Problem Chosen	2023	Team Control Number
C	MCM/ICM	2307166
	Summary Sheet	

基于微分方程和 k 近邻的世界建模的病毒传播特征和难度决定因素

最近，益智游戏《世界大战》在世界范围内广泛传播，在 Twitter 等社交媒体上引起了高度关注。了解《世界大战》热的传播机制和影响游戏难度的因素，可能会对理解互联网时代的病毒式传播和人类大脑联想词语的方式等重要问题有所启发。

我们在 SIR 模型的基础上开发了一个类似流行病学的微分方程模型，描述了报道总数的变化，并使用遗传算法对模型进行拟合，以最小化 MSE。然后，我们使用拟合的模型对 2023 年 3 月 1 日的报告总数进行了点预测。为了获得预测的置信区间，我们使用了 Bootstrap 方法。为了提高 Bootstrap 样本的拟合速度，我们使用了计算速度更快的 Nelder-Mead 方法，该方法基于遗传算法优化的相同初始参数。将 1000 个 Bootstrap 估计由小到大排列，选取第 25 和 975 个估计作为预测区间的下界和上界。

本文构建了两个特征性的词法内部特征。我们通过假设英语词汇中字符出现概率的一阶马尔可夫性来定义规律性和纯洁性(负熵)特征。所有的迹象都表明，它们确实与被猜测的词汇的难度有关。日常生活中重复字符的数量、词汇的使用频率等规律也在本文中被用于预测和解释。

KNN 回归具有保证预测分布和仍然为零的优良特性，因此我们使用 KNN 回归模型对世界大战的分布进行预测。协方差矩阵在本文中被广泛用于变量的初始选择和确定变量之间的关系。在 KNN 回归中，我们使用协方差矩阵选择与分布显著相关的自变量，并使用交叉验证选择最优自变量和 K 值。为了预测特定谜题单词的分数未来分布，我们再次使用了 bootstrap 方法来获得 95%的置信区间。

我们使用《世界大战》玩家的中位数分数来衡量谜题中单词的难度，并将难度分为“简单”、“普通”和“困难”。我们首先基于协方差矩阵筛选与难度显著相关的特征，在初始筛选的基础上使用 KNN 分类器对特征进行分类，并通过交叉验证选择预测准确率最高的 K 值。结果表明，这些特征可以有效地预测单词的难度分类。对于单词难度的预测，我们使用 KNN 分类器和我们建立的与难度相关的特征来将它们分配给现有的类别。

在对数据的进一步探索中，我们最重要的发现是，困难模式百分比的变化与我们最初建立的传染样模型中假设的忠实玩家的变化表现出高度的相似性，这在一定程度上证实了我们模型的合理性。

目录

信

致:《纽约时报》拼图编辑 From: Team 2307166

日期:2023 年 2 月 1 日

主题:回复:在我们的益智游戏，世界

亲爱的编辑,

谢谢你来信询问关于你们著名的世界大战游戏的信息。考虑到它是一款风靡全球的益智游戏，分析它的流行趋势和传播机制，以及它的单词难度，不仅对贵公司很重要，而且对整个互联网趋势的传播和对人类语言的理解也很重要。

首先，在 Twitter 上上传世界游戏分数的数量符合传统的疫情传播模型;不同的是，流行病患者最终失去了传播能力，而一小部分世界大战玩家成为资深玩家，他们继续玩世界大战并传播相关信息。如果 Twitter 上的搜索结果数量真实地反映了世界的热度，那么我们的流行病学模型告诉我们，世界已经进入其生命周期的尾部或平台期;例如，我们以 95%的置信度预测 2023 年 3 月 1 日的报告数量包含在区间[11173.04,17069.15]中;我们从模型中知道，为了延续其传播能力，你应该扩大用户群——这当然是困难的;或者你可以尝试通过增加忠实用户的粘性来延续游戏的生命:比如通过增加特别的活动来培养用户习惯，减缓老用户的退出。

关于你提到的困难模式玩家比例问题;我们的结论是，文字的属性并不会影响这个比例。由于玩家在每天尝试游戏之前不应该知道关于每日单词的信息;而由于我们的模型中忠实玩家的演变与这个比例的趋势相似，这可能意味着这个比例的变化可以单独用游戏的生命周期来解释。可能有些团队认为某些特定的难度指标与这一比例有关，但这可能是因为这些特征随着时间的推移存在整体趋势，就像困难模式玩家的比例一样，所以回归等统计工具可能会错误地将这一趋势视为两者之间的直接关系。

您还要求我们对特定日期和特定单词的结果分布进行预测。根据我们的实验，考虑日期因素的效果并不显著，所以我们只使用单词信息:我们将一些与你希望预测的单词相似的单词的结果分布平均为预测结果。要平均的确切单词数以及如何选择评估相似度的因素是由机器通过一些测试方法确定的。通过假设推文数据具有代表性，我们通过一些特殊的统计方法来确定预测区间。例如，我们对 EERIE 这个词的分布估计是(0.20 4.87 23.55 35.36 23.75 9.94 2.33)。预测区间在预测值上下 5%左右——这取决于具体的尝试次数。我们预测的不确定性主要在于我们没有处理玩家结构的变化——根据我们的职业流行病学模型，忠诚玩家的比例可能会增加，这可能会在半个月后改变对相同单词的尝试次数。

我们认为，人们猜测的次数分布是衡量一个单词难易程度的重要指标。因此，我们相应地将单词难度分为三类:简单、中等和困难。通过做类似于前一段的事情，我们可以预测生词的难度;例如，我们认为 EERIE 是平均难度。重复字母的数量、加权纯度和加权规律性与难度评级相关，前两个正相关，后两个负相关。后两个是我们设计的统计指标，用来衡量单词的难度属性。

除了大家之前感兴趣的问题，还有一些其他的见解，可能会对大家有所帮助:

我们在挑战模式报告的比例上发现了一个不断增加的时间趋势，这与我们在前一节中用来预测报告总数的模型中建立的忠诚玩家存在的假设是一致的。事实上，挑战模式报告比例的时间趋势与我们假设的忠诚玩家比例的时间趋势高度相似，我们有理由相信挑战模式报告比例与忠诚玩家数量存在一定的相关性，这意味着你可以通过观察挑战模式报告比例的趋势来判断世界忠诚玩家比例的变化，从而优化你的商业策略。

我们还发现，世界玩家分数的分布与世界的难度有一定的规律，其中从第一次到第三次尝试的猜对成功率百分比与难度等级呈负相关，而从第四次到第七次以上尝试的猜对成功率百分比与难度等级呈正相关。你可以通过观察玩家分数分布的变化来总结世界游戏难度的变化。

选择带有重复字母的单词作为世界显著降低了玩家在第一次尝试时猜测世界的可能性，这可能与玩家通常选择不包含重复字母的策略作为初始猜测有关。如果你想降低游戏的难度来吸引更多玩家，你可以通过在游戏时指出单词中重复字母的数量来增加玩家猜出谜题的可能性。

