2023 年武汉理工大学大学生数学建模竞赛

基于多元回归预测与聚类分析的糖尿病风险评估模型

摘要

针对问题一，首先对附件 1 的数据进行数据清洗，本文先运用拉伊达原则对附件一的数据进行异常值检验并对离群值进行分析，再统计各数据项的缺失量及其占比，采用基于插补的**热卡填充**方法对缺失值进行**插补**，完成数据预处理后，本文先采用 Lasso 回归的方法对数据进行特征选择及降维处理得到 23 个初步筛选得到的主要变量指标，然后再运用**相关性分析**的方法分析各个变量与血糖值的相关性大小，最后筛选得出年龄、甘油三酯、血红蛋白等 12 个主要变量指标并结合医学知识对其进行合理性分析。

针对问题二，首先，同样需要对数据进行预处理，再进行特征选择和降维处理，以减少特征的维度和去除冗余信息。处理好数据后，本文同时建立了三种模型对血糖值进行预测，分别为 BP 神经网络、支持向量机的线性回归与多元回归分析这三种血糖值预测模型，得到三种预测结果后再通过计算三种模型的均方误差的方法检验三种模型预测结果的准确性，比较得出 **BP 神经网络预测模型**的预测结果最为准确并采用其血糖值预测结果。

针对问题三，由于血糖值是评估糖尿病的风险的重要指标，本文采用 **K-means 聚类算法**建立聚类模型对附件 1 中的血糖值进行聚类评级，利用**肘部法**选取出合适的最佳聚类数 K 值为 4，从而根据聚类结果将糖尿病的风险等级划分为低风险、中风险、中高风险，高风险。再对附件 1 的样本进行糖尿病风险评估，得出各个风险等级的人群体占比，并根据不同风险阶段的人给出不同意见。由于附件 1 中只有少部分人测量乙肝的相关指标，而乙肝患者比正常人患糖尿病的风险要大，所以假设检测乙肝相关指标的样本是**乙肝**潜在患者，将其**单独考虑**并评估其糖尿病风险。

针对问题四，首先对附件 2 进行**数据清洗**，处理方法与附件 1 相同，利用问题二中经比较得出预测效果最好的 **BP 神经网络模型**对血糖值进行预测，再利用问题三的**基于 K-means 聚类算法的风险评估模型**来对附件 2 中的样本糖尿病风险进行评估，并将检测**乙肝**的相关指标的样本**单独考虑**。

本文的现实意义在于，通过建立预测和评估模型，可以一定程度上根据个体的体检数据预测其血糖值来评估患糖尿病的风险。这对于早期发现潜在的糖尿病患者非常重要，有助于采取预防措施和进行早期干预，以减少疾病的发展和并发症的风险，促进糖尿病的预防、管理和公共卫生水平的提升。

关键词：数据清洗 相关性分析 BP 神经网络 K-means 聚类算法 风险评估

目录

一、 问题重述

1.1 问题背景

糖尿病作为一种代谢性疾病，其特征为患者的血糖长期高于标准值，造成糖尿病的原因是胰腺产生不了足够的胰岛素或者人体无法有效地利用胰岛素。血糖正常值是指人空腹的时候血糖值范围为 3.9～6.1 毫摩尔/升，判断是否有高血糖的方法一般是对人体进行两次重复测量，血糖值大于 6.7 毫摩尔/升即可诊断为糖尿病。糖尿病临床表现为频尿、容易口渴、容易饥饿，同时伴随并发症：心血管疾病、非酮症之超渗透压的昏迷、糖尿病酮症酸中毒、中风、慢性肾脏病、足部溃疡等。目前糖尿病种类主要分为：1 型糖尿病、2 型糖尿病、妊娠糖尿病和其他类型糖尿病。作为一种常见慢性疾病，糖尿病目前无法根治，需要通过科学有效的干预、预防和治疗，来降低发病率和提高患者的生活质量。2021 年有关数据显示，全球成年糖尿病患者人数达到 5.37 亿（10.5%），约十分之一的成年人受到影响。相比 2019 年，糖尿病患者增加了 7400 万，增幅达 16%，突显出全球糖尿病患病率的惊人增长。过去的 10 年间（2011 年~2021 年），我国糖尿病患者人数由 9000 万增加至 1 亿 4000 万，增幅达 56%，其中约 7283 万名患者尚未被确诊，比例高达 51.7%。

1.2 问题要求

基于上述背景以及所获取的数据需要建立数学模型解决以下问题：

问题 1：结合附件 1 的检测数据，从 42 个检测指标中筛选出主要变量指标，并请详细说明建模主要变量的筛选过程及其合理性。

问题 2：结合附件 1 的检测数据，根据体检数据建立血糖值的预测模型。

问题 3：结合附件 1 的检测数据，根据体检数据对糖尿病的风险进行评估。

问题 4：结合附件 2 的检测数据，对血糖值进行预测，并对糖尿病风险进行评估。

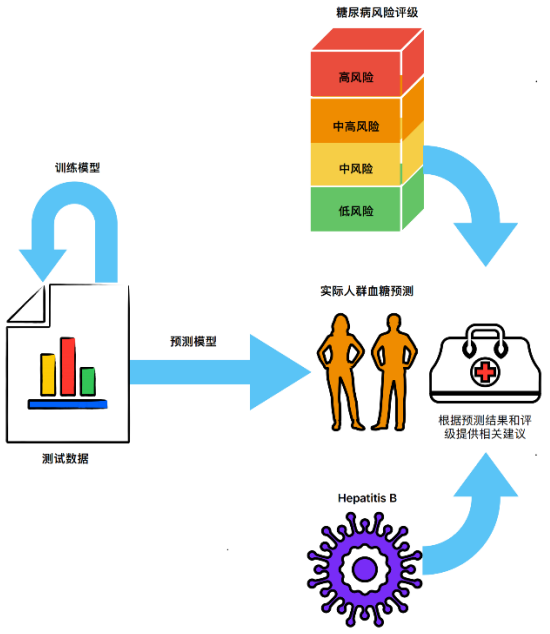

图 1 糖尿病风险评估图

二、 问题分析

2.1 问题一的分析

本问题的目标是从附件一的检测数据中筛选出主要变量指标，以便建立糖尿病预测模型。首先，需要进行数据预处理，包括缺失值处理和异常值处理，以确保数据的准确性和一致性。考虑到附件一数据量庞大，针对缺失值，可以考虑先对各个数据项缺失量进行统计，然后采用插补的方法处理缺失值，针对异常值，先利用拉依达原则来统计异常值并将异常值进行均值替换进而完成数据清洗。针对变量筛选过程，可以采用特征选择方法来确定主要变量指标。由于变量指标过多，考虑先用 Lasso 回归进行变量筛选对数据进行降维处理，再用相关性分析的方法筛选出主要变量指标。

2.2 问题二的分析

针对问题二，本问题的目标是基于附件一的检测数据建立血糖值的预测模型。首先，同样需要对数据进行预处理，同时需要进行特征选择和降维处理，以减少特征的维度和去除冗余信息。建立预测模型可以考虑采用多种回归方法，例如线性回归、Lasso 回归、支持向量回归（SVR）等，同时也可以考虑用机器学习的方法对血糖值进行预测。选择合适的回归方法应考虑数据的特点和模型的性能指标，如拟合优度、预测准确性等。

2.3 问题三的分析

问题三的目标是基于附件 1 的检测数据，利用体检数据对糖尿病的风险进行评估。在进行糖尿病风险评估时，可以采用聚类分析的方法。聚类分析是一种无监督学习方法，通过将样本划分为不同的类别或群组，发现数据中的内在结构和模式。在本问题中，可以利用体检数据中的多个指标，对样本进行聚类，将相似特征的个体划分到同一类别中。再选择适当的聚类算法，如 K-means 算法、层次聚类等，对预处理后的数据进行聚类分析。通过调整聚类的类别数量，可以得到不同数量的糖尿病风险评级。

根据聚类结果，可以对每个聚类簇进行分析和解释，了解不同簇的特征和指标的差异性。进一步，可以计算每个簇内的糖尿病患病率或风险指数，以评估不同簇的糖尿病风险水平。较高风险簇的个体可能需要更密切的监测和干预，为了可视化糖尿病风险评估结果，可以绘制柱状图或饼状图，展示不同风险等级的个体数量或比例。这样可以直观地比较不同风险等级之间的差异，并为医生和决策者提供参考，以制定相应的糖尿病预防和管理策略。

2.4 问题四的分析

本问题的目标是利用附件 2 的检测数据对血糖值进行预测，并进一步评估个体的糖尿病风险。通过建立预测模型，可以根据个体的特征和指标，先预测其血糖水平，再判断其患糖尿病的可能性。首先，对附件 2 的检测数据进行预处理，包括处理缺失值和异常值。根据问题描述，附件 2 的数据是无血糖值的检测数据，因此需要将附件 1 中的血糖值作为目标变量与附件 2 的数据进行合并。通过使用问题二中建立的预测模型，将附件 2 的检测数据作为特征变量，附件 1 中的血糖值作为目标变量，进行训练和拟合，从而建立血糖值的预测模型。最后，根据预测的血糖值和糖尿病的诊断标准，可以对个体的糖尿病风险进行评估。例如，将预测的血糖值与糖尿病的阈值进行比较，判断个体是否属于糖尿病患者或高风险群体。

三、 模型假设

（1） 假设每个样本间相互独立、互不关联，并且他们的各指标之间互不干扰；

（2） 假设本题附件中提供的数据均真实可信；

（3） 假设数据进行预处理后模型求解不再受数据误差的影响；

（4） 假设数据一定程度上服从正态分布。

四、 符号说明

符号	说明
F	主要变量指标
BG	血糖值
w	权重
b	偏置
$\Phi(F)$	映射函数
ξ_i，ξ_i^*	松弛因子
MSE	均方误差

五、 模型的建立与求解

5.1 问题一

5.1.1 数据清洗

数据清洗就是检查数据是否存在缺失值、重复值、异常值等问题并对这些存在问题的数据进行相应处理，使其适合进行分析和建模，提高数据的准确性和可靠性。数据清洗通常是数据处理过程的一个必要步骤，它可以消除数据错误和噪声，并提高分析和建模的精度。

