

**《数据可视化课程设计》报告**

**专 业：** 数据科学与大数据技术

**班 级：** 大数据2201班

**姓 名：** 秦恒睿

**学 号：** 21408080106

**2025年1月**

**题目：美食团购网站数据的可视化分析与研究**

**1、实验目的**

随着互联网技术的发展，美食团购网站已经成为消费者寻找优惠餐饮服务的重要平台。这类网站不仅为餐馆提供了有效的推广渠道，也极大地便利了消费者的日常饮食选择。然而，在海量的交易数据背后隐藏着丰富的信息资源，如何有效挖掘和利用这些数据成为了研究的重点。本项目旨在通过对某知名美食团购网站的数据进行深入分析，揭示出城市餐饮市场中的潜在规律和发展趋势，为企业决策提供科学依据，同时也为消费者带来更加个性化的推荐体验。

探索城市餐饮空间分布特征，通过绘制不同城市的餐饮密度热力图，直观展示各区域内餐馆的数量分布情况。这有助于理解城市的空间布局特点以及商业活动的聚集效应，进而为新餐馆选址提供参考建议。例如，在某些城市中心或交通枢纽附近可能存在较高的餐馆密度，而其他区域则相对稀疏；这种差异可能是由多种因素共同作用的结果，如人口密度、交通便利性等。

分析菜系受欢迎程度及其地域性差异，根据不同菜系的人气指数（如订单量、评价数量）制作条形图，比较各类菜系在当地市场的接受度。一方面，这可以反映当地居民的口味偏好，另一方面也能看出外来文化的渗透程度。比如，在沿海城市可能会发现更多对海鲜类菜品的需求；而在内陆地区，传统的川湘风味可能更受青睐。此外，我们还将探讨同一菜系在不同城市之间的表现是否存在显著差异，以期发现跨区域传播的可能性。

研究评论数与人均消费的关系，采用回归分析方法探究评论数作为自变量时对因变量（即人均消费）的影响。评论不仅是顾客满意度的一种体现，也可能间接影响餐馆的定价策略和服务质量。因此，了解两者之间的关系对于评估餐馆的真实价值具有重要意义。具体来说，我们将检验更多的评论是否意味着更低或更高的人均消费，并尝试解释背后的机制。例如，高评分但价格适中的餐馆可能会吸引更多回头客，从而积累大量好评；相反，一些定位高端的餐厅即使评论较少也可能维持较高的价位。

识别高人气高消费餐馆集群，运用聚类算法找出那些既有较高人气又具备较高消费水平的餐馆群组。这类餐馆往往位于城市的核心商圈或是旅游热点地带，它们不仅满足了本地居民的消费需求，也成为外地游客体验地方特色的好去处。通过对这些餐馆的特征分析，我们可以总结出成功运营的关键要素，如独特的菜品设计、优质的客户服务等。同时，这也为潜在投资者提供了有价值的市场情报，帮助他们更好地把握商机。

本项目的成果将从多个维度丰富现有的餐饮行业研究，特别是在以下几个方面：通过对实际数据的实证分析，验证和完善已有的关于餐饮市场结构、消费者行为等方面的知识体系。为企业管理者制定营销策略、优化产品组合等提供具体的行动指南；也为政府部门规划城市发展、改善公共服务设施布局提供参考。促进健康合理的饮食文化发展，引导公众形成科学理性的消费观念；同时，借助数据分析的力量推动餐饮行业的数字化转型，提升整个产业链的服务效率。

综上所述，本项目不仅致力于解决当前存在的实际问题，还力求探索新的研究方向，为未来的学术探讨和技术应用打下坚实的基础。通过对美食团购网站数据的可视化分析与研究，我们希望能够构建起一座连接数据世界与现实生活的桥梁，让每个人都能从中受益

**2、算法设计**

**2.1数据预处理**

我们对原始数据进行了以下预处理步骤：缺失值处理：删除含有大量缺失值的行，特别是那些影响关键变量（如评论数、人均价格）完整性的记录。数值化转换：将包含货币符号或其他非数字字符的字段转换为纯数值格式，以便于后续计算。异常检测：检查并移除可能存在的极端值或错误录入的数据点，以保证模型训练时不会受到异常数据的影响。

**2.2普通最小二乘法(OLS)回归模型**

**2.2.1 模型原理**

普通最小二乘法（Ordinary Least Squares, OLS）是一种用于估计线性回归模型参数的方法。其基本思想是找到一条直线（或多维空间中的超平面），使得所有观测点到这条直线的距离平方和最小。OLS假设因变量y*y*与自变量x*x*之间存在线性关系，并且误差项服从正态分布。

**2.2.2 数学公式**

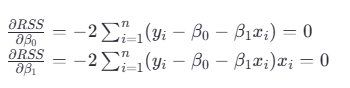
对于单变量线性回归模型，可以表示为：

 (2.1)

目标是最小化残差平方和（RSS）：

 (2.2)

为了找到最优的β0*β*0和β1*β*1，我们求解以下方程组：

 (2.3)

通过求解上述偏导数方程，可以得到β0*β*0和β1*β*1的最佳估计值。

**2.2.3 算法流程**

导入数据：读取并清理数据，确保所有参与计算的列都是数值类型。

定义模型：使用Statsmodels库中的OLS类创建一个回归模型对象。

拟合模型：调用fit()方法来拟合模型，获得最优的β0*β*0​和β1*β*1​。

结果解释：输出回归系数、R²值等统计指标，评估模型的拟合优度。

**2.3 聚类分析：KMeans聚类算法**

**2.3.1 算法原理**

KMeans是一种基于距离的无监督学习算法，旨在将一组未标记的数据划分为k*k*个簇（clusters）。该算法通过迭代优化的方式寻找每个簇的中心点（centroids），并将每个数据点分配给最近的簇心。KMeans的目标是最小化所有簇内部成员与其对应簇心之间的平均距离平方和。

**2.3.2 数学公式**

设C={c1,c2,...,ck}*C*={*c*1,*c*2,...,*ck*}为k*k*个簇心，则KMeans的目标函数可以表示为：

 (2.4)

**2.3.3 算法流程**

初始化：随机选择k*k*个数据点作为初始簇心。

分配簇：根据当前簇心，将每个数据点分配给最近的簇。

更新簇心：重新计算每个簇的新簇心位置。

收敛判断：如果簇心不再变化或变化小于设定阈值，则停止迭代；否则返回步骤2继续执行。

**3、程序调试**

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

import statsmodels.api as sm

from sklearn.cluster import KMeans

import os  
import re

plt.rcParams['font.sans-serif'] = ['SimHei']

plt.rcParams['axes.unicode\_minus'] = False

file\_template = r"C:\Users\秦\Desktop\{}.xlsx"

cities = ['西安', '南京', '广州', '成都', '武汉']

**3.1数据清洗**

def clean\_numeric\_column(series):

"""清理包含货币符号或其他非数字字符的数值列"""

return pd.to\_numeric(series.str.replace(r'[^\d.]', '', regex=True), errors='coerce')

def load\_and\_validate\_data(city):

try:

df = pd.read\_excel(file\_template.format(city))

if df.empty:

print(f"无法读取 {city} 的数据")

return None

# 检查必要的列是否存在

required\_columns = ['地区', '类型', '评论数', '人均', '店名']

if not all(col in df.columns for col in required\_columns):

print(f"{city} 的数据缺少必要列: {required\_columns}")

return None

# 清理并确保评论数和人均消费列为数值类型

df['评论数'] = pd.to\_numeric(df['评论数'], errors='coerce')

df['人均'] = clean\_numeric\_column(df['人均'])

# 删除含有缺失值的行

df = df.dropna(subset=['评论数', '人均'])

if df.empty:

print(f"没有有效的评论数和人均价格数据用于 {city}")

return None

return df

except Exception as e:

print(f"读取 {city} 数据时发生错误: {e}")

return None

# 定义自动分类规则（使用正则表达式），并设置优先级  
category\_rules = [  
 (r'自助', '自助'), # 最高优先级  
 (r'(鱼)', '鱼'),  
 (r'(汤锅)', '火锅'),  
 (r'(火锅)', '火锅'),  
 (r'(烤|烧|烤串)', '烧烤'),  
 (r'(日式|韩式|韩国|日本|料理)', '料理'),  
 (r'(披萨|比萨|牛排)', '西餐')  
]  
# 函数用于根据规则自动分类类型，并考虑优先级  
def auto\_categorize\_restaurant\_type(type\_str):  
 for pattern, category in category\_rules:  
 if re.search(pattern, type\_str, re.IGNORECASE):  
 return category  
 return type\_str # 如果没有匹配，则保持原样  
  
# 读取并处理每个城市的Excel文件，直接保存回原文件  
file\_template = r"C:\Users\秦\Desktop\{}.xlsx"  
for city in ['成都', '广州', '武汉', '北京', '西安', '南京']:  
 file\_path = file\_template.format(city)  
 if os.path.exists(file\_path):  
 df = pd.read\_excel(file\_path)  
 # 应用自动分类规则到类型列  
 df['类型'] = df['类型'].apply(auto\_categorize\_restaurant\_type)  
 df.to\_excel(file\_path, index=False)  
 print(f"Processed and saved {city} data back to {file\_path}")  
print("All files processed.")

# 定义要删除的关键词列表  
keywords\_to\_remove = [  
 '招牌', '推荐', '新品', '精品', '手工', '餐前', '餐后', '秘制', '极品', '特色', '经典'  
]  
# 创建一个正则表达式模式来匹配所有关键词  
# 使用非捕获组(?:)和字符边界\B来匹配可能出现的连续汉字  
pattern = re.compile('|'.join(re.escape(word) for word in keywords\_to\_remove), re.IGNORECASE)  
# 函数用于根据模式清理招牌菜字符串  
def clean\_signature\_dish(dish\_str):  
 if pd.notna(dish\_str): # 检查是否为非NaN值  
 cleaned\_str = pattern.sub('', dish\_str).strip() # 移除匹配的关键词并去除多余空格  
 if cleaned\_str != dish\_str:  
 print(f"Original: {dish\_str} -> Cleaned: {cleaned\_str}")  
 return cleaned\_str  
 return dish\_str  
# 读取并处理每个城市的Excel文件，直接保存回原文件  
file\_template = r"C:\Users\秦\Desktop\{}.xlsx"  
  
for city in ['成都', '广州', '武汉', '西安', '南京']:  
 file\_path = file\_template.format(city)  
 if os.path.exists(file\_path):  
 df = pd.read\_excel(file\_path)  
 # 应用清理函数到所有招牌菜列  
 signature\_columns = ['招牌菜1', '招牌菜2', '招牌菜3']  
 for col in signature\_columns:  
 if col in df.columns:  
 df[col] = df[col].apply(clean\_signature\_dish)  
 # 直接保存回原文件  
 df.to\_excel(file\_path, index=False)  
 print(f"Processed and saved {city} data back to {file\_path}")  
 else:  
 print(f"{city} 的数据文件不存在: {file\_path}")  
print("All files processed.")

