2024 美赛 C 题参考思路解析

题目

2024MCM Problem C: 网球运动中的动力

在 2023 年温布尔登网球公开赛男子组决赛中，20 岁的西班牙新星卡洛斯阿尔卡拉斯击败了 36 岁的诺瓦克-德约科维奇。这是德约科维奇自 2013 年以来首次在温布尔登输掉比赛，也结束了这位大满贯历史上最伟大球员之一的辉煌战绩。

比赛本身就是一场出色的战斗定要轻松获胜，德约科维奇似乎乎注定要轻松获胜因为他在第一盘以 6-1 的比分占据优势（7 局比赛中赢了 6 局）。第二盘比赛却十分紧张，最终阿尔卡拉兹在决胜盘中以 6 获胜。第三盘的情况与第一盘相反，阿尔卡拉法以 6-1 的比分轻松获胜。第四盘开始后，年轻的西班牙人似乎完全控制了局面，但不知何故，比赛的走势再次发生了变化，德约科维奇完全制了局面以 6-3 的比分赢得了这一盘。第五盘也是最后一盘比赛开始后德约科维奇延续了第四盘的优势，但比赛的走向再次发生了变化，阿尔卡拉斯取得了控制权，并以 6-4 的比分赢得了胜利。本场比赛的数据在提供的数据集中，"match_id"为“2023-wimbledon-1701"。您可以使用“set_no"列(等于 1)查看第一盘德约科维奇占优时的所有得分情况。 似乎占优的一方出现了令人难以置信的波动，有时是多分甚至是多局的波动，这通常归因于“势头"。

在字典中，"动量"的定义是“通过运动或一系列事件获得的力量或作用力。”[2] 在体育运动中，一支球队或一名球员可能会觉得他们在比赛中拥有动量或力量/作用力"，但很难衡量这种现象。此外，如果存在“势"的话，比赛中的各种事件是如何产生或改变“势“的，也不是一目了然的。

2023 年温布尔登男子比赛第三轮以后所有比赛每一分详细数据的数据集“Wimbledon_featured_matches.csv”，以及对数据集的描述文件"data_dictionary.csv"和帮助理解数据示例的"data_examples"部分。提供 2023 年温布尔登网球公开赛前两轮之后所有男子比赛中每一分的数据您可以自

行决定加入其他球员信息或其他数据，但必须完整记录数据来源。用这些数据：

大数据题　做题之前，一定要处理数据，

本次数据预处理手段，包括但不限与缺失值处理，填充，标准化

更多是通过数据挖掘建立一个新表，来符合我们本次题目要求

比如主成分分析、LDA 或者是手动处理，

有三个概念需要理解：赢一局，赢一盘，赢得比赛。[i]

局

一局内可能出现的分数为 0，15，30，40，结束，其实等同于 0，1，2，3，结束。

一方赢得一分，记为 15：0，以此类推。

40：40 成为平分（Deuce），此时一方得分称为占先。

一方需至少多赢两分才能赢下一局，此时局数加一。

盘

盘由局所构成，局数先到 6 的一方赢得此盘，如 6-0, 6-1, 6-2, 6-3, 6-4。

同样的，一方在局数上也必须领先两局才能赢下此盘，所以当双方比分为 5-5 时，一方需连赢两局。

若战至 6-6，则需要通过 Tie-break（抢七）的方式决出胜负。

抢七规则

- 先得七分的一方赢下该局及该盘（若 6-6，一方需净胜两分如 9-7）
- 先发球方发第一分，另一方发第二、三分，此后均为双方轮流各发两分
- 第一分在一区发，第二分在二区，第三分在一区，以此往后

