

Epidemiological Study of Hepatitis B Virus Genes: Experimental
Techniques and Data Analysis

乙型肝炎病毒基因流行病学研究： 实验技术与数据分析

马偲航 著



International
Union of Scientific and
Technological Scholars

CIP INFORMATION

Title: Epidemiological Study of Hepatitis B Virus Genes: Experimental Techniques and Data Analysis

题目：乙型肝炎病毒基因流行病学研究：实验技术与数据分析

Author: Sihang Ma

作者：马偲航

ISBN: 979-8-90040-187-4

Publisher: CYPEDIA INTERNATIONAL UNION OF SCIENTIFIC AND
TECHNOLOGICAL SCHOLARS

Price: \$60.00

定价：60 美金



**International
Union of Scientific and
Technological Scholars**

**Epidemiological Study of Hepatitis B Virus Genes: Experimental
Techniques and Data Analysis**

乙型肝炎病毒基因流行病学研究：实验技术与数据分析

Address: 100 N Howard St Ste R, Spokane, Wa, 99201. United States

Tel: +888-754-5177

Email: contact@cypedia.net

Format: 170mmx240mm 1/16 Sheets: 19.25

Word Counts: 120.000

Edition: First edition in June 2026

Printing: First printing in June 2026

**CYPEDIA INTERNATIONAL UNION OF SCIENTIFIC AND TECHNOLOGICAL
SCHOLARS**

非开放获取·电子版请联络出版社购买·版权归出版社所有·侵权必究

Not Open Access. For the electronic version, please contact the publisher for purchase. The copyright belongs to the publisher.
Infringement will be prosecuted.

目录

CONTENTS

第一部分 研究背景	1
第 1 章 乙型肝炎病毒研究概况	1
1.1 乙型肝炎病毒流行病学概况	2
1.2 乙型肝炎病毒生物学特征	7
1.3 基因组结构	10
1.4 编码蛋白及其主要功能	11
1.5 基因变异类型及生物学意义	14
本章小结	17
第 2 章 基因流行病学概述	19
2.1 基因流行病学发展史	19
2.2 基因流行病学的应用类型	20
2.3 人群水平数据在个体疾病风险评估中的价值	22
2.4 多基因风险评分的偏倚与影响因素	22
2.5 基因流行病学核心研究方法体系	23
本章小结	27
第 3 章 实验设计与数据准备	29
3.1 研究选题：以假说为导向	29
3.2 研究设计与对照设置	29
3.3 样本量估算	41
3.4 实验材料选型	44
3.5 样本采集、保存与预处理规范	45
3.6 数据整理与质控管理	47
3.7 控制生物安全风险：排除、降低、保护	51
本章小结	51

第二部分 获取基因数据	53
第 4 章 检测血液中的 HBV	53
4.1 HBV 血清标志物检测	53
4.2 HBV-DNA 提取技术	57
4.3 核酸纯度检测技术	59
4.4 HBV-DNA 定量检测技术	61
本章小结	63
第 5 章 基因扩增与测序技术	65
5.1 PCR 引物设计原则与验证	66
5.2 PCR 扩增技术在 HBV 基因研究中的应用	74
本章小结	81
第 6 章 乙肝病毒基因信息检索与数据库应用	83
6.1 主流核酸数据库	83
6.2 HBV 特化数据库	86
6.3 HBV 基因研究相关实用工具	88
6.4 数据库数据格式转换与整理	90
6.5 数据引用规范与版权注意	93
本章小结	94
第 7 章 基因数据质控与预处理	95
7.1 测序产物位置核对	95
7.2 氨基酸序列数据清洗	96
本章小结	98
第三部分 读取突变位点	99
第 8 章 乙肝病毒基因序列比对与分型分析	99
8.1 序列比对核心原理与工具 (Clustal/MEGA 等)	99
8.2 同源序列比对实操步骤	100
8.3 HBV 基因分型标准与判定方法	104

8.4 比对结果可视化呈现	109
本章小结	116
第 9 章 HBV 基因突变检测方法	119
9.1 突变效应量化	120
9.2 突变分析软件工具	120
9.3 常规突变检测方法（限制性酶切 / 高分辨率熔解曲线）	121
9.4 参考序列的建立	124
9.5 测序法突变检测与结果解读	127
9.6 使用生物信息学工具预测筛选突变位点	139
9.7 不同检测方法优劣对比与适配场景	141
本章小结	142
第四部分 基础统计和可视化	143
第 10 章 R 语言基础绘图及检验	143
10.1 频数堆积图	143
10.2 堆积图可视化组间差异卡方检验	146
10.3 箱线图及组间差异检验	148
10.4 小提琴图合并箱线图	151
本章小结	154
第 11 章 描述性统计：基础数据特征	157
11.1 基线数据统计	157
11.2 森林图	162
本章小结	165
第五部分 病毒进化的方向和速度	167
第 12 章 系统发育和分子进化	167
12.1 进化树构建核心原理（距离法 / 最大似然法）	168
12.2 常用建树工具使用方法	170
本章小结	175

第 13 章 环境压力和进化方向	177
13.1 进化速率和种群动态分析	177
13.2 使用 Python、R、MEGA 软件计算核苷酸多样性 π 值	179
第 14 章 突变对病毒适应性的影响	187
14.1 突变类型与病毒适应性的相关性	187
14.2 预测蛋白质结构并标记突变位点	195
本章小结	197
第六部分 基因突变的功能后果	199
第 15 章 风险分析和诊断标志物	199
15.1 风险因素分析	199
15.2 挖掘生物标志物	214
15.3 突变位点基础特征分析	226
第 16 章 乙肝病毒基因突变与临床表型关联	237
16.1 突变与临床表征的关联	237
16.2 突变位点之间的相关性	260
16.3 中介效应量化	268
16.4 关联分析的统计学方法应用拓展	274
16.5 分子流行病学视角下的突变研究	276
本章小结	281
第七部分 基因变异分析的前沿方法	283
第 17 章 补充论证实验	283
17.1 基于反向遗传学的基因突变功能验证	283
17.2 病毒变异与进化的驱动因素	286
第 18 章 基因变异研究的前沿方法	287
18.1 宏基因组学分析技术	287
18.2 人工智能在突变分析和疫情预测中的作用	288
本章小结	290
参考文献	291

第一部分 研究背景

第 1 章 乙型肝炎病毒研究概况

病毒在全球造成了巨大的健康和经济损失，并深刻影响着人类历史的走向。病毒与宿主、环境间的错综复杂的互作关系，始终是生命科学与公共卫生领域持续深耕的核心议题之一。乙型肝炎病毒（Hepatitis B virus, HBV，后文简称“乙肝病毒”或“HBV”）正是这类具有复杂变异特征的病毒代表。世界卫生组织（WHO）提出 2030 年消灭乙肝计划，目前看来已经不可能实现。从我们 2024 年 -2025 年在某三级甲等医院进行的孕妇 HBsAg 携带情况调查来看，要实现消灭乙肝的目标仍然任重道远。基因突变使乙肝感染的防控和治疗出现很多不确定性，其间的相关性还需要继续研究。

HBV 基因变异的客观存在，也为当前乙肝防控策略的优化完善提供了重要的切入点与思考方向。中国作为人口大国和传统乙肝大国，取得目前的成就已经殊为不易。现有防控工作和监测体系已形成卓有成效的基本框架，但在覆盖范围与监测精度上仍存在提升空间。当前监测网络多以医疗资源集中的城市区域为重点，广大乡镇及农村地区的人群监测尚有不足，导致该部分人群中 HBV 的流行规律、基因变异谱特征缺乏系统且精准的数据支撑。尤为关键的是，针对 HBV 基因变异株的生物学效应研究仍有待深化，如，表达传统疫苗靶向蛋白的 S 基因变异是否会对现有疫苗保护效果产生影响、逆转录酶区耐药突变是否会降低一线抗病毒药物的临床应答率、新型变异株的传播能力与致病特征是否发生改变等核心科学问题，目前仍相对缺乏基于大样本流行病学调查的循证医学证据。

需要客观认识到的是，防控策略的优化往往受制于现实的卫生经济学约束。从全球公共卫生实践来看，医疗资源的投入通常倾向于具有显著成本效益与社会收益的领域，而 HBV 基因变异的精准监测与相关研究，因其技术门槛较高、短期收益不明显，较难获得持续性的大规模资金倾斜。当前医保体系的可持续运行正面临较大压力，叠加区域经济发展的不均衡性，使得三线之下城市及乡镇地区

的人群，在接受高成本的精准检测与个体化治疗时存在现实负担。这种“技术需求—资源供给”之间的矛盾，进一步加剧了 HBV 基因变异相关数据的缺口，也让现有防控策略在应对基因变异挑战时，难以形成兼顾科学性与可行性的实施方案。由此可见，未来的研究方向不仅需要深化变异株与防控及治疗策略的关联性探索，更需要探索低成本、高通量的监测技术与普惠性的干预方案，在有限的资源条件下，实现防控效能的最大化。

基因研究的几个相关领域各有侧重。分子流行病学的探索，能够揭示病毒传播动力学、宿主适应与地理扩散的核心规律；测序与生物信息学技术的进步，为疫情追踪、菌株监测提供了有力工具，助力厘清病毒流行背后的遗传多样性、起源与传播路径。基因组分析是破译病毒遗传学特征、鉴定毒力因子、指导靶向治疗等领域的基础。进化动力学的研究，则能阐明宿主内病毒适应的演变过程，解析抗原变异、免疫逃避与耐药性出现的核心机制，为防控策略的制定提供科学依据。基因变异深刻影响着病毒传播、宿主适应、毒力演变等方面，进而影响临床诊疗效果与疫苗保护效力。研究基因变异的作用结局，探索变异的分子机制、与流行病学的关联，分析预测进化规律，既是回应全球病毒性疾病预防挑战的需求，也是病毒研究核心范式在具体病原体探索中的生动实践。以基因分析为基础，融合分子流行病学、进化生物学、免疫动力学等领域形成兼具宏观和微观视角的研究框架，为解析病原体遗传变异的调控机制、阐释从个体宿主到群体范围的系统发育规律，提供了关键支撑，也成为应对全球病毒性疾病预防挑战的重要研究范式。

本章作为全书的开篇，将立足传统病毒研究领域，概述已充分研究的病毒特征。同时，介绍一些 HBV 相关的核心概念，为后续基因分析等实操内容奠定理论基础。无论是初次接触 HBV 研究的圈外人士，还是需要系统梳理基础理论的相关研究人员，都能通过本章内容建立对 HBV 研究的整体认知框架。

1.1 乙型肝炎病毒流行病学概况

乙肝病毒感染导致急性和慢性乙型肝炎。据世界卫生组织（WHO）发布的《2024 年全球肝炎报告》^[1]，全世界有 2.54 亿人患有乙型肝炎（简称“乙肝”），5000 万人患有丙型肝炎，每天有 6000 人新感染病毒性肝炎。病毒的持续存在，以及宿主对受感染细胞的慢性免疫反应驱动持续的肝损伤，可能导致肝硬化

（cirrhosis）、终末期肝病（end-stage liver disease）和肝细胞癌（hepatocellular carcinoma, HCC），目前尚无可以彻底清除病毒的治疗方法^[2]。在我国 60% 以上肝硬化患者和 80% 以上肝癌患者，主要病因是 HBV 感染^[3]。

在很多国家，乙型肝炎并没有得到很好的预防、诊断和治疗。截至 2022 年底，慢性乙型肝炎感染者的诊断率和抗病毒治疗率分别是 13% 和 3%，在中国诊断率和治疗率分别为 58.78% 和 17.33%^[4]。新增乙肝感染的途径主要是母婴传播以及不安全的输血、注射行为。我国高度重视乙肝防控，并已取得显著成效。自 1992 年启动新生儿乙肝疫苗接种项目，一般人群乙肝病毒表面抗体（HBsAg）阳性率从 1992 年的 9.72% 下降至 2024 年的约 5%，这代表着已经从高流行区下降到接近中低流行区。其中，5 岁以下儿童的 HBsAg 阳性率从 9.67% 降至 0.30%。乙肝疫苗计划免疫估计避免了约 4000 万例 HBV 新发感染和 700 万例相关死亡^[5]。

1.1.1 传播途径

HBV 感染的主要形式为肠外传播，主要传播方式包括母婴传播、血液传播及性接触传播，感染后可表现为急性自限性感染或慢性持续性感染两种临床类型。在我国，新发病例的传播途径以母婴传播为主，该途径所致的新发感染占比达 40%~50%^[6]，且感染多发生于围生期，具体通过 HBV 阳性母亲的血液及（或）体液实现垂直传播。实验室安全也需注意，离心、振荡等操作会产生大量感染性气溶胶，虽然目前为止尚无呼吸道传播的报道。此外，尚无证据表明 HBV 可通过吸血昆虫传播^[7]。关于 HBV 的传播细节，目前已知 HBV 可通过破损的皮肤与粘膜侵入机体，但皮肤粘膜的破损程度阈值、引发感染所需的最低病毒载量等关键参数尚未有明确界定。临床中还存在特殊情况，少量病毒侵入后可能仅诱导机体产生抗-HBs 阳性，随后被免疫系统自发清除，未形成显性感染，这种情况易被误判为未发生传播。

疫苗接种是防控乙型肝炎的核心手段，接种覆盖人群主要包括新生儿、婴幼儿、15 岁以下未接受免疫的人群以及成年高危人群^[8]。其中，新生儿接种被纳入国家免疫保护计划，无论母亲 HBsAg 检测结果为阳性或阴性，均需按规范完成接种。临床中常规采用重组酵母乙型肝炎疫苗，剂量为 10 μg/剂次^[9]。对于母亲为 HBsAg 阳性的新生儿，需在出生后 12 小时内尽早注射乙型肝炎免疫球蛋白（Hepatitis B immunoglobulin, HBIG），并在不同部位同步接种乙型肝炎疫苗，

通过主动免疫与被动免疫结合的方式提升保护效果^[11]。

需要注意的是，即便采取了上述联合免疫策略，仍有 5% ~ 10% 的 HBsAg 阳性且 HBV DNA 高水平的母亲所分娩婴儿会发生 HBV 感染。研究表明，此类免疫失败的危险因素主要包括母亲 HBsAg 阳性状态、HBV DNA 高载量、病毒准种特征以及基因的特定变异类型等^[11]。基于此，针对高风险人群的强化免疫策略尤为重要：对于母亲为 HBsAg 阳性的儿童，或 7 月龄时检测显示抗-HBs 水平偏低的儿童，建议在 2 岁前及时开展加强免疫接种，无需等待抗-HBs 阴转后再进行强化，以确保免疫保护的连续性与有效性^[12]。

1.1.2 基因型及基因亚型分布

乙型肝炎病毒（HBV）感染后的临床结局存在显著个体差异，这一差异与感染病毒的基因型密切相关^[13]。目前，学界以全基因序列异质性 > 8% 为核心判定标准，将 HBV 划分为 A-J 共 10 个基因型，每个基因型下又可进一步细分出多个具有独特序列特征的亚型（具体亚型分类详见表 1）^[14]。

从中国范围内的 HBV 基因型分布格局来看，B、C、D 基因型为主要流行株，A 基因型与 I 基因型则较为罕见，仅在局部地区的特定人群中被检出。这种分布格局呈现出明显的地域聚集特征：C 基因型在我国北方地区占据主导地位，人群感染比例约为 52%；B 基因型则集中流行于南方地区，感染比例可达 43% 左右。值得注意的是，D 基因型属于全球性流行基因型，在我国的分布具有鲜明的区域针对性，主要集中在西藏自治区及宁夏回族自治区等少数民族聚居区域^[15]。

对于局部研究，可能会发现和总体分布不同的分布规律，因为 HBV 不同基因型及亚型的具体分布受到多重因素的综合影响。一方面，人群的地理分布格局、历史迁徙路径等人文地理因素对病毒传播范围的界定起到关键作用，不同地区的人口流动与融合过程直接影响了基因型的跨区域扩散；另一方面，基因型自身的生物学特性也参与了分布格局的塑造，例如不同基因型在传播能力（如体液传播效率、母婴传播概率）、致病性强弱等方面存在固有差异，这些差异会间接影响其在人群中的流行优势。已有多项临床研究证实，在各类 HBV 基因型中，C 基因型与严重肝病（如肝硬化、原发性肝癌）的发生风险关联更为紧密，是诱发不良临床结局的重要风险基因型^[13]。同时，相较于其他基因型，C 基因型 HBV 还具有更强的母婴传播倾向性，这一特性也在一定程度上强化了其在部分地区的流行优势^[13]。

除地域差异外，HBV 基因型分布还呈现出显著的民族特异性。郭晓琪等^[14]

通过多阶段分层整群随机抽样结合系统抽样的方法，从 2020 年全国乙型肝炎普查所建立的 HBsAg 阳性样本库中抽取样本进行专项研究，揭示了不同民族群体中的 HBV 基因型分布特征。研究结果显示：在 B 基因型感染方面，汉族（49%）、彝族（53%）、苗族（94%）及布依族（81%）人群的感染占比较高，其中苗族人群的 B 基因型感染比例尤为突出；C 基因型下属亚型的分布具有鲜明民族指向性，C2 亚型在汉族、藏族、彝族、维吾尔族、蒙古族等多个民族中占比居高，C5 亚型则主要集中在瑶族与壮族人群中；C/D 重组型基因型为藏族人群的优势流行株，在该民族中的占比显著高于其他民族；D 基因型的亚型分布同样存在民族差异，彝族人群感染的 D 基因型均为 D3 亚型，而维吾尔族与哈萨克族人群感染的 D 基因型则全部为 D1 亚型；I 基因型在我国的分布范围较窄，主要见于壮族人群，且该民族所感染的 I 基因型均为 I1 亚型。

表 1 HBV 基因型及亚型主要分布地区^[12]

Table 1 Global distribution map of HBV genotypes and subtypes^[12]

基因型	亚型	主要分布地区
A	A1	非洲、亚洲、拉丁美洲
	A2	欧洲、北美
	A3-A8	非洲
B	B1	日本南部
	B2	东南亚和中国部分地区
	B3	西太平洋群岛及印度尼西亚等
	B4-B10	东南亚
C	C1	南亚和东南亚
	C2	中国和朝鲜半岛
	C3,C4	美拉尼亚和澳大利亚
	新亚型	印尼、菲律宾、中国云南
D	D1-D3	欧洲及地中海地区
	D4	波利尼西亚和密克罗尼西亚
	D5-D9	印度、突尼斯、尼日尔等
E		西非
F、H		中南美洲、加勒比地区
G		欧洲、美洲、非洲、亚洲
I		越南、老挝、印度东部、中国西南部
J		日本、琉球

1.1.3 基因重组、共感染

在 HBV 感染相关研究中，共感染现象逐渐成为关注焦点。自然界中普遍存在多种病原体同时或连续侵袭同一宿主的情况，这种现象被定义为共感染（coinfection）^[15]。在 HBV 感染场景下，基因重组的发生主要源于不同基因型 HBV 的共感染，该过程是推动 HBV 进化的关键动力，但截至目前，HBV 基因重组的具体分子机制仍未完全明确^[16]。随着生物信息技术与软件分析技术的快速发展，HBV 基因重组现象的研究深度与广度得到显著提升。从全球分布特征来看，B/C 型与 C/D 型重组体是最为常见的重组类型，这一分布特点与 B、C 基因型在亚洲地区的高流行率密切相关。其中，B/C 重组体主要流行于东亚及东南亚地区，C/D 重组体集中分布于印度及我国西北地区，A/D 重组体则主要见于意大利和南非（具体分布详见图 1）^[17]。此外，还存在其他类型的 HBV 重组体，但出现频率相对较低。

国内学者潘登等^[18]借助 GenBank 数据库中的 HBV 序列资源，结合 Simplot 软件开展了全国范围的 HBV 重组体筛查研究，结果显示 HBV 重组率高达 46%（755/1642），共检出 31 种重组体，且所有重组体均涉及 C 基因型的参与。具体重组类型分布为：B/C 型重组体占比最高，达 71%（22/31）；C/D 型重组体占比 16.1%（5/31）；C/A 型重组体占比 9.7%（3/31）；C/I 型重组体占比最低，仅为 3.2%（1/31）。这一研究结果进一步证实，C 基因型 HBV 具有较强的重组倾向性^[19]，而其在我国广泛的分布范围，也为不同基因型间的重组提供了更多机会。

从病毒学机制与临床诊疗视角来看，不同基因型 HBV 之间的共感染、HBV 与其他病毒的共感染等现象，正受到越来越多的关注^[20]。与共感染相关的另一重要概念是重叠感染（superinfection），其定义为宿主反复感染相同或不同种类病原体，由于前次感染病原体未被完全清除，导致后续感染病原体负荷增加、毒力增强的情况^[21]。在临床实践中，若存在家族性乙肝病史或其他相关风险因素，即便初始为 HBV 单感染，也可能因先后多次感染不同来源的 HBV 毒株，形成多种 HBV 株系共存的局面，进而发生重叠感染。

HBV 共感染可能引发多种临床结局，在临床环境中，多数情况下会出现单一病毒占据复制主导地位的现象；但也存在特殊情况，即两种或多种病毒的复制水平处于相近量级，形成共显性^[24]。需要注意的是，共感染的临床检出难度较大，

一方面受限于诊疗过程中的检查项目设置与检查对象选择, 另一方面, 可能因后感染病毒的复制被先感染病毒抑制, 或后感染病毒的存在被高载量的先感染病毒掩盖等因素, 导致共感染毒株的准确分离与检测面临较大挑战^[25]。

在研究技术层面, 传统二代测序技术因存在较大的随机性, 难以满足多基因型 HBV 共感染现象的研究需求, 需采用针对性的特殊检测方法, 例如基于 GSPA (基因组特异性引物扩增) 的特异性探针杂交技术^[22]。已有相关研究证实共感染对 HBV 进化的影响, 如 Baig 等^[26]通过对比 HBV/HDV (丁型肝炎病毒) 共感染与 HBV 单感染患者的 HBV S 基因序列突变差异, 提出共感染产生的选择压力可诱导 HBV S 基因发生特异性进化的观点。

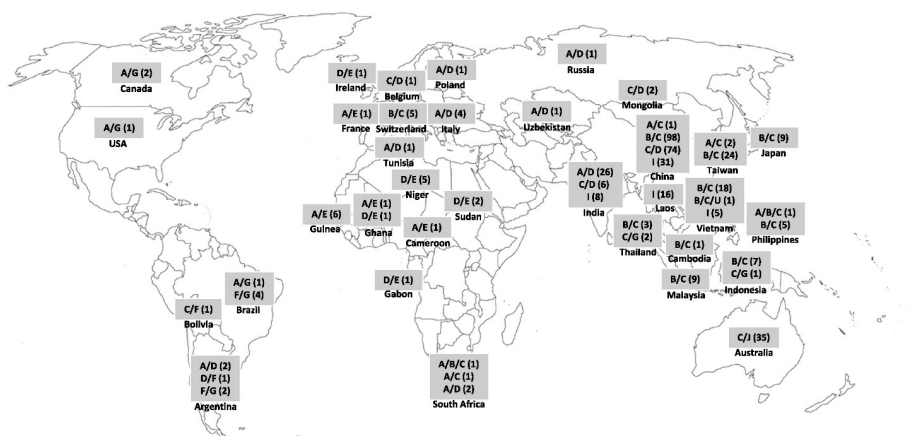


图 1.1 HBV 基因重组全球分布图^[19]

Figure 1.1 Global distribution map of HBV genetic recombination^[19]

1.2 乙型肝炎病毒生物学特征

1.2.1 病毒形态与结构

乙型肝炎病毒属于嗜肝 DNA 病毒科 (*Hepadnaviridae*) 正嗜肝 DNA 病毒属 (*Orthohepadnavirus*)。嗜肝病毒主要靶向感染人体的肝细胞, 其致病机制与肝脏损伤的密切关联。该病毒为有包膜病毒, 呈球形结构, 在电子显微镜观察下, HBV 在感染者的血液中主要以三种形态存在, 分别是大球形颗粒、小球形颗粒和管形颗粒。其中, 只有大球形颗粒具备完整的感染能力, 后两者为病毒复制过

程中产生的无感染性颗粒^[27]。大球形颗粒也被称为 Dane 颗粒（以发现者命名），是 HBV 的完整病毒颗粒，直径约 42nm，其结构从外到内依次为包膜、衣壳和核心基因组，类似“三层包裹”的结构：最外层的包膜源于宿主肝细胞的细胞膜，并非病毒自身合成，包膜上镶嵌着 HBV 的表面抗原（HBsAg），这是病毒识别并结合肝细胞表面受体的关键蛋白，也是临床检测中判断感染的重要标志物；中层为二十面体对称的衣壳（也叫核壳），由病毒的核心抗原（HBcAg）构成，衣壳的作用是保护内部的基因组，同时参与病毒的复制过程；最内层为核心部分，包含了 HBV 的基因组（部分双链环状 DNA）、DNA 聚合酶（兼具逆转录酶活性，这是嗜肝 DNA 病毒科的典型特征）以及一些酶类辅助因子。

小球形颗粒直径约 22nm，是血液中含量最多的 HBV 相关颗粒，同样由 HBsAg 构成，呈中空结构，不含 DNA 和 DNA 多聚酶，因此在具备抗原性的同时却没有感染能力；管形颗粒则是由多个小球形颗粒聚集而成的，直径同样为 22nm，长度在 50~700nm 之间，也无感染性。这两种无感染性颗粒的产生，是由于 HBV 复制过程中，表面抗原的合成量远超过完整病毒颗粒组装所需的量，多余的表面抗原会自行组装形成这两种颗粒释放到血液中。病毒颗粒多样性一定程度上会影响病毒数量检测和宿主免疫反应检测^[28]。

1.2.2 生命周期

HBV 生命周期（life cycle）是治疗靶点挖掘和临床实践生物标志物开发领域的重要研究内容。HBV 的感染和复制过程具有独特性，尤其是其依赖逆转录酶的复制机制，这也是其基因容易发生变异的重要原因之一。首先，HBV 通过包膜上的 HBsAg 识别并结合肝细胞表面的特异性受体。目前已明确的主要受体为牛磺胆酸钠协同转运蛋白（sodium taurocholate cotransporting polypeptide, NTCP）。这一结合过程触发内吞作用以及病毒膜与细胞膜的融合，进而将核壳释放进入细胞质中。在溶酶体的作用下，病毒包膜和衣壳被逐步降解，释放出内部的松弛环状 DNA（relaxed circular DNA, rcDNA），沿微管被转运至细胞核内，并在其中转化为稳定的共价闭合环状 DNA（covalently closed circular DNA, cccDNA）微染色体，这是导致 HBV 持续感染的首要原因^[29]。近年一些研究报道了多种宿主因子参与了 cccDNA 的生成过程。以 cccDNA 为模板，在宿主细胞 RNA 聚合酶作用下生成前基因组 RNA（pregenomic RNA, pgRNA）并被

转运到细胞质，在细胞质内被衣壳包裹。此后，经病毒聚合酶逆转录再次成为 rcDNA。新生成的主要是两个去向，返回细胞核补充进 cccDNA 库，或被包膜包裹后形成新的病毒颗粒分泌到胞外^[30]。

1.2.3 感染自然史、血清学转换

慢性 HBV 感染自然史的划分需综合考量病毒学、生物化学及组织学特征等多方面指标^[31]。HBV 感染后自然病程的发展轨迹主要取决于病毒与宿主之间的相互作用，其中感染 HBV 时的年龄是影响感染慢性化的核心因素之一。开展 HBV 感染自然史模型的相关研究具有重要的临床与公共卫生意义，不仅能够助力明确 HBV 传播规律及再激活风险，还可为提升血液筛查精度与诊断方法灵敏度提供理论支撑。此外，厘清急性自限性感染或持续感染进程中 HBV 标志物的复杂变化规律，对于深入理解抗病毒治疗对慢性乙型肝炎的缓解机制、推进隐性 HBV 感染相关研究均具有关键价值^[31]。

临床上通常将 HBV 感染自然史划分为四个核心阶段，即免疫耐受期、HBcAg 阳性免疫活动期、非活动期以及 HBcAg 阴性免疫活动期^[32]。这一分期标准具有通用性，但并非适用于所有感染者，且各阶段并非必然呈序贯进展。依据 HBV DNA 水平与血清丙氨酸转氨酶（alanine aminotransferase, ALT）水平的分期标准，部分慢性 HBV 感染者无法归入上述任何一个阶段，这类人群被定义为“不确定期”感染者^[33]。已有研究数据显示，按 HBV DNA 与 ALT 水平划分，北美地区约 30%~50% 的慢性 HBV 感染者处于不确定期，我国该比例约为 30%；若进一步结合肝活检开展肝组织病理学评估，不确定期患者的检出比例会进一步升高^[32]。值得关注的是，不确定期患者的免疫分期状态并非固定不变，有报道约半数此类患者在随访过程中会发生分期转换。因此，建立系统性、标准化的 HBV 自然史分期体系，已成为当前 HBV 研究领域的重要方向，其核心目标在于为 HBV 感染者的规范化诊疗提供精准指导。

HBV 感染自然史的各个阶段，可能会发生一种重要的生物学事件，即 HBV 血清转换，其特指病毒抗原自发消失且机体相应抗体产生的过程。其中，HBcAg 血清转换对于慢性 HBV 感染者的病情进展具有重要提示意义，多数情况下，HBcAg 血清转换与患者 HBV-DNA 水平下降及肝病临床症状缓解密切相关。

1.3 基因组结构

HBV 基因组为双链不等长 DNA，全长约 3200 个碱基对（bp），主要由两条链组成：一条为完整的负链（也叫长链、L 链），长度固定为 3200bp，5' 端与病毒的 DNA 聚合酶共价结合；另一条为不完整的正链（也叫短链），长度可变，两条链的 5' 端存在互补的黏性末端，通过碱基配对形成环状结构。为了方便研究和描述，科研人员人为地将 HBV 基因组的负链上，与正链 5' 端互补的位置设定为核苷酸 1（nt 1），并按顺时针方向对基因组进行编号。基于这一编号系统，可将 HBV 基因组划分为四个主要的开放阅读框（ORF）、多个调控序列以及一个复制起点区域。其中，开放阅读框是指基因组中能够编码蛋白质的碱基序列，HBV 的四个 ORF 分别为 S 区、C 区、P 区和 X 区，这四个 ORF 存在大量的重叠序列。这种高度重叠的结构是 HBV 基因组的显著特征，其意义在于能够在有限的碱基长度内编码多种蛋白质，提高基因组的利用效率，但同时也导致了一个重要后果：某一个区域的碱基变异可能会同时影响多个基因的编码，进而产生复杂的生物学效应。

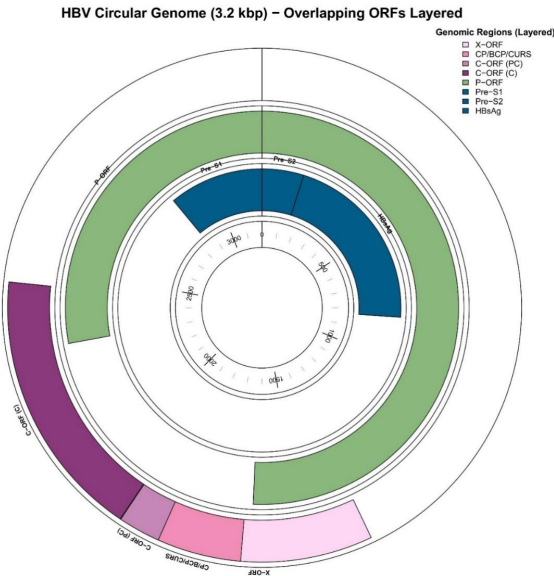


图 2 HBV 基因组环形结构示意图

Figure 2 9The circular diagram of HBV genome

1.4 编码蛋白及其主要功能

1.4.1 HBs 蛋白

HBV-S 基因位于基因组的 nt 155~835 区域, 长度约 680bp, 主要负责编码 HBV 的表面抗原 (HBsAg)。HBs 蛋白的主要功能是作用于 HBV 生命周期以及维持病毒 - 细胞关系^[33]。PreS/S ORF 编码三种不同的、结构相关的包膜蛋白, 它们由不同的起始密码子合成, 分别被称为大 (L)、中 (M) 和小 (S) 蛋白。其中, S 蛋白仅由 S 基因编码, M 蛋白由 preS2 和 S 基因共同编码, L 蛋白则由 preS1、preS2、S 共同编码。L 蛋白介导病毒与细胞膜 NTCP 受体结合。M 蛋白和 S 蛋白主要促进病毒组装和分泌^[34]。此外, HBsAg 是乙肝感染的重要标志物, 临床检测中通过检测血液中的 HBsAg 可判断是否存在 HBV 感染; 同时, HBsAg 也是乙肝疫苗的主要成分, 通过接种疫苗诱导机体产生抗 -HBs 抗体, 可有效预防 HBV 感染。

HBV S 基因多个区域是 T 辅助细胞表位 (T-helper (CD4) epitope): 21~65 aa, 186~197 aa, 215~223 aa。HBV S 基因 99~170 aa 区域被称为主要亲和区 (MHR)。“a” 决定簇 (“a” determinant, 124~147 aa) 是 MHR 中相对保守的区域, 是抗 -HBs 中和结构域的免疫显性决定因子^[35]; 以及 B 细胞表位 (B-cell epitope 100~160 aa)。此外, preS/S ORF 与 Pol ORF 完全重叠, Pol ORF 编码病毒聚合酶, 一种多功能蛋白, 具有逆转录酶, DNA 依赖性 DNA 聚合酶和 RNase H 活性^[36]。a 决定簇首次于 1965 年被描述, 其后, 学者们认为可以根据 a 决定簇上特定位置的氨基酸差异组合定义 HBV 血清学亚型: ayw1、ayw2、ayw3、ayw4、ayr、adw2、adw3、adw4、adrq+、adrq-。CTL (CD8) epitope 位于 206~215 aa。

1.4.2 HBc/HBe 蛋白

HBV-C 基因位于基因组的 nt 1901~2450 区域, 长度约 550bp, 负责编码 HBV 的核心抗原 (HBcAg) 和 e 抗原 (HBeAg)。nt 1814~1901, -10~1 aa 也被划入前 C 基因, 是合成 pgRNA、直接重复序列 1 和 RNA 包膜信号的起始位点。C 基因 (nt1902~2452, 1~183 aa) 编码 e 抗原 (HBeAg) 或 c 抗原 (HBcAg), 具体取决于翻译的开始位置。HBV C-ORF 包含重要编码功能区 and 多个顺式调控

元件。包括多个重要功能区：启动子 aa1； α -螺旋相关区域 aa51-78, aa82-110, aa112-128；B 细胞表位 aa74-89；辅助性 T（Th）细胞作用区 aa1-20, aa28-47, aa50-69, aa72-105；核心蛋白表面暴露区 aa78-82, aa127-133；精氨酸（Arg）富集区；细胞毒性 T 淋巴细胞（CTL）aa141-151；核心蛋白表面暴露区 aa145-153；基因分型差异聚集区 aa147-155。

HBcAg 是一种非颗粒蛋白，存在于患者的血清中，提示病毒处在活跃复制的状态。e 抗原的主要作用是调节宿主免疫反应。HBcAg 诱导免疫耐受，因此与慢性感染有关^[37]。它还可以通过抑制前基因组 RNA 包裹到核衣壳中来调节 HBV 复制^[38]。HBcAg 大部分分泌，但有 15%~30% 成熟 HBcAg 回收回细胞质^[39]。在细胞质中，HBcAg 通过与含 Toll/IL-1 受体（TIR）的蛋白质 Mal 和 TRAM 相互作用并破坏同型 TIR 来抑制 TLR 信号传导^[40]。HBcAg 的功能影响了血清转换和检出窗口期。感染后期 HBcAg 阴性突变体由于免疫逃逸优势更容易存活。HBcAg 和 HBsAg 一样是分泌蛋白，是完整病毒颗粒（Dane 颗粒）的重要组成部分。由于 cccDNA 可能终生存在，抗-HBc 阳性被认为是统计 HBV 暴露率的关键指标。抗-HBc IgM 抗体阳性通常提示近期感染或急性发作，抗-HBc IgG 抗体阳性则提示既往感染或慢性感染。

1.4.3 HBp 蛋白

HBV P 基因长度为 843 aa，因编码 94.2kDa 的病毒聚合酶蛋白（viral polymerase, P），也称 Pol 基因。P 蛋白是一种由 4 个亚结构域组成的多功能蛋白，包含末端蛋白（terminal protein, TP）、间隔区（Spacer）、逆转录酶（reverse transcriptase, RT）和 RNase H（RH）^[41]。RNase H 负责降解逆转录过程产生的 RNA 杂质。

嗜肝 DNA 病毒所特有的末端蛋白 TP 利用病毒前基因组 RNA（pg-RNA）为模板，在 Y63 位点启动 DNA 合成。TP 参与结合或包装 RNA 的过程，这对于病毒的 RNA（比如 pgRNA）的运输、保护或者后续的 DNA 合成前的准备可能很重要。当 P 蛋白合成完成后，TP 区会被移除。Spacer 是连接区，具有适应突变和缺失的灵活性，在病毒复制中也起着关键的谱系结构和功能作用。RT 区是耐药突变热点功能区，也是研究最充分的区域。间隔结构域横跨 P 的氨基酸（aa）

184 至 348（基于 GenBank 参考序列 X02763），夹在 TP 和 RT 之间。HBV P 基因具备很高的基因重叠程度，与 S 区有约 400aa 的重叠。HBV P-S 融合蛋白（HBV Polymerase-Surface Fusion Protein, HPSFP）是一种 43 kDa 的糖蛋白。HPSFP 已在细胞外病毒颗粒、细胞裂解物和亚病毒颗粒中检出，在胞内定位在核周，作用是抑制 HBsAg 分泌^[42]。

1.4.4 HBx 蛋白

X 区位于基因组的 nt 1374~1835 区域，长度约 460bp，负责编码病毒的 X 蛋白（HBV X protein, HBx）。HBx 是一种多功能的非结构蛋白，分子量约 17.5kDa），具有 N-端负调控结构域（N-terminal negative regulatory domain, aa 1-50）和 C-端反活化结构域（C-terminal trans-activating domain, aa 51-154）。HBx 是一种多功能调节蛋白，其核心功能是激活病毒的转录和复制。HBx 可以与宿主细胞的多种转录因子结合，增强 cccDNA 的转录活性，从而促进病毒 mRNA 的合成，进而推动病毒的复制。此外，HBx 还具有调控宿主细胞信号通路、影响细胞周期、抑制细胞凋亡等作用——一方面，它可以通过激活宿主细胞的增殖信号通路，为病毒复制提供有利的细胞环境；另一方面，长期的 HBx 表达可能导致宿主肝细胞的基因表达紊乱，增加肝细胞癌变的风险。

1.4.5 调控序列

HBV 基因组中存在多个调控序列，这些序列不编码蛋白质，但能够调控病毒的转录、复制和组装过程，是病毒生命周期中不可或缺的部分。核心调控序列主要包括以下几种：

1. 启动子：启动子是 RNA 聚合酶结合的位点，能够启动基因的转录。HBV 基因组中存在多个启动子，分别调控不同 mRNA 的转录：① S1 启动子（SP1）：调控前 S1 区和 S 区的转录，产生 2.4kb mRNA；② S2 启动子（SP2）：调控前 S2 区和 S 区的转录，产生 2.1kb mRNA；③ C 启动子（CP）：调控前 C 区和 C 区的转录，产生 3.5kb pgRNA 和 3.5kb mRNA（用于编码核心蛋白）；④ X 启动子（XP）：调控 X 区的转录，产生 0.7kb mRNA。

2. 增强子：增强子是能够增强启动子转录活性的序列，HBV 基因组中主要有两个增强子，分别为增强子 I（Enh I）和增强子 II（Enh II）。Enh I 位于

X 区和 C 区之间，能够增强 CP 和 XP 的转录活性；Enh II 位于 CP 附近，主要增强 CP 的转录活性，对病毒的复制起重要的调控作用。

3. 多聚腺苷酸信号 (polyA 信号)：位于 S 区的 3' 端，负责引导 mRNA 的 3' 端添加多聚腺苷酸尾 (polyA 尾)，这一修饰能够增强 mRNA 的稳定性，促进其翻译。

4. 复制起点：包括负链 DNA 复制起点和正链 DNA 复制起点。负链 DNA 复制起点位于前 S1 区附近，是逆转录酶结合并开始合成负链 DNA 的位点；正链 DNA 复制起点位于 C 区附近，是 DNA 聚合酶开始合成正链 DNA 的位点。

1.5 基因变异类型及生物学意义

1.5.1 突变类型

突变 (mutation) 一词的字面含义即为改变或变异。在遗传学领域，突变指的是生物体遗传物质 (即 DNA 序列) 发生的改变。“突变”是相对的概念，可以是代际对比，也可以是与种群普遍模式对比，取决于特定语境。突变是生物遗传与表型多样性的重要来源，为生物进化提供了基础。部分突变导致整个基因缺失、基因片段缺失、因重要区域发生点突变导致产物功能的丧失等，被称为功能丧失型突变 (loss-of-function mutation)。相反的，突变也可能使基因产物获得新的功能，或原有功能被增强，称之为功能获得型突变 (gain-of-function mutation)。

由 DNA 复制错误导致的突变被称为自发突变 (spontaneous mutation)。自发突变的发生无需研究对象接触任何外界因素或环境诱变剂。当研究对象暴露于某些环境因子时诱发的突变称为诱发突变 (induced mutation)。突变可产生多种不同的效应。部分突变对基因功能无任何影响，这类突变被称为沉默突变 (silent mutation)。点突变 (point mutation) 是指仅影响单个碱基对的突变类型。最常见的核苷酸突变是碱基替换突变，即某一碱基被另一碱基取代。此外，核苷酸的添加 (即插入突变) 或缺失 (即缺失突变) 也会引发突变。有时，某条染色体上的一段 DNA 序列可能转移至另一条染色体，或同一条染色体的其他区域，这类突变被称为易位突变 (translocation)。

1.5.2 准种

由于 HBV 聚合酶缺少校正读码功能, HBV-DNA 复制过程中存在较高的核苷酸突变率, 是其他 DNA 病毒的 10 倍以上, HBV 多聚酶错配率平均为 10^{-4} 到 10^{-5} 。对于慢性乙型肝炎患者, HBV 自然进化过程的突变率为 $1.4 \times 10^{-5} \sim 3.2 \times 10^{-5}$ 次突变/位点/年, 平均突变率 2×10^4 碱基替换/位点/年^[43]。肝移植患者的突变率是慢性乙肝患者 HBV 突变率的 100 倍。HBV 具有较高的复制更新率, 在没有药物干预的情况下, 外周血病毒颗粒浓度为 $10^8 \sim 10^{10}$ 个/ml。外周血病毒半衰期约 1~2 天, 每日产生约 10^{11} 个病毒颗粒^[44]。HBV 突变率因病毒被机体免疫攻击、抗病毒药物作用等因素筛选, 使不同病程中的优势毒株存在差异^[45]。

突变的累积使 HBV 在宿主体内以“准种”的形式存在^[43]。准种是用于描述同种生物遗传异质性的概念, 定义为“不同病毒种群间核酸突变造成的序列长度差异一般不超过核苷酸总长度 2%~5%, 尚不构成病原体不同基因型或血清型, 但存在基因序列差异即基因异质性的现象”^[46]。准种是一个病毒群体, 而非单一病毒。cccDNA 池和 rcDNA 池是两个主要的准种群。自然状态下, 准种通常由一种或几种主序列以及一系列变异序列组成, 当选择压力平衡时可维持相对稳定。在药物干预下, HBV 突变加速, 但准种的形成不会受到抑制。劣势病毒群也可能在环境或病程变化时, 漂变为优势病毒群。HBV 的准种特性被认为是引起乙型肝炎慢性化和抗病毒治疗失败的一个非常重要的病毒学因素。目前测序方法已有较高精度, 主流的二代测序技术准确度可达 98% 以上。然而, 准种现象对测序结果的影响仍然难以避免, 一些研究采用三次复检来排除 DNA 降解、污染等实验误差, 以确认共感染或准种池中的优势毒株。

1.5.3 已报道的 HBV 基因突变位点

preS/S 基因区的突变位点广泛分布于 preS1、preS2 及 S 区, 且集中于关键功能域。preS1 区 (108 aa) 已报道的突变位点包括 P21V、D27Y、A28T、N46D、E54Q、P70S、L74I、I84L、T87S、S101A、P110S 等, 其中多个位点与宫内传播相关, 其余部分位点功能尚未明确。preS2 区 (55 aa) 的突变位点有 T11N、R16K、R18K、F22L、N33D、V35A、P36L、A39V、P41H、I42T、S44P、I45T、F46S、T49A、P52L、N55D, 其中 F22L 与 HCC 进展相关, V35A、

P36L、S44P 等与宫内传播相关，其余位点功能多未阐明。S 区（227 aa）的突变位点分布于多个功能区，TMD1（4~24 aa）包含 T5A、G7R、L9*、G10D、L13I、A17R、F20L、L22V/* 等，主要影响 HBsAg 表达；CD4 epitope（21~65 aa）有 L42R、N48S、L49R、T57I 等，L49R 与 LC 进展相关，T57I 会降低 HBsAg 抗原性；B-cell epitope（100~160 aa）集中了大量免疫逃逸相关突变，如 Y100C、Q101H、S114T、G119D、T126I、T127P、A128V、G130R、T131N、M133T、Y134F、C137S、G145A 等，同时该区域多个位点与宫内传播相关；P/S 基因重合区域的 rtA181T/sW172、rtV191I/sW182、rtM204I/sW196 等突变具有致癌潜力；CD8 epitope（206~215 aa）的 S207R、L213I 与免疫逃逸相关；此外，S 区还存在多个影响 MHR 结构的突变位点。

C 基因区的突变位点主要分布于基础核心启动子（BCP）、preC 基因及 C 基因，且与病毒复制、免疫调节及疾病进展相关。BCP 区（nt 1575-1849）的突变位点包括 A1752G、T1753V/C/A/G、G1757A、A1762T、G1764A、G1764T、C1766G、C1766T、T1768A、C1773T、C1788G、C1799G 等，其中 A1762T/G1764A 双突变与 HBV 相关肝病进展、宫内传播风险相关，1762/1764/1766 三重突变与 pgRNA 转录增加相关；preC 基因（nt 1814~1901，aa -29~1）的突变位点有 A1814C/T（W → L）、G1862T、A1874T（L21*）、G1896A（W28*）、G1899A（G29D）等，涉及 HBeAg 表达影响、免疫逃逸及肝病进展，部分位点与母婴传播相关；C 基因（aa 1~183）的突变位点分布于 CTL 表位、T-helper 表位、B 细胞表位等功能区，如 S21T、D29H、E40D、E40Q、E64D、T67N、V74G、V74S、P79Q、A80T、A80S、V85I、M93V、I97L、I116V、P130T、P130Q、R151Q、G153C、S155T、P156S、R166P 等，部分位点与疾病严重程度、HCC 风险及父婴传播相关。

X 基因区的突变位点覆盖 N 端区域、B 细胞表位、锌指基序、Enhancer II、H-Box、BH3-like 基序、CTL 表位、核心启动子等区域。N 端区域有 C26S、C26R 等突变；B 细胞表位（aa 29-48）包含 P33S、L34I、T36D、T36G、S38P、S39P、P40S、P40A、S41P、L42P、S43P、P43S、P46S、A47T 等，部分位点与免疫逃逸及 HCC 进展相关；锌指基序（aa 58-84）有 D80E、T82S 等；Enhancer II（nt 1627-1774）包含 H86R、Q87R 等，Q87R 与 HBV 相关疾病进展相关；H-Box（aa 88-100）的 I88F、I88C 等位点涉及 HCC 进展及 DNA 损伤结合复合物 1 相

互作用；BH3-like 基序（aa 110-135）及 CTL 表位（aa 115-123）、锌指基序（aa 120-141）存在 D119N、L123W 等突变；核心启动子（nt 1742-1849）的 I127T、I127D/G、K130M、V131I、V131L、F132Y 等突变与慢性 HBV 感染、HCC 进展、病毒复制及细胞侵袭相关，部分位点形成双突变或三突变组合，与肝病进展阶段相关；此外，还有 H139D、C1831T 等位点，功能多未明确。

P 基因的突变位点主要集中于 RT 区，且多与抗病毒药物耐药相关。已报道的 RT 区突变位点包括 E11D、H54Y、N76D、L91I、R110G、T128N、D134E、L145M、Q149K、R153W、L180M、rtL180M+rtM204V、rtL180M+rtM204I、S202G、rtM204V、rtM204I、Q215S、F221Y、P237T、rtN236T、C256S、W257Y、L260F、D263E、I266R、Q267H/R/Y、K270R、E271D、M309K、N337T 等，其中 N76D 与泛昔洛韦临床失败相关，L91I、rtM204V/I 等与拉米夫定耐药相关，D134E、R153W 与替诺福韦相关耐药相关，S202G 与拉米夫定、恩替卡韦耐药相关，Q215S 与拉米夫定、阿德福韦酯耐药相关，F221Y、P237T 与阿德福韦酯耐药相关，部分突变会导致重叠的 S 区基因改变。

总体来看，HBV 各基因区突变位点的研究已积累了丰富的数据，突变位点的分布与基因功能区高度相关，部分位点的功能及临床关联已得到初步阐明，但仍有大量位点的具体作用尚未明确，且不同位点的协同效应、与传播途径及疾病结局的精准关联等仍需进一步探索。

本章小结

本章系统梳理了乙型肝炎病毒（HBV）的研究概况，为全书后续实验技术与数据分析内容构建了坚实的理论基础。在流行病学层面，HBV 作为全球重大公共卫生威胁，其传播以母婴传播为主要途径，疫苗接种虽显著降低了流行率，但仍存在免疫失败风险，而基因型、基因亚型的地理与民族分布差异，以及基因重组、共感染等现象，进一步增加了疾病防控的复杂性。生物学特征方面，HBV 以具有感染性的 Dane 颗粒为核心存在形式，其独特的生命周期依赖逆转录酶介导的复制过程，共价闭合环状 DNA（cccDNA）的持续存在是慢性感染的关键诱因，而血清学转换则是感染自然史不同阶段转换的重要标志。基因组结构上，HBV 约 3.2 kb 的双链环状 DNA 包含 S、C、P、X 四个高度重叠的开放阅读框，

这种紧凑结构使其能高效编码多种功能蛋白——HBs 蛋白参与病毒入侵与免疫识别，HBc/HBe 蛋白调控病毒复制与宿主免疫，HBp 蛋白主导基因组复制且与耐药性密切相关，HBx 蛋白则通过调控转录影响病毒复制与肝细胞癌变风险，再结合多个关键调控序列，共同构成了 HBV 复杂的分子调控网络。基因变异是 HBV 的核心生物学特征，高突变率与准种特性使其在宿主选择压力下不断演化，preS/S、C、X、P 四大基因区的突变位点多集中于功能域，涉及免疫逃逸、传播能力、疾病进展及耐药性等多个方面，虽已有大量位点被鉴定，但多数位点的具体功能、协同效应及临床关联仍需深入探索。本章通过对 HBV 流行病学、生物学、基因组及基因变异的全面概述，为后续聚焦基因变异的实验设计与数据分析提供了清晰的理论框架与研究背景。

第2章 基因流行病学概述

基因流行病学作为基因组学与流行病学深度融合的交叉学科，其核心是通过整合病原体基因组数据与人群流行病学信息，揭示传染病的传播规律、进化特征及防控靶点。根据 Nabukalu Kato M. (2024) 的定义，该学科依托二代、三代核酸测序技术的突破，以“每秒超 10,000 个碱基”的速度获取微生物基因组数据，实现对感染菌株传播路径的高精度追溯；而 Daniel Evans (2024) 进一步明确其实践属性，即是贯穿“病原体监测 - 序列分析 - 数据关联 - 防控干预”全流程的公共卫生工具，而非单一技术手段。从学科边界来看，基因流行病学的研究范畴涵盖三大维度。在病原体层面，解析病毒、细菌、真菌等病原体的基因组结构变异（如 SNP、插入缺失、耐药基因获得），明确亚型分化与毒力、耐药性的关联。在人群层面，结合病例的时空分布、接触史等流行病学数据，构建传播链模型，区分“散发病例”与“聚集性感染”。在公共卫生层面，将基因组分析结果转化为防控策略，如高风险变异株预警、疫苗菌株筛选、感染控制措施优化。

2.1 基因流行病学发展史

基因流行病学的发展并非孤立存在，而是与遗传学、流行病学及相关技术的进步并行推进，形成了鲜明的历史演进特征。

19 世纪是遗传学与流行病学的思想奠基期。查尔斯·达尔文于 1837 年提出进化树理论，并在 1859 年出版《物种起源》，系统阐述了生物进化的核心观点，为后续遗传规律的探索奠定了思想基础。在这一时期流行病学也迎来里程碑式的突破。1854 年约翰·斯诺通过绘制伦敦霍乱流行地图，精准定位了疫情源头，开创了流行病学现场调查与病因推断的先河。这两个领域的早期探索，分别从生物遗传本质和疾病人群分布两个维度，为基因流行病学的诞生埋下伏笔。

同期 20 世纪初，遗传学与流行病学均进入理论体系构建阶段。1905 年，庞尼特方格（Punnett square）的提出为孟德尔遗传规律的可视化提供了工具，而孟

德尔实验的重新发现则直接奠定了现代遗传学的基础；1922 年，费希尔精确检验（Fisher's exact test）的建立，为流行病学研究中的数据统计分析提供了重要方法，推动了流行病学从描述性研究向分析性研究的转变。

20 世纪中期，两大领域迎来关键性突破。1953 年，沃森（Watson）和克里克（Crick）揭示了 DNA 双螺旋结构，标志着分子生物学时代的来临，为从分子水平解析遗传因素的作用机制提供了可能；同期，多尔（Doll）和希尔（Hill）开展的吸烟与肺癌关联研究，确立了危险因素流行病学的研究范式，证实了环境因素在疾病发生中的重要作用。这一时期，分子生物学与危险因素流行病学的并行发展，为基因流行病学的交叉融合提供了坚实的理论与方法学支撑。

20 世纪后期，技术革新成为推动基因流行病学发展的核心动力。1983 年，穆利斯（Mullis）和史密斯（Smith）发明聚合酶链反应（PCR）技术，实现了核酸片段的指数级扩增，使得微量遗传物质的检测成为可能，为高通量基因分型技术的发展奠定了基础。1993 年 TOP500 超级计算机排行榜的设立，反映了高性能计算技术的飞速发展，其指数级增长的运算速度（以 GFLOPS 为衡量标准），为处理海量基因数据和流行病学队列数据提供了算力支持。

进入 21 世纪，基因流行病学进入规模化发展阶段。“量化自我”（quantified self）理念的兴起，推动了可穿戴设备在健康数据采集领域的应用，实现了遗传信息与个体表型、环境暴露信息的整合；23andMe 等个人基因组公司的出现，使得基因检测逐渐走向大众化，积累了海量人群基因型数据。同时，人类基因组计划（HGP）的完成的后续影响持续释放，为全面解析人类遗传密码、挖掘疾病相关基因提供了完整的参考图谱。

2.2 基因流行病学的应用类型

根据 Daniel Evans(2024) 的分类框架,基因流行病学按研究目标可划分为“描述性、探测性、调查性、分析性、预测性”五大类，各类均具备明确的核心目标、技术路径与应用价值：

1) 描述性基因流行病学：刻画病原体的“种群特征”。核心目标聚焦明确特定人群中存在的病原体亚型及变异株类型，为后续常态化监测构建基础。技术路径以系统性采样为核心，结合下一代测序（NGS）技术与基因型分型方法，绘

制病原体亚型在时间与空间维度的分布图谱。核心价值在于建立病原体的“基线种群结构”，为新变异株的出现提供参照标准，支持早期预警与监测体系优化。

2) 探测性基因流行病学：发现“隐性传播”与“未知暴发”。核心目标是明确目标变异株的存在状态，识别传统流行病学方法难以发现的隐性传播链与潜在暴发风险，弥补传统监测的盲区。技术路径采用“主动监测 + 分层采样”策略，针对高风险人群或场景开展病原体测序，通过序列同源性比对分析判断是否存在隐匿传播关联。核心价值在于精准识别无症状携带或隐性传播引发的感染风险，避免防控疏漏，提升监测的全面性。

3) 调查性基因流行病学：解析暴发的“传播路径”与“成因”。核心目标是判定事件是否构成暴发，并明确暴发的传播源头、路径及成因，为针对性防控措施制定提供科学依据。技术路径整合病例对照研究设计与基因组比对技术，通过采集暴发相关病原体样本进行全基因组测序，构建进化树并与非暴发菌株进行对比以明确克隆聚类关系，再结合病例时空分布信息追溯传播链条。核心价值在于从分子层面验证流行病学假设，精确定位暴发源头与传播媒介，支撑防控措施的靶向性实施。

4) 分析性基因流行病学：量化“基因型与结局”的关联。核心目标是明确特定病原体亚型或变异株与疾病相关结局（如重症、耐药、预后差异等）的关联性，为风险分层与精准干预提供依据。技术路径采用队列研究或病例 - 队列研究设计，先界定暴露因素（即特定病原体基因型）与结局指标，收集足量样本后比较不同基因型组的结局发生率，再通过统计学模型调整混杂因素，量化基因型与结局间的关联强度。核心价值在于识别高风险基因型，为差异化防控策略制定与重点人群干预提供数据支撑。

5) 预测性基因流行病学：预判“流行趋势”与“防控需求”。核心目标是预判未来病原体亚型或变异株的流行趋势，优化防控资源配置与技术手段（如疫苗菌株选择、干预时机确定等），实现从被动应对到主动防控的转变。技术路径整合基因组进化数据与流行病学模型，常用方法包括短期流行比例预测的 Nowcast 模型、长期趋势预判的 phylodynamic 模型，以及通过抗原基因序列同源性比对评估保护效果的疫苗匹配分析。核心价值在于提升防控工作的前瞻性与科学性，为防控策略动态调整提供精准指导。

2.3 人群水平数据在个体疾病风险评估中的价值

医学的核心是关注个体疾病的发生原因与诊疗方案，而流行病学则聚焦于人群中健康与疾病的决定因素及分布规律。在评估常见疾病的个体风险时，单纯依赖个体水平的精准测量难以获得可靠结果，必须依托人群水平的数据支持，其根本原因在于以下两点：

1) 常见疾病病因的复杂性与动态性：常见慢性疾病（如2型糖尿病、冠心病、乳腺癌等）的发生并非由单一因素导致，而是遗传因素、环境因素（如饮食、吸烟、运动）及两者交互作用的结果。这些因素构成了复杂的调控网络，涉及多条信号通路、多个基因位点的协同作用，且环境因素的动态变化（如生活方式的改变、暴露剂量的波动）进一步增加了病因的复杂性。单一基因的作用效应往往被其他因素掩盖，仅通过个体数据无法全面捕捉这种复杂关联。

2) 致病过程的随机变异性：疾病的发生不仅受确定性因素的影响，还存在随机变异的作用。遗传物质的随机突变、基因表达的随机波动、环境暴露的随机差异等，都会导致即使在遗传背景和生活环境相似的个体中，疾病发生风险也存在差异。这种随机变异无法通过个体测量来控制，只能通过大样本人群数据的统计分析，将随机误差纳入考量，从而更准确地估计遗传因素与疾病的关联强度。

流行病学通过对人群数据的系统分析，能够识别疾病的“危险因素”（包括遗传变异、环境暴露等）并量化其效应大小。在基因流行病学研究中，人群数据的分析维度涵盖关联分析、比例计算、发病率/患病率统计、聚集性分析、人口统计学特征、时空分布、概率估计等多个方面。通过整合人群中的遗传信息、家族风险、多基因风险评分等遗传指标，结合环境危险因素数据，能够有效分离遗传因素与环境因素的独立作用及交互效应，为个体风险评估提供可靠的科学依据。

2.4 多基因风险评分的偏倚与影响因素

多基因风险评分（PRS）是基因流行病学在疾病风险预测中的重要应用，但在实际应用中可能存在偏倚，影响其预测准确性和人群适用性，主要涉及以下几个核心因素：

1) 遗传与环境的相关性偏倚。遗传因素与环境因素之间并非相互独立，而是存在复杂的相关性，这种相关性会导致 PRS 的预测偏倚。一方面，父母的遗传易感性会影响其为子女创造的生活环境，即“基因-环境相关性”：例如，具有肥胖遗传易感性的父母，可能更倾向于选择高热量饮食，其子女既继承了肥胖相关基因，又暴露于高热量饮食环境，使得 PRS 可能高估遗传因素的独立效应。另一方面，遗传变异可能成为局部环境风险因素的“标记物”：例如，某一遗传变异在特定地区人群中频率较高，而该地区同时存在空气污染等环境危险因素，此时 PRS 与疾病的关联可能混杂了环境因素的效应，导致虚假关联。

2) 基础数据与目标人群的差异偏倚。PRS 的构建通常基于特定人群（如欧洲裔人群）的 GWAS 数据，当将其应用于其他人群（如非洲裔、亚裔人群）时，往往会出现预测效能下降的问题，即“人群分层偏倚”。这一现象的根本原因在于不同人群的遗传存在差异，导致疾病相关遗传变异的频率和效应大小在不同人群中不一致。例如，基于欧洲裔人群构建的 II 型糖尿病 PRS，在非洲裔、亚裔人群中的预测准确性显著降低；而基于多族裔人群数据构建的 PRS，其跨人群适用性则明显提升。这种偏倚的本质是基础数据与目标人群在遗传背景上的不匹配，导致 PRS 无法有效捕捉目标人群中的疾病遗传风险。

3) 偏倚的影响与应对思路。PRS 的偏倚会直接影响疾病风险推断的准确性，可能导致高风险人群误判、干预措施无效等问题。为减少偏倚，研究人员需要采取针对性策略：一是扩大 GWAS 的人群多样性，纳入不同遗传的人群数据，构建多族裔 PRS；二是在 PRS 分析中调整遗传 ancestry 变量，控制人群分层的影响；三是整合遗传数据与环境数据，明确基因-环境交互作用，分离遗传因素的独立效应；四是采用更先进的统计模型，优化 PRS 的构建方法，提高其在不同人群中的预测稳定性。

2.5 基因流行病学核心研究方法体系

流行病学：研究特定人群中健康相关结果的分布和决定因素，以及将此研究应用于控制健康问题。基因组流行病学：利用病原体基因组数据来确定特定人群中传染病的分布和传播情况，以及将此信息应用于控制健康问题。基因流行病学的研究体系围绕它们的定义展开，并且随着技术进步呈现出两个发展趋势，即生

物信息学工具从“数据处理”向“结果解读”延伸；数据应用从“本地存储”向“共享协作”发展。

2.5.1 遗传变异检测和分型技术

遗传变异检测是基因流行病学研究的基础，其技术发展直接推动了研究的规模化和精准化，核心技术包括：

1) 高通量基因分型技术：基于芯片的基因分型技术是 GWAS 的核心技术支撑，能够同时对数十万个甚至数百万个 SNP 位点进行检测，具有高通量、低成本、高准确性的特点。该技术通过设计覆盖全基因组的 SNP 探针，与样本 DNA 杂交后进行信号检测，快速获取个体的基因型信息。例如，Illumina 公司的 HumanOmniExpress 芯片可检测约 70 万个 SNP 位点，广泛应用于大型人群队列的遗传分型。

2) 下一代测序（NGS）技术：包括全基因组测序（WGS）、全外显子组测序（WES）和目标区域测序，能够全面捕捉基因组中的各类变异（SNP、插入缺失、结构变异等）。WGS 可覆盖整个基因组，适用于发现新的疾病相关变异；WES 聚焦于编码蛋白质的外显子区域，性价比高，常用于孟德尔遗传病和复杂疾病的候选基因筛选；目标区域测序则针对特定基因或基因组区域进行深度测序，适用于验证已知变异的关联效应。NGS 技术的优势在于能够检测稀有变异，弥补了芯片技术在变异覆盖度上的不足，为解析罕见变异与疾病的关联提供了可能。

3) 聚合酶链反应（PCR）相关技术：作为经典的分子生物学技术，PCR 及其衍生技术（如实时荧光定量 PCR、巢式 PCR、数字 PCR）在基因流行病学中仍发挥重要作用。实时荧光定量 PCR 可快速检测特定变异的等位基因频率，适用于小规模样本的验证实验或临床检测；数字 PCR 则具有更高的灵敏度和特异性，能够准确量化低频率变异，适用于微量样本（如肿瘤组织、体液样本）的遗传分析。

2.5.2 人群研究设计

基因流行病学的人群研究设计基于传统流行病学的研究框架，结合遗传数据的特点进行优化，主要包括以下类型：

1) 队列研究：队列研究是基因流行病学中最常用的研究设计之一，通过招募无病人群，收集基线时的遗传数据、环境暴露数据和人口学信息，然后进行长期

随访，观察疾病的发生情况，进而分析遗传变异、环境因素及其交互作用与疾病发病风险的关联。队列研究的优势在于能够明确暴露（遗传变异、环境因素）与结局（疾病发生）的时间顺序，减少反向因果偏倚，且可同时研究多种疾病结局。大型人群队列（如英国生物银行、“我们所有人”计划）是开展队列研究的重要平台，能够为基因流行病学研究提供充足的样本量和丰富的表型数据。

2) 病例对照研究：病例对照研究通过选择已确诊的病例和未患病的对照，回顾性收集两者的遗传变异信息和既往环境暴露史，比较病例组与对照组之间遗传变异的频率差异，分析遗传因素与疾病的关联。该方法具有样本量需求较小、研究周期短、成本低的特点，适用于罕见病的研究或初步探索疾病的遗传危险因素。但病例对照研究存在回忆偏倚和选择偏倚的风险，在数据分析时需要通过匹配、分层等方法进行控制。

3) 家系研究：家系研究以血缘关系为基础，通过分析患病家系与正常家系的遗传差异，定位疾病相关基因。适用于孟德尔遗传病的基因定位，通过连锁分析方法，能够快速锁定致病基因的染色体区域。对于复杂疾病，家系研究可通过分析家族聚集性，分离遗传因素的独立效应，例如通过比较亲属与非亲属之间的疾病发病率差异，估计遗传度。家系研究的优势在于能够有效控制环境混杂因素，因为家庭成员通常共享相似的生活环境，但其局限性在于样本招募难度大，尤其是对于罕见病家系。

4) 病例-队列研究与巢式病例对照研究：这两种设计是队列研究的高效化变体，兼顾了队列研究的科学性和病例对照研究的经济性。病例-队列研究从队列中随机抽取一个子队列作为对照，与队列中发生的所有病例进行比较，适用于多种疾病结局的研究；巢式病例对照研究则在队列中发生病例后，按年龄、性别等因素匹配相应的对照，仅对病例和对照的样本进行遗传检测，大大降低了研究成本。这两种设计尤其适用于生物样本库资源的二次利用，在大型队列研究中应用广泛。

2.5.3 数据分析与统计方法

基因流行病学的数据分析需要整合遗传数据、表型数据和环境数据，采用专门的统计方法处理遗传变异的特殊性（如多基因效应、基因-环境交互作用），核心方法包括：

1) 关联分析方法：

a) 单变量关联分析：针对单个遗传变异（如 SNP），采用卡方检验、Fisher 精确检验（适用于小样本）比较病例组与对照组的等位基因频率或基因型频率差异；对于定量表型（如血压、血糖），采用 t 检验或方差分析比较不同基因型组之间的表型均值差异。

b) 多变量关联分析：考虑到多个遗传变异的联合效应或混杂因素的影响，采用逻辑回归（适用于二分类结局）、Cox 比例风险回归（适用于生存结局）、线性回归（适用于定量结局）等多变量模型，调整年龄、性别、遗传 ancestry 等混杂因素后，估计遗传变异与疾病的关联效应（如比值比 OR、风险比 HR、回归系数 β ）。

c) 全基因组关联分析（GWAS）的统计方法：GWAS 需要对海量 SNP 进行关联检验，为控制多重检验导致的假阳性，通常采用 Bonferroni 校正或错误发现率（FDR）控制方法。此外，GWAS 数据分析还需要考虑人群分层的影响，通过主成分分析（PCA）将遗传 ancestry 作为协变量纳入回归模型，减少人群混杂导致的虚假关联。

2) 多基因风险评分（PRS）构建与验证方法：

a) PRS 构建：基于 GWAS 的汇总统计数据，采用加权求和的方法，将多个疾病相关 SNP 的效应值（如回归系数）作为权重，计算个体的 PRS，公式为： $PRS = \sum (\beta_i \times G_i)$ ，其中 β_i 为第 i 个 SNP 的效应值， G_i 为个体在第 i 个 SNP 上的风险等位基因数量。常用的 PRS 构建方法包括多基因评分分析（Polygenic Score Analysis）、LDpred 等，LDpred 方法能够考虑 SNP 之间的连锁不平衡（LD），提高 PRS 的预测准确性。

b) PRS 验证：PRS 构建后需要在独立的验证队列中进行验证，评估其预测效能，常用指标包括受试者工作特征曲线下面积（AUC）、预测值与实际表型的相关系数（ R^2 ）等。此外，还需要分析 PRS 在不同人群中的适用性，评估其跨人群的预测稳定性。

3) 基因 - 环境交互作用分析方法：基因 - 环境交互作用是指遗传变异与环境因素共同作用影响疾病风险，其效应大于两者独立效应的叠加。常用的分析方法包括相乘交互作用模型和相加交互作用模型。相乘交互作用通过在回归模型中纳入遗传变异与环境因素的乘积项进行检验；相加交互作用则采用相对超额风险（RERI）、归因交互作用比例（AP）、协同指数（S）等指标进行评估。基因 -

环境交互作用分析能够揭示遗传因素和环境因素的联合效应，为精准预防和干预提供依据，例如识别出对特定环境暴露敏感的遗传易感人群。

4) 孟德尔随机化(MR)分析方法: MR 分析的核心是选择合适的工具变量(遗传变异)，工具变量需满足三个核心假设：(1)与暴露因素强相关；(2)与混杂因素无关；(3)仅通过暴露因素影响结局，不直接影响结局。常用的 MR 分析方法包括逆方差加权法(IVW)、加权中位数法、MR-Egger 回归等。IVW 是最常用的方法，适用于工具变量与暴露因素存在强关联且无水平多效性的情况；加权中位数法对部分无效工具变量具有稳健性；MR-Egger 回归能够检测和校正水平多效性，减少偏倚。MR 分析的关键在于工具变量的选择，通常基于 GWAS 数据选择与暴露因素显著关联的遗传变异作为工具变量。

本章小结

基因流行病学作为连接精准健康与人群健康的桥梁,其研究方法体系不断完善,从遗传变异检测到人群关联分析,从因果效应验证到公共卫生应用,形成了一套多学科交叉的技术框架。高通量基因分型、下一代测序等技术的发展为研究提供了强大的技术支撑,队列研究、病例对照研究等人群设计方法确保了研究的科学性,孟德尔随机化、多基因风险评分等统计方法则提升了结果的可靠性和实用性。

在传染病防控领域,病原体基因组流行病学方法的应用实现了传染病的精准监测、溯源和预警,为疫情防控提供了重要技术手段。对于无明确流行病学关联(如无共同暴露史、无接触史)的预警,基因组监测通过分析病原体序列同源性,可明确是否为同一传播链导致的“真聚集”,避免过度防控或遗漏潜在爆发,提高了防控效率,补充传统流行病学调查的局限性。在慢性非传染性疾病研究中,基因流行病学揭示了疾病的遗传基础和基因-环境交互作用,为疾病的早期筛查、分层干预和精准治疗提供了科学依据。

