives us the coefficients of thelinear regression.

最后给出了线性回归系数。

As we can see from the results above, the intercept of the model is zero and the slope of the line is one. In other words, the model can be expressed by the following equation:

从上面的结果我们可以看出，模型的截距为零，线的斜率为1。 换句话说，该模型可以用以下等式表示：

**Y** = **x**, (4.19)

Remember that the model is Y = + x.

还记得模型是用Y = + x.表示的。

and therefore the line of best fit is given by a line at crossing the origin, as can be seen in Figure 4.1. The grey circles correspond to the data points used for the regression, and the line is given by Equation (4.19).

因此，最佳拟合线由经过原点夹角为线给出，如图4.1所示。灰色圆圈对应于回归计算中的数据点，并且该线由等式（4.19）给出。

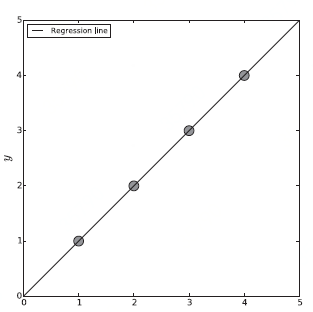


Figure 4.1: The regression procedure for a very well-behaved dataset where all data points are perfectly aligned. The residuals in this case are all zero.

图4.1：非常好的数据集的回归过程，其中所有数据点完全对齐。 在这种情况下，残差都是零。

Not only is it important to be able to carry out the operations that enable us to determine the regression　coefficients, but also we must be able to interpret them. In case of the intercept , we can consider this to be the expected mean value of the predicted variable when the independent variable is not present. By the same token, a “unit” increase in the independent variable is associated with a “unit” increase in the predicted variable.

不仅确定回归系数值的操作很重要，能够解释它们也很重要。 在截距的情况下，我们可以将其视为当不存在自变量时预测变量的预期平均值。出于同样的原因，自变量的“单位”增加与预测变量的“单位”增加相关联。

Not present here means that the dependent variable is zero, x = 0.

此处不存在表示因变量为零，x = 0。

Please note that in the example used in this section, since all the points considered are perfectly aligned, the line of best fit does indeed pass through every point. However, in a more realistic situation the presence of noise cannot be ignored. This is why it is important to get an estimate of the sum of squared residuals. A schematic representation of this situation is shown in Figure 4.2, where the distance from each of the data points to the line of best fit is shown as a dot-dashed line.

请注意，在本节中使用的示例中，由于考虑的所有点都完全对齐，因此最佳拟合线确实通过了每个点。 然而，在更现实的情况下，噪声的存在不容忽视。 这就是为什么残差平方和的估计很重要。 图4.2显示了这种情况的示意图，其中从每个数据点到最佳拟合线的距离显示为点划线。

Although the calculations we have covered above are straightforward, they can be somewhat laborious. This is particularly true in the case where there are more than one or two independent variables at play. Furthermore, the matrix inversion that must be carried out is not a trivial calculation, and we should be careful as there is no guarantee that a given matrix is invertible. In those cases, we say that the matrix is singular or degenerate and methods to approximate the inverse are needed.

虽然我们上面讨论的计算看起来很直接，但计算它们可能有些费力。 在存在多于一个或两个独立变量的情况下尤其如此。 此外，必须执行的矩阵求逆不是一个简单的计算，我们应该小心，因为不能保证给定的矩阵是可逆的。 在这些情况下，我们说矩阵是奇异矩阵或退化矩阵，并且需要近似逆的方法。

The calculations shown are straightforward, but can get somewhat laborious.

计算看起来很简单，但可能会有些费力。

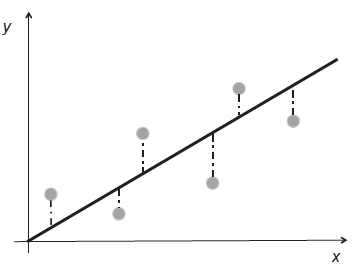


Figure 4.2: The regression procedure for a very well-behaved dataset where all data points are perfectly aligned. The residuals in this case are all zero.

图4.2：非常良好的数据集的回归过程，其中所有数据点完全对齐。 在这种情况下，残差都是零。

We can implement a function in Python with the normal equation given by expression (4.11). A naïve approach may be well suited for simple problems. In a nutshell, we do not really want to solve the problem “by hand” as done in this section, but perhaps we are better off using Python libraries that are readily available for this task such as those in StatsModels and Scikit-learn.

Instead of doing the computations “by hand”, we are better off using a computer.

我们可以使用表达式（4.11）给出的正规方程在Python中实现一个函数。 一种朴素的方法可能非常适合简单的问题。 简而言之，我们并不想像本节所做的那样“手动”解决问题，但也许我们最好使用Python库，例如StatsModels和Scikit-learn。

我们最好不要“手动”进行计算，最好使用计算机。

*4*.*4* Brain and Body: Regression with One Variable

4.4脑与身体：单变量回归

Let us now take a look at running a linear regression with a slightly larger dataset. In this case, we are going to follow one of the examples that we mentioned at the beginning of this chapter. The dataset that we will use looks at relationship of the body mass of an animal and the mass of its brain4. The data is available at http:// dx.doi.org/10.6084/m9.figshare.1565651 as well as at <http://www.statsci.org/data/general/sleep.html>.

现在让我们使用稍大的数据集运行线性回归。在这种情况下，我们将遵循本章开头提到的一个例子。我们使用的数据集着眼于动物体重与其大脑重量的关系4。数据可在http：// dx.doi.org/10.6084/m9.figshare.1565651以及http://www.statsci.org/data/general/sleep.html上获得。

4 Allison, T. and D. V. Cicchetti (1976, Nov 12). Sleep in mammals: ecological and constitutional correlates. Science *194*, 732–734

We will assume that the data has been downloaded into a comma-separated-value (CSV) file with the name mammals.csv and saved in a subfolder called Data. We can use Pandas to manipulate the file. Let us start by looking at a scatter plot of the data. Before we can do this we need to upload the necessary modules:

我们假定数据已下载到名为mammals.csv的逗号分隔值（CSV）文件中，并保存在名为Data的子文件夹中。我们可以使用Pandas来操作文件。让我们从查看数据的散点图开始。在操作之前，我们需要上传必要的模块：

We start by importing the Python libraries that will help us manipulate the data.

我们首先导入可帮助我们操作数据的Python库。

%pylab inline

import numpy as np

import matplotlib.pyplot as plt

import pandas as pd

Note that the %pylab inline command at the beginning of the code above imports NumPy and matplotlib enabling plots to be printed in the Jupyter notebook. We are also explicitly importing these libraries to make the code a bit clearer. Finally, we also import Pandas, and assign to it the alias pd.

请注意，上面代码开头的％pylab内联命令导入NumPy和matplotlib，使得绘图可以在Jupyter笔记本中打印。我们还明确地导入这些库以使代码更清晰。最后我们还导入Pandas，并为其分配别名pd。

The %pylab inline command imports NumPy and matplotlib.

％pylab inline命令导入NumPy和matplotlib。

Let us load the data into a Pandas dataframe called mammals and visualise the data with a scatter plot as shown in Figure 4.3:

让我们将数据加载到名为mammals的Pandas数据框中，并使用散点图可视化数据，如图4.3所示：

mammals = pd.read\_csv(u’./Data/mammals.csv’)

plt.scatter(mammals[’body’], mammals[’brain’])

Using the read\_csv method from Pandas we can read our CSV file.

使用Pandas的read\_csv方法，我们可以读取我们的CSV文件。

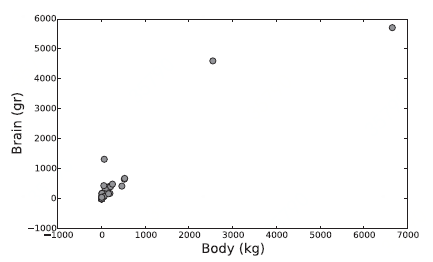


Figure 4.3: A scatter plot of the brain (gr) versus body mass (kg) for various mammals.

图4.3：各种哺乳动物的大脑（gr）与体重（kg）的散点图。

Let us take a look at the number of entries for each of the two variables we are considering. First the variable called body:

让我们看看我们这两个变量各自的条目数。首先看一下body变量：

> body\_data = mammals[’body’]

> body\_data.shape

(62, )

The variable brain gives us the following:

Brain变量的数目用下面的式子来看：

> body\_data = mammals[’brain’]

> body\_data.shape

(62, )

As we can see there are 62 records in the dataset. We have assigned the values of each of the columns of the dataset to new variables for ease of reference. Here we will make use of the StatsModels module to perform the regression. We show how to use Scikit-learn in Section 4.4.1. First we have to add a column of ones to the body variable so that we can accommodate for the calculation of the intercept,. StatsModels has a handy function for this purpose: add\_constant

我们可以看到数据集中有62条记录。我们已将数据集的每个列的值分配给新的变量，以便于参考。在这里，我们将使用StatsModels模块来执行回归。我们将在4.4.1节中展示如何使用Scikit-learn。首先，我们必须在body变量中添加一列1，以便我们可以适应截距的计算，。StatsModels有一个方便的功能：add\_constant

The shape command allows us to see the size of the dataset imported into Pandas.

We make use of StatsModels here. You can see an example of using Scikit-learn in Section 4.4.1.

shape命令允许我们查看导入Pandas的数据集的大小。

我们在这里使用StatsModels。 您可以在第4.4.1节中看到使用Scikit-learn的示例。

import statsmodels.api as sm

body\_data = sm.add\_constant(body\_data)

We make use of the StatsModels library to carry out the regression.