赛

业余比赛多为一盘制，即赢得一盘者赢得比赛。

职业比赛有三盘两胜制及五盘三胜制，分别为先赢两盘及三盘者赢得比赛

(1) 建立一个模型，捕捉赛点发生时的比赛流程，并将其应用到一场或多比赛中。

您的模型应能确定哪位球员在比赛中的某个特定时间段表现更好，以及他们

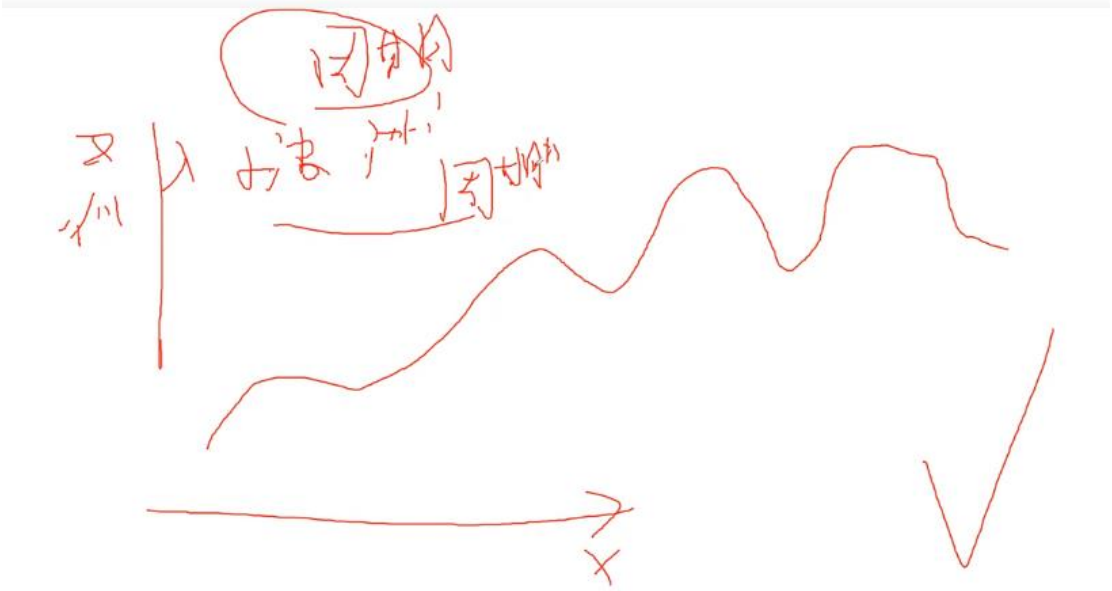
的表现好到什么程度。根据您的模型提供可视化的比赛流程描述。注意:在网球比赛中，发球的一方赢得赛点/比赛的概率要高得多。您可能希望以某种方式将这一因素考虑到您的模型中。

得分　先发球　　对局次数等等

time　　score1 先发　score2

80w 大　Lstm　gru

小波分析周期模型，



数模过程

假设我们有一个一维时间序列数据 $f(t)$，它代表网球比赛中的得分变化。我们想用连续小波变换来分析这个信号的时间-频率特性。

连续小波变换（CWT）的定义如下：

对于给定的小波函数 $\psi_a(t)$，其中 a 是尺度参数，小波函数在尺度 a 下与原信号 $f(t)$ 的卷积定义为：$W_f(a,b) = \int_{-\infty}^{+\infty} f(t)\frac{1}{|a|}\psi^*\left(\frac{t-b}{a}\right)dt$

这里，

- $W_f(a,b)$ 是小波系数，表示信号 $f(t)$ 在时间点 b 和尺度 a 下的局部特征。
- $\psi^*(t)$ 表示小波基函数 $\psi(t)$ 的复共轭。
- 尺度 a 决定了对信号细节探测的分辨率，较大的 a 对应于较低的频率或较慢的变化，较小的 a 则对应于较高的频率或较快的变化。
- 时间位置 b 指定在哪个时间点上执行该尺度下的分析。

在 scipy.signal.cwt 中，我们可以选择不同的小波基来进行变换，但要注意的是，在实际使用中 scipy.signal.cwt 并不直接支持Morlet小波，而是提供了一些其他类型的小波基函数。

例如，如果我们使用Ricker小波（也称为墨西哥帽小波），其数学表达式为： $\psi(t) = (1 - 2\pi^2 f_c^2 t^2)e^{-\pi^2 f_c^2 t^2}$ 其中 f_c 是中心频率。

