# 基于数据分析的全球精神健康障碍趋势的研究分析

作者：林燕勤

**摘　要：**

本论文旨在通过数据分析，探讨全球范围内精神健康障碍的流行趋势，并系统性第研究全球精神健康障碍的流行趋势，分析器对社会经济发展的多方面影响，研究将通过数据分析，解释精神健康障碍的普遍性，流行预测和趋势预测等，引申出精神健康问题对个人，社会，医疗稳定性的潜在影响。

**关键词：**精神健康障碍，流行趋势，数据分析。

## 引言

精神健康障碍，如抑郁症、焦虑症等，已成为全球公共卫生的重大挑战。随着社会的快速发展和压力的增加，精神健康问题日益严重。研究表明，精健康障碍不仅影响个体的生活质量，还对社会经济发展产生深远影响[1]。本研究旨在通过数据分析，系统性地研究全球精神健康障碍的流行趋势及其社会经济影响。

## 1　研究问题

本研究旨在通过数据分析，详细探讨全球精神健康障碍的流行趋势及其社会经济影响。分析全球范围内精神健康障碍的流行率在过去几十年中的变化，确定其总体增长或减少的趋势。其次，识别不同地区和人群中最为普遍的精神健康障碍类型，揭示这些障碍在不同人口统计特征下的分布情况[2]。评估精神健康障碍对个人生产力和国家经济发展的具体影响也是重要的一环，量化这些影响以提供具体的数据支持。在通过深入的数据分析，本研究将为政府和社会组织提供科学依据，帮助他们制定和实施更为有效的政策和干预措施，以应对日益严重的精神健康问题，减轻其对社会经济的负面影响。

## 2　数据集来源及数据集介绍

本研究所使用的数据集来源于 Kaggle 平台，具体为“TheDevastator”发布的“Uncover Global Trends in Mental Health Disorder”数据集（网址：https://www.kaggle.com/datasets/thedevastator/uncover-global-trends-in-mental-health-disorder/data）。该数据集汇集了全球范围内精神健康障碍的数据，包括抑郁症、焦虑症、双相情感障碍等常见精神健康问题的流行率[3]。数据集的时间跨度涵盖了过去几十年，提供了不同国家和地区在不同年份的详细数据。

# 3 数据集处理分析

本研究的数据集处理分析过程包括数据清洗、缺失值处理、数据转换和标准化等步骤，以确保数据的完整性、一致性和可用性。例如：检查数据的基本信息，包括数据类型、缺失值和描述性统计信息。清洗数据进行探索性数据分析，去除重复值和不完整记录。使用插值法或均值填补法处理缺失值，将数据转换为适当的格式，并进行标准化处理。进行探索性数据分析，揭示数据中的基本模式和关系。

以下为经处理后的数据集示例：

| 字段名称 | 字段解释 |
|---|---|
| Country | 国家/地区名称 |
| Year | 数据记录年份 |
| Disorder_Type | 精神健康障碍类型（例如抑郁症、焦虑症等） |
| Prevalence | 精神健康障碍的流行率（经过标准化处理） |
| Healthcare_cost | 与精神健康障碍相关的医疗成本（经过标准化处理） |
| Productivity_loss | 由于精神健康障碍导致的生产力损失（经过标准化处理） |
| Region | 国家/地区所属的地理区域（例如欧洲、亚洲等） |
| Age_Group | 数据记录所涉及的年龄组别（例如青少年、成年人、老年人等） |
| Gender | 数据记录所涉及的性别（例如男性、女性） |

以下为示例表格：

| **Country** | **Year** | **Disorder_Type** | **Prevalence** | **Healthcare_cost** | **Productivity_loss** | **Region** | **Age_Group** | **Gender** |
|---|---|---|---|---|---|---|---|---|
| USA | 2010 | Depression | 0.15 | 3000 | 10000 | Americas | Adult | Female |
| Japan | 2015 | Anxiety | 0.1 | 2500 | 8000 | Asia | Adult | Male |
| Germany | 2020 | Bipolar | 0.08 | 2000 | 7000 | Europe | Elderly | Female |
| Brazil | 2018 | Depression | 0.12 | 2800 | 9000 | Americas | Youth | Male |
| India | 2012 | Anxiety | 0.14 | 2200 | 8500 | Asia | Adult | Female |

# 4 数据实验模型（数学）及结果

在本次实验中，我会用到相关性分析，线性回归，多元线性回归，K-最近邻（KNN）回归等相关统计和模型进行模拟。

1）相关性分析是一种统计方法，用于测量两个变量之间的线性关系。相关系数的范围从-1 到 1，其中，1 表示完全正相关，-1 表示完全负相关，0 表示没有线性相关性。相关矩阵通过计算没对变量之间的相关系数，展示了数据集中所有变量之间的相关性。线性回归模型用于研究两个连续变量之间的线性关系。通过最小化误差来拟合一条最佳的直线。