**3.2普通最小二乘法(OLS)回归模型分析评论数与人均价格的关系**

def analyze\_price\_correlation(df, city):

if df is None or df.empty:

print(f"没有有效的数据用于分析 {city} 的评论数与人均价格的关系")

return

X = df[['评论数']]

y = df['人均']

X = sm.add\_constant(X)

model = sm.OLS(y, X).fit()

predictions = model.predict(X)

print(model.summary())

plt.figure(figsize=(12, 8))

plt.scatter(df['评论数'], df['人均'], label='实际值')

plt.plot(df['评论数'], predictions, color='red', label='拟合线')

plt.title(f'{city} 评论数与人均价格的关系')

plt.xlabel('评论数')

plt.ylabel('人均价格')

plt.legend()

plt.show()

**3.3 KMeans聚类算法识别高人气高消费餐馆集群**

def find\_high\_popularity\_high\_consumption\_clusters(df, city):

if df is None or df.empty:

print(f"没有有效的数据用于聚类分析 {city} 的高人气、高消费餐馆")

return

# 确保只使用数值列进行聚类

X = df[['评论数', '人均']].dropna()

if X.empty:

print(f"没有有效的评论数和人均价格数据用于聚类分析 {city}")

return

kmeans = KMeans(n\_clusters=3, random\_state=42)

cluster\_labels = kmeans.fit\_predict(X)

# 创建一个新的数据框以保留原始餐馆名称和其他信息

df\_clusters = df[['店名', '评论数', '人均']].copy()

df\_clusters['Cluster'] = cluster\_labels

# 找出高人气、高消费的簇

high\_cluster = df\_clusters.groupby('Cluster')['人均'].mean().idxmax()

high\_density\_clusters = df\_clusters[df\_clusters['Cluster'] == high\_cluster]

print(f"{city} 高人气、高消费的餐馆集群：")

print(high\_density\_clusters[['店名', '评论数', '人均']])

# 可视化聚类结果

plt.figure(figsize=(12, 8))

plt.scatter(df\_clusters['评论数'], df\_clusters['人均'], c=df\_clusters['Cluster'], cmap='viridis')

plt.title(f'{city} 餐馆聚类分析')

plt.xlabel('评论数')

plt.ylabel('人均消费')

plt.show()

**3.4可视化**

for city in cities:

df = load\_and\_validate\_data(city)

if df is not None and not df.empty:

plot\_density\_heatmap(df, city)

plot\_cuisine\_popularity(df, city, min\_percentage=2) # 你可以调整min\_percentage参数来改变阈值

analyze\_price\_correlation(df, city)

find\_high\_popularity\_high\_consumption\_clusters(df, city)

# 每个城市餐馆的词云图，包括类型列

def generate\_wordcloud(texts, title, signature\_weight=1, type\_weight=0.25):

# 分离招牌菜和类型列

signature\_dishes = [word for word in texts if word not in combined\_data['类型'].unique()]

types = [word for word in texts if word in combined\_data['类型'].unique()]

# 计算词频，应用权重

signature\_counts = Counter(signature\_dishes)

type\_counts = Counter(types)

weighted\_signature = {word: count \* signature\_weight for word, count in signature\_counts.items()}

weighted\_types = {word: count \* type\_weight for word, count in type\_counts.items()}

# 合并词频

final\_counts = Counter(weighted\_signature) + Counter(weighted\_types)

try:

wordcloud = WordCloud(font\_path='simhei.ttf', width=800, height=400,

background\_color='white').generate\_from\_frequencies(final\_counts)

except OSError: # 如果找不到字体文件

print("Warning: simhei.ttf not found, using default font.")

wordcloud = WordCloud(width=800, height=400, background\_color='white').generate\_from\_frequencies(final\_counts)

plt.figure(figsize=(10, 5))

plt.imshow(wordcloud, interpolation='bilinear')

plt.axis("off")

plt.title(title)

plt.show()

# 生成词云图

for city in set(combined\_data['City']):

if city in excluded\_cities:

continue

city\_data = combined\_data[combined\_data['City'] == city]

# 合并所有招牌菜列，并去除NaN值

signature\_dishes = []

for col in ['招牌菜1', '招牌菜2', '招牌菜3']:

signature\_dishes.extend(city\_data[col].dropna().astype(str).tolist())

types = city\_data['类型'].dropna().astype(str).tolist()

signature\_dishes.extend(types)

if signature\_dishes:

generate\_wordcloud(signature\_dishes, f'{city}招牌菜及菜系词云图')

def plot\_density\_heatmap(df, city):

if df is None or df.empty:

print(f"没有有效的数据用于绘制 {city} 的热力图")

return

area\_counts = df['地区'].value\_counts().reset\_index()

area\_counts.columns = ['地区', '餐馆数量']

heatmap\_data = area\_counts.pivot\_table(index='地区', values='餐馆数量', aggfunc='sum').fillna(0)

plt.figure(figsize=(12, 8))

sns.heatmap(heatmap\_data, cmap="YlGnBu", cbar\_kws={'label': '餐馆数量'})

plt.title(f'{city} 不同区域的餐饮密度')

plt.xlabel('') # 移除X轴标签，因为只有一个列

plt.ylabel('地区')

plt.show()

def plot\_cuisine\_popularity(df, city, min\_percentage=2):

if df is None or df.empty:

print(f"没有有效的数据用于绘制 {city} 的菜系受欢迎程度图")

return

cuisine\_counts = df['类型'].value\_counts()

total\_restaurants = cuisine\_counts.sum()

filtered\_cuisines = cuisine\_counts[cuisine\_counts / total\_restaurants \* 100 >= min\_percentage]

if filtered\_cuisines.empty:

print(f"没有满足条件的菜系用于 {city}")

return

cuisine\_data = filtered\_cuisines.reset\_index()

cuisine\_data.columns = ['类型', '数量']

plt.figure(figsize=(12, 8))

sns.barplot(x='数量', y='类型', data=cuisine\_data, palette='viridis')

plt.title(f'{city} 不同菜系的受欢迎程度（排除占比小于{min\_percentage}%的菜系）')

plt.xlabel('餐馆数量')

plt.ylabel('菜系类型')

plt.show()

def plot\_cuisine\_pie\_chart(city\_data, city, threshold=1.3): # 修改阈值为1%

cuisine\_counts = city\_data['类型'].value\_counts(normalize=True) \* 100

# 合并占比小于threshold%的类别到"其他"

other\_categories = cuisine\_counts[cuisine\_counts < threshold].sum()

top\_categories = cuisine\_counts[cuisine\_counts >= threshold]

if other\_categories > 0:

top\_categories['其他'] = other\_categories

plt.figure(figsize=(8, 6))

plt.pie(top\_categories, labels=top\_categories.index, autopct='%1.1f%%', startangle=140)

plt.title(f'{city}不同菜系占比')

plt.show()

# 绘制饼图

for city in set(combined\_data['City']):

if city in excluded\_cities:

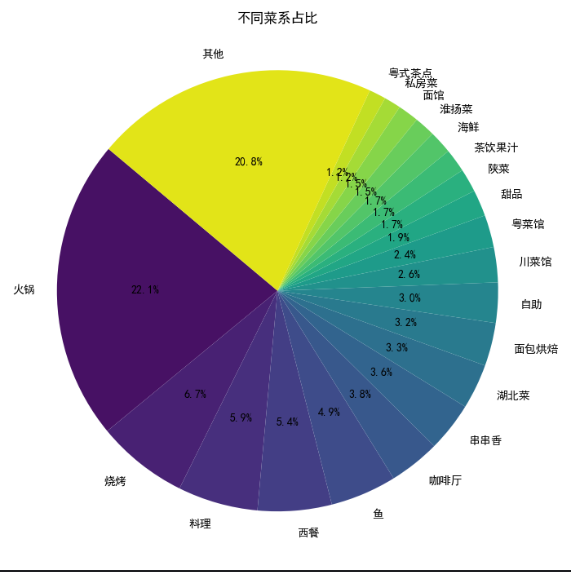
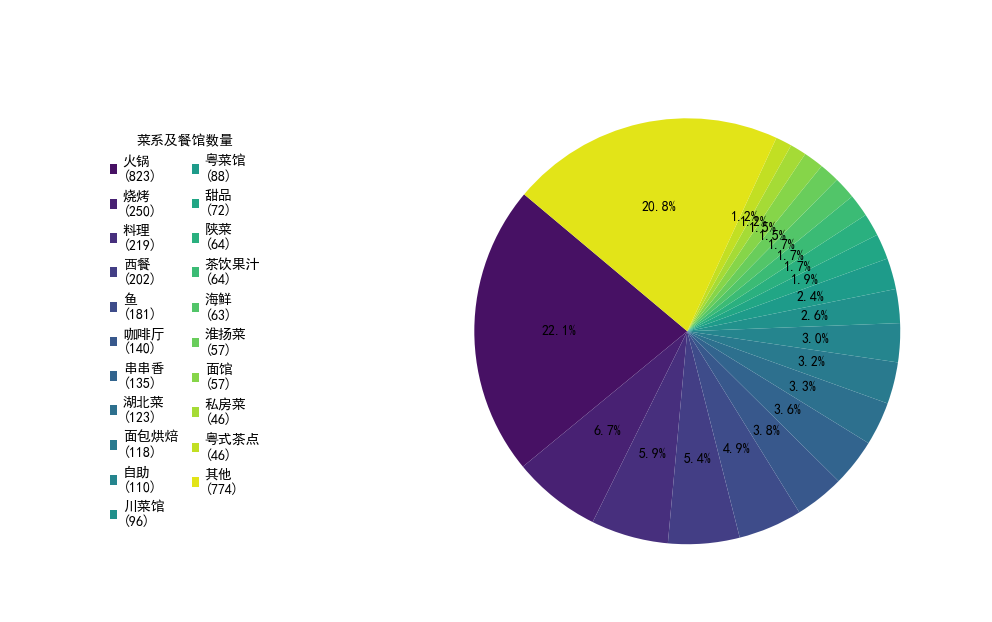
continue

