

一. 课程项目 简介

1.1 小组成员与分工

姓名	学号	分工
侯羽飞	2024E8016082032	数据清洗、基于聚类的梯度提升回归预测
刘星雨	202428016029023	数据清洗、简单单变量回归模型、多元输出回归
郑卓云	202428016029025	数据清洗、基于 LSTM 分城市进行人口预测
唐媛	202428013229094	数据清洗、GBRT 回归模型、集成多模型对预测结果进行优化

一 . 课程项目 简介..... 1

1.1 小组成员与分工..... 1

1.2 项目 任务简介..... 2

二 . 研究方法..... 2

2.1 数据清洗..... 3

2.2 特征选择..... 3

2.3 模型选择..... 3

2.3.1 线性回归模型..... 3

2.3.2 岭回归..... 4

2.3.3 LSTM..... 4

2.3.4 GBDT.....	5
<b>三 . 数据预处理.....</b>	<b>5</b>
3.1 缺失值填充.....	6
3.2 异常值处理.....	6
3.3 数据归约.....	7
<b>四 . 模型搭建与实验.....</b>	<b>8</b>
4.1 简单单变量回归模型.....	8
4.2 多元输出线性回归.....	10
4.3 基于聚类的梯度提升回归预测.....	13
4.3.1 聚类.....	13
4.3.2 回归预测.....	14
4.4 基于 LSTM 的预测.....	15
<b>五 . 实验结果.....</b>	<b>15</b>
<b>六 . 思考与展望.....</b>	<b>16</b>

## 1.2 项目 任务简介

我们组选取的大作业课题源于 DataFountain 上的比赛，项目 名称为 ‘城市人口 分析与预测’，任务是根据给定的特征实现人口 分析和预测的任务。人口 规模预测对于社会经济发展和城市规划建设具有重要的意义，它涉及到多个方面，可以帮助规划城市发展、优化资源配置、支持经济发展决策、改善社会服务、提供

政策制定参考等。

项目 给出了多个城市近几年的人口 数据，包括年龄、职业、收入等特征，如表一所示。通过对这些给定的特征分析每个城市的特点和规律，利用算法构建人口 预测模型，预测给定各个城市 2023 年的总人口 数。

文件名	文件内 容	字段
人口 规模.xlsx	城市人口 规模信息	pr_Population——常住人口 规模
		r_Population——户籍人口 规模
城镇化率.xlsx	城市的城镇化率	urbanizationRate——城镇化率
年龄结构.xlsx	城市人口 的年龄结 构	0-14——0-14 岁人口 规模
		15-64——城市 15-64 岁人口 规模
		65+——城市 65 岁以上人口 规模
就业信息.xlsx	城市的就业情况	unemploymentRate——失业率
		threeIndustriesEmployed——三次产业就业人数
生活水平.xlsx	城市人口 的生活水 平	disposableIncome——人均可支配收入
		towner_ConsumptionExpenditures 城镇居民消费 支出

## 二. 研究方法

由于给定数据集的数据存在异常点和缺失值，所以首先需要对数据进行数据清洗，以确保数据的有效性。其次，项目 给定了大量的特征，并不是所有的特征都是有效的，所以需要筛选较为重要的特征。随后，使用这些特征训练回归模型，以得到最终所需预测的人口 规模。最后，采用均方误差评估模型的预测精度，

MSE 为本赛题的评测参考标准，排行榜的分数计算依据为： $\text{score} = 1/(1+\text{MSE})$ 。

## 2.1 数据清洗

对于缺失值，当缺失值的比例非常低时，可以直接删除包含缺失值的记录或变量。对于连续变量，可以使用均值或中位数；对于离散变量，可以使用众数。同时，也可以根据其他非缺失的变量或观测来预测缺失值，常见的方法有回归插补法、K 近邻插补法、拉格朗日插补法等。

对于异常值，当数据近似服从正态分布时，可以使用  $n$  个标准差法检测异常值。或者采用箱线图进行判别。利用四分位数和四分位距来识别异常值。如果一个值小于  $QL-1.5IQR$  或大于  $QU+1.5IQR$ ，则被认为是异常值。

## 2.2 特征选择

如果两个特征之间的相关性较高，可能会引发多重共线性问题，导致模型稳定性变差。在这种情况下，可以在具有共线性的特征中选择一个保留，其余剔除。可以通过皮尔逊相关系数，筛选强相关的特征，选择其一进行删除。或者采用互信息法，计算每个特征与目标变量之间的互信息量，选择互信息量较大的特征，选择其一进行删除。

由于项目给出的特征较多，即特征维数较高，可能会导致大量冗余信息。可以采用主成分分析(PCA)，降低数据的维度、去除冗余信息、保留重要特征。

## 2.3 模型选择

### 2.3.1 线性回归模型

线性回归是利用数理统计中的回归分析，来确定两种或两种以上变量间相互依赖的定量关系的一种统计分析方法。通过假设因变量与自变量之间存在线性关系，通过最小化预测值与实际值之间的差值来拟合数据。

线性回归模型最常用的损失函数是均方误差，可以使用各种优化算法，如梯度下降、随机梯度下降、Adam 等。这些算法通过迭代地更新权重和截距的值，逐渐减小损失函数的值，直到达到一个可以接受的误差范围或者达到预设的迭代次数。

然而，线性回归假设自变量与因变量之间存在线性关系，但现实中的数据往往难以满足这一假设；对异常值敏感，可能导致模型不稳定；同时，当自变量数量较多时，容易出现多重共线性问题。