我们利用StatsModels库来进行回归。

In the code above we are loading the StatsModels package and using the sm alias to refer to it. We are then adding a column of ones to the body\_data array making it a 622 array. We are ready to run our regression with the ordinary least squares method (OLS) implemented in StatsModels:

在上面的代码中，我们加载StatsModels包并使用sm别名来引用它。然后我们将一列1添加到body\_data数组中，使其成为62×2数组。 我们准备使用StatsModels中实现的普通最小二乘法（OLS）运行回归：

regression1 = sm.OLS(brain\_data, body\_data).fit()

In particular we are using the ordinary least squares method implemented in StatsModels.

特别是我们使用StatsModels中实现的普通最小二乘法。

If you are familiar with R, you are probably familiar with the “formula notation” used to refer to the dependency of a variable on appropriate regressors. So, if you have a function y = f (x), in R you can denote that dependency with the use of a tilde, i.e ~. This is an easy way to deal with denoting the dependency among variables and fortunately StatsModels has an API that allows us to make use of it in Python too:

如果您熟悉R，您可能熟悉用于指代变量对回归量的依赖性的“公式表示法”。 所以，如果你有一个函数y = f（x），在R中你可以用代数来表示这种依赖，即〜。 这是一种简单的方法来处理变量之间的依赖关系，幸运的是StatsModels有一个API，允许我们在Python中使用它：

import statsmodels.formula.api as smf

regression2 = smf.ols(formula =\

’brain ~ body’,\

data = mammals).fit()

The formula API in StatsModels allows us to simply the notation in the linear regression. It uses formulas similar to those in R.

StatsModels中的公式API允许我们简单地使用线性回归中的表示法。 它使用类似于R中的公式。

Please note that the R-style formula does not require us to add the column of ones to the independent variable. The regression coefficients obtained with both methods are the same, and do not forget that the .fit() command is needed in both cases.

请注意，R风格公式不要求我们将1这一列添加到自变量中。使用这两种方法获得的回归系数是相同的，并且不要忘记在两种情况下都需要.fit（）命令。

Do not forget to use the fit() command to carry out the regression.

不要忘记使用fit（）命令来执行回归。

Let us now take a look at the results of fitting the model to the data provided. StatsModels provides a summary method that renders a nice-looking table with the appropriate information. Unfortunately the formatting of the output is more suitable for showing on a computer screen rather than in a book page. Nonetheless, you are encouraged to try the following command in your shell:

现在让我们看一下将模型拟合到所提供数据的结果。StatsModels提供了一种摘要方法，可以使用适当的信息去呈现外观很漂亮的表。不幸的是，输出的格式更适合在计算机屏幕上而不是在书页中显示。尽管如此，我们鼓励您在shell中尝试以下命令：

StatsModels provides a nice summary of the regression using the summary() command. The output is more suitable to be shown on a computer screen: go ahead and try it in your Python implementation.

StatsModels使用summary()命令提供了很好的回归摘要。 输出更适合在计算机屏幕上显示：继续在Python实现中尝试它。

regression1.summary()

You can run the same command for the second model we executed and you will see the same results:

您可以为我们执行的第二个模型运行相同的命令，您将看到相同的结果：

print(regression2.summary())

Among the information provided you will have the following entries:

在提供的信息中，您将拥有以下条目：

OLS Regression Results

=====================================

Dep. Variable: brain

Model: OLS

Method: Least Squares

No. Observations: 62

R-squared: 0.873

Adj. R-squared: 0.871

Part of the summary provided by OLS gives us information about the goodness of fitness via the coefficient.

OLS提供的部分摘要通过系数向我们提供了有关健身良好性的信息。

OLS tells us the name of the dependent variable, the model used and the method as well as the number of observations used. It also tells us the value of also called the coefficient of determination. The values of this number range between 0 and 1, and it tells us how well the data fit the model. A value of 1 is an indication that the regression line obtained fits the data perfectly well, whereas a value of 0 tells us that the linear model is no good. This measure is related to the Pearson correlation coefficient between the dependent and explanatory variables. We could formulate the linear regression model as a maximisation problem for .

OLS告诉我们因变量的名称，使用的模型和方法以及使用的观察数量。它还告诉我们的值也称为决定系数。此数字的值介于0和1之间，它告诉我们数据与模型的匹配程度。值1表示获得的回归线非常适合数据，而值0表示线性模型不好。该度量与因变量和解释变量之间的Pearson相关系数有关。我们可以将线性回归模型表示为的最大化问题。

The coefficient is related to the Pearson correlation.

系数与Pearson相关性有关。

Having said that, there are some drawbacks with only looking at the value of . Namely, that it increases as we add more explanatory variables to the mix. We should therefore be careful when running regression models by adding extra features: An increase in the value of may not be due to the explanatory power of the input, but to the fact that we added that extra input. That is why OLS

also provides information for the adjusted . It is very similar to , but it introduces a penalty as extra variables are included in the model. The adjusted value increases only in cases where the new input actually improves the model more than would be expected by pure chance.

话虽如此，仅仅看一下的值还是有一些缺点的。也就是说，随着我们在混合中添加更多解释变量，它的值会增加。因此，在通过添加额外特征运行回归模型时应该小心：值的增加可能不是由于输入变量的解释力，而是由于我们添加了额外的输入。 这就是为什么OLS

还提供调整后的信息。它与非常相似，但它引入了一个惩罚，因为模型中包含额外的变量。调整后的值仅在新输入实际上比纯机会预期更多地改进模型的情况下增加。

Although the coefficient of determination provides an indication about how well the model fits the data, it should be used with care.

尽管确定系数提供了关于模型与数据的拟合程度的指示，但应谨慎使用。

An alternative is the adjustedvalue.

另一种方法是调整后的值

For the case of our dataset, an = 0.873 is a pretty good outcome as 87% of the total variance in brain mass is explained by the linear regression model based on the body mass. This means that the regression line obtained must be a good one. And as yet we have not even mentioned what this line is and how it can be obtained from OLS.

对于我们的数据集， = 0.873是相当好的结果，因为基于体重的线性回归模型解释了脑重量总变异的87％。这意味着获得的回归线必须是好的。到目前为止，我们还没有提到这条线是什么以及如何从OLS获得它。

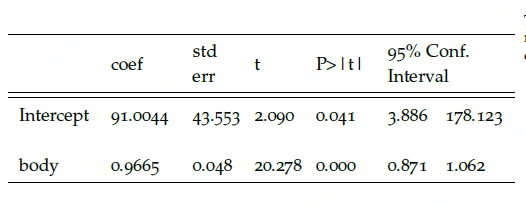


Table 4.1: Results from the regression analysis performed on the brain and body dataset.

表4.1：对脑和身体数据集进行回归分析的结果。

In Table 4.1 we can see the regression parameters obtained from running the ordinary least squares method on the brain and body dataset. The column named “coef” lists the estimated values of the coefficient listed in the table. Notice that the “const” corresponds to the intercept of the model.

在表4.1中，我们可以看到通过在大脑和身体数据集上运行普通最小二乘法获得的回归参数。 “coef”列给出了表中列出的系数的估计值。请注意，“const”对应于模型的截距。

OLS lists the rest of the coefficients using the names of the variables included in the model. The “std err” column corresponds to the basic standard error of the estimate of the coefficient; “t” is the so-called t-statistic and it tells us how statistically significant the coefficient is. The P-value is listed in the “P > |t|” column and it helps us determine the significance of the results considering the null-hypothesis that the coefficient being equal to zero is true. A small P value (typically 0.05) indicates strong evidence against the null hypothesis and you should go with the value obtained for the coefficient. Finally “95% Conf. Interval” gives us the lower and upper values of the 95% confidence interval.

OLS使用模型中包含的变量名称列出其余系数。 “std err”列对应于系数估计的基本标准误差; “t”是所谓的t统计量，它告诉我们系数在统计上是多么重要。 P值列在“P> | t |”列中，它有助于我们确定结果的重要性，考虑系数等于零的零假设为真。小P值（通常≤0.05）表示反对零假设的强有力证据，您应该使用系数获得的值。 最后“95％置信区间“给出了95％置信区间的下限值和上限值。

OLS lists the names of the variables as they appear in our data.

A small P-value indicates evidence against the null hypothesis and so it is rejected.

OLS列出了我们数据中出现的变量名称。

较小的P值表示反对零假设的证据，因此被拒绝。

The results shown in Table 4.1 indicate that the intercept for the model is = 91.0044 and the slope of the line is = 0.9665, leaving us with the following model:

表4.1中显示的结果表明模型的截距= 91.0044，线的斜率= 0.9665，模型如下：

Brain = 0.9665(Body) + 91.0044, (4.20)

and the P-values obtained indicate a rejection of the null hypothesis. We can get the parameters using the method:

获得的P值表示拒绝零假设。 我们可以使用该方法获取参数：

> regression2.params

Intercept 91.004396

mammals.body 0.966496

The regression parameters can be obtained with the params method of the fitted model object.

