

实验五 甲状腺结节超声图像分类实验

一、 实验目的

- 利用机器算法对甲状腺结节超声图像进行分类，通过提取图像特征并训练机器学习模型，实现对甲状腺结节良恶性的自动判别，为临床诊断提供辅助决策依据，提高诊断效率和准确性。

二、 实验设备与环境

- 甲状腺结节超声图像数据集：教学云平台附件下载。
- 计算机设备：计算机一台，可支持图像数据处理和模型训练。
- 编程软件与库（以 python 为例）：
 - Python 编程语言及其相关科学计算库（如 NumPy、Pandas）用于数据处理和分析。
 - OpenCV 库用于图像读取、预处理和基本特征提取操作。
 - Scikit-learn 机器学习库用于构建模型并进行模型评估。

三、 实验内容与步骤

1. 图像预处理（10 分）

- 1) 图像读取与灰度转换：使用 OpenCV 库读取超声图像数据，并将彩色图像转换为灰度图像，降低数据维度，简化后续处理。

```
# 检查图像的通道数
if len(image.shape) == 2: # 图像是灰度图
    gray_image = image
else: # 图像是彩色图
    gray_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
```

- 2) 图像裁剪与归一化：根据甲状腺结节在图像中的位置，裁剪出包含结节的感兴趣区域（ROI），去除图像中无关的背景信息。然后对裁剪后的图像进行归一化处理，使图像像素值在特定范围内（例如 0 - 1），以提高模型训练的稳定性和收敛速度

```
resized_image = cv2.resize(original_image, (new_width, new_height))
right_pixels_to_remove = 30
left_pixels_to_remove = 28
bottom_pixels_to_remove = 23
```

- 3) 图像尺寸调整：将所有图像的尺寸调整为统一大小（如 128x128 像素），以便于特征提取和模型输入。

```
new_width = 256
new_height = 256
```

- 4) 数据增强（可选）：为了增加数据集的多样性，可采用图像翻转、旋转、平移等数据增强技术，对原始图像进行变换并生成新的图像样本。但需注意保持图像与对应病理标签的一致性，确保数据增强后的图像仍然能够准确反映甲状腺结节的真实特征。

✧ 在该实验中，我采用了中值滤波处理图像去除椒盐噪声，减少图像处理中可能会产生的伪影；且相比于模糊图像可能导致边缘部分细节丢失的均值滤波，中值滤波在去噪时能更好地保留图像的边缘特征。

```
# 裁剪图像，去除左右和底部的部分像素
cropped_image = resized_image[:, left_pixels_to_remove:-
right_pixels_to_remove, :]
cropped_image = cropped_image[:-bottom_pixels_to_remove, :]
# 对裁剪后的图像进行中值滤波
median_filtered_image = cv2.medianBlur(cropped_image, 3)
return median_filtered_image
```

2. 特征提取（20 分）

- 1) 灰度特征：在该实验中，使用 `np.mean(gray_image)` 计算图像的灰度均值；使用 `np.var(gray_image)` 计算图像的灰度方差，反映图像灰度值的分散程度；使用 `np.histogram` 计算灰度直方图，并归一化以便于比较。并在最后返回一个包含灰度均值、方差、LBP 直方图和灰度直方图的字典，这些特征可以反映结节内部回声的整体分布情况和均匀性。

```
def calculate_lbp_features(image):
# 检查图像的通道数
    if len(image.shape) == 2: # 图像是灰度图
        gray_image = image
    else: # 图像是彩色图
        gray_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    lbp_image = feature.local_binary_pattern(gray_image, P=8, R=1,
method='uniform')
    hist, _ = np.histogram(lbp_image.ravel(), bins=np.arange(0, 10),
range=(0, 10))
    hist = hist.astype("float")
    hist /= (hist.sum() + 1e-7)
    return hist
```

- 2) 纹理特征：运用灰度共生矩阵（GLCM）提取纹理特征，如对比度、相关性、能量、熵。GLCM 描述了图像中不同灰度值像素对的空间分布关系，对于甲状腺结节的纹理分析具有重要意义。在本实验中，我通过在 0° 方向和距离为 1 上计算 GLCM，并对生成的 GLCM 归一化，有助于后续的特征计算。最后，函数返回四个特征值——对比度、相关性、能量和熵，这些特征能够描述图像的纹理特征，对后续的图像分析和处理非常有用。

```
def calculate_glcm_features(gray_image):
    glcm = feature.graycomatrix(gray_image, distances=[1], angles=[0],
levels=256, symmetric=True, normed=True)
    contrast = feature.graycoprops(glcm, 'contrast')[0][0]
```

```
correlation = feature.graycoprops(glcm, 'correlation')[0][0]
energy = feature.graycoprops(glcm, 'energy')[0][0]
entropy = -np.sum(glcm * np.log(glcm + 1e-7)) # Avoid log(0)
return contrast, correlation, energy, entropy
```