然后，通过调用 cwt() 函数，我们会计算出一系列不同尺度下的小波系数矩阵，从而得到一个时间-尺度图像，它可以直观地显示信号在不同时间尺度下的能量分布情况。

因此，虽然您提到了错误信息 'str' object is not callable，但这不是关于数学建模过程的问题，而是编程实现时对象引用错误，意味着某个字符串被尝试当作函数调用。在正确设置和应用小波函数之后，上述数学建模过程将能顺利进行。

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.signal import cwt, ricker

# 随机生成模拟网球比赛的先发与后发得分数据
np.random.seed(0)  # 设置随机种子以确保结果可复现
num_points = 100  # 数据点数量
time = np.linspace(0, 99, num_points)  # 时间轴（例如可以代表比赛局数）
score1 = np.random.normal(loc=4, scale=2, size=num_points)  # 先发方得分
score2 = np.random.normal(loc=4, scale=2, size=num_points)  # 后发方得分

# 小波分析部分（此处我们仅对先发方得分做小波分析示例），使用 Morlet 小波
morlet = ricker  # Morlet 小波在 scipy.signal 中没有直接提供，这里用 ricker 近似
scales = np.arange(1, 31)  # 小波尺度范围，可以根据实际需求调整
# 对于真实的小波函数库如 pywt，您可以使用'morl'作为小波名
# 在 scipy.signal 中，我们需要自己实现或选择内置函数替代
cwt_result = cwt(score1, morlet, scales / 2)  # 注意尺度可能需要除以 2，取决于具体小波基的定义

# 绘制原始得分曲线和小波变换结果
```

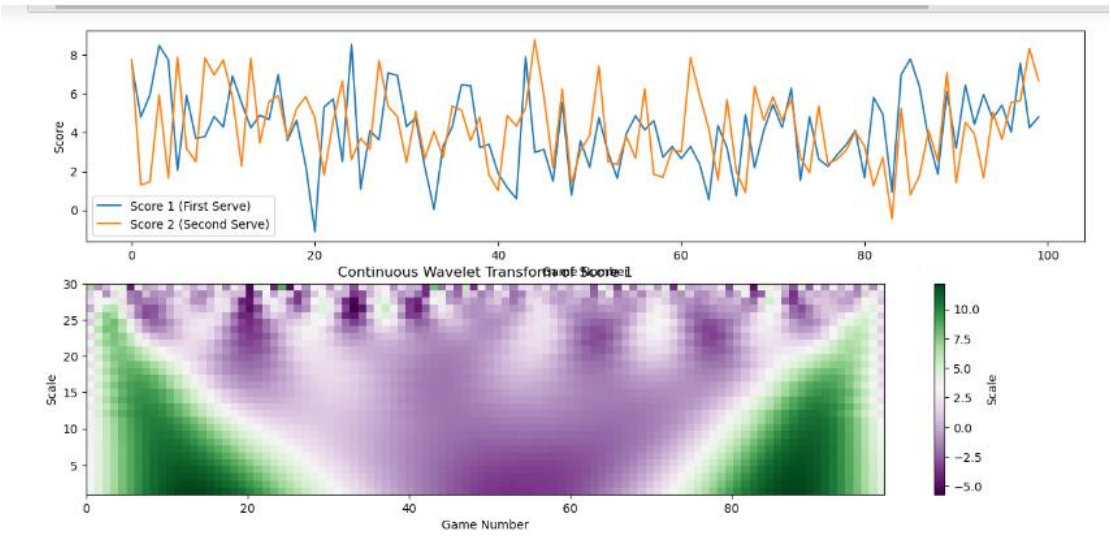
```
plt.figure(figsize=(15, 7))

plt.subplot(2, 1, 1)
plt.plot(time, score1, label='Score 1 (First Serve)')
plt.plot(time, score2, label='Score 2 (Second Serve)')
plt.legend()
plt.xlabel('Game Number')
plt.ylabel('Score')

plt.subplot(2, 1, 2)
plt.imshow(cwt_result, cmap='PRGn', aspect='auto', extent=[time[0],
time[-1], scales[0], scales[-1]])
plt.colorbar(label='Scale')
plt.xlabel('Game Number')
plt.ylabel('Scale')
plt.title('Continuous Wavelet Transform of Score 1')

plt.show()