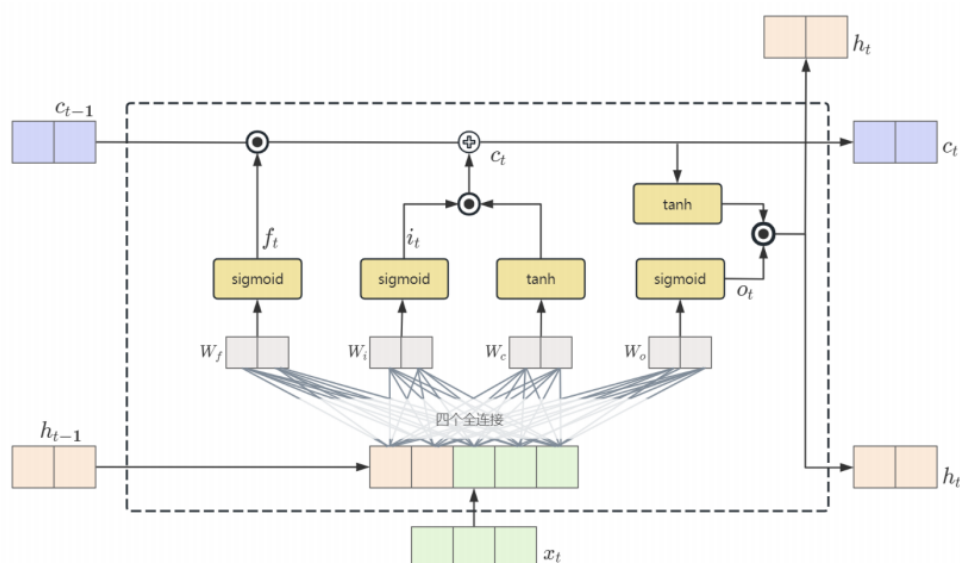
### 2.3.2 岭回归

岭回归，也称为 L2 正则化，是一种专门用于处理多重共线性(特征之间高度相关)问题的线性回归改进算法。通过在损失函数中添加一个正则化项，减少模型复杂度和过拟合的风险，同时提高模型的稳定性。

岭回归能够有效处理特征之间的高度相关性，提高模型的稳定性；通过正则化减少模型的复杂度，降低过拟合的风险；然而，它也需要选择合适的正则化参数，这需要依赖经验或者交叉验证。同时也不能自动选择重要的特征，需要手动调整正则化参数。

### 2.3.3 LSTM

LSTM 是一种特殊的循环神经网络(RNN)，旨在解决传统 RNN 在处理长序列数据时的梯度消失和梯度爆炸问题。它的设计灵感来源于人脑的记忆机制，通过引入“门”结构和“细胞状态”，LSTM 能够在长序列中更好地捕捉依赖关系。



LSTM 包含以下“门”结构。遗忘门决定了哪些信息从记忆单元中遗忘，它使用 sigmoid 激活函数，输出 0 到 1 之间的值，表示保留信息的比例。输入门决定了哪些新信息将被存储在记忆单元中。它包括两部分：sigmoid 激活函数用来决定更新的部分，tanh 激活函数来生成候选值。记忆单元是 LSTM 的核心，能够长时间保留信息。通过遗忘门和输入门的相互作用，记忆单元学习如何选择性地记住或忘记信息。输出门决定了下一个隐藏状态。通过使用 sigmoid 激活函数来决定记忆单元的哪些部分将输出，然后这个值与记忆单元的 tanh 激活的值相乘得到最终输出。

LSTM 能够有效处理特征之间的高度相关性，提高模型的稳定性；通过正则化减少模型的复杂度，降低过拟合的风险。LSTM 的参数估计也具有较好的解释性，可以用于统计推断。

### 2.3.4 GBDT

GBDT 回归模型是一种集成学习算法，通过迭代地添加多个弱学习器（通常是决策树）来构建一个强学习器，用于预测连续型目标变量。

GBDT 会用一个简单的模型（如常数模型）对所有样本做出初始预测。基于

当前模型的预测结果，计算每个样本的真实标签与预测值之间的梯度（对于回归问题通常是真实值减去预测值；对于分类问题，则使用损失函数的负梯度）。将这些残差作为新的目标变量，训练一个决策树来拟合这些残差。决策树的深度和节点数决定了模型的复杂度。将新训练的决策树加入到模型中，更新每个样本的预测值为原预测值加上新决策树的输出。重复上述过程，直到达到预设的迭代次数或满足停止条件。

GBDT 能够自然地评估特征的重要性，这对于特征选择和理解模型有重要价值。在处理高维稀疏数据时，也引入正则化、剪枝策略以及稀疏矩阵运算技术可以有效提升模型的效率和效果。

### 三. 数据预处理

题目提供的原始数据存在大量的缺失值，异常值和重复值，这些问题会导致分析的结果不准确，因此需要通过数据处理，提高数据的准确性，完整性和一致性。具体的处理流程如下图 3.1 所示。



图 3.1 数据预处理流程

#### 3.1 缺失值填充

首先，我们对各属性的缺失值进行了统计，将缺失较多的属性特征进行了删除，缺失值统计如图 3.2 所示。

图 3.2 属性缺失值统计

对于不同属性的缺失值, 我们采用了不同的缺失值填充方案。对于人口密度、居住人口、户籍人口、城镇化率、城镇居民人均收入等属性, 我们采用了线性回归的方式进行填充; 而对于从业人数, 第一、二、三产业就业人数等属性, 我们采用了均值填充的方法。