回归参数可以用拟合模型对象的参数方法获得。

We can use this equation to predict the mass of a mammal given its body mass and this can easily be done with the predict method in OLS. Let us consider new body mass measurements that will be used to predict the brain mass using the model obtained above. We need to prepare the new data in a way that is compatible with the model. We can therefore create an array of 10 new data inputs as follows:

给定体重，我们可以使用这个方程来预测哺乳动物的大脑重量，这可以通过OLS中的预测方法轻松完成。 让我们考虑使用新体重测量值用上面获得的模型预测大脑重量。我们需要以与模型兼容的方式准备新数据。 因此，我们可以创建10个新数据输入的数组，如下所示：

new\_body = np.linspace(0,7000,10)

For the predict method of the model run with the formula API we do not need to add a column of ones to our data and instead we simply indicate that the new data points are going to be treated as a dictionary to replace the independent variable (i.e. exog in StatsModels parlance) in the fitted model. In other words, you can type the following:

对于使用公式API运行的模型的预测方法，我们不需要在数据中添加1这一列，我们只是表明新数据点将被视为字典，以替换拟合模型中的自变量（即StatsModels的说法）。换句话说，您可以键入以下内容：

The predict method of the StatsModel formula API does not need the addition of a column of ones.

StatsModel公式API的预测方法不需要添加一列1。

brain\_pred=regression2.predict(exog=\

dict(body=new\_body))

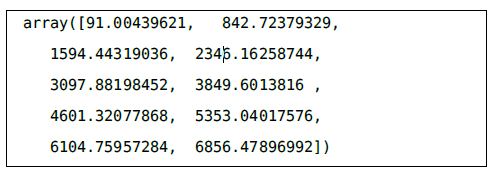
print(brain\_pred)

exog refers to the independent variable.

Exog指的是自变量。

This will generate the following output:

这将生成以下输出：



The numbers shown correspond to the brain mass predictions for the artificial body mass measurements used as input. In Figure 4.4 we can see the regression line given by Equation (4.20) in comparison to the data points in the set. Please note that if you are not using the formula API, the input data will require the addition of a column of ones to obtain the intercept.

显示的数字对应于用作输入的人工体重测量的脑重量预测。在图4.4中，我们可以看到公式（4.20）给出的回归线与集合中的数据点进行比较。请注意，如果您不使用公式API，输入数据将需要添加一列1来获取截距。

We can use the values above to construct the line of best fit given by the model.

我们可以使用上面的值来构建模型给出的最佳拟合线。

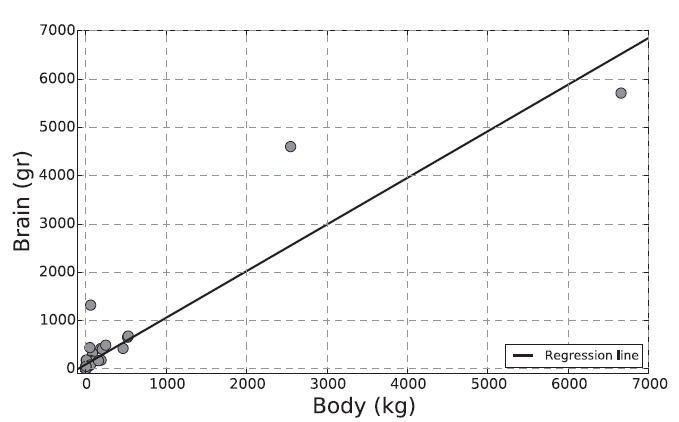


Figure 4.4: A scatter plot and the regression line calculated for the brain (gr) versus body mass (kg) for various mammals.

图4.4：各种哺乳动物的大脑（gr）与体重（kg）的散点图和回归线。

So far so good, but can we do better than that? For example, look at the clustering that happens in the region below 1, 000 kilogram mark for body mass and compare it with the ones that take place after the 2, 000 or 6, 000 kilogram marks. Are the latter outliers?, or can we come up with a better model that encompasses these differences? Let us take a look at a typical transformation carried out in a variety of analyses. But before let us use Scikit-learn.

到目前为止已经做得很好，但我们能做得更好吗？例如，查看在体重低于1,000千克标记的区域中发生的聚类，并将其与在2,000或6,000千克标记之后发生的聚类进行比较。 后者是异常值吗？或者我们能否提出一个包含这些差异的更好的模型？让我们看一下在各种分析中进行的典型转换。 在此之前，我们先使用Scikit-learn。

We have seen how to use StatsModels to perform our linear regression. One of the reasons to use this module is the user-friendly output it generates. Nonetheless, this is not the only way available to us to perform this analysis. In particular, Scikit-learn is another very useful module, and one that we use extensively throughout the book. For completness, in this section we will see how to perform linear regression with Scikit-learn.

我们已经了解了如何使用StatsModels来执行线性回归。 使用此模块的原因之一是它产生对用户友好的输出。尽管如此，这不是我们执行此分析的唯一方法。特别是，Scikit-learn是另一个非常有用的模块，也是我们在本书中广泛使用的模块。出于完整性的考虑，在本节中我们将了解如何使用Scikit-learn执行线性回归。

We can implement a regression model using Scikit-learn.

我们可以使用Scikit-learn实现回归模型。

Let us import the modules to read the data. We will use Pandas to load the data into a data frame called mammals:

让我们导入模块来读取数据。我们将使用Pandas将数据加载到名为mammals的数据集中：

%pylab inline

import numpy as np

import pandas as pd

mammals = pd.read\_csv(u’./Data/mammals.csv’)

This is exactly the same we did in the previous section.

这与我们在上一节中所做的完全相同。

As we discussed in Section 3.10, we know that Scikit-learn expects data to be represented by two-dimensional numeric matrices with M data instances (rows) and N distinct features (columns). In this case, we have 62 instances and one feature. Let us arrange the data as expected by creating appropriate arrays for the dependent and independent variables:

正如我们在3.10节中讨论的那样，我们知道Scikit-learn期望数据由具有M个数据实例（行）和N个不同特征（列）的二维数字矩阵表示。在这种情况下，我们有62个实例和一个特征。 让我们通过为因变量和自变量创建适当的数组来按预期排列数据：

Notice the use of double brackets to get the right shape for the arrays.

注意使用双括号来获得数组的正确形状。

body\_data = mammals[[’body’]]

brain\_data = mammals[[’brain’]]

We are now ready to create our model. First, we need to create an instance of a linear regression model from linear\_model in Scikit-learn. This can be done as follows:

我们现在准备创建我们的模型。首先，我们需要用Scikit-learn的linear\_model创建一个线性回归模型的实例。这可以按如下方式完成：

from sklearn import linear\_model

sk\_regr = linear\_model.LinearRegression()

We carry out linear regression modelling with linear\_model in Scikit-learn.

我们在Scikit-learn中使用linear\_model进行线性回归建模。

With that in place we simply use the fit method for the model and we are done:

有了这个，只需使用模型的拟合方法，我们就完成了：

sk\_regr.fit(body\_data, brain\_data)

We can now check that the intercept and coefficient obtained are the same we calculated with StatsModels. We can also check the value of the coefficient:

我们现在可以检查获得的截距和系数是否与我们使用StatsModels计算的相同。我们还可以检查系数的值：

> print(sk\_regr.coef\_)

[[ 0.96649637]]

> print(sk\_regr.intercept\_)

[ 91.00439621]

> print(sk\_regr.score(body\_data, brain\_data))

0.872662

The coefficients and intercept are obtained with the .coef\_ and .intercept\_ methods.

The coefficient is calculated with the .score method.

使用.coef\_和.intercept\_方法获得系数和截距。

使用.score方法计算系数。

Finally, the predictions can be calculated with the predict method:

最后，可以使用predict方法计算预测：

new\_body = np.linspace(0, 7000, 10)

new\_body = new\_body[:, np.newaxis]

brain\_pred = sk\_regr.predict(new\_body)

Note the use of np.newaxis to render the Numpy array into the format expected by Scikit-learn.

请注意使用np.newaxis将Numpy数组渲染为Scikit-learn所期望的格式。

*4*.*5* Logarithmic Transformation

4.5对数变换

One of the principal tenets of the linear regression model is the idea that the relationship between the variables at play is linear. In cases when that is not necessarily true, we can apply manipulations or transformation to the data that result in having a linear relationship. Once the linear model is obtained, we can then undo the transformation to obtain our final model.

Hence the name!

线性回归模型的主要原则之一是认为变量之间的关系是线性的。如果不一定如此，我们可以对导致具有线性关系的数据进行操作或转换。一旦获得线性模型，我们就可以撤消变换以获得我们的最终模型。

由此得名！

A typical transformation that is often used is applying a logarithm to either one or both of the predictive and response variables.

通常使用的典型转换是将对数应用于预测变量和响应变量中的其中一个或两个都包含。

Applying the logarithmic function is a typical way to transform our data.

应用对数函数是转换数据的典型方法。

Let us see what happens to the scatter plot of the body and brain data we have been analysing when we apply the logarithmic transformation to both of the variables. We will create a couple of new columns in our Pandas dataframe to keep track of the transformations performed:

让我们看看当我们将对数变换应用于两个变量时，我们一直在分析的身体和大脑数据的散点图会发生什么变化。我们将在Pandas数据集中创建几个新列，以跟踪执行的转换：

from numpy import log

mammals[’log\_body’] = log(mammals[’body’])

mammals[’log\_brain’] = log(mammals[’brain’])

Remember that log in Python refers to the base-e logarithm.

请记住， Python中的log 对应于底数为e的对数。

We can plot the transformed data and as we can see from Figure 4.5 the data points are aligned in a way that indicates a linear relationship in the transformed space.

我们可以绘制变换后的数据，正如我们从图4.5中看到的那样，数据点的排列方式表明了变换空间中的线性关系。

Why has this happened? Well, remember that we are trying to use models (simple and less simple ones) that enable us to exploit the patterns in the data. In this case the relationship we see in this data may be modelled as a power law; e.g. y = .