未来,基因流行病学的发展将呈现以下趋势:一是研究人群的多元化,通过纳入不同遗传 ancestry、不同地域的人群,提高研究结果的普适性;二是技术创新,随着单细胞测序、空间转录组等新技术的出现,将能够更精准地解析遗传变异的功能效应和疾病的分子机制;三是多组学数据的整合,结合基因组学、转录组学、蛋白质组学、代谢组学等多组学数据,全面揭示疾病的致病机制;四

是与人工智能技术的深度融合，利用机器学习、深度学习等方法优化风险预测模型，提高疾病预测的准确性和干预的针对性。

本研究所关注的核心问题，“个体差异的病毒遗传驱动因素”“病毒遗传变异与临床现象的相关性”“挖掘潜在生物标志物”等，均属于基因流行病学的研究范畴。基因流行病学通过整合多维度数据，运用人群研究设计、遗传关联分析、分子溯源等核心方法，不仅能揭示特定人群病毒变异现况和趋势，还能高风险人群筛查、防控策略优化、疫苗升级等提供科学依据，是乙肝病毒宫内传播研究从“现象描述”走向“机制解析”“精准防控”的重要学科支撑。

第3章 实验设计与数据准备

突变分析是解析 HBV 基因变异与病程关联、指导临床精准治疗的核心环节，而实验与数据准备的规范性直接决定了后续突变检测结果的准确性、可靠性与可重复性。本章聚焦突变分析前的关键准备环节，从研究选题、样本量估算、实验材料选型、样本处理规范、实验分组设计、数据整理溯源到质量控制要点，系统梳理实操流程、明确操作标准、列明核心注意事项，重点规避前期准备中易出现的样本污染、分组混乱、数据缺失等常见失误，为后续突变检测实验的顺利开展和结果的科学解读奠定坚实基础。在实验设计阶段，重点考虑生物体系（研究对象、样本特点、标本类型等），分组设计，方法技术（实验技术、统计分析方法），可行性（实践难度、成本）。

3.1 研究选题：以假说为导向

开展研究的第一步是制订一个假设，然后决定测量指标和方法。很多研究会选择以假说为导向的研究设计，或者说是任务导向型研究设计。与缺乏明确方向的宽泛研究相比，方向明确的研究在深度和开展难度方面更有优势。比如，假设某个特定突变会降低某种药物疗效、疫苗免疫会促进病毒遗传变异、某种检测方法在实践中具有适用性等。然而在现实世界中，随着研究开展，进程和结果可能会偏离预期，这就需要一个开放辩证的思想，有理有据地对研究目标进行改变或补充。

3.2 研究设计与对照设置

3.2.1 实验分组设计原则

HBV 突变分析的实验分组需围绕研究目的，遵循“随机、对照、重复”的基本原则，确保分组的科学性和可比性，尽量避免各种偏倚的产生，从而保证研究接近现实世界的真实情况。核心设计原则如下：

(1) 明确研究目的和变量：根据研究目的确定分组变量，常见的分组变量包括患者病程阶段（急性乙肝、慢性乙肝、肝硬化、肝癌）、治疗状态（未治疗、正在治疗、治疗有效、治疗无效）、病毒基因型（A~H 型）、变异类型（野生型、突变型）等。

(2) 同质选择：选择同质的样本。各组样本在性别、年龄、病程、基础疾病、城乡人口等非研究变量上需保持均衡，减少混杂因素导致结果结论出现的偏差。

(3) 样本量充足原则：样本量过小会导致实验结果的偶然性增加，无法反映真实情况；样本量过大则会增加实验成本和工作量。需根据研究类型（横断面研究、病例对照研究）和统计方法，通过样本量估算确定最小样本量，一般每组样本量 ≥ 30 例（临床样本）或 ≥ 3 次重复（实验样本）。

(4) 随机性原则：临床样本分组时需采用随机化方法（如随机数字表法），避免人为选择样本导致的选择偏倚；实验样本分组时，需随机分配样本至不同实验组，确保各组样本的基线条件一致。

3.2.2 常见的实验设计类型

按照控制水平和研究对象分组方法主要有两类实验设计：

(1) 真实验设计（true experimental design）：具备随机分组、主动干预、严格对照三大特征，内部效度最高，适用于细胞实验、动物模型实验等可主动操控突变变量的研究。主要类型有，前测 - 后测对照组设计（pretest-posttest control group design）、仅后测对照组设计（posttest-only control group design）、析因设计（factorial design）等。

(2) 准实验设计（quasi-experimental design）：除了核心区别的没有随机分配，在设计上与真实验设计几乎没有差别。一般是因研究对象限制无法随机分配状态，需通过匹配、统计控制等方式弥补偏倚。由于没有随机分配，样本可能存在一些影响结果的固有属性，影响内部效度。但由于更接近现实条件，因此实际使用中更为常见，适用于临床回顾性研究、自然发生的突变关联分析等场景。准实验设计的主要类型有，非等效组设计（nonequivalent groups design）、非等效因变量设计（nonequivalent dependent variables design）、代理前测设计（proxy pretest design）、回归间断点设计（regression discontinuity design）、回归点位移设计（regression point displacement design）、双前测设计（double pretest design）、

切换复制设计（switching replications design）等。

3.2.2.1 前测 – 后测对照组设计

两组实验设计是最简单的真实验设计，用符号描述如下表：

	研究对象	前测	干预	后测
实验组	R	O ₁	X	O ₂
对照组	R	O ₃		O ₄

R 代表随机抽取的研究对象，O 代表干预前、后的观测值，X 代表干预的施加。干预效果（E）使用实验组和对照组干预前后观测值的差值：

$$E=O_2-O_1-(O_4-O_3)$$

个体差异化的属性经随机处理后，对内部效度的影响得到有效控制。需要注意的是尽量避免观测行为本身将偏差引入实验体系。

3.2.2.2 非等效因变量设计

在此之前，先介绍析因设计。在析因设计中，每个自变量称为一个因子，一个因子的每个细分称为一个水平。在析因设计中，可以检查每个自变量对因变量的个体影响，称为主效应（main effects）。还可以检查它们的联合影响，称为相互作用效应（interaction effects）。析因设计的关键在于两个自变量是互相独立且可人为控制的，如教学形式和教学时长。假设有如下问题。已知 HBV-P 基因 rtM204 位点是耐药突变位点，rtM204V/I 突变和 L 型核苷类药物耐药高度相关。现在要评估 rtM204I 耐药突变株与野生型 HBV 毒株对人肝癌细胞系 HepG2.2.15 的侵袭差异，分析感染复数（MOI）对侵袭效率的影响，明确突变株的侵袭特性及与感染条件的交互效应。在这种情况下，两个自变量存在侵袭能力的潜在关联，并非完全独立。且毒株类型是固有属性，不是处理因素。因此，其计算方法和析因设计相通，但分析的核心逻辑不同。在本设计中，“毒株类型的主效应”本质是“两种固有属性不同的毒株，在相同感染条件下的侵袭能力差异”；“MOI 的主效应”是“同一毒株在不同感染剂量下的侵袭效率变化”。关联效应解读为“毒株类型对 MOI 的效应存在调节作用”——即 MOI 升高对侵袭效率的提升效果，因毒株类型不同而存在差异（例如“MOI 从 5 升高到 15 时，WT-HBV 的侵袭率提升 44%，而 MUT-HBV 仅提升 35%”）。

（1）单组实验对象：人肝癌细胞系 HepG2.2.15（可稳定表达 HBV 受体 NTCP，支持病毒侵袭与复制），控制在 20-30 代，确保细胞状态一致性。

(2) 两变量分组设计：以“毒株类型”（自变量 1，下含 2 个水平：野生型 WT-HBV/ 突变型 MUT-HBV）和“感染复数（MOI）”（自变量 2，下含 3 个水平：5、10、15）为两个因素，共 6 个实验组，每组设 6 个复孔（满足析因设计统计效能要求），另设空白对照组（未感染病毒，仅加培养基）

(3) 实验操作：

a) 毒株制备：野生型 HBV 毒株（WT-HBV）：基因型 C，无 rtM204I 突变，病毒滴度调整为 1×10^8 IU/mL；rtM204I 突变株（MUT-HBV）：通过基因编辑技术构建，仅含 P 区 rtM204I 单点突变，其余序列与 WT-HBV 一致，病毒滴度调整为 1×10^8 IU/mL。

b) 细胞铺板：将 HepG2.2.15 细胞以 5×10^4 个 / 孔接种至 24 孔板，加入含 10% 胎牛血清的 DMEM 培养基，37℃、5% CO₂ 培养 24h，至细胞融合度达 70%-80%；

c) 病毒感染：按分组设置的 MOI 值，向各实验组加入对应毒株，空白对照组加入等量无血清 DMEM 培养基；37℃ 孵育 4h 后，吸弃病毒液，用 PBS 洗涤 3 次，加入新鲜完全培养基；

d) 培养与检测：继续培养 48h 后，收集细胞及上清液，进行指标检测。

(4) 检测指标：

a) 病毒侵袭效率：流式细胞术检测细胞内 HBsAg 阳性率（%），反映成功侵袭的细胞比例；

b) 病毒复制水平：实时荧光定量 PCR（qPCR）检测细胞内 HBV-DNA 相对含量（以 WT-HBV MOI=10 组为参照，采用 $2^{-\Delta\Delta Ct}$ 法计算）；

c) 细胞活力：CCK-8 法检测细胞吸光度值（OD₄₅₀），排除病毒侵袭对细胞存活的影响。

注： $2^{-\Delta\Delta Ct}$ 法是实时荧光定量 PCR（qPCR）中用于分析基因相对表达量的标准化方法。Ct 值为循环阈值。首先计算 ΔCt 值，校正样本差异， $\Delta Ct = Ct_{\text{目的基因}} - Ct_{\text{内参基因}}$ 。 $\Delta\Delta Ct = \Delta Ct_{\text{处理组}} - \Delta Ct_{\text{对照组平均}}$ 。目的基因在处理组中的复制水平是对照组的 $2^{-\Delta\Delta Ct}$ 倍。

加载必要的包

library(tidyverse)

library(car) # 方差分析与多重比较


```

library(emmeans)  # 效应量计算与事后检验
library(ggplot2)
library(readxl)

# 1. 导入 Excel 数据
data_path <- "保存地址 / 模拟数据 2_ 混合设计 .xlsx"
hbv_invasion_data <- read_excel(data_path)
cat("数据前 5 行: \n")
print(head(hbv_invasion_data))
cat("\n 数据列名: ", colnames(hbv_invasion_data), "\n")
cat("group 列唯一值: ", unique(hbv_invasion_data$group), "\n")

# 2. 数据预处理: 适配编码规则, 筛选实验组, 转换变量类型
## 2.1 定义空白对照组 (group=9), 单独提取用于校准
blank_control <- hbv_invasion_data %>%
  filter(group == 9) # 空白对照组编码为 9

## 2.2 筛选实验组 (排除空白对照组 group=9)
analysis_data <- hbv_invasion_data %>%
  filter(group != 9) %>% # 仅保留实验组 (group=1-6)
  mutate(
    # 转换因子变量: Virus_type (毒株类型)、MOI (感染复数)
    Virus_type = factor(Virus_type, levels = c("WT-HBV", "MUT-HBV")), #
    保持毒株类型水平顺序
    MOI = factor(MOI, levels = c(5, 10, 15)), # MOI 为 5/10/15, 转为因子变量
    组合分组 = paste(Virus_type, MOI, sep = "_") # 构建组合分组 (用于后续可
    视化)
  ) %>%
  # 移除可能的 NA 行 (可以在原表根据实际情况选择保留或删除)
  drop_na(Virus_type, MOI, HBsAgPos, DNA, OD450)

```

3. 基线校准：以空白对照组 OD450 均值校准细胞活力

3.1 计算空白对照组 OD450 均值

```
blank_od_mean <- blank_control %>%  
  pull(OD450) %>%  
  mean(na.rm = TRUE) # 忽略 NA 值（空白组 OD450 应非 NA）
```

3.2 校准实验组细胞活

```
analysis_data <- analysis_data %>%  
  mutate(  
    OD450adj_calibrated = OD450 / blank_od_mean # 重新校准(覆盖或新增列,  
    根据需求选择)  
  )
```

4. 描述性统计分析（按毒株类型 × MOI 分组）

```
desc_stats <- analysis_data %>%  
  group_by(Virus_type, MOI) %>%  
  summarise(  
    # HBsAg 阳性率（HBsAgPos）描述性统计  
    HBsAgPos_均值 = mean(HBsAgPos),  
    HBsAgPos_标准差 = sd(HBsAgPos),  
    # HBV-DNA 相对含量（DNA）描述性统计  
    DNA_均值 = mean(DNA),  
    DNA_标准差 = sd(DNA),  
    # 校准后细胞活力  
    OD450adj_均值 = mean(OD450adj, na.rm = TRUE),  
    OD450adj_标准差 = sd(OD450adj, na.rm = TRUE),  
    样本量 = n(), # 验证每组复孔数  
    .groups = "drop"  
  ) %>%  
  # 格式化输出（保留 2 位小数）
```

```

mutate(
  HBsAgPos_均值_标准差 = paste0(sprintf("%.1f", HBsAgPos_均值), "±",
    sprintf("%.1f", HBsAgPos_标准差)),
  DNA_均值_标准差 = paste0(sprintf("%.2f", DNA_均值), "±",
    sprintf("%.2f", DNA_标准差)),
  OD450adj_均值_标准差 = paste0(sprintf("%.3f", OD450adj_均值), "±",
    sprintf("%.3f", OD450adj_标准差))
) %>%
select(Virus_type, MOI, HBsAgPos_均值_标准差, DNA_均值_标准差,
  OD450adj_均值_标准差, 样本量)

```

```

cat("\n 各实验组描述性统计结果: \n")
print(desc_stats)

```

5. 双因素方差分析 (Two-way ANOVA) : 检验主效应与交互效应

5.1 检验前提条件 (正态性 + 方差齐性)

5.1.1 正态性检验 (Shapiro-Wilk 检验, 按分组检验)

```

normality_test <- analysis_data %>%
  group_by(Virus_type, MOI) %>%
  summarise(
    HBsAgPos_正态性_p值 = shapiro.test(HBsAgPos)$p.value,
    DNA_正态性_p值 = shapiro.test(DNA)$p.value,
    .groups = "drop"
  )

```

```

cat("\n 正态性检验结果 (p>0.05 为符合正态分布): \n")
print(normality_test)

```

5.1.2 方差齐性检验 (Levene 检验)

```

levene_test_HBsAg <- leveneTest(HBsAgPos ~ Virus_type * MOI, data =

```

analysis_data)

```
levene_test_DNA <- leveneTest(DNA ~ Virus_type * MOI, data = analysis_data)
```

```
cat( "\nHBsAg 阳性率 ( HBsAgPos ) 方差齐性检验: \n" )
```

```
print(levene_test_HBsAg)
```

```
cat( "\nHBV-DNA 相对含量 ( DNA ) 方差齐性检验: \n" )
```

```
print(levene_test_DNA)
```

5. 一般不符合正态分布，选择非参数检验：Kruskal-Wallis 检验

5.2 检验 HBsAg 阳性率 (HBsAgPos)

5.2.1 创建组合因子变量

```
analysis_data <- analysis_data %>%
```

```
  mutate(Group = interaction(Virus_type, MOI))
```

5.2 检验 HBsAg 阳性率 (HBsAgPos)

```
kruskal_test_HBsAg <- kruskal.test(HBsAgPos ~ Group, data = analysis_data)
```

```
cat( "\n===== HBsAg 阳 性 率 ( HBsAgPos ) Kruskal-Wallis 检 验 结 果  
===== \n" )
```

```
print(kruskal_test_HBsAg)
```

```
===== HBsAg阳性率 ( HBsAgPos ) Kruskal-wallis检验结果 =====  
> print(kruskal_test_HBsAg)
```

```
      Kruskal-Wallis rank sum test
```

```
data:  HBsAgPos by Group  
Kruskal-wallis chi-squared = 27.984, df = 5, p-value = 3.666e-05
```

5.3 检验 HBV-DNA 相对含量 (DNA)

```
kruskal_test_DNA <- kruskal.test(DNA ~ Group, data = analysis_data)
```

```
cat( "\n===== HBV-DNA 相 对 含 量 ( DNA ) Kruskal-Wallis 检 验 结 果  
===== \n" )
```

```
print(kruskal_test_DNA)
```

```
> print(kruskal_test_DNA)
```

```
      Kruskal-Wallis rank sum test
```

```
data:  DNA by Group  
Kruskal-wallis chi-squared = 33.854, df = 5, p-value = 2.546e-06
```

6. 事后检验（多重比较）：Dunn 检验

library(FSA) # 需要加载 FSA 包进行 Dunn 检验

6.1 HBsAgPos 的事后检验

```
posthoc_HBsAg <- dunnTest(HBsAgPos ~ Group, data = analysis_data, method =
"bonferroni" )
cat( "\n===== HBsAg 阳性率显著差异的组间比较（Kruskal-Wallis 事后检验）
===== \n" )
print(posthoc_HBsAg$res)
```

```
===== HBsAg阳性率显著差异的组间比较（Kruskal-wallis事后检验） =====
> print(posthoc_HBsAg$res)
      Comparison      Z      P.unadj      P.adj
1  MUT-HBV.10 - MUT-HBV.15 -0.9454159 3.444466e-01 1.0000000000
2  MUT-HBV.10 - MUT-HBV.5  1.6990082 8.931764e-02 1.0000000000
3  MUT-HBV.15 - MUT-HBV.5  2.6444241 8.183006e-03 0.1227450949
4  MUT-HBV.10 - WT-HBV.10 -1.1372394 2.554382e-01 1.0000000000
5  MUT-HBV.15 - WT-HBV.10 -0.1918235 8.478805e-01 1.0000000000
6  MUT-HBV.5 - WT-HBV.10 -2.8362476 4.564705e-03 0.0684705681
7  MUT-HBV.10 - WT-HBV.15 -2.7677392 5.644661e-03 0.0846699130
8  MUT-HBV.15 - WT-HBV.15 -1.8223233 6.840594e-02 1.0000000000
9  MUT-HBV.5 - WT-HBV.15 -4.4667474 7.941781e-06 0.0001191267
10 WT-HBV.10 - WT-HBV.15 -1.6304998 1.029959e-01 1.0000000000
11 MUT-HBV.10 - WT-HBV.5  1.3427646 1.793482e-01 1.0000000000
12 MUT-HBV.15 - WT-HBV.5  2.2881804 2.212702e-02 0.3319052517
13 MUT-HBV.5 - WT-HBV.5 -0.3562437 7.216581e-01 1.0000000000
14 WT-HBV.10 - WT-HBV.5  2.4800039 1.313809e-02 0.1970714061
15 WT-HBV.15 - WT-HBV.5  4.1105037 3.947969e-05 0.0005921953
```

6.2 DNA 的事后检验

```
posthoc_DNA <- dunnTest(DNA ~ Group, data = analysis_data, method =
"bonferroni" )
cat( "\n===== HBV-DNA 相对含量显著差异的组间比较（Kruskal-Wallis 事
后检验）===== \n" )
print(posthoc_DNA$res)
```

```
===== HBV-DNA相对含量显著差异的组间比较（Kruskal-wallis事后检验） ==
> print(posthoc_DNA$res)
      Comparison      Z      P.unadj      P.adj
1  MUT-HBV.10 - MUT-HBV.15 -1.9456384 5.169819e-02 7.754728e-01
2  MUT-HBV.10 - MUT-HBV.5  1.9593401 5.007297e-02 7.510945e-01
3  MUT-HBV.15 - MUT-HBV.5  3.9049786 9.423374e-05 1.413506e-03
4  MUT-HBV.10 - WT-HBV.10 -1.0413276 2.977235e-01 1.000000e+00
5  MUT-HBV.15 - WT-HBV.10  0.9043108 3.658306e-01 1.000000e+00
6  MUT-HBV.5 - WT-HBV.10 -3.0006677 2.693883e-03 4.040825e-02
7  MUT-HBV.10 - WT-HBV.15 -2.9732644 2.946505e-03 4.419758e-02
8  MUT-HBV.15 - WT-HBV.15 -1.0276259 3.041258e-01 1.000000e+00
9  MUT-HBV.5 - WT-HBV.15 -4.9326045 8.114035e-07 1.217105e-05
10 WT-HBV.10 - WT-HBV.15 -1.9319368 5.336731e-02 8.005097e-01
11 MUT-HBV.10 - WT-HBV.5  0.9591175 3.374995e-01 1.000000e+00
12 MUT-HBV.15 - WT-HBV.5  2.9047560 3.675395e-03 5.513093e-02
13 MUT-HBV.5 - WT-HBV.5 -1.0002226 3.172028e-01 1.000000e+00
14 WT-HBV.10 - WT-HBV.5  2.0004452 4.545222e-02 6.817833e-01
15 WT-HBV.15 - WT-HBV.5  3.9323819 8.410830e-05 1.261625e-03
```

7. 结果可视化

```
library(ggplot2)
```

HBsAg 阳性率可视化

```
plot_HBsAg <- ggplot(analysis_data, aes(x = as.factor(MOI), y = HBsAgPos,
color = Virus_type, group = Virus_type)) +
  stat_summary(fun = mean, geom = "line", linewidth = 1.2) +
  stat_summary(fun = mean, geom = "point", size = 3.5) +
  stat_summary(fun.data = mean_se, geom = "errorbar", width = 0.2) +
  labs(
    x = "感染复数 ( MOI )",
    y = "HBsAg 阳性率 ( % )",
    color = "毒株类型",
    title = "HBV 不同毒株对 HepG2.2.15 细胞的侵袭效率对比",
    subtitle = "误差线表示标准误 ( SE )，每组 n=6"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 14, face = "bold" ),
    plot.subtitle = element_text(hjust = 0.5, size = 10),
    legend.position = "bottom"
  )
```

HBV-DNA 相对含量可视化

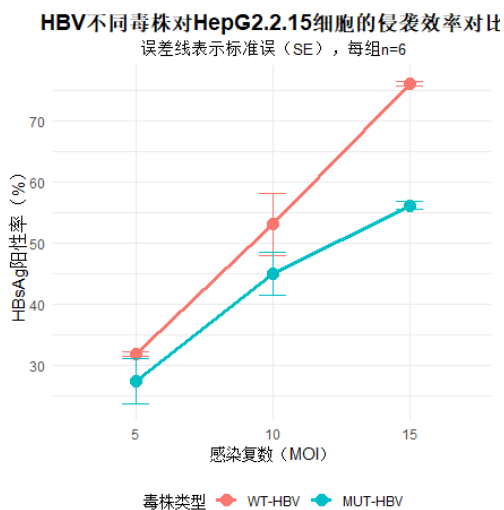
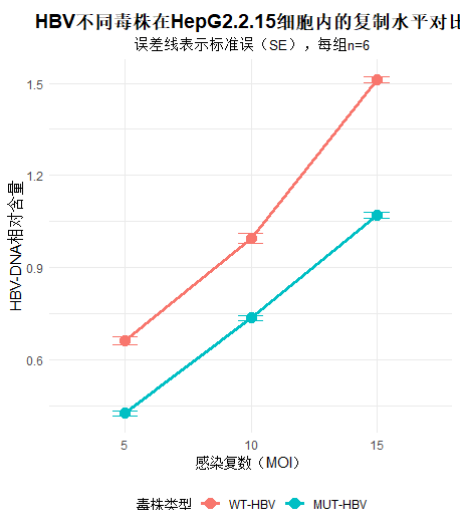
```
plot_DNA <- ggplot(analysis_data, aes(x = as.factor(MOI), y = DNA, color =
Virus_type, group = Virus_type)) +
  stat_summary(fun = mean, geom = "line", linewidth = 1.2) +
  stat_summary(fun = mean, geom = "point", size = 3.5) +
  stat_summary(fun.data = mean_se, geom = "errorbar", width = 0.2) +
  labs(
```



```

x = "感染复数 (MOI)",
y = "HBV-DNA 相对含量",
color = "毒株类型",
title = "HBV 不同毒株在 HepG2.2.15 细胞内的复制水平对比",
subtitle = "误差线表示标准误 (SE), 每组 n=6"
)+
theme_minimal() +
theme(
  plot.title = element_text(hjust = 0.5, size = 14, face = "bold"),
  plot.subtitle = element_text(hjust = 0.5, size = 10),
  legend.position = "bottom"
)
# 显示图表
print(plot_HBsAg)
print(plot_DNA)

```



3.2.3 病例对照研究

病例对照研究 (case control study) 和横断面研究 (cross-sectional study)、

队列研究（cohort study）同属于观察性研究，即研究人员不控制参与者的入组分配。三类研究对比如下表：

内容	横断面研究	病例 - 对照研究	队列研究
研究性质	描述性	分析性	分析性
研究目的	现况调查	产生关于暴露与结果之间潜在关联的初步假设	验证危险因素、预后因素
研究顺序	现况性	一般是回顾性	一般是前瞻性
报告指标	PR	一般用 OR（当结果罕见时，比值比通常近似于相对风险）	RR、AR
应用	分布研究	一病多因	一因多病
适应病种	常见病	一般是罕见病	常见病
证明因果关系	否	否	是
相对论证强度	弱	中	高

病例对照研究和回顾性队列研究的区别在于，病例对照研究根据是否发生相关结局分为病例组和对照组（对照组不一定是健康人群），然后对病例组和对照组的暴露情况进行评估；回顾性队列研究则是根据参与者的暴露程度进行分组，从而确定暴露与结局的关系。病例对照研究的特性使其在因逻辑或伦理原因不能对暴露进行随机处理的情况下，相比其他方法具备独特优势。

回顾性为根据已知结果推测起效变量，前瞻性为已知输入变量等待结果发生。病例对照研究一般是回顾性研究，因为大多数招募方式为根据临床记录审入，但当志愿者招募和数据采集在病例发展中进行时，病例对照研究是前瞻性研究。现实世界情况会更复杂，病例对照研究和队列研究不是非此即彼的关系。如，本研究根据孕检临床记录招募乙肝表面抗原（HBsAg）携带孕妇，根据外周血血清 HBV 开展基因研究，为病例对照研究。但是结局指标为脐血 HBsAg 阳性或其基因变异，在此时还不知道，这是前瞻性队列研究的特征。这种在队列研究中进行病例研究的方式，被称为巢式病例对照研究（nested case-control study）。定义队列是巢式病例对照研究的第一步，嵌套病例对照可以在其后重新设计。此后的步骤分别是，估算样本量、建立数据采集方法、收集和分析数据、撰写病例对照研究报告。

病例对照研究可能遇到的 7 种偏倚和应对策略：

偏倚	定义	应对策略
选择偏倚 (selection bias)	选择志愿者和 / 或研究因素的程序影响结果时发生。	为病例组和对照组选择排除标准，以最大化参与者的入组可能性（确保样本选择流程合理，减少选择偏差）。
确定偏倚 (ascertainment bias)	对“疾病或暴露”的认定不准确时发生。	采用随机抽样选择参与者，并通过盲法（让研究者 / 参与者不知分组情况）避免“选择性识别”（即避免因主观认知而错误判断暴露或疾病状态）。
伯克森偏倚 (Berkson bias , 或入院率偏倚 admission rate bias)	基于“病情重或病种多的患者比较症患者更可能住院”的概念。在医院人群中可能高估“暴露的病例数”，不能代表社会人群的真实关联。	定义对照组时，需评估“对照个体是否因严重程度相似的疾病入院”（确保病例和对照的“入院概率”可比，减少住院偏倚）。
奈曼偏倚 (Neyman bias)	当病例组不能代表目标人群时发生。例如，“短暂性疾病”或“导致早死的疾病”，可能在病例组中代表性不足（因为这类患者可能未被纳入研究，或已死亡）。	优先统计新发病例（ incident cases ）；若不可行，则讨论“使用现患病例（ prevalent cases ）”对研究结局的影响（明确说明数据局限性）。
回忆偏倚 (recall bias , 或记忆偏倚 memory bias)	信息收集依赖于参与者的记忆。通常，“病例（患病者）”对“目标状态”的回忆比“对照（未患病者）”更清晰，这种差异会影响数据准确性。	使用疾病发生前就已系统收集的数据（如病历、前瞻性队列的基线数据），避免依赖“事后回忆”带来的偏差。
观察者偏倚 (observer bias)	观察者（研究者）在收集数据时，因“个体是病例还是对照”而改变信息收集的细节程度（即主观上对病例组 / 对照组的观察标准不一致）。	选择对照时，确保对两组（病例 + 对照）的观察在相同条件下进行；培训观察者并标准化流程和行为（如统一数据记录方式、盲法观察），减少主观偏差。
混杂偏倚 (Confounding Bias)	存在一个混杂变量（与“暴露”和“结局”都相关，但不直接在“暴露→结局”的因果路径中）时发生。该变量会干扰真实的暴露 - 结局关联。	确保研究组间的可比性；采用配对（ Pair ）或分层（ Stratify ）分析，控制混杂因素的影响（如按年龄、性别分层，或配对病例 - 对照）。

3.3 样本量估算

对于基因流行病学而言，样本量估算是研究工作的核心环节，需要在报告中说明样本量的估算方法和评估依据。和其他观察性研究一样，需要使用适当的样

本量来确保研究结果的可靠性。一般希望样本量尽可能大，但在实践中很难实现，尤其是一些已经得到一定控制的流行病，或其他的少见病例、病种，或资源、时间等不支持开展大规模调查。在这种情况下，需要评估样本量的适宜下限，以及实际样本量低于阈值可能造成的不良影响。样本量过少会可能导致遗漏低频率变异，或者无法满足预设分组的统计量、齐次性要求，削弱统计学效力。在一些采用抽样调查的研究中，过小的样本量可能导致抽样误差过大，使得统计结果不能反映总体的真实水平。美国疾病控制中心开发了一个多功能程序 Epi info（<https://www.cdc.gov/epiinfo/support/downloads.html>），用于辅助公共卫生专业人员进行疫情调查、监测数据库管理，支持包含样本量估算在内的统计分析。



一般而言，使用二项分布近似公式作为样本量估算公式：

$$n = \frac{z^2 \times p \times (1 - p)}{d^2} \times deff$$

当 $np \geq 5$ 且 $n(1-p) \geq 5$ 时，二项分布可近似正态分布，公式适用。对于罕见变异（ $p < 5\%$ ），只要 n 足够大，该近似仍成立。公式中 n 为要估算的样本量，其余符号属于样本量估算需明确的四个核心参数：

（1）预期流行率（ p ）：指总体中目标变异（如 H1N1 病毒 H275Y 突变）

的预估发生比例。预期流行率越低，所需样本量越大。在风险评估中， p 值可以是特定分组暴露于风险因素的概率，暴露率越低，所需样本量越大。在耐药研究中， p 值可以是分组预期突变率，并且预计药物干预组 p 值高于对照组的 p 值。 p 值根据既往数据、预试验、系统综述或专家共识确定，结合本地人群特征酌情调整。实在无法确定 p 值，可以调整 p 值的设定，在可行性和覆盖度之间寻找平衡。

(2) 允许误差 (d)：指样本估算的流行率与总体真实流行率之间的最大可接受偏差，反映研究结果的精度。允许误差越小，精度越高，所需样本量越大。 d 的值根据实际需要设定，初步探索性研究可以适当放宽，若需要提高精确度就适当降低，通常 $\leq p/2$ 。

(3) 置信水平 ($1-\alpha$)：指样本估算结果包含总体真实值的概率，反映研究结果的可靠性。置信水平越高，所需样本量越大。一般默认置信水平为 95% (即 $\alpha=0.05$)，对应标准正态分布的 z 值为 1.96。和允许误差相似，高风险决策可以提高置信水平到 99% ($z=2.58$)，探索性研究可降低到 90% ($z=1.645$)。

(4) 研究方法校正系数 ($deff$)：对于抽样研究，简单随机抽样不需要校正。若是分层抽样， $deff$ 可设为 1.1~1.3，整群抽样 (如以医院为群抽样) 设为 1.5~2.0，需根据预试验或既往抽样数据确定，避免因抽样误差低估导致样本量不足。对于随访调查的掉落率，或来源更宽泛的“损耗率”， $deff$ 设为 $1/(1-\text{损耗率})$ 。如掉落率为 20% 时， $n_{\text{校正}}=n/(1-20\%)=1.25n$ 。

对于预期流行率很低的情况，如 $p<0.5\%$ ， $1-p\approx 1$ ，二项分布简化为泊松分布，样本量估算公式调整为：

$$n = \frac{z^2 \times p}{d^2} \times deff$$

若实验组样本量为 n_1 ，对照组样本量为 n_2 ，变异率分别为 p_1 和 p_2 ，两组等比例分配，增加检验效能参数，一般设为 80% ($z=1.28$) 或 90% ($z=1.645$)，样本量估算公式为

$$n = \frac{z_{1-\alpha/2}^2 \times [p_1(1-p_1) + p_2(1-p_2)] + z_{1-\beta}^2 \times [p_1(1-p_1) + p_2(1-p_2)]}{(p_1 - p_2)^2} \times deff$$

在突变率很低的情况下需要对病例进行富集，需要引入“分配比”，如采用试验组：对照组 = 2:1 或其他比例，对照组样本量 $n_2=k \cdot n_1$ ，试验组样本量估算公式为：

$$n_1 = \frac{z_{1-\alpha/2}^2 \times \left[p_1(1-p_1) + \frac{p_2(1-p_2)}{k} \right] + z_{1-\beta}^2 \times \left[p_1(1-p_1) + \frac{p_2(1-p_2)}{k} \right]}{(p_1 - p_2)^2} \times deff$$

3.4 实验材料选型

HBV 突变分析的样本类型以临床样本为主，核心需求是获取高质量的 HBV 核酸（DNA 或 RNA），因此样本选型需围绕“核酸含量充足、完整性好、污染少”的核心原则，结合研究目的和检测技术选择合适的样本类型。常见样本类型及选型要点如下：

3.4.1 血液样本

血液样本是 HBV 突变分析最常用的样本类型，主要包括血清和血浆，其中血清是首选。HBV 主要存在于血液中，病毒载量较高，可直接提取病毒核酸进行突变检测，适用于大多数 HBV 突变分析场景（如耐药突变检测、基因分型、准种分析等）。

选型注意事项：① 样本来源：需采集 HBV 感染者的静脉血，优先选择病毒载量 $\geq 10^3$ IU/mL 的样本——若病毒载量过低，可能导致核酸提取失败或检测结果假阴性；对于低病毒载量样本，需选择灵敏度更高的核酸提取和检测方法。② 样本类型选择：血清与血浆均可使用，但血清中红细胞破裂释放的核酸酶较少，核酸完整性更好，更推荐使用；若使用血浆，需选择 EDTA 或枸橼酸钠抗凝管（避免使用肝素抗凝管，肝素会抑制后续 PCR 反应，且难以彻底去除）。③ 特殊场景：若需研究 HBV 的转录组（如 pgRNA 检测），需采集全血并尽快分离外周血单个核细胞（PBMC），或使用专用 RNA 保存管，避免 RNA 降解。

3.4.2 其他样本

除血液样本外，唾液、尿液、乳汁等样本也可用于 HBV 突变分析。例如，唾液样本获取难度低，可用于 HBV 感染的筛查，但病毒载量较低，需高灵敏度检测方法；乳汁样本适用于哺乳期 HBV 母婴传播风险评估。此类样本选型需结合具体研究目的，且需验证样本中 HBV 核酸的可检测性。

3.5 样本采集、保存与预处理规范

样本采集、保存与预处理是保障研究质量的关键环节，操作不规范可能导致样本信息错误、缺失，核酸降解、污染或产量不足等，进而影响后续突变分析结果。调查需严格遵守《赫尔辛基宣言》，取得实施机构医学伦理委员会授权，取得志愿者知情同意，尊重其意愿并确保安全。

3.5.1 招募志愿者

在研究的样本采集（specimen collection）部分，主要说明志愿者招募和排除标准。包含研究背景和伦理批准；样本来源、样本数量；志愿者招募时期、来源地、入组数量、知情同意；标本数量；标本采集和保存方法。根据实验室结果、调查问卷和临床信息，和预先制定排除标准选择样本，以确保研究的有效性和结果的准确性^[7]。母亲年龄 ≥ 20 岁，妊娠28至42周，母亲血清在定期产前检查、分娩后检测为HBsAg阳性，肝肾功能正常，未感染新冠、流感、HCV、HDV或HIV等其他传染病，未长期使用糖皮质激素药物。由经过培训的调查员对入选母亲进行问卷调查及翻阅电子病历获取母儿基本情况。采集脐血和母亲外周静脉血，及时分装血清后于 -80°C 冰箱保存。

在结果说明的对应部分，报告总样本数，入组母亲例数、新生儿（脐血）例数，入组样本胎盘传播率；符合条件的对照例数，筛选方法后入组数量；基因型、血清亚型比例，及对应获得的序列数量、扩增成功率。

3.5.2 血液样本的采集、保存与预处理

血液样本是HBV突变分析的首选样本，其采集、保存与预处理流程相对简单，但需严格遵循规范，避免样本溶血、污染或核酸降解。

3.5.2.1 采集规范

1) 采集前准备：准备好采血针、采血管（黄管，含分离胶和促凝剂）、止血带/钳、棉签/纱布、碘酒、一次性手套、冻存管等耗材；核对患者信息（姓名、性别、年龄、病历号等），向患者说明采集目的，获得配合；采血人员需佩戴一次性手套，做好个人防护。

2) 外周血采集操作：① 选择患者手臂肘前区静脉，用碘酒棉签擦拭消毒，

待酒精挥发后进行采血；② 采血时避免过度用力挤压血管，防止组织液混入血液导致溶血；

3) 脐带血采集操作：① 胎儿娩出后、胎盘娩出前立即用无菌止血钳夹住脐带近胎儿端；② 使用碘酒纱布由脐带胎盘端向胎盘方向单向擦拭穿刺点；③ 从脐带中段（远离母体侧）进行穿刺，避免穿刺点靠近胎盘端，无菌操作抽取其中 5 ml 脐带静脉血。

4) 采集量：每管采集 3~5mL 血液，若需同时进行多项检测（如病毒载量检测、突变检测、基因分型），可采集 2~3 管。

5) 标记：不同于医疗检验条码，实验用途的标本标记优先考虑简洁清晰和盲法需要。核心信息为采集地代码（非多中心可略）、标本唯一索引号、分组 / 类型代码，并在纸质 / 电子表记录各标本或批次对应的收集时间。冻存管侧面也要标记，以免盖顶标记模糊。使用油性记号笔标记，避免因冰冻导致字迹脱落。

3.5.2.2 保存规范

1) 短期保存：采集后的血液样本若在 24 小时内进行血清 / 血浆分离，可置于 2℃~8℃ 冷藏保存；若超过 24 小时，需先分离血清 / 血浆后再冷藏或冷冻保存。

2) 长期保存：① 血清 / 血浆分离后，转移至无核酸酶的 1.5mL 离心管或冻存管中；② 置于 -20℃ 保存可保存 3 个月~6 个月，置于 -80℃ 可长期保存（1 年以上）；③ 避免反复冻融（最多冻融 3 次），反复冻融会导致核酸降解，若需多次使用，可将样本分装为 200 μ L~500 μ L / 管。

3) 运输规范：生物标本运输需符合相关安全要求，取得相应许可，使用专用标本运输箱，根据运输时长加入足量干冰或冰袋，确保到达时标本仍然冻结；液体标本需密封良好，充装不超过最大容积 80%，避免漏液污染；运输过程中需避免剧烈震荡和温度波动，运输时间尽量控制在 24 小时内。

3.5.2.3 预处理规范（血清 / 血浆分离）

血液样本需分离血清 / 血浆后才能进行核酸提取，预处理流程如下：

1. 血清分离（促凝管）：① 采血后将采血管直立放置，避免颠倒；② 小心将采血管放入离心机，设置离心参数：4℃、3500r/min、8 分钟；③ 离心后，用无核酸酶的吸头吸取上层血清（避免吸入血凝块），转移至标记好的 1.5mL 离心管中，若血清中存在杂质，可再次离心去除杂质。

2. 常见失误及规避：① 溶血问题：剧烈震荡、采血时过度挤压、离心速度

过高都可能导致溶血，溶血会释放红细胞内的核酸酶，降解 HBV 核酸。规避方法：采血时动作轻柔，避免剧烈震荡，离心速度控制在 3000~4000r/min。② 污染问题：采集的脐带血可能会受到母血污染。规避方法：严格遵守消毒规程。③ 标记、记录错误，和编码本不匹配。规避方法：标记记录时应保持专注，遵守编码规则，并有另一人在场核验。

3.6 数据整理与质控管理

前期数据整理与溯源管理是保障突变分析结果可追溯、可重复的重要环节，涵盖样本临床数据、实验操作数据、试剂耗材数据等多方面内容。规范的数据整理可提高实验效率，便于后续结果分析；完善的溯源管理可及时排查实验问题，保障数据的真实性和可靠性。本节明确数据整理的内容、方法及溯源管理的具体要求，规避数据缺失、记录混乱、溯源困难等常见问题。1992 年，美国临床实验室改进修正案（CLIA）颁布，其中包含了临床实验室人员、设施、质量保证、质量控制、能力测试、记录保存和记录保留的规定，并在随后提出质控体系的概念。

3.6.1 数据整理的核心内容

分析前需整理的数据主要是临床 / 人群基础数据和实验操作数据，每类数据需明确整理要点和标准，确保数据完整、准确。

3.6.1.1 临床数据整理

临床数据是解读突变与病程关联的重要依据，需从患者病历中提取并整理，核心内容包括：

- 1) 患者基本信息：姓名、性别、年龄、病历号、联系方式（用于随访）。
- 2) 疾病相关信息：HBV 感染时间、病程阶段（急性 / 慢性 / 肝硬化 / 肝癌）、诊断依据（如病理活检结果、影像学检查结果）、病毒学指标（HBsAg、HBcAg、抗-HBc、HBV DNA 载量、基因型）、肝功能指标（ALT、AST、胆红素、白蛋白等）。
- 3) 治疗相关信息：是否接受抗病毒治疗、治疗药物（如替诺福韦酯、恩替卡韦）、治疗时长、治疗效果（病毒载量变化、血清学转换情况）、是否出现耐药。
- 4) 其他信息：合并疾病（如丙肝感染、糖尿病、高血压、贫血等）、家族病

史（如肝癌家族史）等。

整理方法：采用 Excel 或专业临床数据管理软件（如 EpiData, <https://epidata.dk/>）建立临床数据库，每例患者作为一行，每个信息项作为一列，明确标注数据来源（如病历号、检查报告编号）和采集时间；对于缺失的数据，需及时补充核实，无法补充的需注明原因（如患者拒绝提供、病历记录缺失）。

3.6.1.2 实验操作数据整理

实验操作数据是保障实验结果可重复的关键，需实时记录并整理，核心内容包括样本处理信息，如，标本编号、处理方法及参数、时间等；前期检测数据，如，病毒 DNA 定量结果、血清学标志物检测结果、PCR 体系参数。

整理方法：建立实验操作记录表，每批实验单独记录，明确标本编号、操作日期、操作人员；数据记录需实时、准确，避免事后补记；对于异常数据（如核酸纯度不合格、对照结果异常），需记录异常现象和处理措施（如补做、更换试剂）。

3.6.2 数据整理的规范与要求

为确保整理后的数据可用于后续分析，需遵循以下规范与要求：

1) 数据标准化：对不同格式的数据进行标准化处理，例如，HBV DNA 载量统一用“IU/mL”作为单位，治疗效果统一用“有效 / 无效 / 耐药”标注，避免因数据格式不统一导致的分析误差。

2) 数据核查：整理完成后需进行数据核查，包括逻辑核查（如患者年龄与病程是否匹配、治疗药物与治疗时间是否合理）和准确性核查（如核对临床数据与病历记录是否一致、实验数据与原始记录是否一致）；对于核查出的错误数据，需及时修正并注明修正原因。

3) 数据备份：整理后的数据库需进行多重备份，包括本地硬盘备份、移动硬盘备份、云端备份（如实验室内部云服务器），避免数据丢失；备份数据需定期更新，确保与原始数据一致。

4) 数据保密：临床数据涉及患者隐私，需严格遵守《医疗纠纷预防和处理条例》《数据安全法》等相关法规，对患者个人信息进行脱敏处理（如隐去姓名、联系方式，仅保留病历号）；数据仅用于研究目的，不得泄露给第三方；非必要不使用线上传输数据。

3.6.3 样本质量控制

样本质量是质量控制的核心，需从样本采集、保存、预处理三个环节进行严格控制，确保样本符合实验要求：

1) 采集环节质量控制：① 核查样本来源的准确性：采集前再次核对患者信息，避免采集错误样本；② 控制样本污染：采血时使用一次性耗材，适量充装，避免溢洒，避免交叉污染；③ 避免样本溶血或组织坏死：采血时动作轻柔，避免剧烈震荡；组织样本离体后及时处理保存，避免坏死。

2) 保存环节质量控制：① 核查保存条件：定期检查冰箱温度（ -20°C 、 -80°C 冰箱），确保温度稳定（波动范围 $\leq \pm 2^{\circ}\text{C}$ ）；② 避免反复冻融：样本分装保存，每次使用一管，冻融次数控制在3次以内；③ 样本标记核查：定期核查样本标记，确保标记清晰、无混淆，对标记模糊的样本及时重新标记或废弃。

3) 预处理环节质量控制：核酸提取质量验证：提取后的核酸需检测产量和纯度，DNA 样本 A260/A280 比值需在 1.8~2.0 之间，RNA 样本需在 1.9~2.1 之间；若纯度不合格（如比值低于 1.8，提示存在蛋白质污染；高于 2.0，提示存在 RNA 污染），需重新提取或纯化。

3.6.4 实验操作质量控制

实验操作的规范性是保障结果可重复的关键，需通过标准化操作流程、人员培训、操作核查进行控制：

1) 建立标准化操作流程（SOP）：针对样本采集、保存、预处理、核酸提取等每一个操作环节，制定详细的 SOP，明确操作步骤、参数设置、注意事项，实验人员需严格按照 SOP 操作，不得随意更改操作流程。

2) 人员培训与考核：实验人员需经过系统培训，熟悉 SOP 和实验原理，考核合格后方可独立操作；定期组织技能培训和经验交流，及时更新操作规范，提高操作水平。

3) 操作过程核查：建立双人核查制度，关键操作步骤（如样本编号核对）需由两名实验人员共同核查，确保操作准确；实验过程中实时记录操作情况，对异常操作及时记录并分析原因。

4) 实验环境控制：实验操作需在符合要求的环境中进行，如核酸提取在超净工作台中进行，PCR 反应在专门的 PCR 实验室进行（避免扩增产物污染）；定

期对实验环境进行清洁消毒，监测环境微生物污染情况。

3.6.5 数据质量控制

数据质量控制确保整理后的数据完整、准确、可追溯，需从数据采集、整理、核查三个环节进行控制：

1) 数据采集控制：临床数据采集需由专人负责，核对病历记录，确保数据真实；实验数据需实时记录，避免事后补记，确保责任明确。

2) 数据整理控制：采用标准化的数据格式，对不同来源的数据进行统一整理；建立数据录入核查制度，数据录入后由专人进行复核，避免录入错误。

3) 数据核查控制：定期对整理后的数据库进行逻辑核查和准确性核查，例如，进行离群值筛查，避免录错项目；对缺失数据和错误数据，及时补充修正，无法修正的注明原因；建立数据溯源体系，妥善保存原始数据，确保每一个数据都能追溯到原始记录。

3.6.6 常见质量问题及处理措施

准备阶段常见的质量问题包括样本污染、核酸降解、数据缺失、试剂失效等，需及时发现并采取针对性的处理措施，具体如下：

1) 样本污染：表现为阴性对照检测阳性、不同患者样本出现相同的突变序列。处理措施：立即停止实验，排查污染来源（如耗材、操作环境、操作人员）；对实验环境和耗材进行彻底消毒，更换污染的试剂和耗材；重新采集样本进行实验。

2) 核酸降解：表现为核酸电泳出现拖尾、PCR 扩增无条带或条带微弱。处理措施：检查样本保存条件和冻融次数，若样本已严重降解，需重新采集样本；优化核酸提取流程，确保提取过程中核酸不被降解；提取后的核酸及时保存，避免长时间放置。

3) 数据缺失：表现为临床数据或实验数据不完整。处理措施：及时查阅病历记录、实验原始记录，补充缺失数据；无法补充的，注明缺失原因；对于关键数据缺失的样本，若影响后续分析，需排除该样本。

4) 试剂失效：表现为阳性对照检测阴性、核酸提取效率极低。处理措施：检查试剂有效期和储存条件，确认试剂是否失效；更换新的试剂，重新进行实验；对新试剂进行质量验证，确保试剂有效。

3.7 控制生物安全风险：排除、降低、保护

在病原微生物实验室中，生物危险源，主要是病原微生物能致病的固有危险性、实验室操作和环境造成的特殊存在条件，以及人员环境暴露可能存在的危险触发因素。当危险源发生一定程度的失控，如，含有病原体的未被彻底清理的溶液、肉眼不可见的气溶胶，沾染病原体的器械、衣物、台面等，超过特定控制范围，即形成潜在的生物安全风险。较低生物安全级别的实验室，往往没有系统的生物安全监测和报告机制，使得实验室获得性病毒感染风险很大程度是未知的^[1]。在这种特殊环境下，传播途径可能和公开报道的不一样。比如乙肝病毒的传播途径一般认为是不包括呼吸道传播，但根据 Sewell 等^[2]报道的一项临床实验室调查，26.7% 的实验室获得性感染由病毒引起，其中很大一部分是由乙肝病毒引起，气溶胶暴露或吸入是主要传播方式（92%）。随着个人防护装备和实验设备的发展进步，现在的生物危险源已能得到很好的控制，并尽可能阻断触发因素。

风险控制的三大技术可以概括为：排除、降低和保护，同样适用于生物安全领域。排除，如在生物安全柜操作，最大程度避免感染性气溶胶扩散；实验室内的物品不带出实验室，尤其是笔和实验记录本，避免危险源外溢到控制区外，即是一定程度的排除。降低，是在条件有限，不能完全控制危险源的情况下，降低风险发生的概率和严重程度。如，规范的操作方法和充分的事先练习，可以减少污染物溢洒、飞溅，减少利器伤发生概率。保护，如，进行对应的免疫接种、规范佩戴个人防护装备等。危险是客观存在的，风险控制取决于人的主观意识。操作病原微生物需要提前做好规划，熟悉危险源的性质和控制方式，预设和熟悉应急处置流程，保持谨慎，形成良好的生物安全习惯。限于篇幅，本节仅提炼生物安全医学防护的关键要点，相关理论体系与实操细则的完整叙述，详见张磊主编《生物安全医学防护实训指导》^[1]，可拓展阅读张军、张磊主编《生物安全》^[2]。

本章小结

本章系统阐述了 HBV 突变分析前实验与数据准备的全流程核心要点，明确了该环节是保障后续突变检测结果准确、可靠且可重复的基础前提。整体内容围绕“科学设计、规范操作、质量可控、安全合规”的核心原则展开，为突变分析

实验的顺利开展筑牢根基。

在研究启动阶段，强调需以假说为导向确立研究选题，明确测量指标与方法，同时秉持开放辩证思维应对研究进程中的目标调整；实验分组设计需严格遵循“随机、对照、重复”原则，结合研究类型选择真实验设计或准实验设计，明确分组变量、保障样本同质与样本量充足，针对不同研究场景提供了具体设计案例与分析思路，同时厘清了病例对照研究等观察性研究的核心特征、适用场景及偏倚防控对策，为研究设计的科学性提供支撑。

样本相关环节核心聚焦“样本有效、操作规范”：明确样本量估算的核心参数与适用公式，需结合研究类型、预期流行率等科学确定样本量，平衡可行性与研究效力；样本选型以“核酸优质、污染少”为原则，优先推荐血清等血液样本，明确各类样本适用场景；规范样本采集、保存与预处理全流程，涵盖志愿者招募标准、血液样本采集规范、保存条件及血清/血浆分离步骤，重点规避溶血、污染、标记错误等关键失误。

数据与质量管控以“完整准确、全程可控”为目标：界定临床与实验数据的核心整理内容，明确标准化、核查、备份及保密要求；构建全链条质控体系，覆盖样本处理、实验操作各环节，明确核酸质量验证与数据溯源标准，给出常见质量问题的应对措施；同时确立“排除、降低、保护”生物安全风险控制技术，强调通过规范操作与防护规避实验室感染风险。

综上，本章通过梳理实验与数据准备各环节的实操流程、标准要求与注意事项，形成了完整的前期准备技术体系，既为研究者提供了可落地的操作指引，也凸显了前期准备对突破 HBV 突变分析技术瓶颈、提升研究科学性的关键作用。

第二部分 获取基因数据

第4章 检测血液中的HBV

HBV 血清标志物和 HBV-DNA 作为乙型肝炎病毒复制的核心标志物，其在血液中的检出与定量结果是临床诊断、病情分期、抗病毒治疗监测及耐药预警的关键依据。上世纪 90 年代到本世纪初，检测血清 HBV-DNA 的标准方法是斑点杂交，可以对 DNA 含量进行相对粗略的估计。另一种方法是基于 PCR 技术的检测方法，具备特异性好、流程简便的优点。随着 PCR 技术的发展，以及对 HBV 基因谱及酶切位点认识的完善，基于 PCR 技术的定量检测已成为目前主流的检测方法。血液 HBV-DNA 检测的核心流程包括核酸提取与定量检测两大环节，前者需实现病毒核酸的高效富集与纯化，后者需保障检测的特异性、灵敏度与准确性。

4.1 HBV 血清标志物检测

标志物检查由于简便快速、成本低廉，相对于作为附加项目的基因检测，是一种普适性更好的筛查方法。在医院检查项目中，HBV 血清标志物检测又称“乙肝五项”或“乙肝两对半”，检测内容包括乙型肝炎病毒表面抗原（HBsAg）、表面抗体（抗-HBs）、e 抗原（HBeAg）、e 抗体（抗-HBe）及核心抗体（抗-HBc），其检测结果可明确 HBV 感染状态、免疫应答情况及病情进展阶段，是 HBV 感染筛查与临床评估的基础手段。目前临床常用检测方法以酶联免疫吸附试验（ELISA）为主，同时包括化学发光免疫分析、胶体金免疫层析等技术，本节重点阐述酶联免疫吸附（ELISA）检测的核心原理、通用流程，并简要介绍其他检测方法的技术特点。

4.1.1 ELISA 检测原理

乙肝五项 ELISA 检测主要采用双抗体夹心酶联免疫吸附原理（适用于抗原

检测,如 HBsAg、HBeAg)或间接酶联免疫吸附原理(适用于抗体检测,如抗-HBs、抗-HBe、抗-HBc)。

1) 双抗体夹心原理:将特异性捕获抗体预先包被于酶标板微孔内,形成固相载体;向微孔中加入待检测样本,若样本中存在目标抗原,将与固相载体上的捕获抗体特异性结合;洗涤去除未结合的杂蛋白后,加入辣根过氧化物酶(HRP)标记的特异性检测抗体,该抗体与已结合在固相载体上的抗原结合形成“捕获抗体-抗原-标记检测抗体”免疫复合物;再次洗涤后加入底物(如 TMB),TMB 在 HRP 催化下发生显色反应,生成蓝色产物,加入终止液后转变为黄色;颜色深浅与样本中目标抗原浓度呈正相关,通过酶标仪检测特定波长(通常 450nm)下的吸光度(OD 值),结合标准曲线可实现抗原的定量或定性分析。

2) 间接免疫吸附原理:将特异性抗原预先包被于酶标板微孔内;向微孔中加入待检测样本,若样本中存在目标抗体,将与固相抗原特异性结合;洗涤后加入 HRP 标记的抗人免疫球蛋白抗体(二抗),二抗与样本中的目标抗体特异性结合形成免疫复合物;后续显色、终止及检测流程与双抗体夹心原理一致,通过 OD 值判断样本中目标抗体的存在与否及相对含量。

4.1.2 ELISA 检测流程

不同品牌 ELISA 试剂盒区别主要在专利抗体、酶结合物,以及工作液配比等,从而产生具体配置比例、孵育条件等差异,总体流程和判读方法大同小异。

1) 试剂准备:根据试剂盒说明书要求,将各组分恢复至工作温度。按说明书比例配置洗涤液、标准品、工作液。

2) 上样:在酶标板微孔内分别设置标准品孔、样本孔及空白孔,向标准品孔加入不同浓度的标准品,样本孔加入处理后的待检测样本,空白孔加入等量样本稀释液,加样体积通常为 50~100 μL ,加样时需将液体加入微孔底部,避免触及孔壁产生气泡,全程加样时间控制在 10 分钟内以保证反应一致性。

3) 孵育:上样完成后,用封板膜覆盖酶标板,置于 37℃恒温箱或水浴槽中温育,抗原检测温育时间通常为 60 分钟,抗体检测温育时间可根据试剂盒要求调整,温育过程需保证恒温环境,避免样本蒸发。

4) 洗涤:温育结束后,弃去孔内液体,在吸水纸上拍干,每孔加入稀释后的洗涤液 350 μL ,浸泡 1 分钟后弃去洗涤液,吸水纸上拍干;重复洗涤步骤 3~5 次,

确保去除未结合的游离成分，洗涤不充分会导致背景值升高。

5) 加标记抗体与二次孵育：向除空白孔外的所有微孔中加入底物，每孔 $50\mu\text{L}$ ，覆盖封板膜后 37°C 温育 15~30 分钟（一些试剂盒会要求避光孵育），温育结束后按上述洗涤流程重复洗涤。

6) 显色与终止：向每孔加入底物溶液（A 液与 B 液等体积混合） $90\sim 100\mu\text{L}$ ，覆盖封板膜后 37°C 避光温育 15 分钟左右，当标准品孔出现明显梯度显色时，立即向每孔加入 $50\mu\text{L}$ 终止液终止反应，终止液加入顺序需与底物溶液一致。

7) 检测：终止反应后 15 分钟内，用酶标仪在 450nm 波长（允许 $\pm 10\text{nm}$ 偏差）下测定各孔 OD 值，检测前需预热酶标仪并校准检测通道。

4.1.3 ELISA 检测结果判读

1) 数据校正：计算各标准品及样本复孔 OD 值的平均值，扣除空白孔 OD 值作为校正 OD 值。

2) 标准曲线绘制：以标准品浓度为横坐标，校正 OD 值为纵坐标，采用四参数逻辑函数或线性回归方法拟合标准曲线，要求相关系数（ R^2 ） ≥ 0.99 ，确保标准曲线的可靠性。

3) 结果判定：根据标准曲线计算样本中目标物浓度，结合试剂盒参考范围判断阳性、阴性结果；若样本 OD 值高于标准曲线上限，需稀释后重新检测，最终浓度需乘以稀释倍数；若 OD 值低于检测下限，则判定为阴性或低于检测阈值。

4.1.4 其他 HBV 血清标志物检测方法

除 ELISA 方法外，乙肝五项检测还有以下两种主流技术，各有其技术特点与应用场景：

1) 化学发光免疫分析（CLIA）：以化学发光剂代替 ELISA 中的酶底物，通过抗原抗体特异性结合后触发化学发光反应，检测发光强度实现定量分析。该方法具有灵敏度更高（可检测低浓度标志物）、线性范围更广、检测速度更快、自动化程度高等优势，且减少了显色反应带来的主观误差，适用于大批量样本的精准检测，但仪器与试剂成本较高，广泛应用于三级医院及大型检验机构。

2) 胶体金免疫层析法（GICA）：基于胶体金颗粒的显色特性与免疫层析技术，将特异性抗原或抗体固定于硝酸纤维素膜上，样本通过毛细作用层析移动，

若样本中存在目标物，将与膜上试剂结合形成肉眼可见的红色条带。该方法操作简便、快速（检测时间通常 5~10 分钟）、无需特殊仪器，适用于床旁快速筛查、基层医疗机构及应急检测，但检测灵敏度较低，仅能实现定性分析，无法准确定量，且结果判读易受主观因素影响。

4.1.5 HBV 血清标志物结果解读

HBsAg	HBsAb	HBeAg	HBeAb	HBcAb	意义
-	-	-	-	-	无 HBV 感染史，若不接种疫苗，对 HBV 易感
+	-	-	-	-	HBV 感染早期，或慢性携带者，传染性弱
-	+	-	-	-	有疫苗接种史；HBV 感染康复。已有免疫力（考虑抗体水平）
-	-	+	-	-	非典型急性感染
-	-	-	+	-	感染趋向恢复，无感染性
-	-	-	-	+	急性感染窗口期；既往感染恢复期；既往感染暂未检出 HBsAb
+	+	-	-	-	乙肝亚临床感染早期；不同亚型 HBV 二次感染
+	-	+	-	-	早期 HBV 感染；慢性携带者活动期，感染性强
+	-	-	+	-	HBsAg 携带者易转阴；急性感染趋向恢复，传染性较弱
+	-	-	-	+	俗称“小二阳”，具有一定传染性
-	+	+	-	-	亚临床感染或非典型感染
-	+	-	+	-	感染已恢复
-	+	-	-	+	既往感染，具有免疫力；非典型感染恢复期
-	-	+	-	+	非典型急性感染
-	-	-	+	+	既往感染；急性乙肝恢复期，基本无传染性
+	+	-	+	-	亚临床感染；非典型乙肝
+	+	-	-	+	亚临床感染早期；不同亚型乙肝病毒二次感染
+	-	+	-	+	俗称“大三阳”，可能是慢性 HBV 携带者，或 e 抗原阳性的慢性乙肝，传染性强
+	-	-	+	+	俗称“小三阳”，可能是非活动性 HBsAg 携带者，或 HBeAg 阴性慢性乙型肝炎，传染性较强
-	+	-	+	+	感染恢复期，已有免疫力
-	+	+	-	+	亚临床感染或非典型乙肝
+	-	+	+	-	急性感染趋向恢复；慢性 HBV 携带者
+	+	+	-	-	亚临床感染或非典型乙肝
-	-	+	+	+	感染中期

4.2 HBV-DNA 提取技术

HBV-DNA 提取是后续检测的基础，核心目标是从血液样本（血浆 / 血清）中分离获得高纯度、高完整性的病毒 DNA，同时去除蛋白质、核酸酶、血红蛋白等抑制物及干扰成分。目前临床常用提取技术包括自动化提取与手动提取两类，其中自动化提取因操作标准化、耗时短、污染风险低等优势，更适用于常规诊断实验室。

4.2.1 DNA 提取原理

无论自动化还是手动提取，均基于“裂解 - 结合 - 洗涤 - 洗脱”的核心原理：
① 细胞裂解：利用含去污剂的裂解液破坏病毒衣壳结构，使病毒 DNA 释放至溶液中，部分提取体系可通过蛋白酶降解样本中的蛋白质成分，避免其对核酸的包裹或破坏；② 核酸结合：在高盐环境下，病毒 DNA 特异性结合于硅胶膜、特殊磁珠等吸附介质表面，而蛋白质、多糖等杂质留存于溶液中；③ 洗涤纯化：通过梯度洗涤液依次去除吸附介质表面残留的蛋白质、盐离子等杂质，降低后续 PCR 反应的抑制风险；④ 洗脱回收：在低盐、低 pH 值的洗脱缓冲液作用下，破坏核酸与吸附介质的结合力，使高纯度 HBV-DNA 释放至洗脱液中。

4.2.2 样本预处理

1) 自动化提取样本预处理：以 COBAS AMPLIPREP 分析仪为例，常规诊断实验室因样本量有限，通常选择 200 μ L 处理体积，需至少加入 300 μ L 血清至带 barcode 标记的试管中。提取前需将内参对照与内参定量标准稀释液按比例混合，由仪器自动加入每个样本中，以监控提取过程的有效性 & 后续 PCR 反应的抑制情况。

2) 手动提取样本预处理：按病毒基因组 DNA/RNA 提取试剂盒说明书要求，对血清样本进行预处理，通常加入 200 μ L 血样，确保样本中病毒颗粒的完整性，为后续裂解步骤奠定基础。不同标本类型因成分差异，最好选择适配的试剂盒。提取 DNA 的要求相对较低，一般区别在标本蛋白质含量，故也可不换试剂盒，根据预先调查选择增加蛋白酶用量。

4.2.3 操作流程

4.2.3.1 自动化提取流程

一般自动化 DNA 提取系统配有专机专用试剂，方便参数设定和流程标准化。各品牌自动提取仪使用流程或多或少有区别，但大同小异。以 COBAS AMPLICOR 系统为例：

1) 实验准备：实验需在Ⅱ级生物安全柜内进行，所有耗材（离心管、吸头）需为无 RNase/DNase 的无菌产品；提前检查 COBAS AMPLIPREP 分析仪状态，确保仪器正常运行。

2) 试剂与样本装载：将 COBAS AMPLIPREP Total Nucleic Acid Isolation 试剂盒试剂装载至仪器试剂仓，将标记好的样本管装载至样本仓，同时加入配置好的内参对照混合液。

3) 自动化提取运行：启动仪器提取程序，仪器自动完成样本裂解、核酸结合、梯度洗涤等步骤；最终在 80℃ 条件下，用 75 μ L 样本稀释液洗脱纯化的核酸。

4.2.3.2 人工提取流程

1) 试剂准备：将提取试剂盒各组分恢复到说明书建议的温度（如一些酶要求工作温度为常温），若有悬浊液类试剂，需通过振动充分混匀，若容易挂壁需要瞬离。

2) 裂解反应：取 200 μ L 血清 / 血浆样本加入离心管中，加入适量裂解缓冲液，涡旋振荡混匀后，按试剂盒要求孵育，一般为室温条件人工颠倒混匀和静置循环，使病毒颗粒充分裂解。

3) 核酸结合与洗涤：向裂解液中加入等量结合缓冲液，在柱层析或薄膜层析装置中，混匀后的含 DNA 溶液转移至吸附柱中，离心使核酸结合于吸附膜上；在磁珠法流程中，含磁珠 -DNA 结合颗粒的混悬液在磁力架固液分离，吸除无 DNA 溶液。依次加入洗涤液 1、洗涤液 2 进行梯度洗涤，每次洗涤后吸除上清液（无 DNA 溶液）。

4) 干燥和洗脱：在吸附柱分离法中，将吸附柱开盖置于 65℃ 恒温箱中干燥 5 分钟，去除残留洗涤液；加入 25~50 μ L 洗脱缓冲液，65℃ 孵育 10 分钟后离心，收集洗脱液即为纯化的 HBV-DNA 模板。在磁珠法中，吸除缓冲液后静置，使试剂中的乙醇挥发，但不能太干以免磁珠贴壁；加入无酶水 80 μ L，在孵育床孵育，

参数设为 45℃, 450rpm, 7min。在磁力架固液分离, 吸取含 DNA 的溶液。

4.2.3.3 质控和保存

1) 内参监控: 自动化提取过程中需全程加入内参对照, 若内参无信号, 提示样本存在 PCR 抑制物或提取过程失败, 需重新提取样本。

2) 人工提取流程本身没有质控环节, 其重点在于谨慎操作, 减少 DNA 损失。包括但不限于注意操作流程, 避免误弃含 DNA 溶液、足量蛋白酶和适宜孵育条件以使蛋白质充分裂解、尽量缩短实验时间以减少 DNA 降解。

3) 提取后处理: 提取完成后, 取出洗脱液置于无核酸酶离心管中, 立即用于后续检测或置于 -20℃短期保存、-80℃长期保存, 避免反复冻融 (冻融次数不超过 3 次)。

4.3 核酸纯度检测技术

提取后的 HBV 核酸需进行纯度和浓度检测, 以评估提取质量, 判断是否满足后续实验要求。本节介绍两种常用的检测方法 (Nanodrop 紫外分光光度法、Qubit 荧光定量法), 明确检测原理、实操步骤、结果判断标准及注意事项, 同时介绍核酸完整性的检测方法, 为提取质量评估提供全面指引。

4.3.1 Nanodrop 紫外分光光度法 (快速定量与纯度检测)

紫外分光光度计检测原理为, 核酸在 260nm 波长处有最大吸收峰, 蛋白质在 280nm 波长处有最大吸收峰, 通过检测核酸样本在 260nm、280nm、230nm 波长处的吸光度值, 可实现核酸浓度和纯度的快速检测。浓度计算: DNA 浓度 ($\text{ng}/\mu\text{L}$) = $A_{260} \times 50 \times \text{稀释倍数}$; RNA 浓度 ($\text{ng}/\mu\text{L}$) = $A_{260} \times 40 \times \text{稀释倍数}$ 。纯度判断: A_{260}/A_{280} 比值反映核酸纯度, A_{260}/A_{230} 比值反映核酸中是否存在盐离子、有机溶剂等杂质。

基本流程: ① 仪器预热: 打开 Nanodrop 仪器, 预热 10 分钟, 校准仪器 (用无核酸酶水作为空白对照); ② 样本准备: 取 $1\mu\text{L}$ 提取后的核酸样本, 若浓度过高, 用无核酸酶水稀释 (稀释倍数根据预估浓度确定, 一般稀释 5~10 倍); ③ 检测: 用无核酸酶水擦拭检测平台, 滴加 $1\mu\text{L}$ 空白对照 (无核酸酶水), 点击 “Blank” 进行空白校准; 校准完成后, 擦拭检测平台, 滴加 $1\mu\text{L}$ 样本, 点击

“Measure”进行检测；检测完成后，记录 A260、A280、A230 值及浓度值；④清理：检测完成后，用无核酸酶水彻底擦拭检测平台，关闭仪器。

结果判断标准：① 浓度要求：HBV 核酸浓度 $\geq 20\text{ng}/\mu\text{L}$ （适用于 PCR、测序等实验），低病毒载量样本浓度 $\geq 10\text{ng}/\mu\text{L}$ ；② 纯度要求：DNA 样本 A260/A280 比值 1.8~2.0，RNA 样本 A260/A280 比值 1.9~2.1（比值 <1.8 ，提示存在蛋白质污染；比值 >2.0 ，提示存在 RNA 污染）；③ 杂质判断：A260/A230 比值 ≥ 2.0 （比值 <2.0 ，提示存在盐离子、酚、乙醇等杂质）。

注意事项：① 检测平台需保持清洁，避免样本残留导致交叉污染；② 样本体积需准确（ $1\mu\text{L}$ ），否则会影响检测结果；③ 该方法易受杂质干扰，若样本中存在大量蛋白质、盐离子等，检测结果会偏高，需结合 Qubit 荧光定量法验证。

4.3.2 终点法荧光定量

荧光定量法在质检环节和后面的定量环节有点区别，该环节多为终点法，其结果判读和 Nanodrop 类似，最后得到的是纯度比值。而荧光定量是基于循环阈值（Ct 值）和标准曲线计算的拷贝数。荧光定量法核酸纯度检测的基本原理为，基于特异性荧光染料与核酸的结合作用，荧光探针仅与 DNA 或 RNA 结合，不与蛋白质、盐离子等杂质结合，通过检测荧光强度，可实现核酸的精准定量。该方法特异性高，不受杂质干扰，检测结果更准确。

以 Qubit 荧光计为例，基本流程：① 试剂准备：取定量试剂盒中的缓冲液和荧光染料，按 1:200 的比例混合（如 $100\mu\text{L}$ 缓冲液 + $0.5\mu\text{L}$ 荧光染料），制备工作液，轻轻颠倒混匀，避免产生气泡；② 标准曲线制备：取 Qubit 标准品 1 和标准品 2，各加入 $190\mu\text{L}$ 工作液和 $10\mu\text{L}$ 标准品，轻轻吹打混匀；③ 样本制备：取 $198\mu\text{L}$ 工作液加入检测管中，再加入 $2\mu\text{L}$ 提取后的核酸样本，轻轻吹打混匀；④ 检测：打开 Qubit 仪器，选择对应的检测程序（DNA 或 RNA），依次放入标准品管和样本管，点击“Read”进行检测；检测完成后，记录样本浓度值。