异常值分析

通过观察采集到的数据，经常会发现一些不符合客观规律的数值，或者与其他大部分的数据有较大的差异，这种数据称之为异常值。

在假设附件一 42 个数据项检查数据在一定程度上服从正态分布的前提下，我们可以利用拉依达原则来统计异常值，基本原理为：当某个样本个体的残差绝对值 $v_i \geq 3\sigma$ 时的概率为 99.7%，而不在 3σ 区域的概率仅为 0.3%时,可将其视为异常值[1]。

缺失值处理

数据缺失的内在机理可以划分 3 种类型[2], 分别是完全随机缺失(MCAR)、随机缺失(MAR)和非随机缺失(MNAR).其中 MAR 相比于 MCAR 更加常见和符合现实，也是缺失值处理方法最主要的研究对象。MAR 指仅在某个特定的组内缺失值是随机产生的，而不同组之间不一定是随机的。对应到附件一检测数据，可以发现其中数据并非完全随机缺失。

在实际中，减少缺失值的最简单有效的方法就是对无回答进行事前预防，尽量降低无回答率[3]。尽管如此还是会不可避免的出现缺失值，因此如何对缺失值进行处理显得尤为重要。将缺失值的处理方法大致分为以下四类：第一类是加权的方法，第二

类是个案删除方法，第三类是基于插补的方法，第四类是基于模型的方法。

在处理缺失值之前，本文先对 42 个数据项进行分类再对各个类别数据项的缺失数进行了统计，下图是五个类别数据缺失值的占比情况：

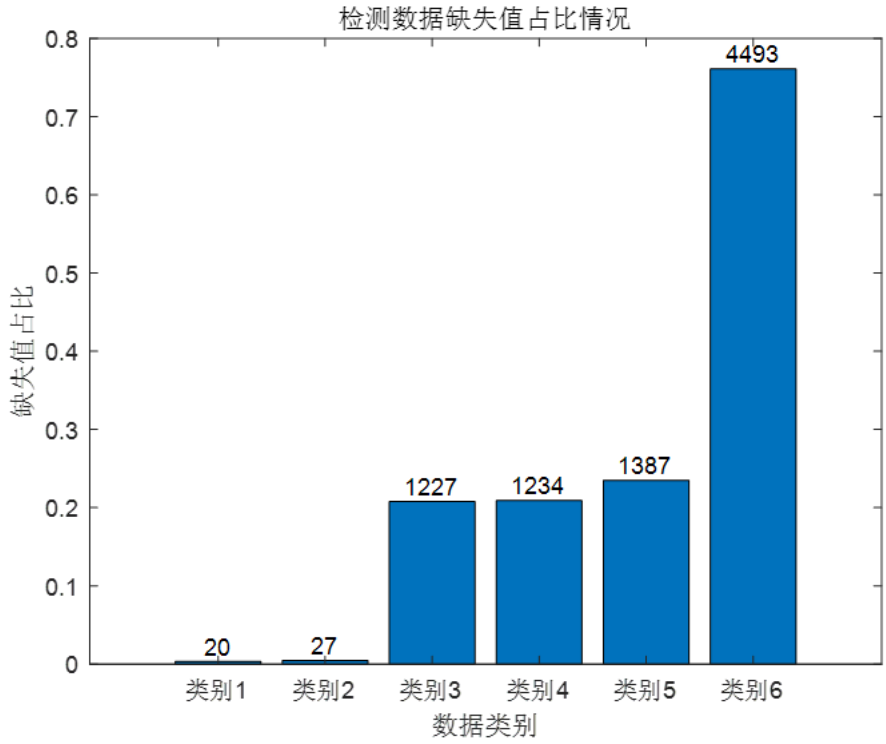

图 2 缺失值占比情况

数据项分类及缺失数分别为：

类别 1：红细胞体积分布宽度、红细胞压积、红细胞平均体积、红细胞平均血红蛋白浓度、红细胞平均血红蛋白量、红细胞计数、白细胞计数、血小板计数、血红蛋白、单核细胞、嗜碱细胞、嗜酸细胞、中性粒细胞、淋巴细胞

类别 2:血小板体积分布宽度、血小板平均体积、血小板比积

类别 3:低密度脂蛋白胆固醇、高密度脂蛋白胆固醇、甘油三酯、总胆固醇

类别 4:丙氨酸氨基转换酶、天门冬氨酸氨基转换酶、r 谷氨酰基转换酶、总蛋白球蛋白、白球比例、碱性磷酸酶、白蛋白

类别 5:肌酐、尿素、尿酸

类别 6:乙肝 e 抗体、乙肝 e 抗原、乙肝核心抗体、乙肝表面抗体、乙肝表面抗原

本文主要运用基于插补的热卡填充方法来对缺失值进行插补。

热卡填充（Hot deck imputation）也叫就近补齐，对于一个包含空值的对象，热卡填充法在完整数据中找到一个与它最相似的对象，然后用这个相似对象的值来进行填充。通常会找到超出一个的相似对象，在所有匹配对象中没有最好的，而是从中随机的挑选一个作为填充值。不同的问题选用不同的标准对相似来进行判定，经常采用的是使用相关系数矩阵来确定哪个变量(比如变量 y)与缺失数据所在变量(比如变量 x)最相关。然后把所有个案按 y 值的大小进行排序。那么变量 x 的缺失数据就可以用排在缺失数据前的那个个案的数据来代替了[4]。

处理完后的数据各数据项分布如下图：

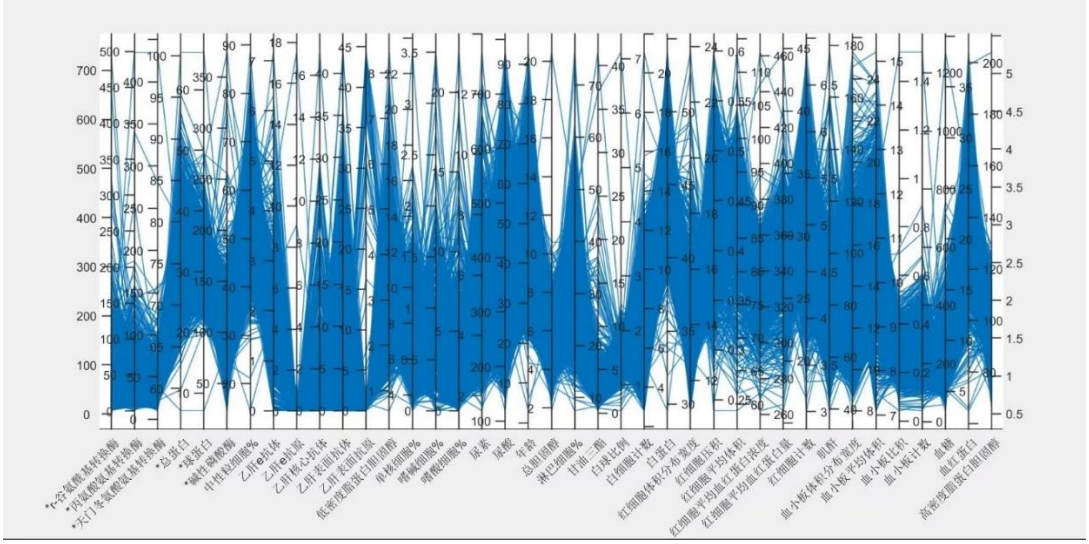

图 3 预处理后数据分布

5.1.2 筛选主要变量

由于衡量糖尿病的指标是血糖值，本文从附件 1 中剩余的 41 个变量指标中筛选出影响血糖值的主要变量指标，由于指标数量较多，各个变量指标之间关系错综复杂，可能存在多重共线性的问题。Lasso 回归是一种用于特征选择和回归分析的机器学习算法。它可以对具有大量特征的数据集进行特征选择，从而找到最主要的变量指标，可以提高模型的泛化能力。本文先采用 Lasso 回归来进行降维处理。

Lasso 回归通过加入 L1 正则化项来实现特征选择，利用交叉验证确定合适的正则化参数和最优的惩罚函数。其次，最优的惩罚函数将一些变量指标的回归系数进行惩罚，使其绝对值之和小于常数，促使模型中残差平方和最小化，从而将某些变量指标的系数压缩至 0，排除在外。最终根据系数的大小来判断指标的重要性。较大的系数表示对血糖值有更大的影响，较小的系数表示对血糖值的影响较小或无影响，以此达到对附件 1 中多元高维变量指标的降维效果，初步得出影响血糖值的 23 个主要变量指标[5]。

变量筛选主要是通过统计方法从繁多的变量中选出对响应变量最大的解释变量，它是统计分析和推断的重要环节。变量筛选的结果好坏直接影响模型的质量，进而对统计分析与预测精度产生极大的影响。在对数据进行降维处理后，本文继续通过建立相关系数矩阵并比较不同变量与血糖的相关系数，进而筛选出主要变量。

在研究两个变量的相关性时，可以使用以下方法：

1.皮尔逊相关系数（Pearson Correlation Coefficient）：用于度量两个连续型变量之间的线性相关性。然而，如果两个变量之间的关系是非线性的，皮尔逊相关系数可能无法准确捕捉到相关性。

2.斯皮尔曼等级相关系数（Spearman's Rank Correlation Coefficient）：它通过将每个变量的取值转化为等级或排序来度量变量之间的相关性，因此它适用于非线性关系。斯皮尔曼等级相关系数基于变量的秩次，而不是具体的数值，因此不受异常值的影响。

