第5章

基于人工智能的数据分析

本章概要

人工智能在数据分析中主要通过算法模型挖掘规律。例如，逻辑回归（判断两类结果，如是否患病）、决策树（按特征分层判断，预测用户行为）、聚类算法（如 K 均值自动分组相似数据）。神经网络模仿人脑处理复杂模式（如图像识别），而随机森林结合多个模型提升准确性。现代技术还能实时分析金融风险或优化推荐系统。这些算法将散乱的数据转换为可理解的趋势预测或分类结论，成为各领域智能决策的核心工具。

学习目标

（1）掌握机器学习的基本概念与流程。
（2）理解监督学习、无监督学习与强化学习的区别。
（3）掌握数据集划分与模型评估的方法。
（4）了解常用机器学习算法的原理与应用。
（5）掌握回归分析与分类分析的基本方法。
（6）理解聚类分析与主成分分析的基本思想。
（7）了解神经网络与深度学习的基本结构。
（8）能够使用 AI 工具辅助编程与数据分析。

5.1 机器学习概述

机器学习是人工智能的核心分支，旨在让计算机从数据中自动学习规律并做出预测或决策。主要分为监督学习（如用带标签的数据训练分类模型）、无监督学习（如聚类未标注数据发现隐藏模式）和强化学习（通过试错反馈优化策略）。例如，预测天气、识别手写数字或推荐电影都依赖机器学习模型。其优势在于能处理海量数据，发现人眼难以察觉的关联，广泛应用于医疗诊断、金融风控、自动驾驶等领域，是智能化时代的基础工具。

机器学习通常需经过如图 5-1-1 所示的 5 个步骤。

（1）数据准备。收集含特征和标签的原始数据，如在电商客户分析中，年龄、收入特征、购买行为标签等。

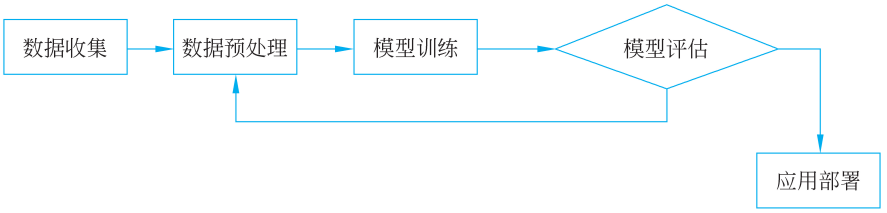

图 5-1-1　机器学习的 5 个步骤

(2) 数据预处理。清洗数据,形成数据集。数据集中的每条数据都称为样本(sample)或示例(instance),数据集通常需要划分为用于学习规则的训练集和用于模拟测试的测试集。

(3) 模型训练。选择算法,利用数据,在有样例的情况下练习,即监督学习,或者自己通过大量数据归纳,最后总结出规律,得到模型,即无监督学习。

(4) 模型评估。用测试集验证效果,防止过拟合,即死记硬背,不会融会贯通。

(5) 应用部署。对新来的数据,可以利用模型完成任务,如分类、回归预测、聚类等。

5.2　机器学习算法的分类

机器学习算法主要分为三种类型:监督学习、无监督学习和强化学习。每种学习类型都适用于不同的场景和任务。

5.2.1　监督学习

监督学习是机器学习中最常见的一种类型,它使用已标记的数据训练模型,学习输入与输出之间的映射关系。在监督学习中,算法从带标签的数据集中学习,找到规律,确定模型,然后对新的、未见过的数据,利用模型进行预测。主要的监督学习有分类和回归。

监督学习的效果取决于训练集,即用于训练模型的数据集,一般占全部数据集的 70% 左右,包含输入和对应的正确输出(标签)。

机器通过多轮学习,每轮都利用损失函数校正。这里的损失函数是衡量机器学习模型预测结果与真实值差距的数学标尺。它通过计算预测误差,为模型提供明确的优化方向——就像导航提示“当前偏离路线 200m”,驱动模型通过自动调整内部参数来减少误差,最终提升预测的准确性。