结果判断标准：① 浓度要求与 Nanodrop 法一致（ $\geq 20\text{ng}/\mu\text{L}$ ）；② 该方法检测范围为 $10\text{pg}/\mu\text{L} \sim 1\mu\text{g}/\mu\text{L}$ ，若样本浓度超出检测范围，需稀释后重新检测；③ 标准曲线的 R^2 值需 ≥ 0.99 ，否则检测结果无效，需重新制备标准曲线。

注意事项：① 工作液需现配现用，避免长时间放置导致荧光染料失效；② 标准品和样本需充分混匀，否则会影响荧光强度；③ 检测管需清洁无荧光污染，

避免影响检测结果。

4.3.3 PCR 产物长度和完整性检测（琼脂糖凝胶电泳法）

电泳在 DNA 提取环节用处主要是在 PCR 结束后检测 PCR 产物长度和完整性。其基本原理为，琼脂糖凝胶具有分子筛作用，核酸分子在电场中按分子量大小迁移，分子量越大，迁移速度越慢。通过电泳可观察核酸的条带完整性，判断是否存在降解。

基本流程：① 凝胶制备：称取 0.5g 琼脂糖，加入 50mL 1X TAE 缓冲液（科昊生物工程有限公司 TAE-500，50X TAE），加热溶解后冷却至 50~60℃，加入 5 μL 胶染色液（YeaGreen nucleic acid gel stain, Y256040），轻轻混匀，倒入制胶板中，插入梳子，室温凝固 15 分钟；② 上样：拔出梳子，将凝固后的凝胶放入电泳槽中，加入 TAE 缓冲液（没过凝胶），将样本和 DNA Marker（天根公司 DNA Marker III MD103）依次加入上样孔中；③ 电泳：设置电泳参数（电压 180V，时间 15 分钟）；④ 成像：电泳完成后，将凝胶放入凝胶成像系统中，观察核酸条带，紫外曝光 1000ms。

结果判断标准：① 完整的 DNA 样本应呈现清晰的单一条带，无明显拖尾（拖尾提示核酸降解或没有成功聚合）。

注意事项：① 目前很多品牌的胶染料基本没有毒性，但操作时仍需遵守一般规则，佩戴一次性 PE 手套或橡胶手套；② 电泳电压不宜过高，否则会导致条带模糊；③ 电泳时间不宜过长，避免体系过热；④ 出现杂带也没关系，只要目标条带存在且不与杂带贴近、重合即可。

4.4 HBV-DNA 定量检测技术

用于定量检测血清中乙型肝炎病毒（HBV）DNA 的方法已被证明可用于治疗前评估、临床分期、抗病毒治疗监测、耐药性出现的检测以及抗病毒治疗停止后复发的检测。基于 PCR 原理的荧光定量检测是目前主流的 HBV-DNA 定量检测方法。该方法通过特异性扩增靶序列并结合荧光信号检测实现定量分析。目前临床主流技术为实时荧光定量 PCR（Real-Time PCR），该技术将扩增与检测在同一封闭体系中完成，可减少污染风险并缩短检测周期。此外，终点荧光 PCR

与常规 PCR 电泳检测也可用于 HBV-DNA 检测,但灵敏度与定量准确性相对较低。

4.4.1 荧光 PCR 定量检测原理

荧光 PCR 核心原理为,以提取的 HBV-DNA 为模板,在 Taq 聚合酶、dNTPs、特异性引物及荧光探针 (TaqMan 探针) 的作用下,通过变性 - 退火 - 延伸的循环反应实现靶序列的指数级扩增。荧光探针预先结合靶序列,扩增过程中 Taq 聚合酶的 5' → 3' 外切酶活性降解探针,使探针上的报告荧光基团与淬灭基团分离,产生荧光信号;荧光信号强度与扩增产物量呈正比,通过实时监测每一轮循环的荧光强度,绘制扩增曲线,根据曲线对应的循环阈值 (Ct 值) 与已知浓度的标准曲线对比,计算样本中 HBV-DNA 的拷贝数。

4.4.2 实验准备

4.4.2.1 试剂准备

DNA 荧光定量试剂通常包括反应液、阳性质控品 (一些试剂盒还有不同梯度浓度的 HBV 阳性定量参考品)、阴性质控品、酶液。一些试剂要求冰冻保存,需预先在冰上解冻并恢复至工作温度。在冰上配置反应体系预混液。

4.4.2.2 样本准备

在冰上解冻;若样本 HBV-DNA 浓度过高 (超过检测线性范围上限),需用无酶纯水进行梯度稀释。

4.4.3 操作流程

1) 上样:使用上节提取的 DNA,按照试剂盒说明书,与阳性质控品、阴性质控品加入相应的反应管,封好,混匀,瞬离。将反应管放入仪器反应槽。

2) 准备:一些品牌的仪器需要提前校准。

3) 设置信息:在电脑程序上选择检测区域,标记样品区和质控品位置,备注样品名称,设置标准品浓度。输入反应体系体积。选择荧光基团类型、检测通道、淬灭通道。设置循环参数,一般为 95℃ 预变性 10min,然后循环 35~40 次,根据说明书设置引物结合温度、反应时长,标定荧光读取阶段。设置结果文件保存位置。

4.4.4 解读结果

1) 若曲线整体偏斜, 根据结果曲线调整基线 (baseline) 和阈值 (threshold), baseline 的 start 值一般在 3~15 范围内调节, stop 值在 5~20 范围内调节。拖动阈值线使其高于阴性对照, 重新分析。

2) 阳性结果: 扩增曲线呈 S 形, 且 Ct 值 ≤ 35 , 提示血液中存在 HBV-DNA; 若样本 HBV-DNA 浓度超过检测线性范围 (通常为 $2 \times 10^3 \sim 1 \times 10^8$ copies/mL), 需稀释后重新检测。

3) 阴性结果: 无特异性扩增信号, Ct 值 >38 , 且内参信号正常, 提示血液中 HBV-DNA 拷贝数低于检测下限 (通常为 1×10^3 copies/mL)。

4) 可疑结果: 扩增曲线呈 S 形, 但 Ct 值介于 35~38 之间, 需重新提取样本进行检测; 若复测结果一致, 判定为阳性; 若复测 Ct 值 >38 或未检出, 判定为阴性。

本章小结

本章系统阐述了血清 HBV 检测的核心技术体系, 涵盖血清标志物检测, 核酸提取、纯度检测及定量检测三大关键环节, 从筛查到确证, 重视质控和结果解读, 贴近应用实际。提取环节以“裂解 - 结合 - 洗涤 - 洗脱”为核心原理, 明确自动化与人工两种主流模式的流程规范, 强调自动化提取的内参监控、人工提取的 DNA 损失控制及样本保存要求。核酸纯度检测作为衔接环节, 列举 Nanodrop 紫外分光光度法 (快速便捷)、Qubit 荧光定量法 (特异性高) 及琼脂糖凝胶电泳法 (产物完整性验证), 适合研究人员根据自身实验条件进行参考。定量检测以实时荧光定量 PCR 为核心, 阐释其特异性扩增与荧光检测的定量原理, 概述实验流程要点 (具体 SOP 需根据实际使用的试剂说明书), 明确通用结果判读标准。综上, HBV-DNA 检测需严守各环节规范, 强化全流程质控, 确保原始数据真实、精准、可靠, 为后续开展相关应用提供科学支撑。

第5章 基因扩增与测序技术

在上一章已完成 HBV 的筛查和 DNA 提取。要知道具体的基因信息，需要对低浓度的 HBV-DNA 进行扩增和测序。同时，对完整或部分 HBV 基因组进行 Sanger 测序是 HBV 感染诊断的金标准。扩增与测序是基因研究的核心技术，前者实现 HBV 靶基因的特异性富集，后者完成基因序列的精准解析，为 HBV 突变分析、基因分型、耐药监测、病毒进化研究等提供关键数据支撑。然而，这些研究领域的开展需要分析病毒在宿主体内的遗传多样性。该分析需要进行全基因组测序（whole-genome sequencing, WGS），因为特定基因测序可能会被重组现象误导研究。Lumley 等^[1]报告了 HBV 全基因组测序基因方法和结果，同时认为乙肝病毒的基因多样性是全基因组测序的关键挑战。当毒株改变后，仍然需要重新设计并测试引物，既往结果不易复现。

Sanger 测序和 Illumina 测序是目前常见的测序方法。Sanger 测序是成熟稳定的测序方法，但在 HBV 遗传多样性分析方面力有不逮；Illumina 可以进行深度测序虽然在描述遗传多样性方面有效，但无法对长片段 DNA 进行测序，同时会在扩增环节引入偏差。随着分子生物学技术发展，在大型患者队列调查和宿主体内 HBV 遗传多样性研究等领域，已经引入高通量测序（high-throughput sequencing, HTS）技术。HTS 技术及下游分析完美解决了 Sanger 测序和 Illumina 测序的问题，并大幅拓展了 HBV 基因组研究方向。虽然 HTS 技术具备很多引人注目的优势，但其所需要的软、硬件支持仍然较大，在很多地区和机构不易开展。本章着重介绍基于 Sanger 测序的传统引物设计方法和 PCR 方法，简化核心技术原理，细化实操流程，重点突出 HBV 专属的扩增与测序优化要点，同时对比不同测序技术的优劣及结果差异，明确全流程质量把控标准，为实验人员提供可落地的技术指引。

5.1 PCR 引物设计原则与验证

引物是 PCR 扩增特异性的核心保障，针对 HBV 基因高变异、部分双链环状的特性，引物设计需遵循“保守性、特异性、适配性”原则，同时通过实验验证确保引物的扩增效率与特异性。目前有多种引物设计工具可供选择，如 NCBI Primer-Blast、Primer3、Oligo7、Snapgene8 等。引物设计是一门大学问，想要进一步学习可以另找材料，如 Chhandak Basu 主编的《PCR Primer Design》。本节着重建立引物设计的标准化基础流程，并以 HBV 基因研究为例，阐述引物设计的一般准则、软件使用及引物验证方法，以便读者快速入门。

5.1.1 引物设计的一般准则和 HBV 补充准则

在设计引物时，应考虑以下因素：

1) 引物的 3' 端：引物分子的 3' 端对于 PCR 的特异性和灵敏性非常重要。应该避免在此位置有 3 个或更多 G 或 C 碱基，因为这可能会使引物非特异性退火。应该避免 3' 端为胸腺嘧啶，因为它比其他核苷酸更容易发生错配。应检查引物对在 3' 端的互补性，互补性过好通常会导致引物自身退火，形成引物二聚体，损害扩增过程中的引物利用率。

2) 引物长度：对于大多数 PCR 应用而言，18-22 个碱基的引物长度通常最佳。较短的引物可能导致非特异性 PCR 产物的扩增。

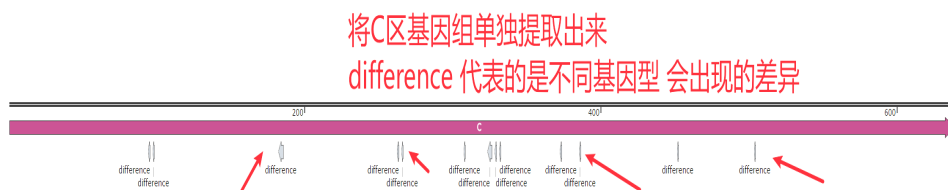
3) 解链温度 (T_m)：PCR 的特异性在很大程度上取决于引物的解链温度 (T_m)（即一半引物与模板退火时的温度）。通常，当两个引物的 T_m 相似（在 $2-4^{\circ}\text{C}$ 范围内）且在 $55-65^{\circ}\text{C}$ 范围内时，理想值为 60°C ，可获得良好结果。使用华莱士（Wallace）公式估算 T_m 值： $T_m = 2(A+T) + 4(G+C)$ 。实际上 T_m 值受到序列的影响较为复杂，不是这么简单就能算出的。还有一种邻近（nearest neighbor）可以估算 T_m 值，比华莱士公式更为精确。

4) GC 含量：引物的 GC 含量应在 40% 至 60% 之间。

5) 不良结构：引物不良结构一般由计算机程序计算。连续形成三个以上碱基对容易导致发卡结构。同向或对向引物间形成多个碱基对，会增大引物二聚体稳定性。以上情况都需要避免。

对于 HBV 引物设计，有以下补充准则：

1) 保守性原则: HBV 基因组存在 A、B、C、D 等多种基因型及大量变异株, 引物需设计在各基因型共有的保守区(可通过 NCBI GenBank 下载不同基因型 HBV 全序列, 用 Snapgene 软件查看变异位点, 或用 Clustal X 软件进行序列比对, 筛选保守区); 优先选择编码区保守序列(如 S 基因的 a 决定簇上游保守区、C 基因保守区), 避免非编码区的高变异区域。如图所示, Snapgene 里位点差异标注为 difference。如下图所示为 Snapgene 显示不同基因型位点差异。



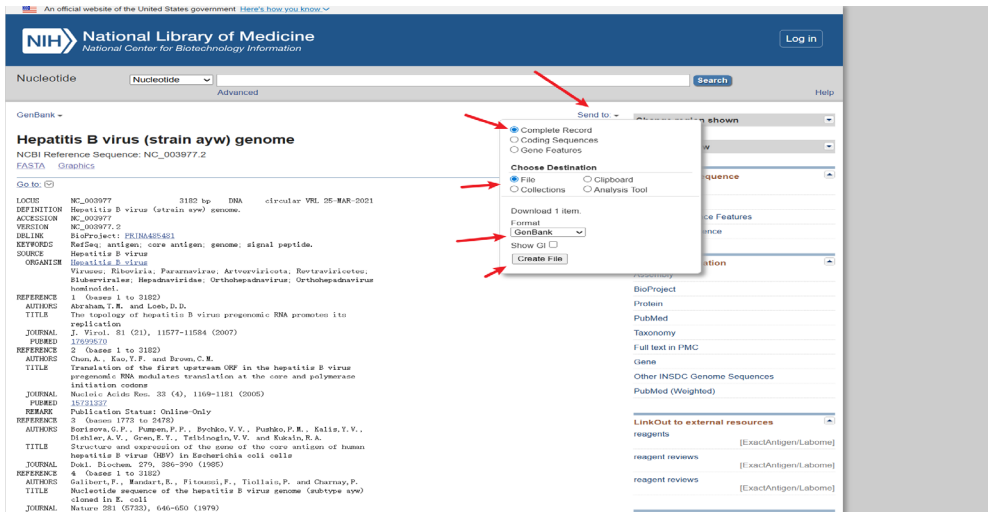
2) 特异性原则: 引物序列需与 HBV 靶基因唯一互补, 避免与人类基因组、其他病毒基因组(如 HCV、HIV)同源; 可通过 NCBI BLAST 工具进行特异性比对, 确保引物无非特异性结合位点; 引物自身及上下游引物之间避免形成发夹结构($\Delta G \leq -3\text{kcal/mol}$)或二聚体($\Delta G \leq -5\text{kcal/mol}$), 否则会降低扩增效率。

3) 适配性原则: ① 长度: 18~25bp, 过短易导致特异性差, 过长易导致退火温度过高、扩增效率低; ② T_m 值: 55~65°C, 上下游引物 T_m 值差异 $\leq 2^\circ\text{C}$, 便于设置统一的退火温度; ③ GC 含量: 40%~60%, 避免 GC 含量过高(>65%)导致引物二级结构形成; ④ 扩增片段长度: 常规 PCR 片段长度 200~1000bp(适配后续测序), qPCR 片段长度 80~200bp(提高扩增效率和定量准确性)。

4) 实用原则: 若用于突变检测, 可将引物 3' 端设计在突变位点附近, 提高检测灵敏度; 引物 5' 端可添加酶切位点(如 EcoR I、BamH I), 便于后续克隆。

5.1.2 下载参考序列

在 NCBI 数据库(https://www.ncbi.nlm.nih.gov/nuccore/NC_003977)下载基因序列, 文件可以保存为 GeneBank 格式或 FASTA 格式:



5.1.3 引物设计流程

1) 明确靶标：确定需检测的 HBV 耐药突变位点（如拉米夫定耐药相关的 rtL180M、rtM204V/I，位于 HBV P 基因），下载包含该位点的基因序列。

2) 序列比对：用 MEGA 软件对下载的序列进行多重比对，筛选保守区。

3) 引物设计：用 Snapgene (<https://www.snapgene.com/>)、Oligo (<https://www.oligo.net/downloads.html>) 或 Primer 软件, 基于筛选的保守区设计上下游引物, 设置参数 (长度 18~25bp、Tm 值 55~65℃、GC 含量 40%~60%、扩增片段长度 400~600bp)。相比之下, Oligo 可以自动搜索和评价功能, 在操作简便性上具有优势, 类似自动挡汽车。Primer 软件和 Oligo 相似, 但更简单。Snapgene 类似手动挡汽车, 直接按住鼠标左键在序列上划动, 可以自由选择引物长度和结合位置。同时, Snapgene 支持查看基因型差异位点。引物初步选定后, 需要进行引物不良结构检查, 并检查上下游引物 Tm 值差值。若不符合一般准则要求, 需重新选定引物。

4) 特异性验证：将设计的引物序列提交至 NCBI BLAST (https://www.ncbi.nlm.nih.gov/tools/primer-blast/index.cgi?LINK_LOC=BlastHome) 功能, 结果应只有 HBV。如图所示, BLAST 界面 Template 部分应该只有点, 如果出现碱基, 表示该位置跟模板不匹配, 可能导致非特异性扩增。如果碱基出现在 3' 端, 可能导致 PCR 失败。

Forward primer:

Reverse primer:

Or, upload FASTA file: 未选择任何文件

Primer Parameters

Use my own forward primer (5'→3' on plus strand):

Use my own reverse primer (5'→3' on minus strand):

PCR product size: Min Max

of primers to return:

Primer melting temperatures (T_m): Min Opt Max Max T_m difference

Primer Location Preference: ☐ Prefer primers at 3' side of the template

Exon/intron selection

A refseq mRNA sequence as PCR template input is required for options in the section

Exon junction span:

Exon junction match: Min 5' match Min 3' match Max 3' match

Intron inclusion: ☐ Primer pair must be separated by at least one intron on the corresponding genomic DNA

Intron length range: Min Max

Note: Parameter values that differ from the default are highlighted in yellow

Primer Pair Specificity Checking Parameters

Specificity check: ☒ Enable search for primer pairs specific to the intended PCR template

Search mode:

Database:

Exclusion: ☐ Exclude predicted Refseq transcripts (accession with XM, XR prefix) ☐ Exclude uncultured/environmental sample sequences

Organism:

Enter an organism name (or organism group name such as enterobacteriaceae, rodents), taxonomy id or select from the suggestion list as you type

Primer specificity stringency: Primer must have at least total mismatches to unintended targets, including at least mismatches within the last bps at the 3' end. Ignore targets that have or more mismatches to the primer.

Max target amplicon size:

Allow splice variants: ☐ Allow primer to amplify mRNA splice variants (requires refseq mRNA sequence as PCR template input)

☒ Show results in a new window ☒ Use new graphic view

Primer-BLAST

JOB ID: JC7SHN4905XQq7au287ymKHV466MevizQ

Input PCR template: none

Specificity of primers: Target templates were found in selected database: Core nucleotide BLAST database

Other reports: [Search Summary](#)

Detailed primer reports

Download primer pairs

Primer pair 1	Sequence (5'→3')	Length	T _m	GC%	Self complementarity	Self 3' complement
Forward primer	ATGCAACTTTTCAACTCTGCCTAATCATC	30	64.31	40.00	4.00	1.00
Reverse primer	CTAACAATGAGATTCCGAGATTGAGATCT	30	62.58	40.00	6.00	6.00

Products on target templates

AM053337.1 Hepatitis B virus isolate 129-CY92-12 core protein gene, complete cds

product length = 639

Forward primer 1 ATGCAACTTTTCAACTCTGCCTAATCATC 30

Template 103 192

Reverse primer 1 CTAACAATGAGATTCCGAGATTGAGATCT 30

Template 801 772

nc_009856.1 Hepatitis B virus isolate SHB1CA, complete genome

product length = 639

Forward primer 1 ATGCAACTTTTCAACTCTGCCTAATCATC 30

Template 1014 1043

Reverse primer 1 CTAACAATGAGATTCCGAGATTGAGATCT 30

Template 2452 2423

AB267397.1 Hepatitis B virus DNA, complete genome, isolate LAMB PL 6, derived from a lamivudine-resistant patient

product length = 639

Forward primer 1 ATGCAACTTTTCAACTCTGCCTAATCATC 30

Template 1014 1043

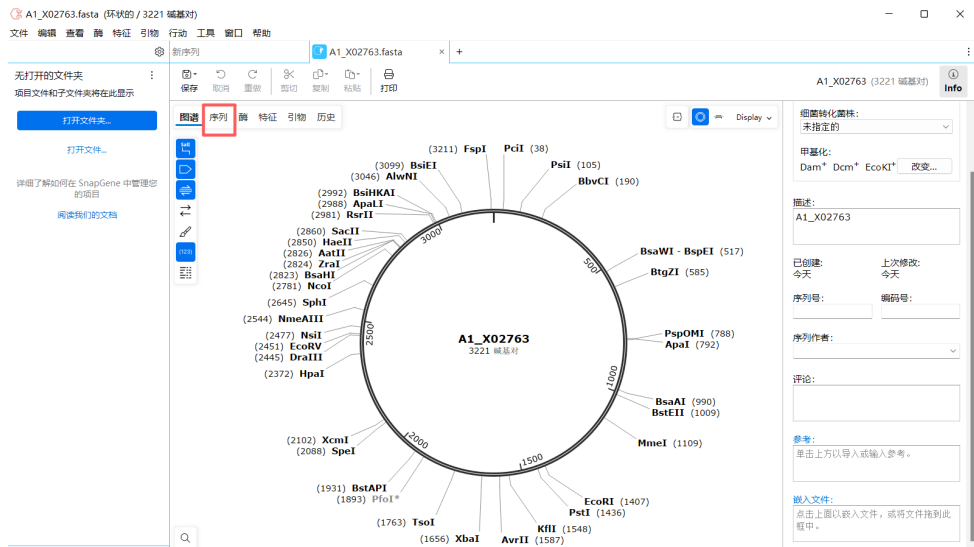
Reverse primer 1 CTAACAATGAGATTCCGAGATTGAGATCT 30

Template 2452 2423

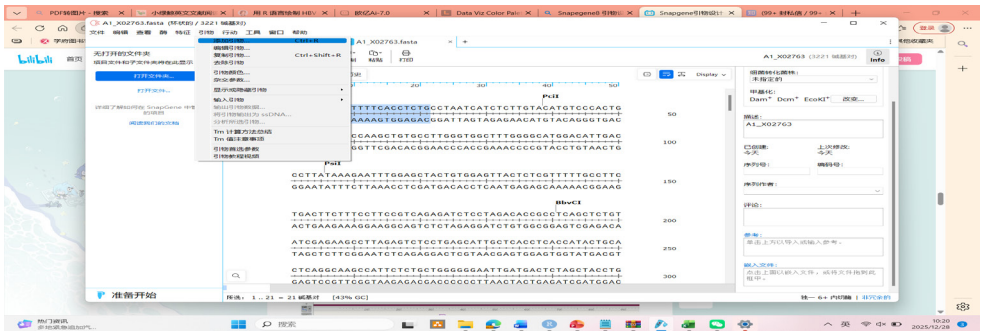
5) 引物合成: 选择 PAGE 纯化级引物 (纯度 $\geq 95\%$, 适配后续测序), 合成后用无酶水溶解, 稀释至 $10 \mu\text{mol/L}$, -20°C 保存。

5.1.4 Snapgene.8 软件引物设计

1) 在文件导入序列后，点击如图框出的“序列”按钮。在工具栏“工具”选项选择“Align Multiple Sequences，与参考序列比对，显示基因型别差异位点。



2) 分别选择顶部链、底部链作为正向、反向引物，点击“输入模板”保存引物。勾选工具栏双向箭头按钮“Show translations and ORFs”检查引物没有覆盖到“*”标记的终止子。



3) 按住“Ctrl”键点击两个引物，可以选中引物及其扩增区域，此时“分析所选引物”转为可用状态。点击可以使用 Snapgene 网页工具分析引物性质。

ORDER MENU You are using the IDT China portal.

OligoAnalyzer

Sequence

5' MOD INTERNAL 3' MOD MIXED BASES

CATGCAACTTTTCACCTCTG

Bases 21

CLEAR SEQUENCE

Try the new batch mode here

Parameter sets

qPCR (Default)

Target type

DNA

Oligo Conc

0.2 μ M

Na⁺ Conc

50 mM

Mg²⁺ Conc

3 mM

dNTPs Conc

0.8 mM

ANALYZE

HAIRPIN

SELF-DIMER

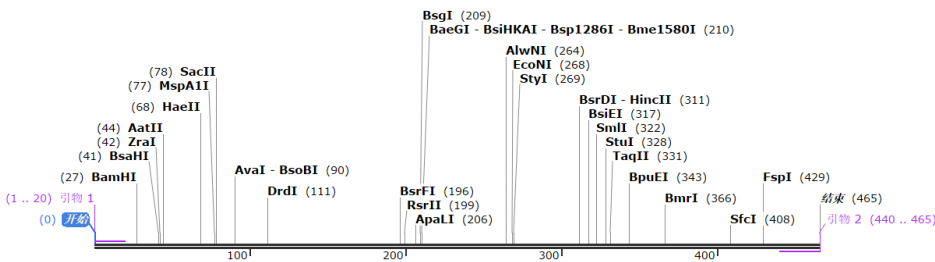
HETERO-DIMER

NCBI BLAST

TM MISMATCH

ADD TO ORDER

4) 点击“行动”-“聚合酶链式反应(PCR)”，可以显示扩增区域包含的酶位点。



5.1.5 Oligo.7 软件引物设计

Search Parameters

Search Ranges

Parameters

Sequence

Constraints

Scores

General

Constraints

More Constraints

Primer Length:

21 $\pm$ 2 nt

Acceptable 3'-Dimer Δ G:

-2.5 kcal/mol

Strongest Dimer Overall Δ G:

-7.5 kcal/mol

Min. Acceptable 3'-Loop Δ G:

1.2 kcal/mol

Min. Acceptable Loop Δ G:

-0.3 kcal/mol

3'-terminal Stability Range:

-7.2 $\pm$ 0.8 kcal/mol

5'-terminal Stability Range:

-8.5 $\pm$ 1.1 kcal/mol

Internal 12-mer Stability (siRNA):

-22.0 $\pm$ 2.0 kcal/mol

Min. GC Clamp Stability:

-7.7 kcal/mol

Primer Tm Range:

58.0 $\pm$ 2.0 $^{\circ}$ C

Tm of Probe - Tm of Primer:

5.0 $\pm$ 1.0 $^{\circ}$ C

Tm of Loop - Tm of Primers:

8.5 $\pm$ 1.5 $^{\circ}$ C

Oligonucleotide Tm Limits: 45.3 to 74.7 $^{\circ}$ C

Search Method: Nested Primers

Defaults

Cancel

OK

Sequence file: NewSequence.seq [1 to 590]

Search method: Nested Primers

Positive strand search range:

1 to 590

Negative strand search range:

1 to 590

☐ Find products in checked region(s) only

PCR product length:

250 to 500

☐ Choose sequencing primers every: 500 nt $\pm$ 50

☐ Choose hybridization probes every: 1000 bp $\pm$ 50

☐ No overlapping primers/probes

☒ Allow only one overlapping (external) primer

☐ No multiple products in one sequence region

☐ Cover the entire search area with:

☒ Overlapping products

☐ Non-overlapping products with max. 100 bp gaps

Perform False Priming Sites or Homology Searches:

☐ Within the search ranges only

☒ On the entire sequence

Check Region(s)

Defaults

Cancel

OK

• 候选引物

#	Forward Primer	Reverse Primer	Upper Oligo	Lower Oligo	Score	Product Length	Opt. T _a	%GC
	2	3	4	5	1			
1	186	479	221	456	802	315	57.2	53.3
2	14	479	40	456	767	488	59.2	56.8

• ΔG 绝对值应 ≤ 9

NewSequence:186F19				NewSequence:479R22			
5' TTGCTTTCTCTGCCGTTC 3'				5' GTGAACAGACCAATTTATGCCT 3'			
Length: 19-mer				Length: 22-mer			
Score: 890 points				Score: 876 points			
5' Position: 186				3' Position: 479			
T _m Δ _m : 57.0 57.6 °C				T _m Δ _m : 56.1 55.9 °C			
ΔG/Δg (25 °C): -30.3 -29.3 kcal/mol				ΔG/Δg (25 °C): -32.5 -31.2 kcal/mol			
ΔS/Δs: -427.6 -400.7 cal/*K * mol				ΔS/Δs: -485.6 -458.2 cal/*K * mol			
ΔH/Δh: -157.8 -148.8 kcal/mol				ΔH/Δh: -177.3 -167.8 kcal/mol			
3'ΔG: -6.7 kcal/mol				3'ΔG: -7.9 kcal/mol			
Degeneracy: 1				Degeneracy: 1			
P.E.#: 492/492				P.E.#: 458/458			
1/E: 6.52 nmol/A ₂₆₀				1/E: 4.66 nmol/A ₂₆₀			
37.6 μg/A ₂₆₀				31.7 μg/A ₂₆₀			

• 二聚体检验：为了防止引物之间相互形成二聚体，点击“Duplex Formation”分析“Forward Primer”和“Reverse Primer”，确保 $\Delta G \leq 4.5\text{kcal/mol}$ ，hydrogen bonds ≤ 3 ，若是巢式 PCR，也需查看“Upper Oligo”和“Lower Oligo”。如图所示为 Forward and reversed primer，很难同时避免氢键数大于 3，导致易形成二聚体。但是看自由能，可知氢键形成难度很大，二聚体形成相对不易。

Forward Primer Duplexes	Reverse Primer Duplexes
File: NewSequence.seq Forward Primer NewSequence:186F19	File: NewSequence.seq Reverse Primer NewSequence:479R22
Free 3'-end	
The most stable dimer overall: # of hydrogen bonds = 4; $\Delta G = -1.9 \text{ kcal/mol}$	The most stable 3'-dimer: # of hydrogen bonds = 2; $\Delta G = 0.1 \text{ kcal/mol}$
5' TTGCTTTCTCTGCCGTTC 3'	5' GTGAACAGACCAATTTATGCCT 3'
3' CCTTGCCGTCTCTCTTCCTT 5'	3' TCCGTATTTAACCAACAAGTG 5'
	The most stable dimer overall: # of hydrogen bonds = 4; $\Delta G = -1.9 \text{ kcal/mol}$
	5' GTGAACAGACCAATTTATGCCT 3'
	3' TCCGTATTTAACCAACAAGTG 5'

• 发卡结构 (hairpin steam) 检验

Forward Primer Hairpin Stems	Reverse Primer Hairpin Stems
File: NewSequence.seq Forward Primer NewSequence:186F19	File: NewSequence.seq Reverse Primer NewSequence:479R22
No stems found.	No stems found.

• 碱基组成

b) 对阳性样本的 PCR 产物进行测序验证，测序结果与 HBV 靶基因序列比对，确保扩增片段为目的基因，无错配或缺失。

2) 扩增效率验证（qPCR 引物专属）：用梯度稀释的 HBV 标准品（如 $10^2 \sim 10^6$ IU/mL）进行 qPCR 扩增，绘制标准曲线，通过对标准曲线进行线性回归算得斜率（ m ），扩增效率（ E ）的计算公式为：

$$E = 10^{-\frac{1}{m}}$$

理想扩增效率为 90%~110%，若效率低于 90%，需优化引物序列或扩增条件；若效率高于 110%，可能存在非特异性扩增，需重新设计引物。如果标准曲线线性关系良好，斜率 m 将是一个负值，其绝对值和扩增效率反比，越小越好。

3) 灵敏度验证：用低浓度 HBV 样本（如 10^2 、 10^3 IU/mL）进行扩增，检测引物能否扩增出目的条带（常规 PCR）或准确定量（qPCR）；灵敏度需满足实验需求（如临床耐药监测引物灵敏度需 $\geq 10^3$ IU/mL）。

4) 稳定性验证：用不同批次的引物、不同实验人员、不同 PCR 仪进行重复扩增，检测扩增结果的一致性（条带亮度、Ct 值变异系数）；Ct 值变异系数 $\leq 5\%$ ，说明引物稳定性良好。

5.2 PCR 扩增技术在 HBV 基因研究中的应用

PCR 扩增技术是 HBV 基因研究中最基础的靶基因富集技术，常规 PCR 主要用于目的基因片段的扩增与克隆，荧光定量 PCR（qPCR）用于 HBV 病毒载量检测与基因表达分析。本节简化两种技术的核心原理，以 HBV-S 基因扩增为例阐释常规 PCR 实操流程，根据文献结论模拟数据复现 RT-PCR 鉴别突变型（等位基因）的应用。

5.2.1 核心原理

常规 PCR 基于 DNA 半保留复制原理，通过变性（ 95°C ，双链 DNA 解链）、退火（引物与靶基因互补结合，温度依引物 T_m 值调整）、延伸（ 72°C ，Taq 酶催化 dNTP 聚合形成新链）三个步骤的循环重复，实现靶基因片段的指数级扩增（ 2^n ， n 为循环数），最终获得大量同源目的片段。

荧光定量 PCR (qPCR)：在常规 PCR 体系中加入荧光基团（如 SYBR Green 染料或 TaqMan 探针），扩增过程中荧光信号随扩增产物量的增加而增强，通过实时监测荧光信号强度，实现靶基因的定量分析。其中 SYBR Green 染料非特异性结合双链 DNA，适用于简单片段定量；TaqMan 探针特异性结合靶基因序列，适用于高特异性定量（如 HBV 变异株检测）。

5.2.2 常规 PCR 实操流程（以 HBV X 基因扩增为例）

1. 实验准备：
- 试剂准备: 2X PCR 预混液(APExBIO 公司 PCR Master Mix, K1034, 含 Taq 酶、dNTPs、染料、缓冲液)、上下游引物、模板 DNA (HBV 核酸提取物)、无酶水。

•耗材准备: 无酶 PCR 管、枪头、八联排；

•设备准备: PCR 仪、桌面离心机。
2. 反应体系配制（20 μL 体系）：

试剂名称	用量（μL）	说明
PCR 预混液	10.0	提供扩增所需酶、dNTPs 及缓冲液
上游引物（10 μmol/L）	0.8	终浓度 0.4 μmol/L，依引物浓度调整
下游引物（10 μmol/L）	0.8	与上游引物浓度一致
模板 DNA	3.0	DNA 浓度 ≥ 10ng/μL，避免抑制剂带入
无酶水	6.4	补足体系至 20 μL

3. 扩增程序设置（适配 HBV X 基因部分，片段长度约 190 bp）：
- 预变性：95℃，3min（彻底解链 HBV 部分双链环状 DNA，避免因模板结构导致的扩增不充分）；

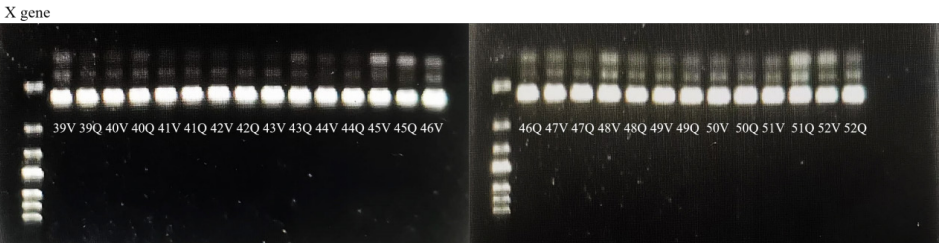
•循环阶段（35 个循环）：95℃，30s（变性）；54℃，20s（退火，依引物 Tm 值调整 ±2℃。第二轮 55℃）；72℃，1min15s(延伸, 延伸时间按 1kb/min 调整)；

•终延伸：72℃，10min（确保扩增片段完整延伸）；

•保温：4℃，∞（防止扩增产物降解）。

4. 产物验证：取 5 μL PCR 产物进行 1.5% 琼脂糖凝胶电泳（120V，25min），若出现单一清晰条带（大小与目的片段一致），说明扩增成功；若出现杂带、引物二聚体或无条带，需排查原因并优化（见 5.5 节）。如表所示为本研究所用的 HBV X 基因巢式 PCR 引物：

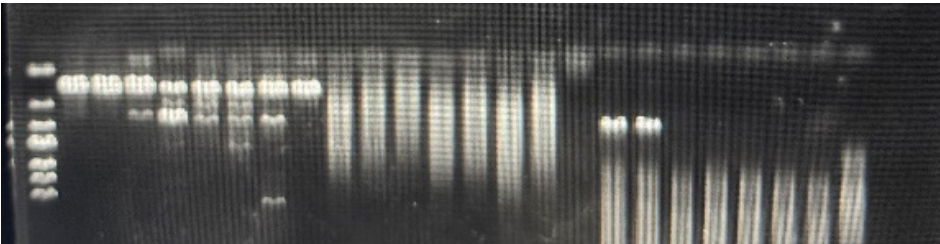
编号	位置	引物序列 5' -3'	引物长度 bp	产物长度 bp
XF1	1366-1385	TTGCTTTCTCTGCCGTTCC	19	252
XR1	1659-1637	GTGAACAGACCAATTTATGCCT	22	
XF2	1401-1422	CACCTCTCTTTACGCGGTCTC	21	193
XR2	1636-1615	AGCCTCCTAGTACAAAGACCT	21	



HBV X 基因 1% 琼脂糖凝胶电泳图像。

HBV C 基因巢式 PCR 引物：

编号	位置	引物序列 5' -3'	引物长度 bp	产物长度 bp
CF1	197-218	TTTCACCTCTGCCTAATCATC	21	341
CR1	538-517	AGGCCCATATTAACATTGACA	21	
CF2	232-252	TACTGTTCAAGCCTCCAAGC	20	306
CR2	538-517	AGGCCCATATTAACATTGACA	21	



HBV DNA 1% 琼脂糖凝胶电泳图像。C 基因：1~8；S 基因 9~24。

5.2.3 等位基因鉴别实时 RT-PCR 技术

等位基因鉴别实时 RT-PCR（Allelic Discrimination Real-Time RT-PCR）是基于实时荧光定量 PCR 技术，通过设计特异性探针区分目标基因不同等位基因（如野生型 vs 突变型）的检测方法。其核心原理是：针对基因变异位点（如单核苷酸多态性 SNP），设计分别结合野生型和突变型序列的特异性探针，利用探针的荧光信号差异，直接判定样本的基因型（纯合野生型、纯合突变型、杂合型）。

该方法适用于病毒耐药突变检测、病毒毒力相关变异筛查、病毒基因型分型、临床样本快速检测等应用。下表为常见的 RT-PCR 所需试剂。

组分	体积(μL)	说明
$2 \times \text{One-Step RT-qPCR Master Mix}$	12.5	含 Taq 酶、dNTP、缓冲液等
上游引物 (H1N1 NA-F)	0.5	终浓度 100nM
下游引物 (H1N1 NA-R)	0.5	终浓度 100nM
VIC 标记 H275 探针	0.5	终浓度 50nM
FAM 标记 Y275 探针	0.5	终浓度 50nM
RT 酶 (逆转录酶)	0.2	若用 One-Step Mix 可省略
模板 RNA	5	临床标本: 5ng; 分离株: $1 \mu\text{L}$ 稀释上清
RNase-free 水	4.3	补至 $25 \mu\text{L}$

Krishna 等人^[1]的研究是典型的通过 RT-PCR 技术区分突变型的应用。他们对甲型 H1N1 流感病毒奥司他韦耐药相关的 H275Y 突变进行了检测, 使用 VIC 标记 823 位野生型 C 碱基, FAM 标记 823 位突变型 T 碱基, 探针末端带小沟结 合物 (MGB) 提高特异性。根据其结果结论模拟数据进行复现 (RT-PCR 模拟数 据.xlsx)。下图为模拟数据表:

ID	H275RFU	Y275RFU	Genotype
S1	4800	300	Homozygous H275/H275
S2	4750	320	Homozygous H275/H275
S3	4900	280	Homozygous H275/H275
S4	4850	310	Homozygous H275/H275
S5	4700	330	Homozygous H275/H275
S6	4950	290	Homozygous H275/H275
S7	4820	305	Homozygous H275/H275
S8	4780	315	Homozygous H275/H275
S9	4920	285	Homozygous H275/H275
S10	4880	295	Homozygous H275/H275
PC_Y	3500	4900	Homozygous Y275/Y275
HE_01	4200	4000	Heterozygous H275/Y275
NTC_1	250	200	Undetermined
NTC_2	260	210	Undetermined
NTC_3	240	190	Undetermined
NTC_4	270	220	Undetermined

```
# 加载必要的包
library(ggplot2)
library(dplyr)
library(readxl)

# 1. 读取数据
raw_data <- read_excel(“保存地址 /RT-PCR 模拟数据 .xlsx” ,2)

# 2. 探针信号独立标准化
standardize_probe <- function(x, probe_type) {
  # H275 探针信号范围：文献图 X 轴 0.02~0.047
  if (probe_type == “H275” ) {
    (x - min(x, na.rm = T)) / (max(x, na.rm = T) - min(x, na.rm = T)) * (0.047 - 0.02)
+ 0.02
  }
  # Y275 探针信号范围：文献图 Y 轴 -0.015~0.035
  else if (probe_type == “Y275” ) {
    (x - min(x, na.rm = T)) / (max(x, na.rm = T) - min(x, na.rm = T)) * (0.035 -
(-0.015)) + (-0.015)
  }
}

processed_data <- raw_data %>%
  mutate(
    H275_stand = standardize_probe(H275RFU, “H275” ),
    Y275_stand = standardize_probe(Y275RFU, “Y275” )
  )

# 3. 绘图
ggplot(processed_data, aes(x = H275_stand, y = Y275_stand, color = Genotype,
shape = Genotype)) +
  geom_point(size = 2.5) +
  # 配色
```

```

scale_color_manual(values = c(
  "Homozygous H275/H275" = "red" ,
  "Homozygous Y275/Y275" = "blue" ,
  "Heterozygous H275/Y275" = "green" ,
  "Undetermined"        = "black"
)) +
scale_shape_manual(values = c(
  "Homozygous H275/H275" = 16,
  "Homozygous Y275/Y275" = 16,
  "Heterozygous H275/Y275" = 16,
  "Undetermined"        = 15
)) +
# 设置坐标轴范围
xlim(0.02, 0.047) +
ylim(-0.015, 0.035) +
# 设置绘图风格
theme_bw() +
theme(
  panel.grid = element_blank(),
  legend.position = "bottom" ,
  legend.title = element_blank()
) +
labs(
  x = "H275" ,
  y = "Y275" ,
  title = "Result of allelic discrimination real-time RT-PCR"
)

```

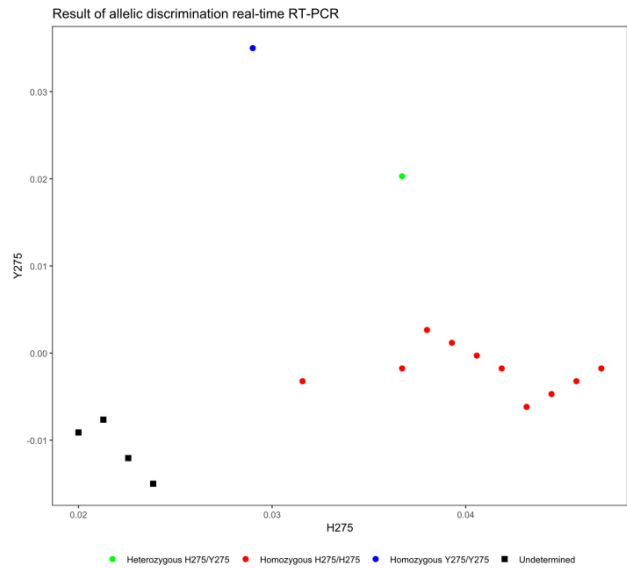


图 4.2.3. 根据 Krishna 等人^[1]的报告模拟数据绘制的突变分型检测示意图

等位基因鉴别实时 RT-PCR 技术通过双探针（H275/VIC + Y275/FAM）的荧光信号分布，一次性区分 3 种关键基因型。图中，纯合野生型（红色点，H275/H275）集中在 X 轴（H275）右侧、Y 轴（Y275）低位 → 仅 H275 探针有特异性信号，说明这些样本是对奥司他韦敏感的野生型毒株；纯合突变型（蓝色点，Y275/Y275）：Y275 探针有信号，代表奥司他韦耐药的突变型阳性对照；杂合型（绿色点，H275/Y275）：位于 X/Y 轴中高值区 → 两种探针均有信号，说明样本同时存在野生型 + 突变型毒株（可能是耐药株刚出现的混合感染阶段）；未确定型 / NTC（黑色方块）：位于 X 轴左侧、Y 轴低位 → 无特异性信号，排除阳性，是阴性对照或无效样本。该研究为临床用药和疫情防控提供了直接的实验室证据。

5.2.4 HBV 基因 PCR 扩增优化要点

1) 模板处理优化：HBV 为部分双链环状 DNA，预变性温度需提高至 95℃、时间延长至 5~10min，确保双链完全解链；若模板中抑制剂（如胆红素、蛋白质）较多，可在扩增前加入抑制剂去除试剂（如 BSA，终浓度 0.1~0.5mg/mL），或稀释模板降低抑制剂浓度。

2) 引物设计优化：针对 HBV 高变异特性，引物需设计在保守区（如 HBV

C 基因、S 基因保守区），避免因变异导致引物结合失败；引物长度控制在 18~25bp， T_m 值差异 $\leq 2^{\circ}\text{C}$ ，减少非特异性扩增。

3) 反应体系优化：HBV 基因扩增易出现非特异性条带，可适当降低引物浓度或减少循环次数、提高退火温度（一次不要提高太多，视情况提高 $0.2^{\circ}\text{C} \sim 0.8^{\circ}\text{C}$ ）；若出现片状拖带可能是引物过多，可以减少循环次数或减少引物用量。若扩增片段较长（ $>1\text{kb}$ ），需使用高保真长片段 Taq 酶，同时增加延伸时间（ $1.5 \sim 2\text{kb}/\text{min}$ ）。

4) 无条带：一般是因为引物结合区域变异，导致引物不能正常结合到模板 DNA。也可能是模板变性不彻底，或溶液中含有杂蛋白、Taq 酶抑制成分等原因。重做或更换引物。

5) 防止污染优化：HBV 扩增灵敏度高，易出现气溶胶污染，需严格分区操作（试剂配制区、样本处理区、扩增区）；每批实验设置空白对照（无酶水替代模板）、阴性对照（非 HBV 样本），监测污染情况。

本章小结

本章围绕 HBV 基因扩增与测序核心技术，结合 HBV 基因组高变异、部分双链环状的特性，系统阐述了技术原理、实操流程、专属优化要点及质量控制标准，为 HBV 基因研究（突变分析、基因分型、耐药监测等）提供了可落地的技术指引。

核心要点总结如下：

① 引物设计是 PCR 扩增特异性的核心保障，需遵循保守性、特异性、适配性及实用原则，针对 HBV 需优先选择各基因型共有的编码区保守序列，通过 NCBI BLAST 等工具验证特异性，并经实验验证扩增效率、灵敏度与稳定性；常用 Snapgene、Oligo 等软件辅助设计，可灵活选择引物结合区域并评估不良结构。

② PCR 扩增技术是 HBV 靶基因富集的基础，常规 PCR 适用于目的基因片段的扩增与克隆，荧光定量 PCR（qPCR）用于病毒载量检测与基因表达分析；针对 HBV 需优化预变性条件（ 95°C 、 $3 \sim 10\text{min}$ ）以确保双链完全解链，同时通过调整引物浓度、退火温度等参数减少非特异性扩增，强化分区操作以防控污染。等位基因鉴别实时 RT-PCR 可通过特异性双探针荧光信号差异，快速区分野生型、突变型及杂合型毒株，为耐药监测提供直接实验室证据。

③ 测序技术方面，Sanger 测序是 HBV 感染诊断的金标准，适用于已知变异验证，但在遗传多样性分析中存在局限；Illumina 测序可实现深度测序但无法覆盖长片段且易引入扩增偏差；高通量测序（HTS）技术则有效解决上述问题，拓展了 HBV 全基因组研究方向，但对软、硬件要求较高。全基因组测序（WGS）是分析 HBV 遗传多样性的关键手段，可避免重组现象导致的研究偏差。

④ 全流程质量把控是实验可靠性的关键，需覆盖样本处理、引物设计与验证、PCR 扩增、测序及数据分析各环节，通过设置对照、优化实验条件、重复验证等方式，及时排查无条带、杂带等异常情况，确保结果准确可重复。

本章内容衔接前序 HBV 筛查与 DNA 提取技术，为后续 HBV 基因功能研究、临床用药指导及疫情防控奠定了坚实技术基础。

第6章 乙肝病毒基因信息检索与数据库应用

随着基因克隆技术、聚合酶链式反应（PCR）技术、DNA 测序技术、互联网技术等发展，数据库和相关生物信息学方法工具得到更广泛的应用。很多科学期刊已要求在论文发表前必须将对应的新序列存入公共数据库。但对于特定领域，如 HBV 研究，相关的数据库发展还较为滞后。目前通用数据库和工具的发展已经转向大规模实证观察以及与特定基因类别、蛋白质、结构域等相关特征的比较。大规模的转录组学、蛋白质组学、结构基因组学和互作组学国际倡议通过提供 RNA 和蛋白质水平表达数据的交叉链接，以及许多蛋白质结构和实验验证的高通量蛋白质-蛋白质互作证据，进一步提高注释序列的可用性。但对于 HBV 相关库和应用，暂时主要还是传统的基于统计或物理化学性质的评估。部分应用已经建立数据互通，如，SWISS-MODEL 可以调用 HBV 蛋白质结构数据库进行结构预测，但受限于数据完整性不足，只能对特定的区域进行预测。

乙肝病毒（HBV）基因信息的高效、准确获取是开展应用研究的基础。公共核酸数据库与 HBV 专属数据库汇聚了全球范围内的 HBV 基因序列及关联信息，是科研与临床研究的核心数据资源。本章以“实操落地”为目标，手把手讲解主流数据库的检索流程，重点聚焦 HBV 相关数据的获取技巧，同时覆盖数据筛选验证、格式转换及引用规范等关键环节，帮助实验人员快速掌握数据库应用技能，保障数据使用的效率与科学性。需要注意的是，临床决策不应仅基于数据库、软件、网页工具等算法运行结果。

6.1 主流核酸数据库

目前，已有多个公共数据库存储和分发病毒相关数据，包括基因组序列、基因和蛋白质注释、病毒结构、生物学、病理学和分类学信息等。一级数据库、Uniport 数据库、国际核酸数据库联盟数据库（GenBank/EMBL(ENA)/DDBJ）构成核苷酸和蛋白质水平序列数据的主要存储库。此外，还有一些更聚焦病毒领域的通用数据库。如，ViralZone、ViPR 等。此外，还有国际病毒分类委员会

（International Committee on the Taxonomy of Viruses, ICTV）发布的报告被视为病毒分类的标准和参考。另有一些数据库包含临床和人群信息，可以为流行病学相关研究提供支持，如，Nhanes 数据库。瑞士生物信息学研究所维护了一个数据库汇总网站 Expasy（官网：<https://www.expasy.org/>）方便用户按需访问。数据库非常多，但从实用角度看，很多数据库并不需要去了解，因为它们的使用很不方便。相比之下，一些网页工具、软件可以实现数据库调用，并提供更贴合实际应用需求的操作界面，这些工具、软件对于工作繁忙的临床和科研人员而言更值得关注。因此，本节不赘述其他较为少用的数据库，主要以应用最广泛的 GenBank（NCBI）为例，讲解检索流程。

6.1.1 GenBank 数据库核心检索流程（HBV 基因检索为例）

GenBank 是最常用的通用数据库，由美国国家生物技术信息中心（NCBI）维护，可通过 NCBI 的 Entrez 检索系统实现多维度精准检索，具体步骤如下：

1) 进入检索系统：打开浏览器，输入 NCBI 官网地址（<https://www.ncbi.nlm.nih.gov/#!/home/contact>），在首页上方检索框的下拉菜单中选择“Nucleotide”（核苷酸数据库，即 GenBank 核心库），点击检索框准备键入。

2) 基础关键词检索：在检索框中输入 HBV 相关检索词，若要搜索全长序列，输入“Hepatitis B virus[Organism] AND complete genome[Title]”；若需特定基因（如，S 基因），可优化检索式为“Hepatitis B virus[Organism] AND S gene[Title]”。输入完成后点击“Search”执行检索。



3) 结果筛选优化：点击左列筛选条件栏，每个项目下有“Customize”提供选项或“Custom range”支持选择数值范围。如，在 Release date 下 筛选近期更新数据。

4) 结果排序：点击结果列表上方的“Sort by”下拉菜单，选择“Relevance”

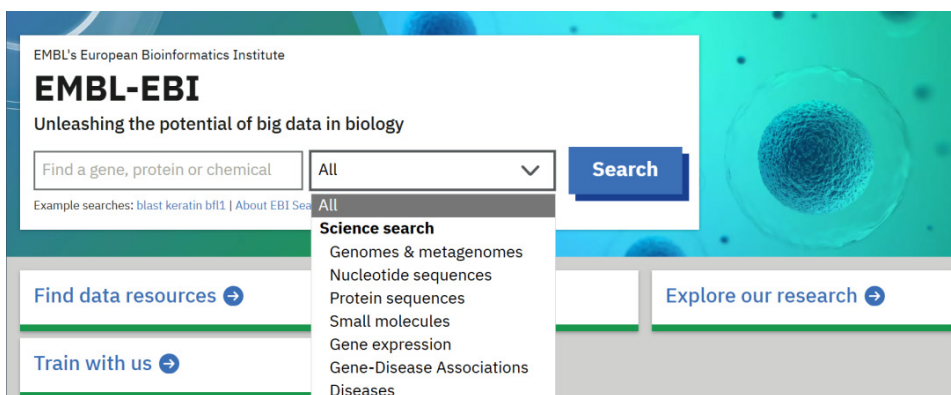
(相关性)或“Update date”(更新日期),优先查看核心或最新数据。

5) 序列详情查看: 点击检索结果中的序列标题(如“Hepatitis B virus isolate EIJK093 S protein (S) gene, partial cds”),进入序列详情页。页面核心信息包括:
① 序列基本信息 (ACCESSION: 唯一标识符,如 PP797566,可用于精准引用);
② 注释信息 (FEATURES 栏,包含基因位置、编码区、来源等); ③ 原始序列 (ORIGIN 栏,包含完整碱基序列)。

6) 序列下载: 在序列详情页,点击右上角的“Send to”下拉菜单,选择“File”,在弹出的窗口中选择下载格式(推荐 FASTA 格式,通用性最强),点击“Create File”完成下载;若需批量下载多条序列,可在检索结果列表中勾选目标序列,再通过“Send to”批量导出。

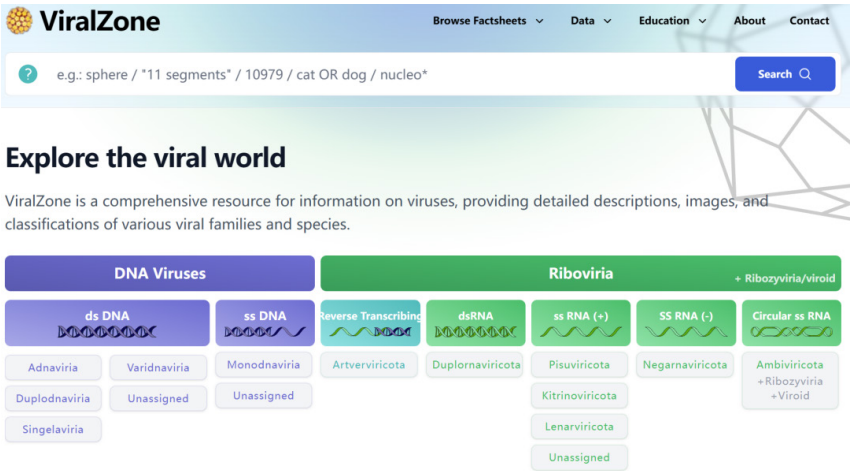
6.1.2 ENA (EMBL) 数据库

ENA 数据库由欧洲生物信息研究所 (EBI) 维护,官网地址 (<https://www.ebi.ac.uk/>), 核心检索流程与 GenBank 类似,同样是点击 all 下拉菜单选择 Nucleotide sequence。ENA 数据库支持原始测序数据的上传和下载,提供多种在线分析工具,和 GenBank (NCBI) 通用序列唯一标识符 (检索式不通用)。



6.1.3 Uniport 项目 ViralZone 数据库

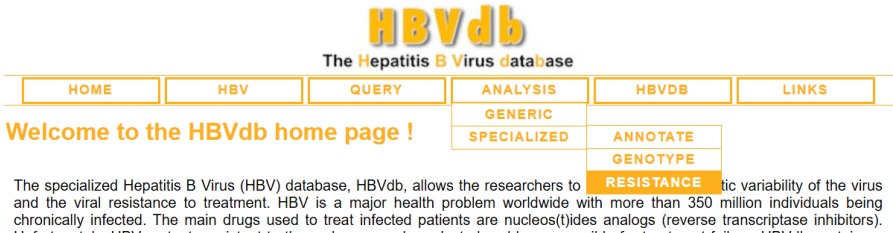
SIB 维护的 ViralZone (官网: <https://viralzone.expasy.org/>) 数据库侧重知识库汇总。



6.2 HBV 特化数据库

6.2.1 HBVdb 数据库

乙型肝炎病毒数据库（Hepatitis B virus database, HBVdb）是由里昂大学微生物研究所、里昂生物信息学中心（PBIL）和罗纳 - 阿尔卑斯生物信息学中心（PRABI）开发的，专注于 HBV 变异与耐药研究的数据库（官网：<https://hbvdb.lyon.inserm.fr/HBVdb/HBVdbIndex>）。HBVdb 包括 3 个模块。第一个模块是一个知识数据库，通过自动化系统收集公开可用的序列，并使用一致的命名法对其进行注释。在获得适当的结构模板后，将计算乙型肝炎病毒（HBV）蛋白质的三维模型。第二个模块是 HBV 序列和临床数据关联。第三个模块是一个分析模块，包含一些通用的基础工具，主要是序列比对和三维结构模拟。HBVdb 的核心优势是整合了大量临床耐药数据，可根据用户提供的序列直接检索是否含有耐药突变位点，依次点击：ANALYSIS→SPECIALIZED→RESISTANCE



Most similar reference genome used: [AB219428](#) (genotype B)

Query features:

CDS	From-To	Reference	CDS alignment
PreS1	<1..>630	<1..>630	view
PreS2	<1..>630	<1..>630	view
S	155..>630	155..>630	view
P	<1..>630	<1..>630	view
SP	<489..>630	<489..>630	view

Query provisional genotype: **B**

Query-Reference alignment: [view](#)

Query as HBVdb text entry: [view](#)

Drug resistance status: **Sensitive**

Query sequence with provisional genotype by position:

```

      10      20      30      40      50      60      70      80
.....|.....|.....|.....|.....|.....|.....|.....|
ctccaccactttccacaaactottcaagatcccagagtcagggcctgtactttcctgtgtgggtccagttcaggaa
BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
```

6.2.2 HIV Drug Resistance Database

斯坦福艾滋病病毒耐药性数据库 (HIV Drug Resistance Database, HIVDB, 官网: <https://hivdb.stanford.edu/HBV/HBVseq/development/HBVseq.html>) 用于 监测抗药性耐药 (ADR) 和三重耐药 (TDR)、开发新型抗逆转录病毒 (ARV) 药物, 以及帮助艾滋病病毒 (HIV) 护理提供者管理 HIV 耐药患者 (HIVDR)。该数 据库包含一个免费且公开可用的基因型耐药性解读系统, 其中也包含了 HBV 耐 药突变信息。



Stanford University

HIV DRUG RESISTANCE DATABASE

A curated public database to represent, store and analyze HIV drug resistance data.

HOME

GENOTYPE-RX

GENOTYPE-PHENO

GENOTYPE-CLINICAL

HIVDB PROGRAM

VISTAS PROGRAM

ABOUT HIVDB

SUPPORT HIVDB!

HBVseq

Search Engine for Query Sequences

HBVseq accepts user-submitted HBV RT sequences, determines their genotypes, and compares them to a genotype consensus reference sequence. The differences between the submitted sequence and the consensus reference sequence are used as query parameters for interrogating a local HBV drug resistance database (HBVrt DB) to retrieve the prevalence of each mutation according to genotype and treatment. HBVrt DB was constructed by annotating the body of publicly available HBV RT sequences.

ANALYSIS

MUTATION LIST

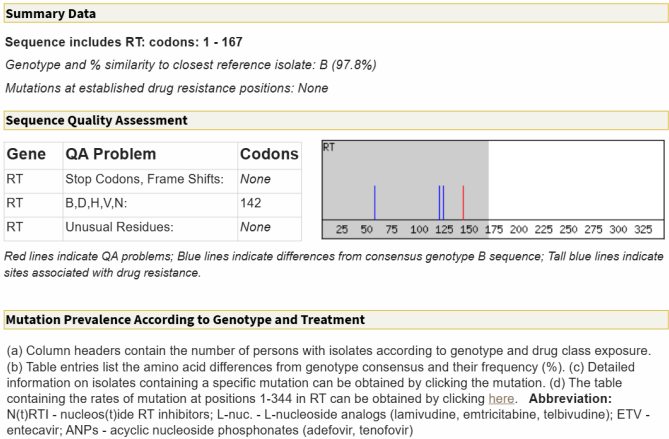
Enter Protease & RT Mutations

OR

ANALYSIS

SEQUENCES

Enter Complete Sequences



Pos	NA	AA	N(t)RTI Naive Persons (by Genotype)										N(t)RTI Treated Persons (by Treatment)			
			A	B	C	D	E	F	G	H	I	pooled	L-Nuc. and/or ETV	ANPs	L-Nuc. and/or ETV + ANPs	
			3938	1943	3671	4226	1340	664	40	77	89	15943	824	157	327	
53	AAM	KN	V_{8.3} S_{7.4} I_{6.3} T_{2.6} P_{1.1} A_{1.0} N_{0.8}	N_{2.2} S_{1.9} H_{1.1} T_{1.0} K_{0.8} G_{0.1} P_{0.1}	S_{6.4} T_{0.2} R_{0.1} G_{0.0} V_{0.0} G_{0.1} P_{0.0}	N_{7.9} K_{2.8} T_{0.2} F_{0.6} T_{0.3} A_{0.1} D_{0.1}	S_{2.3} N_{1.0} S_{1.8} T_{0.2} G_{0.2} A_{0.1} D_{0.1}	T_{35.6} S_{1.8} P_{0.6} K_{0.3}	S	T_{1.4} 						

6.2.3 Hepatitis B Virus Repository

Hepatitis B Virus Repository (HBVR) 是全球首个 HBV 多组学数据库（官网：<https://www.hbvr-nsut.in/>（网站已关闭或国内不能访问）），包含与乙型肝炎病毒相关的信息，特别侧重于该病毒的基因组、系统发育和治疗研究。它能够提供更复杂的分析结果，如密码子使用和上下文、CpG 岛以及系统发育分析。此外，还分别对病毒蛋白和基因组序列的糖基化模式和引物解释进行了分析。该平台预测了具有治疗重要性的新候选物，如疫苗表位、小干扰 RNA、微 RNA，以及病毒基因组序列上有用的药物和药物靶标信息。

6.3 HBV 基因研究相关实用工具

6.3.1 HBV 泄露扫描检索工具

HBV 存在一种替代翻译起始（alternative translation initiation）机制，即在 同一开放阅读框（ORF）内部的几个起始密码子 (AUG) 形成不同的激活组合，使单个 mRNA 分子可翻译不同 N 末端的同工型蛋白质（isoform proteins）^[1]。这

种机制改变了蛋白质功能和 / 或亚细胞定位, 从而以有限的基因组扩展了蛋白质组。“泄露扫描”(leaky scanning) 是这一机制的核心原理, 指翻译启动机制不是常规的从起始密码子开始的核糖体扫描模型, 而是由 AUG 上下文决定的。在这种情况下, 通常由来自 Kozak 的共识序列启动翻译。Kozak 序列即 (A/G)XXAUGG, 其中 -3 (A/G) 和 +4G 分别是有效启动的第一个和第二个最重要的碱基 (+1 是起始三联体的第一个碱基)^[2]。已有在线工具 (<https://atgpr.dbcls.jp/>) 被开发出来, 用以研究替代转录起始可能导致多少种同工型蛋白质被翻译出来, 其基本原理正是根据 Kozak 共识规则识别 cDNA 序列中起始密码子。这一工具支持直接贴入 DNA 序列, 操作简便, 分析速度快。



A program for identifying the initiation codons in cDNA sequences.

Salamov, AA., Nishikawa, T., Swindells, MB

Query Sequence : QUERY

```

1 CTCACCACT TTCACCAA CTCTTCAAGA TCCAGAGTC AGGGCCCTGT ACTTTCCTGC
61 TGGTGCTCC AGTTCAAGAA CAGTGAAGCC TCTCAGACT ACTGTCTCTG CCAATGCTGC
121 AATCTTATCG AAGACTGGGG ACCCTGTACC GACATGAGG AACATGCGAT CAGGACTCTT
181 AGAGCCCTCG CTCTGTATAC AGGGGGGTTT TTCTTGTGTG ACAAAGATCC TCGAATACCC
241 ACAGAGTCTA GACTGTGTGT GGACTTCTCT CAATTTTCTA GGGGGAANAC CCGTGTGTCT
301 TGGGCAAAAT TCCAGTCCG AAATCTCCAG TCATCCACNA ACCTGTGTCT CTCGAATTTG
361 TCGTGTTAT CCGTGATGT GTCTCGGGG TTTTATCATC TTCTTGTGCA TCGTGTGTCT
421 ATGCGTCATC TCTGTGTGTG TTCTCTGGGA CTATGAGGT ATGTGCGCG TTGTGCTCTN
481 ANTCACAGA TCANCACAA CACAGACCGG ACCATGCAAA ACCTGACAGA CTCGTCTCA
541 AGGAACCTCT ATNNTCCTT CATGTGTCTG TACAAACCT ACGAGCGGAA ACTGACCTG
601 TATTCACATC CCAATCTCTT GGGCTTTTCG
  
```

No. of ATG from 5' end	Reliability	Frame	Identity to Kozak rule A/GXXATGG	Start (bp)	Finish (bp)	ORF Length (aa)	Stop codon found?	Sequence
1	0.41	2	AXXATGG	155	630	158	No	MENTASGLLPVLLVQGFLLTKLITPQSLDSWTSUNPLGXPVCLG QNSQQTSSSHXTCPPICPGYBDMCLRRFTIFXCTLLLLFLVLLDY NQMLPVCPXPGSXTTSTGPCKTCTTPAGTGSXPSCCCTKPTDGNCTCI PIPSWAF
3	0.14	1	GXXATGc	421	630	70	No	MPHLLVGSGLXRYVARLSXSRITXNNQHRDMQNLHDCSRNLYSLMLL VXTYGRKLHLYSHPTILGFR
2	0.06	2	tXXATGc	377	630	84	No	MCLRRFTIFXCTLLLLFLVLLVYXQMLPVCPXPGSXTTSTGPCKTC TTPAGTGSXPSCCCTKPTDGNCTCPIPSWAF
5	0.05	1	AXXATGc	514	630	39	No	MQNLHDCSRNLYSLMLLKYTYGRKLHLYSHPTILGFR
4	0.04	2	GXXATGc	461	630	56	No	MLPVCPXPGSXTTSTGPCKTCTTPAGTGSXPSCCCTKPTDGNCTCPI PSWAF

6.3.2 P 基因区耐药性预测

Genafor(官网: <https://hbv.geno2pheno.org/>)是一个功能丰富的医学检索工具, 提供了 HBV、HCV、HEV 等病毒序列 - 临床信息关联数据库的调用方式。

6.3.3 蛋白质三维结构预测

SWISS-MODEL (官网 <https://swissmodel.expasy.org/interactive>) 支持基于氨

核酸序列的蛋白质结构预测，可同时输入多条序列。序列需符合 FASTA 格式，即包含“>序列名称”。可导出为 PDB 文件。

max planck institute
informatik



[Home](#)
[About Genator](#)
[Contact](#)
[Membership and Donations](#)
Services
[genozphero \[reistance\]](#)
[genozphero \[consortor\]](#)
[genozphero \[HIV2\]](#)
[genozphero \[ISG\]](#)
[genozphero \[relaxase\]](#)
[genozphero \[gag\]](#)
[genozphero \[pro\]](#)
[genozphero \[rev\]](#)
[genozphero \[pol\]](#)
[genozphero \[gag-pol\]](#)
[genozphero \[gag-pro\]](#)
[genozphero \[rev\]](#)
[genozphero \[RNAse\]](#)
[genozphero \[hdgmg\]](#)
[Service access statistics](#)
[Related Sites](#)
[Events](#)
[News](#)
[Links](#)

Genozphero [HIV] 2.0

PAGE: Input **Results** Resistance Escape References Contact

Predictions are for research use only, we don't guarantee for any prediction!

On submitting below a HIV pol-gene's surface-gene DNA sequence you will obtain a sequence alignment to the HBV genotype D consensus sequence or alternatively to a consensus sequence of the respective genotype, a list of mutations according to the reverse transcriptase domain of the pol-gene and the S/HB protein of the surface-gene and predictions of phenotypic resistance of the respective virus to live antiretroviral drugs. Please note that for reliable predictions the sequences must contain a substantial part of the RT domain.

No clinical decision should be based only on the result of the used algorithm.

1. Identifier

Do not use patient names!

upload up to 4 files (plain sequence or FASTA format):

基因文件 | 未选择文件

基因文件 | 未选择文件

基因文件 | 未选择文件

基因文件 | 未选择文件

或粘贴到:

2. HBV sequence containing the RT domain:

3. Sequence ambiguities: ☐ use genotype D consensus sequence

4. Option: Alignment width:

5. Action: Align and Predict

6. Regions:

- ☒ RT domain, SHB protein
- ☐ Phe Core region
- ☐ LHB protein
- ☐ MHB protein
- ☐ SHB_nt protein at NT level
- ☐ RT_nt protein at NT level
- ☐ Core protein

7. CSV: ☐ direct csv download

9. Parameters:

☐ overwrite default parameters	gap open cost: 6
	gap extension cost: 3

SWISS-MODEL[®]

Modelling

Repository

Tools

Documentation

Log in

Create Account

Start a New Modelling Project 

Target Sequence(s):
(Format must be FASTA,
Clustal,
plain string, or a valid
UniProtKB.AC)

Paste your target sequence(s) or UniProtKB.AC here

 Upload Target Sequence File...

 Validate

Project Title:
Optional

Email:
Optional

Search For Templates

Build Model

Supported Inputs 

Sequence(s) 

Target-Template Alignment 

User Template 

DeepView Project 

By using the SWISS-MODEL server, you agree to comply with the following [terms of use](#) and to cite the corresponding [articles](#).