city\_data = combined\_data[combined\_data['City'] == city]

plot\_cuisine\_pie\_chart(city\_data, city, threshold=1.3) # 使用新的阈值1%

**4、实验结论**

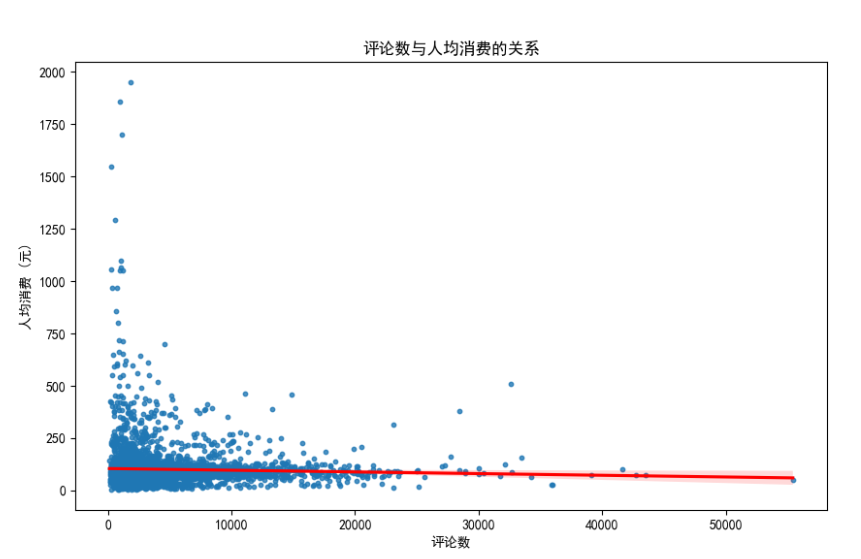
**4.1不同菜系占比分析**

****

**图1 不同菜系占比饼图**

该图表汇总了中国五大热门城市（成都、广州、武汉、西安、南京）的餐馆类型及其数量分布。我们可以看出饮食习惯的变化，其中对传统与创新融合的需求，火锅的高占比反映了人们对热辣口味的喜爱，同时也体现了中国传统美食在现代化进程中的持续流行。随着社会经济的发展，火锅店不仅保留了传统的烹饪方式，还不断创新菜品和服务模式，如自助式小火锅、特色蘸料等。烧烤虽然也受到欢迎，但其排名低于火锅，可能是因为健康饮食观念逐渐深入人心。现代消费者更加关注食物的安全性和营养价值，这从西餐、甜品、面包烘焙等较健康的餐饮选择的增长可以看出。在多元文化影响下的饮食多样化，料理（日式、韩式等）、西餐的比例表明，随着国际交流的加深，外国菜系在中国也获得了广泛的接受度。年轻一代更愿意尝试不同国家的美食，促进了餐饮市场的多元化。快节奏生活下的便捷需求，咖啡厅、茶饮果汁等轻餐饮类别的增长

**4.2 OLS回归模型分析评论数与人均消费的关系**



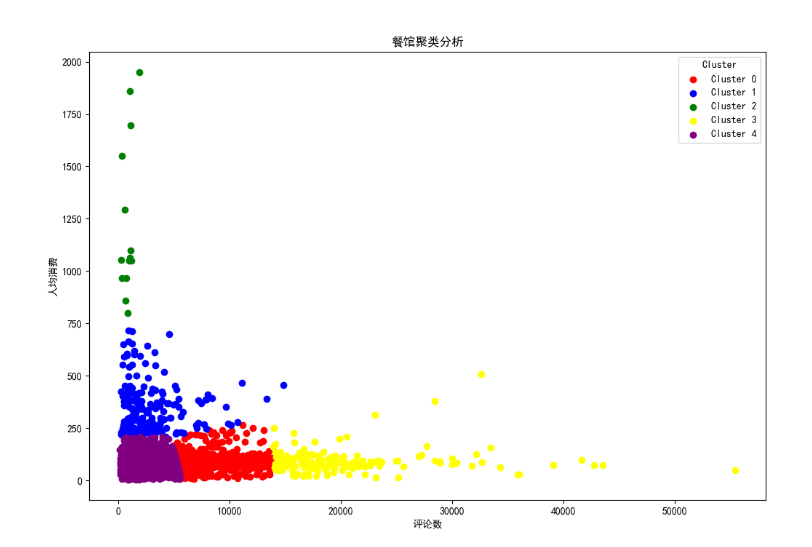
**图2 评论数与人均消费的关系**

散点分布：在评论数较少时（0-10000），人均消费呈现出较大的波动，从低到高都有分布。随着评论数的增加，人均消费逐渐趋于稳定，且大部分餐馆的人均消费集中在较低的区间（0-500元）。

线性回归线：从图中可以看出，线性回归线的斜率接近于0，这表明评论数与人均消费之间的关系较弱。回归线在评论数较高时几乎水平，说明随着评论数的增加，人均消费的变化不大。当评论数较少时，人均消费的波动较大，可能是因为这些餐馆的评价样本较少，导致人均消费的不确定性较高。当评论数较多时，人均消费趋于稳定，且大部分餐馆的人均消费集中在较低的区间（0-500元）。这表明评论数较多的餐馆通常价格较为亲民，且受到广泛欢迎。

通过线性回归分析，我们可以得出以下结论：评论数与人均消费的关系较弱：线性回归线的斜率接近于0，说明评论数与人均消费之间没有明显的线性关系。评论数较少时，人均消费波动较大：这可能是由于样本较少导致的不确定性。评论数较多时，人均消费趋于稳定：大部分餐馆的人均消费集中在较低的区间，说明评论数较多的餐馆通常价格较为亲民。

**4.3 KMEANS聚类分析**



**图3 聚类分析结果**

图片展示了一个散点图，横轴表示评论数，纵轴表示人均消费。数据点被分为5个不同的簇（Cluster 0, Cluster 1, Cluster 2, Cluster 3, Cluster 4），每个簇用不同的颜色表示。

Cluster 0 (红色)：这些餐馆的评论数较少，但人均消费相对较低。可能是小型餐馆或新开业的餐馆，尚未积累大量评论。Cluster 1 (蓝色)：这些餐馆的评论数较少，但人均消费较高。可能是高档餐厅或特色餐厅，虽然评论不多，但价格较高。Cluster 2 (绿色)：这些餐馆的评论数较多，且人均消费较高。可能是受欢迎的高档餐厅，既有较高的评价也有较高的价格。Cluster 3 (黄色)：这些餐馆的评论数较多，但人均消费较低。可能是大众化的餐馆，受到广泛欢迎，但价格亲民。Cluster 4 (紫色)：这些餐馆的评论数较少，且人均消费较低。可能是小型餐馆或新开业的餐馆，价格较低但尚未积累大量评论。

Cluster 0 和 Cluster 4 包括小型餐馆或新开业的餐馆，这些餐馆评论数较少且价格较低。这类餐馆通常满足日常用餐需求，如快餐和小吃，成本控制较好，价格亲民。在经济不景气时期，它们受到更多关注，因为消费者倾向于选择性价比高的餐饮选项。随着经济发展和生活水平提高，这类餐馆可能会面临更大的竞争压力。对于追求实惠和便利的消费者来说，这类餐馆可以提高他们的生活满意度，但长期依赖可能会影响生活质量。

Cluster 1 主要由高档餐厅或特色餐厅构成，虽然评论不多，但价格较高。这类餐馆提供高品质的服务和独特的就餐体验，吸引高收入人群及游客。在经济繁荣时期，这类餐馆会吸引更多顾客，特别是在旅游城市或文化中心更为常见。随着消费升级，这类餐馆的需求可能会增加，显著提升消费者的幸福感，但消费频率较低，主要作为特殊场合的选择。