经过多轮训练后效果如何,可以通过模型评估来测试。模型评估是通过测试集来验证模型的真实能力。其关键原理包含以下三个层面。

(1) 隔离验证。用训练集之外的测试集检测模型,防止过拟合。

(2) 设定合理的量化指标。如准确率、精确率、召回率等来衡量模型的质量。

(3) 检验模型的泛化能力。即模型处理未知数据的能力,通过交叉验证(类似多科目综合测评)检验模型的稳定性。

通过这样的评估闭环。即训练时用损失函数优化,评估时用独立指标验证,二者共同确保模型既“学得好”又“用得稳”。

5.2.2 无监督学习

无监督学习是机器学习中不需要标签数据指导的训练范式,其核心目标是通过分析数据的内在结构,发现隐藏模式或特征关联。在无监督学习中,算法没有正确的输出作为指导,而是通过分析数据本身的特性来发现规律。比如通过计算某个特征点与某个样本点之间的距离,随后将其归类为相似或不相似。这一过程并不会生成明确的函数规则用于判断,只是依照预先定义的度量阈值进行归类。无监督学习主要用于解决以下三类任务。

(1) 聚类分析。将数据划分为具有相似特征的组别,例如,K 均值算法可以根据消费者的购买行为自动划分用户群体,常用于市场细分、生物物种分类等。

(2) 降维处理。用于提取关键特征降低数据复杂度,例如,主成分分析将基因表达数据压缩至可视化维度,常用于消除冗余信息,提升计算效率。

(3) 关联规则挖掘,发现数据项间的共生关系。典型应用如超市购物篮分析、买啤酒的顾客常常同时购买花生。

无监督学习与监督学习的本质区别如下。

(1) 输入数据仅含特征而无预设标签。

(2) 评估标准依赖数据本身的分布特性(如簇内紧密度、维度信息保留率)。

(3) 更接近人类探索未知事物的认知方式。

例如在社交网络分析中,无监督学习可自动识别用户社区结构,揭示未标注的群体行为特征。它的目标是让数据自己“开口说话”,揭示人眼难以直接看出的结构和规律。

5.2.3 强化学习

强化学习通过试错机制与环境互动,根据奖励或惩罚调整策略,实现长期目标。在强化学习中,算法通过与环境互动获得反馈(奖励或惩罚),并根据这些反馈调整其行为以最大化累积奖励。强化学习特别适用于需要序列决策的任务,例如,玩游戏或机器人控制。简单来说就是通过让参与实验的角色不断与周围的环境做交互,产生新的数据,随后进一步用于训练。

5.2.4 常见的机器学习算法

常见的机器学习算法有很多,如线性回归、逻辑回归、支持向量机、K 均值聚类、主成分分析、决策树等。机器学习如同解决问题的“工具箱”,不同任务需要选用特定工具。因此,当面对现实中的数据挑战时,首先需明确核心目标,比如当需要完成数据预测任务时,可以选用线性回归,绘制预测曲线;当需要进行类别判断时,可以选择逻辑回归、支持向量机,对数据类别做判定;当需要发现隐藏的分组时,可以选择 K 均值聚类;当需要简化数据维度时,可以选择主成分分析,为数据瘦身;当需要提高数据的可解释性时,可选择决策树,提供可解读的“决策说明书”。当遇到更复杂的任务时,还可能需要多算法协作。下面介绍这些常用的机器学习算法的工作原理与应用场景。

由于 scikit-learn(简称 sklearn)是 Python 目前最流行的开源机器学习库,专为数据科学和预测分析设计,具有覆盖监督学习(分类/回归)和无监督学习(聚类/降维)的特点,内置经典算法,如线性回归、决策树、SVM、随机森林等,并提供数据预处理、模型评估、参数调优