最好的，

MCM 战队#2307166

1 介绍

1.1 背景与问题重述

《世界谜题》是一款益智游戏，玩家尝试在 6 次或更少的时间内猜出一个 5 个字母的单词，并在每次猜出后获得反馈。其流程如图 1 所示。

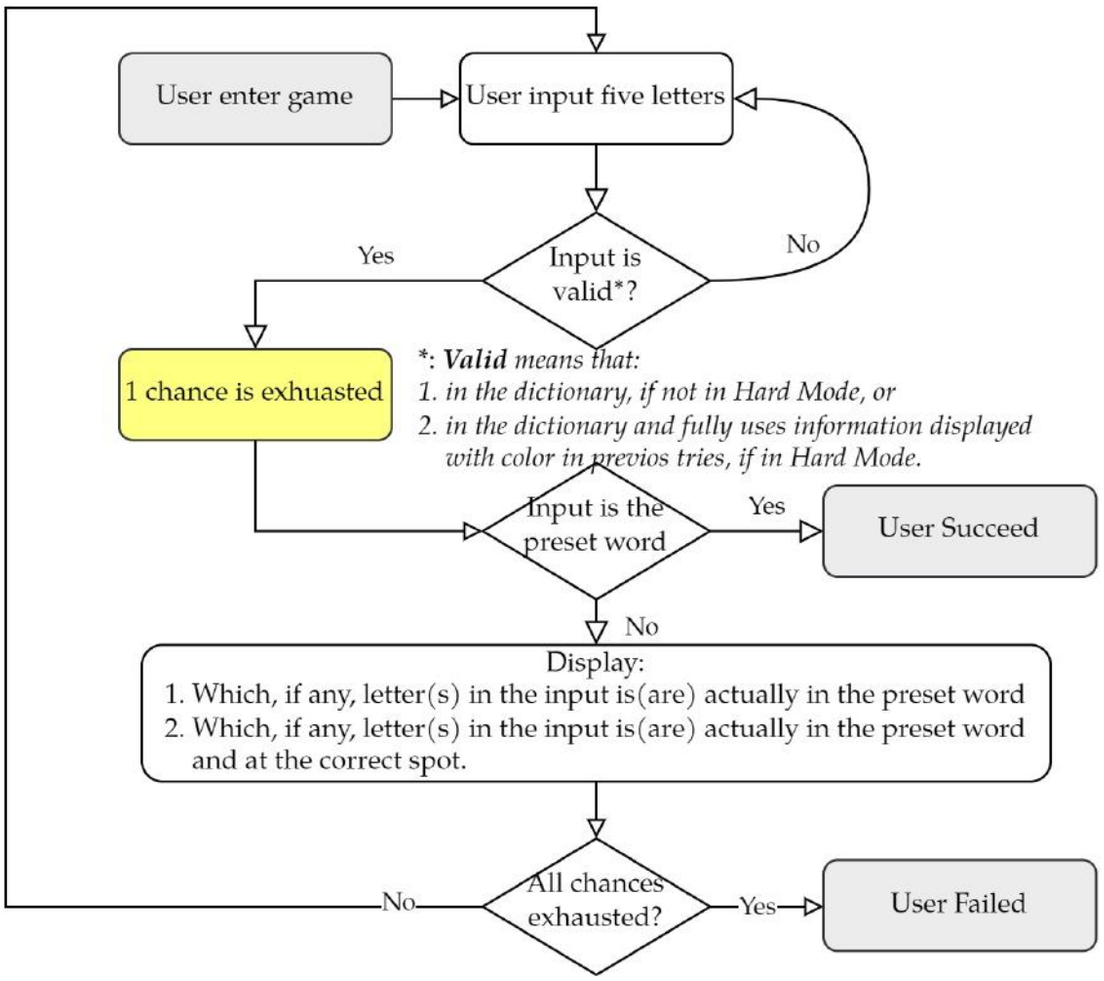

图 1:世界的流程图

我们受《纽约时报》雇佣，只使用他们发给我们的数据来分析和回答以下问题。

问题 1 问题 1 由两个子问题组成:

1.日期和其他因素如何影响报告结果的数量?如果给定一定的日期，如何预测报告结果的数量?(问预测区间)

2. 单词本身的属性是否会影响在困难模式下打出的分数所占的百分比?(需要进行机制分析)

问题 2 如果给定日期和解词，如何预测尝试的分布?哪些影响结果的因素可以省略?(需要一个预测区间，或者对预测模型的预测能力进行分析)

问题 3 如何衡量难度?如何根据难度将解题词分成不同的组?对于每一组，单词的身份是什么?如何对新解词进行分类?(需要分析分类模型的预测能力)

问题 4 在数据集中发现有趣的特征!

2 建模准备

2.1 假设

主要假设有:

英语中的词给定数据中的词是英语词，因此与字典中的其他英语词具有一些共同的概率同一性。

看起来像有效英语单词的单词更容易被记住，并被优先考虑。

特定于模型的假设将在后续章节中介绍。

2.2 符号

本文使用的主要符号列于表 1。

表 1:符号

Symbol	Definition
w_i	word i
c_i	character i
$c_{i,j}$	j-th character in word i
T_i	transition process i
Reg (w_i)	regularity of word i
Irr (w_i)	irregularity of word i
Pur (w_i)	purity of word

2.3 数据清洗

在检查数据集时，我们在数据集中发现了两种类型的异常值:单词长度不是 5 和异常低的总报告数。异常数据的详细信息如下。

Date	Contest number	Word	Number of reported results
2022/12/16	545	rprobe	22853
2022/11/26	525	clen	26381
2022/4/29	314	tash	106652
2022/11/30	529	study	2569

为了保证数据的正确性，我们选择了去除异常数据，而不是校正和插值。

对于 KNN 回归和分类，为了避免量级对计算观测间距离的影响，我们根据下面的等式对数据进行了归一化。

$$x_{ij}^{*} = \frac{x_{ij} - \overline{x}}{sd\left(X_j\right)},$$

其中xij是标准 j 的第 i 个观测值，Xj是标准 j 的所有观测值，xij*是归一化数据。归一化数据的均值为 0，方差为 1。

3 Word Feature Engineering

3.1 规律

在英语中，有效的单词通常具有规则的形式模式，这是人类很难捕捉和形式化的，但它可以影响人们在玩《世界大战》时的选择，因为那些规则的单词可能对英语用户来说更熟悉。

假设每个字符的概率分布只是由它在单词中的前一个字符决定的，我们尝试用一个类似 bi -gram 的[1]链来对这个过程进行建模，该链估计在单词中已知前一个字符的情况下，一个字符出现的可能性;为了利用单词的开头和结尾的信息，我们在单词的开头和结尾制作了两个虚拟字符 BOW 和 EOW，如图 2 所示。

正则性被定义为每个字符向其后继字符传递过程的概率的几何平均值，如公式所示

$$\mathrm{Reg}(w_i) = \sqrt[n]{\prod_{j=1}^{n} P\left(c_{i,j} \mid c_{i,j-1}\right)},$$

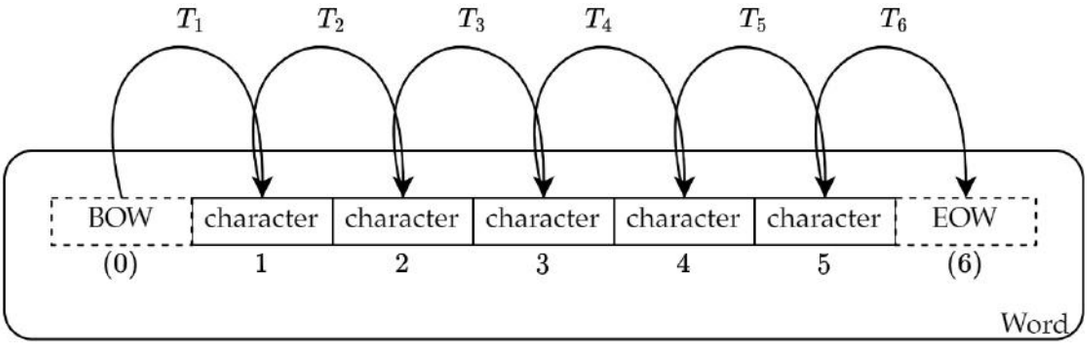

图 2:虚拟人物和过渡过程

其中，使用$P(c_i \mid c_{i-1})$，有时在本文中也表示为$P(T_i)$来表示从字符 i - 1 过渡到字符 i 的可能性。

3.2 纯度

如果给定单词中的一个字符，我们可以获得多少关于其后续字符的信息?这在《世界大战中是一个重要因素，因为游戏的反馈会告诉我们猜测的字母是否真的存在于当天的单词中，所以如果接下来的单词更加不可预测，那么解决谜题就会变得更加困难。

角色的纯洁性ci定义如下:

$$\mathrm{Pur}(c_i) = \sum_{j=1}^{n} P(c_{i+1} \mid c_i) \log\left(P(c_{i+1} \mid c_i)\right),$$

即$-\mathrm{Entro}(c)i+1$，根据 3.1 节的假设，每个字符的概率分布仅由其在单词中的前一个字符决定。

一个字符的纯度越高，从该字符推断出的下一个字符的数量就越少，出现某些特定字符的概率就越高。也就是说，猜对下一个字符的概率就越大，即下一个字符的不确定性就越小，如图 3 所示。概率和概率是由来自 bestwordlist.com[6]的 5 个字母单词生成的，没有任何频率加权。