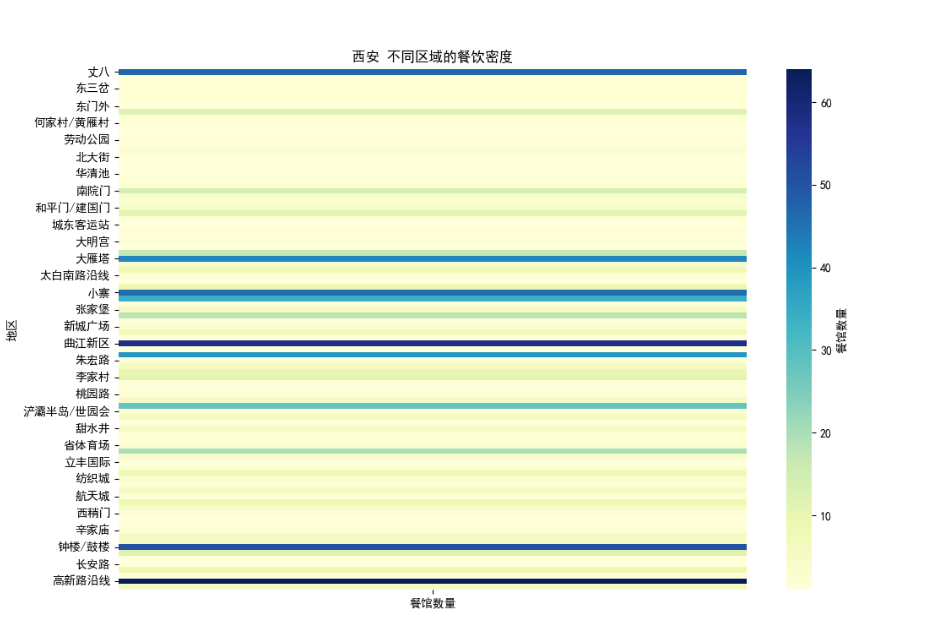
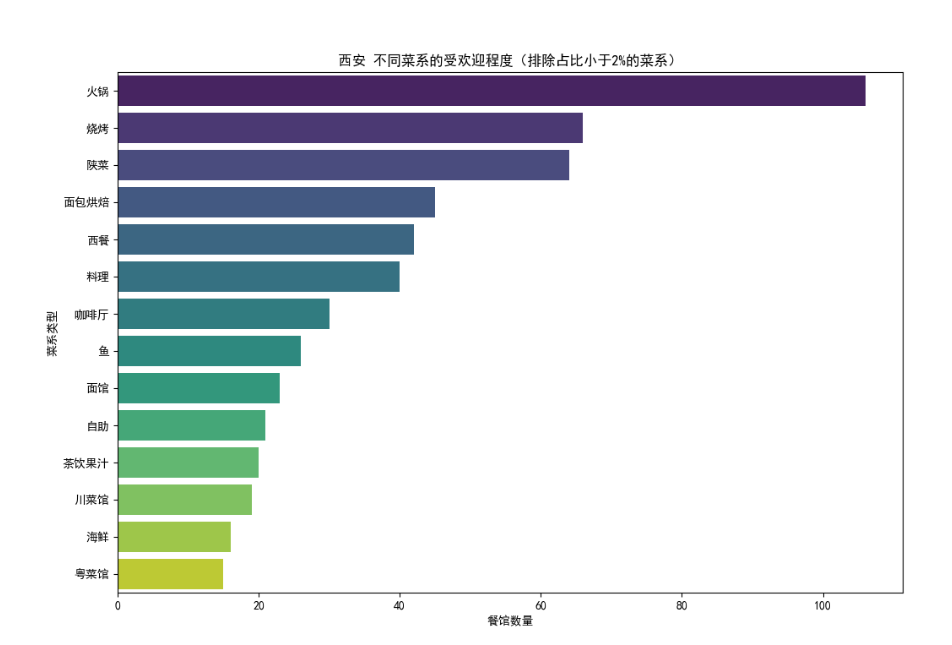
Cluster 2 是受欢迎的高档餐厅，既有较高的评价也有较高的价格。这类餐馆已经积累了大量正面评价，成为城市中的标志性餐饮场所，具有较高的品牌价值。它们提供精致的菜肴和独特的就餐体验，满足特定消费群体的需求。随着经济水平的提高和消费升级，这类餐馆的需求持续增长，并成为社交活动的重要场所，增强社区凝聚力，显著提升消费者的幸福感。

Cluster 3 是大众化的餐馆，评论数较多且价格亲民。这类餐馆满足日常用餐需求，提供性价比较高的餐饮服务。它们在经济不景气时期特别受欢迎，因为消费者倾向于选择性价比高的选项。随着健康意识的提升，这类餐馆需要不断创新菜品以吸引顾客。对于追求实惠和便利的消费者来说，这类餐馆提高了他们的生活满意度，但长期依赖可能影响生活质量。

综上所述，不同类型的餐馆在社会经济背景、饮食习惯、恩格尔系数以及居民幸福感等方面表现出不同的特点。随着经济和社会的发展，各类餐馆都需要不断创新和调整策略以适应市场需求，从而更好地服务于消费者并提升整体的社会福祉。

**4.4各城市具体情况**

**4.4.1西安**

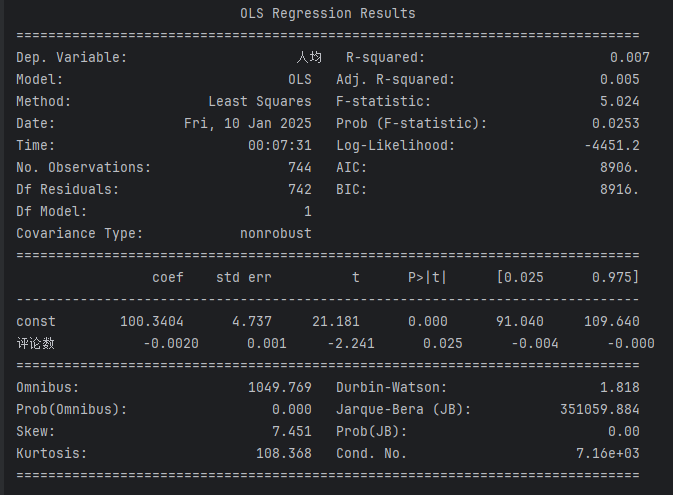
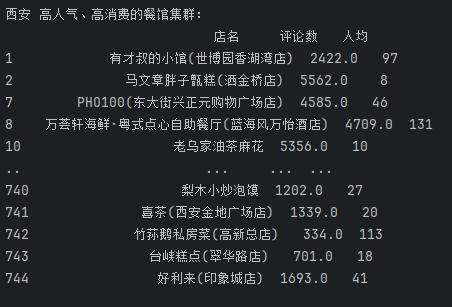


**图4 西安菜系受欢迎程度及分布区域热力图**

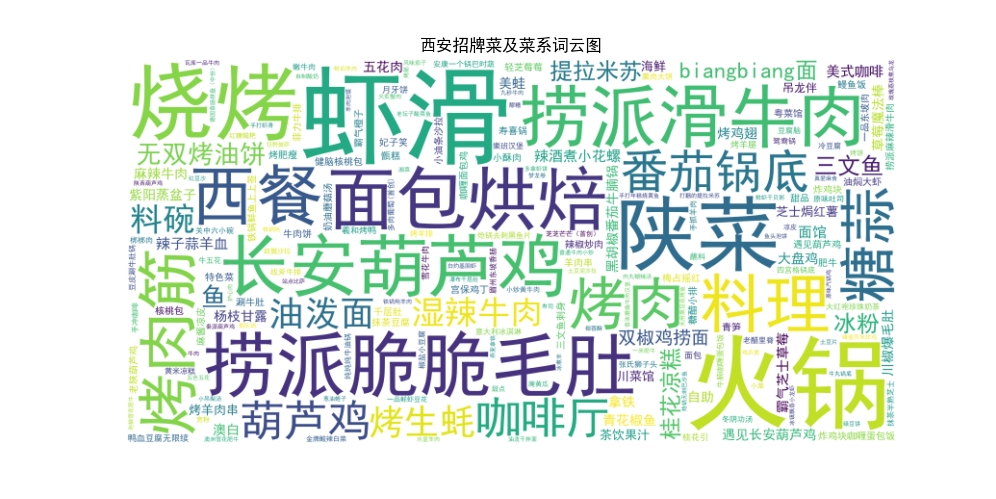
西安是中国历史上十三个朝代的古都，其饮食文化融合了西北地区的豪放与中原文化的典雅。肉夹馍、羊肉泡馍、凉皮等传统小吃体现了这座城市深厚的历史积淀。西安的美食注重原汁原味，讲究火候和调料的搭配。此外，西安还保留了许多传统的宴席形式，如长安八大碗，展示了古代宫廷宴席的风采。

火锅和烧烤是西安最受欢迎的菜系，这与当地饮食文化密切相关。陕菜作为本地特色菜系，也有较高的受欢迎程度。面包烘焙和西餐在西安也有一定的市场，但不如传统菜系受欢迎。粤菜馆和海鲜的受欢迎程度最低，餐馆数量最少。

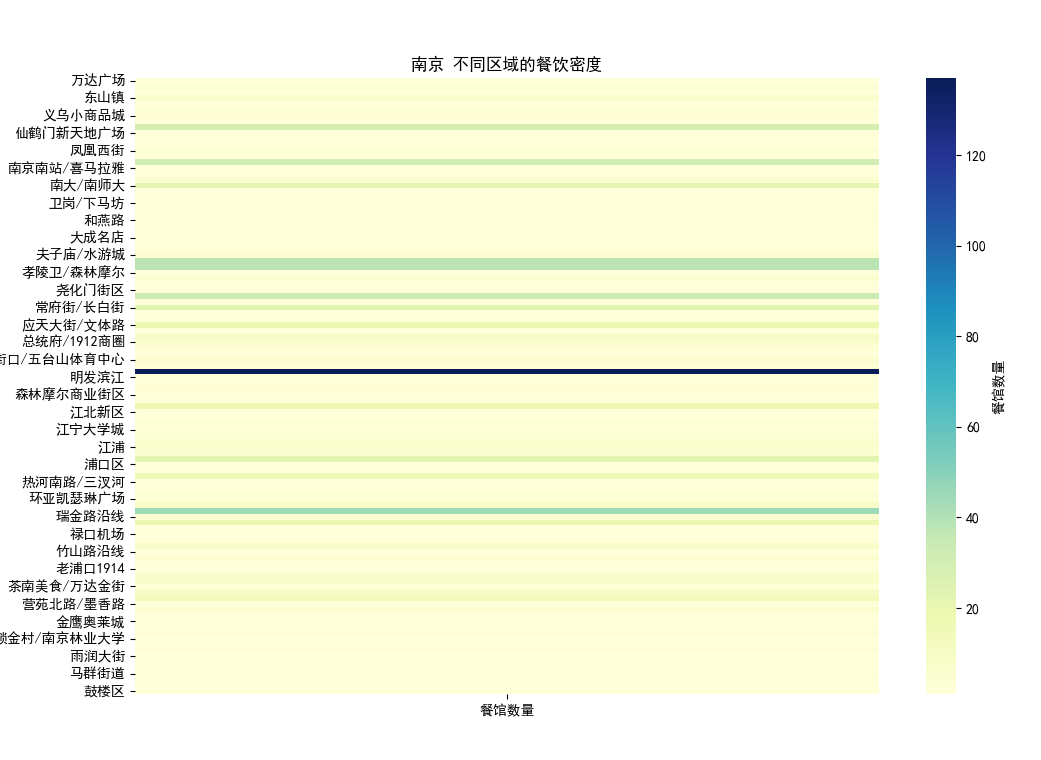
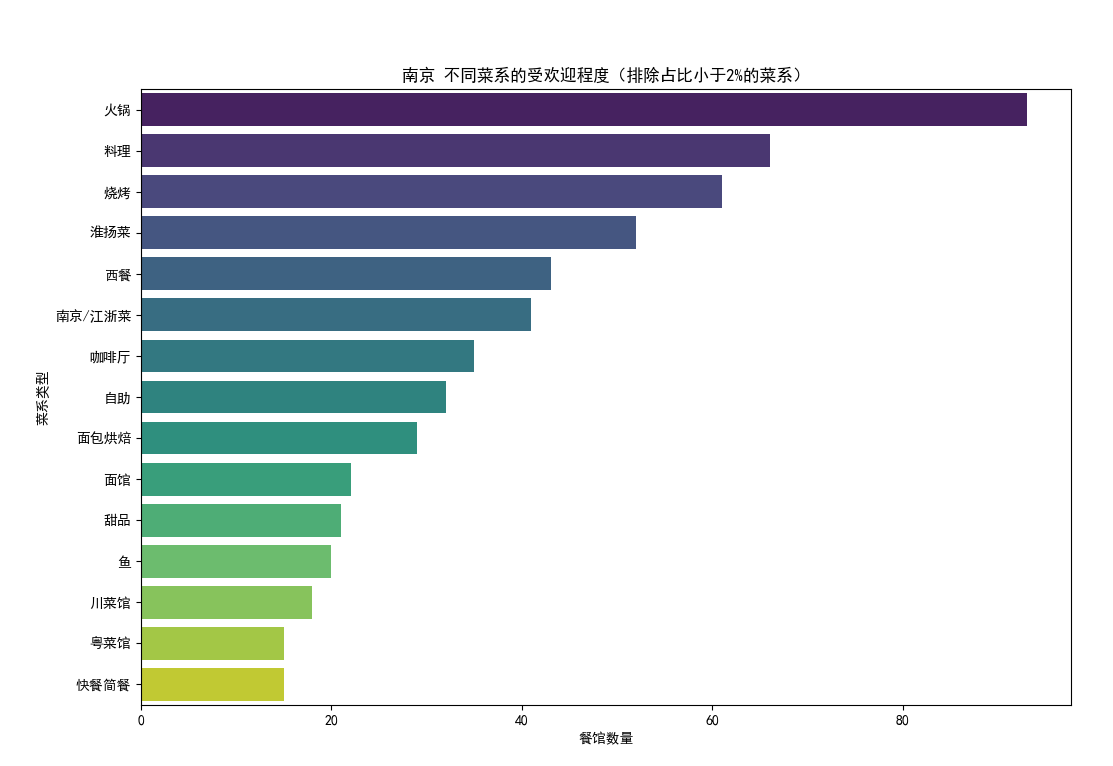
钟楼，鼓楼，小寨商圈（包括赛格国际购物中心），大雁塔景区及曲江新区，高新区（丈八、高新路沿线等），南门/书院门历史文化街区，北院门回坊美食街，大明宫国家遗址公园周边这些区域的餐饮密度较高，说明这些地方是餐饮业较为集中的区域。