6.4 数据库数据格式转换与整理

不同数据库下载的 HBV 序列格式不同（如 FASTA、GenBank FlatFile、EMBL 格式），后续分析工具（如序列比对和进化树构建软件 MEGA、R 包）对数据格式有特定要求，需进行格式转换；同时，原始数据需规范化整理，便于批量分析。本节讲解常用格式及转换、整理方法。

6.4.1 常用序列数据格式及适用场景

格式类型	特点	适用场景
FASTA 格式	简洁，首行以 “>” 开头为序列标识，后续为碱基 / 氨基酸序列，无多余注释	序列比对、进化树构建、PCR 引物设计等绝大多数分析工具，应用广泛
GenBank FlatFile 格式	包含完整注释信息（序列来源、基因位置、编码区等），结构规范，便于计算机读取	注释信息提取、基因结构分析
EMBL 格式	与 GenBank FlatFile 类似，注释信息侧重功能域，字段命名略有差异	跨数据库数据整合、功能域分析

6.4.2 格式转换方法

6.4.2.1 网页工具转换（Readseq）

Readseq 由 MAFFT 维护，打开浏览器，输入官网地址（<https://mafft.cbrc.jp/alignment/server/cgi-bin/readseq.txt>）。

- 1) 上传数据：点击 “Browse”，选择需要转换的原始文件（如 GenBank 格式文件）；若序列较短，可直接粘贴 FASTA/GenBank 格式文本。
- 2) 选择转换格式：在 “Output Format” 下拉菜单中选择目标格式（如 “FASTA”）。
- 3) 执行转换：点击 “Submit”，等待几秒后，页面会显示转换后的序列；点击 “Download” 保存转换后的文件。

Readseq -- biosequence conversion tool

Sequence data

Upload sequence file: or paste data or URL in box below

See [here](#) for help.

Options

Output sequence format:

Return biosequence data:

- ☐ Download to file
- ☒ View in browser

Change sequence case to:

- ☒ No change
- ☐ lower
- ☐ UPPER

☐ Remove gap symbols:

☐ Calculate checksum of sequences

Select ☒ all, or ☐ sequences by number:

☐ Translate bases (list as from-base:to-base pairs)

Feature selection

Features to select/remove:

- ☒ No selection
- ☐ Extract sequence of selected features
- ☐ Remove sequence of selected features

Subrange

of sequence feature locations to extract. E.g.,
-1000..0 is 1000 bases upstream to start of feature;
end..end+100 is from end to 100 bases downstream.

These selections apply only when
input data includes parsed feature tables.

Readseq by D.G. Gilbert, 2.1.30 (12-May-2010)
Software at <http://iubio.bio.indiana.edu/soft/molbio/readseq/java/>

6.4.2.2 软件转换（BioEdit，支持批量转换）

1) 从官网（<https://thalljiscience.github.io/>）下载软件。启动 BioEdit，导入原始序列文件（支持批量导入多个文件）。

2) 选择转换功能：点击“File”→“Save As”，在弹出的窗口中选择目标格式（如“FASTA Format”）。

3) 批量保存：若导入多个序列，可点击“File”→“Batch Save”，选择目标格式和保存路径，实现批量转换与保存，提升效率。

6.4.2.3 软件转换（Rstudio，支持批量转换）

在 R 软件进行基因分析前需要检查数据格式，确保符合函数包的要求。

将 FASTA 格式转为 genlight 格式

```
dnaB<-fasta2genlight(“保存地址/aligned1_B_Sgene.fas”,chunkSize =  
10,parallel = F)
```

```
print(dnaB)
```

将 FASTA 格式转为 DNABin 格式

```
dnabinB<-fasta2DNABin(“保存地址/aligned1_B_Sgene.fas”)
```

```
str(dnabinB)
```

将 FASTA 格式序列文件定义为比对文件

```
aliB<-read.alignment(“保存地址/aligned1_B_Sgene.fas”,format = “fasta”)
```

```
print(aliB)
```

5.4.3 数据整理规范（便于后续分析）

• 序列标识规范化：FASTA 格式的序列标识行（首行）需包含核心信息，推荐格式：>AccessionID_Genotype_Source_Date（如 >NC_003977.2_GenotypeA_HumanBlood_2020），避免默认的冗长标识，便于后续数据追溯。这是数据库格式，如果要进行绘图，最好简化以使图像美观，如只保留核心的 ID 号 NC3977 作为分析数据库中的唯一索引代码。

• 批量数据分类：将不同基因型、不同研究目的的序列按文件夹分类存放（如“HBV_GenotypeB_CompleteGenome”“HBV_GenotypeC_SGene”），避免数据混乱。在一些分析项目中，如核苷酸多样性分析、组间相关性分析等，也要求预先建立子集。

- 一些序列需要拆分，自然也有合并的需求。建立 .txt 文本文档，输入 copy * 要合并的文件格式 新文件名称及格式，如 “copy *.fasta RefSeqSet.fasta”，然后重命名为 “combine.bat” 文件，点击按管理员运行，即可合并该文件夹内所有 .fasta 格式文件为一个名为 RefSeqSet 的 .fasta 文件。

- 在一些分析项目中，过多不必要的变量会影响分析思路 and 文件加载速度。此外，不同项目可能需要进行不同的数据预处理，这就需要建立与分析项目配套的 Excel 信息表：从备份表拷贝出去，检查唯一索引代码和变量名和原始数据表一致。

6.5 数据引用规范与版权注意

公共数据库的 HBV 基因数据受版权保护，合理引用与规范使用是科研诚信的基本要求，同时可保障研究成果的可追溯性。本节明确数据引用的核心规范、版权注意事项及常见错误规避方法。

6.5.1 数据引用核心规范

1) 优先引用原始文献：若下载的 HBV 序列有对应的发表文献（可通过数据库详情页的 “Publication” 字段查看），需优先引用该原始文献，而非直接引用数据库；引用格式遵循学术期刊要求（如 APA 格式、GB/T 7714 格式）。

2) 数据库引用规范：

a) 若序列无对应原始文献（仅数据库收录），需引用数据库及序列的 Accession ID，以 GenBank 序列引用为例：Hepatitis B virus isolate XXXX, complete genome. GenBank: NC_003977.2. National Center for Biotechnology Information (NCBI), Bethesda, MD, USA.

b) HBV 专属数据库引用：Hepatitis B virus genotype B sequence. HBVdb Database: HBV000123. <https://hbvdb.lyon.inserm.fr/HBVdb/HBVdbIndex>, accessed 2025-XX-XX.

c) 批量数据引用：若使用多个数据库序列（如 10 条以上），可在正文或致谢中说明数据来源及 Accession ID 列表，示例：“本研究使用的 HBV 序列均来自 GenBank 数据库，Accession ID 分别为：NC_003977.2、AB033559.1、...”。

6.5.2 版权注意事项

- 1) 非商业使用原则：公共数据库的 HBV 数据仅供科研、教育、临床诊断等非商业用途，禁止用于商业盈利活动（如未经授权的试剂盒开发、数据售卖）。
- 2) 数据修改与再发布：若对下载的序列进行修改（如拼接、注释补充），再发布时需注明原始数据来源（Accession ID）及修改内容，避免侵权；部分数据库要求再发布前需获得授权，需提前查看数据库的“Terms of Use”。
- 3) 隐私保护：临床来源的 HBV 序列（如患者样本分离的序列）可能包含隐私信息，数据库已进行匿名化处理，使用时不得尝试追溯患者身份，严格遵守医学伦理要求。

本章小结

本章围绕 HBV 基因信息检索与数据库应用，以实操为核心，系统讲解了主流公共核酸数据库（GenBank/ENA）和 HBV 特化数据库（HBVdb）的检索流程，细化了序列筛选验证、格式转换与数据整理的具体方法，明确了数据引用规范与版权注意事项，核心要点总结如下：① 高效获取 HBV 数据需根据研究目的选择数据库，公共数据库适合广泛检索，特化数据库适合精准获取耐药、变异相关数据；② 检索时通过组合关键词、限定条件提升效率，下载后需从完整性、准确性、相关性维度筛选验证，避免低质量数据影响分析结果；③ 数据格式转换可通过在线工具（READSEQ）或软件（BioEdit、Rstudio）实现，规范化整理数据（标识统一、分类存放、配套表格）便于后续批量分析；④ 引用数据需优先引用原始文献，明确标注 Accession ID，严格遵守版权规定，避免科研诚信问题。此外，介绍了多种实用的 HBV 基因分析软件、网页工具。本章内容覆盖 HBV 数据获取的全流程，提供了可直接落地的操作指引，帮助实验人员快速掌握数据库应用技能，为后续基因分析工作奠定坚实的数据基础。

第7章 基因数据质控与预处理

原始基因测序数据（一代测序峰图数据、二代测序短读长数据等）中普遍存在低质量碱基、接头残留、重复序列等干扰信息，直接影响后续突变分析、基因分型、进化树构建等结果的准确性。这一部分主要是测序公司完成，但测序结果不一定尽善尽美，个别位点重叠峰、信号弱，以及测序始、末的杂信号等因素，可能导致一些碱基不能正确识别。此外，错配等偶然因素也可能导致结果完全不能匹配但PCR产物长度和预期相符。在这些情况下，测序结果报告的都是“正常”。本章侧重测序数据到手之后的处理，主要包括序列位置核对、错误碱基和空缺值处理，介绍基因序列预处理的基本经验，衔接测序实验与下游的分析环节。

7.1 测序产物位置核对

一般而言PCR结果和引物设计时预期的结果是一样的，但也会有特殊情况。在双向测通的要求下，有些特殊情况会导致产物长度和预期长度接近，但产物位置和预期位置不同，却也不算失败。根据作者的测序经历，发现存在正向引物结合不好，但反向引物可以；或正向引物扩增区存在终止子等，导致正向和反向引物长度不同。也可能存在完全测错，但看起来正常。开始长度不一也是可能存在的情况，而引物设计时侧重于性能，也可能导致产物不会刚好从完整密码子开始。因此，测序数据到手后需要对其位置进行核对，确认其对应的氨基酸序列开始位置，作为后续编号的基础。

参考序列的氨基酸序列贴到Word软件，把MEGA软件氨基酸序列开头几个氨基酸输入查找框确定氨基酸序列正确，若不正确，在MEGA退回到核苷酸界面，删除第一列碱基后转氨基酸界面再试。此外，符合通用标准的核苷酸或氨基酸编号规则是保证研究结果可信度的关键。氨基酸命名方法在不同文献也有差异，目前一般标准是根据成熟蛋白质编码，如S1M、E2K/R/D、G7D/R。不能直接根据DNA翻译的氨基酸序列编码为preS11M, S1I, S3S, 因为要考虑实际的起始位置。

下表为 HBV 氨基酸序列分区定位方法及 Excel 公式。假设 HBV-S 基因氨基酸序列存在 B3 格：

S genomic region	Subregion	Equation	Value
MGGWSSKPRQGMGTNLSVPNPLGF FPDHQLDPAFGANSNNPDWDFNPN KDHWPEAIKVGAGDFGPGFTPPHG GLLGWSPQAQGILTTVPAAPPPVST NRQSGRQPTPISPPLRDSHPQAMQ WNSTTFHQALLDPRVRGLYFPAGG SSSGTVNVPVTTVSPISSIFSRTGDPA PNMESTTSGFLGPLLVLQAGFFLLT RILTIPQSLDSWWTSLNFLGGAPTC PGQNSQSPTSNHSPTSCPPICPGYR WMCLRRFIIFLFILLCLIFLLVLLD YQGMLPVCPLLPGTSTTSTGPCKTCTIP	preS1	"=LEFT(B3,108)"	MGGWSSKPRQGM
	preS2	"=MID(B3,109,163)"	SPPLRDSHPQAMQV
	complete S	"=LEN(B3)"	400
	S	"=MID(B3,164,400)"	IFSRTGDPAPNMES1

PreS1

MGGWSSKPRQGMGTNLSVPNPLGFFPDHQLDPAFGANSNNPDWDFNPN
KDHWPEAIKVGAGDFGPGFTPPHG
GLLGWSPQAQGILTTVPAAPPPVSTNRQS
GRQPTPI

PreS2

SPPLRDSHPQAMQWNSTTFHQALLDPRVRGLYFPAGGSSSGTVNVPVTTV
SPISSIFSRTGDPAPNMESTTSGFLGPLLVLQAGFFLLTRILTIPQSLDSWWTSLNF
LGGAPTCPGQNSQSPTSNHSPTSCPPICPGYRWMCLRRFIIFLFILLCLIFLLVL

S

IFSRTGDPAPNMESTTSGFLGPLLVLQAGFFLLTRILTIPQSLDSWWTSLNFL
GGAPTCPGQNSQSPTSNHSPTSCPPICPGYRWMCLRRFIIFLFILLCLIFLLVLLD
YQGMLPVCPLLPGTSTTSTGPCKTCTIPAQGTSMFPSCCCTKPSDGNCTCIPIPS
WAFARFLWEWASVRFSWLSLLVPFVQWFAGLSPTVWLSVIWMMWYWGPSLY
NILSPFLPLLPIFFCLWVYI

7.2 氨基酸序列数据清洗

后续的数据分析要求氨基酸序列不能有不能识别的字符，比如空缺值占位符“-”、终止子“*”、未识别氨基酸符号“?”。本节主要处理以上问题，使氨基酸序列符合后续数据分析的要求。

```

# 加载必要的库
library(Biostrings)
library(openxlsx)
# 氨基酸序列清洗函数，使用临时占位符 X 替代错误字符
clean_fasta_with_X <- function(file_path) {
  lines <- readLines(file_path)
  # 过滤出非序列行
  header_lines <- grep( "^>" , lines)
  seq_lines <- grep( "^[^>]" , lines)
  # 只保留有效的氨基酸字符（A-Z，不包括 B, J, O, U, X, Z）
  valid_aa_pattern <- "[^ARNDCQEGHILKMFPSTWYV]"
  cleaned_seq_lines <- lapply(seq_lines, function(i) {
    seq <- lines[i]
    gsub(valid_aa_pattern, "X" , seq)
  })
  cleaned_lines <- character(length(lines))
  cleaned_lines[header_lines] <- lines[header_lines]
  cleaned_lines[seq_lines] <- unlist(cleaned_seq_lines)
  # 写入一个新的清理后的文件
  cleaned_file <- "cleaned_AAseqs.fas"
  writeLines(cleaned_lines, cleaned_file)
  return(cleaned_file)
}
# 导入已比对的序列文件，执行清理并将清理后的数据写入新的文件
clean_seq <- clean_fasta_with_X( "保存位置 / 文件名 .fas" )
sequences <- readAAStringSet(clean_seq)
print(sequences)

```

```

> print(sequences)
AAStringSet object of length 36:
width seq
[1] 190 SILSKTGDVPVNMENIASGLLGPLLVLQAGFFLLTKILTIPOSLSWWTSLNF...TPAQGTSMFPSCCCCKPMDGNCTCIPSSWAFAKYLWEWASVRFWSLSLLVP B Refseq
[2] 190 SILSKTGDPAAPNMENIASGLLGPLLVLQAGFFLLTKILTIPOSLSWWTSLNF...TPAQGTSMFPSCCCCKPTDGNCTCIPSSWAFAKYLWEWASVRFWSLSLLVP 003
[3] 190 SILSKTGDPAAPNMENIASGLLGPLLVLQAGFFLLTKILTIPOSLSWWTSLNF...TPAQGTSMFPSCCCCKPTDGNCTCIPSSWAFAKYLWEWASVRFWSLSLLVP 014
[4] 190 SILSKTGDPAAPNMENIASGLLGPLLVLQAGFFLLTKILTIPOSLSWWTSLNF...TPAQGTSMFPSCCCCKPTDGNCTCIPSSWAFAKYLWEWASVRFWSLSLLVP 023
[5] 190 SILSKTGDVPVNMENIASGLLGPLLVLQAGFFLLTKILTIPOSLSWWTSLNF...TPAQGTSMFPSCCCCKPTDGNCTCIPSSWAFAKYLWEWASVRFWSLSLLVP 029

```

本章小结

本章讲解了获得测序公司基因数据后，开始数据分析前，会遇到的两个关键问题，即位置核对与氨基酸序列清洗。位置核对的关键目的是确保后续氨基酸突变位点编号正确，利用了 Word 和 Excel 的筛选、查找和字符长度选择、计算功能。氨基酸序列清洗通过一个 R 函数实现，原理是通过临时占位符对错误字符进行替换。以上，通过标准化的方法，使这两个环节的处理更为客观，从而符合科学性和严谨性要求，是后续进一步分析的基础。

第三部分 读取突变位点

第 8 章 乙肝病毒基因序列比对与分型分析

乙肝病毒（HBV）基因序列比对与分型分析是下游诸多分析应用的基础。序列比对通过寻找不同 HBV 毒株间的碱基同源关系，支持分型分析和变异检测的实现。Erik S.W. 认为序列比对是一种隐藏在数理逻辑下的艺术（具体可参考 The Art of Multiple Sequence Alignment in R.（<https://bioc.r-universe.dev/DECIPHER/doc/ArtOfAlignmentInR.pdf>））。现在的软件可以近乎做到“一键式”操作，但它背后是多个函数在承担接收序列、间隙交错、动态规划、迭代优化、返回对齐结果等多种功能。

基因分型是基于序列比对的核心应用之一。HBV 基因分型依据受到广泛认可的 8% 序列差异划分为 A-H 型。不同基因型在传播途径、致病性、治疗响应等方面存在显著差异，对临床诊疗具有重要指导意义。序列比对和基因分型都涉及很多专业理论，但如同之前的数据库使用章节，本章立足于实现目的，以“工具实操 + 案例结合”为核心，简化理论原理，细化 Clustal、MEGA 等主流工具的操作步骤，明确 HBV 基因分型标准与判定方法，同时规避常见误区，讲解结果可视化技巧，确保分析过程易懂、可复刻。

8.1 序列比对核心原理与工具（Clustal/MEGA 等）

序列比对是通过算法寻找两条或多条序列间的同源区域（相同或相似碱基），明确序列差异（如碱基替换、插入、缺失）的过程，是 HBV 基因分析的基础步骤。本节简化核心原理，介绍常用工具的特点与适用场景，为后续实操选择合适工具提供依据。

8.1.1 核心原理简化

序列比对的核心是“最大化同源碱基匹配，最小化序列差异”，核心逻辑包括两点：① 同源性判断：相同碱基（A-A、T-T）为完全匹配，相似碱基（如 A-G，均为嘌呤；C-T，均为嘧啶）为部分匹配，通过匹配度量化序列同源性；② 空位插入：当序列长度存在差异时，合理插入空位（用“-”表示），确保同源区域对齐（如 HBV 不同基因型的 S 基因序列，需插入空位对齐保守区）。根据比对序列数量，可分为双序列比对（两条序列对比，如样本与参考序列比对）和多序列比对（多条序列对比，如批量样本分型分析）。

8.1.2 常用工具特点与适用场景

工具名称	核心特点	HBV 分析适用场景	优势与不足
Clustal X (2.1 版本)	免费开源，支持多序列比对，可设置比对参数（如空位罚分），输出多种格式结果	批量 HBV 样本多序列比对（如 10~50 条全基因组 / 靶基因序列），为分型分析打基础	优势：操作简单、可视化强；不足：大样本（>100 条）比对效率低
MEGA (1.1 版本)	集成序列比对、进化树构建、分型分析等功能，支持双序列 / 多序列比对，操作界面友好	包含适用于小规模 ClustalW 和适用于批量的 MUSCLE，适用于基础比对	优势：功能全面、方便裁剪、应用灵活；不足：内置 MUSCLE 在大批量比对时表现不如 MAFFT。超过 50 条序列容易卡顿
BLAST(在线工具)	NCBI 开发的在线比对工具，基于局部比对算法，可快速匹配同源序列	单条 HBV 样本与参考序列比对（如验证样本序列准确性）、未知序列同源性检索	优势：速度快、无需安装；不足：不支持批量多序列比对
MAFFT	开源工具，支持大样本多序列比对，比对精度高，可通过命令行或在线使用	大批量样本(>100 条) 比对，很多专业应用推荐的比对工具	优势：速度比以上其他软件快、最高的比对准确性；不足：使用命令行模式需要适应

实操建议：基础比对优先选择 MEGA 或 Clustal X（少量）；单条序列验证用 BLAST 在线工具；大批量样本分析用 MAFFT 命令行模式。

8.2 同源序列比对实操步骤

本节以 24 条 HBV S 基因序列(下载自 GenBank)为例，分别讲解 Clustal X(多

序列比对)和BLAST(双序列比对)的实操步骤,覆盖“数据准备-参数设置-比对执行-结果导出”全流程。

8.2.1 Clustal X 多序列比对

1) 数据准备:

a) 序列文件: 将24条HBV参考序列合并并在同一个FASTA格式文件“RefSeqSet0”。

b) 工具准备: 下载安装Clustal X 2.1(官网: <https://www.clustal.org/clustal2/>)。

2) 导入序列:

a) 点击菜单栏“File”→“Load Sequences”,选中RefSeqSet0,点击“打开”;若序列格式正确,软件会显示序列名称、长度等信息,点击“OK”完成导入。注意,文件保存路径不能有中文,不然会报错“can not open file”。

b) 检查序列: 导入后在序列列表中查看,确保无序列缺失或格式错误(若出现红色警告,需重新检查FASTA格式)。

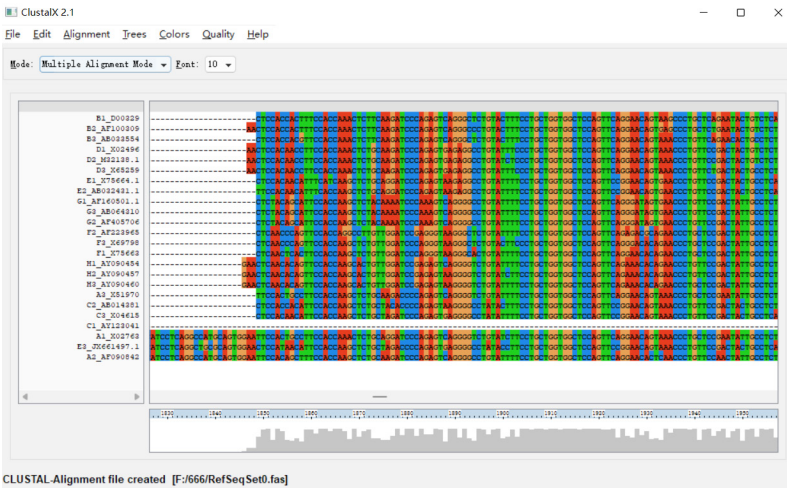
3) 设置比对参数(HBV专属优化): 点击菜单栏“Alignment”→“Alignment Parameters”,弹出参数设置窗口: ① pairwise alignment(双序列比对): Gap opening penalty(空位开放罚分)=10.0, Gap extension penalty(空位延伸罚分)=0.2; ② multiple alignment(多序列比对): Gap opening penalty=10.0, Gap extension penalty=0.2; ③ DNA weight matrix(DNA权重矩阵)=IUB(适用于DNA序列,默认即可); ④ 勾选“Use negative matrix”(使用负矩阵,提升相似碱基匹配精度)。点击“OK”保存参数设置。

4) 执行比对: 点击菜单栏“Alignment”→“Do Complete Alignment”,在弹出的窗口中选择保存地址,并修改输出格式。“.aln”是Clustal默认格式,可用于后续MEGA分析,也可以改为通用的.fasta格式,点击“Save”,软件开始执行比对,等待1~5分钟(根据序列数量调整)。

5) 结果查看与导出: 比对完成后,软件自动显示比对结果: 相同碱基显示为“*”,相似碱基显示为“:”或“.”,空位显示为“-”。点击“File”→“Save Alignment”,确认保存格式为ALN和FASTA,完成导出,用于后续分型分析。

8.2.2 MAFFT 多序列比对

1) 工具准备：打开浏览器，进入 MAFFT 官网下载页面（https://mafft.cbrc.jp/alignment/software/?use_xbridge3=true&loader_name=forest&need_sec_link=1&sec_link_scene=im&theme=light），选择 Windows 版本（熟悉 Linux 的可以下载对应版本，功能更完善、更稳定，只要不涉及 RNA 比对、条数没有上万条，在大多数情况没有明显区别）。



MAFFT version 7
Multiple alignment program for amino acid or nucleotide sequences

Download version
Mac OS X
Windows
Linux
Source

Online version
Alignment
mafft --add
Merge
Phylogeny
Rough tree
Merits & limitations
Algorithms
Tips
Benchmarks
Feedback

About
MAFFT is a multiple sequence alignment program for unix-like operating systems. It offers a range of multiple alignment methods, L-INS-i (accurate; for alignment of < -200 sequences), FFT-NS-2 (fast; for alignment of < -30,000 sequences), etc.

Download and Installation

- Mac OS X
- Linux
- Windows
- Source
- Changelog

The latest version is 7.526 (2024/Apr)

Versions 7.463 - 7.486 had a serious bug in the FFT-NS-i option; unnecessarily much memory was requested and the calculation sometimes failed. This bug also affected the --auto option. Please use 7.487 or higher. (2021/Jul/25)

2) 设置环境变量：在 win 系统界面搜索高级系统设置，点击“高级”栏下“添加环境变量”按钮。在系统变量下新建，粘贴 mafft.bat 的保存地址，不包含自身文件名。

3) 文件准备：将上节的 FASTA 格式文件“RefSeqSet0”保存在软件的根目录。

4) 启动软件：双击 mafft.bat。

5) 导入数据：软件启动后显示“Input file?”，将文件名 RefSeqSet0.fas 输入后回车完成导入。

6) 设置输出信息：窗口询问 “Output file?”，输入文件名及格式，回车确认。这时候还会再问一次 “Output format?” 并给出多个选项，选择 Fasta format 对应序号键入，回车确认。

7) 设置比对策略：窗口询问 “Strategy?”，并给出选项。选择方法如上。可以选择 --auto，软件会自己根据序列特点选择。在序列数 <200 条，长度 <2000 bp 时，可以选择通用的 L-INS-i 模式，或看序列是否整齐进一步区分，如果序列长度相似，可以选 G-INS-i。如果差异较大，可以选 E-INS-i 模式。如果序列长度较大、条数较多，想要快速比对，可以选择 FFT-NS-1 模式。

8) 窗口询问 “additional arguments?”，不需要附带参数直接回车。

9) 开始比对：软件汇总上述选择，按 “Y”（Yes）确认。

10) 完成比对：输出文件名。

```
Input file? (FASTA format; Folder="C:\Program Files (x86)\DAMBE\mafft-win")
@ RefSeqSet0.fas
OK. infile = RefSeqSet0.fas

Output file?
@ RefSeqSetAligned.fas
OK. outfile = RefSeqSetAligned.fas

Output format?
  1. Clustal format / Sorted
  2. Clustal format / Input order
  3. Fasta format / Sorted
  4. Fasta format / Input order
  5. Phylip format / Sorted
  6. Phylip format / Input order
@ 4
OK. arguments = --inputorder

Strategy?
  1. --auto
  2. FFT-NS-1 (fast)
  3. FFT-NS-2 (default)
  4. G-INS-i (accurate)
  5. L-INS-i (accurate)
  6. E-INS-i (accurate)
@ 1
OK. arguments = --auto --inputorder

Additional arguments? (--ep # --op # --kappa # etc)
@

command=
"/usr/bin/mafft" --auto --inputorder "RefSeqSet0.fas" > "RefSeqSetAligned.fas"
OK?
@ [Y] Y
```

8.2.3 实操常见问题解决

Clustal X 导入序列失败：多为保存地址含有中文。也可能是 FASTA 格式错误，检查序列标识行是否以 “>” 开头，序列中是否含中文、空格或特殊字符，删除

多余字符后重新导入。比对结果空位过多：若 HBV 序列长度差异过大（如缺失大片段），可在参数设置中提高空位开放罚分（如从 10.0 调整为 15.0），减少不合理空位。

8.3 HBV 基因分型标准与判定方法

HBV 基因分型依据全基因组或靶基因（如 S 基因、C 基因）的序列差异，目前已明确 A-H 共 8 种主要基因型，不同基因型的序列差异 $\geq 8\%$ ，同一基因型内不同毒株的序列差异 $\leq 4\%$ 。本节明确分型标准，讲解基于进化树和序列一致性的两种核心判定方法，结合案例实操。

8.3.1 HBV 基因分型核心标准

序列差异标准：① 基因型间差异：全基因组序列差异 $\geq 8\%$ ，S 基因序列差异 $\geq 4\%$ ，C 基因序列差异 $\geq 5\%$ ；② 基因型内差异：全基因组序列差异 $\leq 4\%$ ，靶基因序列差异 $\leq 2\%$ 。也有一些研究 S 基因序列差异阈值也设为 8%。

参考序列标准：需使用国际公认的 HBV 各基因型参考序列（如 GenBank Accession ID：A 基因型 AF090841.1、B 基因型 AB033559.1、C 基因型 AB048704.1、D 基因型 X02496.1 等），确保分型准确性。

待检验基因选择标准：优先选择 S 基因（保守区多，序列易获取）或全基因组。

8.3.2 判定方法一：基于进化树的分型法（MEGA 软件）

进化树分型法是 HBV 分型的金标准，核心逻辑：将样本序列与各基因型参考序列构建进化树，若样本序列与某一基因型参考序列聚为一支（置信度 $\geq 70\%$ ），则判定为该基因型。以 10 条 HBV S 基因序列为例，实操步骤如下：

1) 数据准备：

2) 比对文件：使用上节获得的 RefSeqSetAligned.fas 比对文件。

3) 工具准备：打开 MEGA 11 软件。

4) 导入比对文件：点击菜单栏“File”→“Open a File/Session”，选中 ALN 格式比对文件，点击“打开”；在弹出的窗口中选择“Data Type”为“Nucleotide”（核酸序列），点击“OK”完成导入。

5) 构建进化树：点击菜单栏“Phylogeny”→“Construct/Test Neighbor-Joining Tree”（邻接法，最常用的进化树构建方法）。

6) 参数设置：① “Test of Phylogeny”选择“Bootstrap method”（自举检验，评估分支可靠性），“Bootstrap replications”=1000（检验次数，越高结果越可靠）；② “Substitution Model”（替代模型）选择“Kimura 2-parameter”（K2P 模型，适用于 DNA 序列短距离进化分析）；③ 勾选“Pairwise deletion”（成对删除缺失数据，避免数据浪费）；④ 其他参数默认。点击“Compute”，软件开始构建进化树，等待几分钟分钟。

7) 分型判定：树中每个分支代表一条序列，分支上的数字为自举检验值（置信度）；若某样本序列与 B 基因型参考序列聚为一支，且置信度 $\geq 70\%$ ，则判定该样本为 B 基因型。

8.3.3 判定方法二：基于序列一致性的分型法（R 脚本批量判定）

序列一致性分型法适合快速初步分型，核心逻辑：计算样本序列与各基因型参考序列的一致性（计算差异率），阈值设为 8%，差异率保留升序第一。序列数量不多可以使用网页工具（http://jphmm.gobics.de/submission_hbv.html）分型。

```
# 导入必要的库
library(seqinr)
library(ape)
library(writexl)
# 读取比对后的 DNA 序列文件
fasta_file <- “保存路径”
sequences <- read.fasta(file = fasta_file)
# 提取序列名和序列内容
seq_names <- sapply(sequences, attr, “name” )
seq_data <- sapply(sequences, function(x) paste(x, collapse = “” )) # 将每个序列拼接为一个字符串
# 检查读取的序列
print(seq_names)
print(seq_data)
```

```
# 将序列转换为 DNABin 对象
# 首先需要将字符向量转换为列表形式
dna_list <- strsplit(seq_data, "" ) # 将字符串按字符分割
dna_bin <- as.DNABin(dna_list) # 转换为 DNABin 对象
# 检查 DNABin 对象
print(dna_bin)
# 将参考序列和待比较序列分开
reference_sequences <- seq_data[grepl( "[A-H]1$", seq_names)] # 提取参考序列
comparison_sequences <- seq_data[!grepl( "[A-H]1$", seq_names)] # 提取待比较序列
# 计算两个序列的差异率
calculate_difference_rate <- function(reference, comparison) {
  # 确保两个序列长度相同
  if (nchar(reference) != nchar(comparison)) {
    stop( "序列长度不一致" )
  }
  # 计算差异
  differences <- sum(strsplit(reference, NULL)[[1]] != strsplit(comparison, NULL)
[[1]])
  # 计算差异率
  difference_rate <- differences / nchar(reference) # 分母为参考序列的长度
  return(difference_rate)
}
# 初始化一个列表来存储结果
results <- list()
# 循环遍历每个待比较序列
for (comparison_seq in comparison_sequences) {
  # 初始化一个列表来存储与每个参考序列的比较结果
  comparison_results <- list()
}
```

```

# 循环遍历每个参考序列
for (reference_seq in reference_sequences) {
  # 计算差异率
  diff_rate <- calculate_difference_rate(reference_seq, comparison_seq)
  # 判断差异率是否小于等于 8%
  if (diff_rate <= 0.08) {
    comparison_results[[length(comparison_results) + 1]] <- list(reference =
reference_seq,
                                                                    comparison = comparison_seq,
                                                                    difference_rate = diff_rate)
  }
}
# 将每个待比较序列的结果存入总结果中
results[[length(results) + 1]] <- list(comparison_sequence = comparison_seq,
                                       matches = comparison_results)
}
# 打印结果
for (result in results) {
  cat(“待比较序列 :” , result$comparison_sequence, “\n” )
  if (length(result$matches) > 0) {
    cat(“与以下参考序列匹配 ( 差异率 ≤ 8% ) :\n” )
    for (match in result$matches) {
      cat(“ - 参考序列 :” , match$reference, “差异率 :” , match$difference_
rate, “\n” )
    }
  } else {
    cat(“未找到匹配的参考序列。 \n” )
  }
  cat(“\n” )
}

```

```
# 创建一个数据框来存储结果
output_data <- data.frame(
  ComparisonSeq_ID = character(),
  Genotype_ID = character(),
  Difference_Rate = numeric(),
  stringsAsFactors = FALSE
)
# 遍历每个待比较序列
for (result in results) {
  comparison_seq <- result$comparison_sequence
  comparison_seq_name <- seq_names[which(seq_data == comparison_seq)] # 获取待比较序列的名称
  # 如果有匹配的参考序列
  if (length(result$matches) > 0) {
    # 找到差异率最低的参考序列
    min_diff <- min(sapply(result$matches, function(x) x$difference_rate))
    best_match <- result$matches[[which(sapply(result$matches, function(x)
x$difference_rate) == min_diff)]]
    # 获取对应的参考序列名称
    reference_name <- seq_names[which(seq_data == best_match$reference)]
    # 将结果添加到数据框中
    output_data <- rbind(output_data, data.frame(
      ComparisonSeq_ID = comparison_seq_name,
      Genotype_ID = reference_name,
      Difference_Rate = min_diff,
      stringsAsFactors = FALSE
    ))
  }
}
# 将结果输出到 Excel 文件
```



```
output_file <- “保存路径 /comparison_results.xlsx”
write_xlsx(output_data, path = output_file)
cat( “结果已成功导出到 Excel 文件 :” , output_file, “\n” )
```

分型结果如下表（ comparison_results.xlsx ）

	A	B	C
1	ComparisonSeq_ID	Genotype_ID	Difference_Rate
17	F2	F1	0.015761821
18	F3	F1	0.019264448
19	G2	G1	0.001751313
20	G3	G1	0.001751313
21	H2	H1	0.001751313
22	H3	H1	0.003502627
23	17	B1	0.036777583
24	18	B1	0.031523643
25	20	C1	0.043782837
26	22	B1	0.03502627
27	25	B1	0.026269702
28	28	C1	0.007005254

8.4 比对结果可视化呈现

比对与分型结果的可视化可直观展示序列差异、基因型聚类关系，通常用于报告中的整体情况概述部分。本节讲解常用的可视化方式及工具操作，包括序列比对图谱、进化树美化、分型结果统计图表。

8.4.1 进化树美化与导出（ MEGA 软件 ）

MEGA 生成的进化树可通过美化提升可读性，实操步骤：

- 1) 打开进化树文件：在 MEGA 中打开构建好的进化树，点击 “Edit” → “Edit Tree” 进入编辑模式。
- 2) 分支标注：勾选 “Show Bootstrap Values”（显示分支置信度），调整字体大小；
- 3) 颜色区分：选中同一基因型的所有分支，点击 “Color” → 选择对应颜色（如 B 基因型红色、C 基因型蓝色），直观区分不同基因型；
- 4) 简化结构：删除冗余的分支标签，调整树的布局为 “Rectangular”（矩形布局，最易读）。
- 5) 导出保存：点击 “File” → “Export Current Tree”，选择导出格式（PNG、

SVG、PDF 均可），设置分辨率（建议 300dpi，适合打印），点击“Save”完成导出。

8.4.2 进化树美化与导出（iq-Tree 软件 & R 软件）

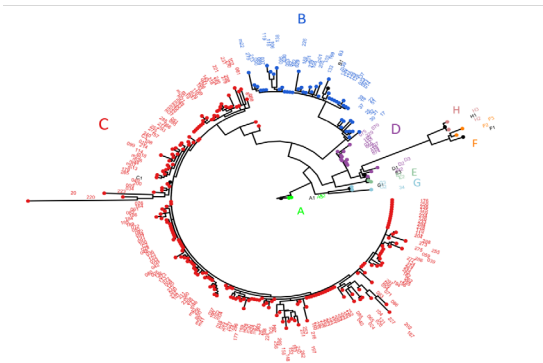
获得分组信息后即可继续可视化。首先绘制系统发育树，使用 iq-Tree 软件（github 链接：<https://iqtree.github.io/>）计算并建立最大似然法系统发育树。在命令窗输入

```
>cd/d F: ( 假设文档保存在 F 盘 )  
>iqtree -s “保存路径” -m MFP -alrt 1000
```

“-m MFP”意为自动寻找最优模型，“-alrt 1000”意为进行 1000 次重复。获得树图文件后，与分组信息一同导入 Rstudio，使用 ggtree 函数包绘制系统发育树，成图后使用 PPT 标注基因型组别。

```
# 加载必要的库  
library(tidyverse)  
library(ggtree) # 绘制 tree 的包  
library(ggnewscale) # 在 tree 填加外圈填色需要的包  
library(readxl)  
## 导入 iq-Tree 软件绘制的 MLtree  
tree <- read.tree(“保存路径 1”) # 读取树图 .treefile 文件  
group <- read_xlsx(“保存路径 2”,1) # 读取分组信息表 .xlsx 文件  
groupINfo <- split(x=group$ComparisonSeq_ID, f = group$Genotype_ID) # 切割分组信息表 (DataFrame -> list)  
phy10 <- groupOTU(.data= tree, .node =groupINfo) # 进行分组配对  
# 绘制圆形系统发育树  
t <- ggtree(phy10, layout = “circular”, size=0.5  
) + # layout, 树的形状；branch.length 是否区分枝长  
  geom_tiplab(mapping = aes(color = group),size=1.5,offset=0.02)+ # 显示每个样品的名字，并添加颜色（颜色随 group 分组）
```

```
scale_color_manual(values = c( "#000000" ," green" ," #1f5dd6" ," #e31a1c" ," #984EA3" ," #88c195" ," #FF7F00" ," #8bc8da" ," #d58386" ),labels=c(
  "NA" ," A1" ," B1" ," C1" ," D1" ," E1" ," F1" ," G1" ," H1" ))+ # 分组
配色
geom_tippoint(size = 0.9, aes(color = group))+# 显示端点
guides(color=" none" )
t
```



本研究人群携带准种 HBV 基因型分布

8.4.3 分型结果统计图表（Excel）

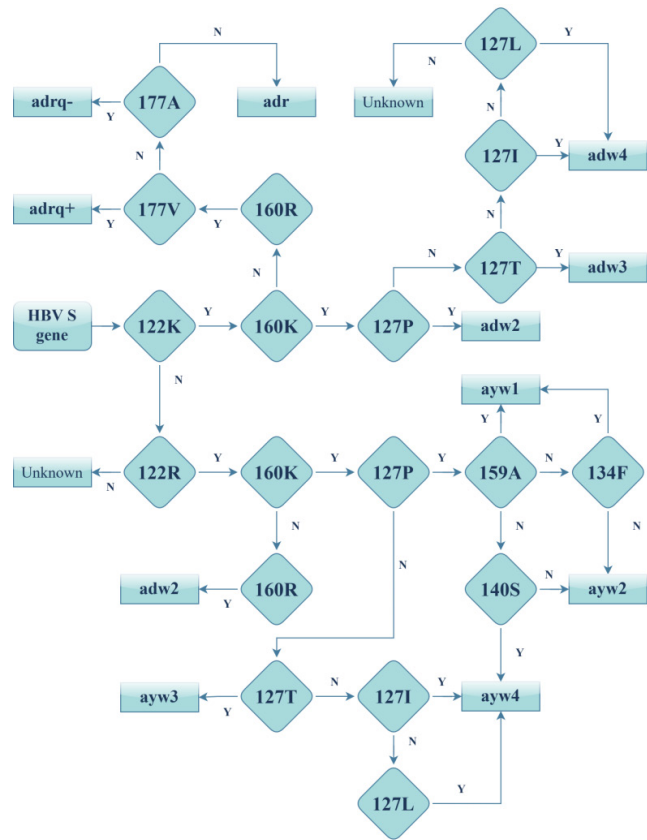
批量样本分型结果可通过统计图表展示（如柱状图、饼图），便于流行病学分析，以 Excel 为例：插入数据透视表。Genotype_ID 作为列标签，ComparisonSeq_ID 作为计数项（求和）。之后就可以制作常规的三线表、饼图、条形图等进行展示。

行标签	计数项:ComparisonSeq_ID
A1	2
B1	89
C1	589
D1	20
E1	2
F1	2
G1	3
H1	2
(空白)	
总计	709

也可使用付费软件 GraphPad Prism（官网 <https://www.graphpad.com/>）进行计算和可视化，该软件自带引导功能，只要输入原始数据表，即可获得统计和可视化实现路径。GraphPad 的具体教程网上很多，可自行检索。

8.4.4 HBV 血清亚型分型

基因亚型和血清亚型对突变分析的影响相对没有基因型差异造成的影响大，在以往认识中，HBV 亚型分型主要应用于流行病学调查和部分临床事件归纳，在很多研究中不是必要步骤。因此，亚型分型相对缺乏系统性和标准化。一些在线工具可以做到基因型分型，但是能做到亚型分型的在线工具或软件很难找到。



依据 HBV S 基因的血清分型程序处理流程

Figure 7.5.4. Algorithms for determining HBV subtype from the primary structure of the HBV S gene.

```

# 加载包
library(writexl)

# 定义函数来确定血清型
determine_serotype <- function(sequence) {
  # 提取相关氨基酸
  aa122 <- substr(sequence, 122, 122)
  aa127 <- substr(sequence, 127, 127)
  aa160 <- substr(sequence, 160, 160)
  aa134 <- ifelse(nchar(sequence) >= 134, substr(sequence, 134, 134), NA)
  aa140 <- ifelse(nchar(sequence) >= 140, substr(sequence, 140, 140), NA)
  aa159 <- ifelse(nchar(sequence) >= 159, substr(sequence, 159, 159), NA)
  aa177 <- ifelse(nchar(sequence) >= 177, substr(sequence, 177, 177), NA)
  # 根据新的分型规则确定血清型
  if (aa122 == "K" && aa160 == "K" ) {
    if (aa127 == "P" ) {
      return( "adw2" ) # 规则 1
    } else if (aa127 == "T" ) {
      return( "adw3" ) # 规则 2
    } else if (aa127 == "I" || aa127 == "L" ) {
      return( "adw4" ) # 规则 3、4
    }
  } else if (aa122 == "K" && aa160 == "R" ) {
    if (aa177 == "V" ) {
      return( "adrq+" ) # 规则 5
    } else if (aa177 == "A" ) {
      return( "adrq-" ) # 规则 6
    } else {
      return( "adr" ) # 规则 7
    }
  } else if (aa122 == "R" && aa160 == "K" ) {

```

```
if(aa127 == "P") {  
  if(aa159 == "A") {  
    return("ayw1") # 规则 7  
  } else if(aa134 == "F") {  
    return("ayw1") # 规则 8  
  } else if(!is.na(aa134) && aa134 != "F" && aa159 != "A") {  
    return("ayw2") # 规则 9  
  }  
} else if(aa127 == "T") {  
  return("ayw3") # 规则 10  
} else if(aa127 == "I" || aa127 == "L") {  
  return("ayw4") # 规则 11、12  
}  
} else if(aa122 == "R" && aa160 == "R") {  
  return("ayr") # 规则 13  
} else if(aa122 == "R" && aa160 == "K") {  
  if(aa127 == "P") {  
    if(aa140 != "S" && aa159 != "A") {  
      return("ayw2") # 规则 14  
    } else if(aa140 == "S" && aa159 != "A") {  
      return("ayw4") # 规则 15  
    }  
  }  
}  
}  
return("Unknown") # 如果没有匹配的规则, 返回" Unknown"  
}  
# 读取 FASTA 文件  
read_fasta_file <- function(file_path) {  
  sequences <- list()  
  seq_id <- NULL
```

```

sequence <- ""

con <- file(file_path, open = "r" )
while (TRUE) {
  line <- readLines(con, n = 1)
  if (length(line) == 0) break # 结束条件

  line <- trimws(line) # 去除空格
  if (startsWith(line, ">")) { # 如果是序列 ID 行
    if (!is.null(seq_id)) { # 保存前一个序列
      sequences[[seq_id]] <- sequence
    }
    seq_id <- substr(line, 2, nchar(line)) # 获取序列 ID
    sequence <- "" # 重置序列
  } else {
    sequence <- paste0(sequence, line) # 拼接序列
  }
}
if (!is.null(seq_id)) { # 保存最后一个序列
  sequences[[seq_id]] <- sequence
}
close(con) # 关闭连接
return(sequences)
}

# 主程序
file_path <- "保存地址 /Serotyping.fas"
sequences <- read_fasta_file(file_path)
results <- sapply(sequences, determine_serotype)
for (seq_id in names(results)) {
  cat(sprintf( "序列 ID: %s, 血清型 : %s\n" , seq_id, results[[seq_id]]))
}

```



```
}  
# 创建一个空的数据框以存储结果  
results_df <- data.frame( 序 列 ID = character(), 血 清 型 = character(),  
stringsAsFactors = FALSE)  
# 处理每个序列并保存结果  
for (seq_id in names(sequences)) {  
  serotype <- determine_serotype(sequences[[seq_id]])  
  results_df <- rbind(results_df, data.frame( 序列 ID = seq_id, 血清型 = serotype,  
stringsAsFactors = FALSE))  
}  
# 导出结果  
output_file <- “保存路径 /Serotyping_Results.xlsx”  
write_xlsx(results_df, output_file)  
cat( “结果已导出到 :” , output_file, “\n” )
```

表 7.5.4 本研究对象的 HBV 血清型分布

Genotype	Subtype					Total
	adw2	adr	adrq+	ayw1	ayw2	
B	45(100%)	0	0	0	0	45
C	11(5.5%)	9(4.5%)	179(89.9%)	0	0	199
D	0	0	0	1(8.3%)	11(91.7%)	12
G	1(100%)	0	0	0	0	1

本章小结

本章围绕 HBV 基因序列比对与分型分析，以工具实操为核心，结合案例系统讲解了核心原理、实操步骤、分型判定、结果验证及可视化方法，核心要点总结如下：① 序列比对是分型的基础，新手优先选择 Clustal X 或 MEGA（全流程可视化），批量比对使用 MAFFT 软件；② 分型判定优先采用进化树法（金标准），结合序列一致性法快速筛查，判定时需严格遵循 HBV 分型标准（基因型

间差异 $\geq 8\%$)；③分型结果需通过交叉验证、参考序列替换验证等多维度确认，规避参考序列选择不当、参数设置错误等常见误区；④比对与分型结果可通过 BioEdit（比对图谱）、MEGA（进化树）、Excel（统计图表）实现可视化，提升结果可读性。本章提供的实操步骤可直接复刻，为 HBV 遗传特征分析、临床诊疗指导及流行病学调查提供可靠技术支撑。

第9章 HBV 基因突变检测方法

突变 (mutation) 是基因序列的改变, 是生物多样性的主要成因^[1]。核苷酸突变反映了准种间差异, 其中, 突变所在基因区是需要重点关注的内容, 因为这往往导致突变株间功能上的差异。突变分析流程主要是基因结构信息收集、突变信息收集, 以及将突变位点信息整合到基因结构信息上, 将突变和效应进行关联。突变效应可能有益、有害或是中性突变, 取决于它们的环境或位置。大多数非中性突变是有害的, 因为在大多数情况, 现有基因序列已经是经历了长期选择后自然产生的最优解, 突变产生的熵增对这一模型产生负担。受突变影响的碱基对越多, 突变的影响就越大, 突变的有害概率也就越大^[2]。

乙肝病毒 (HBV) 基因突变是病毒逃逸宿主免疫、产生耐药性、影响临床预后的关键分子机制, 精准检测突变位点 (如 P 基因耐药突变、S 基因免疫逃逸突变) 对临床诊疗方案制定、疗效监测具有重要指导意义。由于简并性的影响, 是否具有显著的突变效应, 很多时候取决于突变是否改变蛋白质结构和功能, 尤其对于 HBV 而言, 因为除了构建病毒体外, 大多数病毒-宿主作用由病毒蛋白诱发。此外, HBV 基因组非常紧凑, 很多非编码区也包含基因关键元件, 对编码功能进行调控。因此, 一般认为 HBV 研究中, 氨基酸突变比核苷酸突变具有更大的研究价值。尽管如此, 也有很多研究关注 HBV 核苷酸突变。也需要注意到, “突变”这一术语原义即是核苷酸突变。因此, 在分析过程中不能简单地用氨基酸序列覆盖一切。

本章系统覆盖 HBV 基因突变检测的主流方法, 从常规实验室方法到测序法、生物信息学预测法, 详细讲解各方法的核心原理、实操要点与结果判读标准; 通过对比不同方法的优劣, 明确各场景下的最优适配方案; 同时聚焦检测过程中的漏检、误判风险, 提供针对性的规避策略与灵敏度、特异性优化方法, 为精准开展 HBV 突变检测提供全面技术支撑。

9.1 突变效应量化

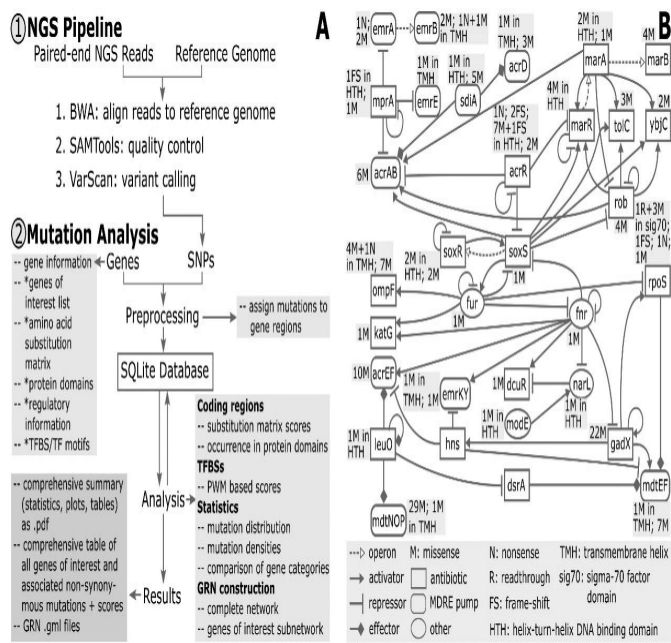
在分子系统生物学中，适应性及其他新增特性是研究人员所关注的重要内容。关于突变的影响，目前已有被广泛接受的总结。单突变可能产生显著影响，但更为常见的情况是，众多小影响突变的积累汇聚成大影响并引起了进化^[3]。突变效应可能是有益的、有害的或中性的，取决于生物体所处的环境或突变所在的位置。一般情况下，突变效应和受突变影响的碱基对数量成正比。由于突变具有少利多害性，有害突变的概率也和突变数量成正比。对这些效应基于分布进行量化（distributions of mutational effects, DMEs），从而帮助理解突变效应。然而，很多因素难以处理，使 DMEs 分析受到很大的干扰。Onuora, A 等^[1]总结最主要的因素：（1）获得可靠的信息，支持将突变归纳到有害、中性、有利效应类别上；（2）捋清突变的相互作用，区分上位效应（epistasis）。

9.2 突变分析软件工具

Markus 等人^[2]开发的 MutaNET 工具可以实现基因调控网络中基因突变的自动化分析，虽然是应用于人类基因组，但其分析流程在 HBV 基因突变分析中也值得借鉴。遗憾的是，人类基因相关的在线注释数据库可选择的选项很多，但是 HBV 基因功能数据库很少见，也很难访问。因此，制作基于 Excel 表格的突变-效应数据库，在 2.2 读取氨基酸突变位点后，可以生成突变位点对应的效应。当然，其效果完全依赖于突变位点编码的准确性及数据库突变数据的丰富程度。虽然不具备联网更新能力，但在本地运行很方便，且数据库准确性更有保障。

Yongwook 等人^[4]开发的 PROVEAN（Protein Variation Effect Analyzer）软件工具（http://provean.jcvi.org/seq_submit.php）具备非常实用的功能，即预测氨基酸取代或缺失是否会对蛋白质的生物学功能产生影响。通过对序列变体进行过滤，软件可以识别重要的非同义（nonsynonymous）或 indel 变体（indel variants）。这一工具的优点是对基因来源物种没有明确限制，在众多自动突变分析软件中很少见；操作简便，尤其在分析缺少报道的新发现位点功能时提供很大帮助。缺点是其原理还是基于序列检索，在此基础上对突变位点的性质进行预测，其数据库不能检查，且结果受数据库影响大。因此，即使获得中性或者非中性结论，也需

要其他方法共同验证。另外一个软件，Sorting Intolerant From Tolerant (SIFT) 具备相同功能，使用相似原理，但其基于经典模式生物数据进行预测，更不适合 HBV 突变分析。Capriotti 等人^[5]开发的 I-Mutant 软件（<https://folding.biofold.org/cgi-bin/i-mutant2.0.cgi>）使用一种基于支持向量机（SVM）的方法，预测单点突变对蛋白质结构稳定性的影响。这一软件优点同样是对序列来源物种没有要求；操作简便，只要输入氨基酸序列、突变位点、突变后的氨基酸，即可开始自动预测。缺点是不能批量处理



MutaNET 工具的工作流程

9.3 常规突变检测方法（限制性酶切 / 高分辨率溶解曲线）

常规突变检测方法具有操作简便、成本低、耗时短的特点，适合临床实验室快速筛查已知热点突变（如 HBV P 基因 rtL180M、rtM204V 耐药突变），核心包括限制性酶切分析（RFLP）和高分辨率溶解曲线分析（HRM）。本节详解两种方法的实操步骤、结果判读及漏检误判规避技巧。

9.3.1 限制性酶切分析（RFLP）：针对已知酶切位点突变

核心原理：利用限制性内切酶识别特定碱基序列（酶切位点）的特性，若 HBV 目标基因区域存在突变，可能导致酶切位点消失或新增，通过凝胶电泳分离酶切产物，根据片段大小差异判断是否存在突变。以检测 rtM204V 突变（碱基替换：A → G，导致酶切位点消失）为例，实操步骤如下：

1) 样本制备：提取患者血清 / 血浆中的 HBV DNA，通过 PCR 扩增含 rtM204 位点的 P 基因片段（扩增引物需覆盖酶切位点，产物长度约 300bp），确保 PCR 产物纯度高（无杂带）。

2) 酶切反应体系配置（20 μ L 体系）：PCR 产物 10 μ L、限制性内切酶（如 Mbo I，识别 rtM204 野生型位点）1 μ L、酶切缓冲液 2 μ L、无菌去离子水 7 μ L，轻轻混匀后短暂离心。

3) 酶切反应：将反应体系置于 37°C 水浴锅中孵育 2~4 小时（确保酶切充分），随后 65°C 水浴 10 分钟灭活内切酶。

4) 凝胶电泳检测：配制 2% 琼脂糖凝胶，加入核酸染料（如 EB），取 10 μ L 酶切产物点样，以 1 $\times$ TAE 为电泳缓冲液，120V 电压电泳 30 分钟，凝胶成像系统观察结果。

5) 结果判读：

a) 野生型（未突变）：rtM204 位点为 A，存在 Mbo I 酶切位点，酶切后产生 2 条片段（如 150bp+150bp）；

b) 突变型（rtM204V）：碱基替换为 G，酶切位点消失，无酶切产物，仅显示原始 PCR 片段（300bp）；

c) 杂合型（野生型 + 突变型）：同时显示 300bp、150bp+150bp 三条带。

9.3.1.1 漏检误判规避与注意事项

1) 漏检风险：仅能检测已知酶切位点的突变，无法检测未知突变；若突变未改变酶切位点（如沉默突变），会导致漏检。规避：明确检测目标为已知热点突变，避免用于未知突变筛查。

2) 误判风险：PCR 扩增不充分、酶切不完全（如孵育时间不足）会导致假阳性（野生型误判为杂合型）；酶污染会导致假阴性（突变型误判为野生型）。规避：严格控制 PCR 扩增条件，确保酶切反应时间充足；设置阳性对照（已知

突变样本）、阴性对照（已知野生型样本）和空白对照（无模板）。

3) 适用局限：仅适用于单个或少数几个热点突变检测，无法同时检测多个突变位点。

9.3.2 高分辨率熔解曲线分析（HRM）：快速筛查未知突变

核心原理：DNA 双链在升温过程中会逐渐解链（熔解），熔解温度（ T_m 值）与 DNA 序列相关（G/C 含量越高， T_m 值越高）。若 HBV 目标基因存在突变，会导致 DNA 双链稳定性改变，熔解曲线形态（ T_m 值、曲线斜率）发生差异，通过高分辨率熔解曲线仪检测差异，实现突变筛查。实操步骤如下：

1) 样本制备：提取 HBV DNA，采用荧光染料（如 SYBR Green I）标记的 PCR 反应体系扩增目标基因片段（如 S 基因免疫逃逸突变区域，产物长度 200~400bp，避免过长导致熔解曲线不清晰）。

2) HRM 检测：将 PCR 产物置于高分辨率熔解曲线仪中，设置升温程序（通常从 60℃ 升至 95℃，升温速率 0.1℃/s，确保分辨率），仪器自动采集荧光信号，生成熔解曲线。

3) 结果判读：

a) 野生型样本：熔解曲线形态一致， T_m 值稳定（如 85℃ $\pm$ 0.5℃）；

b) 突变型样本：熔解曲线形态与野生型存在差异（如 T_m 值降低 0.8℃ 以上、曲线出现额外峰）；

c) 杂合型样本：熔解曲线为野生型与突变型的混合形态， T_m 值介于两者之间，曲线斜率更平缓。

9.3.2.1 漏检误判规避与注意事项

1) 漏检风险：小片段插入/缺失突变、沉默突变可能导致熔解曲线差异不明显，引发漏检；目标片段过长（>400bp）会降低检测灵敏度。规避：选择 200~400bp 的目标片段；设置野生型对照，采用高分辨率仪器（温度分辨率 $\geq$ 0.01℃）。

2) 误判风险：PCR 产物污染、荧光染料浓度过高会导致曲线异常，引发误判；HBV 准种多样性高（多个突变共存）可能导致曲线形态复杂，难以判读。规避：严格无菌操作，设置空白对照；对曲线异常样本，采用测序法进一步验证。

3) 优势：可同时筛查多个未知突变，无需设计特异性探针，操作快速（整个流程 $\leq$ 2 小时），适合大规模样本初筛。

9.4 参考序列的建立

参考序列的确立和检验 (Establishment and validation of reference sequences) 是关键步骤, 决定了后续基因突变位点检验的结果和解释。参考序列, 是参考基因组 (reference genome) 的一部分, 代表某一物种的典型基因组序列, 是通过多种个体基因组序列汇总和优化生成的理想化模型。很多研究直接使用数据库的某条序列作为参考序列, 如朱慧等人^[13]的研究直接使用中国流行的 adw 型流行株 AY167097.1 作为参考序列, 武佳欣等人^[8]的研究从 NCBI 下载 HBV 野生型参考序列并校正; Liu 等人^[9]的研究没有明确说明如何选择参考序列。一些研究使用类似内群的概念, 从实验对象序列集生成参考序列, 如 Wang 等人^[10]的研究; 一些研究使用相同原理, 但是选择从数据库参考序列集合生成参考序列, 如 Yin 等人^[11]HBV S 基因突变和宫内传播相关性的研究。一些使用数据库序列的研究, 没有说明序列来源, 并在分析中进行来源对象和实验对象的对照, 如参考序列来源人群特征、获取方法 (直接提取测序 / 克隆纯化) 等, 和所做研究的区别。系统发育分析, 常见的方向有演化位置确定, 分型, 和突变分析, 使用参考序列的目的不完全一样。

系统发育中的两个概念可以帮助理解参考序列的生物学意义。外群 (outgroup), 或称外系群, 指的是一个参照物, 它与任何一个研究对象的遗传距离, 都远于任何两个研究对象之间的遗传距离, 一般用于研究系统发育关系。从定义可知, 外群需要和研究对象有足够差异, 但也不能差异过大, 以至于失去参照意义。外群的起源是为了解决“置根”问题, 即确定整个类群共同祖先的位置。因此, 外群的进化关系最好是已经相对确定, 且分化时间早于内群。若缺少一个合适的外群时, 定根会很困难。中点定根 (midpoint rooting) 可用于处理这种情况, 即根可以置于两个分类操作单元间最长支的中点上。然而, 该方法假设两个最不同的谱系以相同的速率进化, 在现实中很可能不成立。且只有树中所有分支的进化速度大致相同, 且实际的外群与其它分类群间的支的长度不太短时, 这种方法才具备较高准确度。此外, 在建立系统发育树时一般会使用多个参考序列, 类似于外群思想, 因为使用一个以上的外群通常可以进一步改善推导的树状拓扑的准确度^[12]。

“内群”是和“外群”相对的概念，指共享演化史的感兴趣的分支（clade），用于分支内的差异比较研究。有研究建议在绘制系统发育树时先用内群建树，包括最大似然估计（ML）和自展检验（bootstrap）过程，即先建立无根树，然后使用 EPA 去把外群放在树上，可以防止外群影响内群的枝长^[13]。

使用数据库中某条已发表基因序列，优点是简化工作内容，不需要再进行数据清洗和共享；具备较高的权威性；以及最重要的，可以和同样使用该序列作为参考序列的研究进行横向对比。缺点是，外来参考序列实际上经历了未知的演化过程，直接用于突变分析，可能引入难以确认的误差，对结果解释产生潜在影响。使用样本序列集生成参考序列，优点是形成本土群，易于临床和流行病学意义解释。缺点是本土群演化程度接近，检出的突变位点可能很少，且不易和其他研究进行比较。使用数据库序列集合生成参考序列类似于标准化思想，是相对折衷的方法。优点是对数据库序列本身的特殊性进行均一化，减少突变分析误差，且在使用相同序列集合的研究之间可比。缺点是仍然存在可比性问题。

这里使用 Wang 等人^[10]研究中使用的方法，使用 R 语言复现。这一方法为，使用某型基因序列中频率最高的同一位置的碱基作为参考^[13]，建立序列集合中各基因型的参考序列。如，在某个核苷酸位置，B 型序列中 A 出现了 60 次，而其他碱基的出现频率均低于 A，则选择 A 作为该基因型该位置的参考碱基。而 C 基因型中该位置 T 出现了 40 次，而其他碱基的出现频率均低于 T，则 T 作为 C 基因型该位置的参考碱基。读者可在 NCBI 官网（<https://www.ncbi.nlm.nih.gov/>）下载 HBV Genotype A 演示序列：X02763、AF090842、X51970。使用 MEGA v.11 软件比对，调整位置，使序列长度一致。然后，建立合并脚本。命名一个名为 combine.bat 的文本文档，输入 copy *.fasta RefSeqSet.fasta，使用管理员权限运行。随后，即可使用以下程序提取公共序列作为参考序列。

```
library(Biostrings)
# 从 RefSeqSet.fasta 文件读取序列
seqs <- readDNAStringSet(“保存地址 /RefSeqSet.fasta”)
print(“读取的 fasta 序列信息：”)
print(seqs)
# 将序列转换为数据框（行：位置，列：序列 ID，值：碱基）
# 1. 提取每个序列的碱基（转换为字符并拆分）
```

```
seq_list <- lapply(seqs, function(seq) {  
  strsplit(as.character(seq), "")[[1]] # 拆分序列为单个碱基  
})  
# 2. 检查所有序列长度是否一致（确保能转换为数据框）  
seq_lengths <- sapply(seq_list, length)  
if (length(unique(seq_lengths)) != 1) {  
  stop("所有序列长度必须一致！")  
}  
# 3. 转换为数据框（行 = 位置，列 = 序列 ID）  
seq_data <- as.data.frame(do.call(cbind, seq_list), stringsAsFactors = FALSE)  
colnames(seq_data) <- names(seqs) # 列名设置为 fasta 中的序列 ID  
print("转换后的序列数据框：")  
print(seq_data)  
# 生成参考序列的函数（与原逻辑一致）  
generate_reference <- function(sequences) {  
  reference_sequence <- apply(sequences, 1, function(x) {  
    base_counts <- table(x)  
    reference_base <- names(base_counts[which.max(base_counts)]) # 取最频繁  
    的碱基  
    return(reference_base)  
  })  
  return(paste(reference_sequence, collapse = ""))  
}  
# 按类型分离序列  
A_sequences <- seq_data[, grepl("^A", colnames(seq_data))] # 筛选 A 型序列  
# 生成参考序列  
A_reference <- generate_reference(A_sequences)  
# 输出结果  
print("A 型参考序列：")  
print(A_reference)
```

9.5 测序法突变检测与结果解读

测序法是 HBV 突变检测的“金标准”，可直接获取目标基因的碱基序列，精准识别突变类型（碱基替换、插入、缺失）与位置，分为一代测序（Sanger 测序）和二代测序（NGS），前者适合单个样本已知热点突变验证，后者适合批量样本全基因组突变筛查及低丰度突变检测。本节详解两种方法的实操要点、结果解读及漏检误判规避。

9.5.1 二代测序（NGS）：全基因组突变与低丰度突变检测

9.5.1.1 相对核苷酸替换率和突变模式

突变信息的汇总被称为突变模式（mutation patterns），包括点突变模式、基因区突变模式、核苷酸突变模式、氨基酸突变模式等，目前尚无具体定义。突变模式分析是一种工具性质明显的研究方法，研究重点主要是点突变检测和分析，包括替代、插入、缺失突变分析。Topal 等人^[6]认为了解替代突变模式对研究分子进化和理解替代突变的分子基础非常重要。

突变谱是突变模式的相近概念。在一些研究中，突变谱（mutation spectrum）的概念或用法类似于突变模式的宏观规律或可视化结果^{[14],[15]}；William S.D. 等人^[9]将突变谱描述为“不同局部核苷酸背景下的相对突变率”；Freek 等人^[10]认为突变谱显示了碱基替换中每种突变类型的相对贡献。突变谱的宏观属性体现在其概念包含种群，除了涵盖突变率造成的影响，还涉及环境选择和遗传漂移等因素的影响。因此突变谱具备承上启下的作用，支持后续的突变特征和临床表征相关性研究。

突变频率是突变模式的重要研究内容。1982 年，Gojobori T. 等人^[14]提出相对替换频率（relative substitution frequency）的概念，以及通过碱基组成对突变数量进行标准化的方法。相对替换频率指的是，在一条 A、U、C、G 四种碱基都各有 25 个共计 100bp 的假设序列中，从一种特定核苷酸到另一种核苷酸的替换频率。这是一种研究区域突变意义的方法。由于序列间的碱基组成差异在标准化过程中被消除，因而相对替换频率被认为比原始的碱基替换频率更有参考价值。2022 年，Freek M. 等人^[17]发布 R 软件包 MutationalPatterns，支持常见的突变模式研究任务，包括单核苷酸变体（SNVs）、插入和缺失（Indels）、双碱基置换

(DBS) 和较大的多碱基置换 (MBS)。一些研究中突变类型 (mutation type) 出现次数只算一半, 如 A>C、A>G、A>T、C>A、C>G、C>T、CpG>TpG^[18], 或是 A>C、A>G、A>T 换成 T>C、T>G、T>A^[17]。以下基于 Gojobori T. 等人的方法进行相对替换频率计算。

```
# 载入包
library(readxl)
library(ape)
library(dplyr)
library(ggplot2)
# 读取数据
group_data <- read_excel(“保存地址 /X.gene.BPT.group.xlsx”)
dna_sequence <- readDNAStringSet(“保存地址 /LYXseq.fas”)
# 设定参考序列的名称
ref_name <- “AY123041”
ref_seq <- dna_sequence[[ref_name]]
# 检查参考序列是否在数据集中
if (is.null(ref_seq)) {
  stop(“Reference sequence not found.”)
}
# 初始化替换计数矩阵
substitution_counts <- matrix(0, nrow = 4, ncol = 4, dimnames = list(c(“A”, “T”, “C”, “G”), c(“A”, “T”, “C”, “G”)))
# 计算替换频率
calculate_substitution_counts <- function(ref_seq, seq) {
  for (i in seq_along(ref_seq)) {
    ref_base <- as.character(ref_seq[i])
    seq_base <- as.character(seq[i])
    if (ref_base != seq_base && ref_base %in% c(“A”, “T”, “C”, “G”)
    && seq_base %in% c(“A”, “T”, “C”, “G”)) {
      # 统计替换发生的次数
```

```

      substitution_counts[ref_base, seq_base] <<- substitution_counts[ref_base,
seq_base] + 1
    }
  }
}
# 对所有序列进行计算
for (i in seq_along(dna_sequence)) {
  if (names(dna_sequence)[i] != ref_name) {
    calculate_substitution_counts(ref_seq, dna_sequence[[i]])
  }
}
# 查看替换计数矩阵
print( "Substitution Counts:" )
print(substitution_counts)
# 计算总替换计数
total_substitutions <- sum(substitution_counts)
# 检查总替换计数是否为零
if (total_substitutions == 0) {
  stop( "No substitutions found." )
}
# 计算相对替换频率
relative_frequencies <- substitution_counts / total_substitutions * 100 # 转换为
百分比
# 查看相对替换频率
print( "Relative Frequencies:" )
print(relative_frequencies)
# 将结果转换为数据框
result <- as.data.frame(as.table(relative_frequencies))
names(result) <- c( "From" , "To" , "Relative_Frequency" )
# 自定义颜色

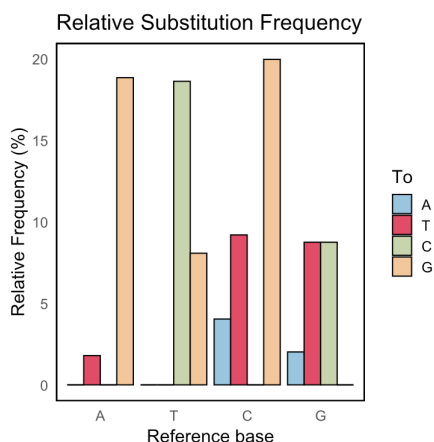
```



```
custom_colors <- c("A" = "#91bfd9", "T" = "#de425b", "C" = "#c2cfa5",  
"G" = "#efc192")  
# 绘制替换频率图  
ggplot(result, aes(x = From, y = Relative_Frequency, fill = To)) +  
  geom_bar(stat = "identity", position = "dodge", color = "black") + # 黑色边框  
  labs(title = "Relative Substitution Frequency",  
    x = "Reference base",  
    y = "Relative Frequency (%)") +  
  theme_minimal(base_size = 15) + # 设置基本字体大小  
  theme(panel.background = element_rect(fill = "white"), # 白底  
    panel.border = element_rect(color = "black", fill = NA, linewidth = 1), # 黑框  
    legend.position = "right", # 图例位置  
    panel.grid.major = element_blank(), # 删除主要网格线  
    panel.grid.minor = element_blank()) + # 删除次要网格线  
  scale_fill_manual(values = custom_colors) # 使用自定义颜色  
# 保存结果  
write_xlsx(result, "相对替换频率结果.xlsx", row.names = FALSE)
```