本文采用建立皮尔逊相关系数矩阵的方法来对降维处理后的 23 个变量指标与血糖值的联系。

下图是相关系数矩阵分析流程图：

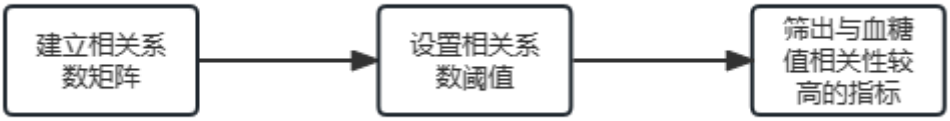

图 4 相关系数矩阵建立流程图

首先建立相关系数矩阵分析各个指标与血糖值之间的相关性。通过计算每个指标与血糖值的相关系数，我们可以初步了解各个指标与血糖值之间的线性关系。

下面为建立相关系数矩阵的计算过程：

$$\mathrm{E}\,(X) = \bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{1}$$

$$\sigma_X^2 = \frac{1}{\mathrm{n}-1}\sum_{\mathrm{i}=1}^{\mathrm{n}} (\mathrm{X_i} - \bar{\mathrm{X}})^2 \tag{2}$$

$$\mathrm{cov(X,Y)} = \frac{1}{\mathrm{n}-1}\sum_{\mathrm{n}}^{\mathrm{i}=1}(\mathrm{X_i} - \bar{X})\,(\mathrm{Y_i} - \bar{Y}) \tag{3}$$

$$\mathrm{r} = \frac{\sum_{i=1}^{n}(\mathrm{X}_i - \bar{X})\,(\mathrm{Y}_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}\,(\mathrm{X}_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}\,(\mathrm{Y}_i - \bar{Y})^2}} \tag{4}$$

$$X_k = \begin{bmatrix} x_{11} & \cdots & x_{1i} & \cdots & x_{1(n-k)} \\ \vdots & & \vdots & & \vdots \\ x_{m1} & & x_{mi} & & x_{m(n-k)} \\ \vdots & & \vdots & & \vdots \\ x_{k1} & \cdots & x_{ki} & \cdots & x_{k(n-k)} \end{bmatrix} \tag{5}$$

然后设置相关系数阈值。根据实际需求和领域知识，我们可以设置一个合适的相关系数阈值，这里我们设置为 0.12。这个阈值决定了我们认为与血糖值相关性较高的指标。

最后筛选出与血糖值相关性较高的指标。通过遍历相关系数矩阵，选择相关系数大于阈值的指标作为主要变量指标。这些指标与血糖值具有较强的线性关系，可能对血糖值的预测具有较高的影响力。

通过这种方法筛选得到主要变量及对应的相关系数如下表

表 1 主要变量的相关系数

指标	相关系数
碱性磷酸酶	0.1332
低密度脂蛋白胆固醇	0.1369

尿素	0.1303
年龄	0.2498
性别	0.1422
甘油三酯	0.1724
红细胞压积	0.1222
红细胞平均血红蛋白浓度	0.1343
红细胞计数	0.1235
*r-谷氨酰基转换酶	0.1308
血红蛋白	0.1469

考虑到血红蛋白是红细胞的主要成分，我们可以将红细胞压积与红细胞平均血红蛋白浓度作为红细胞计数和血红蛋白的因变量，因此将这两个指标略去并排序得到各个主要变量指标的相关系数及大小排列如下。

表 2 主要变量指标的相关系数

指标	相关系数
年龄	0.2498
甘油三酯	0.1724
血红蛋白	0.1469
性别	0.1422
低密度脂蛋白胆固醇	0.1369
碱性磷酸酶	0.1332
*r-谷氨酰基转换酶	0.1308
尿素	0.1303
红细胞计数	0.1235
总胆固醇	0.1144

即通过建立相关系数矩阵分析的方法我们得到的重要特征指标为：年龄、甘油三酯、血红蛋白、性别、低密度脂蛋白胆固醇、碱性磷酸酶、*r-谷氨酰基转换酶、尿素红细胞计数以及总胆固醇。

5.1.3 评价合理性

通过筛选相关性较高的指标，可以减少变量的数量，提高模型的简洁性和可解释性。相关性较高的指标可能与血糖代谢过程密切相关，可以提供关于血糖水平的有用信息。这些指标可能是糖尿病的影响因素或指标，具有一定的生物学意义。比如，随着年龄的增长，机体的代谢率可能下降，血糖控制可能会受到影响，增加患糖尿病的风险，高水平的甘油三酯可能与胰岛素抵抗、肥胖和糖尿病发展相关，可能会对血糖水平产生负面影响；男性和女性在血糖控制方面可能存在一些差异，但具体影响因素复杂，可能受到激素、体脂分布、生理差异等多种因素的影响；血红蛋白水平与血糖控制没有直接关系，但在长期高血糖情况下，血红蛋白糖化（糖化血红蛋白）水平可能升高，进而影响血糖水平；高水平的 LDL-C 和总胆固醇可能与胰岛素抵抗和糖尿病风险增加相关，高胆固醇水平也可能增加动脉粥样硬化的风险，从而影响血糖代谢，

等等。

因此，可以认为通过相关性分析筛选出的主要变量指标具备一定的合理性。

5.2 问题二

要在多个变量影响血糖值的情况下对血糖值进行预测，可以适用的模型主要有：统计模型、数值模型、灰色模型、机器学习模型和组合预测模型等[6]，本文针对问题二采用 BP 神经网络、支持向量机的非线性回归以及线性回归模型分别对血糖值进行预测。

5.2.1 BP 神经网络

人工神经网络（ANN）模型是一种模拟人脑并根据自然神经网络的特性而建立的预测模型[7]。神经网络模型将网络分为三个层次，分别是输入层、隐藏层和输出层 ，其中输入层与输出层的节点数往往是固定的，中间层则可以自由指定。结构图如下。

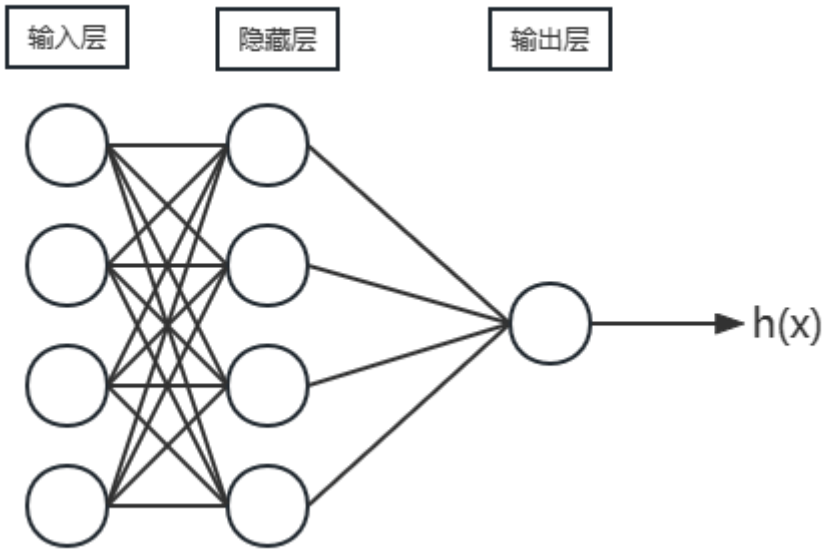

图 5 神经网络结构图

注：

(1)结构图中的拓扑与箭头代表着预测过程时数据的流向，跟训练时的数据流有一定的区别；

(2)结构图中的圆圈代表“神经元”，连接线代表“神经元”之间的连接，每个连接线对应着不同的权重，权值需要训练得到。

根据计算方法、用途等不同，可将人工神经网络模型分为基于前向型的后向传播神经网络（BP 网络）、基于随机型的模拟退火神经网络（SA-ANN）、基于反馈型的霍普菲尔德神经网络（Hopfield-ANN）等。其中，BP 神经网络模型具有很强的非线性映射能力及很好的鲁棒性和容错性。

根据问题一可知，检测指标与血糖值之间并不是简单的线性关系，所以本文采用 BP 神经网络模型对血糖值进行预测，其核心步骤可以由下面的流程图表示。

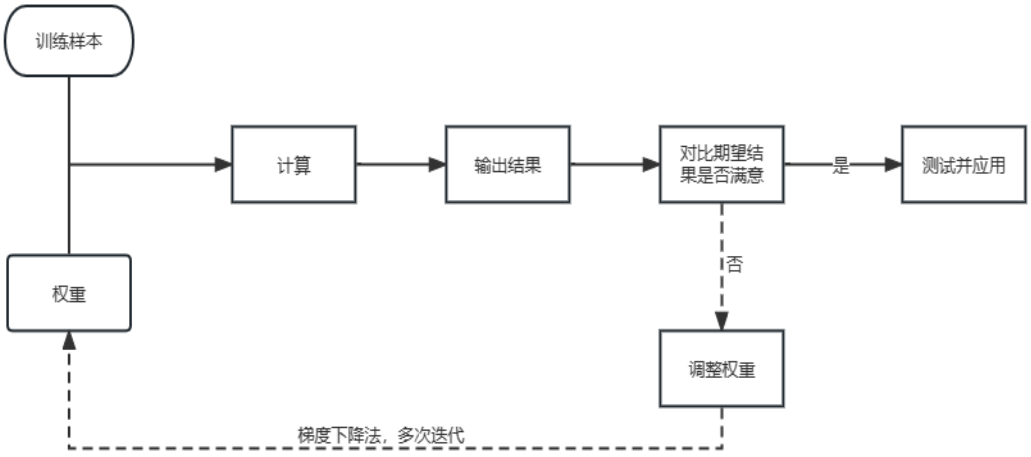

图 6 BP 神经网络流程图

我们筛出由问题一已经得出的主要变量指标的数据和血糖值，将其作为训练样本，放入 BP 神经网络中进行多次迭代训练，直到得到满意的预测结果。

5.2.2 支持向量机非线性回归

支持向量机回归(SVR)是支持向量机在回归问题上的应用模型。我们针对本题建立支持向量机回归的非线性回归模型，首先利用核函数，将非线性问题转化为线性，从而在高维特征空间中构造出最优分离超平面提高了对非线性问题的处理效果，避开了显示表达式，具有算法简单、数据计算量小、易于实现等特点。更重要的是，本问题数据有多个维度，而回归函数由支持向量的样本所确定，因此不会因样本维数的增加导致计算量的剧增，不会出现维数灾难，适合本题情况。

