数据挖掘大作业最终报告

——选题《阿里音乐流行趋势预测》

唐育洋 2120151034

敖 权 2120150974

付升宇 2120150984

简介

本次项目的题目来源于天池大数据比赛中的阿里音乐流行趋势预测大赛。 下面对本次项目做简单介绍。

经过7年的发展与沉淀，目前阿里音乐拥有数百万的曲库资源，每天千万的用户活跃在平台上，拥有数亿人次的用户试听、收藏等行为。在原创艺人和作品方面，更是拥有数万的独立音乐人，每月上传上万个原创作品，形成超过几十万首曲目的原创作品库，如此庞大的数据资源库对于音乐流行趋势的把握有着极为重要的指引作用。

本次项目以阿里音乐用户的历史播放数据为基础，通过对阿里音乐平台上每个阶段艺人的试听量的预测，挖掘出即将成为潮流的艺人，从而实现对一个时间段内音乐流行趋势的准确把控。通过获取的阿里音乐数据，我们首先将进行数据预处理，然后将数据可视化，以便能够得到一些有用的信息。然后使用时间序列预测算法进行预测得到结

果，并对结果进行评价。

问题描述

项目能够获得开放的抽样的歌曲艺人数据，以及和这些艺人相关的 6 个月内（20150301-20150830）的用户行为历史记录，包括 100 个艺人，超过 26000 首歌曲，15000000 条用户行为记录。数据表如下，每一行表示一条行为记录，所有可能涉及到隐私的数据在获取之前均已进行脱敏处理。行为记录包括用户 ID、歌曲 ID、用户播放的的精确小时时间、行为类型(播放、下载、收藏)、记录收集日期等。

表 1.用户行为记录表

列名	类型	说明	示例
user_id	String	用户唯一标识	7063b3d0c075a4d276c5f06f4327cf4a
song_id	String	歌曲唯一标识	effb071415be51f11e845884e67c0f8c
gmt_create	String	用户播放时间（unix时间戳表示）精确到小时	1426406400
action_type	String	行为类型：1，播放；2，下载，3，收藏	1
Ds	String	记录收集日（分区）	20150315

同时，也提供了歌曲艺人的数据表，包括歌曲 ID、艺人 ID、歌曲发行时间（精确到天）、歌曲初始播放次数、歌曲语言、艺人性别或者组合等。通过这个表的信息，我们可以得到上面用户操作的每首歌曲相关的艺人信息。

表 2.歌曲艺人信息表

列名	类型	说明	示例
song_id	String	歌曲唯一标识	c81f89cf7edd24930641afa2e411b09c
artist_id	String	歌曲所属的艺人Id	03c6699ea836decbc5c8fc2dbae7bd3b
publish_time	String	歌曲发行时间，精确到天	20150325
song_init_plays	String	歌曲的初始播放数，表明该歌曲的初始热度	0
Language	String	数字表示1,2,3…	100
Gender	String	1,2,3	1

我们需要做的就是预测在接下来的两个月时间，也就是 20150901-20151030 范围内的每个艺人每天的歌曲播放次数，按照如下表格格式进行提交。

表 3.提交的预测表

列名	类型	说明	示例
artist_id	String	歌曲所属的艺人Id	023406156015ef87f99521f3b343f71f
Plays	String	艺人当天的播放数据	5000
Ds	String	日期	20150901

技术方案

在本次项目中，我们将使用到的分为数据预处理、数据可视化、时间序列预测三部分。

数据预处理

由于获得的数据是不分用户、不分歌曲、不分艺人的，而最后需要按照艺人来进行预测，因此，我们首先将用户行为数据按照艺人进行分割，每个艺人相关的用户行为数据独立为一个文件，格式上与原用户行为表相同。同时，为了寻找更多有效规律，我们也按照小时、星期做了数据统计