为什么会这样？好吧，请记住，我们正在尝试使用模型（简单和更简单的模型），使我们能够利用数据中的模式。在这种情况下，我们在这些数据中看到的关系可以建模为幂;例如y = 。

The transformation has helped us convert our problem into a simpler one.

变换帮助我们将问题变得更为简单。

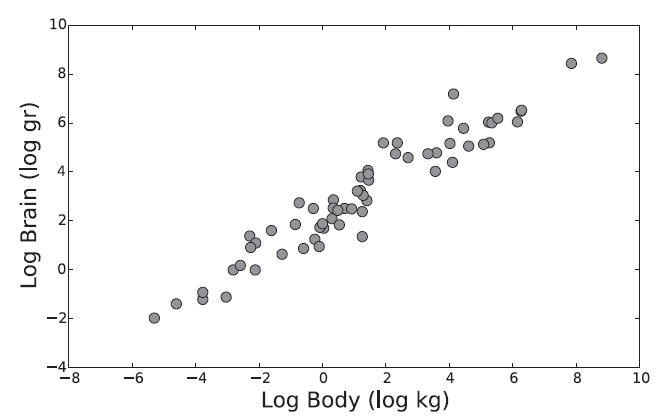


Figure 4.5: A scatter plot in a log-log scale for the brain (gr) versus body mass (kg) for various mammals.

图4.5：各种哺乳动物的脑（gr）与体重（kg）的对数 - 对数标度的散点图。

The log-log transformation applied to the data maps this nonlinear relationship to a linear one, effectively transforming the complicated problem into a simpler one:

对数据图的对数 - 对数变换将这种非线性关系映射到线性关系，有效地将复杂问题转化为更简单的问题：

log(y) = log(), (4.21)

log(y) = b log(x),

= b, (4.22)

where we are using the notation log for the inverse of the exponential function e. As we can see, we have transformed our power law model, a nonlinear model in the regressor, into a form that looks linear as shown in Equation (4.22).

我们使用符号log表示指数函数e的逆运算。正如我们所看到的，我们已经将我们的幂模型（回归量中的非线性模型）转换为看起来是线性的形式，如公式（4.22）所示。

In engineering and other disciplines they use the notation ln for this function.

在工程和其他学科中，他们使用符号ln来实现此功能。

Now that we have this information at hand, we can train a new model using the transformed features attached to the mammals dataframe:

现在我们掌握了这些信息，我们可以使用mammals数据集的变换后的特征训练一个新模型：

The only difference here is that we are using the transformed variables in the OLS function.

这里唯一的区别是我们在OLS函数中使用变换后的变量。

log\_lm=smf.ols(formula = ’log\_brain ~ log\_body’,\

data = mammals).fit()

If we print the log\_lm.summary() we will see some of the following information:

如果我们打印log\_lm.summary（），我们将看到以下一些信息：

OLS Regression Results

=====================================

Dep. Variable: log\_brain

Model: OLS

Method: Least Squares

No. Observations: 62

R-squared: 0.921

Adj. R-squared: 0.919

However, the results returned by OLS are remarkably different.

但是，OLS返回的结果却截然不同。

The logarithmic transformation has increased the value of from 0.873 to 0.921. We can see the value of the sum of squared residuals with the following command:

对数变换使的值从0.873增加到0.921。我们可以使用下面的命令查看残差平方和的值：

> log\_lm.ssr

28.9227104215

We can obtain the sum of the square residuals with the ssr method.

我们可以用ssr方法得到平方残差的总和。

Let us now take a look at the statistics for the model as well as the all important coefficients. As we can see from Table 4.2, the new model has an intercept of = 2.1348 and a slope of = 0.7517. The regression line for this model can be seen in Figure 4.6.

现在让我们看一下模型的统计数据以及所有重要的系数。从表4.2可以看出，新模型的截距为= 2.1348，斜率为= 0.7517。该模型的回归线如图4.6所示。

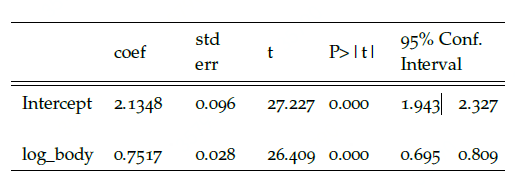


Table 4.2: Results from the regression analysis performed on the brain and body dataset using a log-log transformation.

表4.2：使用对数 - 对数转换对脑和身体数据集进行的回归分析的结果。

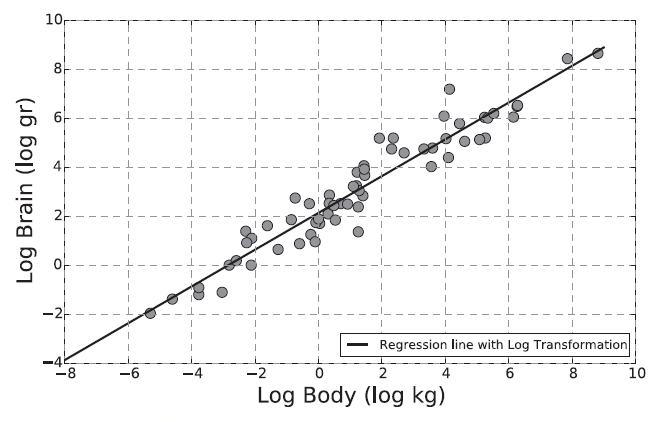


Figure 4.6: A log-log scale figure with a scatter plot and the regression line calculated for the brain (gr) versus body mass (kg) for various mammals.

图4.6：具有散点图的对数 - 对数比例图和针对各种哺乳动物计算的脑（gr）与体重（kg）的回归线。

Remember that the coefficients obtained above are for the transformed data and if we wanted to relate this to the original features we would need to undo the transformation. In this case we have a model given by

请记住，上面获得的系数是针对转换后的数据，如果我们想将其与原始特征联系起来，我们需要撤消转换。在这种情况下，我们有一个模型给出

Brain = A (4.23)

Where A =

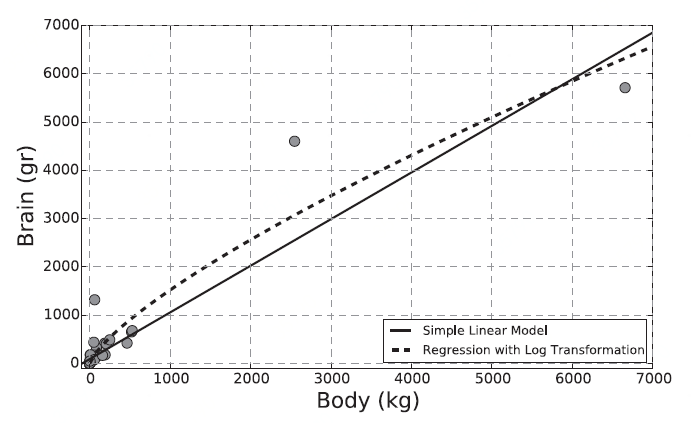


Figure 4.7: A comparison of the simple linear regression model and the model with logarithmic transformation for the brain (gr) versus body mass (kg) for various mammals.

图4.7：简单线性回归模型与各种哺乳动物的脑（gr）与体重（kg）的对数变换模型的比较。

In Figure 4.7 we can see the original scatter plot and a comparison of the two models. It is easy to see how the logarithmic transformation allows for greater flexibility, capturing those data points which are a struggle for the simple linear model. This comparison demonstrates that it is possible to build a variety of models to explain the behaviour we observe in the data. In Section 4.7 we will see how we can fit a polynomial regression to the same dataset. Nonetheless, please remember that carrying our appropriate splitting of training and testing sets, together with cross-validation is an unrivalled way to decide which model, among those tried, is the most suitable to use with unseen data.

The logarithmic transformation enabled us to capture those data points which are a struggle for the linear model.

在图4.7中，我们可以看到原始散点图和两个模型的比较。很容易看出对数变换如何允许更大的灵活性，捕获那些为简单线性模型而挣扎的数据点。这种比较表明，可以构建各种模型来解释我们在数据中观察到的行为。在4.7节中，我们将看到如何将多项式回归拟合到同一数据集。尽管如此，请记住，携带我们适当的训练和测试集以及交叉验证是一种无与伦比的方法，可以决定哪些模型最适合与尚未获得的数据一起使用。

对数变换使我们能够捕获那些为线性模型而挣扎的数据点。

*4*.*6* Making the Task Easier: Standardisation and Scaling

4.6使任务更容易：标准化和扩展

Given that the main underlying concept behind linear regression is the assumption of a linear relationship, transformations such as the one covered in the previous section make the task easier for both the learning algorithm and for us. As you can imagine, there may be many more tricks up our jackalope sleeves to transform and pre-process the data in order to facilitate our modelling. In this section we are going to present a couple of the most widely used techniques to transform our data and provide us with anchors to interpret our results.

鉴于线性回归背后的主要基本概念是线性关系的假设，如上一节所述的转换使得学习算法和我们的任务变得更容易。可以想象，我们的jackalope sleeves可能会有更多的技巧来转换和预处理数据，以便于我们的建模。在本节中，我们将介绍一些最广泛使用的技术来转换我们的数据，并为我们提供解释结果的锚点。

Remember that the main assumption of linear regression is the existence of a linear relationship.

Data pre-processing is nothing new in the data science workflow.

请记住，线性回归的主要假设是存在线性关系。

数据预处理在数据科学工作流程中并不是什么新鲜事。

One of those techniques consists on centring the predictors such that their mean is zero, and is often used in regression analysis. Among other things it leads to interpreting the intercept term as the expected value of the target variable when the predictors are set to zero. Another useful transformation is the scaling of our variables. This is convenient in cases where we have features that have very different scales, where some variables have large values and others have very small ones.