该模型的数据集为

$$D = \{(F_{i\times j}, BG_i)\}_{j=1}^{n} \in R^n \times R, i = 1,2 \dots m \tag{6}$$

式中，$F_{i\times j} = \{F_{i1}, F_{i2} \dots F_{in}\}$为问题一中的出的主要变量指标的数据，即输入向量；$BG$为血糖值，即输出向量;n 为主要变量指标的数量, m 为样本总数。

选取核函数$K(x,y) = \exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right)$代替内积运算，通过非线性映射将训练集数据映射到一个高维特征空间，在高维特征空间进行线性回归,定义非线性映射为

$$f_c(F) = \omega\Phi(F) + b \tag{7}$$

式中w为垂直与超平面的权向量即权重，b为偏置, $\Phi(F)$为映射函数。

选取合适的惩罚系数,为了减少干扰，增加松弛因子ξ_i，ξ_i^*，构造出凸二次规划问题

$$\min\{\frac{1}{2}\mid\omega\mid^2+C\sum_{i=1}^{m}(\xi_i+\xi_i^*)\}$$
$$s.t.\quad\begin{cases}\omega\Phi(F_i)+b-BG_i\le\varepsilon+\xi_i\\ BG_i-\omega\Phi(F_i)\le\varepsilon+\xi_i^*\\ \xi_i,\xi_i^*\ge 0,i=1,2,\dots,m\end{cases}\tag{8}$$

引入拉格朗日函数 L,

$$L(\omega,\xi_i,\xi_i^*,b)=\frac{1}{2}\mid\omega\mid^2+C\sum_{i=1}^{l}(\xi_i+\xi_i^*)-\sum_{i=1}^{l}\alpha_i(\varepsilon+\xi_i-y_i+(\omega\cdot x_i)+b)-\sum_{i=1}^{l}\alpha_i^*(\varepsilon+\xi_i^*+y_i-(\omega\cdot x_i)-b)-\sum_{i=1}^{l}(\eta_i\xi_i+\eta_i^*\xi_i^*)\tag{9}$$

上式对参数ω，b，ξ_i，ξ_i^*的偏导数都应等于零，即：

$$\begin{cases}\dfrac{\partial L}{\partial\omega}=\omega-\sum_{i=1}^{m}(\alpha_i-\alpha_i^*)F_i=0\\ \dfrac{\partial L}{\partial b}=\sum_{i=1}^{m}(\alpha_i-\alpha_i^*)=0\\ \dfrac{\partial L}{\partial\xi_i}=C-\alpha_i-n_i=0\\ \dfrac{\partial L}{\partial\xi_i^*}=C-\alpha_i^*-\eta_i^*=0\end{cases}\tag{10}$$

将（11）带入（9）得到对偶优化问题为

$$Q(\alpha,\alpha^*)=-\varepsilon\sum_{i=1}^{m}(\alpha_i^*+\alpha_i)+\sum_{i=1}^{m}BG_i(\alpha_i^*-\alpha_i)-\frac{1}{2}\sum_{i,j=1}^{m}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_i^*)K(\mathrm{F}\cdot F_i)\tag{11}$$

相应的预测函数则为：

$$f(F)=\omega\Phi(F)+b=\sum_{i=1}^{m}(\alpha_i-\alpha_i^*)K(F\cdot F_i)+b \tag{12}$$

式中$(\alpha_i-\alpha_i^*)\neq 0$对应的样本就是支持向量。

5.2.3 多元回归分析

在问题二中血糖值受多个变量指标影响，因此我们在研究血糖值与其他变量指标之间的关系时，可以采用多元回归分析方法。

若通过实验或观测获得成批的离散数据，所谓的拟合问题实质上就是为这些离散的数据建立对应的、近似的连续模型，一般建立的连续模型为一个函数表达式或一条曲线。其中插值方法是比较古典的拟合方法之一,由于获取的数据往往是离散的数据点，要建立与之对应的连续模型，插值的拟合方法要求目标函数必须过已知的离散点，从而建立连续函数对非插值点进行近似计算。由于目标函数要求必须过已知离散点，所以拟合出来的图像一般欠缺圆滑度，而最小二乘法在拟合问题中，只要求目标函数近似已知离散数据点的分布总体轮廓，并不要求一定要过已知的离散数据点，其拟合精确性在于尽可能地近似已知离散数据点，即与已知数据点的误差按某种意义尽可能的小，通常采用误差的平方和最小的原则[7]。

本问题我们决定将问题一得出的主要变量指标作为自变量，将血糖值作为因变量，采用最小二乘法来估计回归模型的系数，其形式为

$$S(F)=a_0+a_1\varphi_1(F_1)+a_2\varphi_2(F_2)+\cdots a_n\varphi_n(F_n)=\sum_{i=0}^{n}a_i\varphi_i(F_i)+\varepsilon \tag{13}$$

式中$S(F)$是因变量，F_i是自变量，a_i是回归系数，$\varphi_i(F_i)$是函数族，ε是随机误差项。n 为主要变量指标的数量。

最小二乘法的目标是寻找一个$S(F)$的使得误差平方和最小化

$$\begin{aligned} I=\min\left|\left|\delta^2\right|\right|_2^2&=\sum_{j=0}^{m}\delta^2=\sum_{j=0}^{m}\left[S\left(F_{i\times j}\right)-BG_j\right]^2\\ &=\sum_{j=0}^{m}\left[a_i\varphi_i(F_i)-BG_j\right]^2\ (i=0,1,2\ldots,n)\end{aligned} \tag{14}$$

式中δ是残差，即预测值与实际值的差值，m 为样本总数。

由求多元函数的极值的必要条件，有

$$\frac{\partial I}{\partial a_k}=2\sum_{i=0}^{m}[\sum_{j=0}^{n}a_i\varphi_i(F_i)-BG_j]\varphi_i(F_i)=0\ ,k=1,2\ldots,n \tag{15}$$

为了求解上述最小化问题，可以对求偏导数，得到 n+1 个方程组成的方程，经求解最终求得回归系数的估计值和多元回归模型的拟合结果即预测值。下图分别为多元回归分析、BP 神经网络、支持向量机的非线性回归模型所得出的预测值与实际值的散点图：

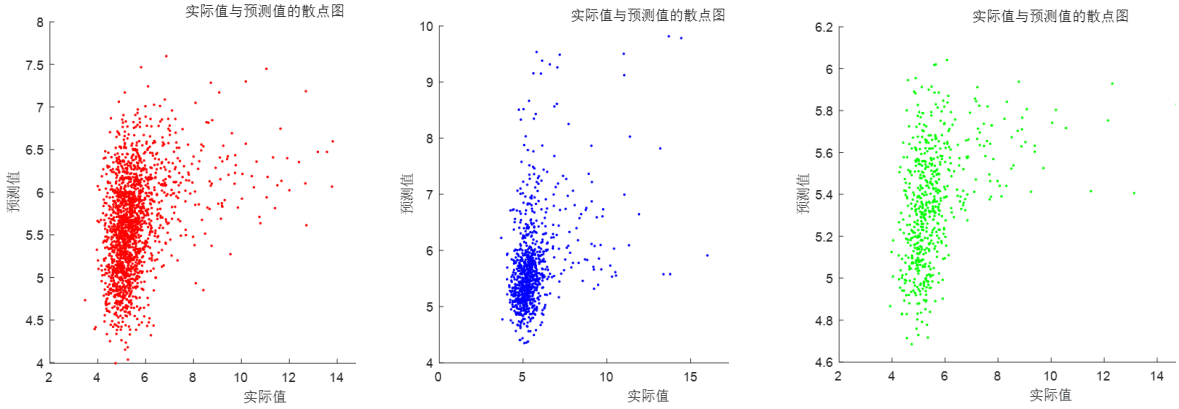

图 7 三种模型预测值与实际值散点图

5.2.4 模型评估

为了检验 BP 神经网络、支持向量机非线性回归、线性回归三个模型所预测结果的准确性，本文主要采用平均绝对误差（MSE）、均方根差（RMSE）和平均绝对百分误差（MAPE）来进行模型检验，公式如下[8]

$$MSE = \frac{1}{n}\sum_{k=1}^{n} | t_k - y_k | \tag{16}$$

$$RMSE = \sqrt{\frac{1}{n}(\sum_{k=1}^{n}(t_k - y_k)^2)} \tag{17}$$

得到三个预测模型的误差如下表

表 3 三种预测模型的误差

预测模型/评估指标	多元回归	神经网络	向量机
均方根误差(RMSE)	1.2512	1.1600	1.2750
均方误差（MSE）	1.5655	1.3456	1.6255

由上表可以看到，BP 神经网络预测模型的误差是三个模型中误差相对最小的，因此采用神经网络的预测结果。

5.3 问题三

在进行糖尿病风险评估时，可以采用聚类分析的方法。聚类分析是一种无监督学习方法，通过将样本划分为不同的类别或群组，发现数据中的内在结构和模式。

5.3.1 模型建立

K-Means 聚类算法是一种无监督学习，同时也是基于划分的聚类算法，一般用欧式距离作为衡量数据对象间相似度的指标，相似度与数据对象间的距离成反比，相似度越大，距离越小该算法的主要作用是将相似的样本自动归到一个类别中，划分为若干个通常是不相交的子集，每个子集称为一个“簇（cluster）”，聚类既能作为一个单独过程，用于找寻数据内在的分布结构，也可作为分类等其他学习任务的前去过程[9]。

根据问题背景，判断是否有高血糖的标准一般是对人体进行两次重复测量，血糖值大于 6.7 毫摩尔/升即可诊断为糖尿病，因此，血糖值是评估糖尿病的风险的重要指标。本文采用 K-means 聚类算法建立聚类模型对附件 1 中的血糖值进行聚类评级。