单词$i$is 的纯度简单定义为

$$w_i = \sum_{j=1}^{n} \mathrm{Pur}\left(c_{i,j}\right).$$

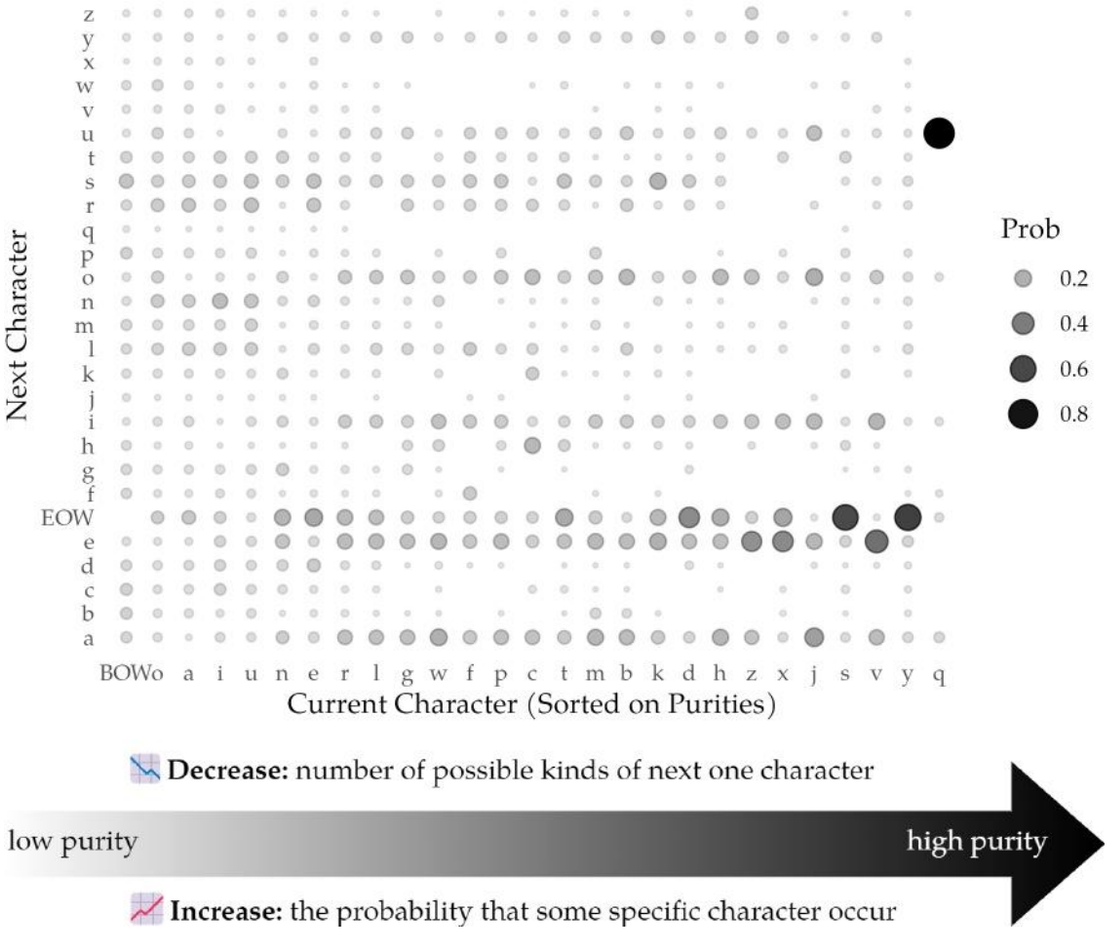

图 3:纯洁性和后继性

3.3 重复

单词的重复指数包括两个项目:重复字母的次数和重复字母的最大次数。重复字母定义为在一个单词中出现 2 次以上或等于 2 次的字母。重复字母的次数是指一个单词中出现重复字母的次数。例如，单词“apple”中除了“p”之外没有重复字母，所以单词“apple”中重复字母的个数为 1，而单词“可可豆”中的字母“c”和“a”都是重复字母，所以单词“可可豆”中重复字母的个数为 2。最大重复字母数指的是一个单词中重复字母的最大数量。比如，“木乃伊”中的“m”重复了 3 次，那么“木乃伊”中字母的最大重复次数是 3 次，“可可”中的“c”和“a”重复了两次，那么“可可”中字母的最大重复次数是 2 次，而没有重复字母的单词中字母的最大重复次数和字母的最大重复次数是 0 次。

单词重复对 Wordle 难度的影响是明显的:Wordle 只会提供关于答案单词是否包含输入字母以及该字母是否在正确位置的信息，而不会提供该字母是否重复或重复次数的信息。如果答案单词包含重复的字母，它可能会误导玩家尝试其他字母，从而增加尝试次数，使 Wordle 更加困难。同时，如果重复的字母重复的次数越多，Wordle 提供的答案单词信息就会进一步减少，从而再次增加 Wordle 的难度。

3.4 频率

这是一个合理的假设，即人们会优先考虑他们最熟悉的可用单词空间中的单词，而不管选择是否受到可用信息的限制，因此日常使用中的单词频率是评估单词难度和预测尝试分布的一个有价值的因素。

3.5 特征生成

对于词频，我们简单地使用了 Dave Hermit[2]的频率列表，他的语料库来源是电影字幕

规律性是在 bestwordlist.com 上的一个 5 个字母的单词列表上进行训练的[6];生成频率加权版本和未加权版本;值得注意的是，在训练加权版本时，使用 log 函数来避免 float 下溢，因此将 0 的频率设置为 1，以避免算术问题。

Purity 是在一些单词列表上用与 regular 相似的方法进行训练的。

4 对报告结果的数量进行建模

4.1 直觉:趋势中有什么

为了全面了解报告结果的数量是如何变化的，我们手工分析了趋势，并得出了一些简单的结论。

首先，趋势中不存在肉眼可见的周期性。考虑到小游戏的特点，周频率周期性是可能的，但我们在使用 f 检验时发现周频率下没有周期性。

整体趋势由一个相对陡峭的上升段和一个长尾的下降段组成，序列相当平滑。这些属性太过明显，不容忽视。这种趋势符合疾病流行中受感染个体数量的典型序列的形状。而且从机制上讲，玩《world》和分享分数的行为的传播可能非常具有病毒性，所以我们采用类似流行病学的方法来对这一过程进行建模。

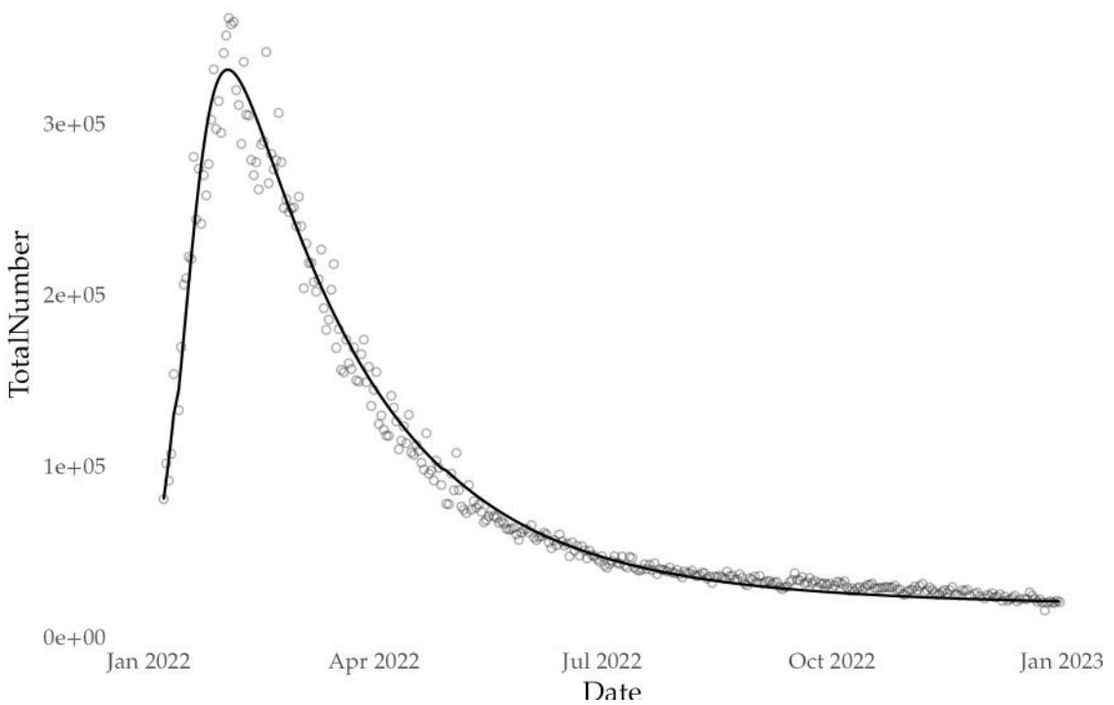

图 4:报告分数的数量和 Our Model 的拟合曲线

4.2 术语和假设

4.2.1 术语准备

我们对本节的术语定义如下:

TP 推特游戏:玩《world》并发布推特的行为;对于那些选择 TP 的人。

4.2.2 假设

行为的一致性 TPers 每天都做 TP，而非 TPers，包括那些从 TPers 转变过来的人，从不做 TP。

病毒式营销非 TPers 转化为 TPers 的概率取决于他们可能接触的 TPers 的数量。

潜在客户虽然一些非 TPers 有可能转变为 TPers，但其他非 TPers 可能无法成为 TPers。初始潜在客户的数量用本节Nin 表示。

从 TPers 转换而来的非 TPers 永远不会再变成 TPers。

4.3 从 SIR 模型到我们的 PCQL 模型

4.3.1 SIR 模型回顾

一个基本的 SIR 模型由四个微分方程组成[3]:

$$\frac{\mathrm{d}S(t)}{\mathrm{d}t} = \frac{-\beta \times S(t) \times I(t)}{N},$$
$$\frac{\mathrm{d}I(t)}{\mathrm{d}t} = \frac{-\beta \times S(t) \times I(t)}{N} - \gamma \times I(t),$$
$$\frac{\mathrm{d}R(t)}{\mathrm{d}t} = \gamma \times I(t),$$

其中 S 表示易感人群;I 是指受感染人群;R 是恢复的人群。这可以很好地匹配我们的问题，因为我们可以将 S 视为目标客户，将 I 视为分享者，将 R 视为尝试过并厌倦了这种行为的人。

然而，这里还是有一些不同于流行病的地方:虽然大多数分享分数的玩家可能只是在赶时髦，但并不是所有的分分者都会很快被治愈;一些分享分数的人会养成这个习惯，把玩游戏和分享分数当成自己的日常任务。所以我们要在 SIR 模型中添加一些新的东西。

4.3.2 我们的 PCQL 模型

我们的 PCQL 模型是一个带有 4 个变量和 4 个参数的 ODE 模型，即:

可变 P(potential)那些可能成为 TPers 的人，有机会成为人群。

变量 C(人群)一般的 TPers，他们很快就会感到无聊;他们中的一小部分每天都会成为退出者或忠诚者。

可变 Q(united)那些厌倦游戏或上传分数的人。

可变 L(忠诚)那些不会轻易放弃 TP 的人。

参数β参与因子)(参与是指参与 TP)，表示 TP 对电位的吸引力，单位时间内 t 时刻吸引和参与的数为人$\beta \times P(t) \times (C(t)+L(t)) / N$

.

参数 γ(厌倦系数)表示被吸引去做 TP 的人厌倦 TP 的比率，在单位时间内，在 t 时刻成为 TP 者的退出人数为$\gamma \times (t)$。

参数λ(转换因子)表示人群 TPers 成为忠诚 TPers 的可能性;单位时间内 t 时刻 TPers 转化为忠诚 TPers的数量为$\lambda \times .(t)$。

参数ϕ(Loyal Player BoredomCoefficient)表示世界大战忠诚 TPers 厌倦 TP 的比率，在单位时间内 t 时刻不再做 TP 的忠诚 TPers 的数量为$\phi \times L(t)$。

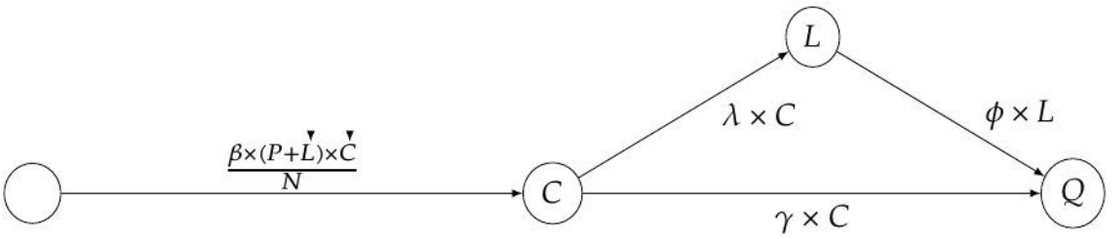

图 5:每时间步平均过渡量

图 5 给出了模型的直观印象。

$$\frac{dP(t)}{dt} = \frac{-\beta \times P(t) \times (C(t) + L(t))}{N},$$
$$\frac{dC(t)}{dt} = \frac{\beta \times P(t) \times (C(t) + L(t))}{N} - \gamma \times C(t) - \lambda \times C(t),$$
$$\frac{dQ(t)}{dt} = \gamma \times C(t),$$
$$\frac{dL(t)}{dt} = \lambda \times C(t) - \phi \times L(t).$$

4.4 模型拟合与预测

该模型使用 MSE 作为损失函数来估计模型参数。由于微分方程系统的形式解难以计算，因此不能用传统的最小二乘方法估计，因此只能直接计算微分方程的数值解，并使用优化算法最小化 MSE。同时，由于常规算法无法利用微分方程计算预测区间，因此采用 Bootstrap 方法获得点预测和预测区间。

Bootstrap 方法需要大量的计算，并且使用优化算法进行拟合的时间成本较高，因此我们选择了计算速度更快的 Nelder-Mead 方法。由于 Nelder-Mead 算法需要指定函数的初始解，因此我们首先使用遗传算法[5]获得较好的初始解，然后使用 Nelder-Mead 方法选择使 MSE 最小的参数，遗传算法获得的初始参数如下表所示。从这组参数得到的模型曲线如图 4 所示。

β	γ	λ	ϕ
1.77e-01	1.77e-02	1.04e-03	1.14e-03

生成 1000 个 Bootstrap 样本，在每个样本中使用遗传算法获得的初始参数作为 Nelder-Mead 方法的初始解，拟合以最小化 MSE 参数并预测 2023 年 3 月 1 的报告数量，得到 1000 个 Bootstrap 预测。将得到的预测值按升序排列，选取 2.5%和 97.5%四分位数的预测值作为 95%预测区间的上界和下界。得到的预测区间如下。

Mean	2.5%	97.5%
14689.9	11173.04	17069.15

因此，我们估计 2023 年 3 月 1 日的报告数量包含在[11173.04,17069.15]区间内，置信水平为 95%。

5 Word 本身会影响困难模式比例吗

5.1 这似乎是真的

我们使用协方差矩阵来识别影响参与困难模式的玩家百分比的变量。我们统计单词对应的七个属性:词频、加权纯度、未加权纯度、加权规律性、未加权规律性、重复字母数和最大重复字母数与参与困难模式的玩家百分比之间的相关系数，结果如图 6 所示。

协方差矩阵显示，除了加权规则和非加权规则外，单词的所有属性都与参与困难模式的玩家百分比无关。因为玩家在玩《世界大战》之前并不知道这些单词，所以大多数单词的属性与参与困难模式的玩家比例没有显著相关性是合理的。

困难模式中参与者百分比的加权和未加权规律之间存在相对显著的相关性，这似乎意味着世界的属性可以在一定程度上影响困难模式中参与者的百分比。然而，我们使用加权规则性和未加权规则性分别对模式的参与百分比进行了线性回归，结果如下表所示。