3) 形态特征：提取结节的形态学特征，包括周长、面积、圆形度、长轴短轴比等。这些特征有助于判断结节的形状规则性，恶性结节往往具有不规则的形态特征，与良性结节存在差异。

- ✧ 面积 (Area): 使用 `regions[0].area` 获取第一个区域的面积，表示该结节的大小。
- ✧ 周长 (Perimeter): 使用 `regions[0].perimeter` 获取周长，表示结节边界的长度。
- ✧ 圆形度 (Circularity): 通过公式 $\text{circularity} = (4 * \text{np.pi} * \text{area}) / (\text{perimeter} ** 2)$ 计算。圆形度越接近 1，表示越接近圆形；不规则的结节圆形度会较低。
- ✧ 长轴长度 (Major Axis Length): 使用 `regions[0].major_axis_length` 获取结节的长轴长度。
- ✧ 短轴长度 (Minor Axis Length): 使用 `regions[0].minor_axis_length` 获取结节的短轴长度。
- ✧ 长短轴比 (Aspect Ratio): 通过 $\text{aspect_ratio} = \text{major_axis_length} / \text{minor_axis_length}$ 计算。如果短轴长度为 0，返回 0，避免除零错误。

```
labeled_image = measure.label(gray_image > 0.5)
regions = measure.regionprops(labeled_image)

if regions:
    area = regions[0].area
    perimeter = regions[0].perimeter
    circularity = (4 * np.pi * area) / (perimeter ** 2) if perimeter >
0 else 0
    major_axis_length = regions[0].major_axis_length
    minor_axis_length = regions[0].minor_axis_length
    aspect_ratio = major_axis_length / minor_axis_length if
minor_axis_length > 0 else 0
else:
    area = perimeter = circularity = major_axis_length =
minor_axis_length = aspect_ratio = 0
```

3. 特征选择与降维（10 分）

如果提取的特征可能存在冗余或相关性较高的情况，为了提高模型训练效率和避免过拟合，可以采用特征选择或降维算法对特征进行处理。例如：

1) 主成分分析 (PCA)：PCA 是一种常用的降维算法，它通过线性变换将原始特征投影到新的低维空间，使得投影后的数据方差最大，从而保留主要的特征信息。本实验保留了 10 个主成分，将高维特征向量转换为低维表示。

```
pca = PCA(n_components=10)
X_train_pca = pca.fit_transform(X_train_balanced)
X_val_pca = pca.transform(X_val_scaled)
```

```
X_test_pca = pca.transform(X_test_scaled)
```

- 2) 递归特征消除 (RFE)：RFE 基于特征重要性评估逐步去除不重要的特征。使用 SVM 模型作为评估器，每次迭代，RFE 会根据 SVM 模型的评估结果，逐步去除不重要的特征，直到剩余 5 个特征。通过 RFE 可以选择出与甲状腺结节分类最相关的特征子集。

```
model = SVC()
rfe = RFE(model, n_features_to_select=5)
X_train_rfe = rfe.fit_transform(X_train_pca, y_train_balanced)
X_val_rfe = rfe.transform(X_val_pca)
X_test_rfe = rfe.transform(X_test_pca)
```

4. 机器学习模型训练与优化 (SVM 分类器) (30 分)

- 1) 数据划分：将经过预处理和特征提取后的数据划分为训练集、验证集和测试集。通常按照 70%、15%、15% 的比例进行划分，确保训练集用于模型训练，验证集用于模型参数调整 and 性能评估，测试集用于最终模型性能的独立测试。

```
# 首先将数据划分为训练集 (70%) 和临时集 (30%)
X_temp, X_test, y_temp, y_test = train_test_split(X, y, test_size=0.3,
random_state=42)
# 然后从临时集中划分出验证集 (15%) 和训练集 (85%)，即从 30% 的临时集中划分出 50%
作为验证集
X_train, X_val, y_train, y_val = train_test_split(X_temp, y_temp,
test_size=0.5, random_state=42)
```