下图是 K-means 聚类算法的基本流程

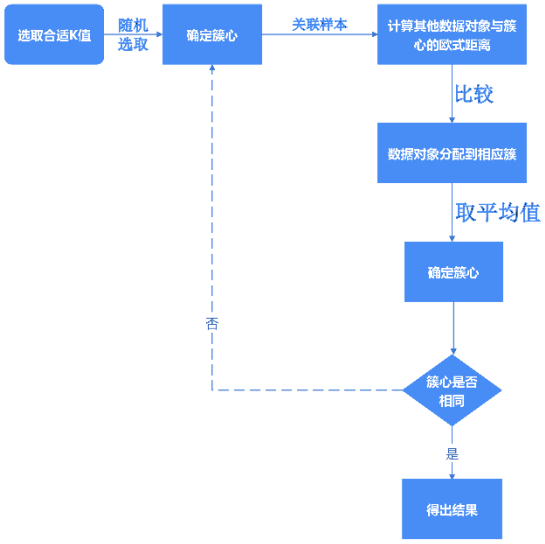

图 8 K-means 基本流程

主要步骤[10]如下：

Step1:随机地从 N 个样本数据中选择 K 个对象，其中每个对象均代表一个簇的初始均值或簇心；

Step2:对剩余的对象，计算其到各个簇心的欧氏距离，将其分配至距离最近的簇中；

Step3:取每个聚类中的样本均值作为新的簇心。

接下来，依次重复步骤 2 和 3 直到簇的均值不再发生变化，聚类中心不再改变。

其中，两个 n 维向量$x = (x_1, \dots, x_n)$和$y = (y_1, \dots, y_n)$之间的欧氏距离$d(x, y)$定义如下：

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{18}$$

根据此算法，对体检数据进行聚类处理。首先对已有血糖信息数据特征提取，提取与血糖值相关的特征信息，按照算法进行归类处理，计算聚类中心，反复迭代后，最终输出聚类结果。

5.3.2 K 值的选取

定义聚类类别数即选取合适的 K 值是 K-Means 聚类算法的关键步骤。

本文采用肘部法进行确定 K 值，K 值的最优解是以成本函数最小化为目标，成本函数为各个类畸变程度之和，每个类的畸变程度等于该类重心与其内部成员位置距离的平方和，在这个平方和随 K 值的变化过程中，会出现一个拐点也即“肘”点，下降率突然变小时即认为是最优的 K 值。

本文对附件 1 中的血糖值进行聚类，绘制肘部图如下图所示，其横坐标为聚类类别数 K，纵坐标为畸变程度。

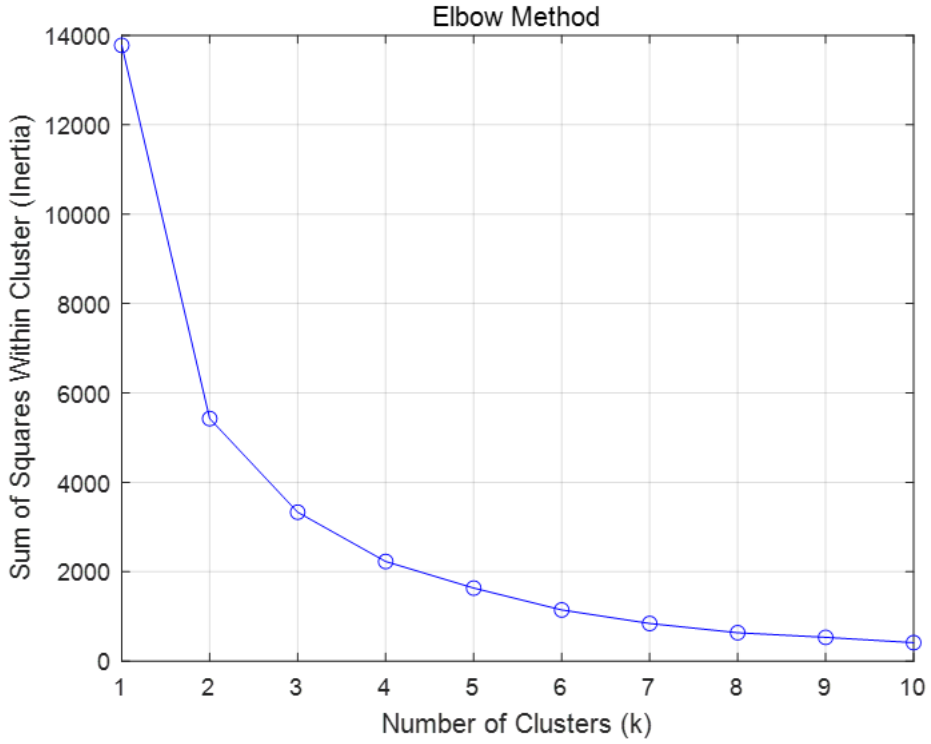

图 9 肘部图

由图可见，当类数从 1 增加到 4 时，总畸变程度下降较快；但当类别数超过 4 时，总畸变程度变化趋势变缓。对此，本文确定 K=4 为最佳聚类数。

5.3.3 输出结果与分析

根据 K-Means 聚类结果将糖尿病的风险等级划分为低风险、中风险、中高风险，高风险,分别对应下图的 1、4、3、2 等级。下表为输出得到的风险等级及对应血糖值范围。

表 4 风险等级及血糖范围

风险级别	低风险	中风险	中高风险	高风险
样本数量	3947	1653	246	59
最低血糖值	3.07	5.59	7.86	12.6
最高血糖值	5.58	7.84	12.51	38.43

聚类分析所得数据可视化如下图所示。

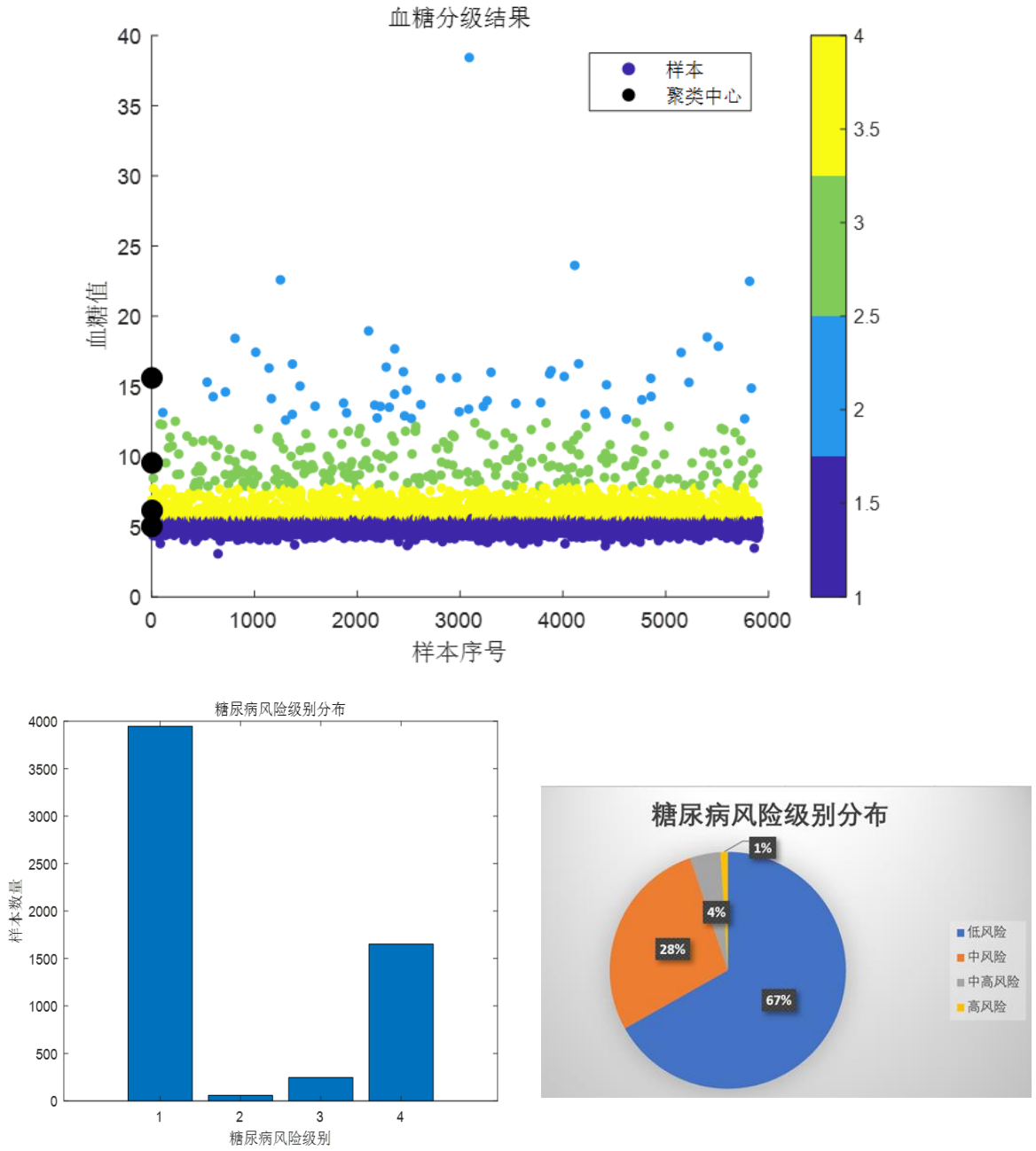

图 10 风险级别分布图

根据糖尿病的血糖值判断标准 6.7 毫摩尔/升来对糖尿病风险进行评估，由可视化分析所得图可知样本中低风险与中风险的人占多数。

低风险等级的人群通常没有明显的糖尿病风险因素或仅存在少量风险因素。他们可能有健康的生活习惯，包括均衡的饮食、适度的体育锻炼和正常体重。血糖水平通常在正常范围内，没有糖尿病的症状或并发症。中风险等级的人群可能有一些潜在的糖尿病风险因素，如家族病史、超重或肥胖、不健康的饮食习惯等。他们的血糖水平可能略高于正常范围，但还未达到糖尿病的诊断标准，针对这个等级的人群，建议进行定期的血糖监测、生活方式改变和相关的健康咨询。