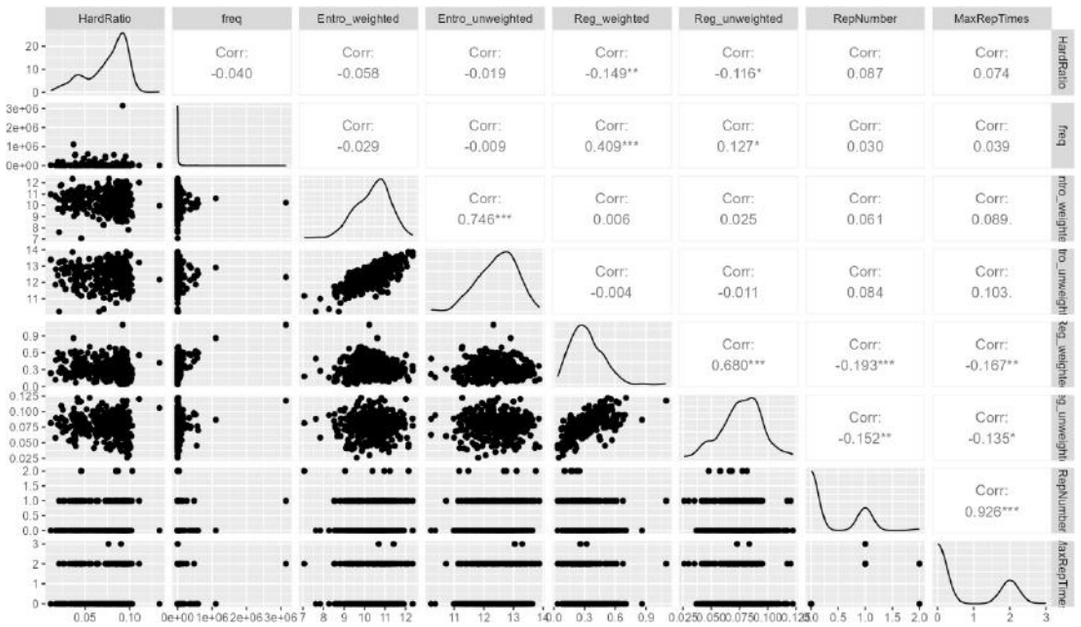

图 6:协方差矩阵

表 5:硬模式百分比对加权正则性的回归结果

	Estimate	Std.Error	t value	Pr(>|t|)	
(Intercept)	0.082177	0.00274	29.987	<2e-16	***
Reg_weighted	-0.021913	0.007788	-2.814	0.00517	**
Multiple R-squared: 0.02212					
Adjusted R-squared: 0.01933					

表 6:硬模式百分比对未加权正则性的回归结果

	Estimate	Std.Error	t value	Pr(>|t|)	
(Intercept)	0.082177	0.00274	29.987	<2e-16	***
Reg_unweighted	-0.148223	0.067780	-2.187	0.0294	**
Multiple R-squared: 0.01348					
Adjusted R-squared: 0.01066					

加权规则性与硬模式百分比回归模型的 R2 和 Adjust R2 仅为 0.022 和 0.019，而未加权规则性与硬模式百分比回归模型的 R2 和 Adjust R2 仅为 0.013 和 0.011，这意味着加权规则性仅解释了约 2%的挑

战模式百分比变化。而未加权的规律性只能解释大约 2%的硬模式变化。未加权的规律性只能解释大约 1.2%的硬模式百分比变化，这不足以解释。此外，加权规则性的回归系数为-0.022，未加权规则性的回归系数为-0.01，而数据集中观察样本的加权规则性 95%集中在 0.081 ~ 0.612 和未加权规则性的范围内 95%集中在 0.041 ~ 0.109 的范围内，这意味着数据集中加权规则性和未加权规则性的极端变化对硬模式百分比的平均影响仅为 1%。这意味着加权规则和未加权规则的极端变化对数据集中挑战模式百分比的平均影响仅为 1%，与挑战模式百分比的平均值 7.52%相比，这一影响并不显著。

5.2 It is Very Likely Not True

除了缺乏解释力外，显著相关最有可能是由于伪回归。下表显示了加权规则性和硬模式对时间的百分比回归结果。

表 7:硬模式百分比对时间的回归结果

	Estimate	Std.Error	t value	Pr(>|t|)	
(Intercept)	-2.893e-04	1.751e-03	-0.165	0.869	***
Date	1.988e-04	4.453e-06	44.655	<2e-16	***
Multiple R-squared: 0.8507					
Adjusted R-squared: 0.8503					

表 8:加权正则性对时间的回归结果

	Estimate	Std.Error	t value	Pr(>|t|)	
(Intercept)	0.3814810	0.0305641	12.481	<2e-16	***
Date	-1.668e-04	7.77e-05	-2.146	0.0325	*
Multiple R-squared: 0.01299					
Adjusted R-squared: 0.01017					

回归结果表明，硬模式百分比与时间呈正相关，而加权正则性与时间呈负相关，二者之间的负相关很可能是由时间趋势引起的，而不存在因果关系。

总之，我们得出的结论是，加权规则和非加权规则与硬模式百分比之间的显著相关性并不能证明单词属性影响玩家是否选择硬模式，这一发现也与我们之前的假设一致，即玩家在参与《world》之前不知道谜题单词。

6 预测分数分布

在预测报告分数的分布时，我们需要保证分数之和等于 100%，而 k -近邻回归方法可以很好地满足这一约束，因此我们选择 KNN 回归来预测分数的分布。

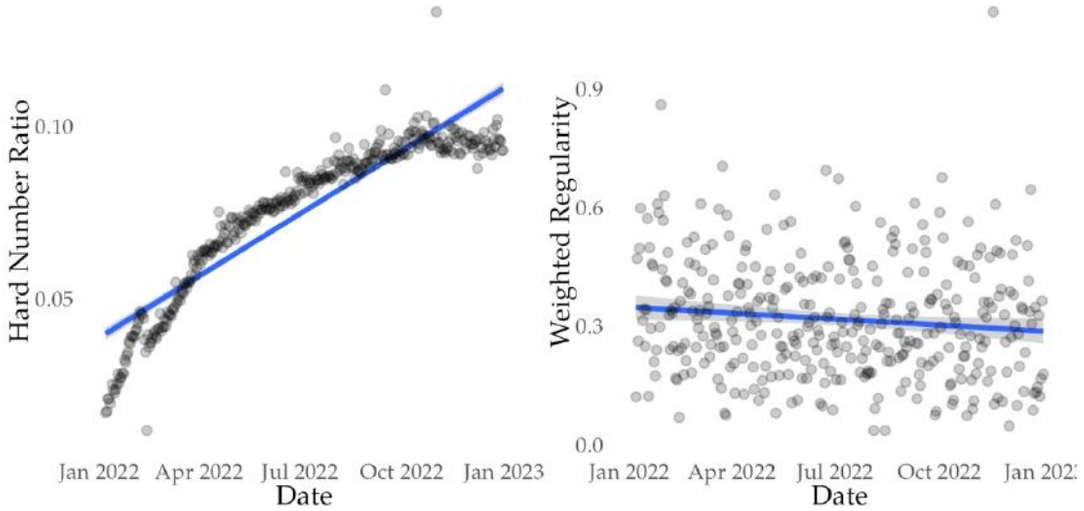

图 7:硬模式报告率及加权规律时间趋势

6.1 KNN 回归

KNN 是一种典型的非参数方法，其基本思想是计算训练集中观测值之间的距离，选择与观测值最接近的 K 个观测值，取其对应因变量的平均值作为模型的拟合值，其计算方法如下:

$$\hat{f}(x_0) = \frac{1}{K} \sum_{x_i \in N_0} (f(x_i)),$$

其中$N0$ is 为最接近于 s1 的 K 个观测 0 值的集合，f^2(0s1)为模型的拟合值。

由于 KNN 使用因变量的平均值作为预测，因此它保证了 7 个分数与观测值拟合后得到的预测值总和仍然为 100%。

6.2 特征选择和参数调优

由协方差矩阵可知，猜词得分的分布与 Hard Mode 结果数、词频、重复字母数、词汇纯度、词汇加权纯度、词汇规则性、词汇加权规则性、重复字母最大重复次数有显著相关。还可以看出，词汇的纯度与词汇的加权纯度、词汇的规律性与词汇的加权规律性之间存在着显著的相关关系，重复使用可能导致预测效果不佳;同时，由于我们不知道预测中 Hard Mode 人数的数据，如果我们使用 Hard Mode 人数的预测值，预测结果将是不可靠的，所以我们需要进行