- 2) 机器学习模型初始化：创建 SVM 分类器对象，根据数据集二分类和高维的特点选择合适的 SVM 核函数为线性核，提高计算效率；并设置惩罚因子 C 为 1，它控制了对误分类样本的惩罚程度，C 值越大，模型对误分类的容忍度越低，但可能导致过拟合；C 值越小，模型越倾向于容忍误分类，但可能欠拟合；最后，使用 probability=True 启用概率估计，这允许 SVM 输出每个样本的分类概率，有助于在需要概率输出的任务中（如 ROC 曲线绘制或阈值调整）使用。

```
# 创建 SVM 分类器，启用概率估计
svm_classifier = SVC(kernel='linear', C=1.0, probability=True)
```

- 3) 模型训练：使用训练集数据对 SVM 模型进行训练，将提取的特征作为输入，对应的病理标签作为输出，通过 SMO 算法解决二次规划问题，SMO 在每一步迭代中选择两个拉格朗日乘子进行优化，保持其他乘子不变，从而逐步逼近最优解。在训练过程中，SMO 会不断调整 SVM 模型的参数，使得模型在训练集上的损失函数最小化，从而学习到甲状腺结节特征与良恶性之间的关系。

```
svm_classifier.fit(X_train_rfe, y_train_balanced)
```

- 4) 模型验证与参数调整：在训练过程中，使用验证集数据对模型进行评估，计算评估指标（如准确率、召回率、F1 值等）。根据验证集的评估结果，调整 SVM 模型的参数，以提高模型在验证集上的性能。重复训练和验证过程，直到模型性能不再提升或达到预设的迭代次数。

```
y_val_pred = svm_classifier.predict(X_val_rfe)
val_accuracy = accuracy_score(y_val, y_val_pred)
```

```
val_report = classification_report(y_val, y_val_pred, zero_division=1)
print("Validation Accuracy:", val_accuracy)
print("Validation Classification Report:\n", val_report)
```

5. 模型评估与可视化（30 分）

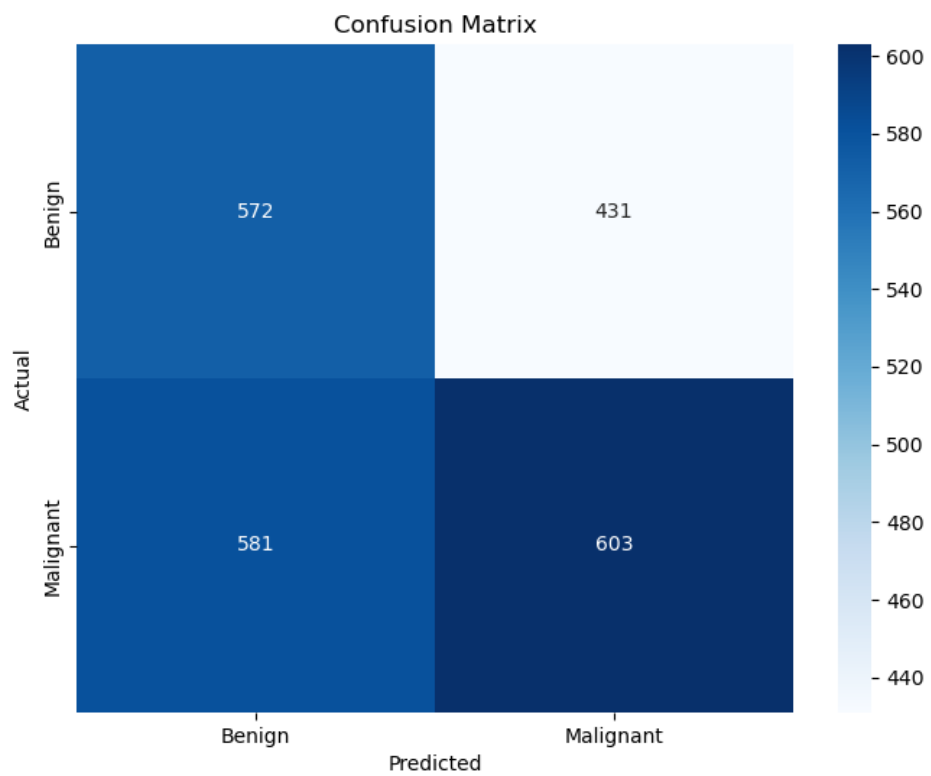
- 1) 性能指标计算：使用测试集数据对训练好的 SVM 模型进行最终评估，计算各种性能指标，包括准确率（Accuracy）、精确率（Precision）、召回率（Recall）、F1 值、受试者工作特征曲线（ROC 曲线）下面积（AUC）等。
 - 准确率：表示模型正确分类的样本数占总样本数的比例，反映了模型的整体分类准确性。
 - 精确率：针对预测为正类（如恶性结节）的样本，计算真正为正类的样本比例，衡量模型对正类预测准确性。
 - 召回率：表示实际为正类的样本中被模型正确预测为正类的比例，反映了模型对正类样本的捕捉能力。
 - F1 值：综合考虑精确率和召回率的调和平均值，用于平衡两者之间的关系，更全面地评估模型性能。
 - ROC 曲线：以假阳性率（FPR）为横坐标，真阳性率（TPR）为纵坐标绘制曲线，AUC 值表示 ROC 曲线下的面积，AUC 值越大，说明模型的分类性能越好，能够更好地区分正类和负类样本。
- 2) 混淆矩阵分析：绘制混淆矩阵，直观展示模型对甲状腺结节良恶性分类的结果分布情况。混淆矩阵的行表示实际类别，列表示预测类别，通过矩阵中的数值可以清晰地看到模型将良性结节误判为恶性结节以及将恶性结节误判为良性结节的数量，有助于分析模型在不同类别上的分类效果和错误类型。
- 3) 绘制 ROC 曲线和准确率—召回率曲线等可视化图形，直观展示模型在不同阈值下的性能表现。ROC 曲线可以帮助评估模型在不同假阳性率和真阳性率之间的权衡，而准确率—召回率曲线则更侧重于展示模型在不同召回率水平下的精确率变化情况，有助于根据实际需求选择合适的模型阈值。