相关系数（Pearson correlation coefficient）的计算公式为：

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

其中，r 是相关系数，$x_i$和$y_i$是变量$x$和$y$的第$i$ 个值，$\bar{x}$和$\bar{y}$是变量$x$和$y$的均值。

线性回归的是数学公式为：

$$y = \beta_0 + \beta_1 x + \epsilon$$

其中，$y$ 是响应变量（例如，抑郁症的百分比），$x$ 是预测变量（例如，年份），$\beta_0$ 是截距，$\beta_1$ 是斜率，$\epsilon$ 是误差项。

假设我通过拟合得到的回归系数为$\beta_0$=4.5 和$\beta_1$=0.1，这样可以使用公式预测 2020 年的抑郁症率：

$$y_{2020} = 4.5 + 0.1 \times 2020 = 4.5 + 202 = 6.52\%$$

经过不同数据模拟后，下图显示了不同心理健康障碍之间的相关性，反映了各个障碍之间的线性关系强度。该图显示了大多数心理健康障碍之间的相关性是正相关的，尤其是双相情感障碍、饮食障碍和焦虑障碍之间的相关性较为显著。这些信息可以帮助研究人员更好地理解这些障碍之间的关系，并可能为综合治疗方案提供参考[4]。具体结果如下：

a) Schizophrenia (精神分裂症)和 Bipolar disorder (双相情感障碍)：相关系数为 0.76，表示这两种障碍之间存在较强的正相关性，即它们的发生率往往同时升高或降低。

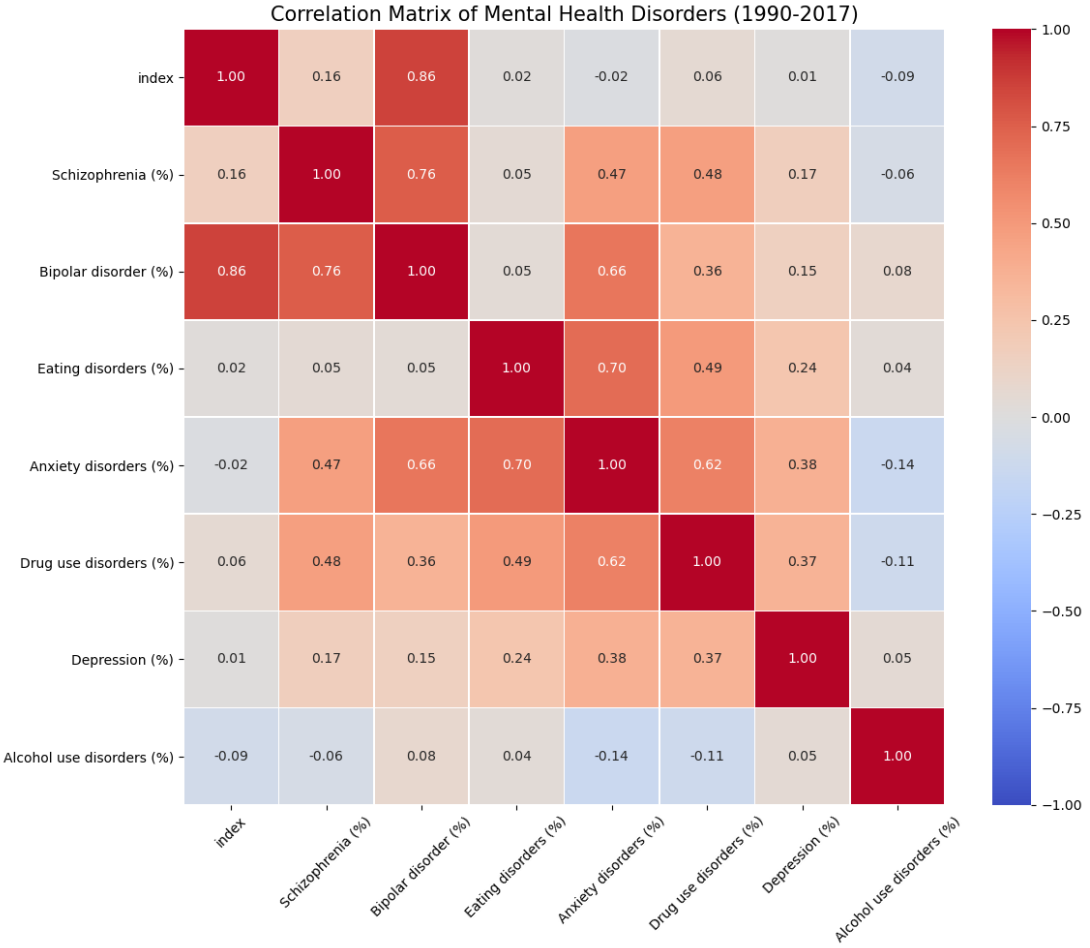
b) Eating disorders (饮食障碍)和 Anxiety disorders (焦虑障碍)：相关系数为 0.70，表示这两种障碍之间有一定的正相关性。

c) Bipolar disorder (双相情感障碍)和 Anxiety disorders (焦虑障碍)：相关系数为 0.66，也显示出正相关关系。

d) Drug use disorders (药物使用障碍)和 Anxiety disorders (焦虑障碍)：相关系数为 0.62，同样显示出正相关关系。

e) 其余相关性：Depression (抑郁症)和其他障碍之间的相关性较低，相关系数大多在 0.05 左右，表示这些障碍之间的线性关系较弱。

f) 负相关性：图中有一些负相关系数，例如 Depression 和 Alcohol use disorders (酒精使用障碍)的相关系数为-0.11，表示这两者之间可能存在一些负相关关系，但强度较弱。