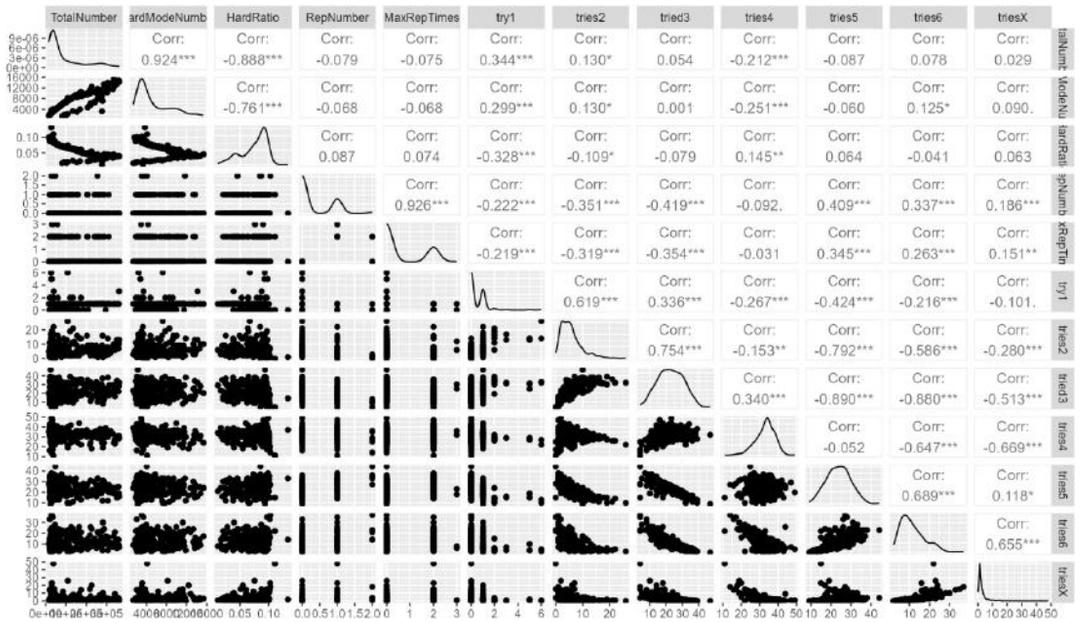

图 8:协方差矩阵(1)

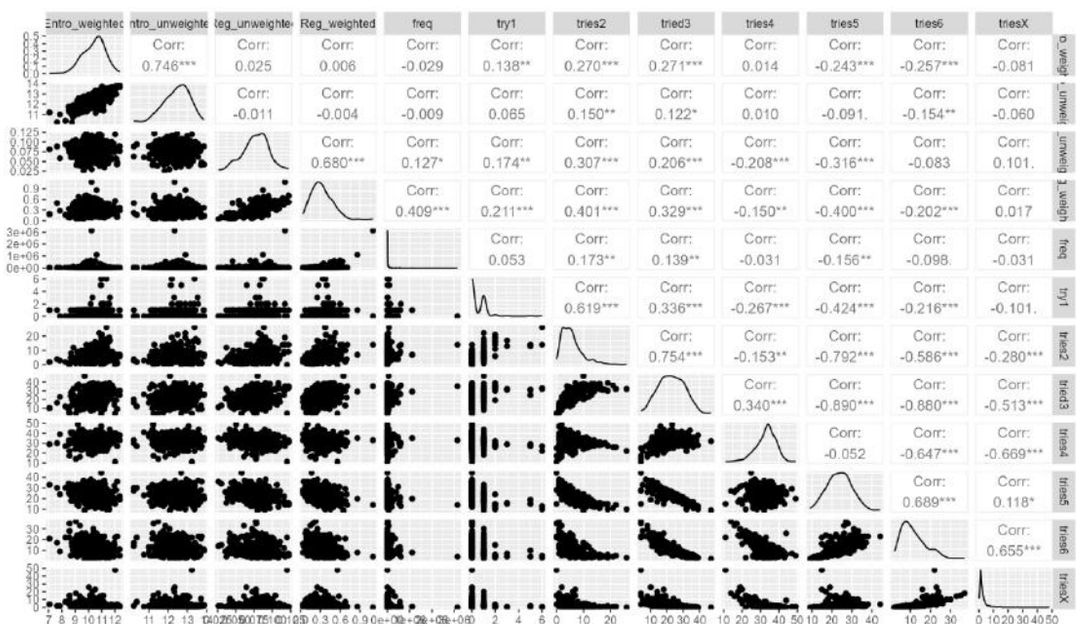

图 9:协方差矩阵(2)

进一步在这五个变量中进行过滤。另外，从 KNN 方法的思想可知，K 的选择决定了模型的性能。我们使用交叉验证方法计算不同自变量的不同 K 值的 KNN 模型的预测误差，并将交叉验证误差最小的 K 值用于模型。为了确保适度的计算工作量，我们使用了十倍交叉验证。

在进行拟合之前，为了消除幅度对观测值之间距离的影响，我们使用以下公式对自变量进行标准化，将每个自变量的平均值标准化为 0，将方差标准化为 1。

$$X_i^* = \frac{X_i - \overline{X}}{sd(X)}$$

交叉验证结果如图 10 所示。

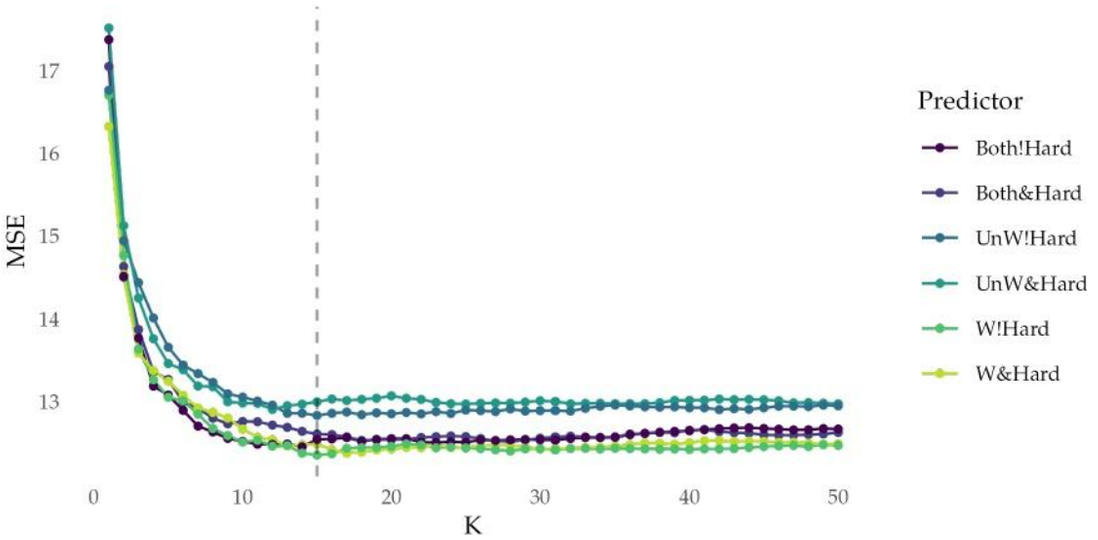

图 10:交叉验证结果

*：“&”包括在内而“！”不包括；“UnW”是未加权的单词特征；“W”是单词频率加权的单词特征。

从图中可以看出，在 K=15 时，使用加权正则性和加权纯度且不考虑挑战模式数量的 KNN 模型表现最好。因此，我们选择词频、重复字母数、单词加权纯度、单词加权规律性和重复字母的最大重复次数作为模型自变量，并设置模型 K=15。模型的 MSE 为 12.35。

6.3 预测和 Bootstrapping

现在我们确定 EERIE 的每个属性。

Frequency	3264
Weighted Purity	10.86
Weighted Regularity	0.32
Number of Repeated Word	1
Max Times of Repeated	3

将 EERIE 的属性输入到 KNN 模型中，得到其预测分布如下表所示。

1 try	2 tries	3 tries	4 tries	5 tries	6 tries	X
0.20	4.87	23.55	35.36	23.75	9.94	2.33