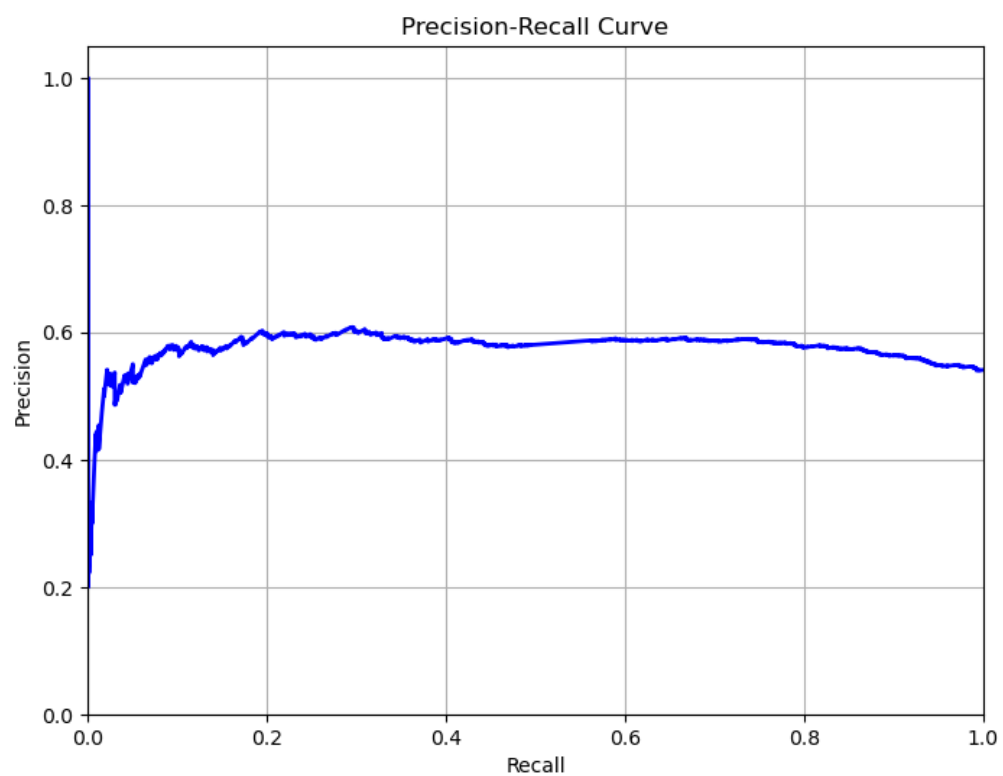
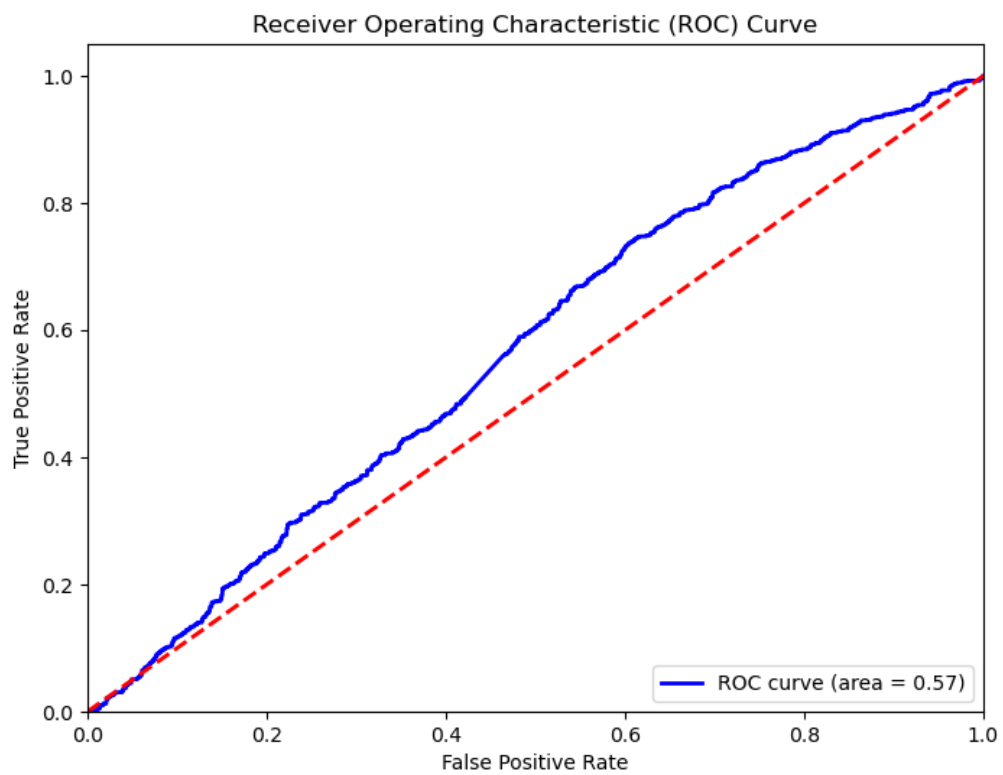
四、 源程序