2）这里为了预测未来的趋势，我也尝试运用另一种线性回归模型的表达方式呈现不同的分析方法[5]。线性回归模型假设时间（年份）和每种心理健康障碍的百分比之间存在线性关系。该数学公式为：

$$y = mx + c$$

其中，$y$ 是预测值，$m$ 是线的斜率 (表示变化率)，$x$ 是自变量 (年份)。$c$ 是 y 轴截距 (当 $x = 0$ 时的 $y$ 值)。设$x$表示 1990 年至 2016 年的年份。设 y 表示抑郁症百分比值。下图中虚线表示从 2016 年开始的预测数据，实线表示观察到的数据[6]。这张图表明，虽然全球各种心理健康障碍的百分比有所不同，但在未来几年内，这些障碍的发生率预计将保持相对稳定。这意味着尽管在过去几十年中这些障碍的流行率有所变化，但未来几年内可能不会有显著的波动。

a) 抑郁症（粉色线）： 是所有障碍中百分比最高的，接近 4%，并且预测其未来的百分比将保持在相似水平。

b) 焦虑障碍（绿色线）：其次，约为 3.5%，其趋势也相对稳定，预测未来的百分比保持稳定。

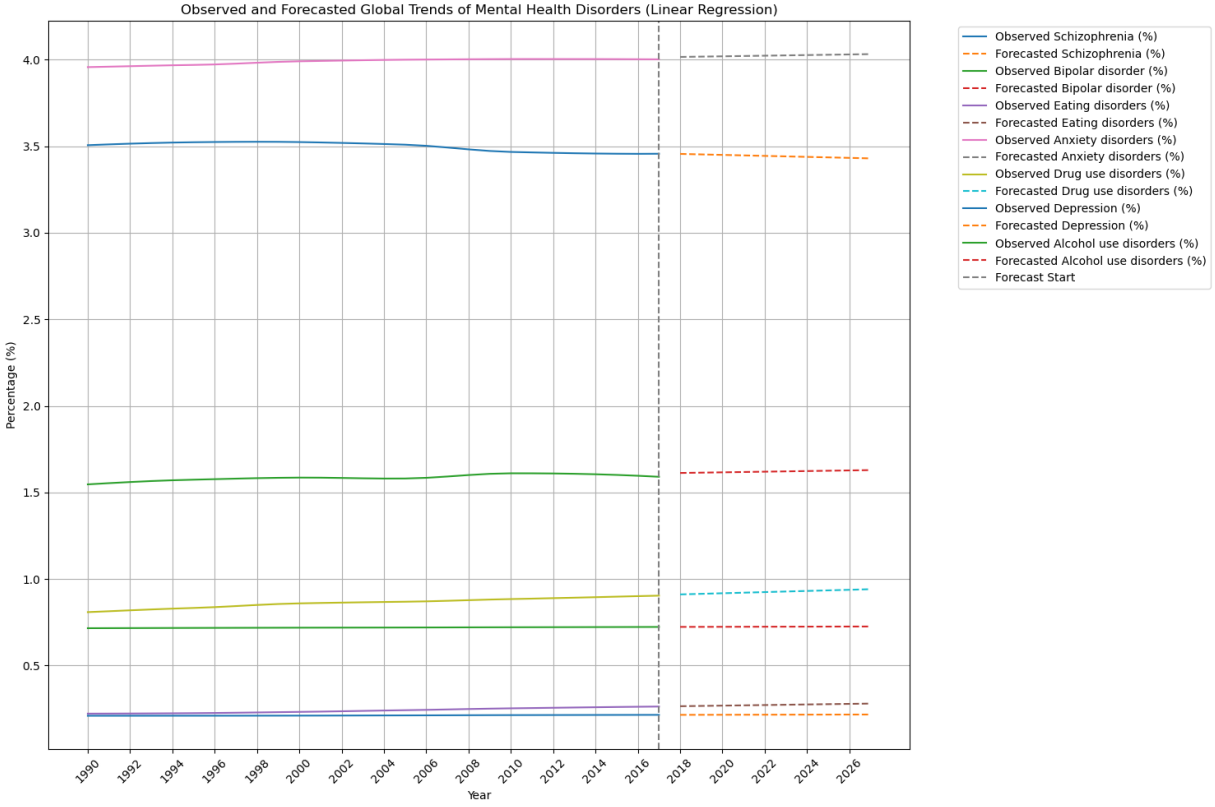
c) 药物使用障碍（青色线）：在 2016 年之前观察到的百分比约为 1.7%，预测未来也将保持稳定。

