Introduction to Big Data Final Report

陈文雁，王铎磊

12012437@mail.sustech.edu.cn,
wangdl2020@mail.sustech.edu.cn

目录

1 背景介绍

共享单车已经成为城市地区提供便捷和可持续出行选择的新一代交通方式。共享单车系统的可用性对交通管理、环境可持续性和公共健康具有着重要的影响。准确预测租赁单车数量可以为系统运营商、城市规划者和决策者提供有价值的信息，以优化资源分配、提高运营效率，并满足对共享单车服务日益增长的需求。在这个项目中，我们的目标是根据已有的共享单车租赁单车数量预测总租赁单车数量，包括临时租赁的数量和注册租赁的数量。

2 数据探索

2.1 数据描述

现有数据集 day.csv 和 hour.csv 通过阅读文档的描述以及通过代码 x 检测两个 csv 文件，我们得知了数据包含的每一列的含义以及对应的数据类型，如下图所示。

ColumnName	*Meaning*	*DataType*
instant	record index	int64
dteday	date	object
season	season (1:springer,2:summer,3:fall,4:winter)	int64
yr	year (0: 2011, 1:2012)	int64
mnth	month (1 to 12)	int64
hr	hour (0 to 23)	int64
holiday	weather day is holiday or not	int64
weekday	day of the week	int64
workingday	if day is neither weekend nor holiday is 1, otherwise is 0	int64
weathersit	weather conditions (1 to 4)	int64
temp	normalized temperature in Celsius	float64
atemp	normalized feeling temperature in Celsius	float64
hum	normalized humidity	float64
windspeed	normalized winds peed	float64
casual	count of casual users	int64
registered	count of registered users	int64
cnt	count of total rental bikes (casual and registered)	int64

表 1: 数据集中每一列含义以及数据类型

2.2 各项指标分布情况

考虑到 dteday 的数据类型是 object，并且第一列 instant 已经能够充分体现出数据的时序性，因此，我们分别对 hour.csv 和 day.csv 文件，针对除了 dteday 之外的指标进行了可视化，以 hour.csv 文件的可视化结果为例，具体如下图所示。

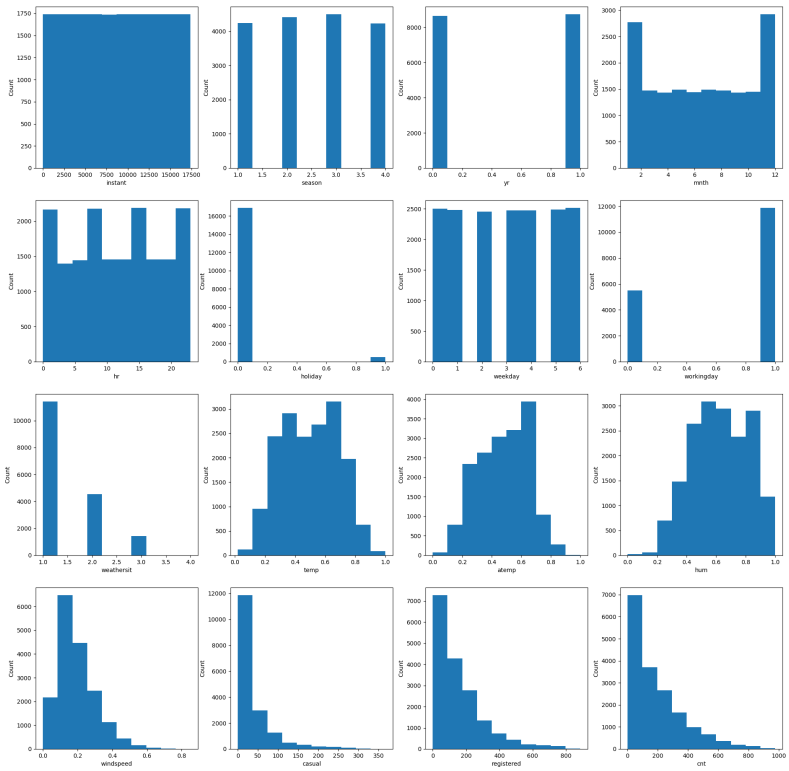


图 1: 各项指标分布情况

通过观察可以发现，casual、register 和 cnt 在 [0, 100] 之间分布紧密，所以，我们决定使用 log10 对 casual、register 和 cnt 进行对数变换，以 hour.csv 文件的可视化结果为例，具体如下图所示。可以观察到，经过对数变换之后，casual、register 和 cnt 这三项指标的表现较之前更趋向于正态分布。