## 3.2 异常值处理

我们针对上述不同属性绘制了箱线图, 如图 3.3-3.5 所示。

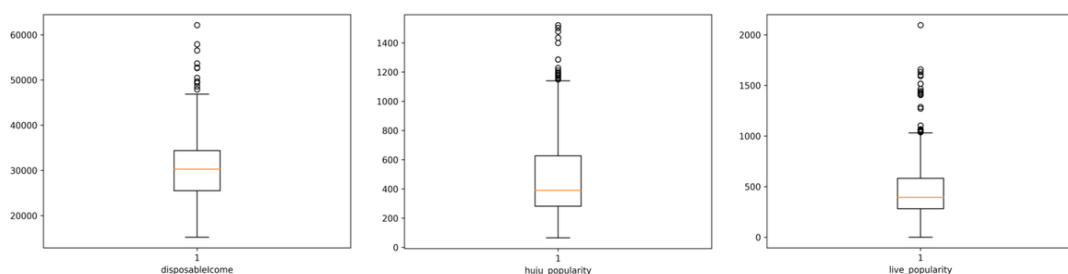


图 3.3 属性箱线图



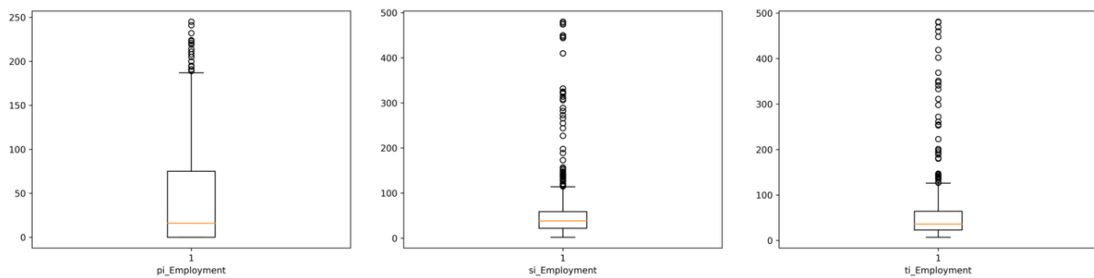


图 3.4 属性箱线图

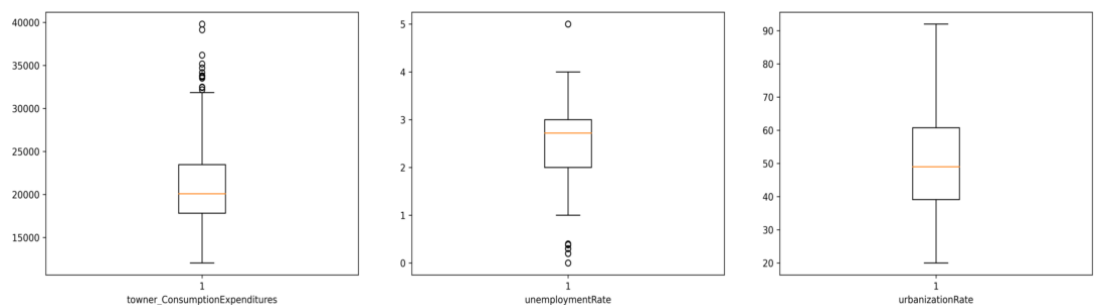


图 3.5 属性箱线图

通过观察箱线图，可以对异常值进行处理。对于如失业率为 0 这类异常值数据，我们将异常值替换为了该属性的均值。

### 3.3 数据归约

为了进行相关性分析，减少强相关的属性。首先将各属性进行归一化，再计算不同属性之间的 Pearson 相关系数，热力图如图 3.6 所示。

可以发现第二产业就业人数，第三产业就业人数，从业人数以及人口规模这四个属性之间有明显的正相关关系，又因为人口规模与最终需要预测的户籍人口数量有正相关关系，所以删除了第二产业就业人数，第三产业就业人数，从业人数这几个属性。