的全流程支持。同时，其所有模型遵循 fit()（训练）、predict()（预测）、score()（评估）的调用逻辑，学习成本低，简单易用、功能全面，适合从入门到工业级应用。因此，本节相关机器学习算法的体验会利用 sklearn 实现。

5.3　数据集及算法评价

5.3.1　数据集

在机器学习中，数据集通常被划分为以下几部分。

- 训练集（Train Set）。一般占数据集全部的 60%～80%，用于模型训练，形成模型。
- 验证集（Validation Set）。一般占数据集全部的 10%～20%，是模型训练过程中单独留出的样本集，它通过对模型的能力进行初步评估调整模型的参数。
- 测试集（Test Set）。一般占数据集全部的 10%～20%，仅用于模型的最终性能评估。

例如，对于包含了猫图片和狗图片的数据集，训练集中包含猫与狗物种的图片，让模型学习得到猫图片的特征和狗图片的特征，随后在验证集与测试集中，会包含没有出现在训练集中的全新的猫图片和狗图片，测试集中甚至可能会有狼的图片来让模型区分狼图片更近似于猫的特征还是狗的特征。而狼图片的出现便意味着能够衡量模型的泛化能力，即对完全不存在于训练集中的内容的识别能力。

在划分三类数据集时，应注意以下三方面。

（1）数据代表性。各子集需保持与原始数据相同的分布（如疾病数据中的患者/健康人比例一致）。

（2）数据独立性。测试集需完全隔离，仅在最终阶段使用（避免“偷看答案”）。

（3）时间敏感性。时间序列数据应按时间划分（如用前 3 年的数据训练，后 3 个月的数据验证和测试），而非时间序列数据则需随机打乱后划分，避免局部偏差。

另外还要考虑到一些特殊场景的处理方法，如对于小样本数据，可以采用 K 折交叉验证（如 5 折：每次用 80%训练、20%验证，循环 5 次取平均）；对于类别不平衡的数据，可以使用分层抽样，以确保各类别的比例一致；对于不同来源的数据（如不同医院的病历），则需按来源划分测试集。

5.3.2　泛化能力与评价指标

泛化能力（Generalization Ability）是机器学习中最关键的概念之一，它衡量了模型在从未见过的数据上的表现能力。一个具有良好泛化能力的模型不仅能在训练数据上表现良好，还能在新的、未见过的数据上做出准确的预测，即模型是否具备“举一反三”或“学以致用”的能力，是机器学习的最终目标。一般来说，模型评价的各项指标实际上就是在评价其泛化能力。

例如，一个用于预测房价的模型如果只能准确预测训练集中出现过的房子价格，而无法预测新房子的价格，那么这个模型就没有实际价值。真正有用的模型应当能根据新房子的特征（如面积、位置、房龄等）准确预测其价格。

模型的泛化能力常受到过拟合（Overfitting）和欠拟合（Underfitting）的挑战。过拟合

是指模型在训练数据上表现很好,但在测试数据上表现较差,记住了训练数据的细节,包括噪声和异常值,而没有学习到数据的一般模式。例如,一个学生在考试前死记硬背,记住了大量的习题和答案,但在考试中遇到稍微变形的题目就无法解答,这就是过拟合的表现。过拟合可能由模型过于复杂、训练数据集太小、没有使用正则化技术或没有进行充分的交叉验证等原因引起。解决过拟合的方法包括正则化、数据增强、交叉验证和集成学习。

欠拟合则是指模型在训练数据上表现较差,无法捕捉数据中的模式和趋势,模型过于简单,无法学习到数据的复杂模式。欠拟合就像一个学生没有努力学习,连基本的概念都没有掌握,大部分的题目都答不上来。欠拟合可能由模型过于简单、训练时间不足或数据质量差等原因引起。解决欠拟合的方法包括增加模型复杂度、增加训练时间和优化算法参数。