**图5 KMEANS聚类及OLS回归结果**

**图6 西安招牌菜及菜系词云图**

**4.4.2南京**

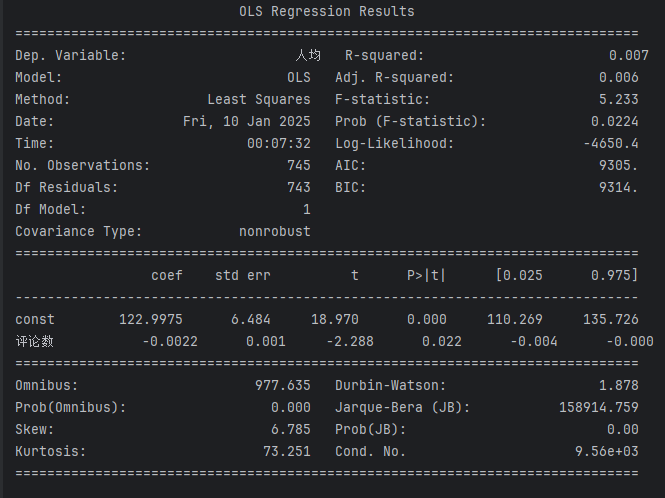
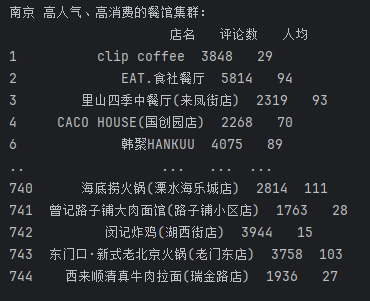


**图7 南京菜系受欢迎程度及分布区域热力图**

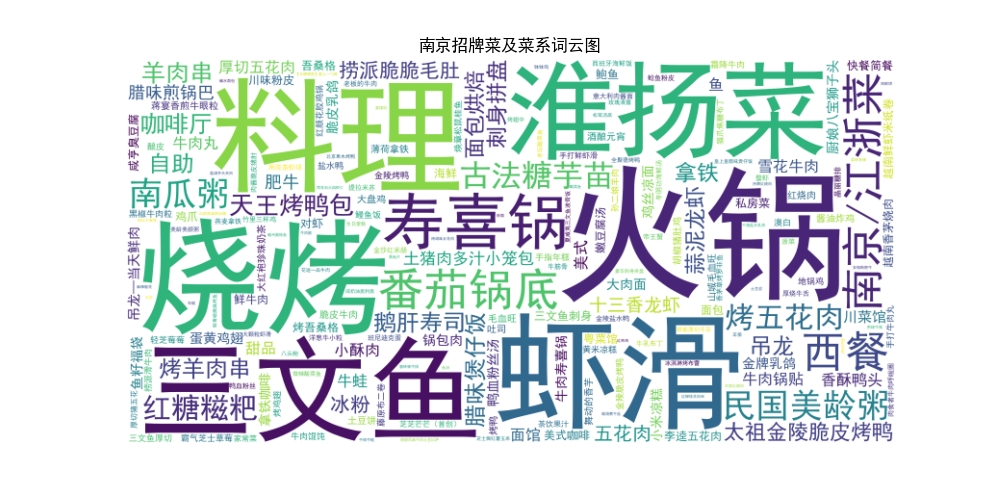
南京位于长江下游地区，属于淮扬菜系的一部分。这里的菜肴口味偏清淡，讲究色香味形俱佳。盐水鸭是南京最著名的特产之一，选用当地特有的麻鸭制作而成，皮白肉嫩，咸香适中。此外，南京还有许多其他特色美食，如鸭血粉丝汤、桂花糖芋苗等，每一道都散发着江南水乡的独特韵味。南京的美食不仅仅是一种味觉享受，更承载着这座城市的历史记忆和人文情怀。

火锅、料理和烧烤是南京最受欢迎的菜系，这与当地饮食文化密切相关。淮扬菜作为本地特色菜系，也有较高的受欢迎程度。南京/江浙菜在南京也有一定的市场，但不如传统菜系受欢迎。咖啡厅、自助、面包烘焙、面馆和甜品在南京也有较高的受欢迎程度。快餐简餐的受欢迎程度最低，餐馆数量最少。

万达广场夫子庙/水游城，新街口商圈（包括总统府/1912商圈），仙林大学城，河西新城（万达广场、金鹰奥莱城等），鼓楼区（环亚凯瑟琳广场、瑞金路沿线等），老浦口1914这些区域的餐饮密度较高，它们或是因为地理位置优越、或是因为特定人群聚集而形成了独特的餐饮市场。

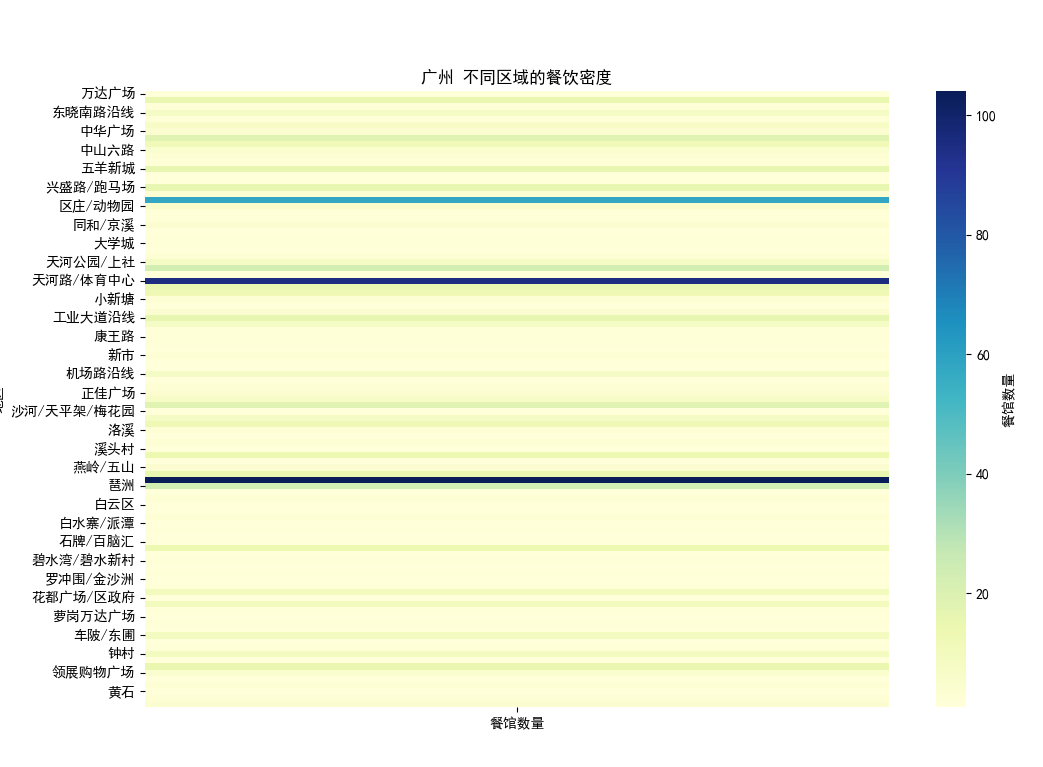
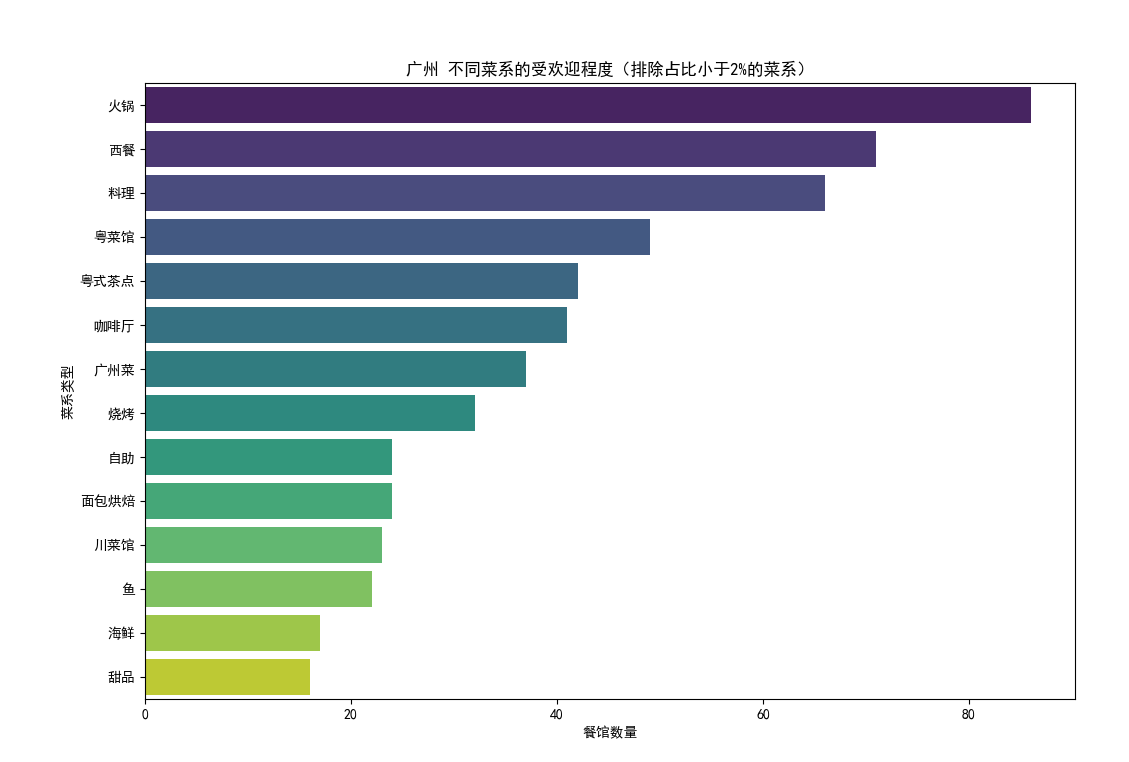