我们使用 Bootstrap 方法计算 KNN 模型的预测区间，得到了每个分数预测值的 95%置信区间，如图 11 所示。

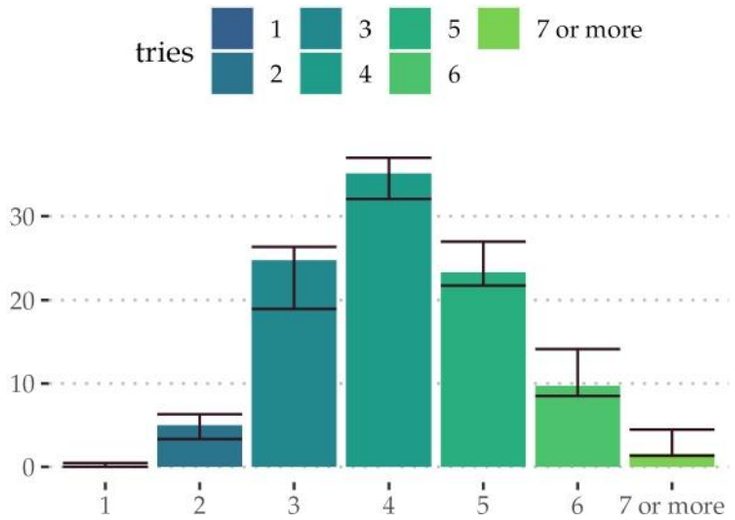

图 11:EERIE 和 95%置信区间的预测尝试分布

可以看出，模型的误差范围很小。

7 难度分类

7.1 难度评价

世界大战的难度可以通过玩家猜测谜题单词的次数来衡量，尝试次数越高，世界大战的难度就越高。我们可以从数据集中得到最终尝试次数的分布，我们定义尝试次数的中位数来衡量世界的难度。中位数尝试次数指的是玩家在分数分布的 50%处尝试猜单词的次数。“abbey”的尝试次数中位数是 5 次。

由于中位数为 6 的单词只有三个，我们将中位数为 6 的单词与中位数为 5 的单词组合在一起，得到三个难度等级:简单、正常和困难，分别对应于中位数为 3、4 和 5 或 6。

7.2 难度预测

我们使用协方差矩阵对难度对应的单词属性进行过滤，并过滤出重复字母的数量、加权纯度和加权规律性作为与分类相关的属性。我们使用 KNN 来验证分类的准确性。

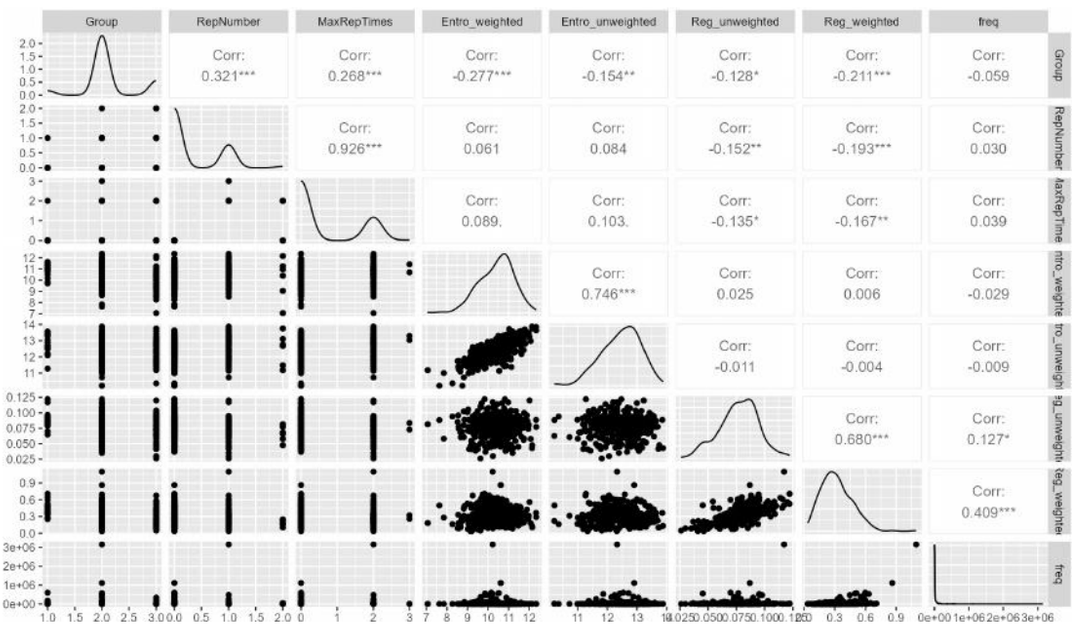

图 12:协方差矩阵

我们使用交叉验证以准确性(即正确预测的数量占所有预测的百分比)作为选择标准来获得 KNN 的最佳 K 值，交叉验证的结果如图 13 所示。

我们选择 K=8 作为参数，预测准确率为 75.28%。

我们将 EERIE 的三个数据代入，得到 EERIE 的预测结果

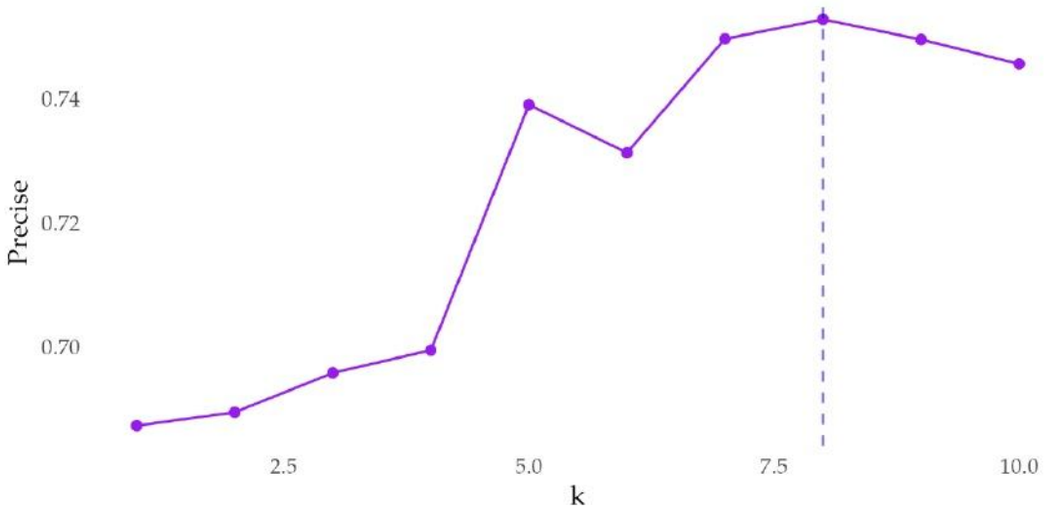

图 13:交叉验证的结果

属于普通难度。基于 K=8 时模型的交叉验证误差，我们认为该预测的准确率约为 75.28%，准确率区间为[63.16%，82.76%]。

8 其他有趣的见解

如图 14 所示，随着时间的推移，困难模式占报告总数的比例逐渐增加。事实上，这一现象与我们的 PCQL 模型是一致的，在 PCQL 模型中，我们假设存在忠实的 TPers，即忠实玩家，并且忠实玩家的比例会随着时间的推移而增加，因为忠实玩家比普通玩家退出游戏的可能性要小得多，如图 15 所示。我们可以假设忠诚玩家的比例偏好挑战，所以困难模式比例的增加一定程度上反映了忠诚玩家比例的增加。

第四次猜中字数的比例为词汇难度分界点。协方差矩阵显示，随着与难度正相关的指标增加，第 4 次至第 7 次猜中的字谜比例增加，第 1 次至第 3 次猜中的字谜比例减少。

如下图所示，玩家第一次猜对的比例与总报告分数正相关，这可能是由于随着报告次数的增加，人们从社交媒体上提前知道谜题的概率增加，从而导致第一次猜对的比例与总报告分数正相关。同时，玩家第一次猜测的比例与含有重复字母的谜题单词的数量呈显著负相关，这反映了玩家在玩《世界》时，使用没有重复字母的单词作为初始猜测的可能策略，因为这样可以尽可能多地消除不正确的字母和位置。

