

## 题目

### 摘 要

最后，本文对所建立的模型进行中肯评价、提出改进措施，并对模型进行一定推广。

关键词:； 号隔开

## 一、问题的提出

### 1.1 问题背景

作为一款单词猜谜游戏，Wordle 每天都为玩家提供一个真正的英语单词作为谜题，现已推出 60 多种语言版本，本文以英文版本为例。其玩法为：玩家在 6 次或更少次数内猜出一个 5 个字母的单词以解决谜题。在此过程中，玩家每一次进行的猜测必须为真实存在的单词，提交后系统将对单词进行检测，若该位置显示为绿色则代表谜底单词含有该字母，且其位置正确；若为黄色则代表该字母存在但所在位置错误；若为灰色则谜底中不包含此字母。玩家可选择“常规模式”及“困难模式”进行挑战，其中在“困难模式”下玩家若找到谜底中包含的字母，即显示绿色及黄色砖块，在其后的猜测的单词中必须包含这些字母。Wordle 官方统计了在 Twitter 上分享的每一天玩家通关所需次数。游戏两种模式玩法见图 1 及图 2。

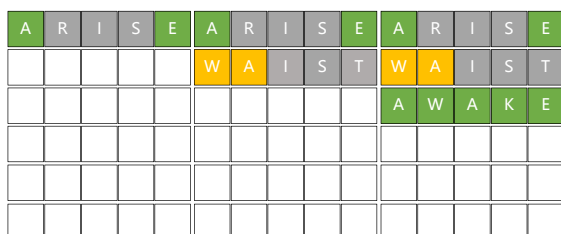


图 1 常规模式

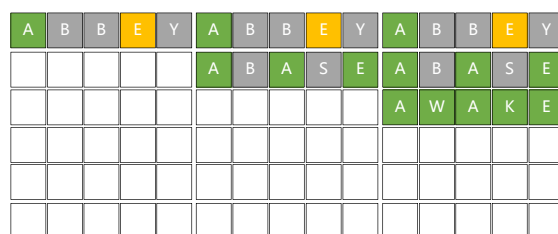


图 2 困难模式

### 1.2 问题要求

- **问题一：**官方所提供报告的结果每天数量均存在差异，建立一个模型以解释该差别并预测 2023 年 5 月 15 日报告结果的数量区间。分析单词属性对于困难模式得分比例是否存在影响，并解释其原因。
- **问题二：**建立模型预测未来谜底单词的报告结果相关百分比，说明该模型预测有哪些不确定性，并以单词“EERIE”的预测为例分析模型效果。
- **问题三：**建立一个依据单词难度分类的模型，并对谜底单词分类。确定各类别中谜底单词属性及单词“EERIE”的难度并分析该模型的准确性。
- **问题四：**进一步分析数据集中其他有趣的特征。
- **问题五：**写一封信函向《纽约时报》的谜底编辑总结分析结果。

## 二、问题的分析

### 2.1 问题的整体分析

该题是一个以热门游戏 Wordle 为背景的数据分析，自然语言处理，预测类问题。

**从分析目的看**，本题需要以单词的属性进行分析，筛选出影响玩家得分比例的主要因素，确定各分类相关联的给定单词的属性并总结，量化结果。同时还须对玩家的得分结果进行预测与研究，为《纽约时报》的谜题编辑提供参考，以进一步优化 Wordle 游戏。因此本题主要需要完成以下几方面的任务：**其一**，分析每日报告结果数量的变化情况；**其二**，确定单词属性与困难模式游玩得分的关系；**其三**，预测特定单词玩家尝试次数的百分比；**其四**，分析单词属性与单词难度的关联性；**其五**，进一步探讨数据集具有的有趣特征；**其五**：写一封信函向《纽约时报》的谜底编辑总结分析结果。