HBV X 区基于碱基置换的相对突变率

Replacement base	Relative substitution frequency from reference base (%)			
	A	T	C	G
A	0.00	1.79	0.00	18.83
T	0.00	0.00	18.61	8.07
C	4.04	9.19	0.00	19.96
G	2.02	8.74	8.74	0.00



研究样本 HBV X 区相对参考序列的相对替换率

9.5.1.2 R 脚本读取氨基酸突变位点并编制为 Excel 表

加载必要的库

```
library(Biostrings)
```

```
library(openxlsx) # 用于写入 Excel 文件
```

清理 FASTA 文件并用 ‘X’ 替换无效字符

```
clean_fasta_with_X <- function(file_path) {
```

```
  lines <- readLines(file_path)
```

过滤出非序列行

```
header_lines <- grep( “^>” , lines)
```

```
seq_lines <- grep( “^[^>]” , lines)
```

只保留有效的氨基酸字符 (A-Z, 不包括 B, J, O, U, X, Z)

```
valid_aa_pattern <- “[^ARNDCQEGHILKMFPSTWYV]”
```

```
cleaned_seq_lines <- lapply(seq_lines, function(i) {
```

```
  seq <- lines[i]
```

```
  gsub(valid_aa_pattern, “X” , seq)
```

```
})
```

合并 header 和清理后的序列

```
cleaned_lines <- character(length(lines))
```

```
cleaned_lines[header_lines] <- lines[header_lines]
```

```
cleaned_lines[seq_lines] <- unlist(cleaned_seq_lines)
# 写入一个新的清理后的文件
cleaned_file <- "cleaned_AAseqs.fas"
writeLines(cleaned_lines, cleaned_file)
return(cleaned_file)
}
# 清理并读取新的序列文件，使用 'X' 作为占位符
clean_seq_file <- clean_fasta_with_X("保存地址 /GenotypeBcutAA.fas")
sequences <- readAAStringSet(clean_seq_file)
# 检查是否成功加载
print(sequences)
# 指定第一条序列作为参考序列
reference_sequence <- as.character(sequences[[1]])
# 【 计算表 】
# 初始化一个数据框来存储突变位点信息
mutation_info <- data.frame(Sequence = character(),
                             Mutation_Position = character(),
                             stringsAsFactors = FALSE)
# 遍历每个序列（从第二个开始，假设第一个序列是参考序列）
for (i in 2:length(sequences)) {
  current_name <- names(sequences)[i]
  current_sequence <- as.character(sequences[[i]])
  # 将序列分割成字符向量
  ref_chars <- unlist(strsplit(reference_sequence, ""))
  current_chars <- unlist(strsplit(current_sequence, ""))
  # 使用字符级别比较查找突变位点，忽略 '-'
  differences <- which(ref_chars != current_chars & current_chars != "-")
  # 如果有突变，记录相关信息
  if (length(differences) > 0) {
    mutation_info <- rbind(mutation_info,
```

```

        data.frame(Sequence = current_name,
                    Mutation_Position = paste(differences, collapse = " ",
                    stringsAsFactors = FALSE))
    }
}
# 保存突变位点信息到 Excel 文件
output_file <- "保存地址 /mutation_info_corrected.xlsx"
write.xlsx(mutation_info, output_file, rowNames = FALSE)
cat( "突变位点信息已保存到 'mutation_info_corrected.xlsx' 文件中。\\n" )
# 【对照表】
# 初始化一个数据框来存储突变位点信息
mutation_info <- data.frame(Sequence = character(),
                             Mutation_Position = integer(),
                             Reference_Amino_Acid = character(),
                             Mutated_Amino_Acid = character(),
                             stringsAsFactors = FALSE)
# 遍历每个序列（从第二个开始，假设第一个序列是参考序列）
for (i in 2:length(sequences)) {
    current_name <- names(sequences)[i]
    current_sequence <- as.character(sequences[[i]])
    # 将序列分割成字符向量
    ref_chars <- unlist(strsplit(reference_sequence, " "))
    current_chars <- unlist(strsplit(current_sequence, " "))
    # 使用字符级别比较查找突变位点，忽略 '-'
    differences <- which(ref_chars != current_chars & current_chars != "-" )
    # 如果有突变，记录相关信息
    if (length(differences) > 0) {
        for (pos in differences) {
            mutation_info <- rbind(mutation_info,
                                    data.frame(Sequence = current_name,

```

```
Mutation_Position = pos,  
Reference_Amino_Acid = ref_chars[pos],  
Mutated_Amino_Acid = current_chars[pos],  
stringsAsFactors = FALSE))  
  
}  
  
}  
  
}  
  
# 保存突变位点信息到 Excel 文件  
output_file <- “保存地址 / 对照表 B.xlsx”  
write.xlsx(mutation_info, output_file, rowNames = FALSE)
```

9.5.1.3 突变模式可视化

突变谱（mutation spectrum）的概念具有总括性，因此其可视化可作为研究内容及结果总览呈现于文章中，如 Liu 等人^[19]的研究。他们的研究对氨基酸突变导致的性质改变进行预测、汇总可能导致移码和提早产生终止子的突变、对比不同群体中相同位点的突变率差异等，并使用图 1 展示突变位点对应的结构域位置。

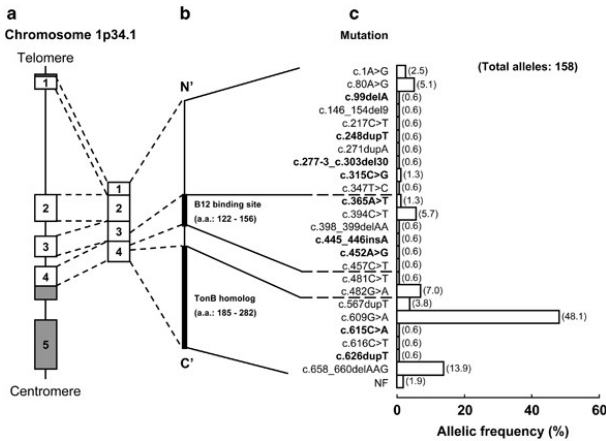


图 1. 在 79 名中国 cbIC-MMA 患者中发现了 MMACHC 突变。（a）Lerner Ellis 等人预测的 MMACHC 基因的基因组结构和（b）蛋白质结构。2 开放和灰色框分别是编码区和非翻译区。黑框代表 B12 结合结构域和 TonB 同源物。（c）在中国 cbIC 患者中发现的突变。括号内为每个突变的等位基因频率。NF 表明在编码区、外显子 - 内含子边界和非翻译区序列中没有发现突变^[19]。

现在使用 R 代码实现不同突变模式可视化方法。

(一) 核苷酸突变谱

加载库

```
library(readxl)
```

```
library(dplyr)
```

```
library(seqinr)
```

```
library(MutationalPatterns)
```

```
library(writexl)
```

```
library(ggplot2)
```

读取分组数据

```
group_data <- read_xlsx(“保存地址 /X.gene.BPT.group.xlsx”, sheet = 2)
```

从 FASTA 文件中读取序列

```
fasta_file <- “保存地址 /LYXseq1.fas”
```

```
sequences <- read.fasta(fasta_file, as.string = TRUE)
```

将序列名称提取为 ID，并存储为数据框

```
seq_data <- data.frame(
```

```
  ID = names(sequences),
```

```
  Sequence = sapply(sequences, function(x) paste(x, collapse = “” ))
```

```
)
```

选择参考序列

```
reference_name <- “AY123041”
```

```
reference_sequence <- seq_data$Sequence[seq_data$ID == reference_name]
```

合并 Excel 数据和序列数据

```
merged_data <- merge(group_data, seq_data, by = “ID” )
```

初始化突变矩阵

```
num_samples <- nrow(merged_data)
```

```
mutation_matrix <- matrix(0, nrow = num_samples, ncol = 12)
```

```
colnames(mutation_matrix) <- c(“A → T”, “A → C”, “A → G”, “T → A”,  
“T → C”, “T → G”,  
“C → A”, “C → T”, “C → G”, “G → A”, “G → T”,
```

“G → C”)

计算突变频数的函数

```
calculate_mutation_counts <- function(data, group_value, reference_sequence) {
```

```
# 根据分组选择数据
```

```
group_data <- data %>% filter(group == group_value)
```

```
# 初始化突变计数
```

```
mutation_counts <- numeric(12)
```

```
names(mutation_counts) <- colnames(mutation_matrix)
```

```
# 逐行遍历基因序列，计算突变
```

```
for (i in 1:nrow(group_data)) {
```

```
sequence <- as.character(group_data[i, "Sequence" ]) # 读取合并后的序列
```

```
# 遍历序列中的碱基
```

```
for (j in 1:min(nchar(sequence), nchar(reference_sequence))) {
```

```
ref_base <- toupper(substr(reference_sequence, j, j)) # 参考碱基
```

```
seq_base <- toupper(substr(sequence, j, j))          # 序列碱基
```

```
# 更新突变计数
```

```
if (ref_base != seq_base) {
```

```
mutation_counts[paste(ref_base, "→", seq_base, sep = " ") <-
```

```
mutation_counts[paste(ref_base, "→", seq_base, sep = " ") + 1
```

```
}
```

```
}
```

```
}
```

```
return(mutation_counts)
```

```
}
```

```
# 计算突变频率并填充突变矩阵
```

```
for (i in 1:num_samples) {
```

```
mutation_counts <- calculate_mutation_counts(merged_data, merged_data[i,  
"group" ], reference_sequence)
```

```
mutation_matrix[i, ] <- mutation_counts
```

```
}
```



```

# 计算每种突变类型的频率和 95% 置信区间
mutation_counts_sum <- colSums(mutation_matrix)
total_mutations <- sum(mutation_counts_sum)
relative_contributions <- mutation_counts_sum / total_mutations
# 避免除以零的错误
if (total_mutations > 0) {
  relative_contributions <- mutation_counts_sum / total_mutations
} else {
  relative_contributions <- rep(0, length(mutation_counts_sum))
}
# 计算标准误差和置信区间
standard_errors <- apply(mutation_matrix, 2, function(x) sd(x) / sqrt(length(x)))
lower_ci <- relative_contributions - 1.96 * standard_errors / total_mutations
upper_ci <- relative_contributions + 1.96 * standard_errors / total_mutations
# 创建一个数据框用于绘图
spectrum_data <- data.frame(
  Mutation_Type = colnames(mutation_matrix),
  Relative_Contribution = relative_contributions,
  Lower_CI = lower_ci,
  Upper_CI = upper_ci
)
# 保存结果到 Excel
write_xlsx(spectrum_data, "突变谱.xlsx")
# 自定义颜色, 设置透明度
fill_color <- "#91b9db" # 主要颜色
alpha_value <- 0.3 # 透明度设置为 30%
# 绘制替换频率图, 设置为单一颜色并增加透明度
ggplot(spectrum_data, aes(x = Mutation_Type, y = relative_contributions)) +
  geom_bar(stat = "identity", position = "dodge",
    fill = fill_color, color = "black", alpha = alpha_value) + # 黑色边框,

```

设置透明度

```
geom_errorbar(aes(ymin = Lower_CI, ymax = Upper_CI), width = 0.2) +  
labs(title = “Relative mutation” ,  
      x = “Mutation type” ,  
      y = “Relative contributions” ) +  
theme_minimal(base_size = 15) + # 设置基本字体大小  
theme(panel.background = element_rect(fill = “white” ), # 白底  
      panel.border = element_rect(color = “black” , fill = NA, linewidth = 1), #
```

黑框

```
legend.position = “right” , # 图例位置  
panel.grid.major = element_blank(), # 删除主要网格线  
panel.grid.minor = element_blank()) # 删除次要网格线
```

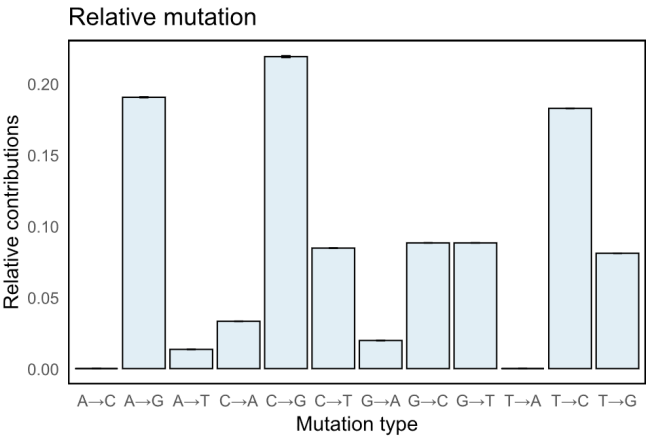
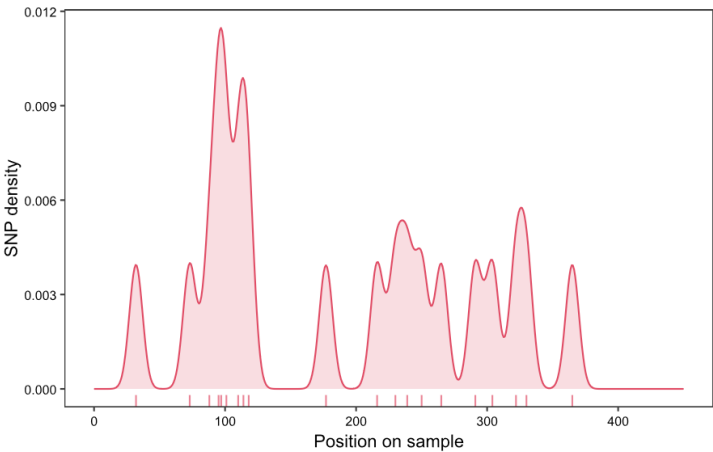


图 HBV X 基因突变谱

表 HBV X 基因突变谱。

Mutation_Type	Relative Contribution	L.CI	U.CI
A → T	0.01	0.01	0.01
A → C	0.00	0.00	0.00
A → G	0.19	0.19	0.19
T → A	0.00	0.00	0.00
T → C	0.18	0.18	0.18

Mutation_Type	Relative Contribution	L.CI	U.CI
T → G	0.08	0.08	0.08
C → A	0.03	0.03	0.03
C → T	0.08	0.08	0.08
C → G	0.22	0.22	0.22
G → A	0.02	0.02	0.02
G → T	0.09	0.09	0.09
G → C	0.09	0.09	0.09



。HBV X 基因 SNP 密度。可以结合 X 基因功能区一起看。可以不同基因型之间比较。

9.6 使用生物信息学工具预测筛选突变位点

生物信息学预测法可在大规模测序数据中快速筛选出具有潜在功能意义的突变位点（如影响蛋白质结构、导致耐药、免疫逃逸的突变），减少无效突变的验证工作，核心依赖于突变功能预测工具和 HBV 专属数据库。本节讲解常用预测工具、实操流程及结果解读。

9.6.1 常用预测工具与适用场景

在之前已经介绍过一些基于数据库的突变预测工具，这里再补充两个：

工具名称	核心功能	HBV 突变分析适用场景	使用方式
SIFT	预测氨基酸突变对蛋白质功能的影响（有害 / 中性）	筛选影响 HBV 蛋白（如聚合酶、表面抗原）功能的突变	在线工具（ https://sift.bii.a-star.edu.sg/ ）
PolyPhen-2	预测突变对蛋白质结构与功能的损害程度（可能有害 / 良性）	精准评估突变的功能影响（如耐药突变对聚合酶活性的影响）	在线工具（ http://genetics.bwh.harvard.edu/pph2/ ）

首先得指出，很遗憾的是这两种工具都不适用于 HBV 研究，因为它们的数据库不包含 HBV 相关的注释信息或没有用过 HBV 数据进行模型训练。因此在很多研究里都是通过关联突变位点和所在功能区、结局信息等，预测突变可能造成 HBV 功能损害或增强。当然，从严谨角度看，这种推测最好是已获得的全长突变信息。

SIFT 由新加坡基因组研究所和美国克雷格·文特尔研究所共同开发。该工具根据序列同源性和氨基酸的物理特性，预测氨基酸的取代是否影响蛋白质功能，多用于分析自然发生的非同义突变或实验室诱导的错义突变。若要使用 SIFT 对 HBV 氨基酸序列进行分析，需要本地部署 SIFT 并自行创建预测数据库。相比之下，PolyPhen 的操作界面更为直观。PolyPhen 是通过有监督学习方法（如，朴素贝叶斯算法）建模预测，最终输出得分介于 0-1 之间，得分和损害蛋白质功能的概率成正比。预测非同义突变对蛋白质结构和功能的影响。其核心判读依据是得分阈值和模型适用场景。HumDiv 模型适用于复杂疾病或表型分析，如，罕见等位基因评估、自然选择分析，对轻度有害变异敏感，强调降低漏诊率。HumVar 适用于单基因遗传病，对强致病性变异敏感，强调降低误诊率。

9.6.2 操作流程

- 1) 数据准备：从 GenBank 下载序列。
- 2) 打开 PolyPhen 页面，以 FASTA 格式输入氨基酸序列；



PolyPhen-2

prediction of functional effects of human nsSNPs

[Home](#)
[About](#)
[Help](#)
[Downloads](#)
[Batch query](#)
[WRESS.db](#)

PolyPhen-2 (Polymorphism Phenotyping v2) is a tool which predicts possible impact of an amino acid substitution on the structure and function of a human protein using straightforward physical and comparative considerations. Please, use the form below to submit your query.

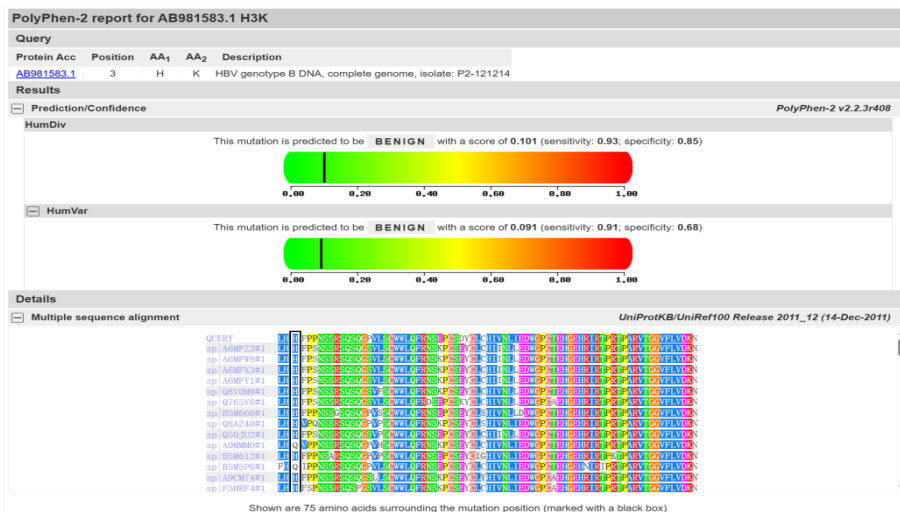
Query Data

Protein or SNP identifier	
Protein sequence in FASTA format	>AB091583.1 HBV genotype B DNA, complete genome, isolate: P2-121214
 LMHFPPSSSGGGQFPLSCWHLGFNSPFCVCLGLIVNLIDWGPGCTENGHHR
 IPTPTARYTGVLGVDMKNTTESRLVDVDFQFSRG
Position	3
Substitution	AA ₁ A R N D C E Q Q G H I L K M F P S T W Y V AA ₂ A R N D C E Q Q G I L K M F P S T W Y V
Query description	

Display advanced query options

3)AA1 为野生型，AA2 为突变型，Position 填氨基酸序列上的突变位置。现在假设发现了 H3K 突变，按上图输入。程序处理需要排队，可以过几分钟点一下 Refresh 按钮刷新；

4) 结果示意:



9.7 不同检测方法优劣对比与适配场景

不同 HBV 突变检测方法在灵敏度、特异性、检测范围、成本、耗时等方面存在显著差异,需根据检测目标(已知热点突变/未知突变、单个样本/批量样本、高丰度突变/低丰度突变)、实验室条件(设备、技术水平)选择最优适配方案。本节通过对比各方法优劣,明确不同场景下的选择策略。

9.7.1 主流检测方法核心指标对比

检测方法	灵敏度（最低检测丰度）	特异性	检测范围	成本（单样本）	耗时	设备要求
限制性酶切（RFLP）	低（ $\geq 20\%$ ）	高（仅识别已知酶切位点突变）	已知热点突变（单个/少数几个）	低（ ≤ 50 元）	短（2~4小时）	低（电泳仪、水浴锅）
高分辨率熔解曲线（HRM）	中（ $\geq 5\%$ ）	中（可能误判复杂突变）	已知/未知突变（单个基因区域）	中（50~100元）	短（1~2小时）	中（高分辨率熔解曲线仪）
一代测序（Sanger）	中（ $\geq 20\%$ ）	高（直接测序，结果可靠）	已知/未知突变（单个基因片段）	中（100~200元）	中（4~8小时）	中（毛细管电泳测序仪）
二代测序（NGS）	高（ $\geq 1\%$ ）	高（需生物学过滤假阳性）	全基因组已知/未知突变（多个位点）	高（500~1000元）	长（1~3天）	高（高通量测序仪、生物信息学分析平台）
生物信息学预测	-（基于测序数据，依赖测序灵敏度）	中（需多工具验证）	筛选功能意义突变	低（免费在线工具）	短（0.5~2小时）	低（电脑，联网）

本章小结

本章系统覆盖了HBV基因突变检测的主流方法，从常规实验室方法（RFLP、HRM）到精准测序法（一代、二代测序），再到功能突变筛选的生物信息学预测法，核心围绕“精准检测、规避漏检误判”展开，核心要点总结如下：① 不同检测方法各有优劣，需根据检测目标、实验室条件分场景适配（如临床快速筛查选RFLP/HRM，精准验证选一代测序，大规模低丰度突变检测选二代测序）；② 结果判读是核心环节，需重点关注峰图质量（一代测序）、变异丰度与测序深度（二代测序）、多工具验证（生物信息学预测），避免因操作或分析不当导致误判；③ 灵敏度与特异性优化需贯穿全流程，从样本质量控制、实验操作优化到数据分析筛选，建立完善的质量控制体系（设置对照、重复验证）；④ 生物信息学预测仅为功能突变筛选手段，最终需结合临床信息和体外实验验证，确保结果的可靠性。本章提供的方法与优化策略可直接指导临床与科研中的HBV突变检测工作，为诊疗方案制定和基础研究提供精准的数据支撑。

第四部分 基础统计和可视化

第 10 章 R 语言基础绘图及检验

10.1 频数堆积图

在突变位点读取章节，获得了如下数据表：

	A	B	C
1	type ▾	site ▾	mutrate ▾
2	B_NIT	S1_56	1.00
3	B_NIT	S1_87	1.00
4	B_NIT	S2_36	0.17
5	B_NIT	S2_41	0.43
6	B_NIT	S2_53	0.43
7	B_NIT	S_143	1.00
8	B_NIT	S_200	0.60
9	B_NIT	S_210	0.40
10	B_NIT	S_213	0.40
11	B_OBT	S1_10	0.14
12	B_OBT	S1_55	0.14
13	B_OBT	S1_56	1.00
14	B_OBT	S1_87	1.00
15	B_OBT	S2_36	0.43
16	B_OBT	S_59	0.11
17	B_OBT	S_103	0.14
18	B_OBT	S_143	1.00
19	B_OBT	S_200	1.00
20	B_DBT	S1_56	1.00
21	B_DBT	S1_87	1.00

type 为分组类型，B_NIT 代表 B 基因型无传播风险组，B_OBT 代表 B 基因型隐匿性传播组，C_DBT 代表显性传播组；site 为突变位点；mutrate 为突变率。现在要绘制各突变位点突变频数堆积图。代码如下：

```
# 导入数据
library(readxl)
dat01<-read_excel(“保存地址 / 文件名.xlsx” ,1)
```

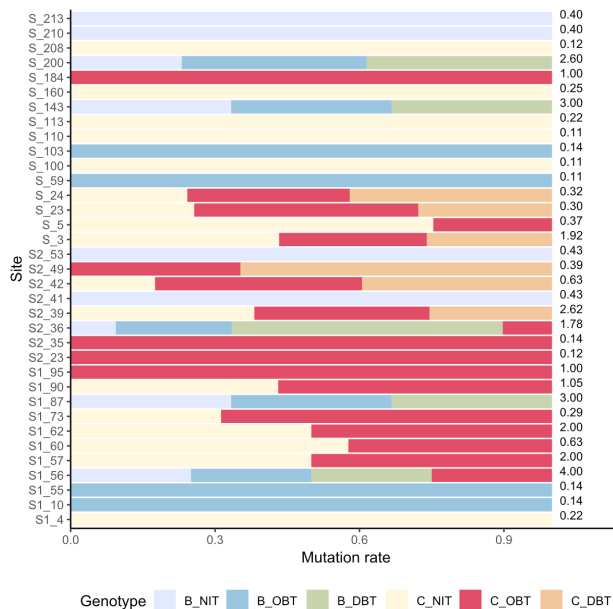


```
head(dat01)
# 定义 X 轴的顺序
dat01$site<-factor(
  dat01$site,
  levels = c( "S1_4" ,      "S1_10" ,      "S1_55" ,      "S1_56" ,
"S1_57" ,      "S1_60" ,      "S1_62" ,      "S1_73" ,      "S1_87" ,
"S1_90" ,      "S1_95" ,      "S2_23" ,      "S2_35" ,      "S2_36" ,
"S2_39" ,      "S2_41" ,      "S2_42" ,      "S2_49" ,      "S2_53" ,
"S_3" ,        "S_5" ,        "S_23" ,        "S_24" ,        "S_59" ,
"S_100" ,      "S_103" ,      "S_110" ,      "S_113" ,      "S_143" ,
"S_160" ,      "S_184" ,      "S_200" ,      "S_208" ,      "S_210" ,
"S_213"
))
# 定义叠加顺序
dat01$type<-factor(
  dat01$type,
  levels = rev(c( "B_NIT" , " B_OBT" , " B_DBT" , " C_NIT" , " C_
OBT" , " C_DBT" )))
library(tidyverse)
dat01 %>%
  group_by(site) %>%
  summarise(total_number=sum(mutrate)) %>%
  ungroup() %>%
  mutate(ratio=total_number/sum(total_number)) %>%
  mutate(ratio=scales::percent(ratio)) -> dat02
# 绘制堆积条图
ggplot(data = dat01,aes(x=site,y=mutrate,fill=type))+
  geom_bar(stat = "identity",position = "fill" )+
  scale_fill_manual(values = c( "B_NIT" = " #e0e7ff" , " B_
OBT" = " #91bfdb" , " B_DBT" = " #c2cfa5" , " C_NIT" = " #fff6da" , " C_
```

```

OBT" = "#de425b" , " C_DBT" = "#efc192" ),
      limits=c( "B_NIT" , " B_OBT" , " B_DBT" , " C_NIT" , " C_
OBT" , " C_DBT" ))+
  geom_text(data=dat02,
    aes(x=site,y=1,label=sprintf( "%.2f" , total_number),
      label=paste0(total_number," \n" , " ( “,ratio,” ” ) ),),
    inherit.aes = FALSE,
    vjust=-0, size=3,nudge_y = 0.04)+
  scale_y_continuous(expand = expansion(mult=c(0,0.1)))+
  coord_flip()+ # 横纵坐标转置
  theme(panel.background = element_blank(),
    axis.line = element_line(),
    legend.position = "top" )+
  labs(x=" Site" ,y=" Mutation rate" )+
  guides(fill=guide_legend(title = "Genotype" ,nrow = 1,byrow = FALSE))+
  theme(legend.position=" bottom" )

```



10.2 堆积图可视化组间差异卡方检验

要比较干预组（Y）和非干预组（NON）DNA 水平分布有没有显著差异，进行卡方检验，并通过堆积条图展示。DNA 为血清 HBV-DNA 水平，在 Excel 表对数化，并按照通用划定方法分为低、中、高三个级别。数据表如下：

	A	B	C	D
1	interve	DNA	LogDN	DNALe
35	Y	18700000.00	16.74403	3
36	Y	18700000.00	16.74403	3
37	Y	72.80	4.287716	1
38	Y	13000000.00	16.38046	3
39	Y	13000000.00	16.38046	3
40	Y	22.40	3.109061	1
41	Y	3270.00	8.092545	2
42	Y	0.00	0	1
43	Y	414000.00	12.93362	2
44	Y	136000000.00	18.72817	3
45	Y	7860000.00	15.8773	3
46	Y	414000.00	12.93362	2
47	Y	10000000.00	16.1181	3
48	NON	27.00	3.295837	1
49	NON	75.00	4.317488	1
50	NON	8720000.00	15.98113	3
51	NON	128.00	4.85203	1
52	NON	453.00	6.115892	1
53	NON	32.10	3.468856	1
54	NON	3470.00	8.15191	2
55	NON	264000.00	12.4837	2

代码如下：

```
# 载入必要的包
```

```
library(ggplot2)
```

```
library(ggpubr)
```

```
library(dplyr)
```

```
library(readxl)
```

```
# 读取数据
```

```
df <- read_xlsx(“保存地址 / 文件名 .xlsx” , 1)
```

```
head(df)
```

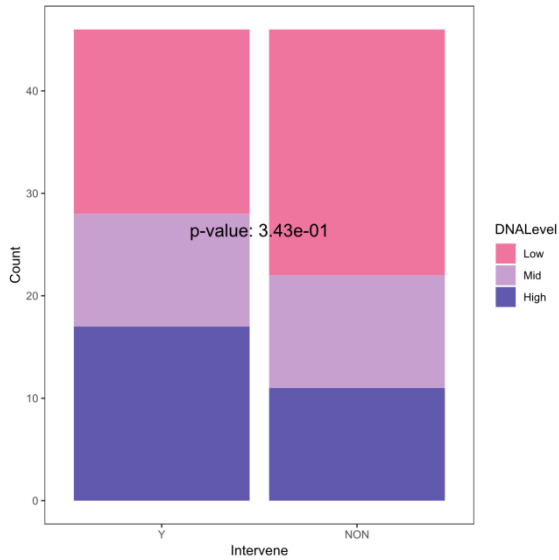
```
# 将 DNALevel 转换为因子，并设置因子水平
```

```
df$DNALevel <- factor(df$DNALevel, levels = c( “1” , “2” , “3” ), labels =
```

```

c( "Low" , "Mid" , "High" ))
# 将 intervene 转换为因子
df$intervene <- factor(df$intervene, levels = c( "Y" , "NON" ), labels =
c( "Y" , "NON" ))
# 计算频数
df_count <- df %>%
  group_by(intervene, DNALevel) %>%
  summarise(count = n(), .groups = 'drop' )
# 绘制堆积图
p <- ggplot(df_count, aes(x = intervene, y = count, fill = DNALevel)) +
  geom_bar(stat = "identity" , position = "stack" ) +
  scale_fill_manual(values = c( "Low" = "#f0759e" , "Mid" = "#c8a0cf" ,
"High" = "#6059ad" )) + # 指定颜色
  labs(x = "Intervene" , y = "Count" ) +
  theme_minimal()
# 进行卡方检验
chisq_test <- chisq.test(table(df$intervene, df$DNALevel))
print(chisq_test)
# 根据检验结果添加显著性标记
p <- p + annotate( "text" , x = 1.5, y = max(df_count$count) * 1.1,
  label = paste( "p-value:" , formatC(chisq_test$p.value, format = "e" ,
digits = 2)),
  size = 5, hjust = 0.5)+
  theme_bw() +
  theme(panel.grid = element_blank())
print(p)

```



10.3 箱线图及组间差异检验

数据表如下：

	A	B	C	D	E	F	G	H	I	J
1	ID	outcon	genoty	DNA	LogDNA	HBsA	HBsA	HBcA	HBcA	age
2	3	1	1	74.40	4.309455942	1	0	0	1	29.27
3	4	1	2	12700000.00	16.35711255	1	0	1	0	27.33
4	5	1	2	9180000.00	16.03253776	1	0	1	0	29.85
5	10	1	2	2040000.00	14.52846037	1	0	1	0	26.11
6	12	1	2	1200.00	7.090076836	1	0	1	0	28.75
7	14	1	1	5670000.00	15.55069968	1	0	1	0	28.53
8	18	1	2	97.60	4.580877493	1	0	0	1	36.71
9	23	1	1	214.00	5.365976015	1	0	0	1	30.47
10	25	1	2	352.00	5.863631176	1	0	0	0	25.99
11	26	2	2	48600000.00	17.69913409	1	0	0	1	36.69
12	28	3	2	276.00	5.620400866	1	0	1	0	26.89
13	29	3	1	8860000.00	15.99705732	1	0	1	0	29.04
14	30	3	1	33900000.00	17.33892557	1	0	1	0	21.18
15	33	2	2	27.00	3.295836866	1	0	0	1	27.16
16	34	1	2	482.00	6.177944114	1	0	0	1	25.69
17	36	2	2	75.00	4.317488114	1	0	0	0	33.10
18	37	3	1	14900000.00	16.51687177	1	0	1	0	26.66
19	39	1	2	184000.00	12.12269104	1	0	0	0	32.76
20	40	3	2	1560000.00	14.26019638	1	0	1	0	30.05
21	41	3	2	126.00	4.836281907	1	0	1	0	31.34
22	42	1	2	1040.00	6.946975992	1	0	0	1	29.29
23	48	1	2	0.00	0	1	0	0	1	32.53
24	49	2	2	80400.00	11.29476946	1	0	0	1	29.71

代码如下：

载入包

```
library(ggplot2)
```

```
library(ggpubr)
```

```
library(dplyr)
```

```
library(readxl)
```

```
library(tidyr)
```

```
# 读取数据表
```

```
df <- read_xlsx(“保存地址 /ID_Outcome_biomarker.xlsx” , 1)
```

```
# 查看表头
```

```
head(df)
```

```
# 将 HbeAg 和 HbeAb 转换为因子，并设置因子水平
```

```
df$HBeAg <- factor(df$HBeAg, levels = c( “0” , “1” ), labels =  
c( “HBeAg-” , “HBeAg+” ))
```

```
df$HBeAb <- factor(df$HBeAb, levels = c( “0” , “1” ), labels =  
c( “HBeAb-” , “HBeAb+” ))
```

```
# 创建一个新的因子变量来表示四个组
```

```
df$Group <- factor(c(as.character(df$HBeAg), as.character(df$HBeAb)),  
                  levels = c( “HBeAg-” , “HBeAg+” , “HBeAb-” , “HBeAb+” ))
```

```
# 【报错】 Error in `<-`:
```

```
#! Assigned data `factor(...)` must be compatible with existing data.
```

```
#✖ Existing data has 240 rows.
```

```
#✖ Assigned data has 480 rows.
```

```
#i Only vectors of size 1 are recycled.
```

```
#Caused by error in `vectbl_recycle_rhs_rows()`:
```

```
# ! Can't recycle input of size 480 to size 240.
```

```
# 【解决办法】在以上创建新的因子变量前进行扩容：
```

```
# 创建一个新的因子变量来表示四个组（修改版）
```

使用 `pivot_longer` 将数据框从宽格式转换为长格式，从而创建一个新的变量 `Group` 来表示不同的组。

然后再将 `Group` 转换为因子并设置其水平

```
df_long <- df %>%
```

```
  pivot_longer(cols = c(HBeAg, HBeAb), names_to = "GroupType", values_to = "Group")
```

将 `Group` 转换为因子，并设置因子水平

```
df_long$Group <- factor(df_long$Group, levels = c("HBeAg-", "HBeAg+", "HBeAb-", "HBeAb+"))
```

创建一个包含颜色的向量，顺序与组的顺序对应

```
group_colors <- c("#c8a0cf", "#f0759e", "#78b9db", "#24d4ce")
```

使用 `ggplot` 绘制箱线图，并手动设置组的填充颜色

```
p <- ggplot(df_long, aes(x = Group, y = LogDNA, fill = Group)) +
```

```
  geom_boxplot() +
```

```
  labs(x = "Group", y = "LogDNA") +
```

```
  scale_fill_manual(values = group_colors) +
```

```
  theme_minimal()
```

添加显著性标记

注意：多组比较需要先设置比较列表 `my_comparisons`

```
my_comparisons <- list(
```

```
  c("HBeAg-", "HBeAg+"),
```

```
  c("HBeAb-", "HBeAb+"),
```

```
  c("HBeAg-", "HBeAb-"),
```

```
  c("HBeAg+", "HBeAb+")
```

```
)
```

```
plot1 <- p + stat_compare_means(aes(group = Group),
```

```
  label = "p.format",
```

```
  method = "t.test",
```

```
  comparisons = my_comparisons,
```

```
  p.adjust.method = "bonferroni") +
```

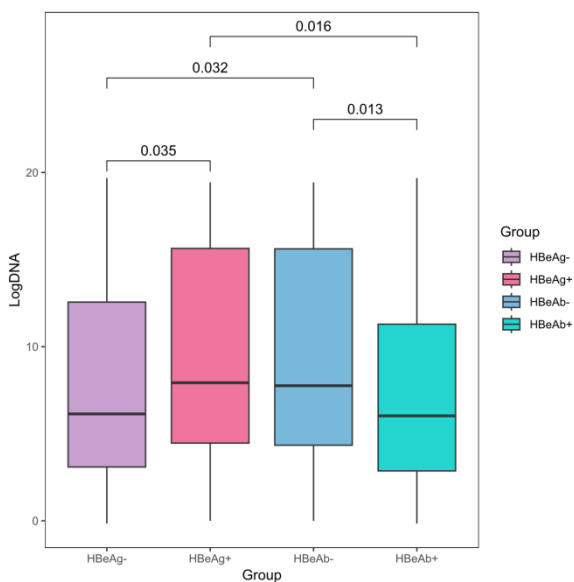


```

theme_bw() +
theme(panel.grid = element_blank())

print(plot1)

```



10.4 小提琴图合并箱线图

数据表部分如下：

SEQN	LUXSME	LUXCAF	LSMLEV	CAPLEV	LSMBILEV	CAPBILEV	LBXVI	VD	Group
93707	4.90	209.00	0	0	0	0	58.20	2.75	1
93712	3.70	218.00	0	0	0	0	59.90	1.80	1
93713	7.10	206.00	0	0	0	0	63.50	2.35	1
93714	6.80	293.00	0	3	0	1	47.50	5.65	1
93715	6.80	255.00	0	1	0	1	70.60	3.75	1
93729	3.70	242.00	0	1	0	1	46.70	18.70	1
93731	5.60	286.00	0	2	0	1	32.00	1.55	1
93735	9.10	363.00	1	3	1	1	89.10	0.60	1
93738	3.30	182.00	0	0	0	0	63.40	8.10	1
93756	3.30	242.00	0	1	0	1	65.80	5.50	1
93759	3.60	298.00	0	3	0	1	121.00	3.05	1
93760	4.80	291.00	0	2	0	1	47.40	1.55	1
93763	6.10	324.00	0	3	0	1	84.00	4.15	1
93766	4.00	255.00	0	1	0	1	52.90	1.20	1
93767	4.80	340.00	0	3	0	1	81.80	5.00	1

代码如下：

```
# 载入包
library(ggplot2)
library(ggpubr)
library(dplyr)
library(readxl)
library(tidyr)

# 读取数据表
df0 <- read_xlsx(“保存地址 /HBV 和 VD 补剂分组肝病结局差异.xlsx”, 1)
# 查看表头
head(df)

# 移除包含 NA 或 Inf 值的行
df <- na.omit(df0)

# 将 Group 转换为因子，并设置因子水平
df$Group <- factor(df$Group, levels = c(“1”, “2”, “3”, “4”), labels =
c(“HBV+/SpVD-”, “HBV-/SpVD-”, “HBV+/SpVD+”, “HBV-/SpVD+”))
# 创建一个包含颜色的向量，顺序与组的顺序对应
group_colors <- c(“#c8a0cf”, “#f0759e”, “#78b9db”, “#24d4ce”)

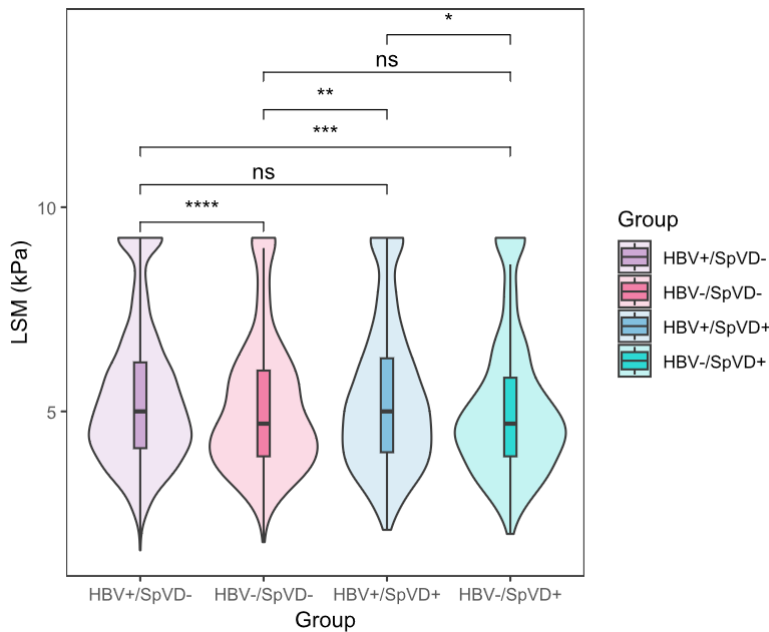
# 【LSM】
# 计算 Q1, Q3 和 IQR
Q1 <- quantile(df$LUXSMED, 0.25)
Q3 <- quantile(df$LUXSMED, 0.75)
IQR <- Q3 - Q1
# 设置离群值的边界
lower_bound <- Q1 - 1.5 * IQR
upper_bound <- Q3 + 1.5 * IQR
```

```

# 温莎化处理
df_winsorized <- df %>%
  mutate(LUXSMED = ifelse(LUXSMED < lower_bound, lower_bound,
                           ifelse(LUXSMED > upper_bound, upper_bound, LUXSMED)))
# 检查温莎化后的数据
summary(df_winsorized)
# 结合箱线图和小提琴图
p <- ggplot(df_winsorized, aes(x = Group, y = LUXSMED, fill = Group)) +
  geom_violin(trim = TRUE, alpha = 0.3) + # 小提琴图
  geom_boxplot(width = 0.1, position = position_dodge(0.9), outlier.shape = NA)
+ # 箱线图
  labs(x = "Group", y = "LSM (kPa)") +
  scale_fill_manual(values = group_colors) +
  theme_minimal()
# Wilcoxon 检验的比较列表
my_comparisons <- list(
  c("HBV+/SpVD-", "HBV-/SpVD-"),
  c("HBV+/SpVD-", "HBV+/SpVD+"),
  c("HBV+/SpVD-", "HBV-/SpVD+"),
  c("HBV-/SpVD-", "HBV+/SpVD+"),
  c("HBV-/SpVD-", "HBV-/SpVD+"),
  c("HBV+/SpVD+", "HBV-/SpVD+")
)
# 添加显著性标记
plot1 <- p + stat_compare_means(aes(group = Group),
                                label = "p.signif",
                                method = "wilcox.test",
                                comparisons = my_comparisons,
                                p.adjust.method = "BH") +
  theme_bw() +

```

```
theme(panel.grid = element_blank()) + guides(fill = "none" )  
# 显示图形  
print(plot1)
```



本章小结

本章围绕基因流行病学研究中常用的 R 语言数据可视化与统计检验方法展开，系统介绍了 4 类核心分析工具的实现逻辑与代码应用，覆盖了从“频数分布展示”到“组间差异验证”的完整流程：

首先，针对突变位点的分布特征，通过频数堆积图实现了不同基因型传播风险组在各突变位点的突变率占比可视化，结合 `factor()` 函数自定义坐标轴顺序，清晰呈现了位点间的突变模式差异；同时通过叠加总突变数与占比标签，增强了图形的信息密度。

其次，针对分类变量的分布差异，采用堆积图结合卡方检验的方式，直观展示了干预组与非干预组的 HBV-DNA 水平分布特征，并通过卡方检验量化了组间分布的统计学差异，实现了“可视化 + 统计验证”的结合。

再次，针对连续型生物标志物（如 LogDNA）的组间差异，通过箱线图 + T 检验分析了 HBeAg、HBeAb 不同状态组的指标差异；过程中通过 `pivot_longer()` 解决了数据格式不匹配的问题，并采用 Bonferroni 校正控制多组比较的假阳性风险，确保了检验结果的可靠性。

最后，针对非正态分布的连续变量（如 LSM），利用小提琴图合并箱线图的方式，同时呈现了数据的分布形态与统计摘要；通过温莎化处理离群值、结合 Wilcoxon 秩和检验（BH 校正），兼顾了数据的稳健性与组间差异的统计学验证。

本章的方法体系既满足了基因流行病学研究中“可视化呈现数据特征”的需求，也通过对应的统计检验实现了“差异的量化验证”，为后续的病因关联分析、风险分层等研究提供了基础的数据支撑；同时，代码中对常见报错（如数据行数不匹配）的解决方法，也为实际数据分析中的问题排查提供了参考。

第 11 章 描述性统计：基础数据特征

关联分析前需先进行描述性统计，明确突变分布与临床表型的基础特征，常用指标与方法：

1) 分类变量（如突变状态：阳性 / 阴性；临床表型：急性 / 慢性、纤维化分期 F0~F4）：用频数（n）和构成比（%）描述，如“1200 例慢性乙肝患者中，核心启动子双突变阳性 420 例（35.0%）”。

2) 连续变量（如病毒载量、ALT 水平、病程年限）：用均数 $\pm$ 标准差（ $\bar{x} \pm s$ ）或中位数（四分位数）[M（Q1, Q3）] 描述，如“双突变阳性患者的平均病程为 8.5 ± 3.2 年，野生型患者为 5.2 ± 2.1 年”。

3) 组间比较：分类变量用 χ^2 检验或 Fisher 精确检验（样本量 < 40 或理论频数 < 5 时）；连续变量用 t 检验（正态分布）或 Mann-Whitney U 检验（非正态分布），如“双突变阳性与野生型患者的纤维化分期分布差异有统计学意义（ $\chi^2=45.2$, $P < 0.001$ ）”。

11.1 基线数据统计

通过正态检验，连续变量基础统计学数据都不符合正态分布。这种情况下，Kruskal-Wallis H 检验用于比较组间非正态分布的连续变量。单因素方差分析用于比较服从正态分布的连续变量，SNK-q 检验用于组间比较。t 检验用于比较两组连续变量。Kruskal-Wallis H 检验用于比较组间非正态分布的连续变量，Mann-Whitney U 检验用于比较两组非连续或非正态分布变量。使用 χ^2 检验比较两组分类变量。连续变量表示为（ $\bar{x} \pm s$ ）或中位数（P25-P75），差异在 $P < 0.05$ 时被认为具有统计学意义。

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
	ID	Age	Age_grc	Genoty	Subtyp	学历	职业	HBV_DNA	母亲HE	母亲HE	母亲HE	母亲HE	母亲HE	乙肝五	服药史	既往孕	出生孕	生产方	婴儿性	婴儿体	传播分
4	3	29.27222	2	B	adw2	5	3	74.4	1	0	0	1	1	2	0	0	0	2	0	0	1
5	4	27.33056	2	C	adrq+	4	5	20300000.0	1	0	1	0	1	1	1	1	0	2	1	0	1
6	5	29.85	2	C	adrq+	5	5	12700000.0	1	0	1	0	1	1	0	1	0	2	1	1	1
7	9	21.80556	1	C	adrq+	2	2	225.0	1	0	0	1	1	2	0	0	0	1	0	0	1
8	12	28.75278	2	C	adrq+	2	4	2040000.0	1	0	1	0	1	1	0	0	0	1	1	0	1
9	14	28.525	2	B	adw2	5	1	5670000.0	1	0	1	0	1	1	0	1	0	2	0	0	1
10	17	27.73333	2	B	adw2	5	5	139.0	1	0	0	1	1	2	0	0	0	1	0	0	1
11	18	36.71889	4	C	adrq+	4	1	1200.0	1	0	0	1	1	2	0	1	0	2	1	0	1
12	20	26.03889	2	C	adrq+	5	1	88.1	1	0	0	1	1	2	0	1	0	1	1	0	1
13	22	28.48056	2	B	adw2	5	4	60.6	1	0	0	1	1	2	0	0	0	1	0	0	1
14	23	30.47222	2	B	adw2	4	4	214.0	1	0	0	1	1	2	0	1	0	1	1	0	1
15	24	32.13056	3	C	adrq+	2	5	18.5	1	0	0	1	1	2	0	1	0	1	1	1	2
16	25	25.99444	2	C	adrq+	5	1	97.6	1	0	0	0	1	3	0	0	0	2	0	1	1
17	26	36.69167	4	C	adrq+	3	5	352.0	1	0	0	1	1	2	0	1	0	1	0	0	2
18	28	26.88889	2	C	adrq+	3	5	48600000.0	1	0	1	0	1	1	0	1	0	2	1	0	3

图 . 基础临床信息表 (XA).xlsx

```
# 导入必要的库
library(readxl) # 用于读取 Excel 文件
library(dplyr) # 用于数据处理
library(ggpubr) # 用于统计检验

# 读取数据
data <- read_excel(“保存地址 / 基础临床信息表 (XA).xlsx” ,1)

# 替换空格和 #N/A 为 NA
data[data == “” ] <- NA
data[data == “#N/A” ] <- NA

# 使用基础 R 包 na.omit 删除缺失值
cleaned_data <- na.omit(data[, c(“Age”, “HBV_DNA”, “Genotype”, “Subtype”,
“乙肝五项” , “传播分组” )])

# 描述性统计
descriptive_stats <- cleaned_data %>%
group_by( 传播分组 ) %>%
summarise(
Age_median = median(Age, na.rm = TRUE),
Age_IQR = IQR(Age, na.rm = TRUE),
HBV_DNA_median = median(HBV_DNA, na.rm = TRUE),
HBV_DNA_IQR = IQR(HBV_DNA, na.rm = TRUE)
```

)

统计分类变量的频数和百分比

```
genotype_stats <- cleaned_data %>%
  group_by( 传播分组 , Genotype) %>%
  summarise(Genotype_count = n()) %>%
  mutate(Genotype_percent = Genotype_count / sum(Genotype_count) * 100)
```

```
subtype_stats <- cleaned_data %>%
  group_by( 传播分组 , Subtype) %>%
  summarise(Subtype_count = n()) %>%
  mutate(Subtype_percent = Subtype_count / sum(Subtype_count) * 100)
```

```
乙肝五项_stats <- cleaned_data %>%
  group_by( 传播分组 , 乙肝五项 ) %>%
  summarise( 乙肝五项_count = n()) %>%
  mutate( 乙肝五项_percent = 乙肝五项_count / sum( 乙肝五项_count) * 100)
```

Kruskal-Wallis H 检验

```
kruskal_test_age <- kruskal.test(Age ~ 传播分组 , data = cleaned_data)
kruskal_test_hbv_dna <- kruskal.test(HBV_DNA ~ 传播分组 , data = cleaned_
data)
```

Mann-Whitney U 检验（比较传播分组 1 和 2）

```
mann_whitney_age <- wilcox.test(Age ~ 传播分组, data = cleaned_data[cleaned_
data$ 传播分组 %in% c(1, 2), ])
mann_whitney_hbv_dna <- wilcox.test(HBV_DNA ~ 传播分组 , data = cleaned_
data[cleaned_data$ 传播分组 %in% c(1, 2), ])
```

卡方检验

```
# Genotype 的卡方检验
genotype_table <- table(cleaned_data$Genotype, cleaned_data$ 传播分组 )
genotype_chi_square <- chisq.test(genotype_table)

# Subtype 的卡方检验
subtype_table <- table(cleaned_data$Subtype, cleaned_data$ 传播分组 )
subtype_chi_square <- chisq.test(subtype_table)

# 乙肝五项的卡方检验
乙肝五项_table <- table(cleaned_data$ 乙肝五项 , cleaned_data$ 传播分组 )
乙肝五项_chi_square <- chisq.test( 乙肝五项_table)

# 汇总检验结果
chi_square_results <- data.frame(
  Variable = c( "Genotype" , "Subtype" , "乙肝五项" ),
  Statistic = c(genotype_chi_square$statistic, subtype_chi_square$statistic, 乙肝五
项_chi_square$statistic),
  P_Value = c(genotype_chi_square$p.value, subtype_chi_square$p.value, 乙肝五
项_chi_square$p.value)
)

# 输出描述性统计
print(descriptive_stats)
write_xlsx(descriptive_stats, "保存地址 /descriptive_result.xlsx" )

# 输出分类变量的统计
print(genotype_stats)
print(subtype_stats)
print( 乙肝五项_stats)
write_xlsx(genotype_stats, "保存地址 /descriptive_genotype.xlsx" )
```

```

write_xlsx(subtype_stats, “保存地址 /descriptive_subtype.xlsx” )
write_xlsx( 乙肝五项_stats, “保存地址 /descriptive_ 乙肝五项.xlsx” )

# 输出检验结果
print(kruskal_test_age)
print(kruskal_test_hbv_dna)
print(mann_whitney_age)
print(mann_whitney_hbv_dna)
print(chi_square_results)

# 汇总检验结果
ST_results <- data.frame(
  Test = c(“Kruskal-Wallis Age”, “Kruskal-Wallis HBV_DNA”, “Mann-Whitney
Age”, “Mann-Whitney HBV_DNA” ),
  Statistic = c(kruskal_test_age$statistic, kruskal_test_hbv_dna$statistic, mann_
whitney_age$statistic, mann_whitney_hbv_dna$statistic),
  P_Value = c(kruskal_test_age$p.value, kruskal_test_hbv_dna$p.value, mann_
whitney_age$p.value, mann_whitney_hbv_dna$p.value)
)
write_xlsx(ST_results, “保存地址 /ST_results.xlsx” )

```

表 1 HBV S 基因型的 246 名孕妇 HBV 基因型和人口统计学特征

统计项目	传播分组			κ^2	P 值
	NBT	OBT	DBT		
年龄	29.3(3.5)	29.0(3.6)	29.5(3.3)	0.87 ^a	0.646
HBV-DNA	1.88×10^2 (6.86×10^3)	7.37×10^2 (1.84×10^6)	8.86×10^6 (5.21×10^7)	16.69 ^a	< 0.001
基因型				8.54 ^b	0.073
B	11(10.8%)	19(21.6)	6(26.1%)		
C	82(80.4%)	67(76.6%)	16(69.6%)		
D	9(8.8%)	2(2.3%)	1(4.3%)		
亚型				12.85 ^b	0.117

续表

统计项目	传播分组			κ^2	P 值
	NBT	OBT	DBT		
adr	6(5.9%)	1(1.1%)	0		
adrq+	69(67.6%)	65(73.9%)	13(56.5%)		
adw2	18(17.6%)	20(22.7%)	9(39.1%)		
ayw1	1(1.0%)	0	0		
ayw2	8(7.8%)	2(2.3%)	1(4.3%)		
血清标志物				11.45 ^b	0.022
HBsAg+/ HBcAg+/ HBcAb+	26(25.5%)	33(37.5%)	14(60.9%)		
HBsAg+/ HBcAb+ /HBcAb+	60(58.8%)	44(50.0%)	8(34.8%)		
HBsAg+/ HBcAb+	16(15.7%)	11(12.5%)	1(4.3%)		

注：计量资料中，括号外数据为中位数，括号内数据为四分位数间距；分类资料中，括号外数据为例数，括号内数据为百分比^a：Kruskal-Wallis H检验；^b：卡方检验

11.2 森林图

数据表如下：

	B	C	D	E	F	G	H	I
1	Frequency	Rate	χ^2	Pvalue	RR(95%CI)			
2	240					1.195	0.846	1.921
3								
4								
5	42	45.65	6.661	0.010	1.557(1.129, 2.146)	1.557	1.129	2.146
6	8	8.70	0.091	0.762	0.851(0.494, 1.466)	0.851	0.494	1.466
7	42	45.65	5.582	0.018	0.665(0.486, 0.91)	0.665	0.486	0.91
8								
9	19	20.65	3.054	0.081	0.678(0.473, 0.971)	0.678	0.473	0.971
10	71	77.17	0.486	0.486	1.183(0.817, 1.713)	1.183	0.817	1.713
11	2	2.17	1.636	0.201	2.368	2.368	2.368	2.368
12								
13								
14	18	62.07	0.182	0.670	0.799(0.395, 1.618)	0.799	0.395	1.618
15	1	3.45	0.258	0.612	2.27	2.27	2.27	2.27
16	10	34.48	0.026	0.871	1.061(0.517, 2.177)	1.061	0.517	2.177
17								
18	6	20.69	0.407	0.524	0.676(0.296, 1.545)	0.676	0.296	1.545
19	21	72.41	0.708	0.400	1.524(0.720, 3.277)	1.524	0.72	3.227
20	2	6.90	0.002	0.964	0.711(0.191, 2.643)	0.711	0.191	2.643
21								

代码如下：

```
## 基础森林图测试
```

```
library(xlsx)
```

```
library(forestplot)
```

```
library(meta)
```

```
library(extrafont)
```

```
fp<-read.xlsx(“文件地址 / 文件名.xlsx”,1,header=FALSE)
```

```
head(fp)
```

```
forestplot(labeltext=as.matrix(fp[,1:3]),
```

```
mean=fp$X4,lower=fp$X5,upper=fp$X6,
```

```
is.summary=c(T,T,T,F,F,T,F,F,T,F,F),
```

```
zero=1,boxsize=0.3,lineheight=unit(5,' mm' ),colgap=unit(4,' mm' ),
```

```
lwd.zero=2,lwd.ci=1.5,
```

```
col=fpColors(box = '#ABC56E' ,summary = '#B55489' ,lines =  
‘black’ ,zero = ‘#3FA0C0’ ),xlab = ‘RR estimates’ )|>
```

```
fp_add_lines(h_1=gpar(lwd=2),h_3 = gpar(lty = 2),
```

```
h_14 = gpar(lwd = 1, columns = 1:3, col = “#000044” ))
```

```
# END
```

```
## 添加缩进测试
```

```
fp1_mod <- fp1
```

```
# 指定要缩进的行号
```

```
rows_to_indent <- c(5, 6, 7, 9, 10, 11, 14, 15, 16, 18, 19, 20)
```

```
# 对这些特定的行添加缩进
```

```
for (row in rows_to_indent) {
```

```
fp1_mod[row, 1:6] <- paste( “ “, fp1_mod[row, 1:6]) # “ “内空格就是缩  
进量
```

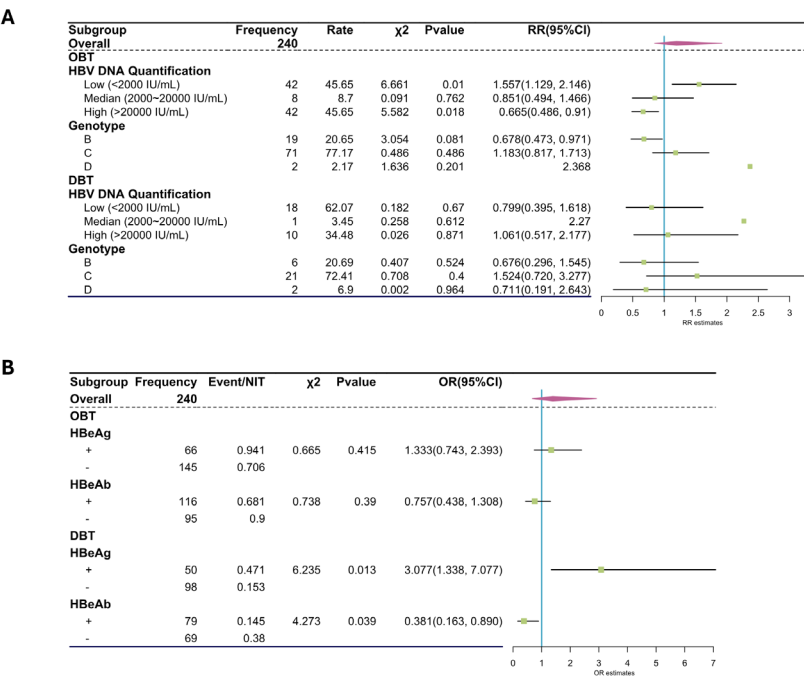
```
}
```

```
# 更新 forestplot 的调用以使用修改过的数据
plot2<-forestplot(labeltext=as.matrix(fp1_mod[,1:6]),
                  mean=fp1_mod$X7,lower=fp1_mod$X8,upper=fp1_mod$X9,
                  is.summary=c(T,T,T,T,F,F,T,F,F,T,T,F,F,T,F,F,F),
                  zero=1,boxsize=0.3,lineheight=unit(5,' mm' ),colgap=unit(4
, ' mm' ),
                  lwd.zero=2,lwd.ci=1.5,txt_gp = fpTxtGp(
                  ticks = gpar(cex = 0.8)),
                  col=fpColors(box = '#ABC56E' ,summary = '#B55489' ,lines =
'black' ,zero = '#3FA0C0' ),xlab = 'RR estimates' )|>
  fp_add_lines(h_1=gpar(lwd=2),h_3 = gpar(lty = 2),
              h_21 = gpar(lwd = 2, columns = 1:6, col = "#000044" ))

print(plot2)
# END

## 按照结局分类的 OR 值和卡方检验森林图
fp2<-read.xlsx(“地址 / 文件名 .xlsx” ,1,header=FALSE)
head(fp2)
# 指定要缩进的行号
rows_to_indent <- c(5, 6, 8, 9, 12, 13, 15, 16)
# 对这些特定的行添加缩进
for (row in rows_to_indent) {
  fp2[row, 1:6] <- paste( “ “, fp2[row, 1:6])
}
# 更新 forestplot 的调用以使用修改过的数据
plot3<-forestplot(labeltext=as.matrix(fp2[,1:6]),
                  mean=fp2$X7,lower=fp2$X8,upper=fp2$X9,
                  is.summary=c(T,T,T,T,F,F,T,F,F,T,T,F,F,T,F,F,F),
                  zero=1,boxsize=0.3,lineheight=unit(5,' mm' ),colgap=unit(4
```

```
, ' mm' ),  
  
lwd.zero=2,lwd.ci=1.5,txt_gp = fpTxtGp(  
  ticks = gpar(cex = 0.8)),  
  col=fpColors(box = '#ABC56E' ,summary = '#B55489' ,lines =  
'black' ,zero = '#3FA0C0' ),xlab = 'OR estimates' )|>  
  fp_add_lines(h_1=gpar(lwd=2),h_3 = gpar(lty = 2),  
    h_17 = gpar(lwd = 2, columns = 1:6, col = "#000044" ))  
print(plot3)
```



本章小结

本章聚焦基因流行病学关联分析前的基础数据特征刻画，系统阐述了描述性统计的核心方法、实现流程及可视化呈现方式，为后续深入的关联分析奠定了数据基础：

首先，明确了描述性统计的核心框架：针对分类变量（如基因型、临床表型、传播分组）采用频数与构成比描述分布特征，针对连续变量（如年龄、HBV-DNA 水平）结合分布类型选择中位数（四分位数）或均数 $\pm$ 标准差表征集中趋势与离散程度；组间比较则根据变量类型和分布特征匹配对应的统计方法——分类变量采用 χ^2 检验（或 Fisher 精确检验），正态分布连续变量采用 t 检验 / 单因素方差分析，非正态分布连续变量采用 Kruskal-Wallis H 检验（多组）或 Mann-Whitney U 检验（两组），并以 $P < 0.05$ 作为统计学差异的判定标准。

其次，通过实战代码实现了基线数据的完整统计分析：从 Excel 数据读取、缺失值处理，到分传播分组计算连续变量的中位数与四分位数间距、分类变量的频数与百分比，再到针对性的统计检验（Kruskal-Wallis H 检验分析多组连续变量差异、 χ^2 检验分析分类变量分布差异），并将统计结果导出为结构化表格，形成了“数据清洗 - 统计描述 - 差异检验 - 结果输出”的标准化流程，解决了非正态分布连续变量的统计分析痛点。

最后，介绍了森林图在统计结果可视化中的应用：通过 forestplot 包实现了 RR/OR 值及其置信区间的可视化呈现，支持自定义缩进、颜色、线条样式等格式，可直观展示不同因素（如基因型、临床表型）与结局的关联强度及统计学显著性，弥补了纯数值表格的信息展示短板。

本章的方法体系既遵循了生物统计学的核心原则，又结合乙肝病毒研究的实际数据特征（如 HBV-DNA 非正态分布）优化了分析策略，统计结果的表格化与森林图可视化相结合，既满足了学术报告的规范要求，也提升了数据特征的可读性，为后续的基因 - 表型关联分析、风险因素筛选提供了可靠的基础数据支撑。

第五部分 病毒进化的方向和速度

病毒作为一类非细胞型微生物，其分子进化呈现出与细胞生物截然不同的鲜明特征，核心驱动力源于高突变率、短世代周期和庞大的种群规模。多数病毒尤其是 RNA 病毒，每轮复制的基因组点突变率可达每个基因组至少 1 个突变，这种高频突变与自然选择的结合，使其能快速适应宿主环境变化。同时，病毒繁殖能力极强，单次感染可产生大量子代病毒，确保突变基因能快速传递并在种群中扩散。

病毒的遗传物质类型（双链 DNA、双链 RNA、单链 DNA、单链 RNA）会影响突变率和进化速率，例如 RNA 病毒缺乏严格的复制纠错机制，突变率显著高于 DNA 病毒，但总体而言，所有类型病毒均具备较高的变异潜能。这种快速进化特性不仅是病毒适应宿主免疫压力、实现跨物种传播的关键，也给疫苗研发和抗病毒药物治疗带来巨大挑战，即耐药突变可能在治疗开始后数月甚至数周就会出现。

第 12 章 系统发育和分子进化

系统发育分析是通过比较病毒基因组或蛋白质序列的同源性，重建病毒的进化历史并揭示其亲缘关系的方法，核心产物是系统发育树（简称进化树）。进化树由节点和分支组成，节点代表祖先病毒或现存病毒，分支长度反映进化距离（通常以核苷酸替换数或替换率表示）。在病毒分子流行病学中，系统发育分析的核心应用包括：重建病毒传播链、识别暴发源头、追踪变异株扩散路径、评估病毒进化速率等。其分析流程通常包括五个关键步骤：多序列比对（MSA）、系统发育推断、树根确定、分歧时间估算和传播聚类分析，每个步骤的准确性均依赖于前序步骤的可靠性。

序列进化模型是系统发育分析的理论基础，用于描述核苷酸或氨基酸序列随时间发生替换的概率规律。病毒系统发育分析中最常用的是广义时间可逆模型

(GTR)，该模型假设核苷酸替换是时间可逆的，且允许不同核苷酸之间的替换率存在差异，是最通用的中性、独立、有限位点序列进化模型，其他多种可逆替换模型均可视为 GTR 模型的特例。

除 GTR 模型外，分析中还需考虑位点速率异质性。不同位点的进化速率可能存在差异，常用伽马分布 (Gamma distribution) 或自由速率模型 (FreeRate model) 来描述这种异质性；同时部分位点可能在进化过程中保持不变，需引入不变位点比例 (proportion of invariable sites) 参数进行校正。这些模型参数的合理选择直接影响系统发育推断的准确性。

为量化系统发育分析结果的准确性，需采用标准化评估指标，核心包括：

1) 均方误差 (Mean Squared Error)：衡量估计的成对距离矩阵与真实距离矩阵的差异，直接影响传播聚类工具的准确性，是评估多序列比对工具适用性的重要指标。

2) 曼特尔相关系数 (Mantel Correlation)：包括皮尔逊相关系数和斯皮尔曼相关系数，用于评估估计距离矩阵与真实距离矩阵的相关性，取值范围为 -1 至 1，1 表示完全相关。

3) 罗宾逊 - 福尔兹距离 (Robinson-Foulds Distance, RF Distance)：分为未加权 (URF) 和加权 (WRF) 两种，用于衡量两棵未根树的拓扑结构差异，数值越低表示拓扑结构越相似。

4) 成对得分 (SP Score)：衡量推断比对结果与真实比对结果中共享同源位点对的比例，数值越高表示比对准确性越高。

5) 总列得分 (TC Score)：表示成功比对的列占总比对列的比例，取值范围为 0 至 1，1 表示完全准确比对。

6) 压缩因子 (Compression Factor)：推断比对结果的列数与真实比对结果列数的比值，理想值为 1。

12.1 进化树构建核心原理（距离法 / 最大似然法）

进化树（系统发育树）是通过量化不同序列间的进化距离，模拟生物大分子（如 HBV 基因序列）进化历程的树状结构，核心原理是“同源序列具有共同祖先，序列差异越小，进化关系越近”。常用构建方法包括距离法（如邻接法 NJ）和

最大似然法 (ML)，本节通俗拆解两种方法的核心逻辑、适用场景及优缺点，为工具选择和结果解读奠定基础。

12.1.1 核心概念

1) 进化距离：衡量两条 HBV 序列间的变异程度，通常用“每 100 个碱基的替换数”表示（单位：%），序列差异越大，进化距离越远（如野生型与耐药突变株的进化距离通常 $> 2\%$ ，同一传播链毒株的进化距离 $< 0.5\%$ ）。

2) 树状结构组成：① 节点：代表进化过程中的祖先序列（内部节点）或待分析的目标序列（末端节点，即叶节点）；② 分支：连接节点的线段，分支长度代表进化距离（线段越长，进化差异越大）；③ 分支置信度（自举值）：评估分支可靠性的指标，数值越高（0~100），分支代表的进化关系越可靠（通常 ≥ 70 为可信）。

3) 拓扑结构：进化树的分支排列方式，反映序列间的进化亲缘关系（如“聚为一支”表示进化关系最近，可能来源于共同祖先）。

12.1.2 距离法（以邻接法 NJ 为例）核心原理

距离法是最常用的进化树构建方法，核心逻辑是“先计算所有序列间的进化距离，再通过聚类将进化距离最近的序列优先聚为一支，逐步构建完整进化树”，一般步骤如下：

1) 计算距离矩阵：对所有待分析的 HBV 序列（如 20 条含耐药突变的 P 区序列），两两计算进化距离，生成“距离矩阵”（行和列均为序列名称，矩阵元素为对应两条序列的进化距离）。

2) 初始聚类：从距离矩阵中找到进化距离最近的两条序列（如 Sample1 和 Sample2，距离 0.3%），将其聚为一个临时分支，视为一个“复合节点”。

3) 更新距离矩阵：计算该复合节点与其他所有序列的平均进化距离，用复合节点替换原来的两条序列，更新距离矩阵。

4) 重复聚类：重复步骤 13-15，每次将进化距离最近的节点（单个序列或复合节点）聚为一支，直至所有序列都被纳入树中，形成完整进化树。

适用场景：适用于序列数量较多（ ≤ 100 条）、序列长度较短（ $\leq 1000\text{bp}$ ）的 HBV 样本分析（如临床批量耐药突变株的进化关系初步分析）；优势是计算

速度快、操作简单；不足是对序列进化速率不一致的样本，结果准确性可能下降。

12.1.3 最大似然法（ML）核心原理

最大似然法是目前准确性最高的进化树构建方法，核心逻辑是“基于特定的进化模型，计算所有可能的进化树拓扑结构出现的概率，选择概率最大的拓扑结构作为最终进化树”，通俗理解为“从所有可能的进化路径中，找到最符合当前序列变异特征的路径”。

关键特点：① 依赖进化模型：需提前选择适配 HBV 序列的进化模型（如 K2P、GTR+G 模型），模型越贴合序列实际进化特征，结果越准确；② 计算复杂度高：需遍历大量可能的拓扑结构，计算速度远慢于距离法；③ 结果可靠性高：对序列进化速率不一致、存在复杂变异（如插入/缺失）的样本，准确性优于距离法。