# 如果你想进一步分析周期性特征，可以通过查找小波系数峰值等方式来确定潜在的周期模式。
```



(2)一位网球教练对“势头“在比赛中的作用持怀疑态度。相反，他假设比赛中的波动和一名球员的成功是随机的。请使用您的模型/度量来评估这一说法。

势头定义　　势头连胜

两个表

Times　player1　01020304　Times　player2　010304241 一局

或者

Times　player1　0　　Times　player2　0

1　　1

一盘的

Times　player1　01020304　Times　player2　010304241 一局

或者

Times　player1　0　　Times　player2　0

1　　1

赛（找更多数据）

使用预测+相关性分析,比如小波、arima(平稳性检验,ADF,P,AIC(P,Q)),GM(1,1)

数据量少+卡方检验、箱线图

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.signal import cwt, ricker
from scipy.stats import chi2_contingency

# 假设球员 1 和球员 2 每局比赛得分情况（这里仅模拟了前几局）
player1_scores = [0, 0, 1, 1, 0, 1, 1, 0]  # 球员 1 得分序列
player2_scores = [1, 1, 1, 0, 1, 0, 1, 1]  # 球员 2 得分序列

# 将数据转换为二维数组，以便于后续处理
scores_matrix = np.array([player1_scores, player2_scores])
```

```
# 对于小波分析，由于这个得分序列过于稀疏且离散，不适合直接应用连续小波变换，
# 通常我们会对连续变化的数据进行小波分析以获取时间-频率特性。
# 若需要可视化每个球员得分趋势的小波特征，可能需要构建连续的时间序列数据或考虑其他类型的分析方法。

# 不过，为了展示小波分析的基本过程，我们可以构造一个连续信号来模拟某个球员的得分趋势：
time_axis = np.linspace(0, len(player1_scores) - 1, len(player1_scores))  # 时间轴
player1_smoothed = np.convolve(player1_scores, np.ones(5)/5, mode='same')  # 对得分做平滑处理模拟连续趋势

# 使用 Ricker 小波进行小波变换（这里仅为示例，实际是否适用需根据具体问题判断）
scales = np.arange(1, 100)
# 定义 Ricker 小波的特征频率（这里假设为 1）
# 定义 Ricker 小波的特征频率（这里假设为 1）

wavelet_frequency = 1

# 根据 scales 生成对应的宽度值

widths = scales / wavelet_frequency  # 这里是基于特征频率计算宽度的一个简单示例，实际应用中可能需要更精确转换
```

```
# 使用 Ricker 小波进行小波变换

cwt_result = cwt(player1_smoothed, ricker, widths)
# 绘制小波系数图像
plt.figure(figsize=(10, 6))
plt.imshow(cwt_result, cmap='PRGn', aspect='auto', extent=[0,
len(player1_scores), max(scales), min(scales)])
plt.xlabel('Time')
plt.ylabel('Scale')
plt.title('Continuous Wavelet Transform of Player 1\'s Score Trend')
plt.colorbar(label='Wavelet Coefficients')
plt.show()

# 对于卡方检验，我们需要检查的是两个球员得分之间是否存在显著关联性
# 这里我们将统计在所有回合中两人得分模式的联合频数
contingency_table = np.array([[sum(scores_matrix[0] == 0),
sum(scores_matrix[0] == 1)],
                              [sum(scores_matrix[1] == 0),
sum(scores_matrix[1] == 1)]])

# 执行卡方检验
chi2_statistic, p_value, dof, expected =
chi2_contingency(contingency_table)

# 输出卡方检验结果
print(f"Chi-squared statistic: {chi2_statistic:.2f}")
print(f"P-value: {p_value:.4f}")
if p_value < 0.05:
    print("There is a significant correlation between the two players'
```

```
scoring patterns.")
else:
    print("There is no significant correlation between the two players'
scoring patterns based on this test.")
```