**图8 KMEANS聚类及OLS回归结果**



**图9 南京招牌菜及菜系词云图**

**4.4.3广州**

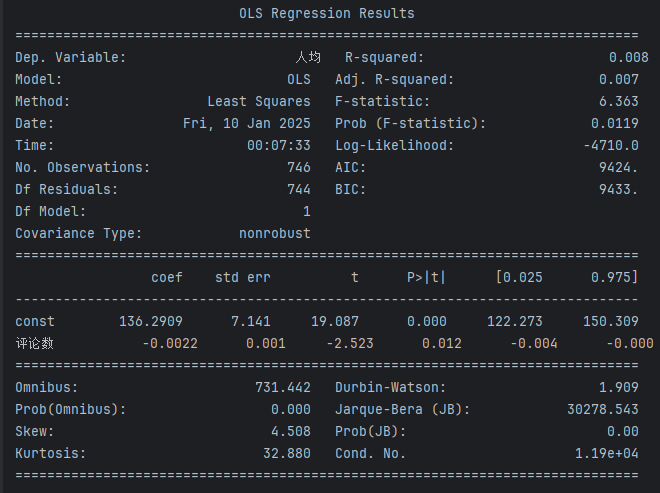
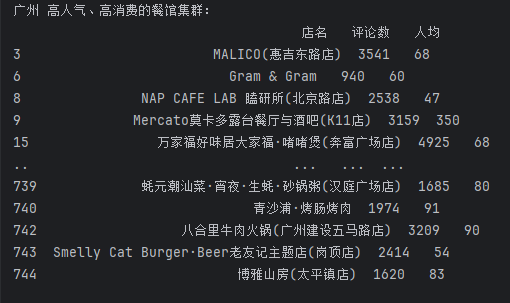


**图10 广州菜系受欢迎程度及分布区域热力图**

广州作为广东省的省会，拥有悠久的历史和深厚的文化底蕴。粤菜以其选材广泛、烹饪精细而闻名于世。早茶文化在广州尤为盛行，点心种类繁多，包括虾饺、叉烧包、蛋挞等，每一道都精致美味。除了早茶，广州还有许多特色菜肴，如白切鸡、煲仔饭、艇仔粥等，展现了广东人对食材原汁原味的追求。此外，广州还是一个国际化大都市，世界各地的美食在这里都能找到踪迹。

火锅和西餐：在广州非常受欢迎，这与当地饮食文化密切相关。火锅以其多样性和社交属性受到广泛喜爱，而西餐则反映了广州国际化的一面。粤菜馆和粤式茶点：作为本地特色菜系，粤菜和茶点在广州市民中有着深厚的根基，不仅在日常生活中占据重要地位，也是游客体验广州文化的首选。咖啡厅和广州菜：咖啡厅在广州也非常受欢迎，反映出城市快节奏的生活方式和对休闲生活的追求。广州菜则以其精致和多样化的特点吸引着众多食客。海鲜和甜品：广州靠近海洋，海鲜自然成为重要的美食选择。甜品则是广州人日常生活中不可或缺的一部分，尤其是各种传统的糖水和糕点。

兴盛路/跑马场，天河公园/上社，体育中心，小新塘，工业大道沿线 ，康王路，新市较为集中。

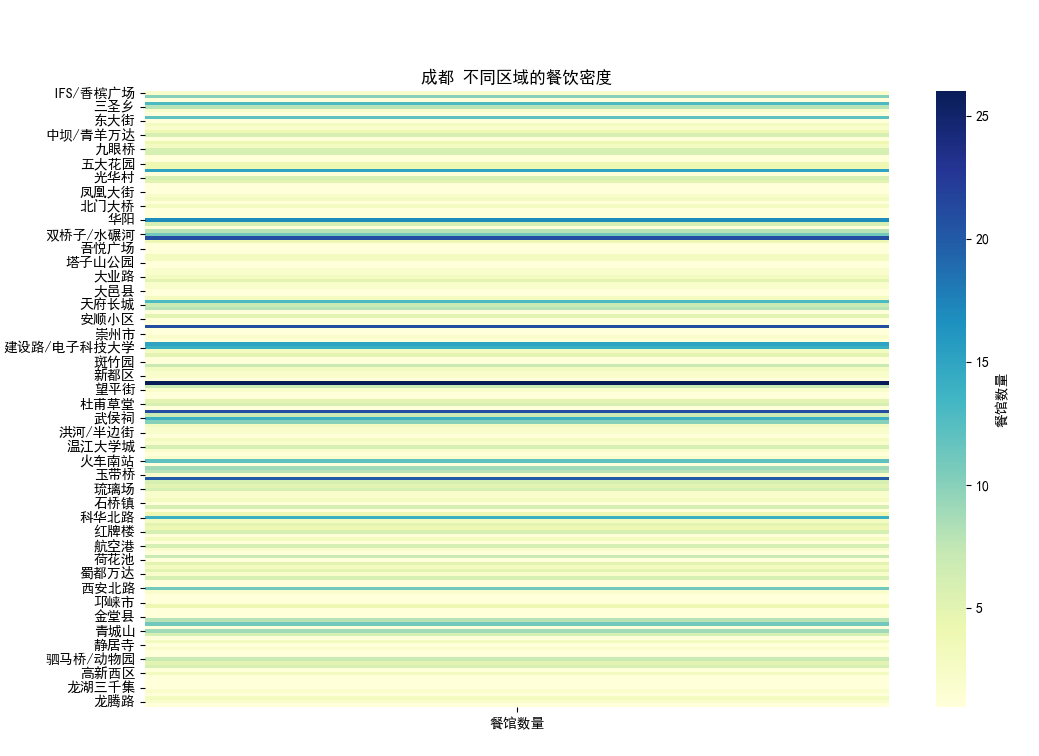
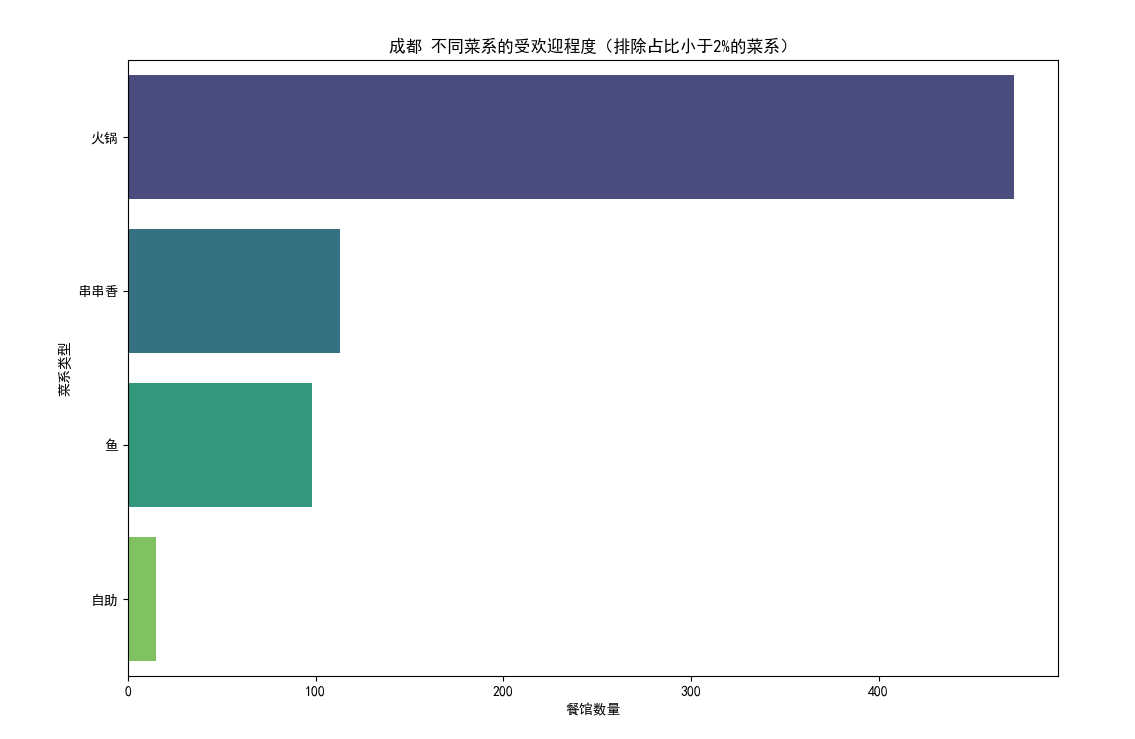