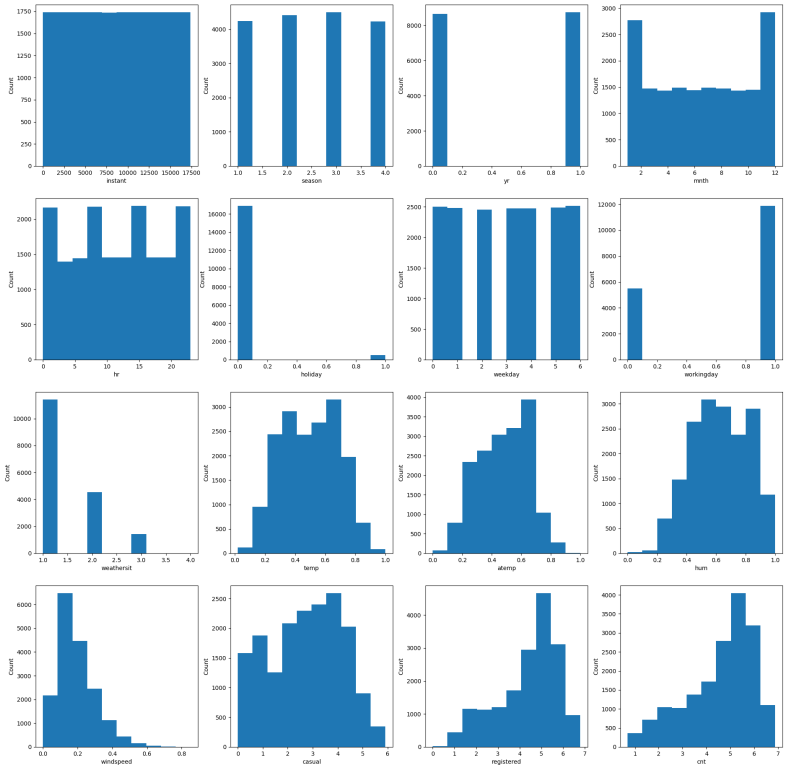


图 2: 各项指标分布情况 (经过对数变换之后)

2.3 单一指标与租赁数量分布情况

我们从众多的指标中选取了一天中每小时、星期以及季节，分别探究它们和租赁数量之间的关系。

2.3.1 一天中每小时与租赁数量

下图是一天中每小时与租赁数量的关系图。通过观察可以发现，对于 cnt 和 registered，它们在一天之内的变化规律基本保持一致，在 7:00-9:00(早高峰期间)、16:00-19:00(晚高峰期间) 数量达到峰值，12:00-14:00(午休期间) 数量有所减少。对于 casual，从早上到晚上，表现出先增加再减少的趋势，在中下午 (12:00-18:00) 保持较高值。

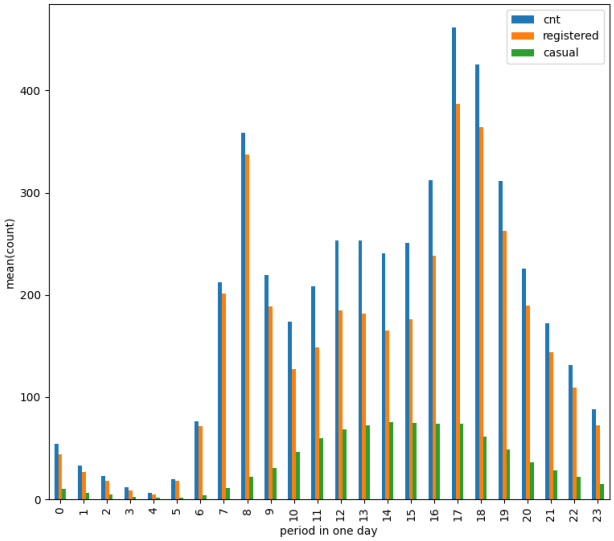


图 3: 一天中每小时与租赁数量的分布情况

2.3.2 一周每天与租赁数量

下图是一周中每天与租赁数量的关系图。通过观察可以发现，对于 cnt 和 registered，它们的在周中的数量明显高于周末，而对于 casual，它在周末的数量明显高于周中。