其中一种技术包括使预测变量居中，使其平均值为零，并且通常用于回归分析。除其他事项外，当预测变量设置为零时，可以将截距项解释为目标变量的预期值。另一个用途是转换是变量的缩放。这在我们具有尺度差别很大的特征的情况下是方便的，其中一些变量具有大的值而另一些具有非常小的值。

Centring the data around their mean leads us to interpret the intercept as the expected value of the target variable when the predictors are set to zero.

将数据围绕其均值居中使我们将截距解释为当预测变量设置为零时目标变量的预期值。

As mentioned above, standardisation and scaling may help us interpret our results: They allow us to transform the features into a comparable metric with a known range, mean, units and/or standard deviation. It is important to note that the transformations to be used depend on the dataset and the domain where the data is sourced from and applied to. It also depends on the type of algorithm and answer sought. For example, in a comprehensive study of standardisation for cluster analysis, Milligan and Copper5 report that standardisation approaches that use division by the range of the feature provide a consistent recovery of clusters. We shall talk about clustering in the next chapter. Let us go through the two techniques mentioned above in a bit more detail.

如上所述，标准化和缩放可以帮助我们解释结果：它们允许我们将特征转换为具有已知范围，平均值，单位和/或标准差的可比度量。值得注意的是，要使用的转换取决于数据集以及数据源的区域和数据将被应用的区域。它还取决于算法的类型和所寻求的答案。例如，在对聚类分析标准化的综合研究中，Milligan和Copper5报告说，使用该特征范围划分的标准化方法可以提供一致的聚类恢复。我们将在下一章讨论聚类。让我们更详细地介绍上面提到的两种技术。

The transformations to be applied depend on the dataset and the domain of application.

要应用的转换取决于数据集和应用的域。

5 Milligan, Glenn W. and Cooper, Martha C. (1988). A study of standardization of variables in cluster analysis. Journal of

Classification *5*(2), 181–204

*4*.*6*.*1* Normalisation or Unit Scaling

4.6.1正则化或单位缩放

The aim of this transformation is to convert the range of a given variable into a scale that goes from 0 to 1. Given a feature f with a range between fmin and fmax the transformation is given by:

此转换的目的是将给定变量的范围转换为从0到1的范围。给定fmin和fmax之间范围的特征f，转换由下式给出：

fscaled = . (4.24)

Unit scaling transforms our data into a scale between 0 and 1.

单位缩放将我们的数据转换为0到1之间的比例。

Notice that this method of scaling will cast our features into equal ranges, but their means and standard deviations will be different. An alternative formulation divides each feature by its range without subtracting the minimum value. We can apply this unit scaling to our data with the preprocessing method in Scikit-learn that includes the MinMaxScaler function to implement unit scaling.

Unit scaling leaves the means and standard deviations unchanged.

请注意，这种缩放方法会将我们的特征转换为相同的范围，但它们的均值和标准偏差会有所不同。 替代公式将每个特征除以其范围，而不是减去最小值。 我们可以使用Scikit-learn中的preprocessing方法将此单位缩放应用于我们的数据，其中包括MinMaxScaler函数以实现单位缩放。

from sklearn import preprocessing

Scikit-learn comes with a preprocessing module.

Scikit-learn附带一个预处理模块。

Once we have loaded the appropriate function, we can apply the scaling as follows:

加载完相应的函数后，我们可以按如下方式应用缩放：

scaler = preprocessing.MinMaxScaler()

mammals\_minmax = pd.DataFrame(\

scaler.fit\_transform(mammals[[’body’, ’brain’]]),\

columns = [’body’,’brain’])

Scikit-learn includes MinMaxScaler to implement unit scaling.

Scikit-learn包含MinMaxScaler以实现单位缩放。

Let us see the minimum and maximum values of the transformed data:

让我们看看转换数据的最小值和最大值：

> mammals\_minmax.groupby(lambda idx: 0).\

agg([’min’,’max’])

body brain

min max min max

0 0.0 1.0 0.0 1.0

Here we are using Pandas to manipulate our dataset.

在这里，我们使用Pandas来操纵我们的数据集。

*4*.*6*.*2* z-Score Scaling

4.6.2 z-Score缩放

An alternative method for scaling our features consists of taking into account how far away data points are from the mean. In order to provide a comparable measure, the distance from the mean is calculated in units of the standard deviation of the feature data.

缩放特征的另一种方法是考虑数据点与均值的距离。为了提供可比较的度量，以特征数据的标准差为单位计算与平均值的距离。

The standard deviation is a measure of dispersion.

标准差度量离散的程度。

In this case a positive score tells us that a given data point is above the mean whereas a negative one is below the mean. The standard score explained above is called the z-score as it is related to the normal distribution. The transformation that we need to carry out for a feature f with mean and standard deviation is given by:

在这种情况下，正分数告诉我们给定的数据点高于平均值，而负数表示低于平均值。上面解释的标准分数称为z分数，因为它与正态分布有关。我们对具有均值和标准偏差的特征f执行的变换由下式给出：

z-score scaling is related to the normal distribution.

z-score缩放与正态分布有关。

fzscore =. (4.25)

Strictly speaking, the z-score must be calculated with the mean and standard deviation of the population, otherwise we are making use of the Student’s t statistic6.

严格地说，z得分必须用人口的平均值和标准差来计算，否则我们正在利用学生的t统计量6。

6 Freedman, D., R. Pisani, and R. Purves (2007). Statistics. International student edition. W.W. Norton & Company

Scikit-learn’s preprocessing method allows us to standardise our features in a very straightforward manner:

Scikit-learn的预处理方法允许我们以非常直接的方式标准化我们的功能：

scaler2 = preprocessing.StandardScaler()

mammals\_std = pd.DataFrame(\

scaler2.fit\_transform(mammals[[’body’,’brain’]]),\

columns = [’body’,’brain’])

Scikit-learn includes StandardScaler to implement z-score scaling.

Scikit-learn包括StandardScaler以实现z-score缩放。

After the transformation we should have features with zero mean and standard deviation one; let us check that this is the case:

转换后我们应该具有零均值和标准差的特征; 让我们检查一下这种情况：

> mammals\_std.groupby(lambda idx: 0).\

agg([’mean’,’std’])

body brain

mean std mean std

0 1.790682e-18 1.008163 -3.223228e-17 1.008163

Once again, the aggregation shown here uses Pandas.

再次，此处显示的聚合使用Pandas。

*4*.*7* Polynomial Regression

4.7多项式回归

In the previous section we have seen how a simple transformation in the input and output variables make a complex model into a simpler one. In fact, we can try fitting different models using more and more complex functions. One important point to note is that a model is said to be linear when it is linear in the parameters. With that in mind, the model are both linear as the parameters are linear. In the case of the examples above, the models are given by second order polynomials in one and two variables, respectively.

在上一节中，我们已经看到输入和输出变量中的简单转换如何将复杂模型转换为更简单的模型。 事实上，我们可以尝试使用越来越复杂的函数来拟合不同的模型。需要注意的一点是，当参数是线性时，模型被认为是线性的。考虑到这一点，模型

y = + x + + , (4.26)

和模型

y =+ x + + + + + , (4.27)

都是线性的，因为参数 是线性的。 在上述示例的情况下，模型分别由一个和两个变量的二阶多项式给出。

A model is said to be linear when it is linear in the parameters.

These two models are linear.

The models above are given by second order polynomials in one and two variables respectively. We can then talk about polynomial regression.

当它的参数是线性时，模型是线性的。

这两个模型是线性的。

上述模型分别由一阶和二阶二阶多项式给出。接下来我们讨论多项式回归。

When using such models to fit our data we talk therefore about polynomial regression and in general the k-th order polynomial model in one variable is given by:

因此当使用这些模型来拟合我们的数据时，我们使用多项式回归，并且一般来说，一个变量中的k阶多项式模型由以下给出：

y = + x + + + + . (4.28)

The techniques for fitting a linear regression model can be used for the models above too.

拟合线性回归模型的技术也可用于上述模型。

This is general polynomial model.

这是一般多项式模型。

The techniques for fitting a linear regression model can be used for the models above too.

拟合线性回归模型的技术也可用于上述模型。

Polynomial models can be very useful in cases where we know that nonlinear effects are present in the target variable. The polynomial model is effectively the Taylor series expansion of an unknown function and thus can be used to approximate it. Furthermore, it is possible to use different orthogonal functions to define the model. For instance, if we decide to use trigonometric functions we end up effectively in the realm of harmonic analysis and the regression would give us the coefficients we would obtain via a Fourier transform.

多项式模型在已知目标变量中存在非线性效应的情况下非常有用。多项式模型是一个未知函数的有效的泰勒级数展开，因此可以用来逼近它。此外，可以使用不同的正交函数来定义模型。例如，如果我们决定使用三角函数，我们最终会高效的进入调和分析领域，回归会告诉我们我们通过傅里叶变换获得的系数。

A polynomial model is effectively the Taylor series expansion of an unknown function.

多项式模型是一个未知函数的泰勒级数展开。

Let us fit a quadratic model to the brain and body dataset we have been using in the previous sections. We can start by adding a feature that corresponds to the square of the body mass:

让我们用上一节中使用的大脑和身体数据集拟合一个二次项的模型。我们可以通过添加一个对应于身体重量平方的特征来开始：

mammals[’body\_squared’]=mammals[’body’]\*\*2

In this case we are trying a quadratic model with our test dataset.