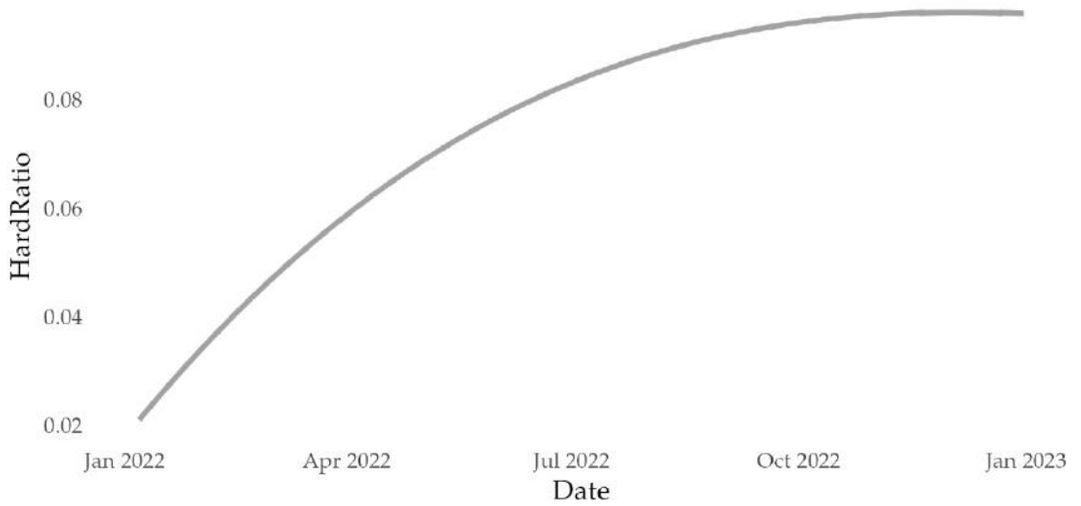

图 14:硬模占总报告数的比例

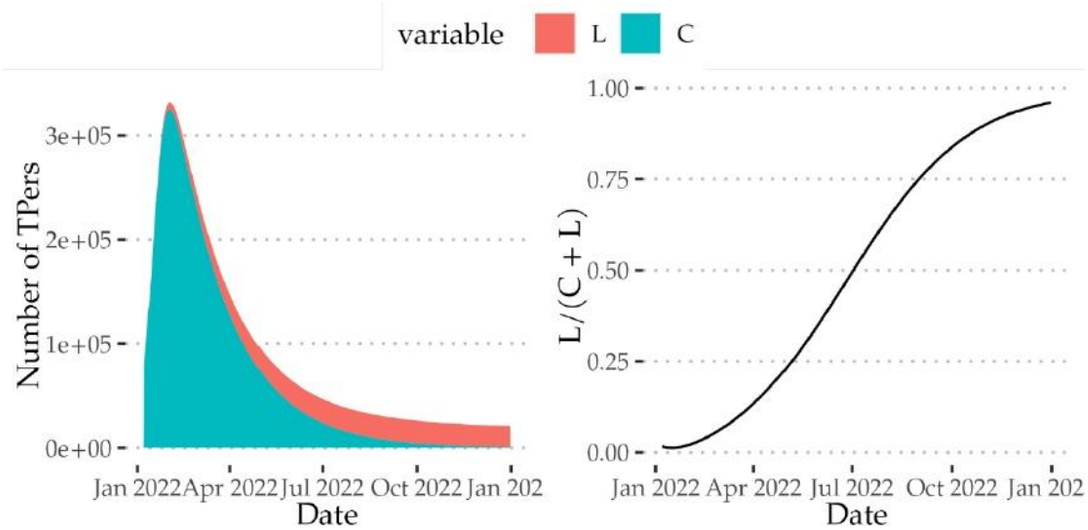

图 15:我们的模型中 Ls 占 TPs 的比例，参数如前所述

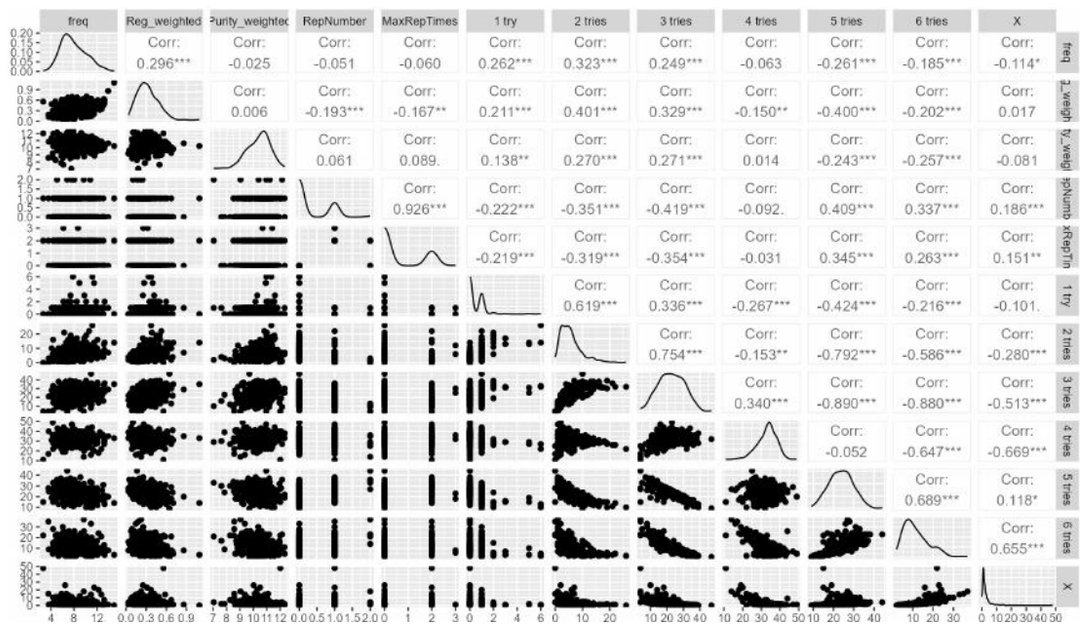

图 16:协方差矩阵

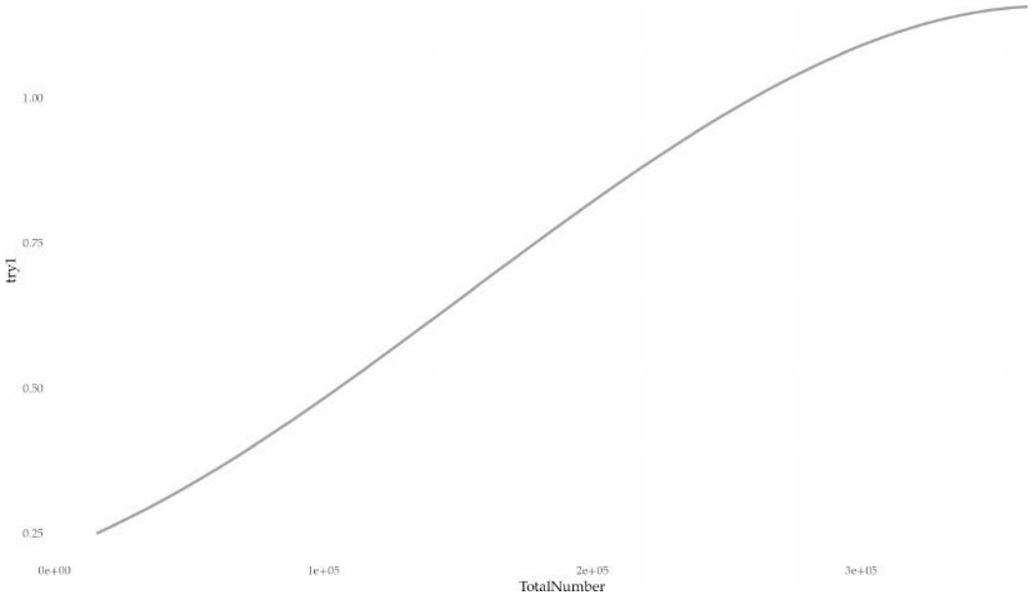

图 17:第一次尝试猜对的比例和报告分数总数

References

[1] Markov A. A. “An Example of Statistical Investigation of the Text Eugene Onegin Concerning the Connection of Samples in Chains.” In: Science in Context 19 (2007), pp. 591 – 600. ISSN: 1474-0664.

[2] Hermit Dave. GitHub - hermitdave/FrequencyWords: Repository for Frequency Word List Generator and processed files. GitHub. URL: https://github.com/hermitdave/FrequencyWords (visited on 02/18/2023).

[3] W. O. Kermack and A. G. McKendrick. “A Contribution to the Mathematical Theory of Epidemics.” In: Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character 115.772 (1927), pp. 700 – 721. ISSN: 09501207.

[4] R Core Team. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing. Vienna, Austria, 2022. URL: https://www.R-project.org/.

[5] Luca Scrucca. “GA: A Package for Genetic Algorithms in R”. In: Journal of Statistical Software 53.4 (2013), pp. 1 – 37. DOI: 10.18637/jss.v053.i04.

[6] Word Lists - 5-Letter Words. URL: https://www.bestwordlist.com/5letterwords.htm (visited on 02/18/2023)