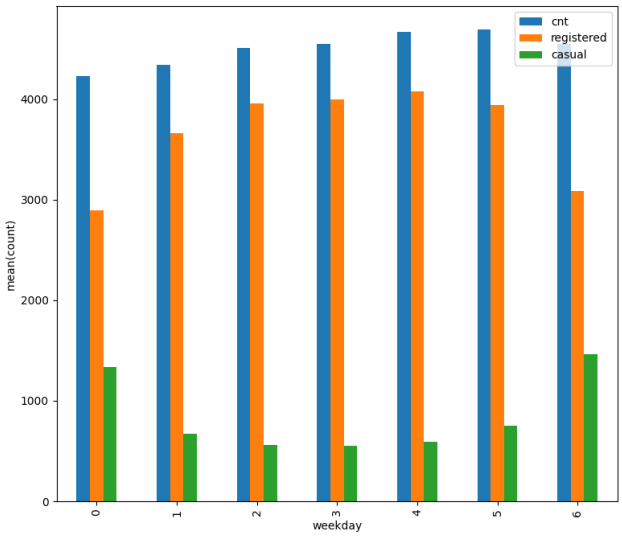


图 4: 一周中每天与租赁数量的分布情况

2.3.3 季节与租赁数量

下图是季节与租赁数量的关系图。通过观察可以发现，对于 cnt、registered、casual，它们的数量都表现为夏季秋季多，春季冬季少。

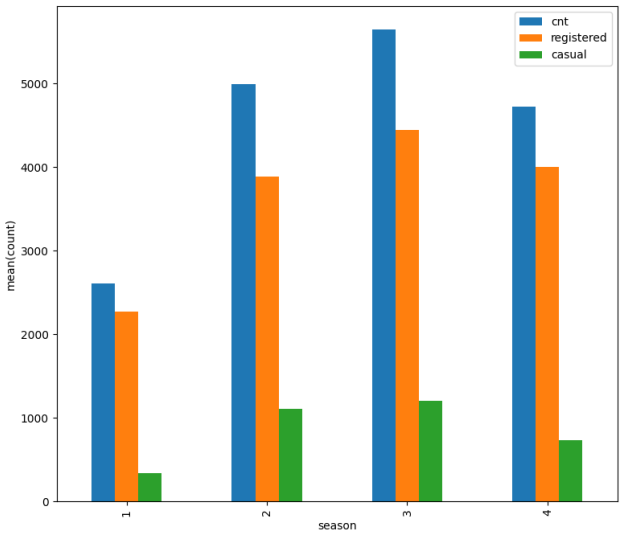


图 5: 季节与租赁数量的分布情况

2.4 租赁数量总体分布情况

考虑到数据中具有明显的时序特征，因此绘制图形的时候时间跨度不宜过大。首先给出了 2011，2012 年度'casual', 'registed', 'cnt' 的总体趋势。考虑到'cnt' 是两者的和，因此在数据探索的时候三者均进行了绘制，具体如下图所示。这里可以相对明显的看出，两年的数据是具有隐约的形状的。我们根据问题 1，2 和 3 的提示，进一步观察了以天、周、月为单位的变化趋势。