卡方检验（Chi-squared test）在统计学中有多种应用场景，比如用于检验分类变量之间的独立性、拟合优度检验等。这里我将提供两种常见情况下的卡方检验数学公式：

1. **列联表的独立性检验**：假设我们有一个r行c列的列联表，其中O_{ij}是观察频数，E_{ij}是期望频数，根据给定的独立性假设计算得出。卡方统计量用于检验各个单元格的观察频数是否符合预期频数分布，其公式如下：

$$\chi^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

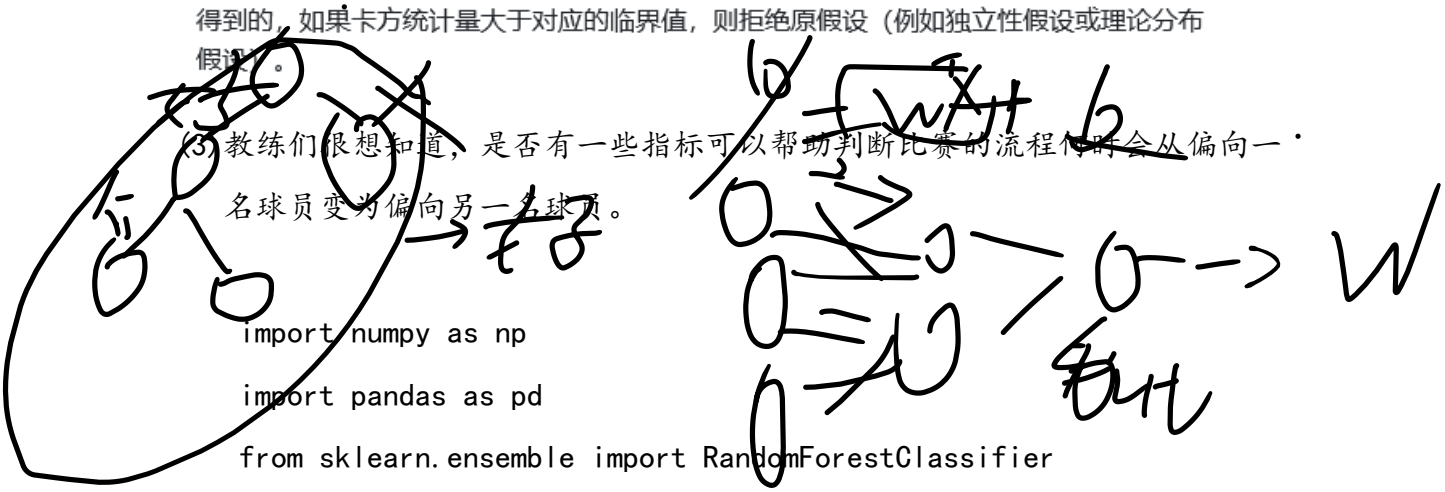
2. **单样本 goodness-of-fit 检验**：当我们要检验一个观测频数分布是否与理论频数分布相匹配时，可以使用单样本卡方检验。设k为类别数，O_i 是第i个类别的观察频数，E_i 是第i个类别的期望频数，那么卡方统计量计算公式为：

$$\chi^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i}$$

这里的自由度通常是k-1，即类别数减一。

无论是哪种情况，显著性水平α下的临界值或p值都是通过查卡方分布表或者使用软件计算得到的，如果卡方统计量大于对应的临界值，则拒绝原假设（例如独立性假设或理论分布假设）。