见附件thyroid_classification.py。

五、 实验结果

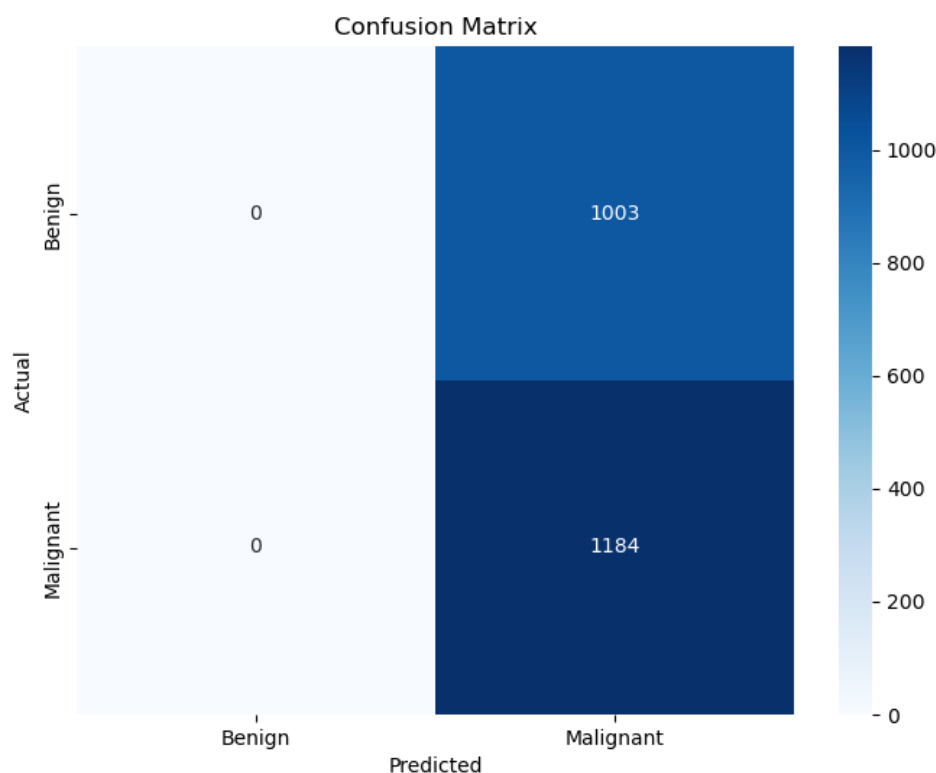
Validation Accuracy: 0.5503724029792239				
Validation Classification Report:				
	precision	recall	f1-score	support
0	0.50	0.60	0.55	1153
1	0.61	0.51	0.55	1398
accuracy			0.55	2551
macro avg	0.55	0.55	0.55	2551
weighted avg	0.56	0.55	0.55	2551
Test Accuracy: 0.5372656607224509				
Test Classification Report:				
	precision	recall	f1-score	support
0	0.50	0.57	0.53	1003
1	0.58	0.51	0.54	1184
accuracy			0.54	2187
macro avg	0.54	0.54	0.54	2187
weighted avg	0.54	0.54	0.54	2187
ROC AUC: 0.5679608135054297				