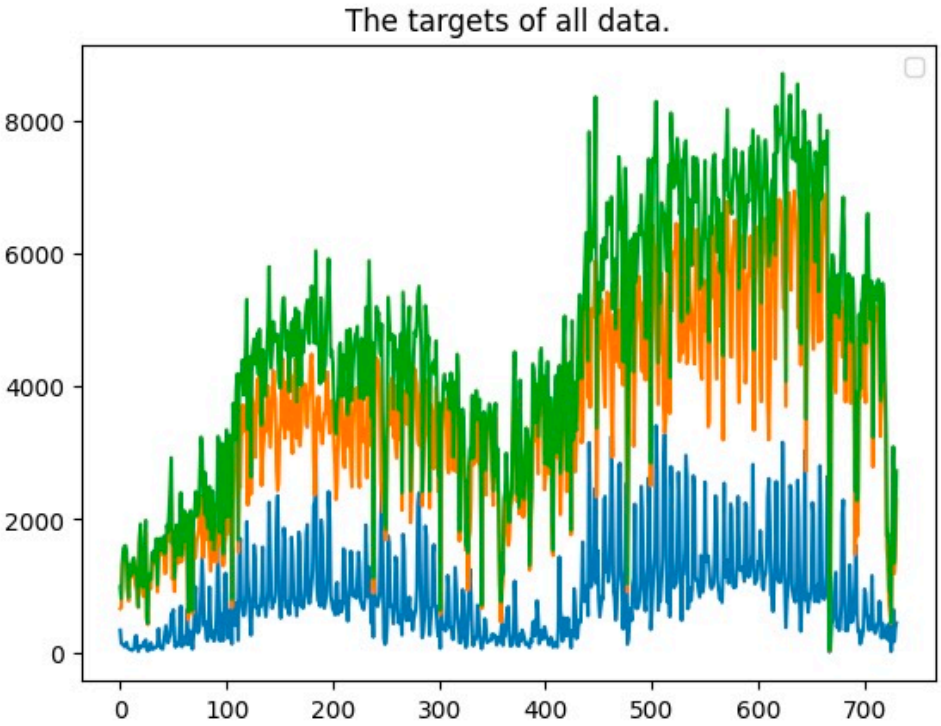
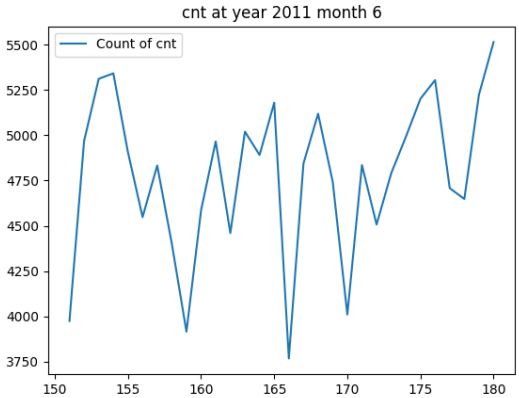