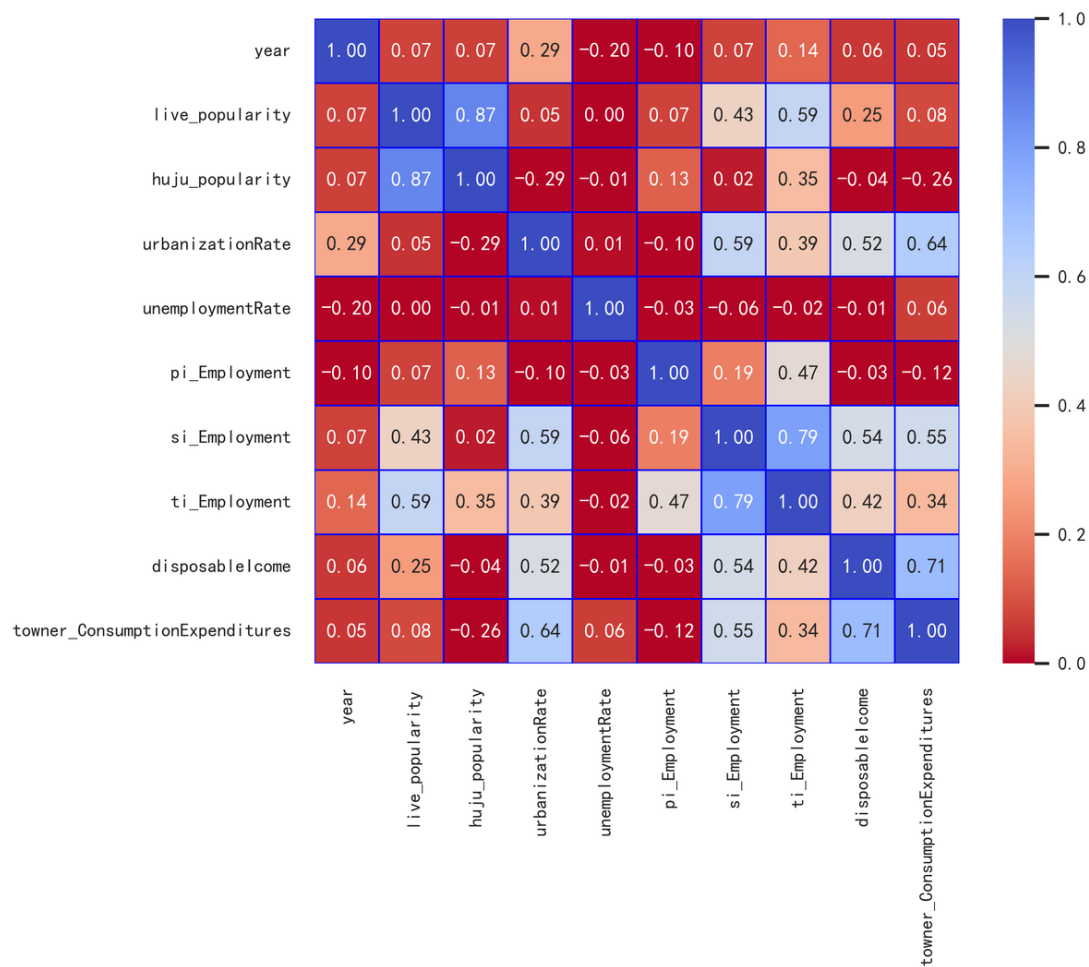


图 3.6 属性相关性热力图

## 四. 模型搭建与实验

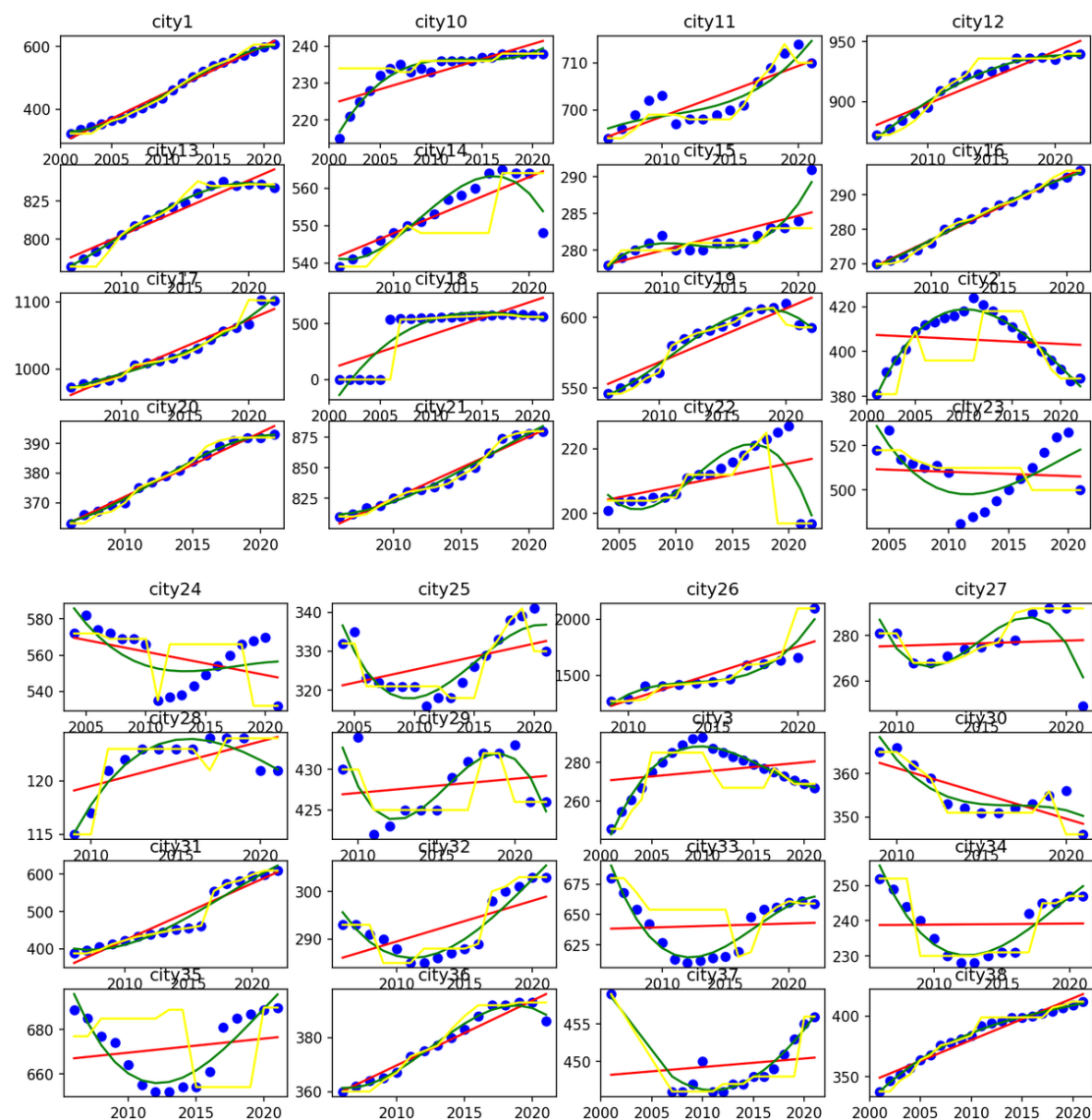
### 4.1 简单单变量回归模型

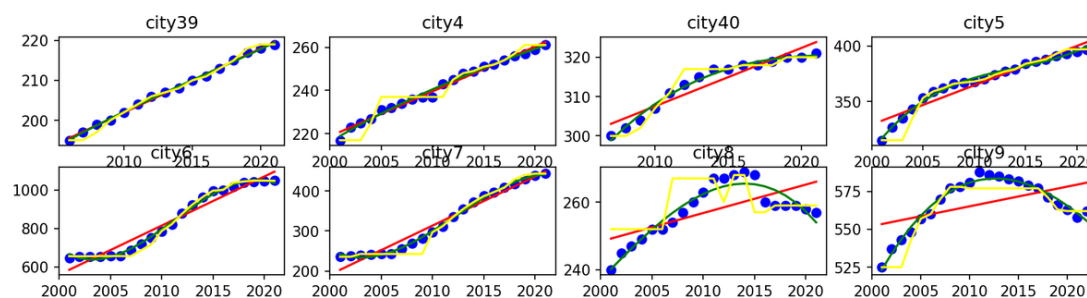
我们首先仅用人口数据进行了预测，对每一个城市的年份-人口做了线性回归 `LinearRegression()`、逻辑斯蒂回归 `LogisticRegression()`与多项式回归 `np.polyfit(xdata,ydata,3)`，并画出图像，在直观上对所有数据有一个基础感受。