在机器学习中,不同类型的模型泛化能力可以用不同的评估指标进行衡量,一般用于解决分类问题的模型常用的性能评估指标包括准确率、精确率、召回率、$F1$ 分数等,如表 5-3-1 所示。而用于回归问题的模型,则会使用均方误差、R^2 分数等指标进行衡量,如表 5-3-2 所示。

表 5-3-1 用于衡量分类问题模型能力的常用指标

指 标	公 式	解 释
准确率	$\frac{TP+TN}{TP+TN+FP+FN}$	TP(True Positive):正确预测为正类的样本数。 TN(True Negative):正确预测为负类的样本数。 FP(False Positive):负类样本被错误地预测为正类数。 FN(False Negative):正类样本被错误地预测为负类数。 预测正确的数量占全部样本数量的百分比
精确率 Precision	$\frac{TP}{TP+FP}$	预测正确的正样本数量占实际全部预测为正样本数量的百分比,避免误判
召回率 Recall	$\frac{TP}{TP+FN}$	预测为正确的正样本数量占全部真正为正样本数量的百分比,避免漏判
$F1$ 值	$\frac{2Precision\times Recall}{Precision+Recall}$	精确率与召回率的调和平均,综合防误判、漏判的能力

表 5-3-2 用于衡量回归预测问题模型能力的常用指标

指 标	公 式	解 释
均方误差(MSE)	$\frac{1}{n}\sum_{i=1}^{n}(y_i-\hat{y}_i)^2$	y_i:第 i 个样本的真实值。 $\hat{y}_i$:第 i 个样本的预测值。 n:测试集中的样本总数。 预测值与真实值的平均平方差(惩罚大误差)
R^2 分数	$1-\frac{\sum(y_i-\hat{y}_i)^2}{\sum(y_i-\bar{y})^2}$	$\bar{y}$:真实值的平均值。 模型预测相比均值预测的提升比例(最高为 1)

在选择机器学习模型时,实际需要考虑多个因素,包括模型的性能、复杂度和可解释性。性能通常使用准确率、精确率、召回率等指标衡量,复杂度通过模型参数数量和计算复杂度衡量,可解释性对于某些应用场景(如医疗、金融)非常重要。此外,训练时间和资源需求也是需要考虑的因素。在实际应用中,数据的质量与数量对模型的性能有重要影响,更多的数

据可以提高模型的泛化能力，但存在边际效应，即当数据量增加到一定程度后，每新增同等数量的数据，对模型性能的提升效果会不明显，甚至反而减弱。干净、准确的数据对模型性能至关重要，噪声和异常值可能损害模型的泛化能力。训练数据和测试数据应当来自相同的分布，否则模型的泛化能力也会受到影响。

5.4　AI 辅助编程

随着人工智能技术的飞速发展，AI 编程助手已成为提升开发效率、优化代码质量的重要工具。国内科技企业相继推出了一系列各具特色的 AI 编程平台，推动软件开发向智能化、自动化迈进。本节将简要介绍腾讯、字节跳动、阿里巴巴等厂商的主流 AI 编程平台。

1. 平台概览

在当前国内的 AI 编程助手领域，主要呈现出腾讯、字节跳动、阿里巴巴等互联网公司多足鼎立的竞争格局。这些平台旨在通过智能化手段简化开发流程，降低编程门槛，并覆盖从代码补全、智能生成到全流程自动化开发的各种场景。这些工具不仅能自动完成重复性的编码任务，还能深入理解项目上下文，提供精准的技术方案，甚至尝试独立完成从需求分析到部署上线的部分开发工作，从而提升开发效率。