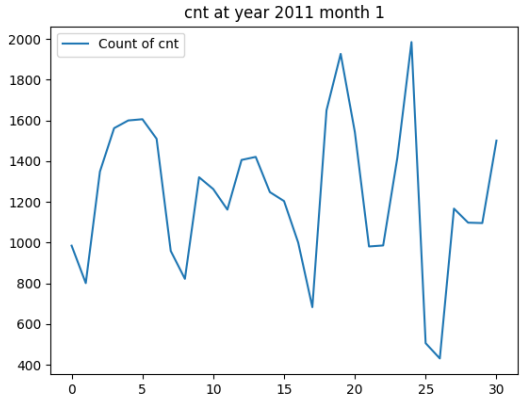
图 6: The targets of all data

3 问题分析

3.1 问题 1

第一个题目按照要求去除相关两列之后，仅仅考虑'cnt' 作为 target，我们可以简单的画出一些趋势图。但是可以明显地发现，没有什么肉眼可见的规律。

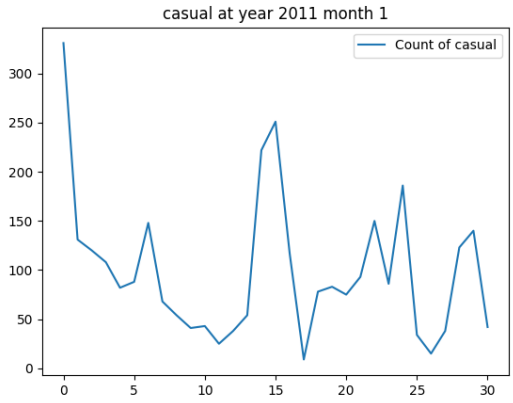


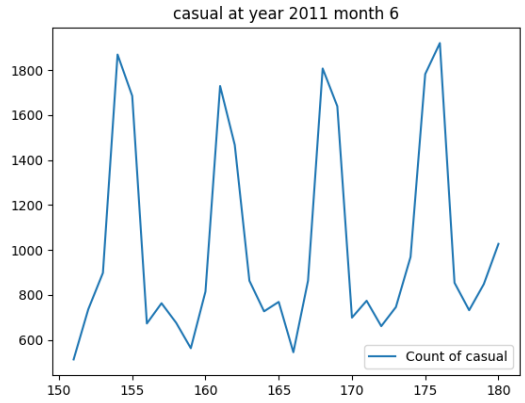


后续会进行时序数据的处理的介绍。

3.2 问题 2

第二个问题与第一个问题相似，但是考虑以'casual' 作为 target，在进行预测之前，我们先对其形式进行可视化，会惊喜地发现一定规律：





也就是说，数据呈现很强的周期性，以一周为单位进行变动。这点为后续的处理提供了一个好的思路。

3.3 问题 3

观测部分数据，如 20110601-20110607，20110101-0107，会发现出现了在注册用户和非注册用户中出现了一些特征：

- 午休现象：工作日期间的中午，注册用户数目会急剧减少，而非注册用户中午仍然较高。周末的整体使用水平较低，对于注册用户而言，早晚有一个高峰期，而非注册用户周末明显增加。