**从数据来源、特征看**，本题的数据来源于 2022 年 2 月 7 日至 2022 年 12 月 31 日 Twitter 玩家分享游戏结果，数据包括“日期”“编号”“谜底单词”“答案提交数”“困难模式参与数”“1~7 次尝试分布比例”。但未提高对数据的分析能力，提升预测效果，挖掘更多有效信息，我们将对 2023 年数据进行爬取，构造与附件数据一致的数据，两者综合进行分析。同时我们发现数据具有高维，量纲不一致等特点，且数据体量较大，因此，本题相对复杂，需对数据进行一定的预处理，以便于后续分析。

**从模型的选择看**，本题数据的时间跨度大，维度较高，且需预测未来的报告数和相关百分比的分布、基于单词难度对单词进行分类。因此本文选择 XGboost 回归模型、随机森林回归模型进行预测分析，同时利用 K-means 模型对单词进行分类，在该分类基础上建立数据标签，并采用 SVM 模型对单词“EERIE”进行预测。

**从编程软件的选择看**，本题为数据分析类，需要进行数据预处理、数据分析、数据可视化，并依据各设问建立不同类别的模型，因此我们选择 Python Jupyter Notebook 对问题进行求解，其交互式的编程范式及轻量化，方便且高效。

## 2.2 问题一的分析

问题一的核心目的有以下几点：**其一**，对报告数据进行预处理，去伪存真；**其二**，研究每日结果数据的变化；**其三**，预测 2023 年 5 月 15 日报告结果数量，并建立一个合理区间；**其四**，研究分析单词属性与困难模式得分的关系。对于给定的数据集，我们发现部分数据在正确性，完整度性等方面存在一定缺陷，因此我们须对数据集进行预处理。由于数据体量较大，因此我们将将总人数，选择困难模式人数及其变化率数据可视化，直观分析变化规律并建立普适性预测模型。同时我们还考虑到游戏人数数据等不符合正态分布，因此我们放弃线性回归而选择 XGBoost 模型进行回归预测。此外我们对单词进行属性分析，利用 Python 自然语言处理 nltk 库，建立单词的属性模型，并绘制皮尔逊相关系数热力图分析其中的关系。

2.3 问题二的分析

2.4 问题三的分析

问题三的核心目的在于**根据单词难度对单词进行分类**。因此我们结合问题一所构造的特征及相关指标，对样本数据进行 K-means 聚类分析，并结合肘部法则选择合适的  $k$  值，绘制 PCA 降维聚类结果散点图，观察聚类效果。在对样本数据进行分类后，将难度定义为词汇的分类标准，采用 SVM 模型对分类的结果进行更详细的定性及定量分析。最后对单词“EERIE”进行分析，确定其难度。此外我们还需对模型进行合理性分析，绘制分类报告、混淆矩阵热力图、ROCAUC 曲线、分类预测结构，从而更好地探讨其效果。

2.5 问题四的分析

2.6 问题五的分析

三、模型的假设

- **假设一：**假设个体在单词认知能力上差异较小。
- **假设二：**假设《纽约时报》每一天的谜底单词均随机抽取，不受人为等因素干扰。
- **假设三：**假设每日参与游戏人数不会出现较为明显的波动，在一定范围内稳定。
- **假设四：**假设若玩家在六次及以内未能成功猜出谜底单词，则游戏失败。并将游戏失败的玩家视作进行 7 次尝试。

四、符号说明

符号	符号说明
$\mu$	样本平均数
$\sigma$	样本标准差
$x_{\text{standard}}$	经过标准化后的数据
$R(x)_{m \times n}$	经过某项处理后的数据特征集
$\hat{y}$	预测值
$L^{(t)}$	目标函数
$\omega$	权重

**注：**这里并未列出部分变量，这是由于它们在不同小节处有不同的含义，因此我们会在每一节中详细讨论它们。

## 五、模型的建立与求解

对于本题，本文模型的建立与求解部分主要分为数据的准备，模型的建立、求解、结果分析。

- **数据的准备：**爬取 2023 年数据，提高模型分析数据的准确性。并对于构建的新数据集进行预处理，方便后续模型的建立；
- **模型的建立、求解、结果分析：**对于给定的数据集及爬取的新数据集，本文依据其特点，建立合适的回归、聚类、分类模型，并进行多种数据可视化，分析模型效果。