下图中我们画了部分城市人口用三种方法分别做回归的结果，其中红色线条是线性回归，绿色线条是多项式回归，黄色线条是逻辑斯蒂回归。可以很明显的看到不同的城市人口变化，三种回归方式各有优劣。变化较为多端的城市人口用

逻辑斯蒂回归会更理想，而变化较稳平缓的城市使用线性回归结果更好。但是也有城市人口在三种模型上表现都一般，特别在城市人口数据发生突变时，这可能是由于就业等因素发生突变的结果，后面我们会将其他因素逐步加入考虑范围。

模型结果暗示了，不同城市的影响因素可能不同，最终的结果或许需要对每一个城市按照模型正确率或损失函数数值来选择最合适的模型预测，或者有权重的综合不同模型预测的结果。





## 4.2 多元输出线性回归

考虑到人口数据和其他数据如就业率、工资、人口密度、年龄结构、城镇化率等因素有着复杂而密切的关系,因此我们考虑使用多输入多输出的模型来构建预测模型。多元输出回归是指在给出输入示例的情况下涉及预测两个或多个目标变量的回归问题,在这里,我们认为所有作为输入的特征和需要的人口都是模型预测的输出,也就是说使用上一年的特征来预测下一年的特征和输出。为了让居住人口的数据尽量平缓,在特征中也使用了上一年居住人口的数据。当然这种方法有一个不好的假设在于,它的核心是假设所有数据都是“不相关”的,模型会为每个数据都生成一个独立的预测。这种方法的主要代码如下:

```
1 model = MultiOutputRegressor(inlinemodel)
2 model.fit(x[:-1], np.hstack([x[1:], y[1:]))) # 预测下一年特征+人口
```

**MultiOutputRegressor** 是一个集成的算法,它本身并不是模型,我们可以选择不同的内置模型,这个外嵌的模型会为输出的特征和人口的每一个数据都训练一个内置模型的参数。我选择了四种内置的模型来进行训练,下面我给出部分城市在这四种训练模型下的结果:

### 1. LinearRegression()

## 2. LinearSVR()

SVM 线性回归，用来实现线性的任务。

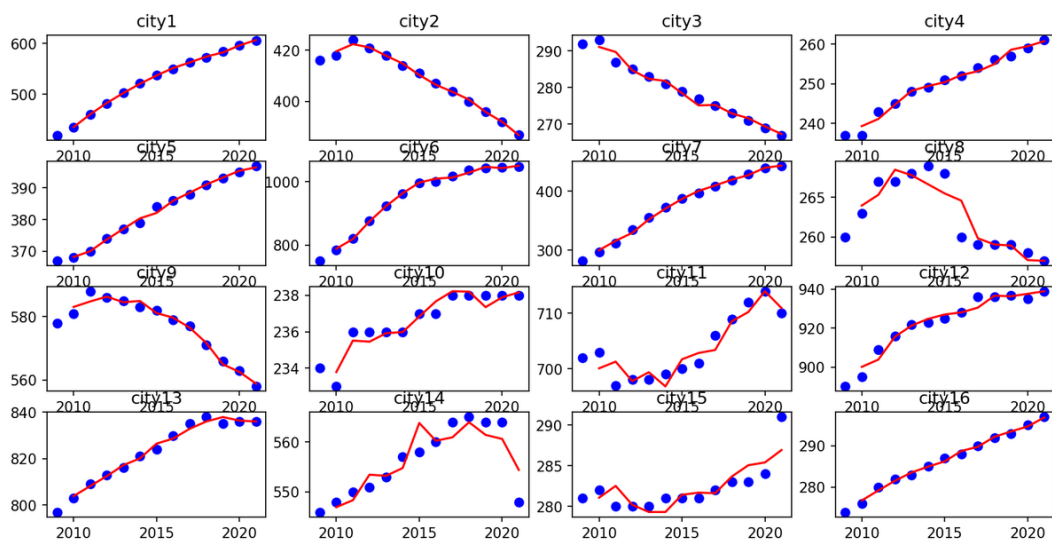
## 3. RandomForestRegressor(n\_estimators=80, max\_depth=5)

随机森林是一种集成学习方法，它通过构建多个决策树来进行预测。它对于处理大量特征、非线性关系和避免过拟合都有一定的优势。可以选择和调整的参数包括决策树的个数、最大深度、最大叶子节点数等等。

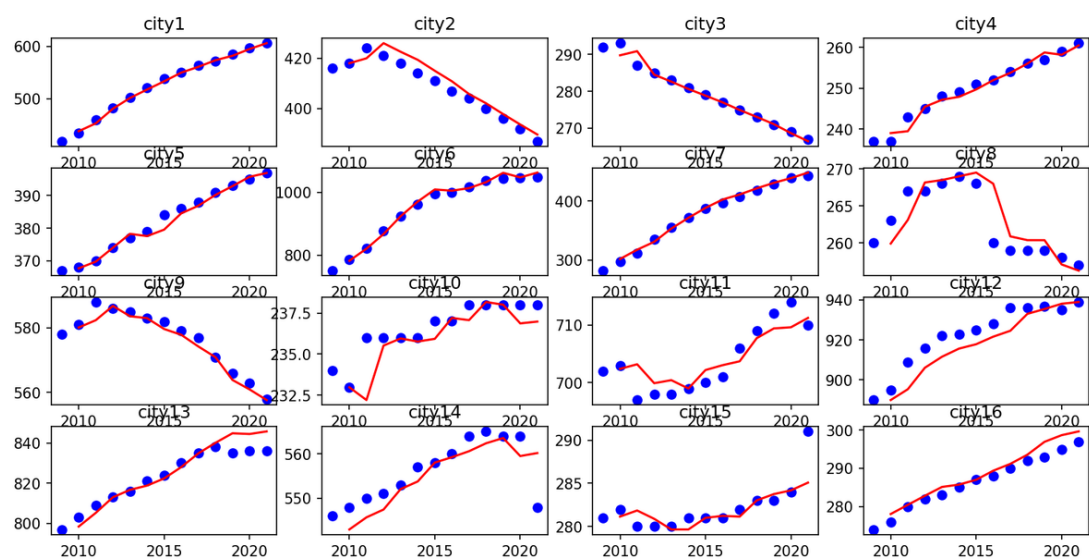
## 4. GradientBoostingRegressor(random\_state=4)