这个例子中，我们尝试在我们的测试集中使用两次项模型。

We can now fit the quadratic model given by Equation (4.26), and using StatsModels is a straightforward task:

我们现在可以拟合由等式（4.26）给出的二次项模型，并且使用StatsModels是一个简单的任务：

poly\_reg=smf.ols(formula=\

’brain~body+body\_squared’,\

data=mammals).fit()

The application of OLS remains the same.

OLS的应用仍然是相同的。

We can take a look at the parameters obtained:

我们看一下得到的参数：

> print(poly\_reg.params)

Intercept 19.115299

body 2.123929

body\_squared -0.000189

The parameter for the quadratic term seems to be small.

二次项的参数似乎很小。

In other words, we have a model given by

换句话说，我们有一个模型

Brain = 19.115 + 2.124(Body) 1.89 . (4.29)

It may seem that the coefficient of the quadratic term is rather small, but it does make a substantial difference to the predictions. Let us take a look by calculating the predicted values and plot them against the other two models:

二次项的系数似乎很小，但它确实对预测有很大影响。让我们看一下预测值，并将它们与其他两个模型进行比较：

poly\_brain\_pred=poly\_reg.predict(exog=\

dict(body=new\_body,\

body\_squared=new\_body\*\*2))body\_squared -0.000189

However, it makes a substantial difference to the predictions made.

然而，它对预测有很大的不同。

As we can see in Figure 4.8, the polynomial regression captures the data points much closer than the other two models. However, by increasing the complexity of our model we are running the risk of overfitting the data. We know that cross-validation is a way to avoid this problem, and other techniques are at our disposal, such as performing some feature selection by adding features one at a time (forward selection) or discarding non-significant ones (backward elimination). In Section 4.9 we shall see how feature selection can be included in the modelling stage by applying regularisation techniques.

正如我们在图4.8中所看到的，多项式回归捕获的数据点比其他两个模型都更接近。然而，通过增加我们的模型的复杂性，我们正经历过度拟合数据的风险。我们知道交叉验证是避免此问题的一种方法，还有一些我们可以控制的其他技术，例如通过每次添加一个特征（前向选择）或丢弃不重要的特征（后向消除）来执行一些特征选择。在第4.9节中，我们将看到如何通过应用正则化技术将特征选择纳入建模阶段。

We have to bear in mind that increasing the complexity of our model, increases the chances of overfitting.

我们必须记住，增加我们的模型的复杂性，会增加过拟合的机会。

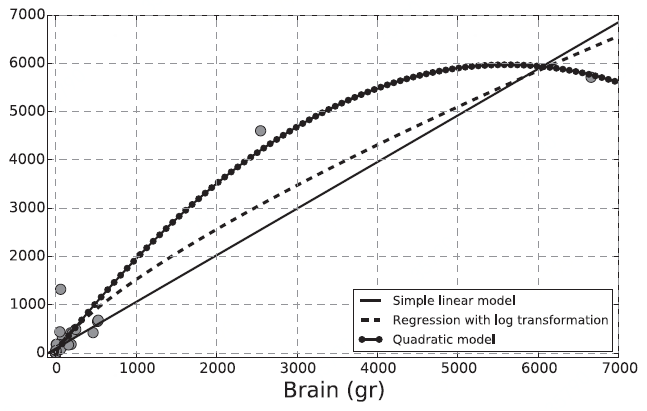


Figure 4.8: A comparison of a quadratic model, a simple linear regression model and a model with logarithmic transformation fitted to the brain (gr) versus body mass (kg) for various mammals.

图4.8：各种哺乳动物的二次项模型、简单线性回归模型和对数变换模型的比较。

When using polynomial regressions there are a number of things that should be taken into account. For instance, the order of the polynomial model should be kept as low as possible; remember that we are trying to generalise and not run an interpolation.

当使用多项式回归时，有许多事情需要考虑。例如，多项式模型的阶数应该保持尽可能低；请记住，我们试图泛化，而不是插补。

Once a model is obtained, take particular care not to overuse the model; extrapolating with the aid of a

polynomial model is a perilous task. There are other technical issues to be aware of. For instance, as the order of the polynomial increases, matrix inversion calculations become inaccurate; this is a form of ill conditioning and it introduces errors in the estimation of the parameters.

一旦获得模型，特别注意不要过度使用模型；借助多项式模型外推是一项危险的任务。还有其他需要注意的技术问题。例如，随着多项式的阶数增加，矩阵逆计算变得不精确；这是病态条件的一种形式，并且在参数的估计中引入误差。

Extrapolating with a polynomial model is a perilous task.

用多项式模型外推是一项危险的任务。

Another aspect to take into account is that if the values of the independent variables are limited to a narrow range, there can also be significant ill conditioning of the problem, or multicollinearity in the features used to train the model. Multicollinearity is the name we give to the situation where two or more features in our model are highly (or even moderately) correlated with each other. This becomes

aggravated in a polynomial regression as higher powers of a feature are highly correlated with each other. Let us look at this empirically by taking the 9th and 10th powers of an array and calculating their correlation:

另一个需要考虑的方面是，如果自变量的值被限制在一个窄的范围内，那么问题也可能存在显著的病态条件，或者用于训练模型的特征中存在多重共线性。多重线性是指模型中的两个或多个特征高度相关（或者适度相关）的情况。这在多项式回归中变得更为严重，因为特征的较高的幂指数使得彼此高度相关。用一个数组的9次方和10次方，并计算它们的相关性：

Multicollinearity arises when two or more features are highly correlated with each other.

当两个或多个特征彼此高度相关时，出现多重共线性。

x = np.random.random\_sample(500)

x1, x2 = x\*\*9, x\*\*10

cor\_mat = np.corrcoef(x1,x2)body\_squared -0.000189

Let us take a look at the correlation matrix:

让我们看看相关矩阵

> print(cor\_mat)

[[ 1. 0.99877083]

[ 0.99877083 1. ]]

As we can see, the correlation between and is high.

我们可以看到， 和 的相关性很高。

As we can see for the random numbers generated by my computer, the correlation coefficient between x1 and x2 is quite close to 1. Multicollinearity results in having wide swings in the values of the parameter estimations when small changes in the data are included. Also, the coefficients obtained may be such that their standard errors are quite high with low significance levels, although they are actually jointly significant and the is high.

可以看到，由计算机产生的随机数，x1和x2之间的相关系数非常接近1。当包括数据的微小变化时，多重线性导致参数估计的值具有宽的摆动。而且，所获得的系数可以使得它们的标准误差相当高而显著性水平较低，尽管它们实际上都很显著且较高。

The numbers may be different in your computer. 计算机产生的随机数字可能不同。

It is important to note that multicollinearity is not exclusive of the use of polynomial models. In fact, it is quite possible that two seemingly independent features included in our data are highly correlated among themselves, having a confounding effect in our model and thus we should avoid using these features together in our model.

重要的是要注意，多重共线性并不排斥多项式模型的使用。事实上，很可能我们的数据中包含的两个看似独立的特征彼此高度相关，对模型产生不好的效果，因此我们应该避免在模型中一起使用这些特征。

Remember that multicollinearity is not exclusive of polynomial models.

请记住，多重共线性并不排斥多项式模型。

*4*.*7*.*1* Multivariate Regression

4.7.1多元回归

In the examples so far, we have concentrated mainly on regression models that have one single independent variable to explain the target we are interested in. As mentioned earlier in this chapter, in the case where we have more than one input variable we are entering the realm of multivariate regression.

在到目前为止列举的例子中，我们主要集中在用一个单独的自变量的回归模型来解释我们感兴趣的目标。正如本章前面提到的，在我们有多个输入变量的情况下，我们就进入多元回归领域。

Multivariate regression refers to having more than one input variable in our model.

多元回归指的是在我们的模型中有多个输入变量。

In a sense we have already - indirectly - seen an example of a multivariate regression in Section 4.7 when we addressed the ideas behind polynomial regression. In that case, the added features were powers of a single input variable. For the more general case of a multivariate regression the features are given by different independent variables.

从某种意义上说，当我们思考多项式回归背后的逻辑时，我们已经-间接地-看到了4.7节中多元回归的一个例子。 在这种情况下，添加的特征是单个输入变量的次方。 对于多变量回归的更一般情况，特征由不同的自变量给出。

Polynomial regression is effectively a multivariate regression problem.

多项式回归实际上是多元回归问题

If we consider a set of M predictors , , , . . . , that are hypothesised to be related to a response variable y, the multivariate regression model can be expressed as:

如果我们考虑一组M个预测变量, , , . . . , ，假设他们与响应变量y相关，多变量回归模型可表示为：

Y = + + . . .+ + , (4.30)

A general multivariate linear model.

一般的多变量线性模型。

and the best part is that the parameter estimation for this model can be achieved with the same techniques discussed in Section 4.3. This means that we can continue using the same StatsModels libraries described in the previous sections. Also, as before, multicollinearity should be avoided when considering the various independent features to be included in our model.

最好的部分是该模型的参数估计可以用4.3节中讨论的相同技术实现。这意味着我们可以继续使用前面章节中描述的相同StatsModels库。 此外，与以前一样，在考虑我们的模型中包含的各种独立特征时，应避免多重共线性。

We can continue using StatsModels for polynomial regression.

我们可以继续使用StatsModels进行多项式回归。

*4*.*8* Variance-Bias Trade-Off

4.8方差-偏差权衡

Now that we have explored the ideas behind describing the relationship among variables with a model of the type given the expression:

现在我们已经探索了用给定表达式模型描述变量之间的关系背后的思想：

y = f (x) + , (4.31)

we can take a look at the expected prediction error obtained when estimating a model (x). This is given by the expectation of the squared error

我们可以看看在估计模型（X）时得到的预期预测误差。这是由平方误差的期望给出的。

The expectation of the squared error can be decomposed into bias, variance and noise.