适用场景：适用于序列数量较少（ ≤ 50 条）、需要精准解析进化关系的场景（如爆发疫情的溯源分析、关键突变株的起源探索）；是科研论文中推荐使用的进阶方法。

12.2 常用建树工具使用方法

进化树构建工具的选择需结合分析目标和样本量，MEGA 工具操作简单、可视化程度高，适合初学者和批量样本的初步分析；PhyML 工具计算速度快、准确性高，支持最大似然法的进阶分析；iq-Tree 适合绘制 ML 树，软件自动探索最适模型参数，操作简便。BEAST 工具是绘制贝叶斯法系统发育树的经典工具。MEGA 和 iq-Tree 建树方法前文已述，这里详解 PhyML 软件和 MrBayes 软件的操作步骤。

12.2.1 PhyML 最大似然法 ML 树构建

PhyML 支持在线和本地客户端两种使用方式，在线版无需安装，适合初学者；本地版计算速度更快，适合大样本分析。本节以在线版为例，实操步骤如下：

1) 工具准备：打开浏览器，进入 PhyML 在线平台（<https://www.atgc-montpellier.fr/phyml/>），选择“Standard PhyML”（标准版本）。

2) 上传比对文件：点击“Choose File”，上传 MEGA 生成的 ALN 格式比对

文件(已删除空白列和低质量区域);在“Data Type”中选择“DNA”,点击“Next”。

3) 进化模型选择: 点击“Automatic model selection”(自动模型选择), 选择“Smart Model Selection (SMS)”, 设置“Substitution matrix”为“All”, 点击“Compute”, 平台会自动筛选适配当前序列的最优进化模型(如 GTR+G 模型)。

4) 模型确认: 模型筛选完成后, 点击“Next”, 确认模型参数(如 GTR 模型的碱基替换率、Gamma 分布参数等), 默认即可。

5) 构建参数设置: Bootstrap analysis(自举检验): 选择“Non-parametric bootstrap”, 设置“Number of bootstrap replicates”=1000。Tree search options(树搜索选项): 选择“Best of NNI & SPR”(结合邻域交换和子树剪切重接算法, 提升拓扑结构准确性), 其他参数默认。

6) 执行构建与结果下载: 点击“Submit”, 平台开始执行最大似然法构建, 等待 10~30 分钟(根据序列数量和模型复杂度调整)。

7) 结果查看与下载: 构建完成后, 页面会显示进化树的拓扑结构和分支置信度; 点击“Download”, 选择“Newick”格式(用于后续美化)和“PNG”格式(直接查看), 完成下载。

8) 进化树美化: 将 Newick 格式文件导入 MEGA 或 FigTree 软件, 调整分支颜色、字体大小、置信度显示位置等, 提升可读性。

12.2.2 MrBayes 贝叶斯法树构建

BEAST 是一个软件集群, 包含 BEAST、BEAUti、FigTree、LogCombiner、TreeAnnotator 等, 主要用于建立基于贝叶斯法的最大可信进化分枝(Maximum Clade Credibility Tree, MCC) 树。从官网(https://beast.community/getting_started)下载 BEAST 软件, 准备好需要建树的序列文件, 这里用的是之前的 RefSeqSet0 截取的 S 区序列。打开 BEAUti 软件:

1) 导入序列文件: 打开 RefSeqSet0.fas, 该软件区分 fas 和 fasta 格式, 用 fas 格式比较稳定。

2) 文件总览(Partitions 栏): 查看核对数据信息。

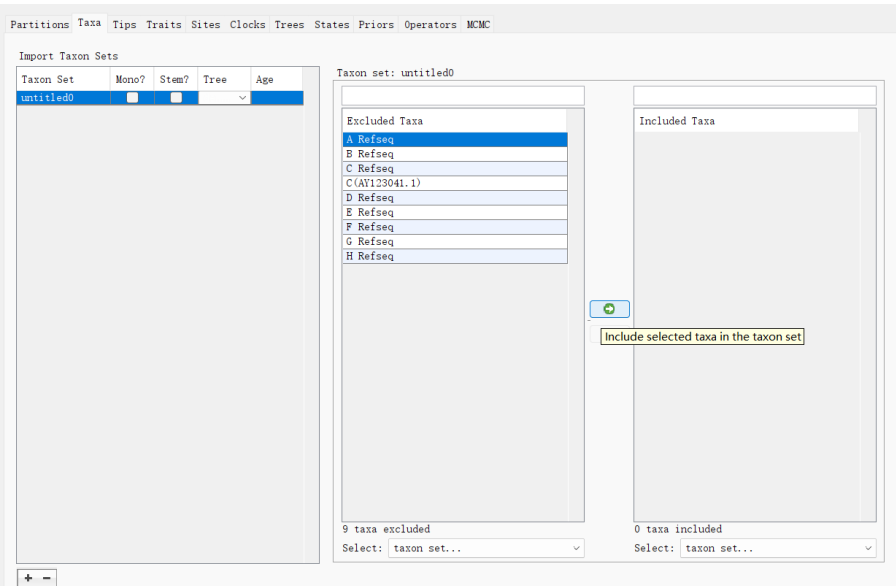
3) 节点校准(Taxa 栏): 通俗而言是建立时间“锚点”, 对于考虑时间的系统发育树, 这一步很重要。这种锚点可以是分支或谱系, 核心在于它们具有已知的共同祖先分化时间。在 Taxa 界面可以通过两种方式建立锚点:

续表

a) 建树集中包含“锚点”分支，点击左下“+”号，选中建树集，选中该分支，点击图中高亮的右箭头，将其移入“Included Taxa”。

b) 引入另外的“锚点”分支，点击左上方“Import Taxon Sets”

左栏 Mono 模式，校准点位于共同祖先分化时间；Stem 模式，校准点位于共同祖先出现时间。



4) 校准端点（Tips 栏）：对于病毒类对象，可以使用采集时间进行分子钟端点校准。勾选“Use tip dates”解锁编辑。在 Date 输入日期，格式为“年. 十进制月份”。也可以在框出的 Parse Dates 设置自动读取时间，前提是已经将每条序列名字加上时间后缀“_YYYY.MM”。软件自动将采集时间最接近当下的序列设为“Height”值原点，自动计算每条序列相对于原点的年龄高度。

☒ Use tip dates			
Parse Dates Import Dates Set Dates Clear Dates Set Uncertainty			
Dates as: Years		Since some time in the past	☐ Specify origin date: unable to parse date
Name	Date	Uncertainty	Height
C Refseq	2025.8	0.0	0.0
A Refseq	2025.6	0.0	0.20000000000004547
F Refseq	2023.4	0.0	2.39999999999998636
B Refseq	2023.1	0.0	2.70000000000000455
D Refseq	2017.3	0.0	8.5
H Refseq	2016.4	0.0	9.3999999999999864
E Refseq	2014.3	0.0	11.5
G Refseq	2012.2	0.0	13.5999999999999909
C (AY123041.1)	2012.1	0.0	13.7000000000000045

5) 分区进化模型设置（Traits 栏）对于非编码区和编码区合并的情形适用，本例跳过。

6) 进化树模型设置

数据类型	替换模型 Substitution model	碱基频率 Base frequencies	位点异质性模型 Site heterogeneity	密码子相关模型
核基因 (非编码)	HKY/TN93	Estimated	Gamma + invariant sites	OFF
核基因 (编码, 近缘)	HKY	Estimated	Gamma + invariant sites	Yang96 模型
核基因 (编码, 远缘)	GTR	Estimated	Gamma + invariant sites	SRD06 模型
线粒体 DNA	TN93	Estimated/empirical	Gamma + invariant sites	看情况 选择
短序列 / 高相似序列	JC/HKY	仅 JC 选择 all equal	None/ invariant sites	OFF
病毒 DNA 保守区	HKY	Estimated	invariant sites	OFF
病毒 DNA 近缘	HKY/TN93	Estimated	Gamma + invariant sites	Yang96 模型
病毒 DNA 远缘	GTR	Estimated	Gamma + invariant sites	SRD06 模型

7) 分子钟模型设置 (Sites 栏) 主要有四个选择, 不同模型的适用场景与核心假设存在显著差异。于 HBV S 基因的进化速率在不同毒株 (比如不同基因型、不同传播链) 之间通常存在差异, 严格分子钟的 “全局均一速率” 假设基本不适用, 直接选择宽松模型。这里对四类模型简要介绍:

a) 严格分子钟 (Strict Clock): 模型的核心假设为全局进化速率均一, 即系统发育树中所有分支的碱基替换速率不存在差异。在 BEAST 的操作流程中, 严格分子钟的校准可通过两种方式实现: 一是在 “Priors” 菜单栏中直接设置 “clock.rate” 参数, 指定已知的 HBV S 基因变异速率; 二是对 “tmrca” 参数赋予合理先验分布, 或直接设定 “treeModel.rootHeight” 的数值范围, 以此约束进化时间的推断。

b) 宽松分子钟 (Uncorrelated Relaxed Clock): 严格分子钟不同, 宽松分子钟允许树中不同分支的进化速率存在差异, 且假定各分支的速率与祖先分支之间无先验相关性, 属于 “非相关” 速率模型。根据速率分布特征的不同, 宽松分子钟可进一步分为对数正态分布 (Lognormal)、指数分布 (Exponential) 与 Gamma 分布三种亚型。

c) 随机局部分子钟 (Random local Clock)：模型的核心思想是将系统发育树划分为若干个“速率区域”，区域内的分支共享同一进化速率，而区域间的速率则存在随机差异，适用于存在速率异质性但异质性模式无明确规律的 HBV S 基因数据集。

d) 固定局部分子钟 (Fixed local Clock)：定局部分子钟同样基于“速率区域”的划分，但区域的界定需依赖先验生物学信息（如 HBV 的基因型分类、宿主来源等），模型会根据预设的区域划分，分别估计不同区域的进化速率，适用于已有明确分组依据的 HBV S 基因分析。

8) 分子钟选择方法主要有三种：

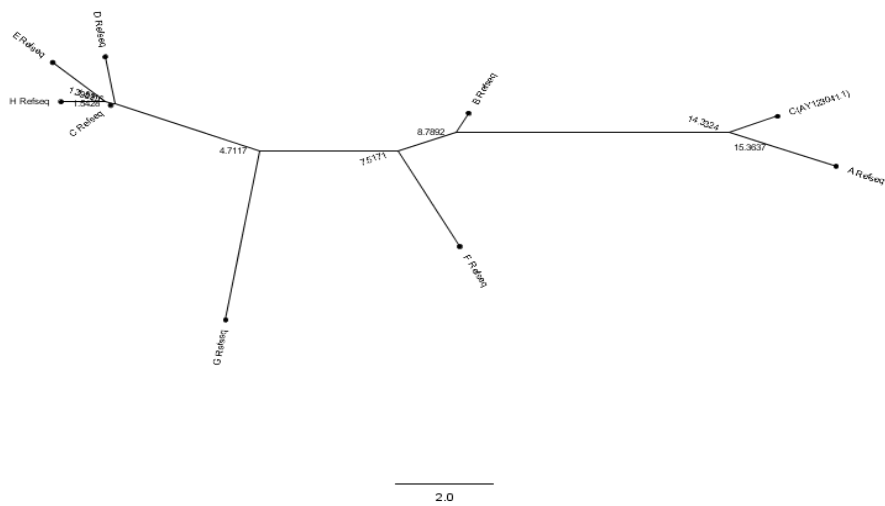
a) 基于变异系数的快速判断：先构建基于非相关宽松对数正态分子钟的分析流程，采用默认参数配置生成 XML 文件，设置较短的 MCMC 链长完成预分析。分析结束后，将输出的日志文件 (.log) 导入 Tracer 软件，重点查看“coefficientOfVariation”（速率变异系数）的估计值。该参数的核心意义为分支间速率的变异程度，若数值趋近于 0，说明 HBV S 基因的序列进化速率在各分支间差异极小，满足严格分子钟的假设；若数值显著大于 0，则提示速率异质性较高，更适合采用宽松分子钟模型。

b) 基于贝叶斯因子的模型比较：贝叶斯因子 (Bayes Factor, BF) 的核心原理是比较不同分子钟模型与树先验组合的边际似然值 (Marginal Likelihood)，以此评估模型对 HBV S 基因数据的拟合优度。操作时，需分别运行严格分子钟与宽松分子钟的分析流程，获得两种模型的日志文件后，将其同时导入 Tracer 软件。在软件中选中待比较的模型文件，依次点击菜单栏的“Analysis—Model Comparison”，将分析类型 (Analysistype) 设置为“Harmonic mean”，重复次数 (Reps) 设为 100，运行后即可得到贝叶斯因子的估计值。一般认为，当 BF 值 ($BF = \Pr(D|M1) / \Pr(D|M2)$) ≥ 3 时，说明模型 M1 对 HBV S 基因数据的拟合效果显著优于模型 M2，该方法可同时实现分子钟模型与树先验模型的最优筛选。

c) 基于路径采样 (path sampling) 法通过构建参数空间的连续路径，逐步积分得到边际似然值，比贝叶斯因子法对边际似然值的估计更为精准，能够有效降低模型比较的误差。该方法适用于对模型选择精度要求较高的基因进化分析

9) 点击右下角“Generate BEAST File”生成 XML 文件。

10) 打开 BEAST 软件，导入 XML 文件，点击“Run”开始运行。



本章小结

本章围绕病毒系统发育与分子进化的核心理论、方法及工具展开,构建了“理论基础 - 核心方法 - 工具实操”的完整知识体系,为病毒分子流行病学研究中的进化分析提供了标准化技术框架:

首先,明确了系统发育分析的核心定位与理论支撑。系统发育分析通过解析病毒序列同源性重建进化树,核心应用涵盖传播链重建、暴发溯源、变异株追踪等,其分析流程(多序列比对、系统发育推断、树根确定、分歧时间估算、传播聚类)具有严格的步骤依赖性。序列进化模型是分析的核心理论基础,广义时间可逆模型(GTR)作为通用模型,结合位点速率异质性描述(伽马分布/自由速率模型)和不变位点校正,可精准刻画病毒序列的进化规律;而均方误差、曼特尔相关系数、罗宾逊-福尔兹距离等标准化指标,则为分析结果的准确性提供了量化评估依据。

其次,系统拆解了两类核心进化树构建方法的原理与适用场景。距离法(以邻接法 NJ 为例)基于序列间进化距离的聚类分析,具有操作简便、计算快速的优势,适用于大样本量、短序列的初步进化关系分析;最大似然法(ML)则基于进化模型筛选概率最优的拓扑结构,准确性更高,是科研中精准解析进化关系(如疫情溯源、关键突变株起源)的首选方法。两种方法的对比的呈现,为研究

者根据样本特征和分析目标选择适配方法提供了明确指导。

最后，详细阐述了主流建树工具的实操流程。针对不同分析需求，分别介绍了 PhyML（在线版）的最大似然树构建步骤，包括比对文件上传、进化模型自动筛选、构建参数设置及结果下载美化；重点详解了 BEAST 软件的贝叶斯法分析流程，涵盖节点校准、端点校准、进化模型选择、分子钟模型（严格 / 宽松 / 随机局部 / 固定局部）的适配原则与选择方法，尤其针对 HBV 序列的特性给出了模型参数配置建议。

的方法体系既兼顾了理论深度与实操可行性，又结合乙肝病毒等具体研究对象的特征优化了技术细节，不仅帮助读者理解系统发育分析的核心逻辑，更能直接指导临床样本分析与科研实践，为病毒进化机制研究、流行病学溯源及防控策略制定提供了坚实的技术支撑。

第 13 章 环境压力和进化方向

病毒表现出高度异质的基因组结构和复制策略。这些基本特征与病毒的突变倾向相关，因此与病毒的遗传多样性和进化性相关。病毒通过突变和自然选择来适应环境，使其能够克服宿主防御，扩大宿主范围，并增加复制成功率。众所周知，突变具有少利多害性，对病毒而言也是如此，大多数蛋白质变化使病毒更难传播。一般而言，RNA 病毒每个细胞每个核苷酸产生 10^6 至 10^4 个新碱基取代，而 DNA 病毒的这些比率范围为 10^8 至 10^6 。但由于细胞基数大，即使在弱感染期间，病毒基因信息也会被复制数百万次。偶然情况下，突变可能会导致蛋白质发生有利于病毒种群增长的变化。具有有益突变的病毒将更有效地在细胞间传播，每个新感染的细胞复制病毒，产生更多具有相同优势的病毒，从而积累优势形状。

欲要阐明病毒基因组进化中的依赖性和选择性压力，需要了解病毒基因组特征。迄今为止，关于病毒基因组特征的信息很少。Holmudden 等^[1]证明，大多数病毒具有高度特异性的基因组特征，通常在同一家族中的物种之间也存在显著差异。他们的研究还揭示了病毒基因组特征与其宿主细胞之间的一致差异，尽管一些病毒表现出轻微的相似性，这可能是由于对其天然宿主的遗传适应所致。这一结果表明，显著的进化选择压力作用于病毒基因组，以塑造和保存其基因组特征，可能对合成生物学领域构建减毒活疫苗和病毒载体产生影响。

13.1 进化速率和种群动态分析

13.1.1 Modelgenerator

使用 Modelgenerator (v85) 工具分析氨基酸替代模型，结果显示最佳替代模型为 TVM+I+G。“I”表示某些位点从未经受任何进化上的改变，进化速率是相同的，“G”表示物种进化速度不同。使用 BEAST 程序进行分析。因 BEAST 不提供 TVM 替代模型，选择次一位的 GTR+I+G。分子钟选择 strict clock。树先验选择 Bayesian Skyline, Skyline Model 选择 Piecewise-constant。MCMC 显示在屏

幕上的马尔可夫链参数（Echo state to screen every）设为 5,000，记录在日志文件的（log parameters every）设为 300。Chain length = 3000 * 序列条数² = 3000 * 58² ≈ 10,100,000，有效量本量（ESS）>200，若不满足则意味着 traces 包含大量相关样本，可能不能很好地代表后验分布，需提高 chain length 以提高 ESS。

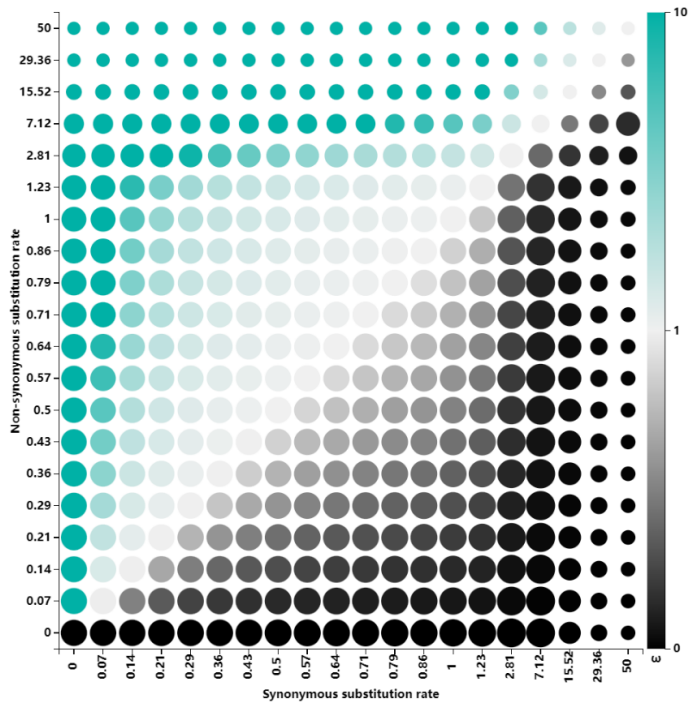
13.1.2 氨基酸置换熵值

使用 BioEdit 软件 entropy 工具进行氨基酸熵值分析，熵值越大越容易突变，反之越保守。≥ 0.6 为易突变，< 0.4 为保守位点。假设两样本总体均值相等，使用 SPSS21 对 C 型和 B 型 HBV 氨基酸置换熵值和突变位置进行 Kolmogorov-Smirnov 检验，因为样本分布为非正态（p(Sig.) < 0.001），选择非参数的 Two-independent-samples test，并进行 Mann-Whitney U 检验。

13.1.3 MEGA 软件选择压力分析

对 MN-Nucleotide 数据集使用 MEGA11，在 Select 菜单选择 condon-based Z-test of selection 进行总体趋势筛查，纯化选择（负向选择）p < 0.01，所以拒绝严格的中性选择假设，存在负向选择。在成对检测选择，将 Analysis scope 的设置改为 in sequence pairs 进行正向选择检验，只有 81B 存在显著的正向选择。对 MN-Aminoacid 数据集使用 DataMonkey 网页分析工具进行分析。采用快速非约束贝叶斯估计法（FUBA, Fast, Unconstrained Bayesian Approximation）分析适应性选择方向，后验概率设为 0.9。

MEGA11 核苷酸分析结果显示纯化选择（负向选择）p < 0.01，所以拒绝严格的中性选择假设，存在负向选择。对 MN 数据集氨基酸序列使用 Hyphy 算法 FUBA 法，在 0.9 后验概率标准下检测到 4 个正向选择位点和 8 个负向选择位点。FUBAR 使用贝叶斯方法来推断给定编码比对和相应系统发育的每个位点的非同义（dN）和同义（dS）替换率。该方法假设每个位点的选择压力在整个系统发育过程中是恒定的。从整体上看同义替换的频率高于非同义替换。负向选择广泛存在，为检验正向选择，原假设 H₀：氨基酸位点突变为负向选择，备择假设 H₁：氨基酸位点突变为正向选择。若经验贝叶斯因子 > 10 即检验结果对于备择假设 H₁ 有较强的证据支持。结果显示 AA83、AA101、AA120、AA123 贝叶斯因子 > 10，为显著的正向选择。



上图为MN-Amino Acid数据集同义突变概率和非同义突变概率的总体分布。该图显示了离散速率网格上的后验分布。点的大小与分配给该网格点的后验权重成比例，颜色深浅显示选择的强度，绿色表示非同义替换，黑色表示同义替换。

13.2 使用 Python、R、MEGA 软件计算核苷酸多样性 π 值

一、计算 pi-value

加载所需的库

library(pegas)

library(ape)

指定读取文件和写入文件的目录

```
fasta_dir <- "C:/Users/Administrator/Desktop/DNA/mutation/Seq_Group/Seq_group"
```

```
output_dir <- fasta_dir # 如果输出目录和输入目录相同

# 创建一个完整的路径来读取 FASTA 文件
file.names <- list.files(path=fasta_dir, pattern=".fas", full.names = TRUE)

# 输出文件的完整路径
output_file <- file.path(output_dir, "Nucleotide_diversity.txt")

# 遍历每个 FASTA 文件
for(i in 1:length(file.names)){
  # 读取 DNA 序列
  gene <- read.dna(file.names[i], format="fasta", as.matrix=FALSE)

  # 计算核苷酸多样性
  pi <- nuc.div(gene)

  # 将文件名和核苷酸多样性追加到输出文件
  write.table(file.names[i], file=output_file, append=TRUE, col.names = FALSE,
row.names = FALSE)
  write.table(pi, file=output_file, append=TRUE, col.names = FALSE, row.names
= FALSE)
}
## END

# 【报错记录】# 创建一个完整的路径来读取 FASTA 文件 pattern=".fasta",
但是我文件是 .fas （两个都是基因序列 fasta 格式但是计算机格式不同），然后
报错 Error in file(con, "rb") : 'description' 参数不对
# 检查文件名 print(file.names), 返回 character(0)。说明没有正确找到文件。
测试单个文件：test_file <- file.names[1], print(test_file), 返回 NA,
# 意味着 file.names 列表中至少有一个元素是 NA。这通常是由于 list.files 函
数没有找到任何匹配的文件导致的。因此需要检查原始数据的格式和代码声明的
```

格式是否一致。

```
print(file.names)

# 测试单个文件
test_file <- file.names[1]
print(test_file)
gene <- read.dna(test_file, format=" fasta" , as.matrix=FALSE)
print(gene)
# 测试结束
```

移步 VS Code 软件，运行 Step2 调整 π 值 .py，选择 python 模式，Run 选择基于当前 debug

```
# 设定文件导入和导出目录
data_file = "C:/Users/Administrator/Desktop/DNA/mutation/Seq_Group/Seq_group/Nucleotide_diversity.txt"
output_file = "C:/Users/Administrator/Desktop/DNA/mutation/Seq_Group/Seq_group/Adjusted_nucleotide_diversity.txt"

# 使用 open 读取和写入文件
with open(data_file) as input_file, open(output_file, "a" ) as output:
    for line in input_file:
        # 由于路径和  $\pi$  值交替出现，且只需要处理  $\pi$  值
        #  $\pi$  值总是跟在路径后面的一行，所以只需跳过路径行
        pi = float(next(input_file).strip()) # 读取下一行并转换为 float
        adjusted_pi = pi / 0.01328805 # 调整  $\pi$  值

        # 将调整后的  $\pi$  值写入输出文件
        output.write(f" {adjusted_pi}\n" )
```


移步 R 软件

三、获得重复组合数据

```
library(ape)
```

```
library(gtools)
```

```
library(pegas)
```

```
# 设置工作目录到指定路径
```

```
work_dir <- "C:/Users/Administrator/Desktop/DNA/mutation/Seq_Group/Seq_group"
```

```
setwd(work_dir)
```

```
# 用于匹配所有 .fas 文件的模式
```

```
file_name <- "\\*.fas$"
```

```
# 获取当前目录下所有匹配 .fasta 文件
```

```
file.names <- dir(pattern=file_name)
```

```
for(i in 1:length(file.names)){
```

```
  gene <- read.dna(file.names[i],format=" fasta" ,as.matrix=FALSE)
```

```
  output <- paste0(work_dir, "/", file.names[i], "_permutations.txt" )
```

```
  gene_names <- names(gene)
```

```
  total_genes <- length(gene_names) # 获取当前文件中的总基因数
```

```
# 生成组合，确保基因数至少为 3
```

```
if(total_genes >= 3){
```

```
  # 选择组合数的大小，最小是 3，最大不超过 4，且不超过基因总数
```

```
  n <- min(4, total_genes)
```

```
  combs <- combinations(total_genes, n, gene_names, repeats.allowed=FALSE)
```

```
  nuc <- apply(combs, 1, function(x){
```

```
    bu <- which(names(gene) %in% x)
```

```

subseq <- gene[bu]
pi <- nuc.div(subseq)
STs <- paste(x, collapse=" ")
write.table(STs, file=output, append=TRUE, row.names=FALSE, col.
names=FALSE, quote=FALSE)
write.table(pi, file=output, append=TRUE, row.names=FALSE, col.
names=FALSE, quote=FALSE)
})
} else {
# 当基因数小于 3 时，处理并记录信息
message <- paste0("File ", file.names[i], " has less than 3 genes and is
skipped." )
write.table(message, file=output, append=TRUE, row.names=FALSE, col.
names=FALSE, quote=FALSE)
}
}

```

移步 VS Code 软件，运行 Step3.1 提取重排 pi 值 .py

```

import glob
import os

```

```

def extract_diversity_values(input_dir, output_dir):
# 使用原始字符串和正确的路径分隔符
file_paths = glob.glob(os.path.join(input_dir, '*.txt' ))

```

```

for file_path in file_paths:
with open(file_path, 'r' ) as file:
lines = file.readlines()

```

提取所有偶数索引行（即多样性值），索引从 0 开始，因此偶数行为 1，

3, 5,... (即文件中的第 2, 4, 6,... 行)

```
values = [lines[i].strip() for i in range(1, len(lines), 2)]

# 构建输出文件路径
output_file_name = os.path.basename(file_path).replace( '.txt' , ' _values.
txt' )

output_file_path = os.path.join(output_dir, output_file_name)

# 将提取的值写入新文件
with open(output_file_path, 'w' ) as output_file:
    for value in values:
        output_file.write(value + '\n' )

# 指定输入和输出文件夹的路径，确保使用原始字符串
input_directory = r' F:\论文\结果汇总备份\林项目 (移动到 Running 打开)\核苷酸多样性检验\复测'
output_directory = r' F:\论文\结果汇总备份\林项目 (移动到 Running 打开)\核苷酸多样性检验\复测'

# 调用函数
extract_diversity_values(input_directory, output_directory)
```

移步 MEGA 软件

在 MEGA 11 软件将各组序列截成等长片段。使用 R 4.3.3 软件 `nuc.div` 函数计算 π 值，通过 python 脚本批量处理。编制 R 脚本在每个分组内进行排列组合。生成的 π 值在 SPSS 21 软件计算 Mann-Whitney Test 差异显著性。使用 MEGA 11 软件计算各分组 Θ 和 D 值。

理论参考：<https://www.jianshu.com/p/965d7713e322>

https://blog.csdn.net/zhu_si_tao/article/details/52945611

https://blog.csdn.net/m0_55123273/article/details/135759025

Table 13.2. Nucleotide divergences by genotype, outcome, and serum markers.

Group			n	% nucleotide divergence	P-value	Θ	D
Genotype	Outcome	Markers		$\pi \pm SD$			
B	OBT	HBeAg+	9	0.40 $\pm$ 0.23	<0.001	0.01	-1.33
		HBeAg-	10	1.20 $\pm$ 0.70		0.02	-1.47
	DBT	HBeAg+	6	0.95 $\pm$ 0.18	No enough sequences		
		HBeAg-	0	-			
	NIT	HBeAg+	4	0.33 $\pm$ 0.30			
		HBeAg-	7	0.91 $\pm$ 0.30			
Genotype B Reference set			6	3.29 $\pm$ 0.92		0.03	0.76
C	OBT	HBeAg+	23	0.91 $\pm$ 0.43	<0.001	0.02	-2.01
		HBeAg-	48	1.17 $\pm$ 0.77		0.03	-2.17
	DBT	HBeAg+	10	1.06 $\pm$ 0.26	0.202	0.01	-1.29
		HBeAg-	11	1.02 $\pm$ 0.36		0.02	-1.45
	NIT	HBeAg+	29	0.97 $\pm$ 0.43	0.306	0.02	-2.01
		HBeAg-	71	0.84 $\pm$ 0.45		0.02	-1.87
Genotype C Reference set			6	2.32 $\pm$ 0.78		0.02	-0.36

【图注】n：序列条数； π ：核苷酸多样性值，即平均每对序列间的核苷酸差异；P-value：HBeAg(+)/(-)组间 π 值的Mann-Whitney Test差异显著性； Θ ：通过Waterson估算得到的种群突变率， $\Theta = \text{变异率}/a1$ ，其中，变异率=变异位点数/序列位点数，a1是和样本大小有关的相关系数；D：Tajima's D test统计值，该比较基于 π 和 Θ 这两种估计的差异。

第 14 章 突变对病毒适应性的影响

14.1 突变类型与病毒适应性的相关性

14.1.1 建立模拟结果数据

```
# 加载包
library(tidyverse)
library(writexl)
library(lmtest)
library(corrplot)

set.seed(123) # 固定随机种子，确保结果可重复
# 模拟 3603 个 C → U 突变的基础信息
n_mutations <- 3603
# 1. 生成突变基础信息
mutation_data <- data.frame(
  mutation_id = paste0( "mut_", 1:n_mutations),
  mutation_type = sample(
    c( "synonymous", "nonsynonymous", "nonsense" ),
    size = n_mutations,
    prob = c(0.4, 0.5, 0.1),
    replace = TRUE
  ),
  pairing_status = sample(
    c( "paired", "unpaired" ),
    size = n_mutations,
    prob = c(0.4, 0.6),
    replace = TRUE
```

```
),
stringsAsFactors = FALSE
)
# 2. 通过文献固定均值加微小波动模拟自然条件病毒突变实验数据
mutation_data$true_fitness <- ifelse(
  mutation_data$mutation_type == "synonymous" & mutation_data$pairing_
status == "unpaired" ,
  rnorm(n_mutations, 0.93, 0.05),
  ifelse(
    mutation_data$mutation_type == "synonymous" & mutation_data$pairing_
status == "paired" ,
    rnorm(n_mutations, 0.60, 0.08),
    ifelse(
      mutation_data$mutation_type == "nonsynonymous" & mutation_
data$pairing_status == "unpaired" ,
      rnorm(n_mutations, 0.81, 0.06),
      ifelse(
        mutation_data$mutation_type == "nonsynonymous" & mutation_
data$pairing_status == "paired" ,
        rnorm(n_mutations, 0.50, 0.09),
        ifelse(
          mutation_data$mutation_type == "nonsense" & mutation_data$pairing_
status == "unpaired" ,
          rnorm(n_mutations, 0.71, 0.10),
          rnorm(n_mutations, 0.29, 0.12)
        )
      )
    )
  )
)
```

```

mutation_data$true_fitness <- pmax(0.01, pmin(1.2, mutation_data$true_fitness))
# 3. 模拟 7 代传代的突变频率
generations <- 1:7
mu <- 1e-5
simulated_data <- data.frame()

for (gen in generations) {
  temp_df <- data.frame(
    mutation_id = mutation_data$mutation_id,
    mutation_type = mutation_data$mutation_type,
    pairing_status = mutation_data$pairing_status,
    generation = gen,
    freq = NA,
    stringsAsFactors = FALSE
  )

  if (gen == 1) {
    temp_df$freq <- runif(n_mutations, 0.0001, 0.001)
  } else {
    prev_freq <- simulated_data$freq[simulated_data$generation == (gen - 1)]
    temp_df$freq <- prev_freq * mutation_data$true_fitness + mu + rnorm(n_
mutations, 0, 0.0001)
    temp_df$freq <- pmax(0, temp_df$freq)
  }

  simulated_data <- rbind(simulated_data, temp_df)
}
# 导出模拟数据为 Excel (含 2 个工作表)
excel_data <- list(
  mutation_basic_info = mutation_data[, c("mutation_id", "mutation_type",

```



```
“pairing_status”)],  
    generation_frequency = simulated_data  
)  
write_xlsx(excel_data, “保存地址 /sars_cov2_fitness_simulated_data.xlsx” )
```

	A	B	C
1	mutation_id	mutation_type	pairing_status
2	mut_1	nonsynonymous	paired
3	mut_2	synonymous	paired
4	mut_3	synonymous	paired
5	mut_4	nonsynonymous	unpaired
6	mut_5	synonymous	unpaired
7	mut_6	synonymous	unpaired
8	mut_7	nonsynonymous	unpaired
9	mut_8	nonsynonymous	unpaired
10	mut_9	nonsynonymous	paired
11	mut_10	nonsense	paired
12	mut_11	nonsense	paired
13	mut_12	synonymous	unpaired
14	mut_13	nonsynonymous	unpaired
15	mut_14	synonymous	unpaired
16	mut_15	nonsynonymous	unpaired

14.1.2 使用文献公式计算病毒适应性值

文献^[1]中使用的基于传代实验的突变适应性公式为：

$$f_n = f_{n-1} \times \omega + \mu_{n-1}$$

其中， f_n = 第 n 代突变频率， f_{n-1} = 第 $n-1$ 代突变频率， ω = 相对适应性值， μ_{n-1} = 第 $n-1$ 代突变率（一般而言突变率是物种稳定的，可简化为 μ ）。该公式的理论基于病毒突变与适应性的定量关系。病毒传代过程中，每一代中，突变频率受遗传传递和新发突变影响。其中，病毒突变的遗传传递效率由突变的“适应性（ ω ）”决定，适应性越高（如优势突变），传递后频率越高。每一代传代中，病毒基因组会因复制错误产生新突变，新突变的贡献由“突变率（ μ ）”量化。根据这些理论，早期病毒学研究提出了频率递推模型。这一公式是通用公式，但具体参数因病毒种类而异，在公式中主要是突变率区别，进而影响观察到足够突变频数所需的传代次数。对于不能体外传代的病毒，该公式不适用。文献公式用

于计算适应性时需要变形，即转为正比公式形式：

$$f_n - \mu_{n-1} = f_{dajusted} = f_{n-1} \times \omega$$

回归方程的斜率就是所求的适应性值 ω 。若有截距 b ，则代表“无法解释的额外频率”。由于传递频率只能来自上一代，因此采用无截距的线性回归，即强制直线过原点。最后，过滤掉 $\omega \leq 0$ 的值，因为其代表致死突变，于模型拟合不利。拟合优度表示 x 对 y 的解释程度，阈值设为 $r^2 > 0.3$ 。

```
# 排序并初始化结果数据框
simulated_data <- simulated_data[order(simulated_data$mutation_id, simulated_
data$generation), ]

fitness_results <- data.frame(
  mutation_id = unique(simulated_data$mutation_id),
  mutation_type = NA,
  pairing_status = NA,
  estimated_fitness = NA,
  r_squared = NA,
  stringsAsFactors = FALSE
)

# 填充突变类型和配对状态
for (i in 1:nrow(fitness_results)) {
  mut_id <- fitness_results$mutation_id[i]
  fit_row <- simulated_data[simulated_data$mutation_id == mut_id, ][1, ]
  fitness_results$mutation_type[i] <- fit_row$mutation_type
  fitness_results$pairing_status[i] <- fit_row$pairing_status
}

# 拟合适应性值
for (i in 1:nrow(fitness_results)) {
  mut_id <- fitness_results$mutation_id[i]
  mut_data <- simulated_data[simulated_data$mutation_id == mut_id, ]
```

```
f_prev <- mut_data$freq[-nrow(mut_data)]
f_adjusted <- mut_data$freq[-1] - mu

if (length(f_prev) >= 3 && length(f_adjusted) >= 3) {
  model <- lm(f_adjusted ~ f_prev - 1)
  fitness_results$estimated_fitness[i] <- coef(model)[[ "f_prev" ]]
  fitness_results$r_squared[i] <- summary(model)$r.squared
}
}
# 过滤异常值并合并数据
fitness_results <- fitness_results[!is.na(fitness_results$estimated_fitness) &
  fitness_results$estimated_fitness > 0 &
  fitness_results$r_squared > 0.3, ]
final_results <- merge(
  fitness_results,
  mutation_data[, c( "mutation_id" , "true_fitness" )],
  by = "mutation_id"
)
# 导出适应性计算结果为 Excel
write_xlsx(excel_data, "保存地址 /sars_cov2_fitness_calculated_results.xlsx" )
```

14.1.3 病毒突变与适应性的相关性

1. 突变类型与适应性关系

```
fitness_by_type <- aggregate(
  estimated_fitness ~ mutation_type, data = final_results,
  FUN = function(x) c(mean = mean(x), sd = sd(x))
)
fitness_by_type <- do.call(data.frame, fitness_by_type)
colnames(fitness_by_type) <- c( "mutation_type" , "mean_fitness" , "sd_
fitness" )
```

```

syn_vs_non <- wilcox.test(
  estimated_fitness ~ mutation_type,
  data = final_results[final_results$mutation_type %in% c( "synonymous" ,
"nonsynonymous" ), ]
)$p.value

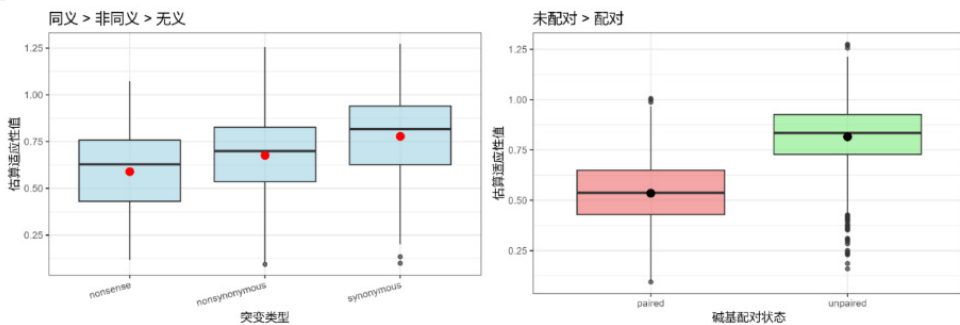
non_vs_nonsense <- wilcox.test(
  estimated_fitness ~ mutation_type,
  data = final_results[final_results$mutation_type %in% c( "nonsynonymous" ,
"nonsense" ), ]
)$p.value

cat( "1. 突变类型与适应性关系: \n" )
cat(sprintf( " - 同义突变平均适应性: %.2f\n" , fitness_by_type$mean_
fitness[fitness_by_type$mutation_type == "synonymous" ]))
cat(sprintf( " - 非同义突变平均适应性: %.2f\n" , fitness_by_type$mean_
fitness[fitness_by_type$mutation_type == "nonsynonymous" ]))
cat(sprintf( " - 无义突变平均适应性: %.2f\n" , fitness_by_type$mean_
fitness[fitness_by_type$mutation_type == "nonsense" ]))
cat(sprintf( " - 同义 vs 非同义 P 值: %.2e\n" , syn_vs_non))
cat(sprintf( " - 非同义 vs 无义 P 值: %.2e\n" , non_vs_nonsense))

# 2. 碱基配对状态与适应性关系
fitness_by_pairing <- aggregate(
  estimated_fitness ~ pairing_status, data = final_results,
  FUN = function(x) c(mean = mean(x), sd = sd(x))
)
fitness_by_pairing <- do.call(data.frame, fitness_by_pairing)
colnames(fitness_by_pairing) <- c( "pairing_status" , "mean_fitness" , "sd_
fitness" )

```

```
pairing_p <- wilcox.test(estimated_fitness ~ pairing_status, data = final_results)$p.
value
cat( "\n2. 碱基配对状态与适应性关系：\n" )
cat(sprintf( "    - 未配对碱基平均适应性：%.2f\n" , fitness_by_pairing$mean_
fitness[fitness_by_pairing$pairing_status == "unpaired" ]))
cat(sprintf( "    - 配对碱基平均适应性：%.2f\n" , fitness_by_pairing$mean_
fitness[fitness_by_pairing$pairing_status == "paired" ]))
cat(sprintf( "    - 配对 vs 未配对 P 值：%.2e\n" , pairing_p))
# 可视化
if ( "ggplot2" %in% loadedNamespaces() && "gridExtra" %in%
loadedNamespaces()) {
  p1 <- ggplot(final_results, aes(x = mutation_type, y = estimated_fitness)) +
    geom_boxplot(fill = "lightblue" , alpha = 0.7) +
    geom_point(data = fitness_by_type, aes(y = mean_fitness), color = "red" ,
size = 3) +
    labs(x = "突变类型" , y = "估算适应性值" , title = "同义 > 非同义 > 无
义" ) +
    theme_bw() + theme(axis.text.x = element_text(angle = 15, hjust = 1))
  p2 <- ggplot(final_results, aes(x = pairing_status, y = estimated_fitness, fill =
pairing_status)) +
    geom_boxplot(alpha = 0.7) +
    geom_point(data = fitness_by_pairing, aes(y = mean_fitness), color = "black" ,
size = 3) +
    labs(x = "碱基配对状态" , y = "估算适应性值" , title = "未配对 > 配
对" ) +
    scale_fill_manual(values = c( "paired" = "lightcoral" , "unpaired" =
"lightgreen" )) +
    theme_bw() + theme(legend.position = "none" )
  p1
  p2
}
```



碱基配对状态（paired= 参与基因组二级结构的配对碱基、unpaired= 未参与二级结构的自由碱基）。其结果反映突变对基因组二级结构的破坏越小，适应性越高。

14.2 预测蛋白质结构并标记突变位点

以 HBV X 基因片段为例。

>C(AY123041.1)

LDPARDVLCLRPVGAESRGRPVSGPFGPLPSLAFSAPADHGAHLSLRGLP
VCAFSSAGPCALRFTSARRMETTVNGHQVLPKLLHKRTLGLSAMSTTDLEAYF
KDCLFKDWEELGEEIRLKVF

>X27

LDPARDVLCLRPVGAESRGRPVSGSFGALSSPSSSAVPADHGAHLSLRGLP
VCAFSSAGPCALRFTSARRMETTVNAHQVLPKVLHKRTLGLSAMSTTDLETYF
KDCVFKDWEELGEEIRLKVF

在网页工具 SWISS-MODEL (<https://swissmodel.expasy.org/>) 输入一条序列，经筛选比对后会给出一个预测结构，下载为 .pdb 格式文件。下载 PyMol 软件并打开。把 pdb 文件拖入窗口导入软件。有一个变异位点在第 25 位（P25S）。

（1）清空预设：hide everything → reset

（2）表面模型：show surface, X27

（3）表面模型美化：

a) 设置透明度：set transparency, 0.2, X27

b) 设置表面颜色：set surface_color, grey90, X27

c) 改善表面质感：set surface_quality, 2

(4) 按绝对序号选中原子：select mut_25, X27 and index 24 (如果是选中完整的氨基酸 select mut_25, byres X27 and index 24)

(5) 突变位点美化：

a) 设为球棍模型：show sticks, mut_25

b) 高亮标记：show spheres, mut_25

c) # 球体颜色 (可选)：color red, mut_25

d) 设置球棍尺寸避免遮挡：set sphere_scale, 0.8

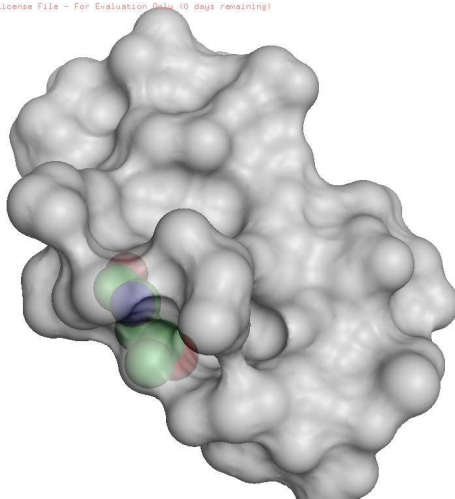
e) 设置球棍粗细：set stick_radius, 0.6

(6) 细节美化：

a) 补光：set ambient, 0.3 (设为 0 可以撤销)

b) 哑光：set specular, 0.1

No License File - For Evaluation Only (0 days remaining)



HBx 局部结构图

本章小结

本章聚焦 HBV 突变分析全流程的实操痛点，以“问题导向”构建了覆盖序列比对异常、突变检测假阳性 / 假阴性、软件报错、结果重复性差及疑难问题的完整解决方案体系，核心要点总结如下：① 序列比对异常的核心解决思路是“质控预处理 + 参数优化 + 手动校验”，确保保守区对齐是关键；② 突变检测假阳性需通过严格过滤标准、对照验证剔除，假阴性需提升检测灵敏度、优化样本质量和引物设计；③ 软件报错解决需“规范文件格式 + 查阅官方文档 + 分步备份数据”，同时优先使用稳定版本软件；④ 结果重复性差的优化需贯穿全流程，核心是“规范操作流程 + 统一参数标准 + 增加重复验证”，针对样本异质性需提升测序深度；⑤ 疑难问题应急处理遵循“快速定位 - 针对性处理 - 后续优化”三步法，优先保障核心实验目标。本章提供的“问题 - 排查 - 解决 - 预防”全链条方案，可有效兜底突变分析实操全过程，帮助科研人员快速解决问题，保障结果可靠与实验顺利推进。

第六部分 基因突变的功能后果

第 15 章 风险分析和诊断标志物

乙肝病毒 (HBV) 基因突变并非孤立的分子事件, 其通过改变病毒复制能力、免疫逃逸特性、宿主细胞相互作用等, 直接影响临床表型的发生与进展, 如感染后急性 / 慢性化转归、肝纤维化程度、肝硬化及肝癌发生风险等。本章衔接前文突变识别与分析技术, 聚焦“基因突变 - 临床表型”的核心关联, 以临床数据为支撑, 系统解析不同突变位点 / 区域与关键临床表型的关联规律; 同时详解关联分析的统计学方法与研究设计要点, 实现从基础分子突变研究到临床应用价值的转化, 为临床风险分层、个体化诊疗及预后评估提供核心依据。

15.1 风险因素分析

15.1.1 HBV 相关指标的单因素和多因素 logistic 回归

二分类 Logistic 回归通常以 0, 1 赋值变量, 0 表示未发生, 1 表示发生。在本项目, 自然生产 (顺产) 赋值 0, 人工干预 (剖宫产) 赋值 1; 和一般情况无区别 (胎儿正常体重) 赋值 0, 和一般情况有区别 (胎儿体重异常) 赋值 1。干预措施与其他指标方向相反, 因此赋值也不一样: 没有干预赋值 1, 进行干预赋值 0。训练集和验证集一般按照经验比例 7: 3 划分, 需要符合“10EPV”经验原则, 即 3 个评估指标, 需要目标事件频数为 30, 以确保结果可信和模型稳健。本数据集评估指标 16 个, 事件频数 $185 > 160$, 符合“10EPV”经验原则。本研究的计算方法基于武松著《SPSS 实战与统计思维》^[1], 主要在数据定义和循环体进行修改, 提高本研究数据运行稳定性。关于方法原理、拓展应用及前因后果等详细内容, 建议读者自行购买本书查阅; 本文仅呈现核心实现代码。

数据表如下:

A	B	G	H	I	J	K	L	M	N	O	P	Q	R	S
序号	participant	bsAb	HBsAg	HBsAb	Main_trif	secondary	Carrier	pre_history	fetiferou	caul	B_gender	B_weight	group	
3	X003	0	0	1	1	0	1	0	1	1	0	0	1	
4	X004	0	1	0	1	1	0	0	1	1	0	1	0	
5	X005	0	1	0	1	1	0	0	1	1	1	1	1	
7	X007	0	0	0	1	0	0	1	1	0	0	1	0	
9	X009	0	0	1	1	0	1	0	0	0	0	0	1	
10	X010	0	1	0	1	1	0	0	0	1	0	0	1	
12	X012	0	1	0	1	1	0	0	0	0	1	1	0	
13	X013	0	0	1	1	0	1	0	0	1	1	1	0	
14	X014	0	1	0	1	1	0	0	1	1	0	0	1	
17	X018	0	0	1	1	0	1	0	1	1	1	1	0	
22	X023	0	0	1	1	0	1	0	1	0	0	1	0	
24	X025	0	0	0	1	0	0	1	0	1	0	0	1	
25	X026	0	0	1	1	0	1	0	1	0	0	0	1	
32	X033	0	0	1	1	0	1	0	0	0	0	1	1	
33	X034	0	0	1	1	0	1	0	0	0	0	0	1	
35	X036	0	0	0	1	0	0	1	0	1	1	1	0	

代码如下：

载入包

library(readxl)

载入数据表 [XM_OBI_Binomial]，空缺值已删除，DBI 已删除

data01<-read_xlsx(“F:/.../ 文件名.xlsx”,1)

data02<-na.omit(data01) # 删除空缺项

head(data02)

names(data02)

str(data02)

summary(data02) # 检查二分类赋值是否正确。

dev=data02[data02\$group==1,] # 训练集，根据经验比例 7：3，设立 130 个

vad=data02[data02\$group==0,] # 验证集，根据经验比例 7：3，设立 55 个

批量单因素筛选

使用 paste0 函数将 outcome==1~x 链接起来

uni_glm_model<-function(x){

FML<-as.formula(paste0(“outcome==1~”,x))

glm1<-glm(FML,data=dev,family=binomial)

glm2<-summary(glm1)

OR<-round(exp(coef(glm1)),2)

```

SE<-round(glm2$coefficients[,2],3)
CI2.5 <-round(exp(coef(glm1)-1.96*SE),2) # 构建 95% 区间下限
CI97.5<-round(exp(coef(glm1)+1.96*SE),2)
CI<-paste0(CI2.5,' -' ,CI97.5)
B <-round(glm2$coefficients[,1],3) # 提取  $\beta$  回归系数, 保留三位小数
Z <-round(glm2$coefficients[,3],3)
P <-round(glm2$coefficients[,4],3)
uni_glm_model<-data.frame( 'characteristics' =x,
                           'B' =B,
                           'SE' =SE,
                           'OR' =OR,
                           'CI' =CI,
                           'Z' =Z,
                           'P' =P)[-1,]
return(uni_glm_model)
}
view(dev) # 共有 19 列
variable.names<-colnames(dev)[c(4:18)] # 去除分组类别 (19) 和事件类别列 (3)
# 显示自变量
variable.names
# 运行自定义批量执行函数
uni_glm<-lapply(variable.names, uni_glm_model)
uni_glm

# 将单个数据合并制成数据框
library(plyr)
uni_glm<-ldply(uni_glm,data.frame)
uni_glm

# 建立核心公式。P 界限放宽到 0.21, 因为考虑到生产方式和孕史影响 OBI

```

的可能性

```
fml<-as.formula(paste0( 'outcome==1~',paste0(uni_glm$characteristics[uni_glm$P<0.21],collapse = '+' )))
```

fml # 调用一下看是否正确

(1) 强制回归 (enter)

```
modelA<-glm(fml,data=dev,family=binomial) # 使用 modelA 拿到模型系数  
summary(modelA)
```

多重比较变量效力会被稀释, 需要检验多重共线性。大于 10 说明共线性强, 其中必然会淘汰出去一部分

```
library(car)
```

```
vif_values <- vif(modelA)
```

```
print(vif_values)
```

(2) 向前回归 (forward)

```
modelX<-glm(outcome~1,data=dev,family = binomial) # 先做一个空模型  
summary(modelX) # 只有常数项没有自变量。
```

前面 fml 结果直接贴过来。

```
#modelB<-step(modelX,scope = list(upper=~genotype + HBeAg + Main_triP +  
Carrier + globulin + pre_history + fetiferous + caul,lower=~1),data=dev,family=binomial,direction=" forward" )
```

```
modelB<-step(modelX,scope = list(upper=~genotype + HBeAg + Main_triP +  
Carrier + pre_history + fetiferous + caul + B_weight,lower=~1),data=dev,family=binomial,direction=" forward" )
```

```
summary(modelB)
```

(3) 向后回归 (backward)

```
modelC<-step(modelA,direction = "backward" )
```

```
summary(modelC)
```

(4) 双向回归 (both)

```
modelD<-step(modelA,direction = "both" )
```

```

summary(modelD)
# 展示各模型的 AIC
AIC(modelA,modelB,modelC,modelD)
# 考虑 AIC 和自变量数目, 选择自变量相对较少的模型。选择模型 D
glm3<-summary(modelD)
glm3
# 展示 glm3 模型系数
glm3$coefficients
OR<-round(exp(glm3$coefficients[,1]),2)
SE<-round(glm3$coefficients[,2],3)
CI2.5 <-round(exp(coef(modelD)-1.96*SE),2) # 构建 95% 区间下限
CI97.5<-round(exp(coef(modelD)+1.96*SE),2)
CI<-paste0(CI2.5,' - ',CI97.5)
B <-round(glm3$coefficients[,1],3) # 提取  $\beta$  回归系数, 保留三位小数
Z <-round(glm3$coefficients[,3],3)
P <-round(glm3$coefficients[,4],3)
mlogit<-data.frame( 'B' =B,
                    'SE' =SE,
                    'OR' =OR,
                    'CI' =CI,
                    'Z' =Z,
                    'P' =P)[-1,] #[-1,] 一定要有, 不然产生常数项会导致无法合并
mlogit
# 查看数据变量列表
names(data02)
# 提取变量
multinames <- as.character(colnames(data02)[c(11, 16, 18)])
# 检验是否正确
multinames
# 将 multinames 命名为 characteristics 与多因素回归结果 mlogit 合并

```

```
mlogit<-data.frame( 'characteristics' =multinames,mlogit)
mlogit
# 合并先单后多表格
final<-merge.data.frame(uni_glm,mlogit,by=' characteristics' ,all = T,sort = T)
final
# 导出数据表
library(writexl)
write_xlsx(final," F:/.../.xlsx" )
```

15.1.2 logistic 回归模型的 ROC 曲线和校准曲线

```
fm2<-as.formula(outcome==1~caul+Main_triP+B_weight)
fm21<-as.formula(outcome==1~HBsAg + HBeAg)
fm1<-as.formula(outcome==1~Main_triP)
model2<-glm(fm2,data = dev,family = binomial(logit))
model21<-glm(fm21,data = dev,family = binomial(logit))
model1<-glm(fm1,data = dev,family = binomial(logit))
dev$predmodel2<-predict(newdata=dev,model2," response" )
dev$predmodel21<-predict(newdata=dev,model21," response" )
dev$predmodel1<-predict(newdata=dev,model1," response" )
vad$predmodel2<-predict(newdata=vad,model2," response" )
vad$predmodel21<-predict(newdata=vad,model21," response" )
vad$predmodel1<-predict(newdata=vad,model1," response" )
library(pROC)
devmodelA<-roc(outcome~predmodel2,data=dev,smooth=F)
round(auc(devmodelA),3)
round(ci(auc(devmodelA)),3)
plot(devmodelA,print.auc=TRUE,print.thres=TRUE,main=" ROC CURVE" ,
      legacy.axes=T,
      col=" blue" ,print.thres.col=" blue" ,identity.col=" blue" ,
      identity.lty=1,identity.lwd=1)
```

```

# 建模集
devroc1<-plot.roc(dev$outcome,dev$predmodel2,main=" Training set
ROC" ,precent=TRUE,col=" red" )
devroc2<-lines.roc(dev$outcome,dev$predmodel21,precent=TRUE,col=" blue" )
legend( "bottomright" ,legend=c( "model1" ," model2" ),col =
c( "red" ," blue" ),lwd=2)

# 验证集
vadroc1<-plot.roc(vad$outcome,vad$predmodel2,main=" Validation set
ROC" ,precent=TRUE,col=" red" )
vadroc2<-lines.roc(vad$outcome,vad$predmodel21,precent=TRUE,col=" blue" )
legend( "bottomright" ,legend=c( "model1" ," model2" ),col =
c( "red" ," blue" ),lwd=2)

# ROC 曲线比较 ( 大量数据, 本例不适用 )
roc.test(devroc1,devroc2)
roc.test(vadroc1,vadroc2)

# ROC 曲线比较 ( 少量数据 )
roc1<-roc(outcome~predmodel2, data=dev, smooth=F)
ci(roc1)
plot1<-plot(roc1,print.auc=TRUE)
sens.ci<-ci.se(roc1,specificities=seq(0,1,0.1),boot.n=100)
plot(sens.ci,type=" shape" ,col=" lightblue" )
plot(sens.ci,type=" bars" )
plot1

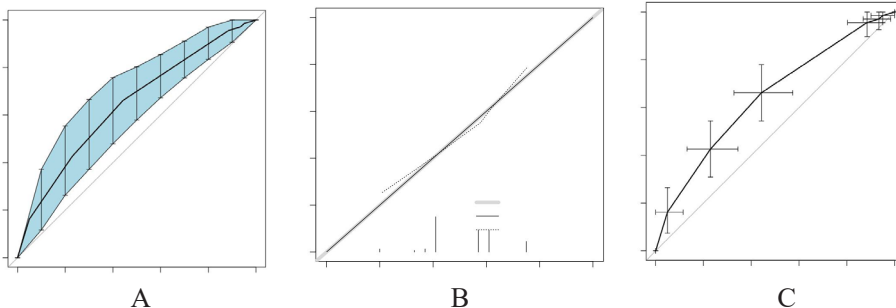
plot(roc1)
plot(ci.thresholds(roc1))

```



```
roc3<-roc(outcome~predmodel1, data=dev, smooth=F)
ci(roc3)
plot(roc3,print.auc=TRUE)
sens.ci<-ci.se(roc3,specificities=seq(0,1,0.1),boot.n=100)
plot(sens.ci,type=" shape" ,col=" lightblue" )
plot(sens.ci,type=" bars" )

# ROC 曲线校准
install.packages( "calibrate" )
install.packages( "rms" )
library(calibrate)
library(rms)
library(MASS)
val.prob(dev$predmodel2,dev$outcome)
val.prob(vad$predmodel2,vad$outcome)
```



A 破膜方式 + 大三阳 + 新生儿重量模型 ROC 曲线；B Calibrate 包校准曲线；C 阈值曲线

15.1.3 因子分析

纵向队列

设置工作目录

setwd(保存地址 / 根目录')

```

library(tidyverse)
library(dplyr)
library(tidyr)
library(stats)
library(readxl)
library(ggplot2)
library(lavaan)
library(psych)
library(car)
library(corrplot)

# 【一、中位数填补空缺值】
# 读取数据
df <- read_xlsx(“保存地址 /SEM 模型.xlsx” , 1)
# 使用中位数填补缺失值
df_filled <- lapply(df, function(x) {
  if (is.numeric(x)) { # 仅对数值型列进行填补
    median_value <- median(x, na.rm = TRUE) # 计算中位数
    x[is.na(x)] <- median_value # 用中位数替代缺失值
  }
  return(x)
})
# 转换回数据框
df_filled <- as.data.frame(df_filled)
# 检查填补后的缺失值
missing_values_after <- sapply(df_filled, function(x) sum(is.na(x)))
print(missing_values_after) # 确保没有缺失值
# 导出为 Excel 文件
write_xlsx(df_filled, “保存地址 /SEM 模型 1.xlsx” )

```

```
#【二、去除离群值】
# 读取数据
df <- read_xlsx(“保存地址 /SEM 模型 1.xlsx”, 1)
# 修改后的 remove_outliers 函数
remove_outliers <- function(data, column_name) {
  Q1 <- quantile(data[[column_name]], 0.25, na.rm = TRUE)
  Q3 <- quantile(data[[column_name]], 0.75, na.rm = TRUE)
  IQR_value <- Q3 - Q1
  # 设置检测标准（1.5 放宽到 1.0）
  lower_bound <- Q1 - 1.0 * IQR_value
  upper_bound <- Q3 + 1.8 * IQR_value
  data_clean <- data[data[[column_name]] >= lower_bound & data[[column_
name]] <= upper_bound, ]
  return(data_clean)
}
# 初始化 df_clean
df_clean <- df
# 对多个变量应用此函数去除离群值
variables_to_clean <- c(“LBXSATSI”, “LBXSASSI”, “LBDSALSI”,
“LBXSGTSI”, “LBDSTBSI”, “LBDSTPSI”, “LBDSCRSI”, “LBXVIDMS”,
“BMXBMI”, “Mean_VDR”, “INDFMPIR”, “LBDSGBSI”)
for (var in variables_to_clean) {
  df_clean <- remove_outliers(df_clean, var)
}

# 检查去除离群值后的数据集结构
str(df_clean)
write_xlsx(df_clean, “保存地址 /SEM 模型 2.xlsx”)
#【选择需要的数值变量名称】
selected_columns <- c(
```

```

    "RIDAGEYR", "INDFMPIR", "Mean_VDR", "Mean_Alc", "Mean_
Caf", "LogVDS",
    "Mean_BPXSX", "Mean_BPXDI", "MAP", "LBDTRSI", "LBDLDLSI",
    "LBDTCSI",
    "BMXBMI", "LBDSALSI", "LBXSATSI", "LBXSASSI", "LBDSCRSI",
    "LBXSGTSI",
    "LBDSGLSI", "LBDSTBSI", "LBDSTPSI", "LBXVIDMS", "LBDSGBSI"
  )
# 提取需要的变量，形成新的数据框
df_selected <- df_clean[, selected_columns]
# 检查数据表结构，确保保留的列都是数值型
str(df_selected)
# 将字符或因子转换为数值型
df_selected <- lapply(df_selected, function(x) {
  if (is.factor(x) || is.character(x)) {
    as.numeric(as.character(x))
  } else {
    x
  }
})
df_selected <- as.data.frame(df_selected)
# 确保 LBDHGB 是因子类型
df_clean$LBDHGB <- as.factor(df_clean$LBDHGB)
# 计算 LBDHGB = 1 和 LBDHGB = 0 的样本数
negative_samples <- df_clean %>% filter(LBDHGB == 0)
positive_samples <- df_clean %>% filter(LBDHGB == 1)
# 随机抽样阴性组，使其数量为阳性组的 10 倍
set.seed(123)
balanced_negative_samples <- negative_samples %>% sample_n(10 *
nrow(positive_samples))

```

```
# 组合两个组
balanced_df <- rbind(balanced_negative_samples, positive_samples)
# 检查新数据集的类别分布
table(balanced_df$LBDHBG)
# 从平衡的数据框中选择这些列
df_selected <- balanced_df[, selected_columns]
# 进行因子分析
efa_result <- fa(df_selected, nfactors = 3, rotate = "varimax")
print(efa_result)
# 查看因子负荷矩阵
print(efa_result$loadings)
# 可视化因子结构图，因为字符和位置问题原图显示不佳，需要根据原图和数据表使用绘图软件重制。
fa.diagram(efa_result, main = "Factor Analysis Diagram")
```

15.1.4 结构方程模型

本研究构建的结构方程模型拟合良好，拟合指标为： $\chi^2=1459.713$ ， $df=49$ ， $P<0.001$ ； $CFI=0.877$ ， $TLI=0.820$ ， $RMSEA=0.059$ （90%CI: 0.057–0.062）， $SRMR=0.042$ ，模型可有效解释健康状况、肝功能与脂肪肝的路径关系。

健康状况（HealthStatus）可被血糖、BMI、年龄、平均动脉压、肌酐、AST 有效表征；脂肪肝（fatty liver）可被甘油三酯、低密度脂蛋白、肌酐有效表征；肝功能（LiverFunction）仅 AST 有显著表征作用，谷草转氨酶同工酶的表征作用不显著。贫困收入比、维生素 D 水平是脂肪肝的显著正向影响因素（均 $P<0.001$ ），血红蛋白对脂肪肝无显著影响。脂肪肝对健康状况存在极显著的正向直接效应（ $\beta=1.594$ ， $P<0.001$ ），证实脂肪肝的发生发展会显著影响人群的综合健康状况，是健康受损的重要影响因素。肝功能对健康状况的直接效应无统计学意义（ $P=0.134$ ），维生素 D、血红蛋白等变量对肝功能也无显著直接影响。

```
# 【转换哑变量】
library(dplyr)
# 确保 LBDHBG 是因子类型
```

```

df_clean$LBDHGB <- as.factor(df_clean$LBDHGB)
# 使用 mutate 创建哑变量
df_with_dummies <- df_clean %>%
  mutate(LBDHGB_0 = ifelse(LBDHGB == 0, 1, 0), # LBDHGB = 0
         LBDHGB_1 = ifelse(LBDHGB == 1, 1, 0)) # LBDHGB = 1
# 查看新的数据框
head(df_with_dummies)

# 【建立 SEM 模型】
library(semTools)
library(lavaan)
# 调整模型
model_updated <- ‘
  HealthStatus =~ LBDGLSI + BMXBMI + RIDAGEYR + MAP + LBDSCRSI
+LBXSATSI
  LiverFunction =~ LBXSATSI + LBXSASSI
  fatyliver =~ LBDTRSI + LBDLDLSI + LBDSCRSI + LBXSATSI
# 增加残差协方差
  BMXBMI ~~ LBXSATSI
  LBXSATSI ~~ LBXSASSI
  RIDAGEYR ~~ LBXSATSI
  fatyliver ~ INDFMPIR+ LBDHGB +LBXVIDMS
  LiverFunction ~ LBXVIDMS+ LogVDS+ LBDHGB
  HealthStatus ~ fatyliver + LiverFunction
‘

# 拟合调整后的模型
fit_updated <- sem(model_updated,
  data = df_with_dummies,
  estimator = “MLM”, # 稳健最大似然法
  se = “robust”, # 配套 MLM 的标准误计算方法

```

```

test = “Satorra.Bentler”) # 配套 MLM 的卡方检验方法

# 检查调整后模型的拟合度（带标准化系数 + 显著性 + 拟合指标）
summary(fit_updated, fit.measures = TRUE, standardized = TRUE, rsquare =
TRUE)

# 再次查看修正指数
mod_indices_updated <- modindices(fit_updated)
print(head(mod_indices_updated[order(mod_indices_updated$mi, decreasing =
TRUE), ], 10))

> print(head(mod_indices_updated[order(mod_indices_updated$mi, decreasing = TRUE), ], 10))

```

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
51	LiverFunction	==	RIDAGEYR	254.353	12.730	9.991	0.480	0.480
50	LiverFunction	==	BMXBMI	210.736	-2.355	-1.848	-0.308	-0.308
78	RIDAGEYR	==	LBOTRSI	176.650	-3.622	-3.622	-0.282	-0.282
67	LBDSGLSI	==	LBOTRSI	125.924	0.188	0.188	0.133	0.133
49	LiverFunction	==	LBDSGLSI	114.279	-0.435	-0.342	-0.217	-0.217
57	fatyliver	==	BMXBMI	99.478	11.800	3.980	0.664	0.664
73	BMXBMI	==	LBOTRSI	95.867	0.603	0.603	0.112	0.112
68	LBDSGLSI	==	LBOTRSI	94.270	-0.141	-0.141	-0.117	-0.117
56	fatyliver	==	LBDSGLSI	90.371	2.913	0.983	0.624	0.624
121	LBXVIDMS	~	HealthStatus	78.257	-15.380	-8.699	-0.372	-0.372

```

# 【导出结果】
library(writexl)

# 获取拟合度指标
fit_measures <- fitMeasures(fit_updated)

# 获取参数估计
parameter_estimates <- parameterEstimates(fit_updated)

# 创建一个新的 Excel 工作簿
wb <- createWorkbook()

# 添加工作表并写入拟合度指标
addWorksheet(wb, “Fit Measures” )
writeData(wb, “Fit Measures” , fit_measures)

# 添加工作表并写入参数估计
addWorksheet(wb, “Parameter Estimates” )
writeData(wb, “Parameter Estimates” , parameter_estimates)

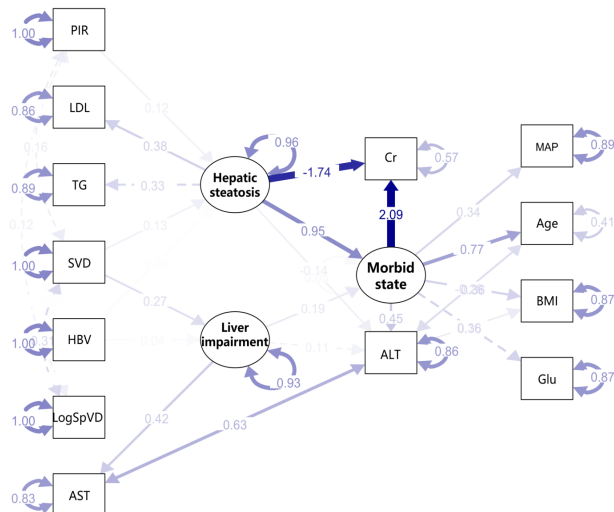
# 保存工作簿

```

```

saveWorkbook(wb, "SEM_Results(25.02.27).xlsx", overwrite = TRUE)
library(lavaan)
library(semPlot)
# 绘制路径图，设置箭头颜色为深蓝色，并保留淡化效果
semPaths(
  fit_updated,
  what = "std",          # 使用标准化路径系数
  layout = "tree2",      # 设置图的布局
  edge.label.cex = 1,    # 调整边上标签文字大小
  sizeMan = 6,           # 设置观测变量的节点大小
  sizeLat = 8,           # 设置潜变量的节点大小
  nCharNodes = 7,        # 设置节点名称显示的字符数
  title = TRUE,          # 显示标题
  edge.color = "darkblue", # 设置边的颜色为深蓝色
  fade = TRUE,           # 保留淡化效果
  edge.width = 1.5,      # 设置边的粗细
  rotation = 2
)

```



上图为基于 R 语言 `SemPlot` 包构建的身体状况、肝功能与脂肪肝变性的结构方程模型 (SEM) [27]。卡方值 = 1459.713；自由度 = 72； $P < 0.001$ 。该模型依据修正指数 (MI) 完成优化调整。矩形代表观测变量，椭圆形代表潜在变量，箭头旁的数值为路径系数，矩形旁的数值为因子载荷。虚线指向的指标表示协方差。