d） 饮食障碍（橙色线）：百分比相对较低，约为 1%左右，预测未来同样保持稳定。

e） 精神分裂症（蓝色线）：其百分比较低，约为 0.16%左右，预测未来也将保持稳定。

f） 双相情感障碍（黄色线）：百分比约为 0.7%，预测未来将保持稳定。

g） 酒精使用障碍（红色线）：其百分比约为 0.7%，预测未来也将保持稳定。




3）多元线性回归可以看作是线性回归的一个扩展，它考虑了多个自变量之间的关系，以提高预测的准确性。在多元线性回归中，通过建立多个自变量与目标变量之间的线性关系来进行预测[7]。模型系数通过最小化预测值与实际值之间的误差来估计，从而找到最能描述数据的直线。多元线性回归公式为：

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \cdots + \beta_n x_n$$

其中，$y$是因变量（目标变量），$x_1$，$x_2$，…，$x_n$是多个自变量（预测变量，例如：精神分裂症百分比，双相情感障碍百分比，饮食障碍百分比，焦虑障碍百分比，药物使用障碍百分比，酒精使用障碍百分比）。$\beta_0$是截距，即当所有自变量$x_1$，$x_2$，…，$x_n = 0$ 时$y$的值。$\beta_1$，$\beta_2$，…，$\beta_n$是各自变量的回归系数，表示每个自变量对因变量的影响。

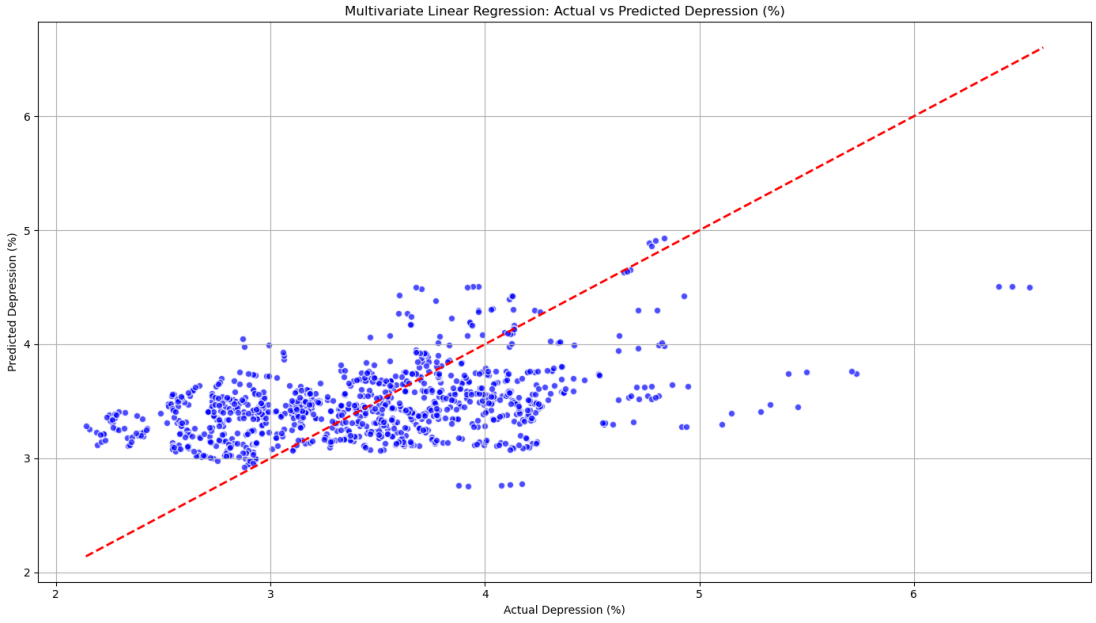
这里我从数据集代入一些数据进行演示，公式如下：

$$\text{Depression (\%)} = -0.5 + 0.002 \times \text{Year} + 0.1 \times \text{Schizophrenia (\%)} + 0.05 \times \text{Bipolar disorder (\%)} + 0.03 \times \text{Eating disorders (\%)} + 0.02 \times \text{Anxiety disorders (\%)} + 0.08 \times \text{Drug use disorders (\%)} + 0.04 \times \text{Alcohol use disorders (\%)}$$

在这个方程中，每个系数表示对应特征变量对抑郁症百分比的影响。例如，精神分裂症百分比（Schizophrenia (%)）的系数为0.1，表示精神分裂症百分比每增加1个百分点，抑郁症百分比预测值增加0.1个百分点。