GBR 是一种集成模型的学习算法，用许多较差的学习算法组成了一个更强大的学习算法。

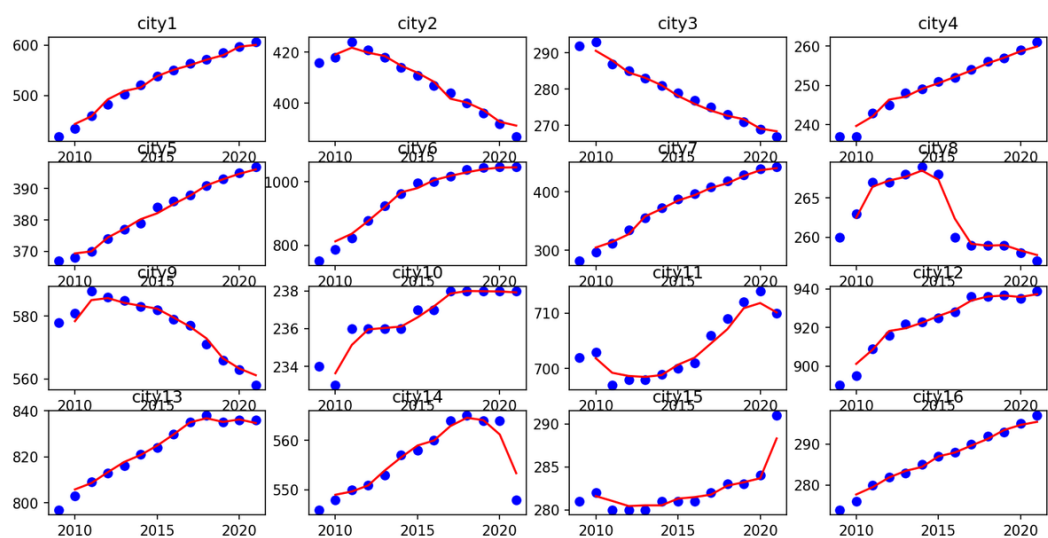
### 1. 内置模型为 LinearRegression()



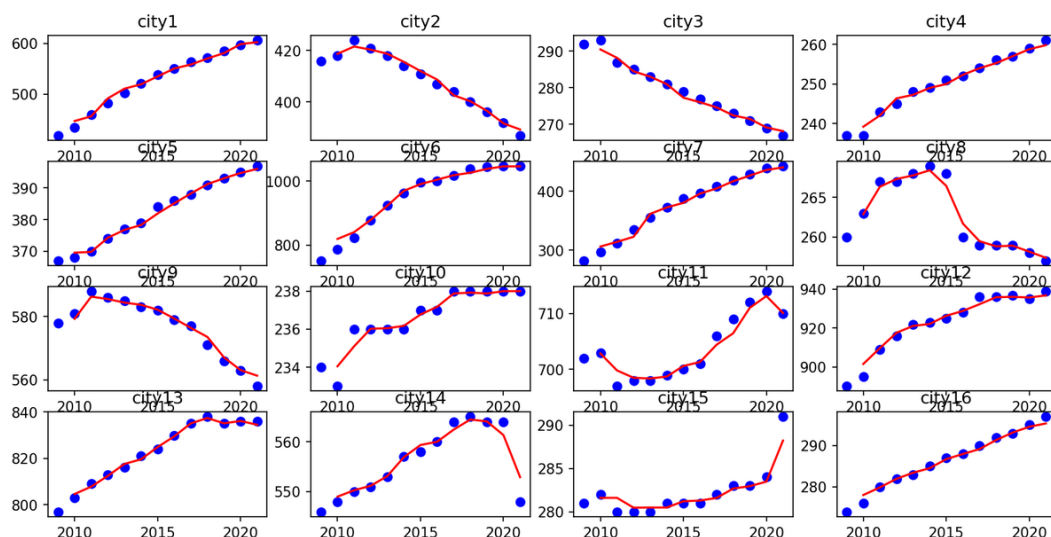
### 2. 内置模型为 LinearSVR()



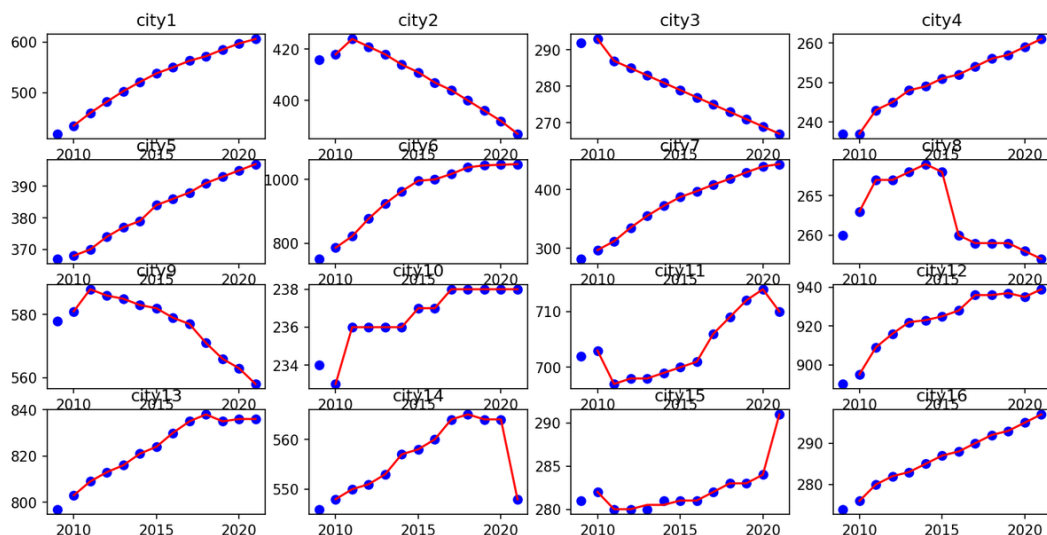
3. 内 置 模 型 为 `RandomForestRegressor( n_estimators=80, max_depth=5)`