**图11 KMEANS聚类及OLS回归结果**



**图12 广州招牌菜及菜系词云图**

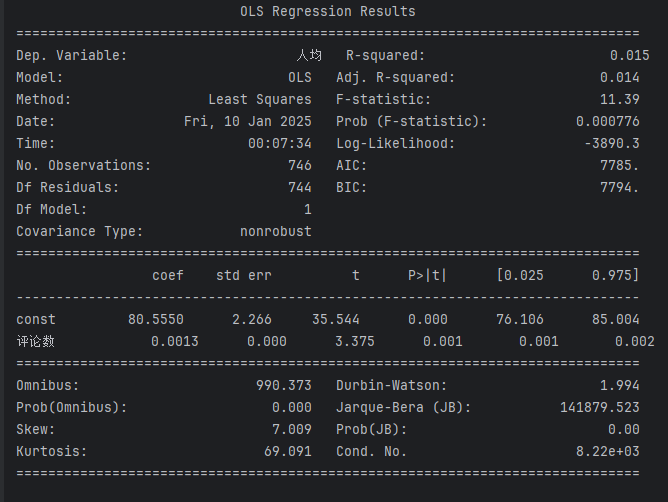
**4.4.4成都**



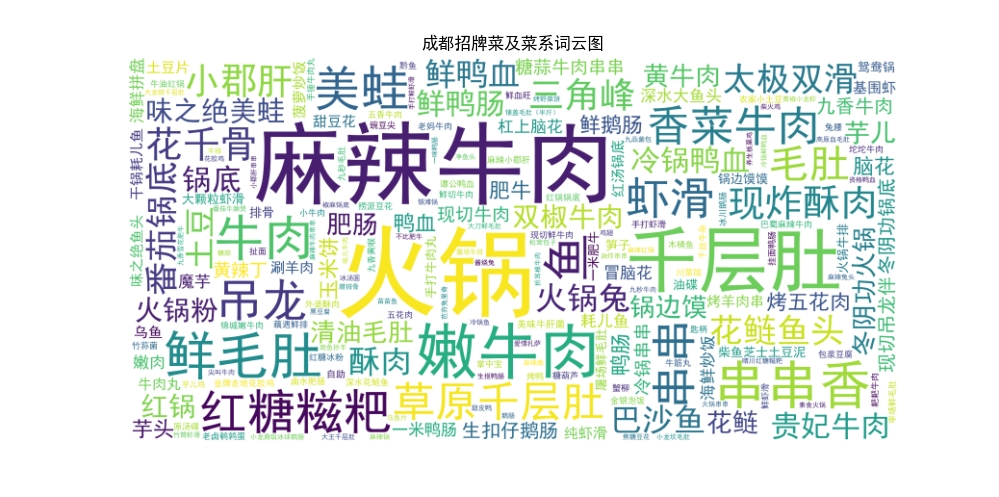
**图13 成都菜系受欢迎程度及分布区域热力图**

成都菜系成分相对其他城市较为单一，由于其地理环境，气候因素，传统川味火锅深厚的历史底蕴，而现代成都火锅将地方特色与创新结合还有火锅较强的包容性，使得火锅断崖式领先。串串香和鱼也较为受欢迎。

分布区域方面有很多地域，商圈餐馆密集，百花齐放。高密度区域主要集中在市中心及热门商圈、交通枢纽和旅游景点周边。例如，IFS/香槟广场作为商业娱乐中心，汇聚了大量的高档餐厅和小吃摊位；九眼桥以其夜生活闻名，周围遍布各类酒吧和餐馆；武侯祠和宽窄巷子等历史文化景点周边，不仅有传统川菜馆，还有许多特色小吃店，满足游客的需求；此外，像中坝/青羊万达、五大花园、塔子山公园、锦江/华兴街等地，也因为其便利的交通和浓厚的生活气息，成为了餐饮业高度发达的区域。

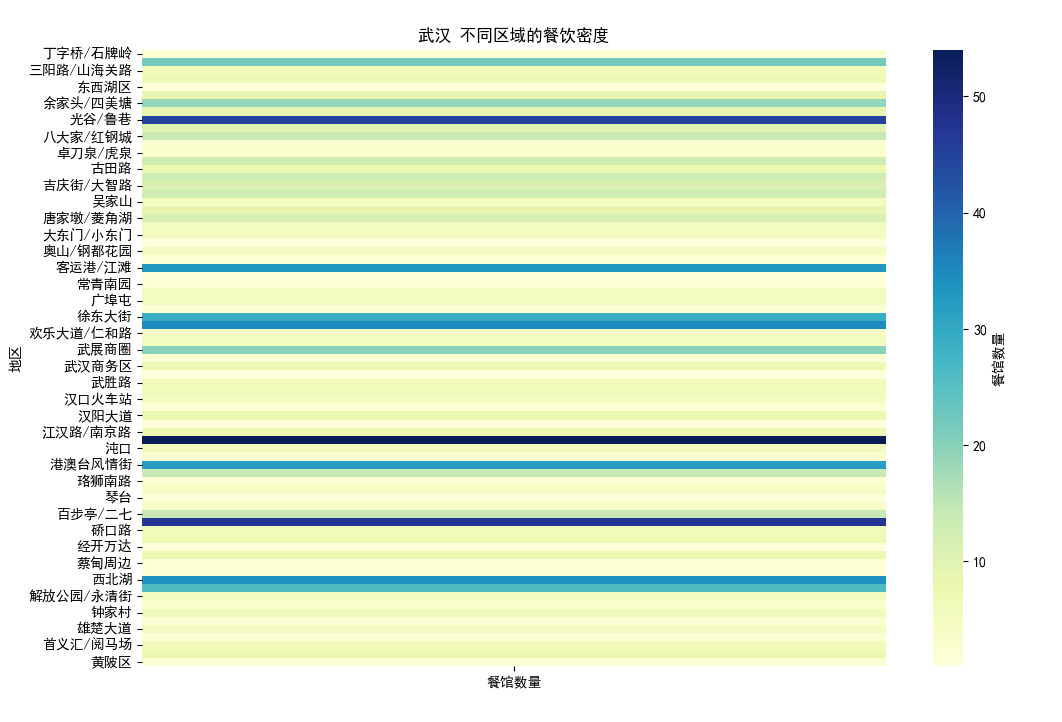
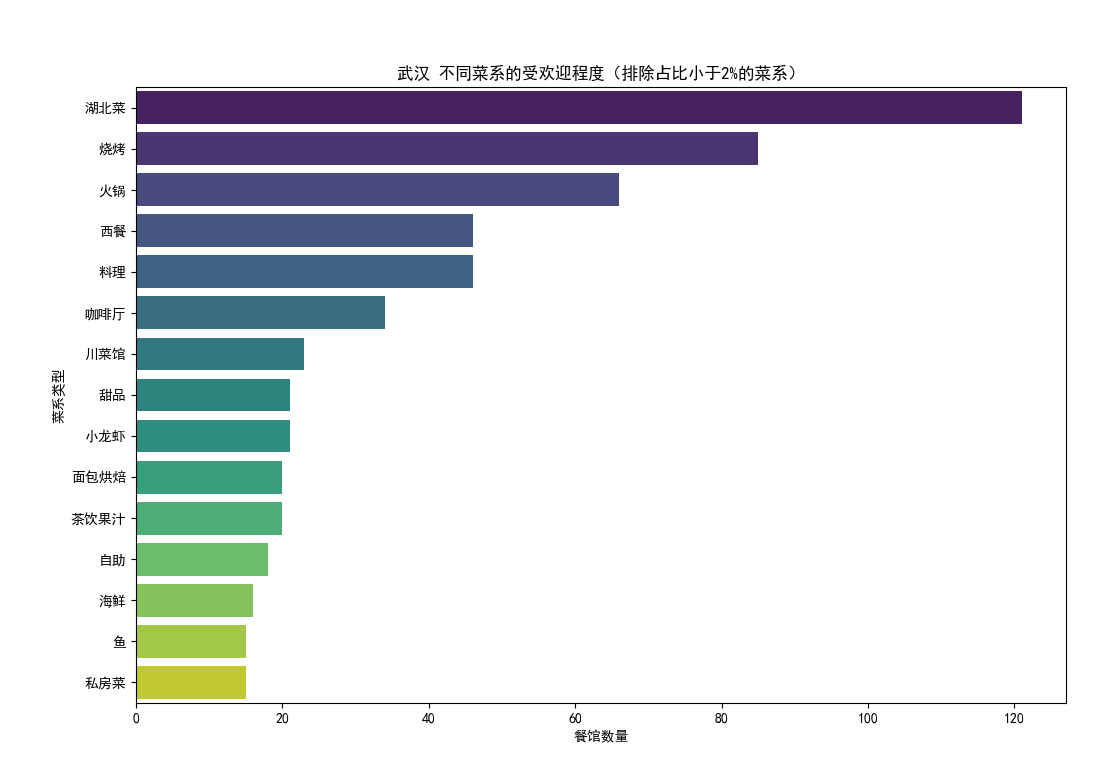
 