### 5.1 数据的准备

本部分我们需要爬取 2023 年的数据，以提高模型分析数据的准确性。同时还需要对数据集进行一定的预处理，对错误值进行修正，以便后续模型的建立。本部分我们所利用的数据收集网站见表 1。

表 1 数据来源

网址	描述
<a href="https://m.stockq.org/life/wordle-history.php#all">https://m.stockq.org/life/wordle-history.php#all</a>	Wordle 每日答案统计
<a href="https://twitter.com/WordleStats">https://twitter.com/WordleStats</a>	Wordle 每日报告结果

#### 5.1.1 2023 年数据爬取

观察原数据集，其统计的为 2022 年 2 月 7 日至 2022 年 12 月 31 日的情况，而缺少 2023 年数据，且问题需要预测 2023 年 5 月 15 日数据，因此为提高模型精度，我们将对 2023 年数据进行爬取，并与原数据集进行合并，使得在游戏编号、时间方面连续。新数据集读者可在附件中查阅。其中部分数据见表 2，新数据集截止至 2023 年 4 月 25 日。

表 2 2023 年部分数据

Date	Contest number	Word	Number of reported results	Number in hard mode
2023/1/1	561	whine	22072	2132
2023/1/2	562	skirt	22252	2094
2023/1/3	563	antic	22018	2072
2023/1/4	564	layer	22394	2207
2023/1/5	565	sleek	22283	2078

### 5.1.2 异常单词的修正

根据 Wordle 游戏规则，每日谜底单词均为 5 个字母长度，因此我们对原数据集所有单词进行分析，筛选出字母数不为 5 个的单词，并依据 Wordle 游戏每日单词统计的数据与游戏编号进行比对，将上述错误单词替换为正确单词，更换结果见表 3。

表 3 单词修正

游戏编号	异常单词	替换单词
207	favor <sup>1</sup>	favor
314	tash	trash
525	clen	clean
545	rprobe	probe

### 5.1.3 人数异常值

为了更好地初步发现异常值，我们绘制出每日游戏报告结果总人数及选择困难游戏模式人数变化图，如图 3。观察该图，我们可以发现，2022 年 11 月 30 日总人数突然减少明显，但其对应的选择困难人数并未有明显变化，且两者数据相近。因此，这里我们定义其为离群点，其明显不符合发展变化规律，对其的处理，我们将再下文展开分析。