  猜测注册用户多是工作组，非注册用户日常比较有空闲，因此可以不受时间约束的使用，进而周末更多时间用来休息，造成数据的增加。

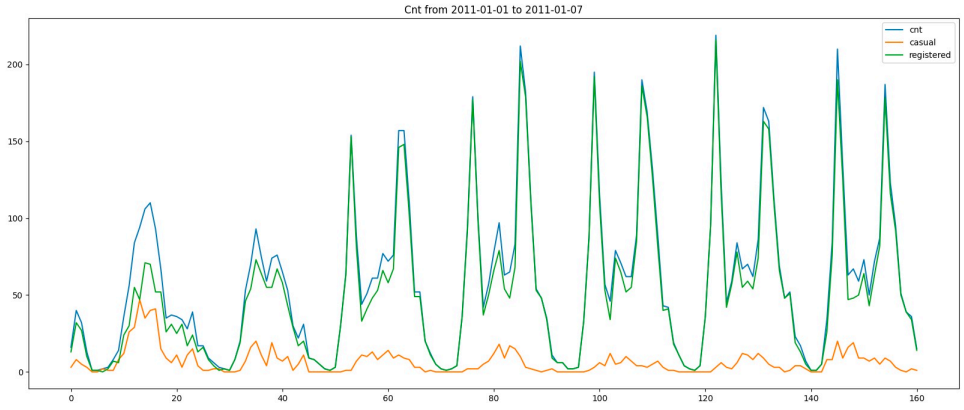


图 7: cnt, casual 从 20110101 至 20110107 的变化

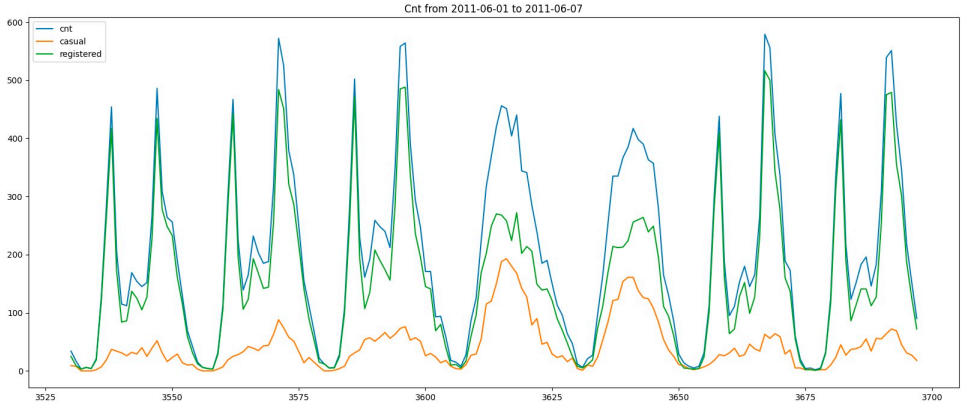


图 8: cnt, casual 从 20110601 至 20110607 的变化

除此之外，我们还观察到注意到，在数据集中，某些天是重要的节日、或者发生了重大事件或者出现了极端气候。在之后的探究中，我们也会根据这些重要时间点来产生新的特征。

3.4 问题 4

问题四是关于特征重要性检验，对于原有的特征，以及新添加进来的特征，我们将采用求解线性回归系数和随机森林特征重要性的方法来找出重要的特征。

3.5 问题 5

问题 5 是异常值的检测问题，可以使用统计的方法，来对离群值进行检测，对应 python 中的 adtk 库等。此处，我们了解到 SVM 在保证 One Class 的情况下，可以用来进行异常值检测，因此采用了 SVM 的算法对'cnt' 属性进行直接的训练测试。

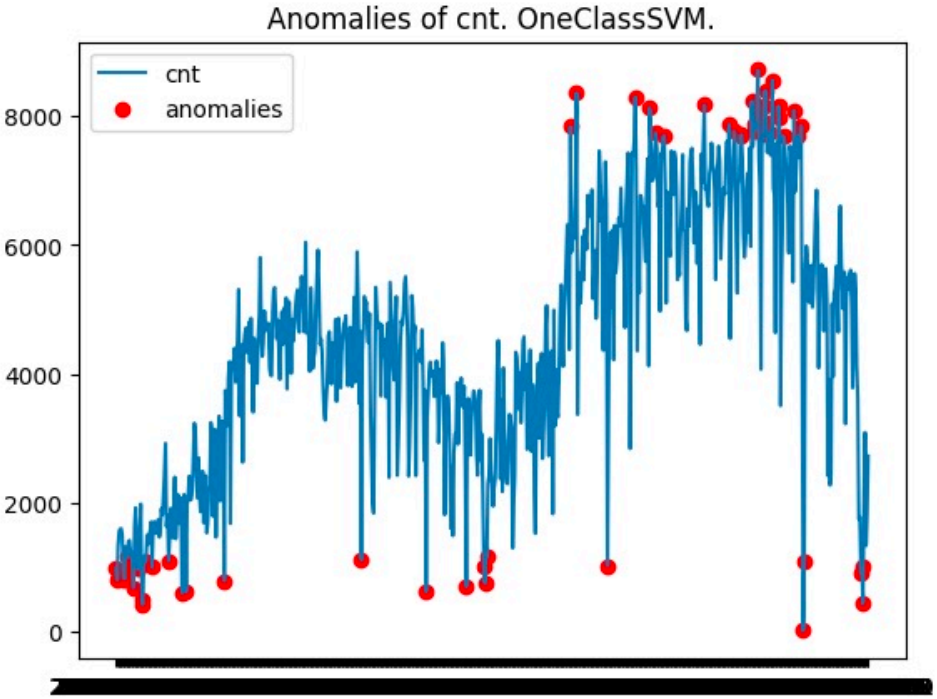


图 9: One Class SVM anomalies with $\gamma = 1e-8, nu = 0.08$

我们可以看到，一天中会的注册用户会出现显著的“午休”现象，而随机用户并没有出现这个现象。因此，可以考虑针对这个特殊情况进行额外的建模处理。

此外，受到节日、天气等因素的影响，我们也发现一些比较异常的情况，如下图所示，在 2012 年 10 月 29 日，受飓风桑迪的影响，可以明显观察到共享单车使用量的急剧减少。