对平方误差的期望可以分解为偏差、方差和噪声。

E. (4.32)

This expectation value can be can be decomposed into portions that correspond to bias, variance and noise respectively.

该期望值可以被分解为分别对应于偏差、方差和噪声的部分。

In order to facilitate the decomposition let us first consider a random variable Z with a probability distribution given by P(Z). We denote the expectation value of Z as E[Z]. Let us calculate the expectation value of :

为了便于分解，让我们首先考虑随机变量Z，由P(Z)给出的概率分布。我们将Z的期望值表示为E[Z]。让我们计算的期望值

E[] = E[] -2E[Z]E[Z] + [Z],

= E[] -2[Z]+ [Z],

= E[] -[Z]

We will use the result of this calculation later on.

稍后我们将使用这个计算的结果。

and thus E[] = E[] + [Z]. (4.33)

因此 E[] = E[] + [Z]. (4.33)

Using the expression above, we can now take a look at the decomposition of the expectation of the squared error:

使用上面的表达式，我们现在可以看看对平方误差的期望的分解：

E = E[ (x) + (x)],

= E[] -2E[y]E[(x)] + E[(x)]

= E[] +

-2E[y]E[(x)]

+E[]

+

= E[(] + (4.34) Variance

+ (4.35) Bias

E[] (4.36) Noice

where the first term (4.34) corresponds to **variance**, the second one (4.35) to the square of the **bias** and finally the third (4.36) is the **noise**.

其中第一项（4.34）对应于方差，第二项（4.35）对应于偏差的平方，最后第三项（4.36）是噪声。

This decomposition shows that apart from the noise, there are two sources of error in our model. Our task is the minimisation of these two error sources that preclude our algorithm from generalising.

这种分解表明，除噪声外，我们的模型中存在两个误差源。 我们的任务是最小化这两个错误源，这两个错误使我们的算法无法泛化。

We need to find a balance between the variance and bias.

我们需要在方差和偏见之间找到平衡点。

On the one hand variance tells us how sensitive the model is to small fluctuations in the training set, on the other hand bias is related to the difference between the expected value of our estimator and its true value. High variance results in overfitting whereas high bias results in under-fitting. Finding a good model is therefore a matter of balancing the bias and the variance. This tradeoff applies to algorithms used in supervised learning.

一方面，方差告诉我们模型对训练集中的小波动有多敏感，另一方面，偏差与我们估计量的预期值与其真实值之间的差异有关。高方差导致过度拟合，而高偏差导致欠拟合。因此，找到一个好的模型是平衡偏差和方差的问题。这种权衡适用于监督学习中使用的算法

High variance gives us more complex models, whereas high bias yields simpler ones.

高方差给我们更复杂的模型，而高偏差产生更简单的模型。

*4*.*9* Shrinkage: LASSO and Ridge

The decomposition of our prediction error into its variance and bias components makes it clear that a balance between the two is required for any regression problem we may encounter. In general, linear regression exhibits high variance and low bias and it should therefore stand to reason that lowering the variance at the expense of the bias is the way to go.

把我们的预测误差分解成它的方差和偏差分量清楚地表明，对于我们可能遇到的任何回归问题，两者之间都需要平衡。一般来说，线性回归显示出高方差和低偏差，因此有理由认为，以偏差为代价降低方差是可行的方法。

Linear regression exhibits high variance and low bias.

线性回归表现出高方差和低偏差。

Furthermore, we have also seen that our ability to interpret the outcome when adding more and more features is diminished. It would be therefore preferable to identify those features that are deemed to be the most important ones. Unfortunately, our linear regression model, as it stands at the moment, does not allow us to do this automatically

此外，我们还看到，当添加越来越多的特征时，我们解释结果的能力会降低。因此，最好确定那些被认为是最重要的特征。不幸的是，我们现在的线性回归模型不允许我们自动执行此操作

Feature selection is not possible within the straightforward linear model.

在简单的线性模型中，特征选择是不可能的。

Let us recall that given our model

让我们回顾一下我们的模型

This is the linear model which we have been using all along.

这是我们一直使用的线性模型。

(4.37)

we are interested in choosing the coefficients and in order to minimise the Ordinary Least Squares (OLS) criterion given by Equation (4.8) which is just the sum of squared errors. The coefficients are in effect a way of determining whether a particular feature is important or not. In particular, the closer the coefficient is to zero the less significant the feature is.

我们感兴趣的是选择系数和，以便最小化由等式（4.8）给出的普通最小二乘（OLS）标准，该标准只是平方误差的总和。

该系数实际上是确定特定特征是否重要的方式。 特别是，系数越接近零，特征越不重要。

The model coefficients are a way to determine if a feature is important or not.

模型系数是确定特征是否重要的一种方式。

Let us then consider replacing the estimates with a smaller value such that

然后让我们考虑用较小的值替换估计值

= (4.38)

The parameter *λ* lets us tune the value of the coefficients.

参数λ让我们调整系数的值。

When λ= 0 our coefficients are unchanged, and as *λ* gets larger and larger, the coefficients start shrinking down to zero. In this manner, with the right choice of *λ* we can get an estimator with an improved error. The estimate is biased, but remember that we were happy to sacrifice some of that to make up for the variance.

当λ= 0时，我们的系数不变，随着λ越来越大，系数开始缩小到零。 通过这种方式，通过正确选择λ，我们可以得到一个具有改进误差的估计器。 估计是有偏差的，但请记住，我们很乐意牺牲其中的一些来弥补差异。

The new estimates are biased though.

然而，新的估计有偏差。

Shrinkage of the coefficients is therefore a form of regularisation as we penalise the model for increased complexity as given by the size of the coefficients. In Section 3.8 we introduced the L2- and L1-norms and it is thus natural to consider these measures for the size of the coefficients.

因此，系数的收缩是正则化的一种形式，因为我们惩罚了由系数的大小增加的模型的复杂性。在第3.8节中，我们介绍了L2-和L1-范数，因此很自然地考虑这些系数大小的度量。

The use of the L2-norm results in the so-called **ridge regression**7:

L2范数的使用导致所谓的**ridge**回归7：

= min

7 Hoerl, A. E. and R. W. Kennard (1970). Ridge regression: Biased estimation for nonorthogonal problems. Technometrics *12*(3),

55–67

whereas the application of the L1-norm leads to the Least

而L1范数的应用导致最少值。

Absolute Shrinkage and Selection Operator or **LASSO**8 for short:

绝对收缩和选择运算符或LASSO8简称：

= min

8 Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. J. R. Statist. Soc. B *58*(1), 267–288

In both cases, as the value ofis increased, the bias increases, whereas the variance decreases. It controls the amount of penalty imposed on the model and therefore it is important to find a good value for this parameter. **Model** **selection** is the process of finding the appropriate value for the hyperparameter, and cross-validation is a good way to tackle this problem.

在两种情况下，随着λ值的增加，偏差增加，而方差减小。它控制对模型施加的惩罚量，因此找到该参数的良好值非常重要。模型选择是为超参数找到适当值的过程，交叉验证是解决此问题的好方法。

The tuning parameteris sometimes called thehyperparameter.

调谐参数有时称为超参数。

We discussed cross-validation in Section 3.12.

我们在3.12节讨论了交叉验证。

For instance, using k-fold cross-validation and a set of possible hyperparameter values λ

, we partition our data in K folds: F1, F2, . . . , Fk. For each value of k = 1, . . . , K, we train on the feature values in the training set Fi with i k and validate on the feature values in Fk.

例如，使用k-fold交叉验证和一组可能的超参数值λ   ，我们将数据分为K个折叠：F1, F2, . . . , Fk。 对于k = 1,…K中的每个值，我们训练训练集Fi中的特征值，其中i≠k并验证Fk中的特征值。

A quick recipe to find a suitable value for the hyperparameter, using cross-validation.

使用交叉验证为超参数λ找到合适值的快速配方。

For each value in the setwe compute our estimate on the training set as well as the error on the validation set and compute the average error over all folds. The latter provides us with a curve that corresponds to the cross-validation error. The value of the hyperparameter to choose is such that it minimises the cross-validation error itself which in turns corresponds to the best score for the model.

对于集合λ中的每个值，我们计算训练集上的估计值以及验证集上的误差，并计算所有折叠的平均误差。 后者为我们提供了与交叉验证错误相对应的曲线。 要选择的超参数的值使得它最小化交叉验证错误本身，而交叉验证错误本身又对应于模型的最佳分数。

Do not be deceived by the similar look between the ridge and lasso regressions; the solutions do have significant differences. Although the ridge regression works well in cases where there are coefficients whose values are actually close to zero, the algorithm never explicitly sets them to this value.

不要被ridge和lasso回归之间的相似性所欺骗，解决方案确实有显著的差异。虽然ridge回归在系数接近0的情况下表现的很好，但算法从未明确地将它们设置为该值。

Ridge and LASSO differ on the type of penalty imposed.

ridge 和 lasso在惩罚类型上有所不同。

Unless λ =

除非λ =

This means that in some cases, with ridge regression the feature selection wished for is not possible, particularly when there are a large number of features involved. In the case of LASSO, the usage of the L1-norm as the penalty means that it is possible for some of the coefficients to be shrunk down to zero, making feature selection possible.