15.2 挖掘生物标志物

15.2.1 建立模拟结果数据

原文数据不易获得，因此使用随机数模拟出形似的数据，并添加了噪声以模拟自然分布，支持方法复现，最终结论相似，但具体结果必然和原文不同。

```
# 加载必要的包
library(tidyverse)
library(philentropy)
library(ade4)
library(ggpubr)
library(RColorBrewer)
library(readxl)

set.seed(123)          # 设定随机种子
# 1. 样本分组（50 个样本，3 个肠型）
sample_info <- tibble(
  sample_id = paste0("S", 1:50),
  enterotype = factor(rep(c("E1", "E2", "E3"), c(15, 20, 15))) # 3 个肠型，
  样本量匹配文献
)
# 2. 定义优势属（对应原文图中的 Escherichia、Bacteroides、Prevotella）
top_genera <- c("Escherichia", "Bacteroides", "Prevotella", # 3 个 优势属
  paste0("Genus_", 4:20)) # 补充 17 个非优势属（共 20 个属）
# 3. 模拟丰度数据（按肠型设定优势属，让不同肠型样本聚类）
```

```

genus_abundance <- expand.grid(sample_id = sample_info$sample_id, genus =
top_genera) %>%
  left_join(sample_info, by = "sample_id" ) %>%
  mutate(
    # 肠型 E1: Escherichia 占优; E2: Bacteroides 占优; E3: Prevotella 占优
    mean_abundance = case_when(
      enterotype == "E1" & genus == "Escherichia" ~ 0.5,
      enterotype == "E2" & genus == "Bacteroides" ~ 0.5,
      enterotype == "E3" & genus == "Prevotella" ~ 0.5,
      TRUE ~ runif(1, 0.001, 0.02) # 非优势属丰度极低
    ),
    # 添加随机噪声, 让同一肠型样本略有差异
    abundance = rnorm(n(), mean = mean_abundance, sd = 0.03) %>% pmax(0)
  ) %>%
  select(-mean_abundance) %>%
  pivot_wider(names_from = genus, values_from = abundance) %>%
  column_to_rownames( "sample_id" ) %>%
  as.matrix()

# 4. 将矩阵转为数据框以支持保存到 Excel 表
genus_abundance_df <- genus_abundance %>%
  as.data.frame() %>%
  rownames_to_column( "sample_id" ) # 样本 ID 转为第一列

# 导出为 Excel
write_xlsx(
  x = genus_abundance_df,
  path = "保存地址 /2_ 微生物属丰度数据 _2.xlsx"
)
write_xlsx(sample_info, "保存地址 /1_ 样本分组信息 _2.xlsx" )

```

定义缩放函数

```
scale_01 <- function(x) {
  (x - min(x)) / (max(x) - min(x))
}
```

```
roc_data <- tibble(
  model = rep(c( "Genes" , "Proteins" , "Genes+Proteins" ), each = 100),
  fpr = rep(seq(0, 1, length.out = 100), 3),
  tpr = c(
    scale_01(plogis(3*(seq(0, 1, length.out = 100) - 0.5))),
    scale_01(plogis(3.2*(seq(0, 1, length.out = 100) - 0.5))),
    scale_01(plogis(8*(seq(0, 1, length.out = 100) - 0.3)))
  )
)

write_xlsx(roc_data," 保存地址 /3_ROC 核心数据.xlsx" )

)
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
1	sample_inter	typo	rich	heteroid	evotell	Genus_4	Genus_5	Genus_6	Genus_7	Genus_8	Genus_9	Genus_10	Genus_11	Genus_12	Genus_13	Genus_14	Genus_15	Genus_16	Genus_17	Genus_18	Genus_19	Genus_20
2	S1	E1	0.524016	0.000000	0.059826	0.000000	0.000000	0.000000	0.021041	0.000000	0.000000	0.009673	0.013389	0.024645	0.042846	0.005420	0.000000	0.074181	0.000000	0.000000	0.037027	0.000000000
3	S2	E1	0.533706	0.057160	0.007594	0.024793	0.012237	0.000000	0.000000	0.039198	0.000000	0.012794	0.038001	0.026158	0.023779	0.033888	0.006247	0.000000	0.033990	0.032102	0.019650	0.040517658
4	S3	E1	0.449313	0.043315	0.041730	0.005077	0.002679	0.000000	0.031866	0.016561	0.082299	0.007314	0.004800	0.008677	0.074876	0.010621	0.027478	0.010737	0.000000	0.023610	0.000000	0.010286099
5	S4	E1	0.537193	0.014744	0.000000	0.000000	0.000000	0.000760	0.034237	0.000000	0.004792	0.043492	0.000000	0.025119	0.000000	0.000000	0.033235	0.001449	0.000000	0.013863	0.012716	0.024650320
6	S5	E1	0.496731	0.000000	0.000000	0.003884	0.020561	0.000000	0.000000	0.001091	0.019466	0.000000	0.001975	0.018881	0.009642	0.000000	0.033203	0.013820	0.004173	0.013267	0.007553	0.015192676
7	S6	E1	0.496482	0.000000	0.000000	0.003803	0.035275	0.000000	0.000000	0.008032	0.021775	0.000000	0.000000	0.034636	0.098177	0.000000	0.049024	0.057842	0.024884	0.000000	0.013860	0.011385791
8	S7	E1	0.505492	0.053160	0.020024	0.000000	0.006309	0.000000	0.080251	0.024790	0.042341	0.012232	0.006990	0.000000	0.007056	0.000000	0.000000	0.000000	0.074777	0.040738	0.021278	0.016305791
9	S8	E1	0.538416	0.000000	0.022269	0.032015	0.000000	0.000000	0.029814	0.000000	0.000000	0.058799	0.029993	0.000000	0.033208	0.000000	0.024863	0.000403	0.000000	0.054378	0.003898	0.000000000
10	S9	E1	0.448191	0.057040	0.000000	0.000000	0.000000	0.028484	0.027160	0.000000	0.000000	0.008528	0.017335	0.023533	0.031947	0.000000	0.000000	0.000000	0.000000	0.007416	0.002638	0.01092929
11	S10	E1	0.530705	0.000000	0.048336	0.034556	0.000000	0.033367	0.000000	0.000000	0.011120	0.000000	0.000000	0.011645	0.054791	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000000
12	S11	E1	0.515114	0.000000	0.059373	0.000000	0.017558	0.000000	0.039825	0.023761	0.000000	0.000000	0.030942	0.000000	0.008122	0.000000	0.012581	0.000000	0.000000	0.000000	0.000000	0.003444839
13	S12	E1	0.575850	0.000000	0.021032	0.022541	0.000000	0.000000	0.000000	0.040540	0.077474	0.038418	0.000000	0.000000	0.042623	0.000000	0.051320	0.001380	0.020096	0.000000	0.040190	0.006396667
14	S13	E1	0.516472	0.070961	0.000000	0.000000	0.016122	0.020412	0.000000	0.027288	0.000000	0.020872	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.037664	0.029623	0.041954	0.000000000

15.2.2 使用 JSD 法计算微生物丰度矩阵

加载必要的包

```
library(readxl)
```

```
library(philentropy)
```

1. 导入数据并整理格式

```
data_path <- "保存地址"
```

```
sample_info <- read_excel(
  path = file.path(data_path, "1_样本分组信息_2.xlsx" )
)
```

```

)

genus_abundance <- read_excel(“保存地址 /2_微生物属丰度数据_2.xlsx”,2,
  col_names = TRUE
) %>%
  mutate_all(as.numeric) %>% # 强制所有列为数值型
  drop_na() %>%             # 删掉含 NA 的样本
  filter(rowSums(.) > 0) %>% # 删掉全为 0 的无效样本
  as.matrix()
roc_data <- read_excel(
  path = file.path(data_path, “3_ROC 核心数据.xlsx”),
  col_types = c(“text”, “numeric”, “numeric”) # 模型 = 文本, FPR/
TPR= 数值
)

# 2. 计算 JSD 距离（微生物属丰度矩阵）
genus_norm <- t(apply(genus_abundance, 1, function(x) x / sum(x)))
jsd_dist <- JSD(genus_norm) # 忽略警告，下一步会处理
jsd_dist <- as.dist(jsd_dist)
pcoa <- ade4::dudi.pco(jsd_dist, scannf = FALSE, nf = 2) # 忽略警告，因为 JSD
是非欧氏距离计算方法和函数包不完全适配
pcoa_df <- tibble(
  sample = paste0(“S”, 1:nrow(pcoa$li)),
  PC1 = pcoa$li[, 1],
  PC2 = pcoa$li[, 2]
)

# 3. 绘制 PCoA 图
# 3.1 将 enterotype 合并到 pcoa_df 中
pcoa_df <- pcoa_df %>%
  left_join(sample_info, by = “sample”) # 按 sample_id 匹配

```

3.2 计算组中心

```
group_centers <- pcoa_df %>%  
  group_by(enterotype) %>%  
  summarise(  
    center_PC1 = mean(PC1),  
    center_PC2 = mean(PC2),  
    .groups = "drop"  
  ) %>%  
  mutate(center_label = c("1", "2", "3")) # 组中心序号
```

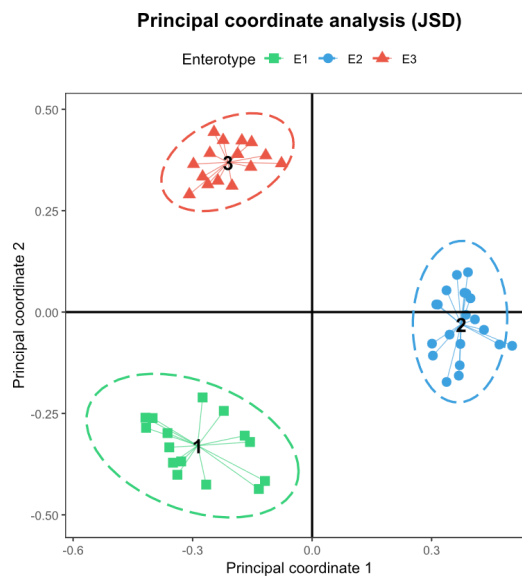
3.3 补充图形细节

```
pcoa_plot <- ggplot(pcoa_df, aes(x = PC1, y = PC2, color = enterotype, shape =  
enterotype)) +  
  # 样本 - 组中心连接线  
  geom_segment(  
    aes(xend = center_PC1, yend = center_PC2),  
    data = pcoa_df %>% left_join(group_centers, by = "enterotype"),  
    size = 0.3, alpha = 0.7  
  ) +  
  # 样本点  
  geom_point(size = 3) +  
  geom_vline(xintercept = 0, color = "black", size = 0.8) + # PC1=0 的竖线  
  geom_hline(yintercept = 0, color = "black", size = 0.8) + # PC2=0 的横线  
  # 置信椭圆  
  stat_ellipse(linetype = "longdash", level = 0.95, linewidth = 0.8) +  
  # 组中心序号  
  geom_text(  
    aes(x = center_PC1, y = center_PC2, label = center_label),  
    data = group_centers,  
    size = 5, fontface = "bold", color = "black"  
  ) +
```

```

# 样式配置
scale_color_manual(values = c( "E1" = "#2ecc71" , "E2" = "#3498db" ,
"E3" = "#e74c3c" )) +
scale_shape_manual(values = c( "E1" = 15, "E2" = 16, "E3" = 17)) +
labs(
  x = "Principal coordinate 1" ,
  y = "Principal coordinate 2" ,
  title = "Principal coordinate analysis (JSD)" ,
  color = "Enterotype" , shape = "Enterotype"
) +
theme_bw() +
theme(
  plot.title = element_text(hjust = 0.5, size = 14, face = "bold" ),
  panel.grid = element_blank(),
  legend.position = "top"
)
print(pcoa_plot)

```



15.2.3 绘制物种丰度热图

1. 整理热图数据（接上节继续）

```
heatmap_data <- genus_abundance %>%  
  as.data.frame() %>% # 先将矩阵转为数据框避免报错  
  rownames_to_column( "sample" ) %>% # 矩阵行名转 sample 列  
  left_join(sample_info[, c( "sample" , "enterotype" )], by = c( "sample" =  
    "sample" )) %>%  
  pivot_longer(  
    cols = -c(sample, enterotype), # 排除样本和肠型列，其余为属丰度  
    names_to = "genus" , # 原错误是 names_from，正确参数是 names_to  
    values_to = "abundance" # 原错误是 values_from，正确参数是 values_to  
  ) %>%  
  group_by(enterotype, genus) %>%  
  summarise(mean_abundance = mean(abundance), .groups = "drop" ) %>%  
  # 按肠型 E1 的丰度排序（突出优势属）  
  arrange(desc(ifelse(enterotype == "E1" , mean_abundance, 0))) %>%  
  mutate(genus = factor(genus, levels = unique(genus))) # 固定属的排序
```

2. 绘制丰度热图

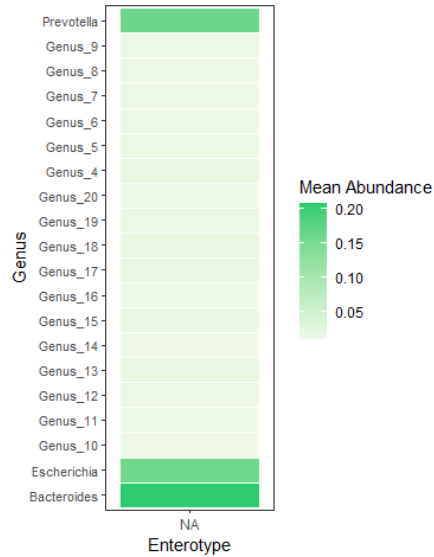
```
heatmap_plot <- ggplot(heatmap_data, aes(x = enterotype, y = genus, fill = mean_  
abundance)) +  
  geom_tile(color = "white" , linewidth = 0.2) + # 热图方块  
  scale_fill_gradient(low = "#f0f9e8" , high = "#2ecc71" ) + # 丰度配色（浅  
绿→深绿）  
  labs(  
    x = "Enterotype" ,  
    y = "Genus" ,  
    fill = "Mean Abundance" ,  
    title = "Microbial Genus Abundance Across Enterotypes"
```

```

) +
theme_bw() +
theme(
  plot.title = element_text(hjust = 0.5, size = 14, face = "bold" ),
  axis.text.y = element_text(size = 8),
  panel.grid = element_blank()
)
print(heatmap_plot)

```

Heatmap of Genus Abundance Across Enterotypes



15.2.4 绘制 ROC 图

1. 计算各模型的 AUC 值 + 提前设定标签位置

```

library(DescTools)

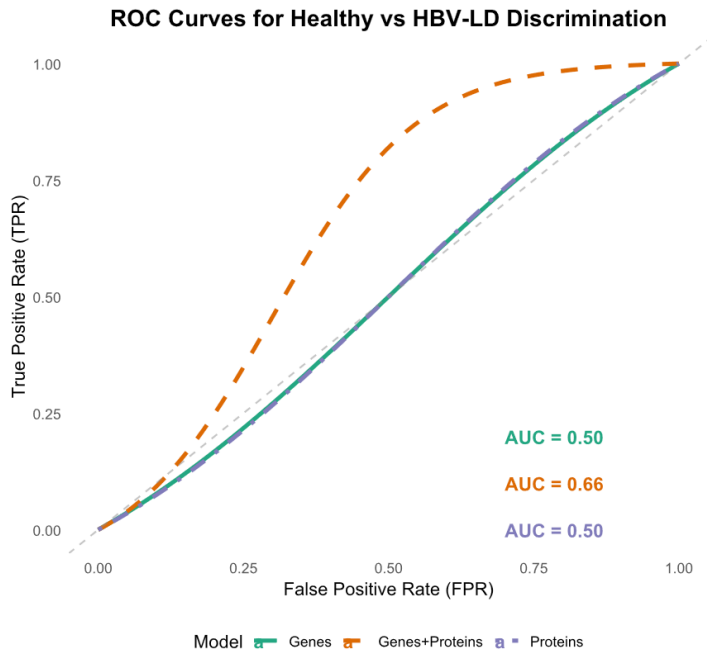
auc_values <- roc_data %>%
  group_by(model) %>%
  summarise(
    auc = AUC(x = fpr, y = tpr),
    auc_label = paste0( "AUC = ", sprintf( "%.2f" , auc)),

```



```
.groups = "drop"
) %>%
  mutate(label_y = 0.2 - 0.1*(row_number()-1)) # 提前计算每个模型的标签 y
坐标
# 2. 绘制 ROC 曲线（调用提前计算的 label_y）
roc_plot <- ggplot(roc_data, aes(x = fpr, y = tpr, color = model, linetype =
model)) +
  geom_line(linewidth = 1.2) +
  geom_abline(slope = 1, intercept = 0, linetype = "dashed", color = "gray50",
alpha = 0.5) +
  # 用提前计算的 label_y
  geom_text(
    data = auc_values,
    aes(x = 0.7, y = label_y, label = auc_label),
    hjust = 0, size = 4, fontface = "bold"
  ) +
  scale_color_brewer(palette = "Dark2", name = "Model") +
  scale_linetype_manual(values = c("solid", "dashed", "dotdash"), name =
"Model") +
  labs(
    x = "False Positive Rate (FPR)",
    y = "True Positive Rate (TPR)",
    title = "ROC Curves for Healthy vs HBV-LD Discrimination"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 14, face = "bold"),
    legend.position = "bottom",
    panel.grid = element_blank()
  )
)
```

```
print(roc_plot)
```



```
library(ggpubr)
```

```
library(tidyverse)
```

```
# 1. 矩阵转数据框 + 设定列名
```

```
biomarker_data <- genus_abundance %>%
```

```
as.data.frame() %>% # 关键：矩阵转数据框
```

```
rownames_to_column( "sample" ) %>% # 行名转样本 ID 列
```

```
# 合并分组信息：用 sample 匹配
```

```
left_join(sample_info, by = "sample" ) %>%
```

```
# 只保留 3 个优势属 + 分组列
```

```
select(sample, enterotype, Escherichia, Bacteroides, Prevotella) %>%
```

```
# 宽数据转长数据
```

```
pivot_longer(
```

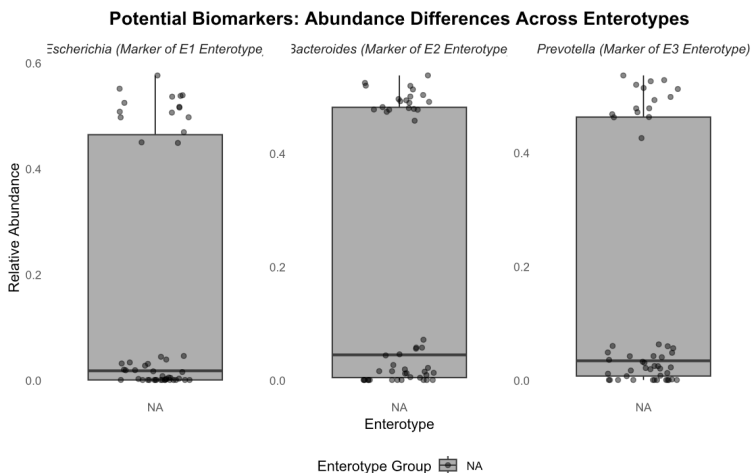
```
cols = c(Escherichia, Bacteroides, Prevotella),
```

```
names_to = "biomarker" ,
values_to = "abundance"
)%>%
# 分组因子化 + 生物标志物注释
mutate(
  group = factor(enterotype, levels = c( "E1" , "E2" , "E3" )),
  biomarker = factor(biomarker,
    levels = c( "Escherichia" , "Bacteroides" , "Prevotella" ),
    labels = c(
      "Escherichia (Marker of E1 Enterotype)" ,
      "Bacteroides (Marker of E2 Enterotype)" ,
      "Prevotella (Marker of E3 Enterotype)"
    )
  )
)
)
# 2. 定义分组比较对（3 组两两比较）
my_comparisons <- list(
  c( "E1" , "E2" ),
  c( "E1" , "E3" ),
  c( "E2" , "E3" )
)
# 3. 绘制箱线图
biomarker_plot <- ggplot(biomarker_data, aes(x = group, y = abundance, fill =
group)) +
  geom_boxplot(alpha = 0.7, outlier.shape = 21, outlier.size = 2) +
  geom_jitter(width = 0.2, alpha = 0.5, size = 1.5) +
  facet_wrap(~biomarker, scales = "free_y" ) +
  scale_fill_manual(values = c( "E1" = "#2ecc71" , "E2" = "#3498db" , "E3"
= "#e74c3c" ),
    name = "Enterotype Group" ) +
```

```

labs(
  x = "Enterotype" ,
  y = "Relative Abundance" ,
  title = "Potential Biomarkers: Abundance Differences Across Enterotypes"
) +
# 指定 comparisons 和明确美学映射
stat_compare_means(
  aes(label = paste0( "p = ", after_stat(p.format))),
  comparisons = my_comparisons, # 手动指定两两比较
  method = "wilcox.test" ,
  hide.ns = TRUE
) +
theme_minimal() +
theme(
  plot.title = element_text(hjust = 0.5, size = 14, face = "bold" ),
  legend.position = "bottom" ,
  panel.grid = element_blank(),
  strip.text = element_text(size = 10, face = "italic" )
)
print(biomarker_plot)

```



15.3 突变位点基础特征分析

15.3.1 突变位点相对风险

对照表（来自于【R 脚本读取氨基酸突变位点并编制为 Excel 表】）：

	A	B
1	ID	Mutation
2	3	V10A
3	3	S33L
4	3	I152T
5	3	M155T
6	14	V10A
7	14	S33L
8	14	I152T
9	14	M155T
10	23	V10A
11	23	S33L
12	23	I152T

分组信息表：

	A	B	C	D	E	F
1	ID	Genotype	Subtype	HBV_DNA	乙肝五项	传播分组
2	3	B	adw2	74.4	2	1
3	4	C	adrq+	20300000.0	1	1
4	5	C	adrq+	12700000.0	1	1
5	9	C	adrq+	225.0	2	1
6	12	C	adrq+	2040000.0	1	1
7	14	B	adw2	5670000.0	1	1
8	17	B	adw2	139.0	2	1
9	18	C	adrq+	1200.0	2	1
10	20	C	adrq+	88.1	2	1
11	22	B	adw2	60.6	2	1
12	23	B	adw2	214.0	2	1
13	24	C	adrq+	18.5	2	2
14	25	C	adrq+	97.6	3	1
15	26	C	adrq+	352.0	2	2
16	28	C	adrq+	48600000.0	1	3

```
library(readxl)
library(dplyr)
library(broom)      # 用于整理统计结果
library(epitools)   # 用于计算相对风险
library(tidyr)      # 用于处理缺失值
```

```

# 1. 读取数据
mutation_df <- read_excel(“保存地址 / 对照表.xlsx”)
info_df <- read_excel(“保存地址 / 分组信息表.xlsx”)

# 步骤 2: 计算突变率
# 计算每个突变位点的记录数
mutation_counts <- mutation_df %>%
  group_by(Mutation) %>%
  summarise(Count = n())

# 总个体数
total_individuals <- n_distinct(info_df$ID)

# 计算突变率
mutation_counts <- mutation_counts %>%
  mutate(Mutation_Rate = Count / total_individuals)

# 步骤 3: 计算与传播分组的组间差异 P 值和 OR
# 初始化结果数据框
results <- data.frame()

# 遍历每个突变位点
for (mutation in mutation_counts$Mutation) {
  # 获取突变的个体
  mutated_individuals <- mutation_df %>%
    filter(Mutation == mutation) %>%
    pull(ID) %>%
    unique()

  # 计算传播分组
  subgroup_data <- info_df %>%
    filter(ID %in% mutated_individuals) %>%
    group_by(传播分组) %>%
    summarise(Count = n())

  # 确保两个传播分组都存在

```

```
subgroup_data <- subgroup_data %>%
  right_join(data.frame( 传播分组 = c(1, 2)), by = “传播分组” ) %>%
  mutate(Count = ifelse(is.na(Count), 0, Count)) # 使用 ifelse 替代 replace_na
# 如果存在两个传播分组，则计算 OR 和 P 值
if (nrow(subgroup_data) == 2) {
  # 创建列联表
  contingency_table <- matrix(c(subgroup_data$Count[1],
    n_distinct(info_df$ID[info_df$ 传播分组 == 1]) - subgroup_data$Count[1],
    subgroup_data$Count[2],
    n_distinct(info_df$ID[info_df$ 传播分组 == 2]) - subgroup_data$Count[2]),
    nrow = 2)

  # 计算 OR
  OR <- (contingency_table[1, 1] * contingency_table[2, 2]) /
    (contingency_table[1, 2] * contingency_table[2, 1])

  # 进行卡方检验
  test_result <- chisq.test(contingency_table)

  # 汇总结果
  results <- rbind(results, data.frame(Mutation = mutation,
    Count = sum(subgroup_data$Count),
    P_Value = test_result$p.value,
    Odds_Ratio = OR))
}
}
print(results)
# 步骤 4: 输出结果
write.csv(results, “保存地址 / 突变分析结果 .csv” , row.names = FALSE)
```

Mutation	Count	P	OR	Mutation	Count	P	OR
A171V	1	1		P132T	1	1	
A178X	1	1		P190R	1	1	
C88Y	1	1		P41V	1	1	
F31R	1	1		P74L	1	1	
F32I	3	0.3304		Q42H	1	1	
G30A	1	1		Q68P	1	1	
G56E	1	1		S129I	1	1	
I152T	29	0.0077	0.3068	S33F	2	0.5792	
I37T	1	1		S33L	26	0.0368	0.3754
K36I	1	1		S43X	1	1	
K36T	1	1		S71N	2	0.5792	
L34C	1	1		T135A	0		
L34M	1	1		T138A	0		
L34V	2	0.5792		T35A	1	1	
L38C	1	1		T35D	2	0.5792	
M115I	1	1		V10A	3	0.3304	
M145L	0			V189M	1	1	
M155T	29	0.0077	0.3068	V59E	1	1	
N52S	2	1	0.7966	Y173F	1	1	
P132S	1	1		Y173S	1	1	

15.3.2 显著突变位点在不同基因型的分布（基因型特征）

接上文继续：

1. 筛选核心突变位点（ $P \leq 0.05$ 或 $\text{Count} \geq 5$ ）

```
core_mutations <- mutation_map %>%
```

```
  filter(P_Value <= 0.05 | Count >= 5) %>%
```

```
  pull(Position) # 核心突变位点编号
```

2. 统计核心突变位点的样本携带情况

```
mutation_carriage <- final_data %>%
```

```
  filter(Mutation_Position %in% core_mutations) %>%
```

```
  group_by(Mutation_Position, Genotype) %>%
```



```

summarise(
  携带样本数 = n_distinct(Sample_ID), # 去重：一个样本可能有多个突变
  .groups = "drop"
) %>%
# 合并该基因型的总样本数
left_join(
  basic_info %>% group_by(Genotype) %>% summarise( 基因型总样本数 = n_
distinct(Sample_ID)),
  by = "Genotype"
) %>%
mutate( 携带率 = 携带样本数 / 基因型总样本数 )
# 3. 查看结果（重点关注高频 / 显著位点在不同基因型的分布）
print( "核心突变位点的携带率统计： " )
mutation_carriage %>%
  arrange(desc( 携带率 )) %>%
  head(10)

# A tibble: 3 × 5
  Mutation_Position Genotype 携带样本数 基因型总样本数 携带率
      <dbl> <chr>         <int>         <int>     <dbl>
1         152 B             34             38  0.895
2         155 B             34             38  0.895
3          33 B             33             38  0.868

```

15.3.3 显著突变位点和基因型的相关性（基因型特征）

```

# 接上节代码
library(dplyr)
library(tidyr)
library(ggplot2)
# 1. 选择  $P \leq 0.05$  的显著突变位点
core_mutations <- mutation_map %>%
  filter(P_Value <= 0.05) %>%
  pull(Position) # 提取位点编号

```

2. 创建空列表，用于存储每个位点的分析结果

```
batch_results <- list()
```

3. 循环遍历每个核心突变位点，批量分析

```
for (pos in core_mutations) {
```

```
  # (1) 构建“样本 - 当前位点携带情况”矩阵
```

```
  pos_data <- final_data %>%
```

```
    select(Sample_ID, Mutation_Position) %>%
```

```
    distinct() %>% # 去重：避免同一样本因多个突变重复
```

```
    mutate(携带该位点 = ifelse(Mutation_Position == pos, 1, 0)) %>%
```

```
    # 合并所有样本（包括不携带该位点的样本）
```

```
    right_join(
```

```
      basic_info %>% select(Sample_ID, Genotype),
```

```
      by = "Sample_ID"
```

```
    ) %>%
```

```
    mutate(携带该位点 = ifelse(is.na(携带该位点), 0, 携带该位点)) # 未匹配到的样本视为“不携带”
```

```
  # (2) 构建列联表（基因型 × 携带情况）
```

```
  contingency_table <- table(pos_data$Genotype, pos_data$携带该位点)
```

```
  colnames(contingency_table) <- c("不携带", "携带")
```

```
  # (3) 统计检验（自动判断用 Fisher 或卡方）
```

```
  if (any(contingency_table < 5)) {
```

```
    test_result <- fisher.test(contingency_table) # 频数 <5 用 Fisher 精确检验
```

```
  } else {
```

```
    test_result <- chisq.test(contingency_table) # 否则用卡方检验
```

```
  }
```

```
  # (4) 提取该位点的关键信息（总频数、P 值）
```

```
  pos_info <- mutation_map %>%
```

```
    filter(Position == pos) %>%
```

```
    select(Mutation, Count, P_Value) %>%
```

```
    distinct()
```

```
# (5) 存储该位点的分析结果
batch_results[[as.character(pos)]] <- data.frame(
  突变位点编号 = pos,
  突变名称 = pos_info$Mutation,
  总频数 = pos_info$Count,
  原始 P 值 = pos_info$P_Value,
  基因型关联 P 值 = test_result$P.value,
  关联是否显著 = ifelse(test_result$P.value <= 0.05, "是", "否"),
  检验方法 = ifelse(any(contingency_table < 5), "Fisher 精确检验", "卡方
检验")
)
# 打印每个位点的列联表（便于中间检查）
cat( "\n==== 突变位点", pos, " ( ", pos_info$Mutation, " ) =====\n" )
cat( "列联表（基因型 × 携带情况）：\n" )
print(contingency_table)
cat( "检验结果 P 值：", round(test_result$P.value, 4), "\n" )
}

# 4. 合并所有位点的结果，生成最终汇总表
final_batch_result <- bind_rows(batch_results) %>%
  arrange( 基因型关联 P 值 ) # 按关联 P 值升序排序（显著的在前）
# 查看最终汇总表
print( "\n==== 所有核心突变位点与基因型的批量关联分析结果 =====" )
print(final_batch_result, n = Inf) # n=Inf 显示所有行
# library(writexl)
# write_xlsx(final_batch_result, "突变位点与基因型关联分析批量结果.xlsx")
```

```
===== 突变位点 152 ( I152T ) =====
列联表 (基因型×携带情况):
```

	不携带	携带
B	114	34
C	193	0
D	12	0

检验结果 P值: 0

```
===== 突变位点 155 ( M155T ) =====
列联表 (基因型×携带情况):
```

	不携带	携带
B	114	34
C	193	0
D	12	0

检验结果 P值: 0

```
===== 突变位点 33 ( S33F S33L ) =====
列联表 (基因型×携带情况):
```

	不携带	携带
B	115	33
C	193	0
D	12	0

检验结果 P值: 0

15.3.4 显著突变位点和 HBV-DNA 水平的相关性

病毒载量 (DNA 水平) 与突变的相关性分析, 其前序步骤为, 比较突变组与无突变组的病毒载量中位数差异。因为病毒载量一般是非正态分布, 使用 Mann-Whitney U 检验差异的统计学显著性。这在基础数据统计部分已经完成。在此之后, 需要进一步分析, 哪些突变单独与病毒载量相关 ($P < 0.05$)。

```
# 接上节代码
```

```
library(dplyr)
```

```
library(tidyr)
```

```
library(ggplot2)
```

```
library(stringr)
```

```
# 1. 清洗 basic_info 的 HBV_DNA
```

```
basic_info_clean <- basic_info %>%
```

```
mutate(
```

```
  HBV_DNA_clean = str_remove_all(as.character(HBV_DNA), "[^0-9\\.\\-]") ,
```

```
HBV_DNA_clean = as.numeric(HBV_DNA_clean)
) %>%
filter(!is.na(HBV_DNA_clean))

# 2. 拆分 mutation_info 为长格式
mutation_info_long <- mutation_info %>%
  separate_rows(Mutation_Position, sep = " , ") %>%
  mutate(Mutation_Position = as.numeric(Mutation_Position))

# 3. 批量分析
core_mutations <- mutation_map %>%
  filter(P_Value <= 0.05 | Count >= 5) %>%
  pull(Position) %>%
  unique()
batch_results <- list()
for (pos in core_mutations) {
  pos_mutation_name <- mutation_map %>%
    filter(Position == pos) %>%
    pull(Mutation) %>%
    unique() %>%
    paste(collapse = "/" )
  # 直接用 mutation_info_long 连接 basic_info_clean
  dna_analysis <- mutation_info_long %>%
    mutate( 携带该位点 = ifelse(Mutation_Position == pos, "携带", "不携带" ))
  %>%
    right_join(basic_info_clean %>% select(Sample_ID, HBV_DNA_clean), by =
"Sample_ID" ) %>%
    mutate(
      携带该位点 = ifelse(is.na( 携带该位点 ), "不携带", 携带该位点 ),
      HBV_DNA_log = log10(HBV_DNA_clean + 1)
    )
}
```

```

wilcox_result <- wilcox.test(HBV_DNA_log ~ 携带该位点, data = dna_
analysis)
cat( "\n==== 位点", pos, "====\n" )
print(wilcox_result)
}

==== 位点 152 ====

      Wilcoxon rank sum test with continuity correction

data:  HBV_DNA_log by 携带该位点
W = 3693.5, p-value = 0.002209
alternative hypothesis: true location shift is not equal to 0

==== 位点 155 ====

      Wilcoxon rank sum test with continuity correction

data:  HBV_DNA_log by 携带该位点
W = 3693.5, p-value = 0.002209
alternative hypothesis: true location shift is not equal to 0

==== 位点 33 ====

      Wilcoxon rank sum test with continuity correction

data:  HBV_DNA_log by 携带该位点
W = 3684.5, p-value = 0.004217
alternative hypothesis: true location shift is not equal to 0

```

结果表明特定位点 33、152、155 突变和高 HBV-DNA 水平有显著相关性。

第 16 章 乙肝病毒基因突变与临床表型关联

16.1 突变与临床表征的关联

16.1.1 Upset 图可视化基因型与传播状态突变频数相关性

UpSetR 创建的 “UpSet” 图，是一种用于可视化集合交集的方法，特别适合于处理复杂的交集和大量集合。它是一个替代传统的 Venn 和 Euler 图的方法。在 “UpSet” 图中，“Intersection Size” 的条形图表示每个交集中元素的数量。Intersection Size 条形图表示每个交集的大小（即包含的元素数量）。更多内容可以参考：https://blog.csdn.net/weixin_53637133/article/details/138433880。

代码如下：

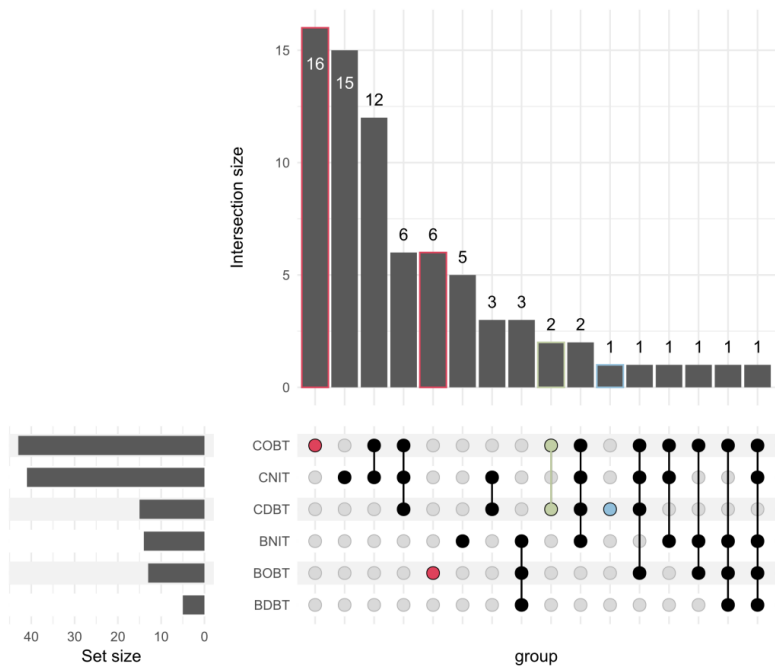
```
# 加载 ComplexUpset 包
library(ComplexUpset)

# 读取数据
data <- read.table(“F:/.../ 文 件 名 .txt” , header = FALSE, sep = “\t” ,
stringsAsFactors = FALSE)
colnames(data) <- c(“type” , “site” )
# 加载 dplyr 用于数据处理
library(dplyr)
library(tidyr)
# 将第一行设置为列名
colnames(data) <- data[1, ]
str(data)
# 移除第一行
data <- data[-1, ]
```



```
str(data)
# 制作二进制指示矩阵
binary_data <- data %>%
  mutate(value = 1) %>%
  spread(type, value, fill = 0)
str(binary_data) # 查看数据结构

# 生成 UpSet 图
upset(
  binary_data,
  intersect = colnames(binary_data)[-1],
  base_annotations=list(
    'Intersection size' =intersection_size(
      text=list(vjust=-1, hjust=0.5, angle=0),
      text_colors=c(on_background=' black' , on_bar=' white' )
    )
  ),
  queries=list(
    upset_query(intersect=c( "BOBT" ), color=" #de425b" ),
    upset_query(intersect=c( "COBT" ), color=" #de425b" ),
    upset_query(intersect=c( "CDBT" ), color=" #91bfdb" ),
    upset_query(intersect=c( "COBT" , "CDBT" ), color=" #c2cfa5" )
  )
)
# 导出数据表
library(writexl)
write_xlsx(binary_data," F:/.../.xlsx" )
```



16.1.2 突变模式和临床表征的 Pearson 相关

突变模式（mutation patterns）研究是一种承前启后的研究思路，从探查突变概况，到分析突变和临床表征的相关性，具备一定的逻辑联系。

现在使用 R 代码实现不同分组间突变模式相关性分析。

```
# 加载所需的库
library(readxl)
library(dplyr)
library(seqinr)
library(MutationalPatterns)
library(writexl)
library(ggplot2)
library(reshape2)
```

```
# 读取分组数据
group_data <- read_xlsx(“保存地址 /X.gene.BPT.group.xlsx” , sheet = 2)
# 从 FASTA 文件中读取序列
fasta_file <- “保存地址 /LYXseq1.fas”
sequences <- read.fasta(fasta_file, as.string = TRUE)

# 将序列名称提取为 ID，并存储为数据框
seq_data <- data.frame(
  ID = names(sequences),
  Sequence = apply(sequences, function(x) paste(x, collapse = “” ))
)
# 选择参考序列
reference_name <- “AY123041”
reference_sequence <- seq_data$Sequence[seq_data$ID == reference_name]
# 合并 Excel 数据和序列数据
merged_data <- merge(group_data, seq_data, by = “ID” )
# 初始化突变矩阵
num_samples <- nrow(merged_data)
mutation_matrix <- matrix(0, nrow = num_samples, ncol = 12)
colnames(mutation_matrix) <- c(“A → T” , “A → C” , “A → G” , “T → A” ,
“T → C” , “T → G” ,
“C → A” , “C → T” , “C → G” , “G → A” , “G → T” ,
“G → C” )

# 计算突变频数的函数
calculate_mutation_counts <- function(data, group_value, reference_sequence) {
  # 根据分组选择数据
  group_data <- data %>% filter(group == group_value)
  # 初始化突变计数
  mutation_counts <- numeric(12)
```

```

names(mutation_counts) <- colnames(mutation_matrix)
# 逐行遍历基因序列，计算突变
for (i in 1:nrow(group_data)) {
  sequence <- as.character(group_data[i, "Sequence" ]) # 读取合并后的序列
  # 遍历序列中的碱基
  for (j in 1:min(nchar(sequence), nchar(reference_sequence))) {
    ref_base <- toupper(substr(reference_sequence, j, j)) # 参考碱基
    seq_base <- toupper(substr(sequence, j, j))          # 序列碱基
    # 更新突变计数
    if (ref_base != seq_base) {
      mutation_counts[paste(ref_base, "→", seq_base, sep = " ")] <-
        mutation_counts[paste(ref_base, "→", seq_base, sep = " ")] + 1
    }
  }
}

return(mutation_counts)
}
# 计算 OBT 和 NBT 组的突变频数
OBT_freq <- calculate_mutation_counts(merged_data, 1, reference_sequence)
NBT_freq <- calculate_mutation_counts(merged_data, 0, reference_sequence)
# 输出结果
result <- data.frame(
  Mutation_Type = names(OBT_freq),
  OBT_Counts = OBT_freq,
  NBT_Counts = NBT_freq
)
print(result)
write_xlsx(result, "保存地址 /result.xlsx" )

```

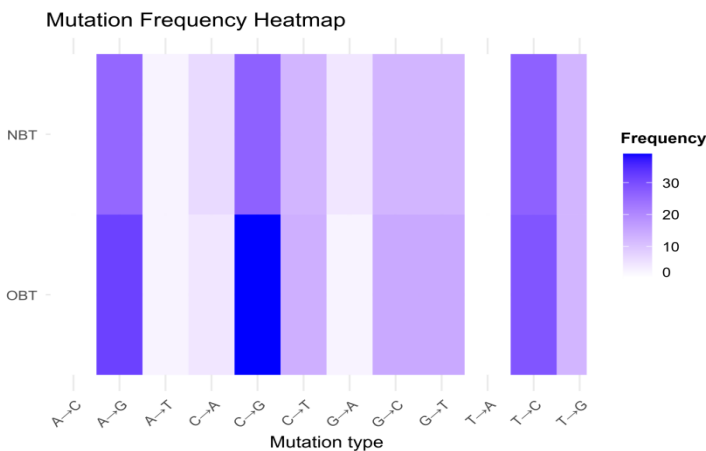
```
# 计算相对替换频率之间的相关系数及 P 值
correlation_result <- cor.test(OBT_freq, NBT_freq, method = "pearson" )
print(correlation_result)
# 创建一个数据框，用于散点图
scatter_data <- data.frame(
  OBT = OBT_freq,
  NBT = NBT_freq,
  Mutation_Type = names(OBT_freq) # 可选：保留突变类型名称用于标注
)
# 绘制散点图
ggplot(scatter_data, aes(x = OBT, y = NBT)) +
  geom_point(size = 3, color = "blue" ) + # 点的颜色和大小
  geom_smooth(method = "lm" , color = "red" , se = FALSE) + # 添加线性
回归线

labs(title = "Scatter Plot of OBT vs NBT Mutation Frequencies" ,
  x = "OBT Mutation Frequencies" ,
  y = "NBT Mutation Frequencies" ) +
theme_minimal() +
theme(text = element_text(size = 12)) # 设置字体大小

# 准备绘制热图的数据
heatmap_data <- data.frame(
  Mutation_Type = names(OBT_freq),
  OBT = OBT_freq,
  NBT = NBT_freq
)

# 将数据转化为长格式
heatmap_long <- melt(heatmap_data, id.vars = "Mutation_Type" )
# 绘制热图 ( guides(fill = FALSE) 不显示图例; fill = "Frequency" 显示图例 )
```

```
ggplot(heatmap_long, aes(x = Mutation_Type, y = variable, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "blue") +
  labs(title = "Mutation Frequency Heatmap",
       x = "Mutation type",
       y = "Groups",
       fill = "Frequency") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) #+
  guides(fill = "none") # 不显示图例
```



OBT 组和 DBT 组 HBV X 基因核苷酸置换频率

Algorithms for determining HBV subtype from the primary structure of the HBV S gene.

下表为 OBT 和 NBT 组 X 基因核苷酸突变相关性表。“相关系数”用于衡量两组相对替换频率（即不同传播分组的突变模式数据）之间的线性相关程度，而“P 值”则用于检验这种相关性是否具有统计学意义（通常 $P < 0.05$ 被认为相关显著）。通过这两个指标，可客观比较不同传播分组在突变模式上的相似性高低。结果显示不同传播分组 X 基因核苷酸突变模式高度相似。

Mutation_Type	OBT_Counts	NBT_Counts	t	P
A → T	2	2	15.922 ^a	< 0.001
A → C	0	0		
A → G	31	25		
T → A	0	0		
T → C	28	26		
T → G	12	12		
C → A	4	6		
C → T	13	12		
C → G	38	26		
G → A	2	4		
G → T	14	12		
G → C	14	12		

^a: Pearson's product-moment correlation coefficient

16.1.3 显著突变位点和有序分类临床表征的相关性

```
# 加载必备包
library(dplyr)
library(tidyr)
library(ggplot2)

# (1) 定义批量分析位点
core_mutations <- mutation_map %>%
  filter(P_Value <= 0.05 | Count >= 5) %>% # 显著位点或高频位点
  pull(Position) %>%
  unique()

# (2) 创建空列表存储结果（列联表 + 检验结果 + 汇总表）
batch_contingency <- list() # 存储每个位点的列联表
batch_test_results <- list() # 存储每个位点的 Fisher 检验结果
batch_summary <- list() # 存储汇总信息

# (3) 循环遍历每个核心位点，批量分析与传播分组的相关性
for (pos in core_mutations) {
  # (3.1) 提取该位点的突变名称（用于结果标注）
```

```

pos_mutation_name <- mutation_map %>%
  filter(Position == pos) %>%
  pull(Mutation) %>%
  unique() %>%
  paste(collapse = "/" )
# 2. 新建临时存储变量 pos
pos_data <- final_data %>%
  select(Sample_ID, Mutation_Position) %>%
  distinct() %>%
  mutate( 携带该位点 = ifelse(Mutation_Position == pos, 1, 0)) %>% # 替换原
“携带位点 33”
# 连接传播分组数据
right_join(
  basic_info %>% select(Sample_ID, 传播分组 ),
  by = "Sample_ID"
) %>%
  mutate( 携带该位点 = ifelse(is.na( 携带该位点 ), 0, 携带该位点 )) # NA 视
为不携带
# 3. 构建列联表 ( 传播分组 × 携带情况 )
contingency_table <- table(pos_data$ 传播分组 , pos_data$ 携带该位点 )
colnames(contingency_table) <- c( “不携带” , “携带” ) # 统一列名, 便于
查看
# 4. 统计检验: Fisher 精确检验
test_result <- fisher.test(contingency_table)
# 5. 计算各传播分组的携带率 ( 补充信息, 便于解读 )
carriage_rate <- pos_data %>%
  group_by( 传播分组 ) %>%
  summarise(
    分组样本数 = n(),
    携带样本数 = sum( 携带该位点 ),

```



```
    携带率 = round( 携带样本数 / 分组样本数 , 3),  
    .groups = "drop"  
  )  
# 6. 存储结果  
batch_contingency[[as.character(pos)]] <- contingency_table  
batch_test_results[[as.character(pos)]] <- test_result  
batch_summary[[as.character(pos)]] <- data.frame(  
  突变位点编号 = pos,  
  突变名称 = pos_mutation_name,  
  传播分组 1 携带率 = filter(carriage_rate, 传播分组 == 1)$ 携带率 ,  
  传播分组 2 携带率 = filter(carriage_rate, 传播分组 == 2)$ 携带率 ,  
  传播分组 3 携带率 = ifelse(nrow(filter(carriage_rate, 传播分组 == 3)) > 0,  
    filter(carriage_rate, 传播分组 == 3)$ 携带率 , 0),  
  Fisher_P 值 = round(test_result$p.value, 4),  
  关联是否显著 = ifelse(test_result$p.value <= 0.05, " 是 ", " 否 ")  
)  
  
# 打印每个位点的结果  
cat( "\n==== 突变位点" , pos, " ( " , pos_mutation_name, " ) =====\n" )  
cat( "列联表 ( 传播分组 × 携带情况 ) : \n" )  
print(contingency_table)  
cat( "各传播分组携带率: \n" )  
print(carriage_rate)  
cat( "Fisher 精确检验结果 ( P 值 ) : " , round(test_result$p.value, 4), "\n" )  
cat( "关联是否显著: " , ifelse(test_result$p.value <= 0.05, "是", "否"), "\n" )  
}
```

```

===== 突变位点 152 ( I152T ) =====
列联表 (传播分组×携带情况):

      不携带 携带
1      139    9
2      144   20
3       36    5
各传播分组携带率:
# A tibble: 3 × 4
  传播分组  分组样本数  携带样本数  携带率
    <dbl>      <int>      <dbl>    <dbl>
1         1         148          9  0.061
2         2         164         20  0.122
3         3          41          5  0.122
Fisher精确检验结果 (P值):  0.1368
关联是否显著: 否

===== 突变位点 155 ( M155T ) =====
列联表 (传播分组×携带情况):

      不携带 携带
1      139    9
2      144   20
3       36    5
各传播分组携带率:
# A tibble: 3 × 4
  传播分组  分组样本数  携带样本数  携带率
    <dbl>      <int>      <dbl>    <dbl>
1         1         148          9  0.061
2         2         164         20  0.122
3         3          41          5  0.122
Fisher精确检验结果 (P值):  0.1368
关联是否显著: 否

```

16.1.4 突变位点和有序分类临床表征的 Kendall 相关性

Kendall 检验一般用于有序多分类资料一致性检验。按照传播状态，不传播、隐性传播、显性传播呈现等级顺序关系，可以使用 Kendall 检验。

```
# 加载必要的包
```

```
library(readxl) # 读原始表
```

```
library(dplyr) # 数据处理
```

```
library(tidyr) # 拆分突变位点
```

```
# 导入数据表
```

```
mutation_info <- read_excel(“保存地址 /mutation_info_corrected.xlsx”)
```

```
colnames(mutation_info) <- c(“Sample_ID”, “Mutation_Position”) # 统一
```

列名

```
basic_info <- read_excel(“保存地址 /基础信息.xlsx”)
```

```
basic_info <- basic_info %>% select(ID, 传播分组) %>% rename(Sample_ID =
```

```
ID) # 统一样本 ID
# 拆分突变位点
mutation_long <- mutation_info %>%
  separate_rows(Mutation_Position, sep = ", ") %>%
  mutate(
    Sample_ID = as.character(Sample_ID),
    Mutation_Position = as.numeric(Mutation_Position)
  ) %>%
  filter(!is.na(Mutation_Position))
# 定义要批量分析的核心突变位点
# (若有“突变分析结果.csv”，可替换为：read.csv(“突变分析结果.csv”)
%>% filter(Count ≥ 5) %>% pull(Position) )
core_sites <- unique(mutation_long$Mutation_Position) # 先默认分析所有有效
# 位点，可按需筛选
core_sites <- core_sites[!is.na(core_sites)] # 过滤 NA
cat(“批量分析的核心突变位点：”, core_sites, “\n” )
# 初始化空列表，存储每个位点的结果
batch_results <- list()
for (site in core_sites) { # 循环遍历每个核心位点
  # (1): 标记当前位点的携带情况
  mut_mark <- mutation_long %>%
    mutate( 携带该位点 = ifelse(Mutation_Position == site, 1, 0)) %>%
    select(Sample_ID, 携带该位点 ) %>%
    distinct()
  # (2): 匹配传播分组
  mut_final <- basic_info %>%
    left_join(mut_mark, by = “Sample_ID” ) %>%
    mutate(
      携带该位点 = ifelse(is.na( 携带该位点 ), 0, 携带该位点 ),
      传播分组 = as.integer( 传播分组 )
    )
}
```

```

) %>%
filter(!is.na( 传播分组 )) %>%
mutate( 传播分组_有序 = factor( 传播分组 , levels = c(1,2,3), labels =
c( "NBT" , " OBT" , " DBT" )))
# (3): Kendall tau 分析
if (length(unique(mut_final$ 携带该位点 )) == 2) {
  kendall_res <- cor.test(
    x = mut_final$ 携带该位点 ,
    y = as.integer(mut_final$ 传播分组_有序 ),
    method = "kendall"
  )
# (4): 计算各传播分组携带率
carriage_rate <- mut_final %>%
  group_by( 传播分组_有序 ) %>%
  summarise(
    样本数 = n(),
    携带样本数 = sum( 携带该位点 ),
    携带率 = round( 携带样本数 / 样本数 *100, 2),
    .groups = "drop"
  )
# (5): 存储当前位点的结果
batch_results[[as.character(site)]] <- list(
  突变位点 = site,
  Kendall_tau = round(kendall_res$estimate, 4),
  P 值 = round(kendall_res$p.value, 4),
  关联是否显著 = ifelse(kendall_res$p.value <= 0.05, " 是 ", " 否 "),
  各分组携带率 = carriage_rate
)
# 打印当前位点结果查看进度
cat( "\n==== 突变位点" , site, "分析结果 =====\n" )

```

```
cat( "Kendall tau 系数: ", round(kendall_res$estimate, 4), "\n" )
cat( "P 值: ", round(kendall_res$p.value, 4), " | 关联是否显著: ",
ifelse(kendall_res$p.value <= 0.05, "是", "否" ), "\n" )
print(carriage_rate)
} else {
# 若某位点只有“携带”或“不携带”一组数据，跳过分析并提示
cat( "\n===== 突变位点", site, "===== \n" )
cat( "数据不足（只有携带或不携带一组），跳过分析 \n" )
}
}
# 整理结果为数据框
summary_table <- lapply(batch_results, function(res) {
data.frame(
突变位点 = res$ 突变位点 ,
Kendall_tau 系数 = res$Kendall_tau,
P 值 = res$P 值 ,
关联是否显著 = res$ 关联是否显著 ,
NBT 组样本数 = res$ 各分组携带率 [res$ 各分组携带率 $ 传播分组 _有序
==" NBT" , "样本数" ],
NBT 组携带率 = res$ 各分组携带率 [res$ 各分组携带率 $ 传播分组 _有序
==" NBT" , "携带率" ],
OBT 组样本数 = res$ 各分组携带率 [res$ 各分组携带率 $ 传播分组 _有序
==" OBT" , "样本数" ],
OBT 组携带率 = res$ 各分组携带率 [res$ 各分组携带率 $ 传播分组 _有序
==" OBT" , "携带率" ],
DBT 组样本数 = res$ 各分组携带率 [res$ 各分组携带率 $ 传播分组 _有序
==" DBT" , "样本数" ],
DBT 组携带率 = res$ 各分组携带率 [res$ 各分组携带率 $ 传播分组 _有序
==" DBT" , "携带率" ],
check.names = FALSE
```

```
)  
}) %>% bind_rows() %>% arrange(P 值) # 按 P 值升序排序，显著结果在前  
  
# 保存汇总表到 Excel  
write_excel_csv(summary_table, “保存地址 / 突变位点与传播分组 Kendall 批量结果.csv” )  
cat( “\n 汇总表已保存到：突变位点与传播分组 Kendall 批量结果.csv\n” )
```

突变位点	W	P
152	0.1395	0.0179
155	0.1395	0.0179
33	0.1348	0.0225

P<0.05 表示 W 系数不为零，且具有统计学意义。

16.1.5 突变位点和无序多分类临床表征的年龄调整 logistic 回归

研究突变和乙肝五项检查结果的关系，主要是看自变量对每个结局与参照组的差异，每个样本相互独立，因此使用无序多分类 Logistic 回归法。在这个过程中，需要考虑混杂因素的影响。这么做的意义在于，通过控制常规潜在混杂因素，以满足统计学严谨性的要求。其次，即使混杂因素没有造成显著影响，通过控制混杂，也可以清除其造成的微小影响，使统计结果更接近“真实效应”。本节通过对年龄进行统一，分别对处理前和处理后进行分析，确认年龄不是显著的混杂因素，并进行了年龄调整的 logistic 回归。

```
# 加载必要的包  
library(readxl)  
library(dplyr)  
library(nnet)          # 多分类回归  
library(broom)         # 处理回归结果  
library(ggplot2)  
library(car)           # VIF 检验  
library(psych)         # 描述性统计辅助  
library(writexl)
```

```
# 1. 读取并合并数据

# 读取表 1
base_data <- read_excel(“保存地址 / 基础信息.xlsx”) %>%
  select(ID, 乙肝五项, 母亲年龄, HBV_DNA) %>% # 提取核心字段
  filter(!is.na(乙肝五项), !is.na(母亲年龄), !is.na(HBV_DNA)) # 删除关键
缺失

# 读取表 2（同一 ID 可能对应多个突变，标记为“有突变（赋值 1）避免发
生重复，导致运行错误”）
# 选择突变频率较高的位点统计
table_b <- read_excel(“保存地址 / 对照表.xlsx”) %>%
  group_by(ID) %>%
  summarise(
    has_target_mutation = ifelse(any(Mutation %in% c(“V10A”, “F32I”, “S33L”,
“I152T”, “M155T”)), 1, 0),
    .groups = “drop”
  )

## 1.2 合并数据
merged_data <- left_join(
  base_data,
  table_b,
  by = “ID”
) %>%
  mutate(
    has_target_mutation = ifelse(is.na(has_target_mutation), 0, has_target_
mutation),
    # 处理 HBV_DNA=0 的情况（避免 log10(0) 报错，替换为 1）
    HBV_DNA = ifelse(HBV_DNA == 0, 1, HBV_DNA)
  )
```

```
print(paste0(“合并后总样本数：”, nrow(merged_data)))
print(“\n 前 5 个样本的突变标记：” )
print(merged_data %>% select(ID, 乙 肝 五 项 , has_target_mutation) %>%
head(5))
```

2. 数据预处理（对无序分类资料重新编码，避免格式错误）

```
analysis_data <- merged_data %>%
mutate(
  cld_stage = case_when(
    乙肝五项 == 1 ~ 1, # 1=HBsAg 单阳
    乙肝五项 == 2 ~ 2, # 2= 小三阳
    乙肝五项 == 3 ~ 3, # 3= 大三阳
    TRUE ~ NA_real_   # 其他数字标记为缺失
  ),
  s_mutation = has_target_mutation, # 替换：x_mutation → s_mutation（对应
S 基因）
  hbv_load_log = log10(HBV_DNA)
) %>%
# 筛选有效样本（再次确保已写入的数据表没有 NA，且样本数 ≠ 0）
filter(
  !is.na(cld_stage),
  !is.na(s_mutation),
  !is.na(hbv_load_log)
)
```

3. 验证样本数（运行前再次确认数据表正确）

```
print(paste0(“\n 最终可分析样本数：”, nrow(analysis_data)))
if(nrow(analysis_data) == 0) stop(“仍无样本！请检查乙肝五项是否只有 1/2/3
三个值” )
```



```
# 4. 描述性统计（以 HBsAg 单阳为对照组）
# 4.1 计算均值、中位数
desc_stats <- analysis_data %>%
  group_by( 乙肝五项 , cld_stage) %>%
  summarise(
    样本量 = n(),
    母亲平均年龄_SD = paste0(round(mean( 母亲年龄 ), 1), " ± ", round(sd( 母
母亲年龄 ), 1)),
    有突变样本数 = sum(s_mutation == 1), # 替换： x_mutation → s_mutation
    突 变 率_percent = round(mean(s_mutation) * 100, 1), # 替 换： x_
mutation → s_mutation
    病毒载量_log 中位数 = round(median(hbv_load_log), 2),
    .groups = "drop"
  ) %>%
  mutate(
    乙肝五项标签 = case_when(
      乙肝五项 == 1 ~ "HBsAg 单阳 ",
      乙肝五项 == 2 ~ " 小三阳 ",
      乙肝五项 == 3 ~ " 大三阳 "
    )
  )
print( "\n=== 描述性统计结果 ===" )
print(desc_stats %>% select( 乙肝五项标签 , 样本量 , 母亲平均年龄_SD, 突变
率_percent))
# 4.2 单因素关联：突变与乙肝五项
contingency_table <- table(
  突变状态 = factor(analysis_data$s_mutation, levels = c(0,1), labels = c(" 无突
变 ", " 有突变 ")),
  乙肝五项 = factor(analysis_data$ 乙肝五项 , labels = c("HBsAg 单阳 ", " 小三
阳 ", " 大三阳 "))
```

```

)
print( "\n=== 突变与乙肝五项列联表 ===" )
print(contingency_table)
# 统计检验
if (any(contingency_table < 5)) {
  mutation_test <- fisher.test(contingency_table)
  test_type <- "Fisher 精确检验"
} else {
  mutation_test <- chisq.test(contingency_table)
  test_type <- " $\chi^2$  检验"
}
print(paste0( "\n=== 突变率组间差异 ( ", test_type, " ) ===" ))
print(paste0( "P 值: ", round(mutation_test$p.value, 3)))
# 4.3 病毒载量分析
# 突变组 vs 无突变组
virus_mutation <- wilcox.test(hbv_load_log ~ s_mutation, data = analysis_data)
print( "\n=== 突变组 vs 无突变组病毒载量 P 值 ===" )
print(round(virus_mutation$p.value, 3))
# 不同乙肝五项的病毒载量
virus_hep <- kruskal.test(hbv_load_log ~ as.factor( 乙肝五项 ), data = analysis_data)
print( "\n=== 不同乙肝五项病毒载量 P 值 ===" )
print(round(virus_hep$p.value, 3))
# 4.4 多分类 Logistic 回归 ( 调整母亲年龄 )
analysis_data$cld_stage_factor <- factor(
  analysis_data$cld_stage,
  levels = c(1,2,3),
  labels = c( "HBsAg 单阳" , "小三阳" , "大三阳" )
)
# 4.5 构建多分类 Logistic 回归模型

```

```
multinom_model <- multinom(  
  cld_stage_factor ~ s_mutation + 母亲年龄, # 替换: x_mutation → s_mutation  
  data = analysis_data,  
  trace = FALSE,  
  maxit = 1000  
)  
  
# 5. 使用 broom 包提取 OR 值和 95%CI  
# 5.1 提取系数并转换为 OR 值  
model_coef <- tidy(multinom_model, exponentiate = TRUE, conf.int = TRUE)  
# 5.2 整理结果  
or_result <- model_coef %>%  
  mutate(  
    # 明确结局变量 (参照组: HBsAg 单阳)  
    结局变量 = case_when(  
      y.level == "小三阳" ~ "小三阳 vsHBsAg 单阳",  
      y.level == "大三阳" ~ "大三阳 vsHBsAg 单阳"  
    ),  
    # 合并 OR 值与 95%CI (文字描述中保留 %, 列名中删除)  
    OR_CI = paste0(round(estimate, 3), "(95%CI:", round(conf.low, 3), "-",  
round(conf.high, 3), ")"),  
    # 保留 3 位小数的 P 值  
    P 值 = round(p.value, 3)  
  ) %>%  
  select(  
    结局变量,  
    自变量 = term,  
    OR 值 = estimate,  
    CI_low = conf.low, # 用 CI_low 替代 "95%CI 下限"  
    CI_high = conf.high, # 用 CI_high 替代 "95%CI 上限"
```

```

    OR_CI,
    P 值
  ) %>%
  mutate( 自变量 = ifelse( 自变量 == "s_mutation", "S 基因突变", 自变量 ))
# 5.3 输出回归结果
print( "\n=== 多分类 Logistic 回归结果 ( 调整母亲年龄 ) ===" )
print(or_result)
# 6. 保存到 Excel
write_xlsx(or_result, "保存地址 / 孕妇 HBVS 基因突变和血清标志物 OR 分
析.xlsx" )

```

7. 使用【森林图】可视化

#7.1 传递数据表

```

forest_data <- or_result %>%
  filter( 自变量 != "(Intercept)" ) %>% # 排除无意义的截距项
  mutate(
    自变量标签 = case_when(
      自变量 == "S 基因突变" ~ "S 基因突变", # 替换: X 基因 → S 基因
      自变量 == "母亲年龄" ~ "母亲年龄 ( 每增长 1 岁 )"
    ),
    结局变量简化 = case_when(
      结局变量 == "小三阳 vsHBsAg 单阳" ~ "→ 小三阳",
      结局变量 == "大三阳 vsHBsAg 单阳" ~ "→ 大三阳"
    ),
    显著性 = ifelse(P 值 < 0.05, "显著 ( P<0.05 )", "不显著 ( P ≥ 0.05 )" )
  )

```

#7.2 定义图形风格

```

theme_paper <- theme_bw() +
  theme(
    plot.title = element_text(size = 12, hjust = 0.5, face = "bold" ),

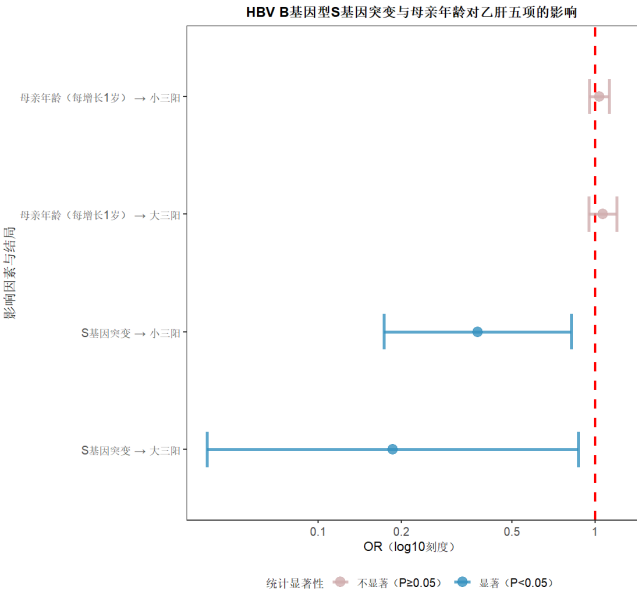
```

```
axis.title = element_text(size = 11),
axis.text = element_text(size = 10),
legend.title = element_text(size = 11),
legend.text = element_text(size = 10),
panel.grid = element_blank(), # 无网格线，适配论文
legend.position = "bottom" # 图例放底部
)

# 7.3 使用 ggplot2 绘制森林图
p1 <- ggplot(forest_data, aes(x = OR 值, y = paste( 自变量标签, 结局变量简化,
sep = " "))) +
  geom_vline(
    xintercept = 1,
    linetype = "dashed",
    color = "red",
    linewidth = 1 # 新版本用 linewidth 控制线宽
  ) +
  geom_point(aes(color = 显著性), size = 4, alpha = 0.8) +
  geom_errorbar(
    aes(xmin = CI_low, xmax = CI_high, color = 显著性),
    orientation = "y", # 横向误差线，替代弃用的 geom_errorbarh
    height = 0.3, # 误差线两端高度
    linewidth = 1.2, # 误差线宽度，替代 size
    alpha = 0.8
  ) +
  # 标注 OR 值和 P 值
  geom_text(
    aes(label = paste0( "OR=" , round(OR 值, 3), "\nP=" , round(P 值, 3))),
    x = max(forest_data$CI_high) * 1.1, # 文本放 CI 右侧，避免超出
    size = 3.2,
```

```
hjust = 0
)+
scale_color_manual(values = c( "#27ae60" , "#e74c3c" )) +
# X 轴对数刻度
scale_x_log10(
  breaks = c(0.1, 0.2, 0.5, 1, 2, 5),
  labels = c( "0.1" , "0.2" , "0.5" , "1" , "2" , "5" )
)+
labs(
  title = "S 基因突变与母亲年龄对乙肝五项进展的影响（OR 值森林图）",
  x = " 比值比（OR，log10 刻度）",
  y = " 影响因素与结局 ",
  color = "统计显著性"
)+
theme_paper
```

p1



对于 HBV B 基因型 S 基因，在调整母亲年龄后，采用多分类 Logistic 回归分析显示，V10A、F32I、S33L、I152T、M155T 突变与“HBsAg 单阳向小三阳进展”显著相关（OR=0.377，95% CI=0.173-0.820，P=0.014），可使进展风险降低 62.3%；这些突变同时与“HBsAg 单阳向大三阳进展”显著相关（OR=0.186，95% CI=0.0396-0.870，P=0.033），可使进展风险降低 81.4%；母亲年龄对乙肝五项类型的进展无显著影响（P 值均 > 0.05）。提示在本研究的孕妇群体中，HBV B 基因型 S 基因突变可能使感染状态倾向于 HBsAg 单阳，相对于发生大三阳和小三阳是一种保护因素。

16.2 突变位点之间的相关性

16.2.1 使用 Pearson 法检验突变位点共现性

从数据角度看，如 A、B 位点突变都与某个或某些临床表征相关，近似于“连锁”的概念，被称为共现（co-occurred）。在这种情况下，有可能 A 和 B 是一起突变的，但实际上和临床表征相关的是 B 位点突变。若不管这个问题，可能会有假阳性的风险。此外，从生物学意义上，这种情况也是有可能发生的。比如某些突变位点距离很近，它们可能在蛋白质三维结构上具备相互作用的可能，比如可能对蛋白质稳定性、极性等特性产生拮抗或强化；可能是在进化过程中保留下的突变组合，有不少研究重点关注这些“双突变”“三突变”等情况。这项研究具有很好的应用前景，如通过协同突变现象，设计筛查标志物组合，提高诊断准确率，避免偶然的单位点突变造成假阳性；协同突变可能与疾病临床严重程度有关，其确认可能可以为临床决策提供支持。

部分数据表如下图：

	A	B	C
1	Sequence	Mutation_Position	
2	003	10, 33, 152, 155	
3	014	10, 33, 152, 155	
4	023	10, 33, 152, 155	
5	029	33, 152, 155	
6	030	33, 145, 152, 155	
7	037	33, 152, 155	
8	053	33, 152, 155	
9	089	33, 152, 155	
10	090	33, 152, 155	
11	096	33, 152, 155	
12	106	32, 34, 35, 152, 155	
13	118	32, 33, 34, 35, 152, 155	
14	119	32, 33, 34, 35, 36, 152, 155	
15	127	33, 152, 155	
16	128	33, 152, 155	
17	129	33, 152, 155	
18	130	33, 152, 155	

代码如下：

```
library(readxl)
library(dplyr)
library(tidyr)
library(pheatmap) # 热图绘图
library(reshape2) # 矩阵转换
```

1. 读取数据表

```
df <- read_excel(“保存地址 /mutation_info_corrected.xlsx”)
```

colnames(df) <- c(“Sequence”, “Mutation_Position”) # 统一列名，防止读取异常

```
head(df)
```

2. 突变率表转换为 0/1 矩阵

```
df_split <- df %>%
```

separate_rows(Mutation_Position, sep = “，”) %>% # 拆分逗号分隔的突变位点

```
mutate(Mutation_Position = as.numeric(Mutation_Position)) %>%
```

```
mutate(mut_status = 1) # 标记：1= 该样本此位点发生突变，0= 未突变
```

3. 生成宽矩阵并转为 data.frame，避免 tibble 行名警告

```
mut_matrix <- df_split %>%
```

pivot_wider(names_from = Mutation_Position, values_from = mut_status, values_fill = 0) %>%

```
as.data.frame()
```

4. 设置行名并删除 Sequence 列，得到纯数值矩阵

```
rownames(mut_matrix) <- mut_matrix$Sequence
```

```
mut_matrix <- mut_matrix %>% select(-Sequence)
```

5. 只保留同时包含 0 和 1 的突变位点，即双水平有效位点

```
mut_matrix <- mut_matrix[, sapply(mut_matrix, function(x) length(unique(x))
== 2)]
```

如果没聚类补充的排版优化：突变位点按数字从小到大排序

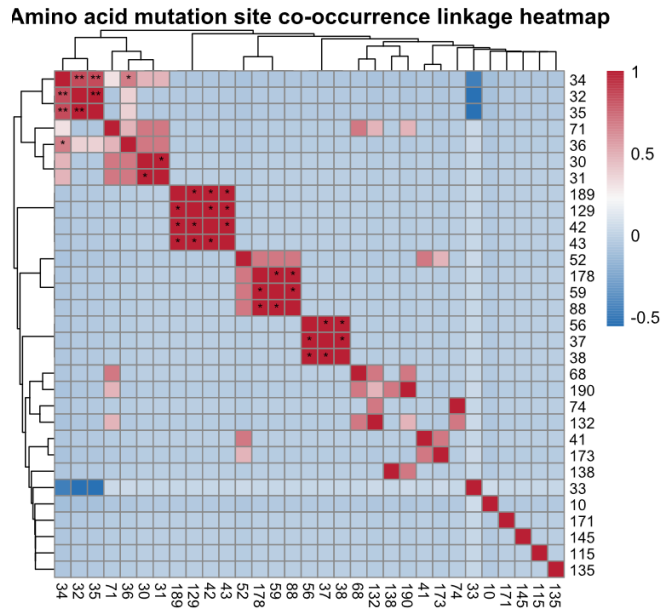