2. 各大平台的详细解析

1）腾讯的 CodeBuddy

腾讯云推出的 CodeBuddy 定位为“AI 全栈工程师”。它不仅是一个代码补全工具，更致力于成为全栈开发体验的核心。

其显著特点在于整合了“产设研”一体化的能力，例如支持 Figma 设计稿一键转化为生产代码，并集成了诸如 Supabase 等 BaaS（后端即服务）解决方案，实现了零配置的数据库和身份验证服务。这使得 CodeBuddy 能够服务于更广泛的用户群体，包括设计师、产品经理等非传统编程人员，助力他们参与甚至主导应用开发流程。

2）字节跳动的 Trae

字节跳动的 Trae 经历了从 MarsCode 到 Trae 的升级，并已成为国内首款 AI 原生集成开发环境（AI IDE），并在 2.0 版本中推出了 SOLO 模式。

SOLO 模式代表了 AI 编程辅助的一个飞跃。在此模式下，Trae 集成了编辑器、终端、浏览器和文档 4 个工具面板，AI 能自主执行开发任务，打通从需求分析、原型设计、界面开发、后端逻辑、优化调试到构建部署的完整链路。用户通过自然语言提出需求，AI 可生成产品需求文档、编写代码、调试并最终部署到云端。Trae 还支持多模态交互，例如上传 UI 设计图或草图即可生成前端页面代码，大幅提升了前端开发的效率。

3）阿里巴巴的通义灵码

阿里巴巴推出的通义灵码则更侧重于企业级开发场景。它深度对接阿里云生态，在服务大型项目、团队协作，尤其是在 Java 技术栈方面表现出色。其核心优势在于对中文语义的精准理解、对国内开发规范的良好适配，以及对企业级数据安全和流程合规需求的满足。通义灵码支持代码补全、智能问答、代码解释、单元测试生成、异常排查等多种功能，并能很好地适应金融、政务等对安全性和合规性要求极高的行业场景。

3. 其他值得关注的平台

除了上述三大平台,国内还有其他一些值得了解的 AI 编程工具。

(1) Coze 扣子平台(字节跳动)。这是一个零代码/低代码的 AI 智能体开发平台。

它支持通过可视化编排的方式,组合插件、知识库和工作流来创建 AI 应用(如聊天机器人、自动化运营工具等),大幅降低了 AI 应用开发的门槛。

(2) 百度文心快码(Baidu Comate)。基于文心大模型,结合了百度多年的编程现场大数据。

它提供代码补全、注释生成、代码解释、优化建议等功能,并支持超过 100 种编程语言和多种主流 IDE(Integrated Development Environment,集成开发环境)。

(3) 深度求索(DeepSeek)。其优势在于强大的逻辑推理和多文档交叉分析能力,适用于代码逻辑验证、技术方案设计等需要深度思考和分析的场景。

对于非计算机专业的学生和初学者,熟悉并善用这些 AI 编程平台,能够更高效地理解代码、完成实践项目、验证学习成果。并在使用过程中不断加深对编程思维、算法和系统设计原理的理解,以便更好地面向未来人机协同的世界。

【例 5-4-1】 登录 DeepSeek 平台,输入提示词"请你帮我写一个能让网页播放烟花的程序,要求鼠标单击哪里就在哪里放出烟花,烟花的大小随机,持续时间(8～10)s 随机",如图 5-4-1 所示,DeepSeek 会根据要求写出网页代码,单击"运行"按钮可以直接展示代码运行结果,如果有修改要求,可以继续写提示词进行修改,读者可以自行尝试。

图 5-4-1 AIGC 辅助代码生成及演示

5.5 回归分析

回归分析是用于预测连续数值的机器学习方法,它通过样本数据学习目标变量和自变量之间的相关关系,建立数学表示模型,利用该模型可以基于新的自变量数据,预测相应的目标数据。根据因变量和自变量的个数不同分为自变量只有一个的一元回归和自变量有多个的多元回归;根据因变量和自变量的相关关系不同分为自变量与因变量是线性关系的线

性回归和自变量与因变量是非线性关系的非线性回归，包括多项式回归、指数回归、对数回归等。广泛应用于经济预测、工程优化等领域，核心价值在于将复杂数据关系转换为可量化的趋势判断。