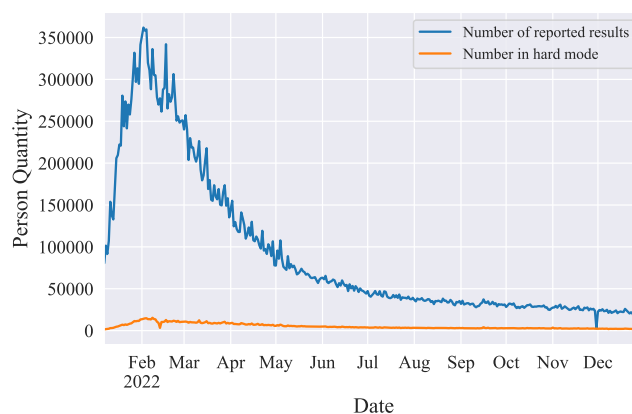


图 3 总人数及选择困难游戏模式人数日变化

同时我们还计算出每日选择困难人数占总人数的比例，绘制出每日选择困难模式人数频率变化及变化率图示，见图 4 及图 5。

<sup>1</sup>该单词错误原因是在表格数据中其后有一空格，该空格会影响后续问题的解决。

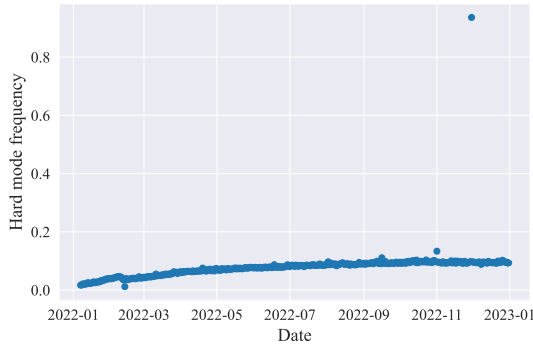


图 4 每日选择困难模式人数频率变化

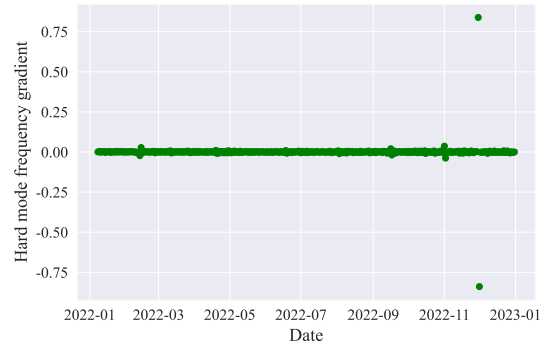


图 5 每日选择困难模式人数频率变化率

通过观察上述两幅图，我们可以清晰发现异常值，但也存在部分图示上难以鉴别的离群点，因此这里我们设定变化率阈值为 0.02，若每日变化率绝对值超过该阈值，且在相邻两天内变化率绝对值超过该阈值，则定义其为离群点，对其进行处理。这是由与若某一天突变严重，会引起当天与后一天变化率均会超过阈值，而真正地异常值应为首次超过的那一天。但异常值为总人数还是选择困难游戏人数，我们还需要结合改日前后几天数据记性判断。通过筛选，我们得到异常数据，见表 4。

表 4 异常数据分析

游戏编号	当日单词	游戏总人数	选择困难模式人数	选择困难人数变化率	异常值
239	robin	277471	3249	-0.022787	3249
240	ultra	261521	10343	0.027840	无
500	piney	27502	3667	0.036272	3667
501	inept	27670	2640	-0.037926	无
529	study	2569	2405	0.838601	2569
530	eject	22628	2200	-0.838937	无

#### 5.1.4 百分比异常值

根据对原数据集中尝试次数百分比的理解，其 7 次累计和应为 1，因此我们计算数据中玩家尝试次数百分比的累计和，我们发现，除游戏编号为 281 的数据累计百分比为 126%，其余数据均在  $(100 \pm 2)\%$  的范围内，因此我们有理由认为该天存在统计误差，并记为异常值。

#### 5.1.5 异常值处理

对于异常值的处理，为了尽可能保留数据信息，我们采用拉格朗日插值法，对上述异常数据进行修正。拉格朗日插值法（Lagrange's Interpolation）是一种多项式插值的方法。对于给定的  $n + 1$  个坐标不同的点，其可以给出一个恰好经过这  $n + 1$  个点的多

项式函数。拉格朗日基本多项式（插值基函数）如下

$$l_j(x) = \prod_{i=0, i \neq j}^n \frac{x - x_i}{x_j - x_i}, j = 0, 1, \dots, n \quad (1)$$

则拉格朗日插值多项式为

$$L(x) = \sum_{j=0}^n y_j l_j(x) \quad (2)$$

此多项式经过给定的  $n + 1$  个点。

经过上述处理后，我们对异常值进行了修正，得到了新的数据，见[表 5](#)及[表 6](#)。

表 5 人数异常值修正

游戏编号	当日单词	游戏总人数	选择困难模式人数	异常值	修正值
239	robin	277471	3249	3249	9249
500	piney	27502	3667	3667	2667
529	study	2569	2405	2569	25569

表 6 百分比异常值修正

类	游戏编号	1 try	2 tries	3 tries	4 tries	5 tries	6 tries	X	百分比和
异常值	281	1	2	18	44	26	26	9	126
修正值	281	1	2	18	44	26	9	1	101

## 5.2 问题一模型的建立与求解

对于该问题，我们需要完成以下几点任务：

- 分析每日报告结果变化情况，并建立模型来解释这种变化；
- 利用上述模型对 2023 年 5 月 15 日的报告结果进行预测，并设定置信区间；
- 分析单词属性是否会影响困难模式下玩家尝试次数的比例，并解释原因。