**图14 KMEANS聚类及OLS回归结果**



**图15 成都招牌菜及菜系词云图**

**4.4.5武汉**

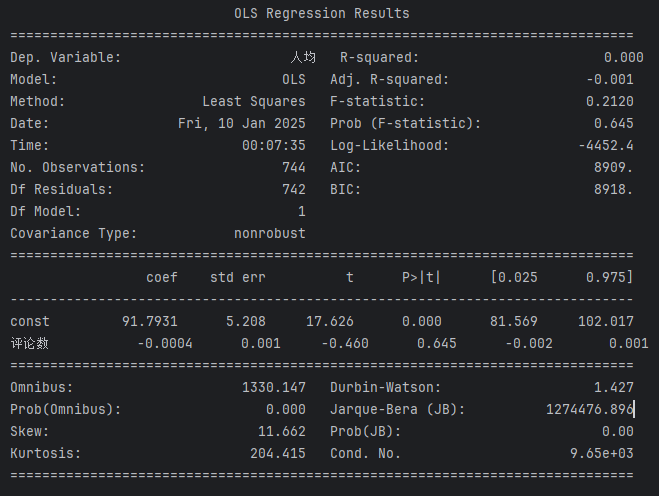
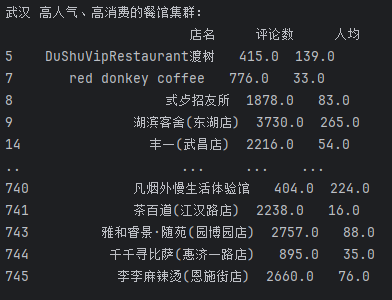


**图16 武汉菜系受欢迎程度及分布区域热力图**

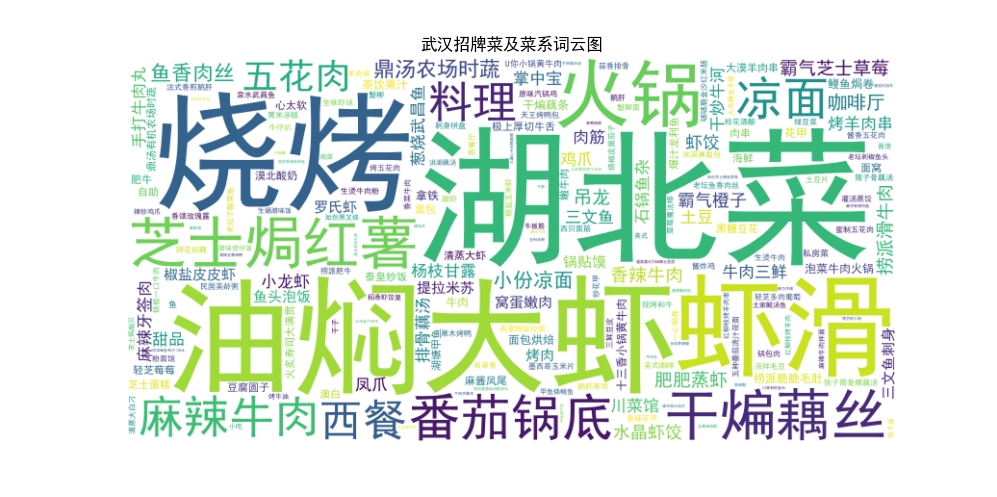
武汉地处长江中游，三镇鼎立，是一座充满活力的大都市。武汉的饮食文化同样丰富多彩，热干面作为城市的名片，几乎遍布大街小巷，芝麻酱浓郁醇厚，面条筋道有嚼劲。除此之外，豆皮、周黑鸭等小吃也广受好评。武汉的早餐文化尤其值得一提，当地人称之为“过早”，形式多样，从街头小吃到餐馆正餐应有尽有。武汉的美食充满了市井气息，反映出这座城市开放包容的性格特点。

湖北菜：作为本地特色菜系，湖北菜在武汉非常受欢迎。其独特的风味和丰富的菜品深受市民喜爱。烧烤：烧烤以其多样性和社交属性受到广泛喜爱，特别是在夏季，户外烧烤成为一种流行的休闲方式。火锅：火锅在武汉也非常流行，尤其是在冬季，人们喜欢通过吃火锅来驱寒暖身。小龙虾：作为武汉的特色美食之一，小龙虾餐馆遍布全市，尤其在夏季，小龙虾节更是吸引了大量食客。西餐和咖啡厅：反映了武汉国际化的一面，满足了年轻人和白领的需求。

丁字桥，石牌岭，三阳路，武昌火车站，江汉路，南京路 ，解放公园餐饮业聚集



**图17 KMEANS聚类及OLS回归结果**



**图18 武汉招牌菜及菜系词云图**

**4.5地域差异引起的饮食文化差异分析**

**4.5.1. 地理位置与气候**

**成都**：位于四川盆地，气候湿润寒冷。这种气候条件使得麻辣口味的食物特别受欢迎，因为辛辣食物有助于驱寒和促进血液循环。

**广州**：地处亚热带季风气候区，夏季炎热多雨，冬季温暖湿润。因此，清淡、滋补的粤菜和茶点成为主流，以适应湿热的环境。

**西安**：位于内陆地区，四季分明，冬季寒冷干燥。这里的饮食偏向面食和炖煮类菜肴，如羊肉泡馍、肉夹馍等，这些食物能够提供足够的热量来抵御寒冷。

**南京**：属于长江中下游平原，四季分明，湿度较高。南京菜讲究原汁原味，注重食材的新鲜度，同时也有一定的甜度，反映了江南水乡的特点。

**武汉**：位于长江中游，气候温和湿润，四季分明。武汉的饮食融合了南北风味，既有辣味又有清淡的选择，特别是著名的热干面和小龙虾。

**4.5.2. 文化背景与历史传统**

**成都**：作为川菜的发源地之一，成都有着深厚的美食文化底蕴，川菜以其麻、辣、鲜、香著称，影响深远。

**广州**：广府文化的重要代表，粤菜是中国四大菜系之一，以其精细刀工、独特调味和多样化的烹饪技法闻名。

**西安**：古都长安的历史赋予了这座城市丰富的饮食文化，许多传统小吃如凉皮、肉夹馍等至今仍深受人们喜爱。

**南京**：六朝古都，南京菜融合了南北方特色，形成了独特的淮扬菜系，其菜品精致、味道醇厚。

**武汉**：九省通衢之地，多元文化的交汇使得武汉饮食既保留了本地特色又吸收了外来元素，形成了独具一格的“汉味”。

**4.5.3. 现代化进程与生活方式**

**成都**：近年来，随着经济快速发展，越来越多的年轻人涌入这座城市，他们追求时尚的生活方式，带动了火锅、串串香等休闲餐饮形式的流行。

**广州**：国际化大都市，西餐、咖啡厅等现代餐饮形式广泛存在，反映了年轻人对新鲜事物的接受度较高。

**西安**：旅游业的发展促进了当地餐饮业的繁荣，传统的陕西菜与新兴的创意餐厅并存，满足了不同游客的需求。

**南京**：作为长三角区域的重要城市，南京不仅保留了传统文化，也积极引入国际化的餐饮概念，如日料、韩料等，在年轻群体中颇受欢迎。

**武汉**：交通枢纽地位使得武汉成为全国交流的重要节点，这为本地餐饮带来了更多的可能性，例如，小龙虾节等活动吸引了大量外地游客。

**4.5.4. 饮食习惯与健康意识**

**成都**：由于气候原因，市民更倾向于选择热辣的食物来驱寒暖身；同时，随着健康意识的提高，轻食、素食等健康餐饮也开始受到关注。

**广州**：强调食材的新鲜度和营养均衡，粤菜中的汤品和粥类就是很好的例子，它们既能补充水分又能调节身体机能。

**西安**：面食为主，但近年来也开始注重膳食结构的调整，增加蔬菜水果的比例，倡导健康饮食理念。

**南京**：重视食材的选择和烹饪方法，避免过度加工，确保食物的天然风味和营养价值。

**武汉**：虽然偏爱重口味的食物，但也逐渐意识到健康的饮食习惯的重要性，开始推广低油少盐的烹饪方式。

**4.5.5结论**

通过上述对比可以看出，五个城市的饮食习惯各具特色，这些差异主要源于地理位置、气候条件、历史文化以及现代社会进程等因素的共同作用。每个城市都有自己独特的美食文化和代表性菜品，同时也都在不断适应时代变化，引进新的餐饮形式，以满足日益多样化的消费需求。

**5、总结与心得体会**

完成本次课程设计，不仅让我对数据处理和可视化有了更深刻的理解，还锻炼了我的实际操作能力和解决问题的能力。通过对成都、广州、西安、南京和武汉这五个城市的餐馆数据进行聚类分析，我学会了如何从大量数据中提取有价值的信息，并通过可视化手段展示出来。

在整个项目中，我也遇到了不少问题和困难。在项目的初期阶段，我遇到了第一个挑战——数据的获取和清理。其中最大的挑战是如何确保数据的质量和一致性。由于原始数据存在大量的噪声和异常值，我在清洗和标准化方面花费了大量的时间和精力。不同来源的数据格式不一，我必须学习并应用多种数据清洗技术，如正则表达式清理非数字字符、缺失值处理等。这一过程虽然繁琐，但也极大地提升了我的数据处理技能。

接下来是选择合适的算法进行聚类分析。起初，我对K-means算法并不熟悉，但通过查阅文献和在线资源，我逐渐理解了其原理和应用场景。在实践中，我学会了如何调整参数以获得最佳的聚类效果。此外，在进行聚类分析时，确定合适的簇数也是一个难题。不过，通过反复试验和不断调整，最终还是找到了满意的解决方案。为了更好地展示分析结果，我选择了Matplotlib和Seaborn这两个强大的可视化库。在这个过程中，我学会了如何创建图表，包括散点图、柱状图和热力图等，并根据需求定制颜色、标签和其他视觉元素。特别是当面对复杂的数据集时，合理选择可视化方式对于有效传达信息至关重要。这次经历大大增强了我的数据可视化能力。

除了技术层面的学习，我还尝试将数据分析与社会经济学知识相结合。例如，通过研究恩格尔系数的变化趋势，我发现它对不同类型的餐馆有着显著影响。这种跨学科的思考方式拓宽了我的视野，使我能从更宏观的角度看待问题，也为未来的研究提供了新的思路。

通过这次项目，我认识到数据可视化不仅仅是技术和工具的应用，更重要的是如何从中挖掘出有意义的故事。未来，我希望能够继续探索更多的数据分析方法和技术，不断提升自己的专业素养。同时，我也期待将所学知识应用于实际生活中，为解决现实问题贡献一份力量。