4. 内 置 模 型 为 `RandomForestRegressor( n_estimators=40, max_depth=5)`



## 5. 内置模型为 GradientBoostingRegressor(random\_state=4)



从上面四种方法可以看到，简单的模型在预测时存在类似“延时”的现象，而复杂的模型预测结构则疑似有过拟合的可能。在最后预测时，我们使用了结果的多种组合方式，例如去掉最高最低后取平均值、根据不同城市模型拟合的 RMSE 选择模型等等，进行了多次尝试，最后获得了一个较为理想的结果。

## 4.3 基于聚类的梯度提升回归预测

不同城市的发展情况各不相同，参考国家统计局的相关数据，城市线级可以分为：一线城市、二线城市、三线城市和非线级城市四类。我们认为不同的城市



类别具备不同的人口 发展特征。所以，首先根据属性特征将题目 所给出的四十个城市进行聚类；其次，对聚类后的城市簇分别训练回归模型。为了提高回归预测的准确性，我们采用了 `xgboost` 和 `lgbboost` 两种基于梯度提升树的回归方案，最终的结果为两者预测结果的平均值。具体的实现流程如图 4.1 所示。

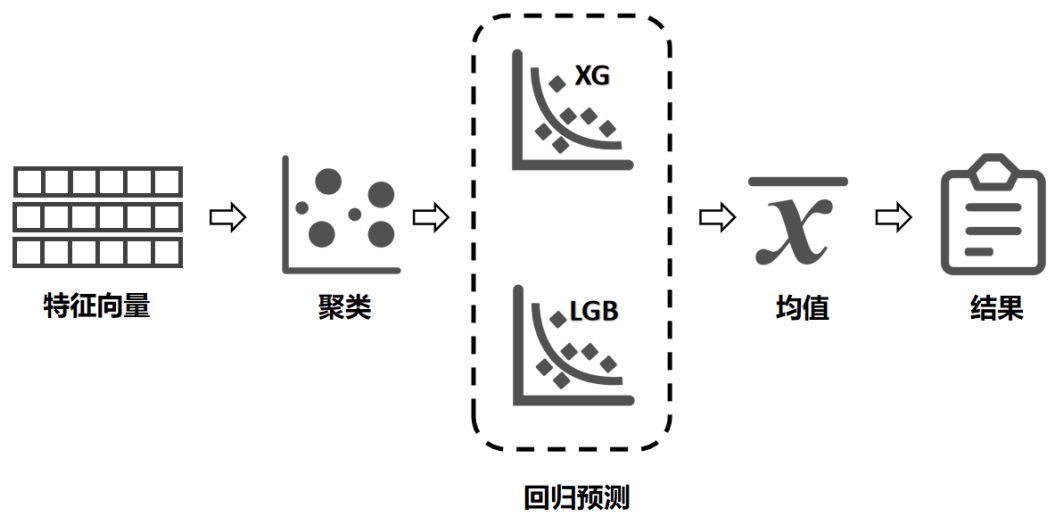


图 4.1 基于聚类的梯度提升回归预测的流程

### 4.3.1 聚类

为了将不同类别城市进行划分，将不同城市的特征数据采用 `k-means` 聚类进行处理。在选择聚类的簇数量时，我们遍历了从 2 开始的所有可能簇数量，并采用手肘法确定了最佳 `k` 值，如图 4.2 所示。



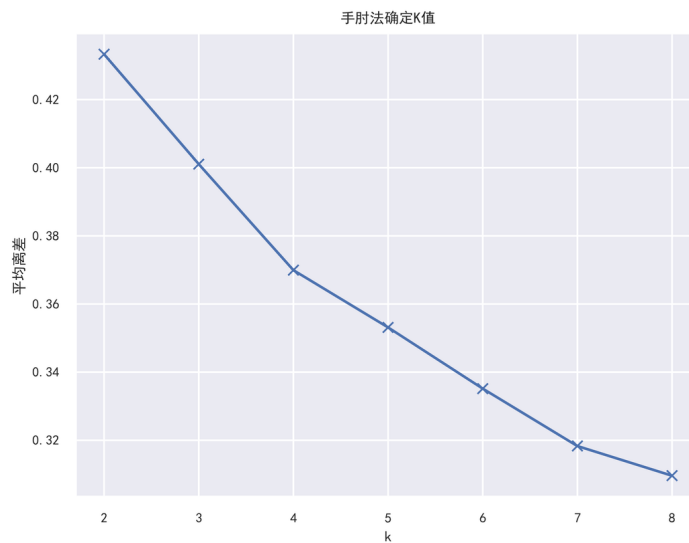


图 4.2 采用手肘法选择最佳簇数量

根据手肘法，当  $K=4$  时是一个较为明显的拐点，所以选定  $K=4$ 。

### 4.3.2 回归预测

XGBoost 的核心思想是通过集成多个弱学习器（通常为决策树）来构建一个强学习器，从而提升模型的预测精度。它通过逐步添加新的决策树来纠正前一个树的错误，以实现目标变量的更准确预测。

LGBBoost 通过构建一系列有序的决策树来逐步改进模型，每个树都尝试纠正前一个树的错误。在回归任务中，它旨在最小化预测值与实际值之间的差异。

我们将特征和输出（城市人口增长率）按照 9: 1 划分为训练集和测试集，用训练集分别训练 XGBoost 和 LGBBoost，并对这两个模型进行参数调优。从而完成了对这两个模型的训练。