因此，该部分我们围绕上述任务进行分析、解决。

### 5.2.1 每日报告结果分析

根据对数据集进行观察，我们发现游戏编号是按顺序进行连续编号的，因此游戏编号可以看作是对数据集无影响的序列。我们按游戏编号将报告结果绘制为散点图，并绘制出数据变化的整体趋势，如[图 6](#)所示。



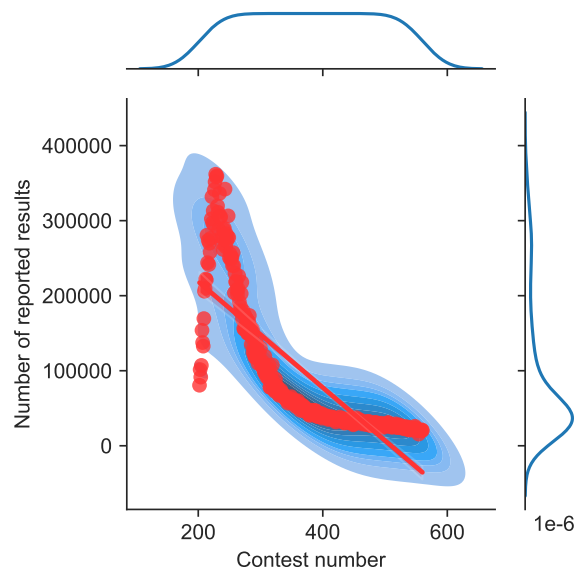


图 6 每日报告结果变化趋势

观察该图，我们可以发现报告结果呈现了先增加后减小的趋势，最后逐渐趋近于稳定。其在一定程度上反映了游戏玩家的人数变化规律。考虑到游戏存在生命周期，我们认为报告结果的现象与游戏生命周期存在关系。因此考虑到游戏或信息的传播一般会经历增长、成熟、衰退和稳定的时期。报告结果是当天玩与分享 Twitter 的 Wordle 玩家的数量。同时我们认为推文的数量主要受玩家数量的影响，同时更多的推文也将吸引更多的玩家。当普通的 Twitter 用户看到一条关于这款游戏的推文时，他们将有  $\alpha$  的概率成为一位新的 Wordle 玩家。当一位玩家试玩该游戏后，他将会有  $\beta$  的概率发送一条推文，成为一位普通玩家。曾时试玩的玩家有  $\gamma$  的概率返回游戏，同时玩家将在多次比赛后有  $\sigma$  的概率第二天不再游戏。因此我们有理由认为报告结果的数量不仅仅与表层的时间序列有关，而有更深次的内在联系。因此这里我们将不考虑时间序列模型进行预测。同时，考虑线性回归模型，此时我们需要检验该数据是否符合正态分布，因此我们绘制随机变量直方图、概率密度图以及 Q-Q 图 (Quantile-Quantile Plot)，如图 7 所示。