回归分析预测过程一般需要以下步骤。

(1) 根据预测目标，确定自变量和因变量。

(2) 建立回归预测模型并评估模型质量。

依据自变量和因变量的历史数据进行计算，通过合理地调整参数，建立合适的回归分析方程，即得到回归分析预测模型。

(3) 计算并确定预测值。

利用回归预测模型，输入新的自变量值，计算预测值，并计算预测值的置信区间。

回归分析模型的性能评估，可以通过均方误差(Mean Squared Error，MSE)、均方根误差(Root Mean Squared Error，RMSE)、R^2 值和 P 值等指标完成，其中 $\mathrm{RMSE}=\sqrt{\mathrm{MSE}}$。

R^2 值和 P 值是用于评估模型质量的两个关键指标，其中 R^2 值用于衡量模型对数据变化的解释能力，取值为 0～1，越接近 1，说明模型的拟合程度越好。

P 值表示回归系数的显著性，即响应变量 Y 受回归变量 X 影响，P 值越小，说明 Y 变量受 X 变量影响越大，即回归系数越显著。例如，P 值小于 0.0001 说明回归模型具有统计显著性。如果 P 值大于 0.05 则说明该回归分析中的响应变量 Y 和回归变量 X 无关。

5.5.1　一元线性回归

一元线性回归的自变量只有一个，其核心是找到一个能够描述自变量 x 与因变量 y 之间直线关系的函数：$y=w_0+w_1x$，其中，w_0 和 w_1 是回归系数。对于线性回归线来说，w_0 表示直线的截距，w_1 表示直线的斜率。

【例 5-5-1】 某公司引进人才后效益不断提升，员工的工资也在同步增长，具体变化如图 5-5-1 左侧的表格所示，请根据以往 10 个月的效益增长率和员工平均工资收入(简称员工工资)，预测员工以后的收入。

公司效益增长率（%）	员工工资（元）
1.2	5200
2.5	5800
3.8	6200
4	6500
5.5	7100
6.3	7500
7	7800
8.2	8200
9.1	8600
10	9000

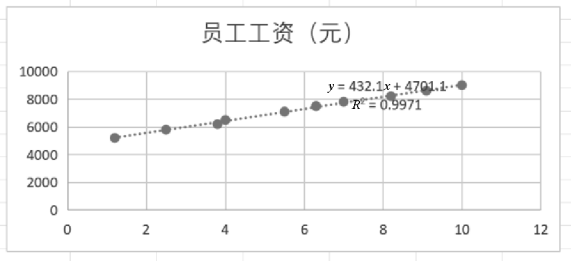

图 5-5-1　利用 Excel 工具建立线性回归模型

(1) 根据预测目标，确定这个问题中的自变量和因变量。

在本例中，预测目标是未来的员工收入，公司效益增长率可以确定是自变量 x，员工平均工资收入(简称员工收入)则是因变量 y。

(2) 建立回归预测模型。

解法 1：利用 Excel 工具。

在 Excel 中输入数据,建立 x 和 y 的散点图,利用“图表工具”中的“设计”选项卡,“添加图表元素”,添加“趋势线”,利用“其他趋势线选项”打开“设置趋势线格式对话框”,在“趋势线选项”中选择“线性”,并勾选“显示公式”“显示 R 平方值”,得到如图 5-5-1 右侧所示的预测模型 $y=432.1x+4701.1$,相应的 $R^2=0.9971$,说明模型拟合得很好。

解法 2:利用 Tableau 工具。

在 Tableau 中连接数据源后,将公司效益和员工工资分别放入列和行,取消聚合建立散点图,并将“分析”选项卡中的“趋势线”拖曳到工作区,选择线性,产生趋势线,光标放置在趋势线上,可以看到回归模型公式,以及 R^2 值、P 值,如图 5-5-2 所示。