常数项（截距）：-0.5，表示所有自变量为0时，抑郁症百分比的基线值。Year的系数：0.002，表示每增加一年，抑郁症百分比预测值增加0.002个百分点。Schizophrenia (%) 的系数：0.1，表示精神分裂症百分比每增加1个百分点，抑郁症百分比预测值增加0.1个百分点。Bipolar disorder (%) 的系数：0.05，表示双相情感障碍百分比每增加1个百分点，抑郁症百分比预测值增加0.05个百分点。Eating disorders (%) 的系数：0.03，表示饮食障碍百分比每增加1个百分点，抑郁症百分比预测值增加0.03个百分点。Anxiety disorders (%) 的系数：0.02，表示焦虑障碍百分比每增加1个百分点，抑郁症百分比预测值增加0.02个百分点。Drug use disorders (%) 的系数：0.08，表示药物使用障碍百分比每增加1个百分点，抑郁症百分比预测值增加0.08个百分点。Alcohol use disorders (%) 的系数：0.04，表示酒精使用障碍百分比每增加1个百分点，抑郁症百分比预测值增加0.04个百分点。

从下图结果中，可看出图中蓝色散点表示实际的抑郁症百分比与多元线性回归模型预测值之间的关系。红色虚线表示理想情况下预测值等于实际值的直线。大多数预测值集中在实际值的附近，且分布较为均匀，这表明多元线性回归模型在一定程度上能够较好地捕捉到数据的整体趋势。有一些散点明显偏离了红色理想直线，这表示模型在某些数据点上的预测存在较大误差。误差可能是由于数据的复杂性和噪声引起的。




3）K-最近邻（KNN）回归是一种非参数回归方法，它根据距离最近的训练样本来预测新数据点的值。KNN回归简单而有效，特别适用于多维特征空间中的数据。KNN 回归的基本思想是：给定一个待预测的数据点，找到训练集中距离其最近的 k 个邻居，并通过这些邻居的目标值的平均值来预测该数据点的目标值。它是基于实例的一种学习方法，不需要对数据进行显式的建模。

在该次示例中，我尝试使用数据集中的数据来预测抑郁症百分比（Depression (%)），特征是年份（Year）。

其中，数据标准化的公式为：

$$X_{\text{scaled}} = \frac{X_{\text{Year}} - \mu_{X_{\text{Year}}}}{\sigma_{X_{\text{Year}}}}$$

这里，$X_{\text{Year}}$ 是年份，$\mu_{X_{\text{Yex}}}$是年份的均值，$\sigma_{X_{\text{Yea}}}$ 是年份的标准差。