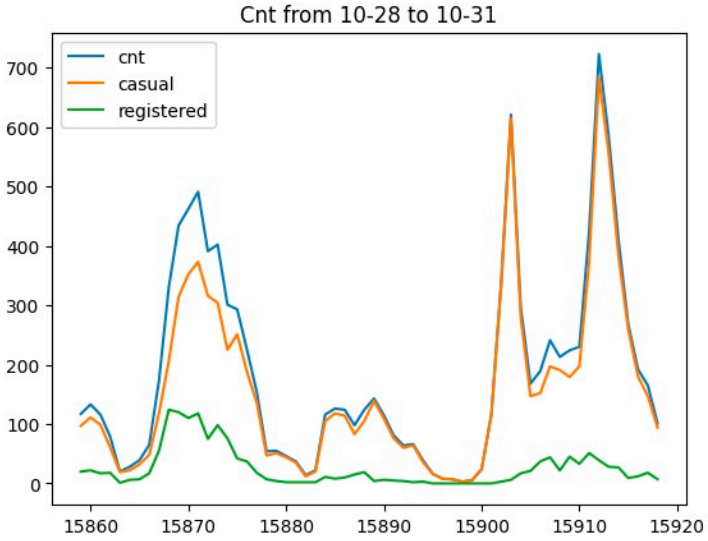


图 10: 飓风桑迪前后共享单车使用量变化

4 数据预处理

4.1 缺失值判断

day.csv 和 hour.csv 分别包含了 16 和 17 个 features。我们对 day.csv 和 hour.csv 进行了缺失值的检验，发现数据集中没有缺失值。

4.2 异常值检测

下面是对 temp、atemp、hum、windspeed、casual、registered、cnt 进行离群值检验的结果。通过观察可以发现，数据集中出现了 casual、registered、cnt 过低，hum(湿度) 过低和 windspeed(风速) 过高的情况，猜测出现这些情况主要与极端天气有关。

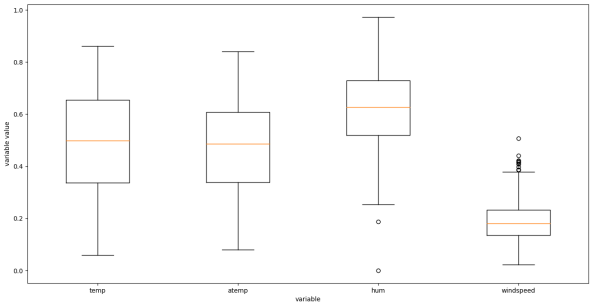


图 11: temp atemp hum windspeed 箱形图

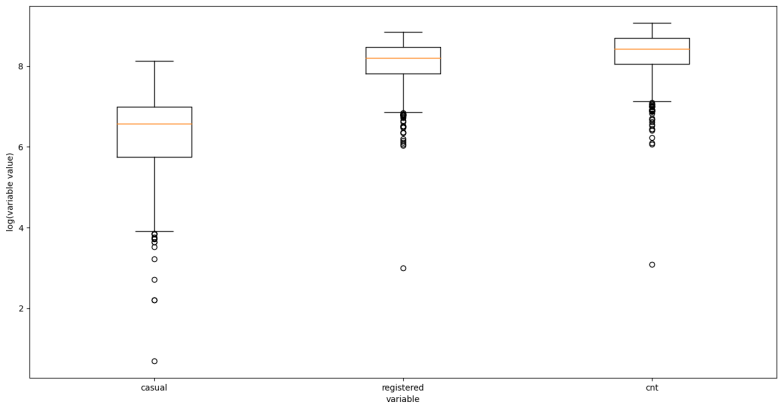


图 12: casual registered cnt 箱形图

4.3 相关性检验

我们对这些 features 就行了相关性检验，结果如下图 6 所示（以 day.csv 数据集的结果为例)。通过观察可以发现，temp 和 atemp、instant 和 yr、mnth 和 season 这三对 features 之间存在明显的线性关系，因此我们决定 drop 掉湿度 atemp、yr、season 这三个 feature。

5 想法与思考

5.1 ML

我们尝试直接使用多种机器学习模型，如 RandomForest, GradientBoost 等，对其进行预测，发现效果一般，具体结果在模型构造部分叙述。

5.2 STL

此处，鉴于学习能力，着重了解了一下 STL 的思想。统计中的 STL 方法认为时间序列数据可以分解成 trend, seasonal, noise 三部分，这里由于 cnt 上升的的不是那么明显，我们可以认为这个分解是加性的，因此采用 statmodels 库的相关函数，可以得到训练集中的相关分解结果。