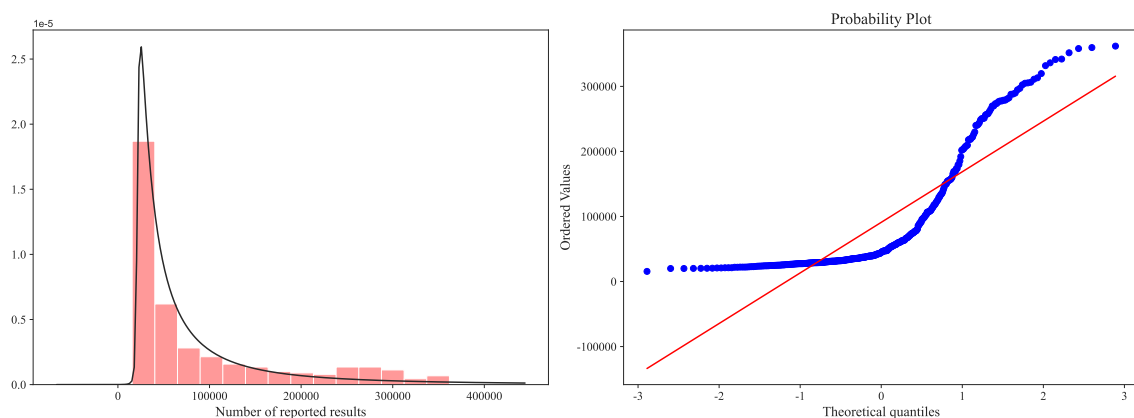


图 7 正态分布分析

dgy 分析

因此我们有理由认为对于该数据集, 进行线性回归分析可能会在一定程度上得到较优效果, 但这并不是在数据集集中部分的效果, 极有可能会对未来的预测产生较大误差, 因此我们考虑建立极端梯度提升 (eXtreme Gradient Boosting, XGBoost) 模型, 首先对每日报告总人数进行预测, 再对选择困难模式人数进行预测分析。XGBoost 算法是一种基于树模型的优化模型, 该算法通过多次迭代, 生成一个新的树模型用于优化前一个树模型, 随着迭代次数的增多, 该模型的预测精度也会相应提高<sup>[1]</sup>。

记通过数据处理后的数据集特征为  $R(x_{ij})_{m \times n}$ , 表示其包含  $m$  天的游戏情况,  $n$  个特征, 在训练中形成的 CART 树的集合记为  $F = \{f(x) = w_{q(x)}, q: \mathbf{R}^n \rightarrow T, w \in \mathbf{R}^T\}$ , 其中  $q$  为树模型的叶节点决策规划,  $T$  为某一树模型叶节点数量,  $w$  为叶节点对应的得分<sup>[2]</sup>。对于预测的  $y$  值, 其计算公式为

$$\hat{y} = \varphi(x_i) = \sum_{k=1}^K f_k(x_i) \quad (3)$$

XGBoost 算法在每一次迭代过程中会保存前面所学习的模型, 会将这些模型加入到新一轮迭代过程中, 因此我们记第  $i$  个模型为预测结果为

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i) \quad (4)$$

XGBoost 算法的目标函数计算公式如下

$$L^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \gamma T + \frac{1}{2} \lambda \sum_{j=1}^T w_j^2 + \text{const} \quad (5)$$

上述公式中,  $l$  为模型误差损失, 描述在该模型下预测值与实际值之间的出差异损失,  $\Omega$  为模型叶节点的正则项惩罚系数,  $\gamma$  与  $\lambda$  为模型的超参数<sup>[2]</sup>。

通常情况下, 我们难以用枚举法得到在模型中所训练出来的树结构, 因此这里采用贪婪算法, 从单叶子节点开始, 通过迭代方法, 将其加入到树结构中, 从而得到最优解, 其计算公式<sup>[3]</sup>如下

$$\mathcal{L}_{split} = \frac{1}{2} \left[ \frac{(\sum_{i \in I_L} g_i)^2}{\sum_{i \in I_L} h_i + \lambda} + \frac{(\sum_{i \in I_R} g_i)^2}{\sum_{i \in I_R} h_i + \lambda} - \frac{(\sum_{i \in I} g_i)^2}{\sum_{i \in I} h_i + \lambda} \right] - \gamma \quad (6)$$

其中  $I_j = \{i | q(x_i) = j\}$  为叶节点  $j$  上的样本集合<sup>[2]</sup>, 且有

$$g_i = \partial_{\hat{y}^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)}) \quad (7)$$

$$h_i = \partial_{\hat{y}^{(t-1)}}^2 l(y_i, \hat{y}_i^{(t-1)}) \quad (8)$$

## 5.2.2 预测 2023 年 5 月 15 日数据

依据上述分析，我们首先以报告结果总人数为预测目标，再以报告选择困难模式玩家人数为预测目标对数据集进行训练以及测试<sup>2</sup>。划分训练集及测试集比例为 9 : 1，通过 XGBoost 算法对训练集进行训练，同时在测试集上进行评估。我们得到训练测试拟合图，见图 8 及图 9。