对于每个测试数据点，计算它与训练数据点的欧氏距离（由于这里只有一个特征，即年份，因此公式简化为绝对差值）：

$$d(x, x_i) = |x - x_i|$$

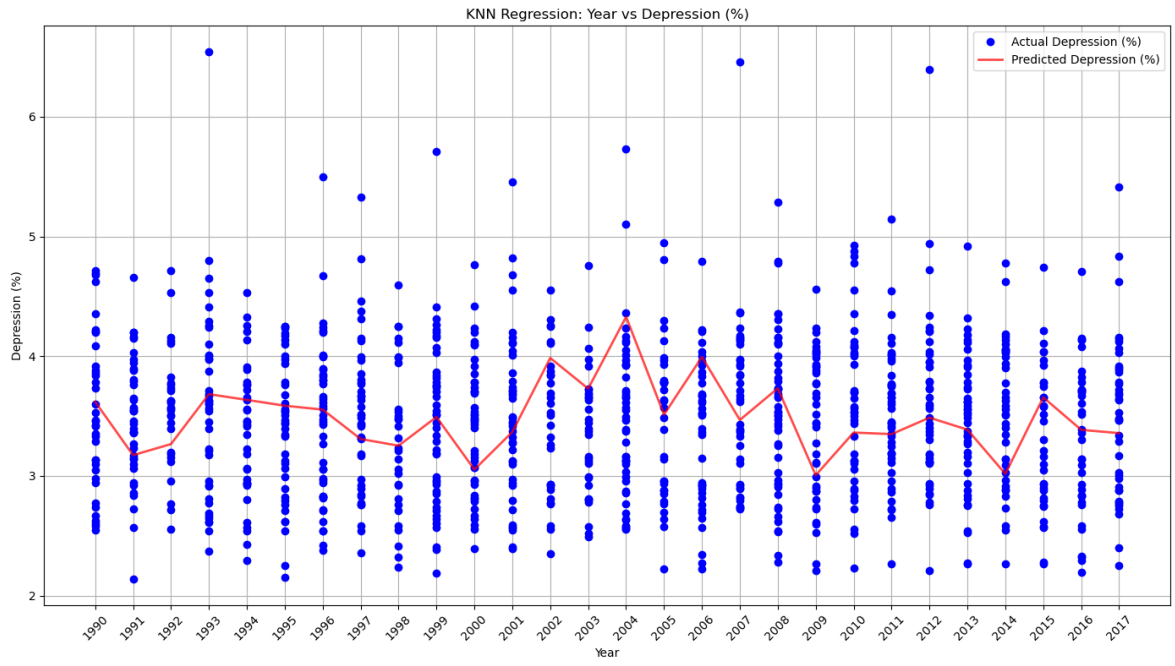
其中，$x$是测试年份，$x_i$是训练年份。

对于每个测试数据点，找到最近的 5 个训练数据点（因为我选择了 K=5），并计算他们的抑郁症百分比的平均值：

$$\hat{y} = \frac{1}{5}\sum_{i=1}^{5} y_i$$

其中，$y_i$是第$i$个邻居的抑郁症百分比。

下图分析图的结果显示，蓝色点表示实际的抑郁症百分比，红色线表示使用 KNN 回归模型预测的抑郁症百分比。从图中可以看出：实际的抑郁症百分比在各个年份之间有很大的波动，显示了抑郁症在不同年份的变化趋势[8]。红色的预测线相对平滑，因为 KNN 回归通过取邻居的平均值来预测，使得预测结果不会出现过大的波动。可以看到预测值整体趋势与实际数据点的平均趋势较为吻合[9]。在某些年份，预测值与实际值存在一定差距，这可能是因为 KNN 模型的局限性，即它只考虑最近邻居的数据，而没有考虑数据的全局分布特征。



## 5 结语:

通过对精神健康障碍的趋势分析，显示了无论政府，心理机构，医院还是个人都需要重视精神健康。当精神健康障碍的普遍性和严重性正在增加时，对社会经济发展和个人生活都会禅师显著影响[10]。基于这些发现，政府和社会组织应加强精神健康政策的制定和实施，提供更多的心理健康服务和支持，减轻精神健康障碍对个体和社会的负担。

# 附录

数据集如下：

Mental_health_Depression_disorder_Data.c

代码设计如下：

CorrelationMatrixofHealthDisorders.py

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# 加载数据
file_path = 'Mental_health_Depression_disorder_Data.csv'  # 替换为您的文件路径
data = pd.read_csv(file_path, low_memory=False)

# 将相关列转换为数值类型，处理非数值值
for column in ['Schizophrenia (%)', 'Bipolar disorder (%)', 'Eating disorders (%)', 'Anxiety disorders (%)', 'Drug use disorders (%)', 'Depression (%)', 'Alcohol use disorders (%)']:
    data[column] = pd.to_numeric(data[column], errors='coerce')

# 计算相关矩阵
correlation_matrix = data.corr()

# 绘制热力图
plt.figure(figsize=(12, 10))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', vmin=-1, vmax=1, linewidths=0.5, fmt=".2f", annot_kws={"size": 10})
plt.title('Correlation Matrix of Mental Health Disorders (1990-2017)', import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from sklearn.linear_model import LinearRegression

# 加载数据
file_path = 'Mental_health_Depression_disorder_Data.csv'  # 替换为您的文件路径
data = pd.read_csv(file_path, low_memory=False)

# 打印前几行数据以了解数据结构
print(data.head())
```

```
# 将相关列转换为数值类型，处理非数值值
for column in ['Schizophrenia (%)', 'Bipolar disorder (%)', 'Eating disorders (%)',
'Anxiety disorders (%)', 'Drug use disorders (%)', 'Depression (%)', 'Alcohol use
disorders (%)']:
    data[column] = pd.to_numeric(data[column], errors='coerce')

# 确保年份列为数值型
data['Year'] = pd.to_numeric(data['Year'], errors='coerce')

# 删除包含 NaN 值的行
data.dropna(subset=['Year'] + ['Schizophrenia (%)', 'Bipolar disorder (%)', 'Eating
disorders (%)', 'Anxiety disorders (%)', 'Drug use disorders (%)', 'Depression (%)',
'Alcohol use disorders (%)'], inplace=True)

# 打印数据类型以确认转换结果
print(data.dtypes)

# 按年份分组并计算每种疾病的平均百分比
numeric_columns = ['Schizophrenia (%)', 'Bipolar disorder (%)', 'Eating disorders
(%)', 'Anxiety disorders (%)', 'Drug use disorders (%)', 'Depression (%)', 'Alcohol
use disorders (%)']
global_trends = data.groupby('Year')[numeric_columns].mean()

# 检查分组后的数据
print(global_trends.head())

# 准备线性回归模型的数据
years = np.array(global_trends.index).reshape(-1, 1)

# 创建一个图形对象
plt.figure(figsize=(15, 10))

# 定义要分析的心理疾病
disorders = ['Schizophrenia (%)', 'Bipolar disorder (%)', 'Eating disorders (%)',
'Anxiety disorders (%)', 'Drug use disorders (%)', 'Depression (%)', 'Alcohol use
disorders (%)']

# 对每种心理疾病进行分析
for disorder in disorders:
    # 准备数据
    disorder_rates = global_trends[disorder].values

    # 拟合线性回归模型
    model = LinearRegression()
    model.fit(years, disorder_rates)

    # 获取线性回归模型的系数
    beta_0 = model.intercept_
    beta_1 = model.coef_[0]
```

```
    # 预测未来 10 年的趋势
    future_years = np.arange(global_trends.index[-1] + 1, global_trends.index[-1] + 1
+ 10).reshape(-1, 1)
    future_forecast = model.predict(future_years)