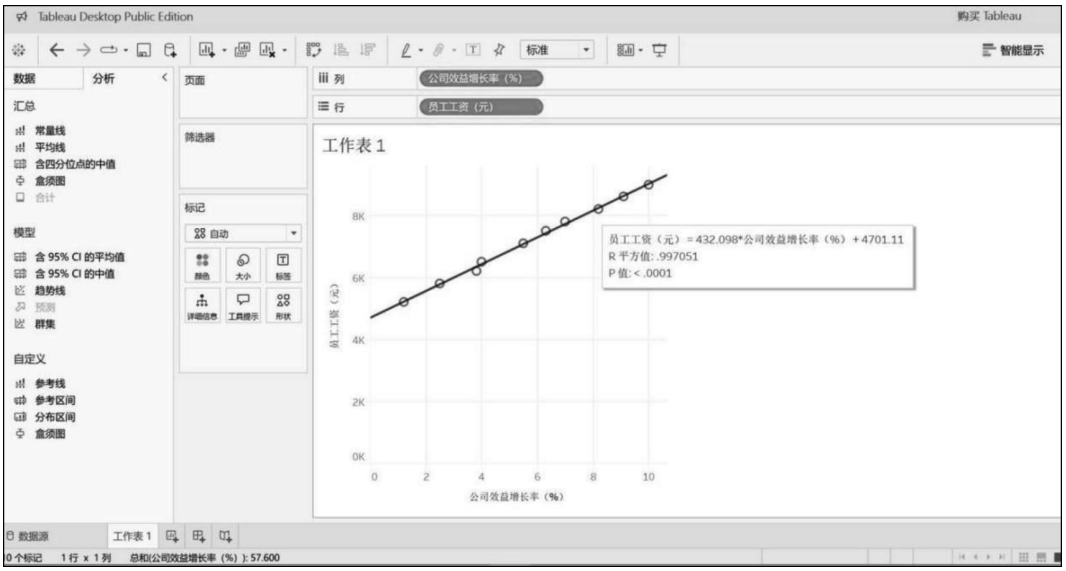

图 5-5-2 利用 Tableau 工具建立线性回归模型

由于不同的工具的计算精度不同,回归公式中的斜率和截距参数值会略有误差。

解法 3:利用 Python 工具。

可以借助大模型工具,如 DeepSeek,用于提供数据文件,请它产生对应的 Python 代码,并复制到 Anaconda 中的 Spyder 编辑环境中运行,结果如图 5-5-3 所示。