中高风险等级的人群具有明显的糖尿病风险因素，如家族病史、肥胖、高血压、高胆固醇等。他们的血糖水平可能接近糖尿病的诊断标准，但尚未达到明确的糖尿病

诊断。针对这个等级的人群，建议进行更加严格的血糖监测、定期的体检和相关的医疗指导。高风险等级的人群具有较高的糖尿病风险因素，如家族遗传、肥胖、高血压、高胆固醇等。他们的血糖水平可能已经达到或超过糖尿病的诊断标准，或已经被确诊为糖尿病。针对这个等级的人群，建议进行密切的血糖监测、规范的治疗方案、定期的随访和相关的营养指导。

5.3.4 乙肝患者的单独考虑

假设附件 1 中测量乙肝表面抗原、乙肝表面抗体、乙肝 e 抗原、乙肝 e 抗体、乙肝核心抗体指标的 id 都有可能患乙肝。

经查找资料，肝功能的异常干扰了糖原的分解和异生以及葡萄糖的生成和利用，引起糖耐量异常，血糖升高。所以肝炎病人较正常人有更高的风险患糖尿病，如肝源性糖尿病，有必要单独考虑。

5.4 问题四

5.4.1 数据预处理

对附件 2 的数据清洗与附件 1 方处理法相同，对缺失值进行热卡填充，检验出的异常值为保留数据的完整性不予删除。

对于测量乙肝表面抗原、乙肝表面抗体、乙肝 e 抗原、乙肝 e 抗体、乙肝核心抗体指标的 id 当作潜在的乙肝患者处理，单独考虑。

5.4.2 预测血糖值

在问题二中，经比较，BP 神经网络的预测效果最好，因此本文采用 BP 神经网络预测模型来对附件 2 中的血糖值，预测结果见附录。

5.4.3 风险评估

本文利用问题三中的基于 K-means 聚类算法的风险评估模型来评估附件 2 的糖尿病风险，评估出的风险等级结果可参照附录及下图。

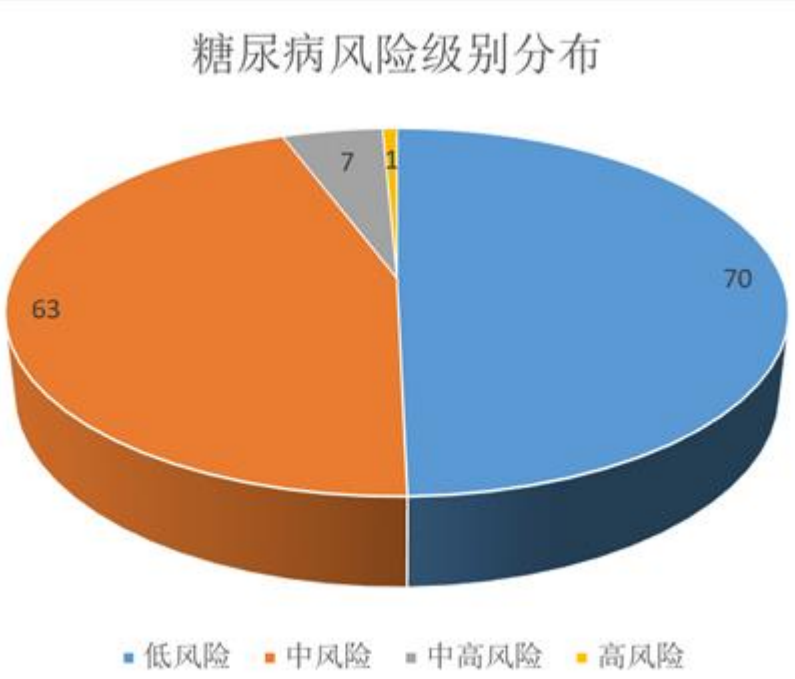

图 11 糖尿病风险级别分布

六、 模型的优缺点评价与改进

6.1 模型的优缺点

在本次问题二中，我们采用了三种训练方式：多元回归模型、支持向量机（SVM）回归模型和神经网络模型，问题三采用了以下是对它们的优缺点进行分析：

1.多元回归模型的优缺点：

优点：

（1）简单而直观：回归模型易于理解和实现，适用于解决线性关系问题。

（2）可解释性强：回归模型的参数具有明确的物理或经济含义，可以解释变量之间的关系。

（3）计算效率高：回归模型的训练和预测速度较快，适用于大规模数据集。

缺点：对非线性关系拟合能力较弱，对异常值和噪声敏感，异常值和噪声可能对模型的性能产生较大影响。

2.支持向量机（SVM）回归模型的优缺点：

优点：

（1）拟合能力强：SVM 回归模型在处理非线性问题时具有较强的拟合能力，可以通过使用不同的核函数来适应不同的数据形状。

（2）鲁棒性好：SVM 回归模型对于异常值和噪声的影响相对较小，具有较好的鲁棒性。

（3）可控制过拟合：通过调节正则化参数和核函数的参数，可以有效控制模型的复杂度，避免过拟合。

缺点：参数选择较困难：SVM 回归模型需要选择适当的核函数和参数，对于大规模数据集和复杂问题，参数调优较为困难；计算复杂度高，耗时较长；不适用于处理大规模数据集。

3.神经网络模型的优缺点：

优点：

（1）高度灵活：神经网络模型可以通过调整网络结构和激活函数等参数来适应不同的问题和数据特征。

（2）非线性拟合能力强：神经网络模型能够拟合复杂的非线性关系，适用于处理高度非线性的问题。

（3）并行计算能力：神经网络模型可以进行并行计算，适用于处理大规模数据集和复杂的计算任务。

缺点：需要大量的数据和计算资源；参数调优较困难；可解释性较差。

4.K-means 算法的优缺点：

优点：

（1）简单而高效，易于实现和理解。

（2）可扩展性，适用于大规模数据集，具有良好的可扩展性。

（3）结果可解释性：K-means 算法得到的聚类结果直观可解释，每个样本都会被分配到一个簇中。

缺点：需要预先指定簇的数量，对初始簇心敏感，K-means 算法对初始簇心的选择非常敏感，初始质心的选择不当可能导致得到不理想的聚类结果；对数据分布要求

较高，K-means 算法假设簇的形状是球形的，对于非球形的簇分布效果不佳。

6.2 模型的改进

在本次建模中，可以考虑以下模型改进的方向：

1.特征工程改进：对于问题二中的预测模型，可以进一步改进特征工程的方法。可以尝试使用更复杂的特征变换方法，如多项式特征、交互特征、特征选择等，以提取更丰富和有用的特征信息。

2.模型调参优化：对于问题二中的不同预测模型，可以通过调参优化来提升模型性能。使用交叉验证和网格搜索等技术，针对不同模型的参数进行优化，以找到最佳的参数组合，提高模型的泛化能力和预测准确性。

3.集成模型方法：考虑使用集成模型方法，如随机森林、梯度提升树等，将多个基础模型组合起来进行预测。通过集成模型的方式，可以利用不同模型的优势，提高整体的预测性能和稳定性。

4.异常值处理：针对问题一中的异常值，可以采用更加有效的异常值处理方法，如基于统计学方法或基于模型的方法，以减少异常值对建模结果的影响。

5.更多建模方法尝试：除了问题二中使用的线性回归、支持向量机和神经网络模型，可以尝试其他的建模方法，如决策树、贝叶斯分类器、K 近邻算法等。不同的建模方法可能具有不同的优势和适用性，可以通过尝试不同的方法来找到更合适的模型。

七、 参考文献

[1]李常春,常克昊,刘禹杉.几种异常值判别准则在气象计量数据处理中的应用[J].电子测量技术,2020,43(23):68-72.DOI:10.19651/j.cnki.emt.2004761.

[2]张波,宋国君.大规模空气质量监测数据缺失处理方法实证研究[J].中国环境科学,2022,42(05):2078-2087.DOI:10.19674/j.cnki.issn1000-6923.20220302.001.

[3]冯丽红. 调查数据缺失值常用插补方法比较的实证分析[D].河北经贸大学,2014.

[4]黄凤娟,付大愚,王金茹.社会体育调查中缺失数据处理方法的比较研究[J].沈阳体育学院学报,2014,33(04):46-49.

[5]郭灶耿. 同型半胱氨酸水平与脑卒中患病风险关联性研究[D].广州医科大学,2021.DOI:10.27043/d.cnki.ggzyc.2021.000247.）

[6]李柚洁,赵顺昱,杨萍等.基于数据分解的大气污染物短期预测组合方法综述[J].环境工程,2023,41(04):213-224.DOI:10.13205/j.hjgc.202304029.

[7]刘书含,孙文强,石晓星等.基于 BP 神经网络的热风炉群煤气消耗量预测[J].中国冶金,2022,32(02):77-83.DOI:10.13228/j.boyuan.issn1006-9356.20210465.

[8]陈玉涛. 基于深度学习的空气质量预测研究[D].青海师范大学,2021.DOI:10.27778/d.cnki.gqhzy.2021.000296.

[9]莫小琴.基于最小二乘法的线性与非线性拟合[J].无线互联科技,2019,16(04):128-129.

[10]杨俊闯,赵超.K-Means 聚类算法研究综述[J].计算机工程与应用,2019,55(23):7-14+63.