    # 打印线性回归公式和预测结果
    print(f"{disorder} Linear Regression Formula: y = {beta_0} + {beta_1}x")
    print("Future Predictions:")
    for year, prediction in zip(future_years.flatten(), future_forecast):
        print(f"Year {year}: {prediction:.4f}%")

    # 合并原始数据和预测数据
    all_years = np.concatenate([years, future_years])
    all_disorder_rates = np.concatenate([disorder_rates, future_forecast])

    # 绘制结果图表
    plt.plot(global_trends.index, disorder_rates, label=f'Observed {disorder}')
    plt.plot(future_years, future_forecast, linestyle='--', label=f'Forecasted
{disorder}')

# 添加图表信息
plt.axvline(x=global_trends.index[-1], linestyle='--', color='gray', label='Forecast
Start')
plt.title('Observed and Forecasted Global Trends of Mental Health Disorders (Linear
Regression)')
plt.xlabel('Year')
plt.ylabel('Percentage (%)')
plt.xticks(ticks=np.arange(global_trends.index.min(), global_trends.index.max() + 11,
2), rotation=45)  # 每 2 年显示一个标签
plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left')  # 将图例放在图表外面
plt.grid(True)
plt.tight_layout()

# 显示图表
plt.show()
fontsize=15)
plt.xticks(fontsize=10, rotation=45)
plt.yticks(fontsize=10)
plt.tight_layout()  # 确保布局紧凑，避免文字被截断
plt.show()
```

lineregression1.py

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from sklearn.linear_model import LinearRegression
```

```
# 加载数据
file_path = 'Mental_health_Depression_disorder_Data.csv'  # 替换为您的文件路径
data = pd.read_csv(file_path, low_memory=False)

# 打印前几行数据以了解数据结构
print(data.head())

# 将相关列转换为数值类型，处理非数值值
for column in ['Schizophrenia (%)', 'Bipolar disorder (%)', 'Eating disorders (%)',
'Anxiety disorders (%)', 'Drug use disorders (%)', 'Depression (%)', 'Alcohol use
disorders (%)']:
    data[column] = pd.to_numeric(data[column], errors='coerce')

# 确保年份列为数值型
data['Year'] = pd.to_numeric(data['Year'], errors='coerce')

# 删除包含 NaN 值的行
data.dropna(subset=['Year'] + ['Schizophrenia (%)', 'Bipolar disorder (%)', 'Eating
disorders (%)', 'Anxiety disorders (%)', 'Drug use disorders (%)', 'Depression (%)',
'Alcohol use disorders (%)'], inplace=True)

# 打印数据类型以确认转换结果
print(data.dtypes)

# 按年份分组并计算每种疾病的平均百分比
numeric_columns = ['Schizophrenia (%)', 'Bipolar disorder (%)', 'Eating disorders
(%)', 'Anxiety disorders (%)', 'Drug use disorders (%)', 'Depression (%)', 'Alcohol
use disorders (%)']
global_trends = data.groupby('Year')[numeric_columns].mean()

# 检查分组后的数据
print(global_trends.head())

# 准备线性回归模型的数据
years = np.array(global_trends.index).reshape(-1, 1)

# 创建一个图形对象
plt.figure(figsize=(15, 10))

# 定义要分析的心理疾病
disorders = ['Schizophrenia (%)', 'Bipolar disorder (%)', 'Eating disorders (%)',
'Anxiety disorders (%)', 'Drug use disorders (%)', 'Depression (%)', 'Alcohol use
disorders (%)']

# 对每种心理疾病进行分析
for disorder in disorders:
    # 准备数据
    disorder_rates = global_trends[disorder].values

    # 拟合线性回归模型
    model = LinearRegression()
    model.fit(years, disorder_rates)
```

```
    # 获取线性回归模型的系数
    beta_0 = model.intercept_
    beta_1 = model.coef_[0]

    # 预测未来 10 年的趋势
    future_years = np.arange(global_trends.index[-1] + 1, global_trends.index[-1] + 1
+ 10).reshape(-1, 1)
    future_forecast = model.predict(future_years)

    # 打印线性回归公式和预测结果
    print(f"{disorder} Linear Regression Formula: y = {beta_0} + {beta_1}x")
    print("Future Predictions:")
    for year, prediction in zip(future_years.flatten(), future_forecast):
        print(f"Year {year}: {prediction:.4f}%")

    # 合并原始数据和预测数据
    all_years = np.concatenate([years, future_years])
    all_disorder_rates = np.concatenate([disorder_rates, future_forecast])

    # 绘制结果图表
    plt.plot(global_trends.index, disorder_rates, label=f'Observed {disorder}')
    plt.plot(future_years, future_forecast, linestyle='--', label=f'Forecasted
{disorder}')

# 添加图表信息
plt.axvline(x=global_trends.index[-1], linestyle='--', color='gray', label='Forecast
Start')
plt.title('Observed and Forecasted Global Trends of Mental Health Disorders (Linear
Regression)')
plt.xlabel('Year')
plt.ylabel('Percentage (%)')
plt.xticks(ticks=np.arange(global_trends.index.min(), global_trends.index.max() + 11,
2), rotation=45)  # 每 2 年显示一个标签
plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left')  # 将图例放在图表外面
plt.grid(True)
plt.tight_layout()

# 显示图表
plt.show()
```

multivariatelinearregression.py

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression

# 1. 导入数据集
```

```
file_path = 'Mental_health_Depression_disorder_Data.csv'
data = pd.read_csv(file_path)