数据可视化

根据数据，按照艺人绘制真实行为记录统计，举例如下：

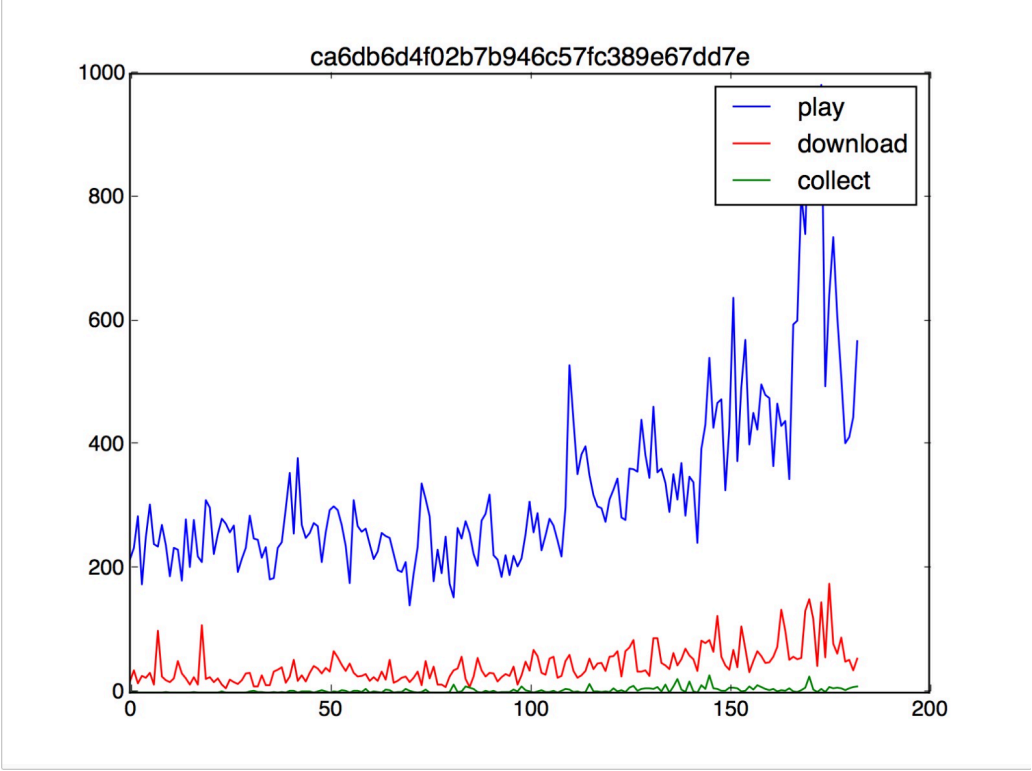

图 1.某艺人真实用户行为统计举例

按照时间，绘制用户行为记录统计图，如下：

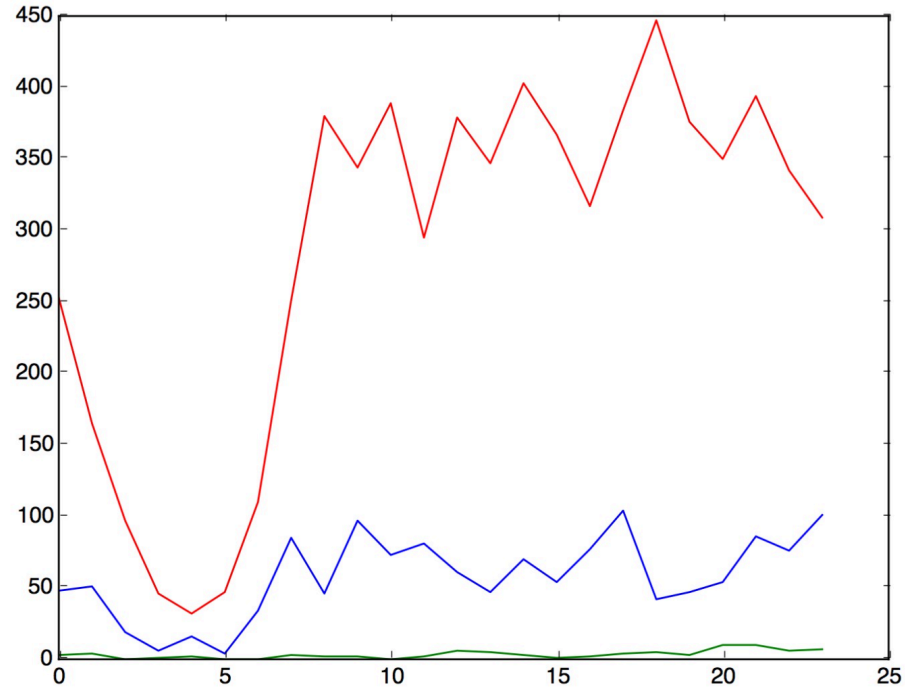

图 2.某艺人按小时统计的用户行为数据举例

时间序列预测

本次实验采用 STL 时间序列分解预测算法，STL 分解算法将时间序列分解为三个分量，分别为：趋势项、季节项、残余项。在应用 STL 分解算法之前，需要对序列预处理得到平稳序列，在这里，我们可以简单理解为序列的均值没有系统的变化（无趋势）、方差没有系统变化，消除了周期性变化。通过对序列做差分操作可以得到平稳序列。