```
mut_matrix <- mut_matrix[,order(as.numeric(colnames(mut_matrix)))]  
# 查看过滤后的有效矩阵  
print(“过滤后有效突变位点 0/1 矩阵（无任何单水平位点）：”)  
print(head(mut_matrix))  
print(paste0(“剩余有效突变位点数：”, ncol(mut_matrix)))  
# 6. 计算位点共现系数矩阵，如果有全 0 或全 1 位点会报错“无标准差”  
co_occur_matrix <- cor(mut_matrix, method = “pearson”)  
# 7. 计算 Fisher 精确检验 P 值矩阵  
n_sites <- ncol(mut_matrix)  
p_matrix <- matrix(NA, nrow = n_sites, ncol = n_sites) # 初始化 P 值矩阵  
rownames(p_matrix) <- colnames(co_occur_matrix)  
colnames(p_matrix) <- colnames(co_occur_matrix)  
# 双层循环计算每对位点的 Fisher 独立性检验  
for (i in 1:n_sites) {  
  for (j in 1:n_sites) {  
    if (i != j) {  
      fisher_res <- fisher.test(mut_matrix[,i], mut_matrix[,j])  
      p_matrix[i,j] <- fisher_res$p.value  
    }  
  }  
}  
# 8. 生成显著性标注矩阵  
sig_matrix <- matrix(“”, nrow = n_sites, ncol = n_sites)  
rownames(sig_matrix) <- colnames(co_occur_matrix)  
colnames(sig_matrix) <- colnames(co_occur_matrix)  
sig_matrix[p_matrix < 0.05 & p_matrix >= 0.01] <- “*” # P<0.05 显著连锁  
sig_matrix[p_matrix < 0.01] <- “**” # P<0.01 极显著连锁  
# 9. 绘制突变位点连锁共现热图  
pheatmap(  
  mat = co_occur_matrix, # 热图主体：位点共现程度（越大越红 = 连锁
```

越强)

```

display_numbers = sig_matrix, # 热图标注显著性星号
number_color = "black",      # 星号颜色
number_fontsize = 10,        # 星号字体大小
fontsize_row = 9,            # 位点行字体大小
fontsize_col = 9,            # 位点列字体大小
border_color = "gray50",     # 热图边框颜色
clustering_method = "complete", # 聚类方法
clustering_distance_rows = "euclidean",
clustering_distance_cols = "euclidean",
# 配色说明: 蓝色 = 无共现, 白色 = 中等, 红色越深 = 共现 / 连锁越紧密
color = colorRampPalette(c("#2166AC", "#F7F7F7", "#B2182B"))(100),
main = "Amino acid mutation site co-occurrence linkage heatmap",
treeheight_row = 20,
treeheight_col = 20
)
library(writexl)
wb <- createWorkbook()
addWorksheet(wb, "过滤后有效突变 01 矩阵")
writeData(wb, "过滤后有效突变 01 矩阵", mut_matrix)
addWorksheet(wb, "位点共现系数矩阵")
writeData(wb, "位点共现系数矩阵", co_occur_matrix)
addWorksheet(wb, "Fisher 检验 P 值矩阵")
writeData(wb, "Fisher 检验 P 值矩阵", p_matrix)
saveWorkbook(wb, "保存地址 / 突变位点连锁分析结果.xlsx", overwrite =
TRUE)

```



16.2.2 编制突变位点共现性占比汇总表

```
# 加载必要的包
library(readxl)
library(dplyr)
library(tidyr)
library(writexl)

# 1. 读取并预处理数据
df <- read_excel(“保存地址 /mutation_info_corrected.xlsx” )
colnames(df) <- c(“Sequence”, “Mutation_Position” )# 统一列名（样本
ID+ 突变位点）
# 拆分逗号分隔的突变位点 → 一行一个 “样本 - 突变位点”
df_split <- df %>%
  separate_rows(Mutation_Position, sep = “,”) %>%
  mutate(
```

```

Sequence = as.character(Sequence),    # 样本 ID 转为字符
Mutation_Position = as.numeric(Mutation_Position) # 突变位点转为数值
) %>%
filter(!is.na(Mutation_Position)) # 过滤无效突变位点
# 统计每个突变位点的总携带样本数（后续计算占比用）
site_total <- df_split %>%
  group_by(Mutation_Position) %>%
  summarise( 位点总携带样本数 = n_distinct(Sequence), .groups = "drop" )
# 总样本数（后续计算占比用）
total_samples <- n_distinct(df$Sequence)
cat( "总样本数: ", total_samples, "\n" )
# 2. 统计任意两个突变位点的共现样本数
# 生成“样本 - 突变位点”宽矩阵（行 = 样本，列 = 突变位点，值 = 1/0）
mut_matrix <- df_split %>%
  mutate(mut_status = 1) %>%
  pivot_wider(
    names_from = Mutation_Position,
    values_from = mut_status,
    values_fill = 0,
    id_cols = Sequence
  ) %>%
  as.data.frame()
rownames(mut_matrix) <- mut_matrix$Sequence
mut_matrix <- mut_matrix %>% select(-Sequence) # 移除样本 ID 列，保留纯
数值矩阵
# 提取所有有效突变位点（至少有 1 个样本携带）
sites <- colnames(mut_matrix)
n_sites <- length(sites)

# 3. 计算共现样本数、占比

```

```
co_occur_table <- data.frame(  
  共突变位点组合 = character(),  
  位点 1 = character(),  
  位点 2 = character(),  
  位点 1 共现样本数 = integer(),  
  位点 1 总携带样本数 = integer(),  
  共现样本数 = integer(),  
  占位点 1 总携带样本数的比例 = numeric(), # 核心比例 (= 位点 1 共现样本  
数 / 位点 1 总携带样本数 × 100%)  
  占总样本数的比例 = numeric(), # 占总样本数的比例  
  stringsAsFactors = FALSE  
)  
  
# 双层循环：遍历所有位点对 (i < j, 避免重复)  
for (i in 1:(n_sites-1)) {  
  for (j in (i+1):n_sites) {  
    site1 <- sites[i]  
    site2 <- sites[j]  
    # 1. 计算核心数值  
    co_occur_n <- sum(mut_matrix[, site1] == 1 & mut_matrix[, site2] == 1) # 共  
现样本数  
    site1_total_n <- filter(site_total, Mutation_Position == site1)$ 位点总携带样本数  
    # 位点 1 总携带数  
    # 2. 计算比例  
    ratio_site1 <- ifelse(site1_total_n > 0, round(co_occur_n / site1_total_n * 100,  
2), 0) # 核心比例  
    ratio_total <- round(co_occur_n / total_samples * 100, 2) # 占总样本比例  
    # 3. 保存到结果表格 (新增列对应计算过程)  
    co_occur_table <- rbind(co_occur_table, data.frame(  

```

```

共突变位点组合 = paste0(" 位点 ", site1, "+", " 位点 ", site2),
位点 1 = paste0("位点", site1),
位点 2 = paste0("位点", site2),
位点 1 共现样本数 = co_occur_n, # 计算比例的分子
位点 1 总携带样本数 = site1_total_n, # 计算比例的分母
共现样本数 = co_occur_n,
占位点 1 总携带样本数的比例 = ratio_site1, # 分子 / 分母 × 100%
占总样本数的比例 = ratio_total
))
}
}
# 4. 整理表格格式
final_table <- co_occur_table %>%
  filter(共现样本数 > 0) %>% # 只保留有共现的组合
  select(
    "共突变位点组合" = 共突变位点组合,
    "位点 1" = 位点 1,
    "位点 2" = 位点 2,
    "位点 1 共现样本数 (分子)" = 位点 1 共现样本数,
    "位点 1 总携带样本数 (分母)" = 位点 1 总携带样本数,
    "共现样本数" = 共现样本数,
    "% of 位点 1 总携带样本数" = 占位点 1 总携带样本数的比例,
    "% of 总样本数" = 占总样本数的比例
  ) %>%
  arrange(desc(`共现样本数`)) # 按共现样本数降序排列
# 5. 输出结果
View(final_table)
write_xlsx(final_table, "保存地址 / 突变位点共现统计表格.xlsx")

```

共突变位点组合	No. of patients	% of 位点 1 总携带样本数	% of 总样本数
位点 152+ 位点 155	35	100	100
位点 33+ 位点 152	34	100	97.14
位点 33+ 位点 155	34	100	97.14
位点 152+ 位点 34	4	11.43	11.43
位点 155+ 位点 34	4	11.43	11.43
位点 10+ 位点 33	3	100	8.57
位点 10+ 位点 152	3	100	8.57
位点 10+ 位点 155	3	100	8.57
位点 33+ 位点 34	3	8.82	8.57
位点 152+ 位点 32	3	8.57	8.57
位点 152+ 位点 35	3	8.57	8.57
位点 155+ 位点 32	3	8.57	8.57
位点 155+ 位点 35	3	8.57	8.57

$\% \text{ of 位点 1 总携带样本数} = (\text{存在该组合共现的样本数} / \text{位点 1 携带样本数}) \times 100\%$

如位点 152 和位点 34 突变，共现样本数为 4，总共 35 个样本存在位点 152 突变，【% of sit152 mutation (n=35)】 $= (4/35) \times 100\% = 11.43\%$ 。原表包含更多信息，根据需要在 Excel 表中自行筛选修改。

16.3 中介效应量化

在基础数据特征分析中，H 检验和卡方检验结果显示，HBV-DNA 和乙肝五项结果在宫内传播分组间具有组间差异显著性。一定程度上，乙肝五项结果和 HBV-DNA 水平具有相关性，因此乙肝五项结果、HBV-DNA 和传播分组之间可能存在中介效应。为了证明或证否，需要对中介效应进行量化。通过假设中介效应存在，建立非线性模型，验证假设的显著性。自变量 X 除了可能直接影响因变量 Y，还可能通过 M 间接影响 Y，则称 M 为中介变量（mediation）[1]。主要的统计参数包括平均因果中介效应（ACME）、平均直接效应（ADE）、总效应（total effect）、中介比例（Prop.Mediated）。其中，ACME 表示中介变量在处理组和控制组中对结果变量的平均影响。ADE 表示处理变量对结果变量的直接影响。总效应表示直接效应和简介效应对结果变量的影响总和。中介比例表示中

介效应占总效应的比例。中介效应的检验有很多种方法。Bootstrap 检验通过重复抽样评估中介效应的稳定性和可靠性。Bootstrap 检验不依赖于正态分布假设，可获得置信区间，且适用于复杂模型。当直接效应或简介效应有一个不显著时，另外一种常见的检验方法，Sobel 法，可以检验中介效应的显著性。Sobel 检验要求数据具备正态分布，且不适用于复杂模型。相比之下，bootstrap 检验的缺点是结果更难理解。

使用 R 包 mediation 前，需要先考虑清楚研究目的和分组，并对应处理好数据表。程序成功运行依赖于正确的数据输入，而结果的正确解读依赖于符合实际意义的分组设置。在本研究中，主要考虑大三阳（HBsAg+/HBeAg+/HBcAb+，编码“1”）和小三阳（HBsAg+/HBeAb+/HBcAb+，“编码 2”）可能存在很大差异，尤其关心大三阳在影响链中起到什么作用。为了方便分析，把三分类结局改成二分类，即未发生宫内传播（NBT，编码“1”）和隐匿性宫内传播（OBT，编码“2”）。因此选择大三阳作为“处理组”，小三阳作为“控制组”。当 HBV-DNA 作为中介变量时，ACME 表示 HBV-DNA 在大三阳和小三阳中对 OBT 和 NBT 发生的平均影响。根据图 1 所示结果，在控制组和处理组中，ACME 均未达到显著性（ $P>0.05$ ），表示 HBV-DNA 作为中介变量对结果的影响不显著。在控制组和处理组中，ADE 的 P 值均为 0.050，恰好达到显著水平，表明乙肝五项对结局的直接影响是显著的。中介比例在控制组和处理组中均未达到显著性（ $P>0.05$ ），说明在这个筛选集合中，HBV-DNA 在 HBV 血清标志物和发生隐匿性宫内传播之间的中介作用并不显著，不如 HBV 血清标志物的直接影响。

代码如下：

```
library(readxl)
library(dplyr)
library(ggpubr) # 用于统计检验
library(mediation) # 用于分析中介效应
library(quantreg) # HBV_DNA 非正态分布，使用分位数回归
library(writexl)
```



```
# 读取数据
data <- read_excel(“保存地址 / 基础临床信息表 (XA).xlsx” ,2)
# 替换空格和 #N/A 为 NA
data[data == “” ] <- NA
data[data == “#N/A” ] <- NA
# 使用基础 R 包 na.omit 删除缺失值
cleaned_data <- na.omit(data[, c(“Age”, “HBV_DNA”, “Genotype”, “Subtype”,
“乙肝五项”, “传播分组” )])
# 检查缺失值
sum(is.na(cleaned_data$ 乙肝五项 ))
sum(is.na(cleaned_data$ 传播分组 ))
sum(is.na(cleaned_data$HBV_DNA))
sum(is.na(cleaned_data$Age))
# 确保因子类型
cleaned_data$ 乙肝五项 <- as.factor(cleaned_data$ 乙肝五项 )
cleaned_data$ 传播分组 <- as.factor(cleaned_data$ 传播分组 )
str(cleaned_data)
levels(cleaned_data$ 乙肝五项 )
levels(cleaned_data$ 传播分组 )
# 中介模型：HBV_DNA 作为中介变量
mediator_model_HBV_DNA <- rq(HBV_DNA ~ 乙肝五项 , data = cleaned_data,
tau = 0.25)
# 确认中介模型拟合情况
summary(mediator_model_HBV_DNA)
# 因果模型：使用传播分组作为因变量，乙肝五项、HBV_DNA 和年龄作为
自变量
outcome_model_ 传播分组 <- glm( 传播分组 ~ 乙肝五项 + HBV_DNA + Age,
family = binomial, data = cleaned_data)
# 确认因果模型拟合情况
summary(outcome_model_ 传播分组 )
```

进行中介分析

```
mediation_result <- mediate(mediator_model_HBV_DNA,
                             outcome_model_传播分组,
                             treat = “乙肝五项”,
                             mediator = “HBV_DNA”,
                             sims = 1000,
                             boot = TRUE) # 添加 boot 参数
```

```
mediation_result <- mediate(mediator_model_HBV_DNA, outcome_model_传
播分组, treat = “乙肝五项”, mediator = “HBV_DNA”, sims = 1000)
```

查看中介分析结果

```
summary(mediation_result)
```

```
Causal Mediation Analysis

Nonparametric Bootstrap Confidence Intervals with the Percentile Method
```

	Estimate	95% CI Lower	95% CI Upper	p-value
ACME (control)	0.03740557	-0.03109123	0.13365971	0.260
ACME (treated)	0.03489036	-0.03112187	0.11247410	0.260
ADE (control)	-0.15773062	-0.33047891	-0.00099593	0.050 *
ADE (treated)	-0.16024584	-0.33440127	-0.00100587	0.050 *
Total Effect	-0.12284026	-0.28466977	0.03005520	0.114
Prop. Mediated (control)	-0.30450581	-3.73059893	2.11050021	0.342
Prop. Mediated (treated)	-0.28403033	-3.57430826	2.05262889	0.342
ACME (average)	0.03614797	-0.03110655	0.12094538	0.260
ADE (average)	-0.15898823	-0.33119653	-0.00100090	0.050 *
Prop. Mediated (average)	-0.29426807	-3.60647024	2.08851296	0.342

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Sample Size Used: 190

Simulations: 1000
```

图 16.3 大三阳、小三阳和 HBV-DNA 对 OBT、NBT 的中介效应量化结果。使用 Bootstrap 检验，重复抽样 1000 次。

接上文，可视化代码如下：

```
library(ggdiagram)      # 用于中介效应路径图绘制
library(DiagrammeR)     # 用于中介效应路径图绘制
```

```
# 提取中介效应分析结果

acme_control <- mediation_result$d0    # ACME for control group
acme_treated <- mediation_result$d1    # ACME for treated group
ade_control <- mediation_result$n0     # ADE for control group
ade_treated <- mediation_result$n1     # ADE for treated group

# 提取置信区间

acme_ci_control <- mediation_result$d0.ci # ACME confidence interval for
control group
acme_ci_treated <- mediation_result$d1.ci # ACME confidence interval for
treated group
ade_ci_control <- mediation_result$n0.ci  # ADE confidence interval for control
group
ade_ci_treated <- mediation_result$n1.ci  # ADE confidence interval for treated
group


# 打印提取的结果以进行检查
print(paste( "ACME Control:" , acme_control))
print(paste( "ACME Treated:" , acme_treated))
print(paste( "ADE Control:" , ade_control))
print(paste( "ADE Treated:" , ade_treated))
print(paste( "ACME CI Control:" , toString(acme_ci_control)))
print(paste( "ACME CI Treated:" , toString(acme_ci_treated)))
print(paste( "ADE CI Control:" , toString(ade_ci_control)))
print(paste( "ADE CI Treated:" , toString(ade_ci_treated)))


# 创建数据框
effects_df <- data.frame(
  Effect = c( "ACME (Control)" , "ACME (Treated)" , "ADE (Control)" , "ADE
(Treated)" ),
  Estimate = c(acme_control, acme_treated, ade_control, ade_treated),
```

```

    CI_Lower = c(acme_ci_control[1], acme_ci_treated[1], ade_ci_control[1], ade_
ci_treated[1]),
    CI_Upper = c(acme_ci_control[2], acme_ci_treated[2], ade_ci_control[2], ade_
ci_treated[2])
  )
  # 把数据框转换为长格式
  effects_long <- effects_df %>%
    mutate(Group = ifelse(grepl( "Control" , Effect), "Control" , "Treated" ))
  %>%
    separate(Effect, into = c( "Effect_Type" , "Group" ), sep = "\\( ")

  # 绘制效果及置信区间图
  library(ggplot2)
  # 检查 effects_long 数据框
  print(unique(effects_long$Group)) # 查看 Group 列中的唯一值

  ggplot(effects_long, aes(x = Effect_Type, y = Estimate, fill = Group)) +
    geom_bar(stat = "identity" , position = position_dodge()) +
    geom_errorbar(aes(ymin = CI_Lower, ymax = CI_Upper), position = position_
dodge(0.9), width = 0.2) +
    theme_minimal() +
    labs(title = "中介效应和直接效应的估计及其置信区间" ,
      y = "效应估计" ,
      x = "效应类型" ) +
    scale_fill_manual(values = c( "Control" ) = "blue" , "Treated" ) =
"orange" )) +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))
  # 定义路径图的数据
  install.packages( "DiagrammeR" )
  library(DiagrammeR)

```

```
# 创建路径图
DiagrammeR::grViz( “
digraph flowchart {
  node [fontname = Helvetica, shape = box, style = filled, fillcolor = lightblue]
  X [label = ‘X (Serum markers)’ ]
  M [label = ‘M (HBV-DNA)’ ]
  Y [label = ‘Y (Transmition group)’ ]

  X -> M [label = ‘a (Mediating effect)’ ]
  M -> Y [label = ‘b (Direct effect)’ ]
  M -> Y [label = ‘a*b (Indirect effect)’ ]
}
”)
```

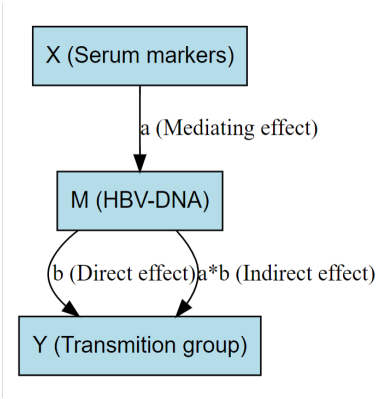


图 2 假设的 HBV 血清标志物、HBV-DNA 对传播现象的中介效应示意图。使用 R 包 DiagrammerR 绘制

16.4 关联分析的统计学方法应用拓展

基因突变与临床表型的关联分析需依托严谨的统计学方法,才能准确识别“真实关联”并排除混杂因素干扰。本节针对关联分析的核心场景（如突变与疾病发生风险、疾病进展速度的关联），详解常用统计学方法的选择、应用条件及结果

解读，确保关联结论的可靠性

16.4.1 关联性分析基本流程

16.4.1.1 二分类临床表型关联：logistic 回归分析

二项 logistic 回归分析突变与“是/否”型临床表型的关联（如突变与肝癌发生：是/否、突变与慢性化：是/否），核心目的是明确突变是否为该表型的危险因素，并用比值比（OR）量化关联强度。

应用步骤：① 确定因变量（临床表型：如肝癌=1，非肝癌=0）和自变量（突变状态：阳性=1，阴性=0）；② 单因素 logistic 回归：分析单个突变与临床表型的关联，计算未调整 OR 值及 95% 置信区间（95%CI）；③ 多因素 logistic 回归：纳入混杂因素（如年龄、性别、病程、治疗状态），计算调整 OR 值及 95%CI，排除混杂因素干扰；④ 结果解读：若调整 OR > 1 且 95%CI 不包含 1， $P < 0.05$ ，则该突变是临床表型的独立危险因素，如“多因素 logistic 回归分析显示，核心启动子双突变是肝癌发生的独立危险因素（调整 OR=3.6，95%CI: 2.5~5.2， $P < 0.001$ ）”。

16.4.1.2 生存/进展类表型关联：生存分析（Kaplan-Meier 法+Cox 回归）

适用于分析突变与临床表型进展时间的关联（如突变与肝纤维化进展时间、突变与肝癌发生时间、突变与患者生存期），核心指标是风险比（HR），量化突变对进展速度的影响。

应用步骤：① 数据准备：确定生存时间（如从确诊慢性乙肝到发生肝癌的时间）和结局事件（如肝癌发生=1，随访结束未发生=0，失访=截尾值）；② Kaplan-Meier 法：绘制不同突变状态（阳性/阴性）的生存曲线，用 log-rank 检验比较组间生存曲线差异，如“X 区截短突变阳性患者的肝癌发生生存曲线显著低于野生型患者（log-rank $\chi^2=38.5$ ， $P < 0.001$ ）”；③ Cox 比例风险回归：单因素分析单个突变对生存结局的影响，计算未调整 HR；多因素分析纳入混杂因素，计算调整 HR 及 95%CI，明确突变是否为独立预后因素，如“多因素 Cox 回归显示，X 区截短突变阳性患者的肝癌发生风险是野生型的 4.8 倍（调整 HR=4.8，95%CI: 3.2~7.2， $P < 0.001$ ）”。

16.4.1.3 多分类/连续型表型关联：有序 logistic 回归/线性回归

有序 logistic 回归适用于多分类有序临床表型（如肝纤维化分期 $F0 < F1 < F2$

< F3 < F4)，分析突变与表型等级变化的关联，如“核心启动子双突变阳性患者的肝纤维化分期显著升高（有序 logistic 回归 $OR=2.5$ ，95%CI: 1.8~3.4， $P < 0.001$ ）”；

线性回归：适用于连续型临床表型（如病毒载量、ALT 水平），分析突变对表型数值的影响，如“前 S 缺失突变阳性患者的 ALT 水平显著高于野生型（线性回归 $\beta=25.3$ ，95%CI: 12.6~38.0， $P < 0.001$ ）”。

16.4.2 统计学方法选择与应用注意事项

1) 匹配方法与研究设计：病例 - 对照研究（如肝癌病例组 vs 非肝癌对照组）常用 logistic 回归；队列研究（如随访慢性乙肝患者观察肝癌发生）常用生存分析；横断面研究仅能分析关联，无法确定因果关系。

2) 样本量估算：关联分析前需进行样本量估算，确保统计检验效能（通常 $\geq 80\%$ ）；如 logistic 回归分析单个突变与肝癌的关联，需至少纳入病例组 100 例、对照组 200 例（根据预期 $OR=3.0$ ， $\alpha=0.05$ ，效能 =80% 估算）。

3) 多重检验校正：若同时分析多个突变位点与同一临床表型的关联，需进行多重检验校正（如 Bonferroni 校正、FDR 校正），避免 I 类错误（假阳性关联），如“经 Bonferroni 校正后，仅 X 区截短突变与肝癌的关联仍有统计学意义（校正 $P < 0.001$ ）”。

4) 因果关系判断：统计学关联 $\neq$ 因果关系，需结合生物学机制（如突变的致癌通路）、时序关系（突变发生在临床表型之前）、剂量 - 反应关系（突变频率越高，临床表型风险越高）综合判断。

16.5 分子流行病学视角下的突变研究

16.5.1 分子流行病学研究框架

分子流行病学（molecular epidemiology）是一门交叉学科研究方法，以流行病学为核心，融合了实验室和生物信息学技术，用于明确疾病病因并推动干预措施的实施。分子流行病学方法在解析外源性环境（病毒）暴露与宿主因素相互作用、识别高危人群和个体方面起到愈加重要的作用。在分子流行病学视角下，基因突

变的生物标志物属性得到强化，其背后的病毒进化动力学和流行病的传播动力学相融合，从而突变个体层面的研究限制，扩展到疾病病因研究、风险评估、疾病预防等开放性范畴。具体而言，核心研究问题有突变的流行分布特征，如，特定突变在不同人群、地区、时间中的发生频率与分布规律；突变与流行病学特征的关联，如，突变与病原体谱系型别、传播链、宿主临床结局等的相关关系；突变的进化与传播规律等。其研究框架一般为，数据收集 → 分子特征解析 → 相关性分析 → 应用转化。在基础数据收集方面和一般研究相似，特别的在于分子特征数据收集和分析方法。前者在于，更加关注基因型的亚型鉴定、突变位点和功能后果注释、人群中特定突变的携带率等。分析方法特别的是，量化基因型 - 突变 - 临床结局的相关性，并将其转化为流行病学实践。

16.5.2 建立模拟数据表

```
# 加载包
library(networkD3)
library(writexl)
library(readxl)
library(dplyr)

# 生成模拟数据保存在 Excel 数据表
sankey_data <- tibble(
  source = c(
    # 第一层级：基因型 A-H → 突变状态
    "Genotype A", "Genotype B", "Genotype C", "Genotype D", "Genotype
E", "Genotype F", "Genotype G", "Genotype H",
    # 第二层级：突变状态 → 耐药状态
    "Presence of Mutations", "Presence of Mutations",
    # 第三层级：耐药状态 → 7 类核心 RAMs 突变位点
    "Resistance to NAs", "Resistance to NAs", "Resistance to
NAs", "Resistance to NAs", "Resistance to NAs", "Resistance to
NAs", "Resistance to NAs",
```


第四层级：7 类突变位点 → 5 类 NA 药物耐药

“L180M+M204V/I” ,” L180M+M204V/I” ,” L180M+M204V/I” ,” H126Y” ,” H126Y” ,” A181T/V” ,” A181T/V” ,” L269I+H126Y” ,” L269I+H126Y” ,” V173L” ,” V173L” ,” M204V/I” ,” M204V/I” ,” L180M+M204V/I+V173L” ,” L180M+M204V/I+V173L” ,” L180M+M204V/I+V173L”

),

target = c(

第一层级对应目标：突变状态

“Presence of Mutations” ,” Presence of Mutations” ,” Presence of Mutations” ,” Presence of Mutations” ,” Presence of Mutations” ,” Without Mutation” ,” Presence of Mutations” ,” Presence of Mutations” ,

第二层级对应目标：耐药状态

“Non-Resistant” ,” Resistance to NAs” ,

第三层级对应目标：7 类 RAMs 突变位点

“ L 1 8 0 M + M 2 0 4 V / I ” ,” H 1 2 6 Y ” ,” A 1 8 1 T / V ” ,” L269I+H126Y” ,” V173L” ,” M204V/I” ,” L180M+M204V/I+V173L” ,

第四层级对应目标：5 类 NA 药物耐药

“Lamivudine(LMV)” ,” Telbivudine(TBV)” ,” Entecavir(ETV)” ,” Lamivudine(LMV)” ,” Telbivudine(TBV)” ,” Adefovir(ADV)” ,” Tenofovir(TDF)” ,” Lamivudine(LMV)” ,” Entecavir(ETV)” ,” Lamivudine(LMV)” ,” Telbivudine(TBV)” ,” Lamivudine(LMV)” ,” Telbivudine(TBV)” ,” Lamivudine(LMV)” ,” Telbivudine(TBV)” ,” Entecavir(ETV)”

),

value = c(

第一层级 value：匹配文献总样本 n=8147

4216, 823, 357, 269, 184, 426, 217, 115,

第二层级 value：无突变 =2693 有突变 =5454 → 耐药突变 =863 非耐药 =4591

4591, 863,

第三层级 value：7 类突变位点的样本量 (设置 L180M+M204V/I=467 占

比 54.1% 核心突变)

```

    467, 280, 69, 47, 32, 29, 39,
    # 第四层级 value: 药物耐药样本量
    840, 816, 217, 816, 798, 29, 8, 201, 67, 198, 187, 276, 264, 189, 176, 63
  )
)
# 导出模拟数据
write_xlsx(sankey_data, "保存地址 / 耐药突变桑基图_数据模板.xlsx")

```

16.5.3 使用桑基图可视化多变量流量关联

桑基图 (Sankey diagram) 属于流程图的一种, 用于可视化节点状态的组别流向。在桑基图中, 流程分支的宽度与相应的数据流量大小成正比, 在揭示关键节点及其连接、数据趋势和交互路径等方面具有明显优势。桑基图的维度 (dimension) 代表流程中各个阶段或类别; 度量 (measure) 是计量指标, 代表流经节点的流量大小。

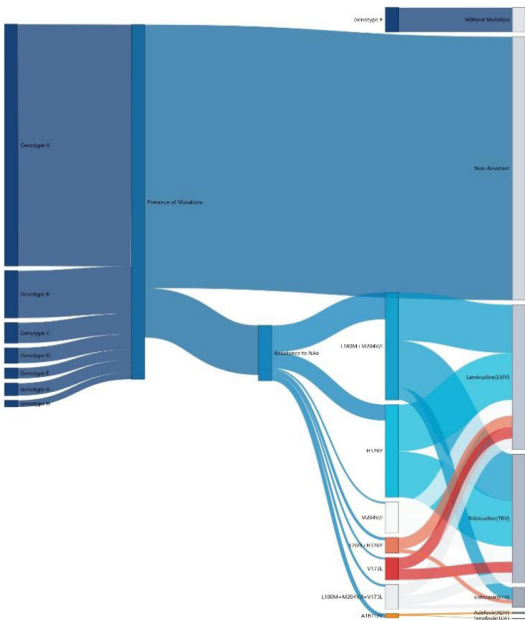
代码如下:

```

library(networkD3)
library(writexl)
library(readxl)
library(dplyr)
sankey_df <- read_xlsx("保存地址 / 耐药突变桑基图_数据模板.xlsx")
# 格式转换
nodes <- data.frame(name = unique(c(as.character(sankey_df$source),
as.character(sankey_df$target))))
sankey_df$source_id <- match(sankey_df$source, nodes$name) - 1
sankey_df$target_id <- match(sankey_df$target, nodes$name) - 1
# 风格设置
RAMs_sankey_full <- sankeyNetwork(
  Links      = sankey_df,

```

```
Nodes      = nodes,
Source      = "source_id" ,
Target      = "target_id" ,
Value       = "value" ,
NodeID      = "name" ,
fontSize    = 11,      # 字体大小
nodeWidth   = 28,      # 节点宽度
nodePadding = 10,      # 节点间距
colourScale = JS( 'd3.scaleOrdinal()
                    .range([ "#002F6C" ," #005B96" ," #0077B6" ," #0096C7" ,"
                    #00B4D8" ," #F77F00" ," #E76F51" ," #D62828" ,
                    "#F8F9FA" ," #E9ECEF" ," #DEE2E6" ," #CED4DA" ,
                    "#B9C2D0" ," #A9B2C0" ," #99A2B0" ," #8992A0" ,
                    "#798290"," #697280"," #596270"," #495260"," #394250" ])' ),
LinkGroup   = "source" , # 流线颜色和源节点一致
NodeGroup   = "name" ,
iterations  = 200,      # 迭代次数
margin      = list(top=20, right=20, bottom=20, left=20) # 边距适配, 防止文字
截断
)
RAMs_sankey_full
# 保存图像
library(htmlwidgets)
saveWidget(RAMs_sankey_full, " 保 存 地 址 /RAMs 桑 基 图 .html" ,
selfcontained = TRUE)
```



本章小结

描述性统计：连续变量用均值 ± 标准差、中位数 + 四分位数；分类变量用频数 + 百分比表示。组间比较：连续变量用 t 检验、Mann-Whitney U 检验；分类变量用卡方检验；小样本用 Fisher 精确检验。相关性分析：用 Kendall tau 系数、Pearson 系数分析基因突变与临床特征的关联。显著性阈值：双侧检验， $P \leq 0.05$ 为差异有统计学意义。本章只涉及核心分析部分。实际上，作为一项研究，需要进行更多的实验，对结果进行多维交叉印证，确保结果可靠。

第七部分 基因变异分析的前沿方法

第 17 章 补充论证实验

乙肝病毒（HBV）的高度变异性、准种复杂性及宿主互作多样性，使其基因分析成为解析感染机制、指导精准诊疗的核心环节。传统基因分析技术受限于读长、灵敏度及通量等瓶颈，难以全面捕捉 HBV 基因组的动态变化与异质性特征。近年来，三代测序、单细胞测序、人工智能及宏基因组学等前沿技术的突破，为 HBV 基因分析带来了革命性变革——从全基因组精准解析到单细胞水平异质性捕捉，从自动化突变识别到微生态互作网络构建，全方位拓展了 HBV 研究的深度与广度。本章聚焦上述前沿技术，系统阐述其技术原理、在 HBV 基因分析中的核心应用场景与独特优势，展望未来技术融合发展趋势，为进阶研究提供技术选型与方向指引。

17.1 基于反向遗传学的基因突变功能验证

反向遗传学通过对病毒基因组进行定向修饰，结合感染性克隆技术拯救重组病毒，是解析基因变异生物学功能的核心手段。该方法突破了传统“从表型到基因”的研究模式，可直接验证特定变异对病毒复制、致病性、免疫逃逸等表型的影响，适用于各类 DNA 病毒与 RNA 病毒的功能研究。

17.1.1 感染性克隆构建技术

感染性克隆是反向遗传学研究的基础，指包含完整病毒基因组 cDNA 的克隆载体，导入宿主细胞后可转录翻译产生有感染性的病毒粒子。目前主流构建平台包括酵母人工染色体（YAC）、细菌人工染色体（BAC）及 YAC-BAC 穿梭系统，各有适配场景与技术特点。

对比项	YAC 系统	BAC 系统	YAC-BAC 穿梭系统
核心技术基础	酵母高效同源重组机制	大肠杆菌 F 因子复制调控	双宿主（酵母 + 大肠杆菌）复制元件，结合同源重组与便捷扩增
外源片段容纳能力	200-500kb，最高可达 1Mb	150-350kb	兼容大型基因组（适配 200kb 以上片段）
限制位点依赖	无，无需依赖病毒 DNA 限制性酶切位点	需在病毒非必需区域整合载体元件，依赖合适位点	无，依托同源重组组装，不受酶切位点限制
关键技术步骤	1. RT-PCR 扩增 RNA 病毒全基因组 cDNA 并分段；2. 与线性化 TAR 载体共转化酵母；3. 同源重组组装全基因组；4. 测序验证后提取质粒	1. 同源重组将载体元件整合至病毒非必需区；2. 大肠杆菌中克隆筛选；3. 直接转染真核细胞拯救病毒	1. 病毒基因组拆分为多个重叠片段；2. 与穿梭载体连接后导入酵母组装；3. 转入大肠杆菌扩增；4. 测序验证后用于病毒拯救
启动子适配	依赖体外转录启动子（T7/SP6）	可引入 CMV 立即早期启动子	兼容 CMV、T7 等多种启动子，适配不同病毒类型
核心优势	1. 容纳大片段基因组，完整保留调控序列；2. 适配复杂基因组病毒；3. 同源重组效率高，组装精准	1. 低拷贝特性，减少基因组重排与嵌合；2. 无需体外转录步骤，转染效率高；3. 大肠杆菌扩增便捷，质粒产量高	1. 结合酵母高效组装与大肠杆菌便捷扩增优势；2. 灵活适配复杂病毒基因组工程；3. 专用载体（如 pCC1BAC-His3）成熟
主要劣势	1. 酵母质粒提取繁琐，产量低；2. 体外转录步骤中 T7/SP6 聚合酶错误率较高；3. 实验周期较长	1. 寻找病毒非必需区限制性酶切位点较困难；2. RNA 病毒 cDNA 可能对大肠杆菌产生毒性；3. 大片段组装效率低于酵母系统	1. 载体构建流程较单一系统复杂；2. 双宿主切换需优化实验条件；3. 专用载体成本较高
适用病毒类型	大型复杂基因组病毒、需完整保留调控序列的 DNA/RNA 病毒	疱疹病毒、冠状病毒、中等大小基因组病毒	大型复杂病毒、需精准组装与高效扩增的病毒（如烈性病毒、基因组片段化病毒）
典型应用案例	大型 DNA 病毒（如痘病毒）、长链 RNA 病毒全基因组克隆	人巨细胞病毒、SARS-CoV-2、乙型肝炎病毒	复杂疱疹病毒、多片段基因组病毒的功能解析

17.1.2 定点突变与病毒重组

Xia 等人开展的一项寨卡病毒突变功能研究，使用了 PCR 介导的定点突变法。他们首先选择前流行株和当前流行株，线性化后体外转录可生成有感染性的病毒 RNA。核心是对 NS1 基因第 188 位变异位点，设计含目标突变的特异性引物（突

变位点位于引物中间区域，两侧各保留 15-20bp 与模板同源的序列）。突变位点在扩增过程中被引入。拼接后的突变型基因组片段，通过限制性内切酶酶切位点插入到原始 cDNA 克隆载体中，将重组载体转化感受态大肠杆菌，筛选阳性克隆后进行双向 Sanger 测序。

17.1.3 变异效应的多维验证

体外细胞水平验证有多种方法，以下部分举例：

(1) 病毒复制能力检测：将野生型与突变型重组病毒以相同 MOI（感染复数）感染宿主细胞，在不同时间点（0h、24h、48h、72h）收集细胞与上清液。采用 qPCR 定量检测病毒核酸拷贝数，Western blot 或 ELISA 检测病毒蛋白表达水平，对比二者在复制动力学上的差异。可结合报告基因系统，如构建含荧光素酶基因的重组病毒，通过检测荧光素酶活性间接反映病毒复制效率。

(2) 宿主免疫通路影响分析：检测变异病毒对宿主天然免疫通路的调控作用，核心聚焦 RIG-I/MAVS/TBK1/IRF3 等关键通路。感染后通过 Western blot 检测 TBK1、IRF3 等蛋白的磷酸化水平，判断通路激活状态；采用 IFN- β -luc 报告基因 assay，将 IFN- β 启动子驱动的荧光素酶质粒与内参质粒共转染细胞，感染病毒后检测荧光素酶活性，量化病毒对 IFN- β 表达的影响。通过免疫荧光染色观察 IRF3 核转位情况，进一步验证通路调控机制。

(3) 蛋白相互作用验证：采用免疫共沉淀（Co-IP）技术验证病毒变异蛋白与宿主蛋白的结合能力。将编码病毒变异蛋白的质粒与候选宿主蛋白质质粒共转染细胞，裂解后用特异性抗体捕获病毒蛋白，通过 Western blot 检测复合物中的宿主蛋白。实验需设置阴性对照（空载体）与阳性对照（已知相互作用的蛋白对），IP 抗体与 WB 抗体需选用不同种属来源，避免交叉反应导致的假阳性。

(4) 体外动物模型验证，将野生型与突变型病毒通过腹腔注射、静脉注射或滴鼻等方式感染动物，定期采集血液、组织样本。检测血清中病毒载量、细胞因子（IFN- β 、IL-6、TNF- α 等）水平，通过组织病理学分析观察靶器官损伤程度。采用免疫组化技术检测组织中病毒抗原分布与免疫细胞浸润情况。

17.2 病毒变异与进化的驱动因素

Symons 等人的研究选取关键 SARS-CoV-2 变异株，使用该种病毒适用的非洲绿猴肾细胞系（Vero-E6）为主要培养体系，同时将 Delta 变异株在 Calu-3 细胞（人肺腺癌细胞）和原代人鼻上皮细胞（HNEC，气液界面培养，模拟人体感染环境）中培养，排除单一细胞环境的偏差。原始株和部分变异株 7 代连续传代，Beta、Gamma、Omicron 株仅 1 代传代（1 个重复）；传代时控制感染复数（ $MOI=0.1$ ），减少共感染导致的缺陷基因组被完整功能基因补足，保证突变检测的准确性。采用滚环 RNA 共识测序（CirSeq）技术，该技术具有超高灵敏度和准确性，可有效排除测序误差和逆转录误差。其方法框架可以概括为：连续传代→追踪突变频率→计算适应性值→验证可靠性。

第 18 章 基因变异研究的前沿方法

18.1 宏基因组学分析技术

HBV 的宿主可能存在不止一种病原体, 如 HCV、HDV、HIV、流感病毒等, 形成复数种病毒互作关系。宏基因组学(Metagenomics)的技术基点是高通量测序, 通过对病原体的广泛测序, 打破了传统单一病原体研究的局限, 实现了 HBV 与微生态、共感染病原体的协同分析, 为解析 HBV 感染的微生态调控机制、共感染的变异协同效应提供了全新视角。此外, 这项技术有助于识别潜在的新出现和再出现的病原体, 并产生有价值的基因组数据, 增进对疾病生物学、宿主 - 病原体相互作用和流行病学的理解。

宏基因组学分析是一种高技术密度的策略概念, 基于宏基因组序列数据, 通过多种数据处理和分析工具、方法实现具体研究目的。Moreno 等人^[2]在一项急性发热性疾病和严重神经系统疾病的病因学调查(etiological investigations)中, 使用了基于纳米孔技术的宏基因组分析。这类疾病由于非特异性症状和广泛的相关感染, 使传统诊断方法力有不逮。Moreno 等人采用 SMART-9N 协议提取 RNA, 在获得测序结果后, 使用过由 Chan-Zuckerberg IDseq 平台 [13] 提供的开源宏基因组云管道进行分析。该分析包括一系列步骤: 使用 Fastp 软件 [14] 进行初步质量评估, 然后使用 minimap2[15] 去除人类宿主序列。随后, 使用 Flye 软件重新组装剩余的 reads, 特别是使用为宏基因组分析量身定制的 metaFlye 参数 [16]。然后使用 DIAMOND 软件 [17] 对得到的 contigs 和 unaligned reads 进行分类, 并与 NCBI Nucleotide 数据库(NT)交叉引用。使用 Aliview[21] 进行人工培养, 以消除生物伪影。系统发育树使用 IQ-TREE2 进行最大似然(ML)估计, 采用核苷酸替代的一般时间可逆(GTR)模型, 支持 1000 次 bootstrap 重复, 命令为 iqtree -s archive. fasta -m GTR fr5 -bb 1000。

宏基因组学可揭示微生物群落结构和多样性, 但无法反映基因表达水平。Xu 等人^[3]的研究在宏基因组学分析的基础上, 融合宏蛋白质组学以体现微生物

物群的功能特征，从而挖掘鉴别乙肝病毒相关肝病的生物标志物。在获得宏基因组和宏蛋白质组数据后，采用 SOAPnuke、Bowtie2 等工具进行数据清洗与过滤；通过 MEGAHIT、MetaGeneMark 等进行基因预测与注释；利用 Kraken2、Diamond 等进行微生物分类学分析；运用无监督聚类（PAM 算法）进行肠型分类；采用随机森林（RF）和支持向量机（SVM）构建集成学习模型，结合留一交叉验证（LOOCV）和 10 折交叉验证进行特征选择与模型验证，通过 ROC 曲线（AUC 值）评估鉴别效果。

18.2 人工智能在突变分析和疫情预测中的作用

18.2.1 病毒变异与进化预测的智能建模

基于进化论视角的深度学习框架为乙肝病毒（HBV）变异研究带来突破性进展，解决了传统工具跨物种适配性差的核心难题。这类进化驱动的预测模型（如 E2VD）通过定制化蛋白质语言模型，精准捕获病毒进化的两大本质特征——少数位点突变与稀有有益突变，实现了跨病毒类型的通用预测。其核心创新在于通过微弱突变放大技术，敏锐感知 HBV 基因组中关键位点突变引发的分子互作网络变化，同时借助多任务焦点损失函数，有效破解有益突变数据稀缺导致的正负样本不平衡问题，将 HBV 潜在高风险突变的预测精度提升至新高度。

定制化预训练策略是该类模型的核心优势，以精简的模型参数量即可实现超越传统大模型的预测效能，能够精准区分 HBV 不同突变类型的功能效应，包括耐药突变、复制调控突变等关键变异的驱动机制解析。该技术可灵活应用于 HBV 宏观进化轨迹重现与微观突变效应预测，既能揭示不同基因型 HBV 的流行扩散规律，也能通过虚拟深度突变扫描，提前预警潜在高致病性变异株，为疫苗与抗病毒药物的主动更新提供核心数据支撑。

18.2.2 多组学整合与病毒微环境解析

单细胞转录组与空间转录组技术的融合应用，推动 HBV 相关肝病研究进入时空多维度解析时代。通过整合 GEO 等公共数据库的单细胞测序数据与空间转录组数据，可在单细胞分辨率下构建 HBV 相关肝细胞癌（HCC）的全景细胞图

谱，精准识别肿瘤细胞、肝星状细胞（HSC）、免疫细胞等 15 种以上细胞亚型的分布特征与功能状态。空间转录组技术进一步实现了细胞亚型的原位定位，明确 HSC 与 HBV 感染肿瘤细胞的紧密邻域关系，为解析病毒感染微环境中的细胞互作机制提供了直接证据。

多组学整合分析技术能够系统揭示 HBV 感染驱动的分子通路异常，发现 HBV 与其他病因相关 HCC 共享上皮 - 间质转化、TGF- β 信号通路等核心促癌通路，为跨病因肝病治疗提供理论依据。结合 InferCNV 分析、细胞轨迹追踪与信号互作网络构建，可清晰描绘 HBV 感染从慢性肝炎到肝癌的细胞分化轨迹，识别组蛋白乳酸化等新型表观遗传修饰通路在病毒致癌过程中的关键作用，同时挖掘 AKR1B10 等兼具预后评估与治疗靶点价值的核心标志物。

免疫微环境的单细胞解析技术为 HBV 感染的免疫调控研究提供了新视角，通过深度剖析外周血与肝脏组织中 CD8⁺ 耗竭性 T 细胞、调节性 CD4⁺T 细胞等免疫细胞的转录组特征，可明确不同感染阶段的免疫功能状态差异，为免疫治疗策略的优化提供精准靶点。

18.2.3 蛋白质结构与功能的精准预测

基于蛋白质语言模型的结构预测技术，解决了 HBV 关键蛋白结构解析的传统难题，尤其在病毒抗原与抗体相互作用研究中展现出独特优势。针对 HBV 表面抗原、聚合酶等关键蛋白的结构预测，该类技术（如 H3-OPT）通过模板模块与深度学习模块的整合，显著提升了功能关键区域的结构预测精度，其预测的蛋白质表面氨基酸分布更接近实验实测结果，为抗原 - 抗体结合机制分析提供可靠的结构基础。

分子动力学模拟与结构预测技术的结合，能够精准评估 HBV 突变对蛋白质结合亲和力的影响，突破传统工具仅依赖序列特征的局限，直接从三维结构层面解析突变导致的功能改变，包括耐药突变对药物结合口袋的影响、免疫逃逸突变对抗原表位的修饰等关键问题。该技术已成功应用于抗体药物设计，通过预测 HBV 特异性抗体的 CDR-H3 环结构，优化抗原结合效率，为 HBV 中和抗体的研发提供高效工具。

18.2.4 技术转化与应用前景

前沿人工智能与多组学技术的融合，正在重塑 HBV 研究的技术范式与转化路径。在临床诊断领域，整合突变预测与多组学标志物的智能诊断模型，可实现 HBV 感染预后风险的精准分层与耐药风险预警，为个体化治疗方案制定提供决策支持。在药物研发方面，病毒进化预测模型与蛋白质结构预测技术的联合应用，能够加速抗病毒药物靶点筛选与药物分子设计，提升药物对变异株的覆盖度与疗效持久性。

疫苗开发领域，基于病毒进化轨迹预测的反向疫苗设计策略，可提前预判 HBV 流行株的变异趋势，设计覆盖多种变异型的广谱疫苗，显著提升疫苗的长期保护效果。在基础研究层面，时空多组学与人工智能技术的结合，将进一步揭示 HBV 感染与宿主互作的分子机制，为攻克慢性 HBV 感染这一全球公共卫生难题提供全新的研究思路与技术支撑。

本章小结

本章系统阐述了三代测序、单细胞测序、人工智能及宏基因组学等前沿技术在 HBV 基因分析中的应用进展。三代测序以超长读长与高灵敏度优势，成为 HBV 全长准种与重组变异解析的核心技术；单细胞测序突破群体平均化局限，实现了 HBV 整合与异质性的单细胞水平精准解析；人工智能提升了突变分析的自动化与精准化程度，为临床结局预测提供了高效工具；宏基因组学揭示了 HBV 与微生态、共感染病原体的协同关系，拓展了研究视角。未来，随着多技术融合、AI 深度应用、微量检测突破及临床转化推进，HBV 基因分析技术将实现从基础研究到临床应用的全面跨越，为 HBV 感染的精准防控与根治提供强大技术支撑。进阶研究可聚焦技术融合创新、临床转化瓶颈突破及复杂机制解析，推动 HBV 研究迈向新高度。

参考文献

第 1 章

- [1] WHO. Global hepatitis report 2024: action for access in low- and middle-income countries. <https://www.who.int/publications/i/item/9789240091672>
- [2] Li D, Hamadani Y, Tu T. Hepatitis B Viral Protein HBx: Roles in Viral Replication and Hepatocarcinogenesis[J]. *Viruses*. 2024; 16(9):1361. <https://doi.org/10.3390/v16091361>
- [3] Chiaho Shih, Shu-Fan Chou, Ching-Chun Yang, Jyun-Yuan Huang, Gansukh Chojilsuren, Ren-Shiang Jhou. Control and Eradication Strategies of Hepatitis B Virus. *Trends in Microbiology*. June 07, 2016.
- [4] Hui Z, Yu W, Fuzhen W, et al. New progress in HBV control and the cascade of health care for people living with HBV in China: evidence from the fourth national serological survey, 2020[J]. *The Lancet Regional Health - Western Pacific*, 2024, 51.
- [5] Deng Y, Meng T, You H, Jia J, Wang Y. Epidemiology, Achievements, and Challenges in the Elimination of Hepatitis B in China. *J Clin Transl Hepatol*. Published online: May 21, 2025. doi: 10.14218/JCTH.2025.00039.
- [6] Xu Y, Liu H, Wang Y, et al. The next step in controlling HBV in China[J]. *BMJ*, 2013, 347; f4503. DOI: 10.1136/bmj.f4503.
- [7] Yeo YH, Nguyen MH. Review article: current gaps and opportunities in HBV prevention, testing and linkage to care in the United States: a call for action[J]. *Aliment Pharmacol Ther*, 2021, 53(1):63-78. DOI: 10.1111/apt.16125
- [8] World Health Organization. Hepatitis B vaccines: WHO position paper, July 2017-Recommendations[J]. *Vaccine*, 2019, 37(2): 223-225. DOI: 10.1016/j.vaccine.2017.07.046.
- [9] Wei KP, Zhu FC, Liu JX, et al. The efficacy of two different dosages of hepatitis B immunoglobulin combined with hepatitis B vaccine in preventing mother-

- to-child transmission of hepatitis B virus: A prospective cohort study[J]. *Vaccine*, 2018, 36(2): 256-263. DOI: 10.1016/j.vaccine.2017.11.037.
- [10] 丁洋, 尤红, 任红, 等. 感染乙型肝炎病毒的育龄女性临床管理共识[J]. *中华肝脏病杂志*, 2018, 26(3): 204-208. DOI: 10.3760/cma.j.issn.1007-3418.2018.03.009.
- [11] XiaoY, SunK, DuanZ, et al. Quasispecies characteristic in “a” determinant region is a potential predictor for the risk of immunoprophylaxis failure of mother-to-child-transmission of sub-genotype C2 hepatitis B virus: A prospective nested case-control study[J]. *Gut*, 2020, 69(5): 933-941. DOI: 10.1136/gut-jnl-2019-318278
- [12] LiAY, LiuZ, SongY, et al. Reduction of the occurrence of occult HBV infection in infants by increasing the dose of hepatitis B vaccine: A large prospective cohort study[J]. *Emerg Microbes Infect*, 2020, 9(1): 1881-1891. DOI: 10.3390/v14091887.
- [13] 张爱民. 中国HBV基因型分布特点和临床意义及慢加急性肝衰竭患者基因突变的研究[D]. 解放军总医院军医进修学院. 2011. <http://www.cnki.net>.
- [14] hen, J., Li, L., Yin, Q. and Shen, T. (2023) A Review of Epidemiology and Clinical Relevance of Hepatitis B Virus Genotypes and Subgenotypes. *Clinics and Research in Hepatology and Gastroenterology*, 47, Article 102180. <https://doi.org/10.1016/j.clinre.2023.102180>
- [15] Liu, Z., Zhang, Y., Xu, M., Li, X. and Zhang, Z. (2021) Distribution of Hepatitis B Virus Genotypes and Subgenotypes. *Medicine*, 100, e27941. <https://doi.org/10.1097/md.00000000000027941>
- [16] 郭晓琪, 张爽, 郑徽, 等. 中国15个民族HBV基因型及基因亚型分布特征分析[J]. *中华流行病学杂志*, 2023, 44(5): 759-764. DOI: 10.3760/cma.j.cn112338-20221130-01021.
- [17] Salas-Benito J.S., De Nova-Ocampo M. Viral Interference and Persistence in Mosquito-Borne Flaviviruses. *J. Immunol. Res.* 2015;2015:873404. doi: 10.1155/2015/873404. [DOI] [PMC free article] [PubMed] [Google Scholar]
- [18] 冀柯宇, 徐琳莉, 袁筱婕. 乙型病毒肝炎分子研究进展[J]. *临床医学进展*, 2025,

- 15(7): 1150-1158. DOI: 10.12677/acm.2025.1572105
- [19] Natalia M. Araujo, Hepatitis B virus intergenotypic recombinants worldwide: An overview, *Infection, Genetics and Evolution*, Volume 36, 2015, Pages 500-510, ISSN 1567-1348, <https://doi.org/10.1016/j.meegid.2015.08.024>. (<https://www.sciencedirect.com/science/article/pii/S1567134815003470>)
- [20] 潘登,周彬,杨洁,侯金林.中国人群乙型肝炎病毒基因型间重组的鉴定[J]. 南方医科大学学报. 2014, 34(10): 1436-1442. doi 10.3969/j.issn.1673-4254.2014.10.08
- [21] Yang J, Xing K, Deng RQ, et al. Identification of hepatitis B virus putative intergenotype recombinants by using fragment typing [J] . *J Gen Virol*, 2006, 87(8): 2203-15.
- [22] Hideaki K., Etsuro O., Fuminaka S., et al., Frequent coinfection with hepatitis B virus strains of distinct genotypes detected by hybridization with type-specific probes immobilized on a solid-phase support, *Journal of Virological Methods*, Volume 110, Issue 1, 2003, Pages 29-35, ISSN 0166-0934, [https://doi.org/10.1016/S0166-0934\(03\)00095-8](https://doi.org/10.1016/S0166-0934(03)00095-8). (<https://www.sciencedirect.com/science/article/pii/S0166093403000958>)
- [23] Adolf K. Maas, Considerations and Conditions Involving Protozoal Inhabitation of the Reptilian Gastrointestinal Tract, *Veterinary Clinics of North America: Exotic Animal Practice*, Volume 17, Issue 2, 2014, Pages 263-297, ISSN 1094-9194, ISBN 9780323297271, <https://doi.org/10.1016/j.cvex.2014.01.008>. (<https://www.sciencedirect.com/science/article/pii/S1094919414000097>)
- [24] Mavilia MG, Wu GY. HBV-HCV Coinfection: Viral Interactions, Management, and Viral Reactivation. *J Clin Transl Hepatol*. 2018 Sep 28;6(3):296-305. doi: 10.14218/JCTH.2018.00016. Epub 2018 Jul 6. PMID: 30271742; PMCID: PMC6160312.
- [25] C. Hannoun, H. Norder, M. Lindh An aberrant genotype revealed in recombinant hepatitis B virus strains from Vietnam *J. Gen. Virol.*, 81 (2000), pp. 2267-2272
- [26] Baig S, Abidi SH, Azam Z, Majid S, Khan S, Khanani MR, Ali S. Evolution of HBV S-gene in the backdrop of HDV co-infection. *J Med Virol*. 2018

- Aug;90(8):1328-1336. doi: 10.1002/jmv.25200. Epub 2018 May 1. PMID: 29663447.
- [27] Magnius, L.; Mason, W.S.; Taylor, J.; Kann, M.; Glebe, D.; Dény, P.; Sureau, C.; Norder, H.; ICTV Report Consortium[J]. ICTV Virus Taxonomy Profile: Hepadnaviridae. *J. Gen. Virol.* 2020, 101, 571–572. [Google Scholar] [CrossRef] [PubMed]
- [28] 王明凤, 曹佳莉, 袁权, 等. 乙型肝炎病毒体外感染和复制的细胞模型[J]. 微生物学报, 2019, 59(12): 2263-2275
- [29] Lee H. W.; Lee J. S.; Ahn S. H. Hepatitis B virus cure: Targets and future therapies. *Int. J. Mol. Sci.* 2021, 22, 213. 10.3390/ijms22010213. [DOI] [PMC free article] [PubMed]
- [30] Zhang T, Yang J, Gao H, Wu Y, Zhao X, Zhao H, Xie X, Yang L, Li Y, Wu Q. Progress of Infection and Replication Systems of Hepatitis B Virus. *ACS Pharmacol Transl Sci.* 2024 Jun 4;7(6):1711-1721. doi: 10.1021/acsptsci.4c00147. PMID: 38898948; PMCID: PMC11184603.
- [31] 中华医学会肝病学会, 中华医学会感染病学分会. 慢性乙型肝炎防治指南(2022年版) [J]. 中华临床感染病杂志, 2022, 15(6) : 401-427. DOI: 10.3760/cma.j.issn.1674-2397.2022.06.001.
- [32] 刘嘉城,吴超,李婕.不确定期慢性乙型肝炎病毒感染者的诊疗新进展.中华肝脏病杂志,2024,32(4):370-374.DOI: 10.3760/cma.j.cn501113-20231129-00252. <https://rs.yiigle.com/CN115399202004/1501778.htm>
- [33] Pollicino T, Cacciola I, Saffioti F, Raimondo G. Hepatitis B virus PreS/S gene variants: pathobiology and clinical implications. *J Hepatol.* 2014 Aug;61(2):408-17. doi: 10.1016/j.jhep.2014.04.041. Epub 2014 May 5. PMID: 24801416.
- [34] Ali M.J., Shah P.A., Rehman K.U., Kaur S., Holzmayer V., Cloherty G.A., Kuhns M.C., Lau D.T.Y. Immune-Escape Mutations Are Prevalent among Patients with a Coexistence of HBsAg and Anti-HBs in a Tertiary Liver Center in the United States. *Viruses.* 2024;16:713. doi: 10.3390/v16050713.
- [35] Seeger C, Mason WS. Hepatitis B virus biology. *Microbiol Mol Biol Rev* 2000;64:51–68.
- [36] Nassal M. Hepatitis B viruses: reverse transcription a different way. *Virus Res*

- 2008;134:235–249.
- [37] Lang T, Lo C, Skinner N, Locarnini S, Visvanathan K, and Mansell A (2011). The hepatitis B e antigen (HBeAg) targets and suppresses activation of the toll-like receptor signaling pathway. *J Hepatol* 55, 762–769.
- [38] W. Carman, H. Thomas, E. Domingo. Viral genetic variation: hepatitis B virus as a clinical example. *Lancet*, 341 (1993), pp. 349-353
- [39] Eren E, Watts NR, Dearborn AD, Palmer IW, Kaufman JD, Steven AC, Wingfield PT. Structures of Hepatitis B Virus Core- and e-Antigen Immune Complexes Suggest Multi-point Inhibition. *Structure*. 2018 Oct 2;26(10):1314-1326.e4. doi: 10.1016/j.str.2018.06.012. Epub 2018 Aug 9. PMID: 30100358; PMCID: PMC6170708.
- [40] Yu Y, Wan P, Cao Y, Zhang W, Chen J, Tan L, Wang Y, Sun Z, Zhang Q, Wan Y, et al. (2017). Hepatitis B Virus e Antigen Activates the Suppressor of Cytokine Signaling 2 to Repress Interferon Action. *Sci Rep* 7, 1729.
- [41] Pley C, Lourenço J, McNaughton AL, Matthews PC. Spacer Domain in Hepatitis B Virus Polymerase: Plugging a Hole or Performing a Role? *J Virol*. 2022 May 11;96(9):e0005122. doi: 10.1128/jvi.00051-22. Epub 2022 Apr 12. PMID: 35412348; PMCID: PMC9093120.
- [42] Soussan, P.; Tuveri, R.; Nalpas, B.; Garreau, F.; Zavala, F.; Masson, A.; Pol, S.; Brechot, C.; Kremsdorf, D. The expression of hepatitis B spliced protein (HBSP) encoded by a spliced hepatitis B virus RNA is associated with viral replication and liver fibrosis. *J. Hepatol.* 2003, 38, 343–348.
- [43] SongY, LuY, LiY, et al. HBx 128-133 deletion affecting HBV mother-to-child transmission weakens HBV replication via reducing HBx level and CP/ENII transcriptional activity[J]. *Viruses*, 2022, 14(9): 1887. DOI: 10.3390/V14091887.
- [44] Harrison TJ. Hepatitis B virus: molecular virology and common mutants. *Semin Liver Dis*, 2006,26(2):87-96.
- [45] 张爱民.中国HBV基因型分布特点和临床意义及慢加急性肝衰竭患者基因突变的研究[D]. 解放军总医院军医进修学院.2011. <http://www.cnki.net>.
- [46] Bartholomeusz A, Locarnini S. Hepatitis B virus mutants and fulminant hepatitis B:

fitness plus phenotype. *Hepatology*, 2001,34(2):432-5.

第 2 章

- [1] Loeffelholz M J, Hodinka R L, Young S A, et al. *Clinical Virology Manual*[M]. 1st ed. 2016:15 April. Print ISBN:9781555819149; Online ISBN:9781683670698; DOI:10.1128/9781555819156..
- [2] Sewell DL. Laboratory-associated infections and biosafety. *Clin Microbiol Rev*. 1995 Jul;8(3):389-405. doi: 10.1128/CMR.8.3.389. PMID: 7553572; PMCID: PMC174631.

第 3 章

- [1] 张磊.生物安全医学防护实训指导.厦门:厦门大学出版社,2024. ISBN: 978-7-5615-9361-5.
- [2] 张军,张磊.生物安全.厦门:厦门大学出版社,2024. ISBN: 978-7-5615-9335-6

第 5 章

- [1] Lumley SF, Kent C, Jennings D, et. al. Whole genome sequencing of hepatitis B virus using tiled amplicon (HEPTILE) and probe based enrichment on Illumina and Nanopore platforms. *Sci Rep*. 2025 Feb 17;15(1):5795. doi: 10.1038/s41598-025-87721-1. PMID: 39962085; PMCID: PMC11832747.
- [2] Krishna S, Jayaram A, Shetty U, Varamballi P, Mukhopadhyay C, Jagadesh A. Detection of H275Y oseltamivir resistance gene mutation among Influenza A(H1N1) pdm09 patients by allelic discrimination real - time RT - PCR. *JMed Virol*. 2024;95:e28764. doi:10.1002/jmv.2876

第 6 章

- [1] Bogaert, A.; Fernandez, E.; Gevaert, K. N-Terminal Proteoforms in Human Disease. *Trends Biochem. Sci*. 2020, 45, 308–320.
- [2] Kozak, M. Regulation of translation via mRNA structure in prokaryotes and eukaryotes. *Genes* 2005, 361, 13–37

第 9 章

- [1] Onuora, A. (2020). Review of software tools for genetic mutation analysis. *International Journal of Engineering Applied Sciences and Technology (IJEAST)*. <https://doi.org/10.1038/NRG2146>

- [2] Markus Hollander, Mohamed Hamed, Volkhard Helms, Kerstin Neininger, MutaNET: a tool for automated analysis of genomic mutations in gene regulatory networks, *Bioinformatics*, Volume 34, Issue 5, March 2018, Pages 864–866, <https://doi.org/10.1093/bioinformatics/btx687>
- [3] Brennan, J., How Can a Mutation in DNA Affect Protein Synthesis? 2018, <https://sciencing.com/can-mutation-dnaaffect-protein-synthesis-2028.html>
- [4] Yongwook Choi, Agnes P. Chan, PROVEAN web server: a tool to predict the functional effect of amino acid substitutions and indels, *Bioinformatics*, Volume 31, Issue 16, August 2015, Pages 2745–2747, <https://doi.org/10.1093/bioinformatics/btv195>
- [5] Capriotti E, Fariselli P and Casadio R (2005). I-Mutant2.0: predicting stability changes upon mutation from the protein sequence or structure. *Nucl. Acids Res.* 33: W306-W310. <http://gpcr.biocomp.unibo.it/cgi/predictors/I-Mutant2.0/I-Mutant2.0.cgi>.
- [6] Topal MD, Fresco JR (1976) Complementary base pairing and the origin of substitution mutations. *Nature* 263:285-289
- [7] 朱慧,杨茵,陈宇清.HBV前C基因突变及分型与HBV母婴传播关系的研究.中国优生与遗传杂志.2010,18(2):15-17. DOI: 1013404/j.cnki.cjhhh.2010.02.012.
- [8] 武佳欣, 杨志清, 张睿君, 李雁笛, 赵甜静, 扈琳珠, 冯永亮, 丰淑英, 汪波, 王素萍. HBsAg阳性母亲HBV核心启动子突变与宫内传播的关系[J]. 中华流行病学杂志, 2020, 41(6): 902-907
- [9] Liu, H., Chen, S., Liu, X. *et al.* Effect of S-region mutations on HBsAg in HBsAg-negative HBV-infected patients. *Virol J* 21, 92 (2024). <https://doi.org/10.1186/s12985-024-02366-2>
- [10] Wang, T., Dai, Y., Zhang, M., Cui, D., Xu, X., Sun, C., & Cheng, J. (2018). Sequence analysis of the Pre-S gene in chronic asymptomatic HBV carriers with low-level HBsAg Corrigendum in /10.3892/ijmm.2019.4055. *International Journal of Molecular Medicine*, 42, 2689-2699. <https://doi.org/10.3892/ijmm.2018.3831>
- [11] Yin Y, Zhang P, Tan Z, Zhou J, Wu L, Hou H. The Association of Pre-S/S Gene Mutations and Hepatitis B Virus Vertical Transmission. *Hepat Mon.* 2016 Mar

- 6;16(3):e32160. doi: 10.5812/hepatmon.32160. PMID: 27226799; PMCID: PMC4876664.
- [12] 小鹤. 外群的选择问题[EB/OL]. 生信札记. [2025-10-06]. <https://geekdaxue.co/read/wusheng@gw7a9p/qo8ekw>.
- [13] Simon A. Berger, Denis Krompass, Alexandros Stamatakis. Performance, Accuracy, and Web Server for Evolutionary Placement of Short Sequence Reads under Maximum Likelihood, *Systematic Biology*, Volume 60, Issue 3, May 2011, Pages 291–302, <https://doi.org/10.1093/sysbio/syr010>
- [14] Gojobori T, Li WH, Graur D. Patterns of nucleotide substitution in pseudogenes and functional genes. *J Mol Evol*. 1982;18(5):360-9. doi: 10.1007/BF01733904. PMID: 7120431.
- [15] Liu, MY., Yang, YL., Chang, YC. *et al*. Mutation spectrum of *MMACHC* in Chinese patients with combined methylmalonic aciduria and homocystinuria. *J Hum Genet* 55, 621–626 (2010). <https://doi.org/10.1038/jhg.2010.81> [Journal of Human Genetics]
- [16] W.S. DeWitt, K.D. Harris, A.P. Ragsdale, & K. Harris, Nonparametric coalescent inference of mutation spectrum history and demography, *Proc. Natl. Acad. Sci. U.S.A.* 118 (21) e2013798118, <https://doi.org/10.1073/pnas.2013798118> (2021).
- [17] Freek M., Arianne M. B., Jurrian D. K., et al. MutationalPatterns: The one stop shop for the analysis of mutational processes. *BMC Genomics*. 2022. doi:10.1186/s12864-022-08357-3 [Bioconductor]
- [18] Sasani, T.A., Ashbrook, D.G., Beichman, A.C. *et al*. A natural mutator allele shapes mutation spectrum variation in mice. *Nature* 605, 497–502 (2022). <https://doi.org/10.1038/s41586-022-04701-5>
- [19] Liu, MY., Yang, YL., Chang, YC. *et al*. Mutation spectrum of *MMACHC* in Chinese patients with combined methylmalonic aciduria and homocystinuria. *J Hum Genet* 55, 621–626 (2010). <https://doi.org/10.1038/jhg.2010.81> [Journal of Human Genetics]

第 13 章

- [1] Holmudden, M., Gustafsson, J., Bertrand, Y.J.K. et al. Evolution shapes and con-

serves genomic signatures in viruses. *Commun Biol* 7, 1412 (2024). <https://doi.org/10.1038/s42003-024-07098-1>

第 14 章

- [1] Symons, J., Chung, C., Verheijen, B.M. et al. The mutational landscape of SARS-CoV-2 provides new insight into viral evolution and fitness. *Nat Commun* 16, 6425 (2025). <https://doi.org/10.1038/s41467-025-61555-x>

第 15 章

- [1] 武松. SPSS实战与统计思维[M]. 第1版. 北京: 清华大学出版社,2019: 262-272. ISBN: 978-7-302-51322-3

第 18 章

- [1] Xia, H., Luo, H., Shan, C. et al. An evolutionary NS1 mutation enhances Zika virus evasion of host interferon induction. *Nat Commun* 9, 414 (2018). <https://doi.org/10.1038/s41467-017-02816-2>
- [2] Moreno, K.M.F.; de Andrade, V.A.; de Melo Iani, F.C.; Fonseca, V.; Lima, M.T.; de Castro Barbosa, E.; Tomé, L.M.R.; Guimarães, N.R.; Fritsch, H.M.; Adelino, T.; et al. Exploring Microorganisms Associated to Acute Febrile Illness and Severe Neurological Disorders of Unknown Origin: A Nanopore Metagenomics Approach. *Genes* 2024, 15, 922. <https://doi.org/10.3390/genes15070922>
- [3] Xu, H., Zhang, J., Wang, F., Chen, Y., Chen, H., Feng, Y., Hou, G., Zi, J., Zhang, M., Zhou, J., Deng, L., Lin, L., Zhang, X., & Liu, S. (2024). Integration of metagenomics and metaproteomics in the intestinal lavage fluids benefits construction of discriminative model and discovery of biomarkers for HBV liver diseases. *Proteomics*, 24, e2400002. <https://doi.org/10.1002/pmic.202400002>