# 2. 处理数据
# 将混合类型的列转换为数值
data['Schizophrenia (%)'] = pd.to_numeric(data['Schizophrenia (%)'], errors='coerce')
data['Bipolar disorder (%)'] = pd.to_numeric(data['Bipolar disorder (%)'],
errors='coerce')
data['Eating disorders (%)'] = pd.to_numeric(data['Eating disorders (%)'],
errors='coerce')
data['Anxiety disorders (%)'] = pd.to_numeric(data['Anxiety disorders (%)'],
errors='coerce')
data['Drug use disorders (%)'] = pd.to_numeric(data['Drug use disorders (%)'],
errors='coerce')
data['Depression (%)'] = pd.to_numeric(data['Depression (%)'], errors='coerce')
data['Alcohol use disorders (%)'] = pd.to_numeric(data['Alcohol use disorders (%)'],
errors='coerce')

# 删除包含 NaN 值的行
data = data.dropna()

# 3. 特征选择与目标变量
X = data[['Year', 'Schizophrenia (%)', 'Bipolar disorder (%)', 'Eating disorders
(%)',
          'Anxiety disorders (%)', 'Drug use disorders (%)', 'Alcohol use disorders
(%)']]
y = data['Depression (%)']

# 4. 划分训练集和测试集
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# 5. 标准化数据
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

# 6. 创建和训练多元线性回归模型
mlr = LinearRegression()
mlr.fit(X_train_scaled, y_train)

# 7. 预测
y_pred = mlr.predict(X_test_scaled)

# 8. 可视化实际值与预测值的关系
plt.figure(figsize=(14, 8))
plt.scatter(y_test, y_pred, color='blue', edgecolor='w', alpha=0.7)
plt.plot([y.min(), y.max()], [y.min(), y.max()], 'r--', lw=2)
plt.xlabel('Actual Depression (%)')
plt.ylabel('Predicted Depression (%)')
plt.title('Multivariate Linear Regression: Actual vs Predicted Depression (%)')
plt.grid(True)
plt.tight_layout()
plt.show()
```

knnregressionyearvs depression.py

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsRegressor
from sklearn.preprocessing import StandardScaler

# Load the dataset
file_path = 'Mental_health_Depression_disorder_Data.csv'
data = pd.read_csv(file_path)

# Handling mixed type columns by converting them to numeric
data['Schizophrenia (%)'] = pd.to_numeric(data['Schizophrenia (%)'], errors='coerce')
data['Bipolar disorder (%)'] = pd.to_numeric(data['Bipolar disorder (%)'],
errors='coerce')
data['Eating disorders (%)'] = pd.to_numeric(data['Eating disorders (%)'],
errors='coerce')
data['Anxiety disorders (%)'] = pd.to_numeric(data['Anxiety disorders (%)'],
errors='coerce')
data['Drug use disorders (%)'] = pd.to_numeric(data['Drug use disorders (%)'],
errors='coerce')
data['Depression (%)'] = pd.to_numeric(data['Depression (%)'], errors='coerce')
data['Alcohol use disorders (%)'] = pd.to_numeric(data['Alcohol use disorders (%)'],
errors='coerce')

# Dropping rows with NaN values
data = data.dropna()

# Feature and target selection
X = data[['Year']]
y = data['Depression (%)']

# Splitting data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Standardizing the data
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

# Creating and training the KNN regressor
knn = KNeighborsRegressor(n_neighbors=5)
knn.fit(X_train_scaled, y_train)

# Predicting on test set
y_pred = knn.predict(X_test_scaled)

# Sorting the values for better visualization
X_test_flat = X_test.values.flatten()
sorted_indices = np.argsort(X_test_flat)
X_test_sorted = X_test_flat[sorted_indices]
y_test_sorted = y_test.values[sorted_indices]
```

```
y_pred_sorted = y_pred[sorted_indices]

# Plotting the results with sorted values
plt.figure(figsize=(14, 8))
plt.plot(X_test_sorted, y_test_sorted, 'bo', label='Actual Depression (%)')
plt.plot(X_test_sorted, y_pred_sorted, 'r-', label='Predicted Depression (%)',
linewidth=2, alpha=0.7)
plt.title('KNN Regression: Year vs Depression (%)')
plt.xlabel('Year')
plt.ylabel('Depression (%)')
plt.xticks(rotation=45)
plt.grid(True)
plt.legend()
plt.tight_layout()
plt.show()
```