首先我们需要将原始时间序列处理为平稳时间序列，下面给出了原始时间序列和处理后的一阶差分时间序列：

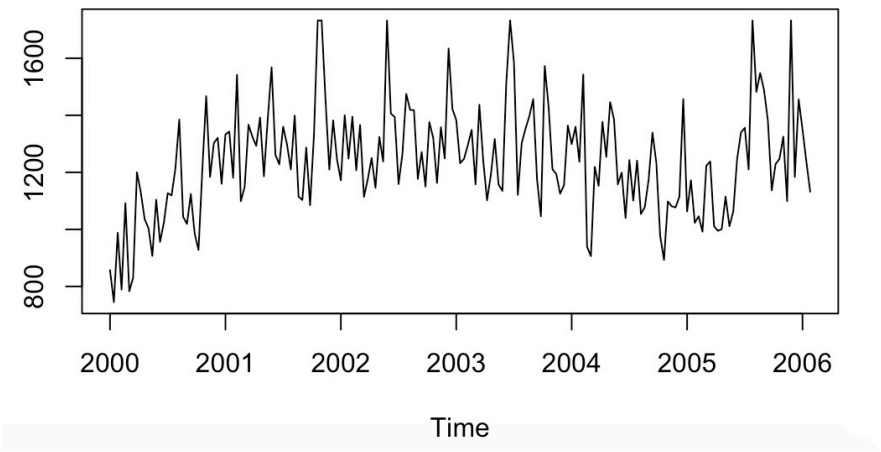

图 3.原始时间序列

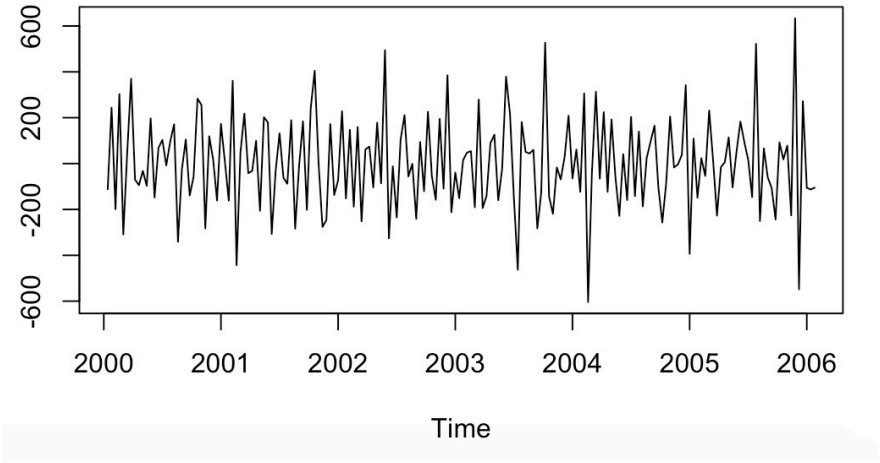

图 4.一阶差分时间序列

从图 4 我们认为，处理后的时间序列可以看作是平稳的，符合预测算法的输入要求。接下来进行预测。

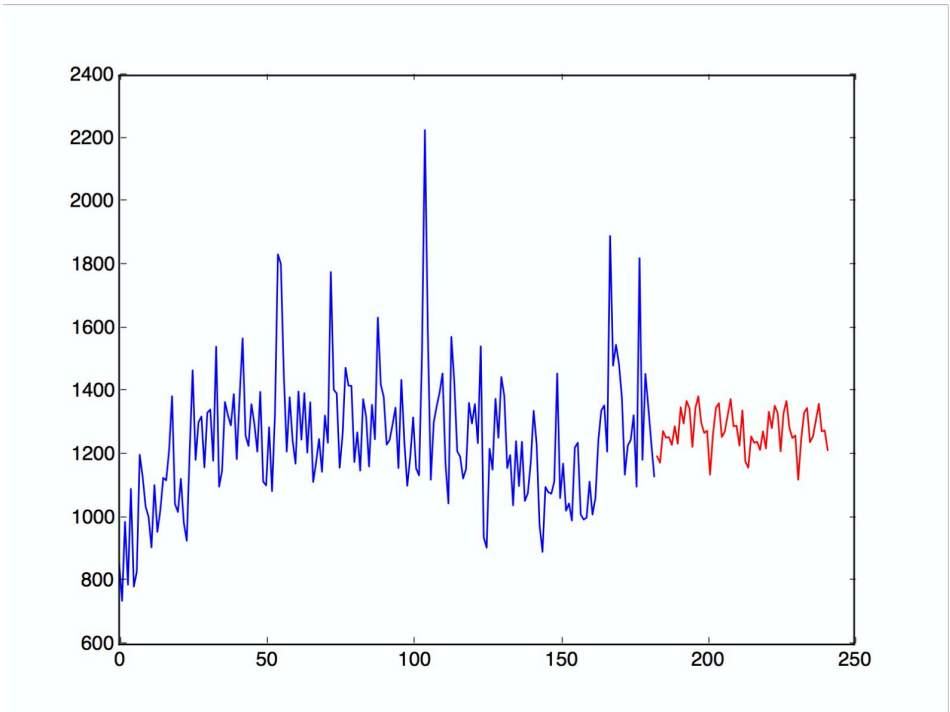

图 5.一个真实的预测结果，蓝色为输入部分，红色为预测部分。

实验和结果

在本次项目中，为了能够得到更好的预测效果，我们进行了多次实验，针对表现出来的问题，我们不断进行了优化。例如，设置一个阈值，如果一阶差分平稳性不够，采用更高阶的差分序列；我们也观察到在波峰波谷位置处预测偏差较大，我们猜想这是由于该艺人发布了某个新歌或新专辑的原因，但是这种属于在一直信息内无法预测的意外事件，因此我们对波峰波谷差异较为明显的进行了平滑处理，使得预测序列不会太突兀；对于那些预测序列的变化趋势，即一阶差分序列，明显不符合之前观察序列的，我们也做了特殊处理；另外，如果预测值为负则直接按零值计算；另有一个小细节，观察数据没有 8 月 31 日的数据，但我们需要预测，即实际预测 61 天，并将第一抛弃，剩余 60 天作为结果文件保存。

对于预测结果，我们采用的是该比赛的评价方式。设艺人 j 在第 k 天的实际播放次数为$\mathrm{T}_{j.k}$，艺人集合为 W，我们根据计算预测得到艺人 j 在第 k 天的播放次数为$\mathrm{S}_{j.k}$，N 为预测的总天数，则对艺人 j 的播放预测和实际的方差归一化方差σ_{j}为

$$\sigma_{\mathrm{j}} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}((S_{j.k} - \mathrm{T}_{j.k})/(\mathrm{T}_{j.k}))^2}$$

艺人 j 所在的权重根据艺人的播放量平方根而定：

$$\varphi_{\mathrm{j}} = \sqrt{\sum_{k=1}^{N}\mathrm{T}_{j.k}}$$

则本次预测的得分为：

$$\mathrm{F} = \sum\nolimits_{j\in W}\left(1-\sigma_{\mathrm{j}}\right) * \varphi_{\mathrm{j}}$$

F 值越大，预测结果越好。

由于比赛方并未公布真实播放记录数据，因此评分只能有比赛方计算。我们的得分一直在 13300 左右变化，一直没有较大幅度改进。