六、 实验总结和心得

在实验过程中，曾出现准确率等指标与混淆矩阵不符合的情况，如下：



分析认为是由于划分后数据集中良性类别的样本数量非常少，模型可能不会预测该类别，从而导致该类别的精确率计算为 0，后通过SMOTE（合成少数类过采样技术）来缓和数据不平衡。

尽管当前模型的性能尚未达到理想水平，但通过数据质量提升、特征选择优化和模型参数调整，未来有潜力提高其在甲状腺结节诊断中的应用价值。结合临床实际需求，机器学习模型可以成为支撑医生决策的重要工具。由前面实验结果分析可知：

- 1. 优势与不足
 - 1) 优势：
 - ✧ 可解释性：SVM 模型在许多情况下具有良好的可解释性，能够输出特征的重要性。
 - ✧ 适应性：可以通过选择不同的核函数来适应不同的数据分布。
 - 2) 不足：
 - ✧ 准确率较低：模型的准确率在 0.55 左右，远未达到理想水平，可能导致临床决策的不确定性。
 - ✧ 类别不平衡：0 类和 1 类的支持度不均衡，可能影响模型的表现，尤其是在召回率方面。
 - ✧ 特征选择不足：RFE 可能未能捕捉到所有相关特征，导致重要特征丢失。
- 2. 存在的问题及改进建议
 - 1) 数据质量：
 - ✧ 问题：数据可能存在噪声、缺失值或不一致性。

- ✧ 建议：进行数据清洗和预处理，将数据储存为csv格式，确保数据质量；应用数据增强技术以增加样本多样性。
- 2) 特征提取方法的有效性：
 - ✧ 问题：当前的特征提取方法可能未能充分利用所有有用的特征。
 - ✧ 建议：探索其他特征选择方法（如基于模型的特征选择、L1 正则化等），并考虑使用更多原始特征。
- 3) 模型选择与优化：
 - ✧ 问题：SVM 模型的参数选择可能不够优化。
 - ✧ 建议：使用网格搜索（Grid Search）来优化 SVM 的超参数，尝试不同的核函数、惩罚因子C和其他超参数。
- 3. 未来研究方向
 - 1) 模型集成：考虑使用集成学习方法（如随机森林、XGBoost）来提高模型的准确性。
 - 2) 深度学习：探索使用深度学习方法（如卷积神经网络）处理图像数据，可能对甲状腺结节的分类更有效。
 - 3) 多模态数据：结合其他类型的数据（如临床信息、影像学数据等），建立更全面的诊断模型。
- 4. 临床应用可行性与潜在价值
 - 1) 辅助诊断：机器学习模型能够提供快速的辅助诊断支持，帮助医生在复杂的病例中做出决策。
 - 2) 风险评估：模型可用于评估患者甲状腺结节的风险，帮助制定个性化的治疗方案。
 - 3) 提高效率：通过自动化分析，减轻医生的工作负担，提高诊断效率。