八、 附录

附录 A 支撑材料清单

- 图表

-附录 B 附件 2 预测结果和风险评估数据

- 代码

-附录 C 问题一 Matlab 源代码
-附录 D 问题二 Matlab 源代码
-附录 E 问题三 Matlab 源代码
-附录 F 问题四 Matlab 源代码

附录 B 附件 2 预测结果和风险评估数据

id	血糖预测	风险评估	感染情况	id	血糖预测	风险	
6001	5.017350168	低风险	未感染	6002	12.41063249	中高风险	感染
6003	5.312681567	低风险	感染	6004	6.674874081	中风险	感染
6005	5.973547322	中风险	感染	6006	4.188621854	低风险	感染
6007	5.147336596	低风险	未感染	6008	5.31203036	低风险	感染
6009	6.08414487	中风险	未感染	6010	5.564858735	低风险	未感染
6011	5.107637878	低风险	未感染	6012	5.726702635	中风险	感染
6013	4.627018574	低风险	感染	6014	6.776443872	中风险	未感染
6015	6.273787038	中风险	感染	6016	5.818328352	中风险	未感染
6017	3.210579798	低风险	未感染	6018	7.009900175	中风险	未感染
6019	5.692633476	中风险	未感染	6020	5.350387776	低风险	未感染
6021	5.116020201	低风险	感染	6022	7.363611368	中风险	未感染
6023	5.559268446	低风险	未感染	6025	5.706802889	中风险	未感染
6026	6.081971015	中风险	未感染	6027	5.42123942	低风险	未感染
6028	5.88064928	中风险	未感染	6030	6.245936232	中风险	未感染
6031	7.075355891	中风险	未感染	6032	8.456239635	中高风险	未感染
6033	6.753350967	中风险	未感染	6034	5.045269868	低风险	未感染
6035	5.250043847	低风险	感染	6036	5.655379484	中风险	未感染
6037	5.141135254	低风险	未感染	6038	5.613615978	中风险	未感染
6039	5.756863792	中风险	未感染	6040	6.028288553	中风险	未感染
6041	6.819812359	中风险	未感染	6042	6.727117671	中风险	未感染
6044	6.532504827	中风险	未感染	6045	5.060753075	低风险	感染
6046	5.983512752	中风险	未感染	6047	5.391800323	低风险	未感染
6048	5.959125781	中风险	未感染	6049	4.128119727	低风险	未感染
6050	5.303211806	低风险	未感染	6051	6.02762451	中风险	未感染
6052	14.42650407	高风险	未感染	6053	5.175714491	低风险	感染
6054	5.588756731	低风险	未感染	6055	5.267231553	低风险	感染
6056	4.965538432	低风险	感染	6057	6.909078742	中风险	未感染
6058	3.319833802	低风险	未感染	6059	7.628513123	中风险	未感染
6060	4.927913057	低风险	未感染	6061	6.481574854	中风险	未感染
6062	5.330385728	低风险	未感染	6063	5.69810437	中风险	未感染
6064	5.596535324	低风险	未感染	6065	5.775736603	中风险	感染
6066	5.89013878	中风险	未感染	6067	7.670965765	中风险	未感染
6068	3.656458911	低风险	未感染	6069	7.608453717	中风险	未感染
6070	4.610179997	低风险	未感染	6072	8.09314073	中高风险	未感染
6073	5.066438566	低风险	未感染	6074	6.45035583	中风险	感染
6075	5.831728025	中风险	感染	6077	5.111070482	低风险	未感染

id	血糖预测	风险评估	感染情况	id	血糖预测	风险
6078	5.300794348	低风险	未感染	6079	4.063253071	低风险	感染
6080	5.516329535	低风险	感染	6081	5.23414336	低风险	感染
6083	6.902726816	中风险	未感染	6085	5.906589569	中风险	未感染
6086	5.799706485	中风险	未感染	6087	0.819813234	低风险	未感染
6088	5.289201766	低风险	感染	6089	6.14931166	中风险	感染
6090	5.187019635	低风险	感染	6091	5.189146372	低风险	未感染
6092	4.718238067	低风险	未感染	6093	5.313611241	低风险	未感染
6094	5.655458725	中风险	感染	6095	5.15640469	低风险	未感染
6096	4.405132723	低风险	未感染	6097	10.07388386	中高风险	未感染
6098	5.934745593	中风险	未感染	6099	5.186272415	低风险	未感染
6100	6.086520441	中风险	未感染	6101	5.33068723	低风险	未感染
6103	5.202086822	低风险	未感染	6104	5.004034016	低风险	未感染
6105	6.264106957	中风险	感染	6106	4.698798983	低风险	未感染
6107	7.693127312	中风险	未感染	6108	4.618476665	低风险	感染
6109	6.34387463	中风险	未感染	6110	6.420896389	中风险	未感染
6111	6.597807753	中风险	未感染	6112	6.653710652	中风险	未感染
6113	4.616476796	低风险	未感染	6114	6.877846747	中风险	未感染
6115	5.056619858	低风险	未感染	6116	7.871231326	中高风险	未感染
6117	5.915617221	中风险	未感染	6118	7.118281411	中风险	未感染
6119	5.168341889	低风险	未感染	6120	7.205566373	中风险	未感染
6121	4.968989031	低风险	未感染	6122	5.869187349	中风险	未感染
6123	6.219459119	中风险	未感染	6124	5.361348901	低风险	未感染
6125	6.850850012	中风险	感染	6126	5.234285575	低风险	感染
6127	5.398163121	低风险	感染	6128	4.778035642	低风险	感染
6129	2.589753561	低风险	未感染	6130	5.363776312	低风险	未感染
6131	5.4293803	低风险	未感染	6132	5.248672825	低风险	未感染
6133	6.792905309	中风险	未感染	6134	4.960015241	低风险	未感染
6135	4.896815593	低风险	未感染	6136	5.797073147	中风险	未感染
6137	6.862991142	中风险	感染	6138	4.126483216	低风险	未感染
6140	4.686845486	低风险	未感染	6141	9.731149307	中高风险	感染
6142	6.241800756	中风险	未感染	6143	6.034056329	中风险	感染
6144	3.279936549	低风险	未感染	6145	7.613919555	中风险	未感染
6146	4.903625635	低风险	未感染	6147	5.003629004	低风险	未感染
6148	5.887433797	中风险	感染	6149	5.326124929	低风险	未感染
6150	9.336448586	中高风险	未感染				

附录 C 问题一：Matlab 源代码

```
clear,clc
load("data.mat");
% 数据预处理：处理缺失值和异常值
data = preprocessData(data);

% 计算变量之间的相关系数矩阵
corrMatrix = corr(data{:,:});
% 选择与血糖值相关性较高的变量
targetVariable = data{:, '血糖'};
correlationThreshold = 0.13; % 相关系数阈值，可根据实际情况调整
% 找到与血糖值相关性超过阈值的变量
highCorrIndices = find(abs(corrMatrix(:, 1)) > correlationThreshold);
selectedVariables = data.Properties.VariableNames(highCorrIndices);

% 打印筛选出的变量
disp('筛选出的主要变量指标：');
disp(selectedVariables(2:end));

% 可视化分析：绘制血糖值与每个主要变量之间的散点图
figure;
numVars = length(selectedVariables);
for i = 1:numVars
subplot(numVars, 1, i);
scatter(data{:, selectedVariables{i}}, targetVariable,'blue','.');
xlabel(selectedVariables{i});
ylabel('血糖值');
end

% 可视化分析：绘制血糖值的直方图
figure;
histogram(targetVariable);
xlabel('血糖值');
ylabel('频数');
title('血糖值分布');

% 数据清洗
function data = preprocessData(data)
% 1.处理缺失值：使用邻近填充
data{:, 2:end} = fillmissing(data{:, 2:end}, 'nearest');


% % 2.处理缺失值：使用均值填充
```

```
    % meanValues = mean(data{:, 2:end}, 'omitnan');
    % data{:, 2:end} = fillmissing(data{:, 2:end}, 'constant', meanValues);

    % % 3.处理缺失值：使用随机填充
    % %  获取数据集的维度
    % [numRows, numCols] = size(data);
    % %  遍历每一列
    % for col = 1:numCols
    % %  获取当前列的缺失值索引
    % missingIdx = isnan(data{:, col});
    % %  如果存在缺失值
    % if any(missingIdx)
    % hotDeckValues = data{~missingIdx, col}; %  从非缺失值中抽取值
    % data{missingIdx, col} = hotDeckValues(randi(sum(~missingIdx), sum(missingIdx), 1)); %  随机填充缺失值
    % end
    % end

    %  处理异常值：3sigma 原则
    features = data{:, [2:end]};
    meanValues = mean(features);
    stdValues = std(features);
    lowerBound = meanValues - 3 * stdValues;
    upperBound = meanValues + 3 * stdValues;
    %  检测异常值并进行处理
    for i = 1:size(features, 2)
    feature = features(:, i);
    outliers = feature < lowerBound(i) | feature > upperBound(i);
    %  将异常值替换为均值
    features(outliers, i) = meanValues(i);
    end
    processedData = [data{:, 1}, features];
    data{:,:}=processedData;
    end
```

附录 D 问题二：Matlab 源代码

```
1.神经网络模型
clear,clc
load("data.mat");
% 数据预处理：处理缺失值和异常值
data = preprocessData(data);

% 将数据拆分为训练集和测试集
trainRatio = 0.7; % 训练集比例
validationRatio = 0.15; % 验证集比例
testRatio = 0.15; % 测试集比例

% 随机打乱数据集
data = data(randperm(size(data, 1)), :);
% 划分数据集
trainSamples = data(1:round(trainRatio*size(data, 1)), :);
validationSamples = data(round(trainRatio*size(data,
1))+1:round((trainRatio+validationRatio)*size(data, 1)), :);
testSamples = data(round((trainRatio+validationRatio)*size(data, 1))+1:end, :);

% 提取输入特征和目标变量
trainFeatures = trainSamples{:, 2:end};
trainTarget = trainSamples{:, '血糖'};

validationFeatures = validationSamples{:, 2:end};
validationTarget = validationSamples{:, '血糖'};

testFeatures = testSamples{:, 2:end};
testTarget = testSamples{:, '血糖'};

% 构建神经网络模型
hiddenUnits = 10; % 隐藏层神经元数目

model = fitnet(hiddenUnits);
model = train(model, trainFeatures', trainTarget');

% 在验证集上评估模型性能
validationPredictions = model(validationFeatures');
mseValidation = mse(validationPredictions - validationTarget');

disp(['在验证集上的均方误差（MSE）：', num2str(mseValidation)]);
```

```
    % 在测试集上进行预测
    testPredictions = model(testFeatures');
    mseTest = mse(testPredictions - testTarget');
    disp(['在测试集上的均方误差（MSE）：', num2str(mseTest)]);
    % 可视化分析：绘制实际值与预测值的散点图
    figure;
    scatter(testTarget, testPredictions,'blue','.');
    xlabel('实际值');
    ylabel('预测值');
    title('实际值与预测值的散点图');

    % 保存模型
    save('BSmodel.mat', 'model');
```