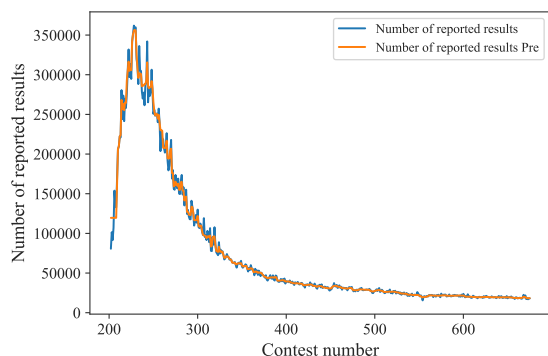


图 8 XGBoost 预测结果-总人数

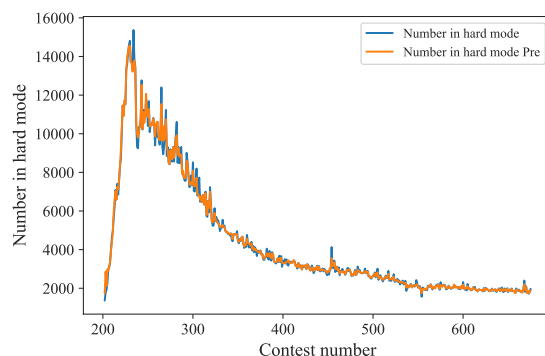


图 9 XGBoost 预测结果-困难人数

同时我们还绘制上述两个模型的预测误差图，见图 10 及图 11。

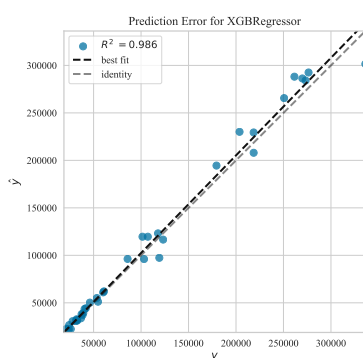


图 10 XGBoost 预测误差-总人数

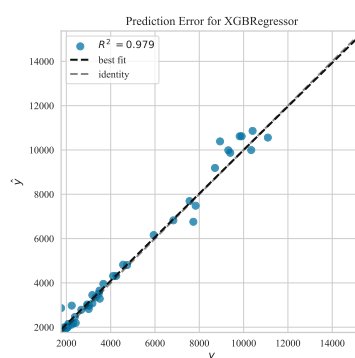


图 11 XGBoost 预测误差-困难人数

此外，我们还计算出模型的平均绝对误差（Mean Absolute Error, MAE）、均方误差（Mean Squared Error, MSE）、均方根误差（Root Mean Squared Error, RMSE），以及拟合优度（Goodness of Fit）即  $R^2$ ，其具体计算方式如下：

### • 平均绝对误差

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (9)$$

<sup>2</sup>对于报告结果总人数预测，我们仅传入游戏编号序列作为自变量；对于报告结果选择困难模式人数预测，我们传入游戏编号及总人数序列作为自变量

- 均方误差

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (10)$$

- 均方根误差

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (11)$$

- $R^2$

$$R^2 = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (12)$$

最终计算结果见[表 7](#)。

表 7 XGBoost 回归预测结果

预测	MAE	MSE	RMSE	$R^2$
总人数	2586	29849352	5463	0.9957
困难人数	97	38334	196	0.9959

根据上述结果，我们可以发现 XGBoost 回归预测对于该数据集有较优表现，预测效果良好，在误差允许范围内。

现我们将对 2023 年 5 月 15 日报告情况进行预测，并且为使分析更具有一般性，我们给出在 95 % 的置信水平下的预测区间。但在上述分析中，我们发现对于每日报告的人数并不服从正态分布，这是由于我们选取 Wordle 从发布至今的每日人数情况进行分析，但游戏从发布至今存在增长、成熟、衰退与稳定的时期。根据[图 8](#)及[图 9](#)所示结果，我们可以发现从 2023 年 6 月开始，Wordle 的人数将会进入稳定期，对于该时期的数据我们有理由认为其每日报告的人数服从正态分布。因此我们可以利用[表 7](#)中的均方根误差（RMSE）来计算预测区间，其计算方式如下：