(3)教练们很想知道，是否有一些指标可以帮助判断比赛的流程何时会从偏向一名球员变为偏向另一名球员。



```
import numpy as np
import pandas as pd
from sklearn.ensemble import RandomForestClassifier
import shap
```

```
# 设置随机数种子以确保可复现性
np.random.seed(42)

# 生成随机的网球比赛数据
def generate_tennis_data(num_matches=100):
    data = []
    for _ in range(num_matches):
        player1_score = np.random.randint(0, 7)
        player2_score = np.random.randint(0, 7)
        match_time = np.random.randint(60, 240)  # 假设比赛时间在
60 到 240 分钟之间
        current_score = f"{player1_score}-{player2_score}"
        data.append([player1_score, player2_score, match_time,
current_score])
    return pd.DataFrame(data, columns=["Player1_Score",
"Player2_Score", "Match_Time", "Current_Score"])

# 创建训练集
train_data = generate_tennis_data(200)
X_train = train_data.drop("Current_Score", axis=1)
y_train = train_data["Current_Score"]

# 使用随机森林模型进行训练
rf_model = RandomForestClassifier(n_estimators=100,
random_state=42)
rf_model.fit(X_train, y_train)

# 创建测试集
test_data = generate_tennis_data(10)
```

```
X_test = test_data.drop("Current_Score", axis=1)

# 进行预测
predictions = rf_model.predict(X_test)

# 使用 SHAP 计算影响预测的权重
explainer = shap.TreeExplainer(rf_model)
shap_values = explainer.shap_values(X_test)

# 输出预测结果和 SHAP 值
for i in range(len(predictions)):
    print(f"Prediction: {predictions[i]}, SHAP Values: {shap_values[i]}")
shap.summary_plot(shap_values, X_test, feature_names=X_test.columns)
plt.show()
```

1 利用提供的至少一场比赛的数据，建立一个模型来预测比赛中的这些波动。哪些因素似乎最有关系(如果有的话)

评价，重要性权重，spsspro 里面的随机森林 RF 的重要性 权重

Bp 神经网络评价计算权重，

Lvq 神经网络模型，都可以加优化算法 SA,GA，GWO

随机森林（Random Forest）是一种集成学习方法，它通过构建多个决策树并取其平均（分类任务）或求和（回归任务）来进行预测。这里我们主要讨论二分类问题的数学公式简化表示。

在随机森林中，单个决策树T的预测可以表示为：

对于一个给定的样本x，假设决策树T根据特征分裂到叶节点j，则该叶节点对应的类别标签预测值 c_j 为：

$$T(x) = c_j$$

随机森林模型对样本x的预测结果是所有单个决策树预测结果的投票或者加权平均：

$$RF(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x)$$

其中：

其中：

- B 表示森林中决策树的数量。
- $T_b(x)$ 是第 b 检测树对样本 x 的预测结果。

对于回归任务，每个树的输出是连续值，所以直接做平均处理；而对于分类任务，每个树输出的是类别的概率或类别标记，通常采用多数投票或概率加权的方式来决定最终类别。

此外，每棵决策树的训练过程中引入了随机性，包括：

- 随机子采样（Bootstrap Sampling）：从原始训练集中有放回地抽取一定数量的样本作为当前树的训练集。
- 随机特征选择：在每个节点分裂时，仅考虑从所有特征中随机选取的一部分特征进行最佳分割点的寻找。

SHAP（SHapley Additive exPlanations）解释器则是基于博弈论中的Shapley值来量化特征对模型预测的影响程度，它提供了一个全局一致、局部精确且具有可加性的特征重要性衡量方式，但其具体计算涉及复杂的理论推导，并非简单的数学公式可以直接表达。

2 鉴于过去比赛中“势头“波动的差异你如何建议球员在新的比赛中与不同的球员交手?

建议：

(4)在一场或多场其他比赛中测试您开发的模型，您对比赛中的波动预测得如何？如果模型有时表现不佳，您是否能找出未来模型中可能需要包含的任何因素如何?您的模型对其他比赛锦标赛、球场表面和其他运动(如乒(如乓球)的通用性如何?

灵敏度分析，rf 如参数改变 1%3%5% 数据改变

评价 rmse，mae 等等越小越好 r2 接近 1 越好 等等

加入数据