2.向量机回归模型

```
clear,clc
    load("data.mat");
    % 数据预处理：处理缺失值和异常值
    data = preprocessData(data);

    % 将数据拆分为训练集和测试集
    trainRatio = 0.8; % 训练集比例
    validationRatio = 0.1; % 验证集比例
    testRatio = 0.1; % 测试集比例

    % 随机打乱数据集
    data = data(randperm(size(data, 1)), :);

    % 划分数据集
    trainSamples = data(1:round(trainRatio*size(data, 1)), :);
    validationSamples = data(round(trainRatio*size(data,
1))+1:round((trainRatio+validationRatio)*size(data, 1)), :);
    testSamples = data(round((trainRatio+validationRatio)*size(data, 1))+1:end, :);

    % 提取输入特征和目标变量
    trainFeatures = trainSamples{:, 2:end};
    trainTarget = trainSamples{:, '血糖'};

    validationFeatures = validationSamples{:, 2:end};
    validationTarget = validationSamples{:, '血糖'};

    testFeatures = testSamples{:, 2:end};
    testTarget = testSamples{:, '血糖'};
```

```
% 构建支持向量机回归模型
svrModel = fitrsvm(trainFeatures, trainTarget, 'Standardize', true);

% 在验证集上评估模型性能
validationPredictions = predict(svrModel, validationFeatures);
mseValidation = mse(validationPredictions - validationTarget);

disp(['在验证集上的均方误差（MSE）：', num2str(mseValidation)]);

% 在测试集上进行预测
testPredictions = predict(svrModel, testFeatures);
mseTest = mse(testPredictions - testTarget);
r2 = corr(testTarget, testPredictions)^2; % 决定系数
disp(['在测试集上的均方误差（MSE）：', num2str(mseTest)]);
disp(['决定系数 (R^2): ' num2str(r2)]);
% 可视化分析：绘制实际值与预测值的散点图
figure;
scatter(testTarget, testPredictions,'green','.');
xlabel('实际值');
ylabel('预测值');
title('实际值与预测值的散点图');
```

3.线性回归模型

```
clear,clc
load("data.mat");
% 数据预处理：处理缺失值和异常值
data = preprocessData(data);

% 分割数据集为训练集和测试集
rng(39); % 设置随机种子，以确保可重现性
trainRatio = 0.7; % 训练集比例
[trainData, testData] = splitData(data, trainRatio);

% 建立血糖值的预测模型
X_train = trainData{:, 2:end}; % 特征变量
y_train = trainData{:, '血糖'}; % 目标变量

model = fitlm(X_train, y_train); % 线性回归模型

% 在测试集上进行预测
X_test = testData{:, 2:end};
y_test_actual = testData{:, '血糖'};
```

```
y_test_predicted = predict(model, X_test);

% 评估预测结果
mse = mean((y_test_actual - y_test_predicted).^2); % 均方误差
rmse = sqrt(mse); % 均方根误差
r2 = corr(y_test_actual, y_test_predicted)^2; % 决定系数

% 打印评估结果
disp('预测结果评估：');
disp(['均方根误差 (RMSE): ' num2str(rmse)]);
disp(['决定系数 (R^2): ' num2str(r2)]);

% 可视化分析：绘制实际值与预测值的散点图
figure;
scatter(y_test_actual, y_test_predicted,'red','.');
xlabel('实际值');
ylabel('预测值');
title('实际值与预测值的散点图');
```

附录 E 问题三：Matlab 代码

```
clear,clc
load("data.mat");

% 数据预处理：处理缺失值和异常值
data = preprocessData(data);

% 选择特征和目标变量
target = data{:,'血糖'};

% 使用肘部法确定 k 值
k_values = 1:10; % 需要尝试的 k 值范围
inertias = zeros(1, length(k_values));

for i = 1:length(k_values)
k = k_values(i);
[~, ~, sumd] = kmeans(target, k);
inertias(i) = sum(sumd);
end

% 绘制肘部法曲线
```

```
figure;
plot(k_values, inertias, 'bo-');
xlabel('Number of Clusters (k)');
ylabel('Sum of Squares Within Cluster (Inertia)');
title('Elbow Method');
grid on;

% 根据肘部法选择最佳的 k 值
best_k = input('请输入最佳的 k 值：');

% 使用最佳的 k 值进行聚类
[idx, centroids] = kmeans(target, best_k);

% 显示分级结果
for i = 1:best_k
disp(['第', num2str(i), '级：']);
clusterIndices = find(idx == i);
clusterTargets = target(clusterIndices);
disp(['样本数量：', num2str(length(clusterTargets))]);
disp(['最小血糖值：', num2str(min(clusterTargets))]);
disp(['最大血糖值：', num2str(max(clusterTargets))]);
disp('----------');
end

% 统计每个聚类的样本数量
clusterCounts = histcounts(idx, 1:best_k+1);

% 显示分级结果和柱状图
figure;
bar(clusterCounts);
xlabel('糖尿病风险级别');
ylabel('样本数量');
title('糖尿病风险级别分布');
xticks(1:best_k);
xticklabels(cellstr(num2str((1:best_k)')));

% 计算每个聚类的样本数量比例
clusterProportions = clusterCounts / sum(clusterCounts);

% 显示分级结果和饼状图
figure;
labels = cellstr(num2str((1:best_k)'));
pie(clusterProportions, labels);
title('糖尿病风险级别分布');
```

```
% 绘制聚类结果的散点图
figure;
scatter(1:length(target), target, 20, idx, 'filled');
hold on;
scatter(1:length(centroids), centroids, 100, 'k', 'filled');
colormap(parula(best_k));
colorbar;
xlabel('样本序号');
ylabel('血糖值');
title('血糖分级结果');
legend('样本', '聚类中心');
```

附录 F 问题四：Matlab 代码

```
clear,clc;
% 加载附件 1 的数据
data = readtable('data\附件.xls', 'VariableNamingRule', 'preserve');
% 数据预处理：处理缺失值和异常值
data=preprocessData(data);

% 将数据拆分为训练集和测试集
trainRatio = 0.7; % 训练集比例
validationRatio = 0.15; % 验证集比例
testRatio = 0.15; % 测试集比例
% 随机打乱数据集
data = data(randperm(size(data, 1)), :);
% 划分数据集
trainSamples = data(1:round(trainRatio*size(data, 1)), :);
validationSamples = data(round(trainRatio*size(data,
1))+1:round((trainRatio+validationRatio)*size(data, 1)), :);
testSamples = data(round((trainRatio+validationRatio)*size(data, 1))+1:end, :);

% 提取输入特征和目标变量
trainFeatures = trainSamples{:, 2:end};
trainTarget = trainSamples{:, '血糖'};

validationFeatures = validationSamples{:, 2:end};
validationTarget = validationSamples{:, '血糖'};

testFeatures = testSamples{:, 2:end};
testTarget = testSamples{:, '血糖'};

% 构建神经网络模型
```

```
hiddenUnits = 100; % 隐藏层神经元数目

model = fitnet(hiddenUnits);
model = train(model, trainFeatures', trainTarget');

% 在验证集上评估模型性能
validationPredictions = model(validationFeatures');
mseValidation = mse(validationPredictions - validationTarget');

disp(['在验证集上的均方误差（MSE）：', num2str(mseValidation)]);

% 在测试集上进行预测
testPredictions = model(testFeatures');
mseTest = mse(testPredictions - testTarget');
disp(['在测试集上的均方误差（MSE）：', num2str(mseTest)]);

% 可视化分析：绘制实际值与预测值的散点图
figure;
scatter(testTarget, testPredictions);
xlabel('实际值');
ylabel('预测值');
title('实际值与预测值的散点图');

% 保存模型
save('BSmodel.mat', 'model');

% 读取无血糖值的检测数据
data2 = readtable('data\附件 2.xls', 'VariableNamingRule', 'preserve');
data3= readtable('data\附件 2：无血糖值的检测数据.csv', 'VariableNamingRule', 'preserve');
dataHB=data3{:,'乙肝表面抗原'};
% 数据预处理：处理缺失值和异常值
data2 = preprocessData(data2);
% 提取特征
testFeatures2 = data2{:, 2:end};
% 使用已有模型进行血糖预测
testPredictions2 = model(testFeatures2');

% 进行糖尿病风险评估
riskScores = string(size(testPredictions2));
riskScores(testPredictions2 >12.6 ) = '高风险'; % 高风险
riskScores(testPredictions2 > 7.8 & testPredictions2 <= 12.6) = '中高风险'; % 中高风险
riskScores(testPredictions2 > 5.6 & testPredictions2 <= 7.8) = '中风险'; % 中风险
riskScores(testPredictions2 <= 5.6) = '低风险'; % 低风险
```

```
riskHB = string(size(testPredictions2));
riskHB(isnan(dataHB))='未感染';
riskHB(~isnan(dataHB))='感染';

% 将预测结果和风险评估添加到数据表中
data2{:,"血糖预测"} = testPredictions2';
data2{:,"风险评估"}= riskScores';
data2{:,"乙肝病毒感染情况"}= riskHB';

% 保存结果到新的 Excel 文件
writetable(data2, 'data\附件 2_结果.xls', 'Sheet', 1);

disp('糖尿病预测和风险评估已完成，并将结果保存至附件 2_结果.xls 文件。');
```