记每日报告人数为  $X$ ，则  $X \sim N(\mu, \sigma^2)$ ，其中  $\mu$  为样本均值，这里由于处于稳定期，我们假设预测值近似为样本均值， $\sigma$  为 RMSE，则标准误差为

$$\text{SE} = \frac{\sigma}{\sqrt{n}} = \frac{\text{RMSE}}{\sqrt{n}} \quad (13)$$

其中  $n$  为样本总量。这里我们设定预测区间为 95 % 的置信水平，则有

$$P(\mu - 2 \cdot \text{SE} < X < \mu + 2 \cdot \text{SE}) = 0.95 \quad (14)$$

因此 95 % 的置信区间为

$$[\mu - 2\sigma, \mu + 2\sigma] \quad (15)$$

通过上述分析，预测具体结果见表 8。

表 8 2023 年 5 月 15 日报告情况预测结果

预测	预测值	RMSE	SE	区间下界	区间上界
总人数	18334	5463	251	17831	18352
困难人数	1874	196	9	1856	1892

因此，我们得到最终预测结果，即 2023 年 5 月 15 日的报告情况为：报告总人数为 [17831, 18352] 人，选择困难模式游戏的人数为 [1856, 1892] 人。

### 5.2.3 单词属性与玩家在游戏中的表现关系

### 5.3 问题二模型的建立与求解

### 5.4 问题三模型的建立与求解

### 5.5 问题四

### 5.6 问题五

## 六、模型的评价与推广

### 6.1 模型的评价

- 模型的优点：

- 1.
- 2.

- 模型的缺点及改进：

- 1.
- 2.

### 6.2 模型的推广

## 参考文献

- [1] 陈振宇, 刘金波, 李晨, 季晓慧, 李大鹏, 黄运豪, 狄方春, 高兴宇, 徐立中. 基于 LSTM 与 XGBoost 组合模型的超短期电力负荷预测 [J]. 电网技术, 2020, 44(02): 614-620. DOI:10.13335/j.1000-3673.pst.2019.1566.
- [2] 杨贵军, 徐雪, 赵富强. 基于 XGBoost 算法的用户评分预测模型及应用 [J]. 数据分析与知识发现, 2019, 3(01): 118-126.
- [3] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16). Association for Computing Machinery, New York, NY, USA, 785-794. <https://doi.org/10.1145/2939672.2939785>.

## 附 录

[A] 图示

## [B] 支撑文件列表

支撑文件列表如下（列表中不包含原始数据集以及中途产生的临时数据文件）：

文件夹名	描述
html 文件	包括所有解决问题的源程序运行结果
ipynb 文件	包括所有解决问题的源程序源代码
py 文件	包括所有解决问题的源程序输出
仿真结果	python 文件 包括附件 1 的 8 次航班全时刻的飞行状态预警

## [C] 使用的软件、环境

**C.1:** 为解决该问题，我们所使用的主要软件有：

- TeX Live 2022
- Visual Studio Code 1.77.3
- WPS Office 2023 春季更新（14036）
- Python 3.10.4
- Pycharm 2023.1 (Professional Edition)

**C.2:** Python 环境下所用使用到的库及其版本如下：

库	版本	库	版本
copy	内置库	matplotlib	3.5.2
jupyter	1.0.0	numpy	1.22.4+mkl
jupyter-client	7.3.1	openpyxl	3.0.10
jupyter-console	6.4.3	pandas	1.4.2
jupyter-contrib-core	0.4.0	pycharts	1.9.1
jupyter-contrib-nbextensions	0.5.1	scikit-learn	0.22.2 psot1
jupyter-highlight-selected-word	0.2.0	sklearn	0.0
jupyterlab-pygments	0.2.2	snapshot_phantomjs	0.0.3
jupyterlab-widgets	1.1.0	xgboost	1.6.1
jupyter-latex-envs	1.4.6	yellowbrick	1.4
jupyter-nbextensions-configurator	0.5.0		



## [D] 问题解决源程序

### D.1 航班 1 数据分析

## D.2 航班 2 数据分析

### D.3 航班 3 数据分析