这意味着，在某些情况下，对于ridge回归，特征选择是不可能的，尤其是在涉及大量特征的情况下。在LASSO的情况下，使用L1-范数作为惩罚意味着可以将一些系数缩小到零，使得特征选择成为可能。

Feature selection is possible with LASSO.

利用LASSO进行特征选择是可行的。

It is important to remember that if the features included in our model are not on the same scale, the estimates obtained with both ridge and LASSO are not fair. In those cases it is recommended to use some scaling as described in Section4.6. Let us know see how we can run ridge and LASSO regressions in Python. In this case we are going to use Scikit-learn to carry out the modelling.

重要的是要记住，如果我们的模型中包含的特征不是相同的，那么使用ridge和LASSO获得的估计是不公平的。在这些情况下，建议使用第4.6节中描述的一些缩放。 让我们知道如何在Python中运行ridge和LASSO回归。在这种情况下，我们将使用Scikit-learn来进行建模。

We discussed scaling in Section 4.6.

我们在4.6节讨论了缩放。

We will continue working with the body and brain dataset we have been using all along, and in order to make things more interesting we will use a feature that corresponds to the cube of the body size. We can do this as follows:

我们将继续使用我们一直使用的身体和大脑数据集，为了使事情变得更有趣，我们将使用与身体尺码的立方相对应的特征。 我们可以这样做：

Assuming that we have already added the square of the bodysize.

假设我们已经添加了身体尺码的平方。

mammals[’body\_cubed’]=mammals[’body’]\*\*3

A contrived feature, but it will serve our purposes for a demo.

一个人为的功能，但它将用于我们的演示目的。

Let us first start by scaling our data using z-score scaling:

首先让我们使用Z-score缩放来缩放数据：

from sklearn import preprocessing

X = mammals[[’body’,’body\_squared’,’body\_cubed’]]

Y = mammals[[’brain’]]

Xscaled = preprocessing.\

StandardScaler().fit\_transform(X)

Yscaled = preprocessing.\

StandardScaler().fit\_transform(Y)

As we know, this scaling can be done with StandardScaler.

正如我们所知，这种缩放可以用StandardScaler来完成。

Not only are we interested in finding the coefficients that describe each of the ridge and LASSO models, but we also want to find a good value for λin each case. In order to do that we will need to carry out the cross-validation procedure as described earlier in this section.

我们不仅对寻找描述ridge和LASSO模型中的系数感兴趣，而且我们还想在每种情况下为λ找到一个好的值。为了做到这一点，我们需要执行本节前面描述的交叉验证过程。

Fortunately, Scikit-learn provides us with GridSearchCV which is a helpful function that lets us perform an exhaustive search over specified parameter values by implementing a fit and a score methods. The latter will let us choose the value for our hyperparameter λ. First, we need to create our test and training sets:

幸运的是，Scikit-learning为我们提供了GridSearchCV，这是一个有用的功能，它允许我们通过实现fit和score方法对指定的参数值上进行穷举搜索。后者将让我们选择我们的超参数λ的值。首先，我们需要创建我们的测试和训练集：

Scikit-learn lets us carry out an exhaustive search of parameter combinations with the GridSearchCV method.

Scikit-learn允许我们使用GridSearchCV方法对参数组合进行详尽的搜索。

import sklearn.model\_selection as ms

XTrain, XTest, yTrain, yTest =\

ms.train\_test\_split(Xscaled, Yscaled,\

test\_size= 0.2, random\_state=42)

We split our data into training and testing sets.

我们将数据分成训练和测试集。

We can now define a set of parameters to be used in our search:

我们现在可以定义一组要在搜索中使用的参数：

from sklearn.model\_selection import GridSearchCV

from sklearn.linear\_model import Ridge, Lasso

lambda\_range = linspace(0.001,0.2,25)

lambda\_grid = [{’alpha’: lambda\_range}]

We need to define a dictionary that holds the set of values to be searched. Note that Scikit-learn refers to the hyperparameter as *a*.

我们需要定义一个字典来保存要搜索的值。 请注意，Scikit-learn将超参数称为*a*。

Our search will use each of the values in the lambda\_grid dictionary and carry our cross-validation with the number of folds desired:

我们的搜索将使用lambda\_grid字典中的每个值，并使用所需的折叠数进行交叉验证：

model1 = Ridge(max\_iter=10000)

cv\_ridge = GridSearchCV(estimator=model1,\

param\_grid=lambda\_grid,\

cv=ms.KFold(n\_splits=20))

cv\_ridge.fit(XTrain, yTrain)

model2 = Lasso(max\_iter=10000)

cv\_lasso = GridSearchCV(estimator=model2,\

param\_grid=lambda\_grid,\

cv=ms.KFold(n\_splits=20))

cv\_lasso.fit(XTrain, yTrain)

GridSearchCV takes care of the cross-validation step using the set of parameters to be searched. In this case we are using k-fold cross-validation.

GridSearchCV使用一组要搜索的参数来处理交叉验证步骤。在这种情况下，我们使用k-折叠交叉验证。

In Figure 4.9 we show a heatmap with the values for the hyperparameters used in our search and their method, as well as their corresponding cross-validation mean scores. We can obtain the actual values selected with the help of the best\_params\_ method as follows:

在图4.9中，我们显示了一个热图，其中显示了搜索中使用的超参数值及其方法，以及它们相应的交叉验证平均值。我们可以通过best\_params\_方法获得的实际值如下：

> cv\_ridge.best\_params\_[’alpha’],\

cv\_lasso.best\_params\_[’alpha’]

(0.133666666667, 0.00929166666667)

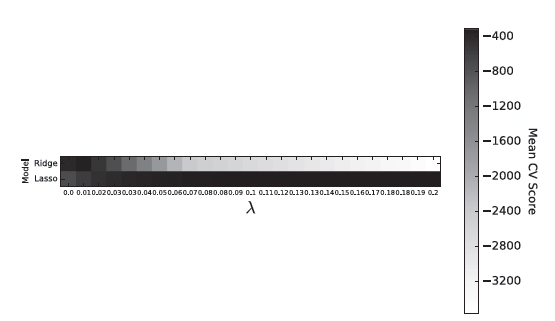


Figure 4.9: Using GridSearchCV we can scan a set of parameters to be used in conjunction with cross-validation. In this case we show the values of λused to fit a ridge and LASSO models, together with the mean scores obtained during modelling.

图4.9：使用GridSearchCV，我们可以扫描一组参数，与交叉验证一起使用。 在这种情况下，我们显示用于拟合ridge和LASSO模型的λ的值，以及在建模期间获得的平均分数。

We can now use these parameters with the corresponding models to extract the coefficients:

我们现在可以将这些参数与相应的模型一起使用来提取系数：

> bestLambda\_lasso=cv\_lasso.best\_params\_[’alpha’]

> Brain\_Lasso = Lasso(alpha=bestLambda\_lasso,\

max\_iter=10000)

> Brain\_Lasso.fit(XTrain,yTrain)

> print(Brain\_Lasso.coef\_)

[ 1.65028091 -0. -0.76712502]

Similar code can be written for the ridge model.

可以为ridge模型编写类似的代码。

as we can see, the second coefficient has been shrunk down to zero with the application of the LASSO regression.

我们可以看到，随着LASSO回归的应用，第二个系数已经缩小到零。

Finally, let us take a look at the residual sum of squares obtained with the test dataset:

最后，让我们看一下使用测试数据集获得的残差平方和：

> lasso\_prediction = Brain\_Lasso.predict(XTest)

> print(‘‘Residual sum of squares:%.4f ’’\

% np.mean((lasso\_prediction - yTest)\*\*2))

Residual sum of squares: 0.0114

Once again, similar code can bewritten for the ridge model.

再一次，可以为ridge模型编写类似的代码。

*4*.*10* Summary

In this chapter we have addressed the topic of regression analysis. It is a natural first step in our data science and analytics journey as it is one of the most widely used techniques out there. Mastering regression is a must for a jackalope data scientist.

在本章中，我们讨论了回归分析的主题。这是我们数据科学和分析之旅的第一步，因为它是最广泛使用的技术之一。掌握回归是jackalope数据科学家必须的技能 。

We have seen how regression allows us to describe relationships between input features and the target variable, bearing in mind that correlation does not imply causation.

我们已经看到回归如何允许我们描述输入特征和目标变量之间的关系，同时要记住相关性并不意味着因果关系。

Using the language of linear algebra we implemented the Ordinary Least Squares (OLS) model to solve the linear regression problem and extended this to a multivariate situation. Furthermore, we have also seen how the models can be used to carry out polynomial regression.

使用线性代数的语言，我们实现了普通最小二乘（OLS）模型来解决线性回归问题，并将其扩展到多变量情境。此外，我们还看到了如何使用模型进行多项式回归。

The use of appropriate transformations to our input (and output) data was shown to be beneficial to our modelling task, and the interplay between bias and variance in our models is an important concept to take into account at the modelling stage.

对我们的输入（和输出）数据使用适当的变换被证明有利于我们的建模任务，并且我们模型中偏差和方差之间的相互作用是在建模阶段需要考虑的重要概念。

The continuous tension between bias and variance can be used to our advantage in the form of regularization techniques such as ridge and LASSO, allowing us to fine tune our models in a very flexible manner. Regression is indeed a useful tool that every jackalope data scientistshould have in the toolbox.

在偏差和方差之间连续的矛盾可以用来在规则化技术（如ridge和LASSO）中为我们提供优势，使我们能够以非常灵活的方式微调我们的模型。回归确实是每个jackalope数据科学家在工具箱中应该具有的有用工具。