将数据输入训练好的两个模型，得到分别的预测结果。取 XGBoost 和 LGBBoost 预测结果的均值，得到最终的答案。

## 4.4 基于 LSTM 的预测

LSTM(长短期记忆网络)是一种适合处理和预测时间序列数据的深度学习模型。LSTM 能够学习到时间序列中的长期依赖关系,对于城市各个特征这种可能存在复杂时间动态的数据,它能够有效捕捉数据中的趋势和模式,从而提供准确的预测和分析。此外,LSTM 模型在处理非线性和高维时间序列数据方面表现出色,使其成为城市人口数据分析的理想选择之一。



考虑到  $\tanh$  函数则可以产生更新单元状态的候选值,保持信息的连续性和稳定性,本实验选择  $\text{sigmoid}$  函数和  $\tanh$  函数作为激活函数。而常用的损失函数中,均方误差(Mean Squared Error, MSE)适用于回归任务,用来衡量预测值与真实值之间的差距,因此本实验采用 MSE 作为损失函数。

由于每个城市的序列特征不尽相同,我们对每个城市分别构建 lstm 模型,并按照 9:1 的比例将每个数据划分为训练集和测试集,接着将数据输入对应城市的模型进行预测。

在模型构建阶段,实验利用 Keras 神经网络框架实施 Adam 算法,根据所选择的层类型和参数设置对模型的权重和偏置自动初始化。本实验采用固定的批量处理大小,将一次训练模型计算的数据样本量 `batch_size` 设置为 8。`epoch` 参数则表示模型完整地遍历一次训练集中的所有样本的次数,本实验中 `epoch` 被设置为 200。考虑到在城市人口的体重预测中,需要考虑历史数据或未来数据对预测的影响,因此模型的时间步被设为 2。实验还采用了 Optuna 超参数优化库来寻找合适的参数组合。

# 五. 实验结果

使用集成学习策略，将多种模型结果进行融合以提升预测精度。

	-	default13272604	13	2024-08-01 11:02	0.40759169
4	 2	default13305382	<div>队伍名: default13305382 队伍成员 (1个): 刘星雨-唐媛-侯羽飞-郑卓云</div>	2024-11-24 20:15	0.36894322
5	-	default13228376		2024-11-24 17:46	0.30879884

## 我的成绩

到目前为止，您的最好成绩为 **0.36894322** 分，第 **4** 名，在本阶段中，您已超越 **156** 支队伍。

## 我的提交

## 队伍提交

result.csv 		
所在赛程	状态 / 得分	提交时间
经典赛	<b>0.36894321753</b> <a href="#">查看日志</a>	<b>2024/11/24 20:15</b>
备注：无备注信息		

# 六. 思考与展望

通过总结和分析，我们思考模型的精度限制与以下几个方面有关:

## 1、数据质量。

数据的完整性和准确性对最终预测的效果至关重要。数据的缺失、不完整记录或异常数据都会直接降低预测效果。在进行预测前最重要的一步就是高质量的数据清洗、异常值处理和缺失值的填补，因此，在本次实验在进行预测之前，我们花费了近两周的时间处理和清洗数据，。

## 2、模型复杂度。

模型的选择应在复杂性与过拟合风险之间取得平衡。线性模型虽然简单、可解释性强，但难以捕捉数据中的非线性关系，特别是在城市人口分析过程中，就业信息、年龄结构等多个特征值之间并非简单的线性相关；而复杂模型如梯度提升树或神经网络虽然强大，但需要更多的数据支持以避免过拟合。

## 3、参数设置。

模型参数的设置对模型性能有重要影响。例如在聚类的过程中，参数  $k$  的选择直接影响了最终聚类的效果，合理的参数设置可以帮助模型更好地学习数据中的模式，除了手动实现参数优化效果之外，未来还可以探索自动化超参数优化的方法（如贝叶斯优化）。

## 4、验证与评估方法。

因为大赛所给的数据中并没有包含 2023 年的特征集和人口数据，因此无法直接构造测试集来评估 2023 年城市人口的预测结果，因此模型的验证和调优需要更多地依赖于交叉验证和验证集的设计。通过使用更科学的验证方法，可以减少数据集划分偏差对模型的影响。此外，考虑到数据的年度性质，验证集的时间跨度也应合理设计。

## 5、特征缺失

正如前文所说，因为大赛所给的数据中并没有包含 2023 年的特征集和人口数据，因此在模型训练完成后，我们采用前一年该城市的特征数据来判断下一年的常住人口。而在大型社会事件发生时，如地震、疫情、新的人才引进政策等，它们可能会导致模型特征值的巨变，并没有模型和拟合可以判断这样的突发事件的影响。数据中也没有 2022 年和 2023 年其他特征的任何信息，在这种情况下，

是很难预测出城市人口突变的。

除此之外，我们在实验开始还遇到一个令人啼笑皆非的问题。datafountain网站的经典赛并没有指明要预测的是 A 赛数据还是 B 赛数据，所以我们前期一直在处理 A 赛的数据，但是反复调试后发现得分均在 0.002 左右徘徊，后来才意识到在一开始选择要处理的数据时就已经出错，要预测的应该是 B 赛数据，虽然在错误的方向上花费了一定的时间，但万幸我们及时发现了问题并修正。总结经验、吸取教训，后续在进行数据分析的时候前一定要先确认预测的数据。特别是当更换模型、反复测试后均没有较大的进展，应该重新回过头去检查原始数据，避免类似的错误再次发生。

## **未来展望**

在本实验中，我们基于所给城市在 2006~2020 年间的人口规模、城镇化率、就业信息等多维度数据，结合多种算法和数据处理方法，完成了对城市人口发展的分析和预测任务。然而，由于城市人口的不稳定性和数据本身的局限性，未来的研究和实践中还有诸多改进空间：例如，城镇化率、就业信息等在 2006-2010 年均出现了大篇幅缺失，有些城市的特征数据严重不全，这些数据的大篇幅缺失都会影响模型的预测结果。另外，未来还可以考虑引入深度学习模型（如时序神经网络、Transformer）以更好地捕捉长期趋势和复杂的时间依赖性。