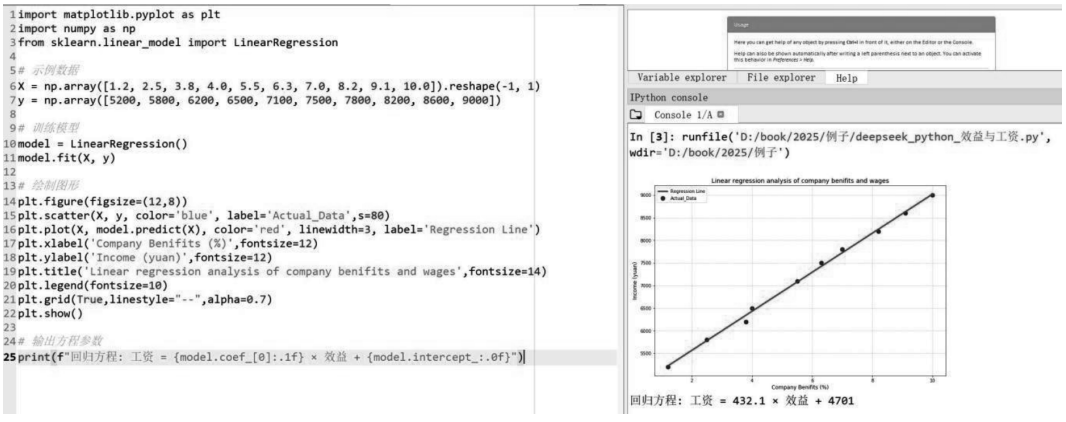

图 5-5-3 利用 Python 代码生成线性回归模型

在图 5-5-3 中，左边是建立本次预测分析的回归模型的 Python 代码，其中在第 3 行引入了 sklearn 中的线性回归库，第 6 和 7 行将自变量和因变量的历史数据以数组形式提供给 x 和 y 变量，第 10 行的作用是将模型初始化为线性回归，第 11 行表示进行模型训练，第 14～22 行表示将模型以图的形式绘制和展示，第 25 行将回归方程输出。执行该程序代码后，在右侧的窗口中，会显示回归模型。

如何获得最佳的回归线（即求解 w_0 和 w_1 的值）呢？最小二乘法是用于拟合回归线最常用的方法。对于样本数据，最小二乘法通过最小化每个数据点到线的垂直偏移距离来计算最佳拟合线，如图 5-5-4 所示。sklearn 中的 Linear Regression()方法已经完成了这项基础工作，直接调用即可初始化模型。

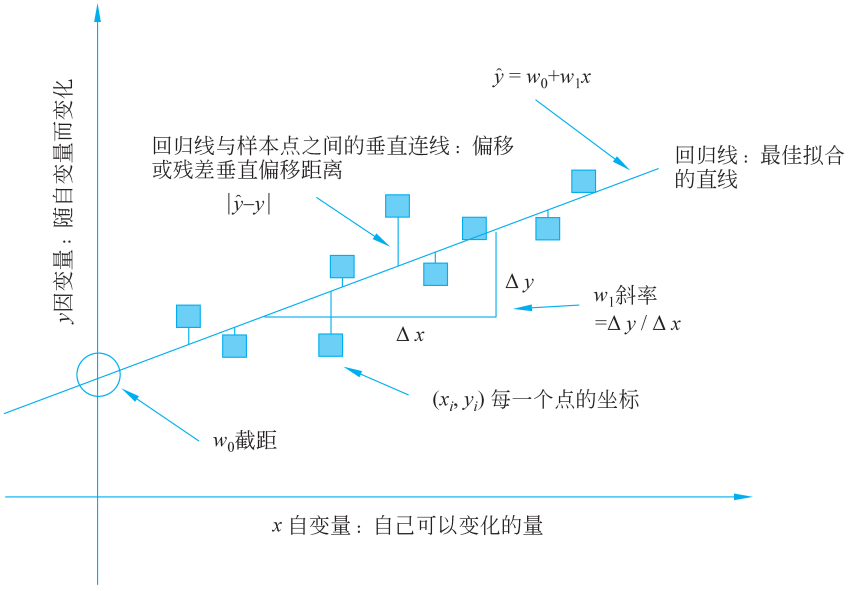

图 5-5-4　通过最小二乘法计算最佳拟合线

（3）利用回归模型进行预测。

有了回归模型公式，将新的 x 变量值，代入公式，便可计算得到预测的 y 值，例如，第 11 个月公司的效益增长是 10.2%，代入公式，$y=432.1\times10.2+4701.1$，计算可得到预测的员工工资为 9108.52 元。

5.5.2　一元多项式回归

一元多项式回归是线性回归的扩展形式，用于描述因变量（Y）与自变量（X）之间的非线性关系。它通过引入自变量的高次项（如 X^2、X^3），将简单的直线拟合升级为曲线拟合，从而更准确地捕捉现实世界中复杂的数量关系，多项式回归的关键特征如表 5-5-1 所示。

表 5-5-1　多项式回归的关键特征

特　点	说　明	生 活 案 例
非线性拟合	用曲线代替直线描述关系	例如学习时间与成绩：先升后降的倒 U 形曲线
多项式阶数	方程中最高次项的次数（degree）	例如二次多项式：$Y=aX^2+bX+c$