而 STL 的预测方法仅限于把数据进行分解，实现预测功能还需要额外的处理，比如移动平滑法等。

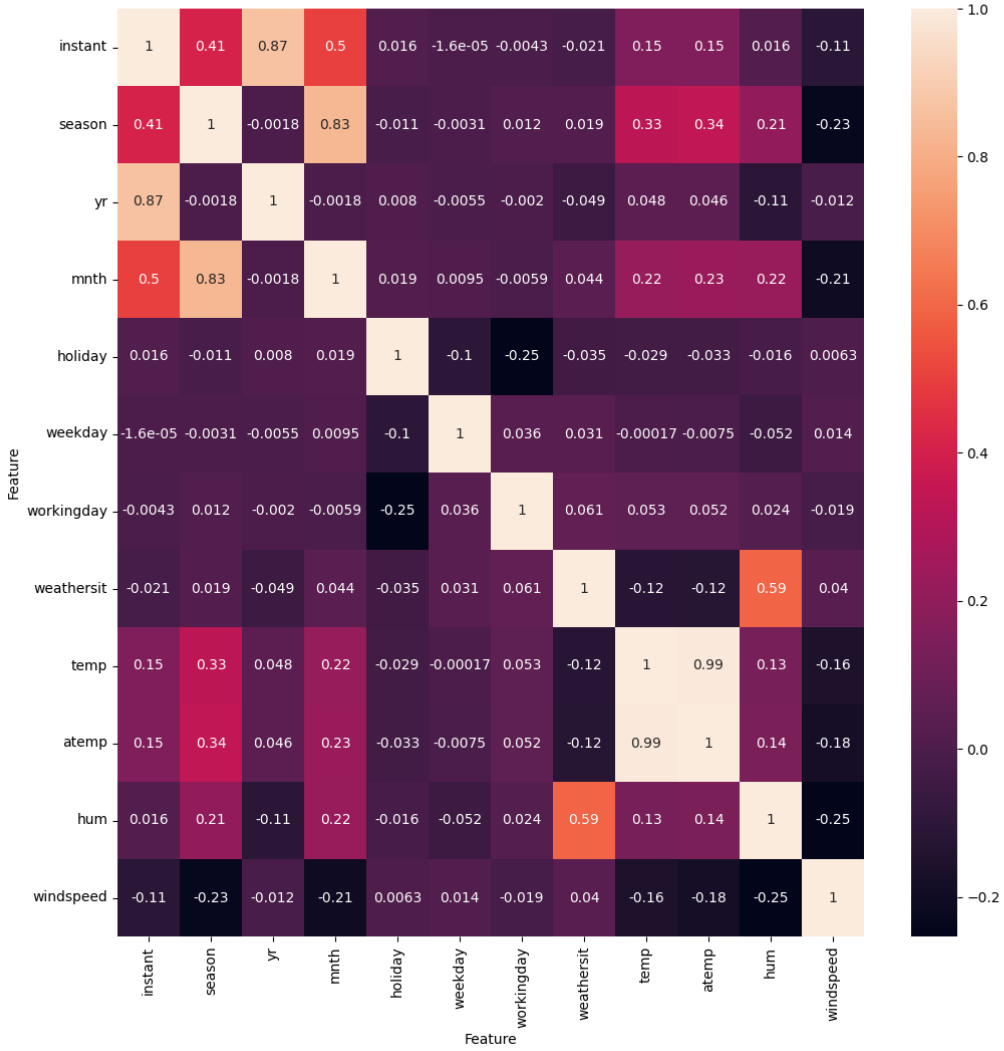


图 13: Features 相关性热力图

5.3 STL + ML

考虑到常见 STL 分解后的预测使用移动平滑法等方法，某种程度上过于依赖于历史性的数据，因此，我们采用额外的回归模型，对于 STL 分解后的 trend, season 等部分进行预测，并在预测的时候通过加权的方法，来寻找此模型下的最优解。

5.4 ANN

我们考虑了简单的 Deep Learning，尤其是全联接网络，尝试从数据中暴力的获取一个信息。建立了约 20 * 20 的全联接网络后，在服务器上通过 40000 代的测试，发现

loss 不再继续下降。因此，否决了简单的神经网络的思路，经过了解，觉得循环神经网络可能能够相对有效的解决时序数据这个问题。

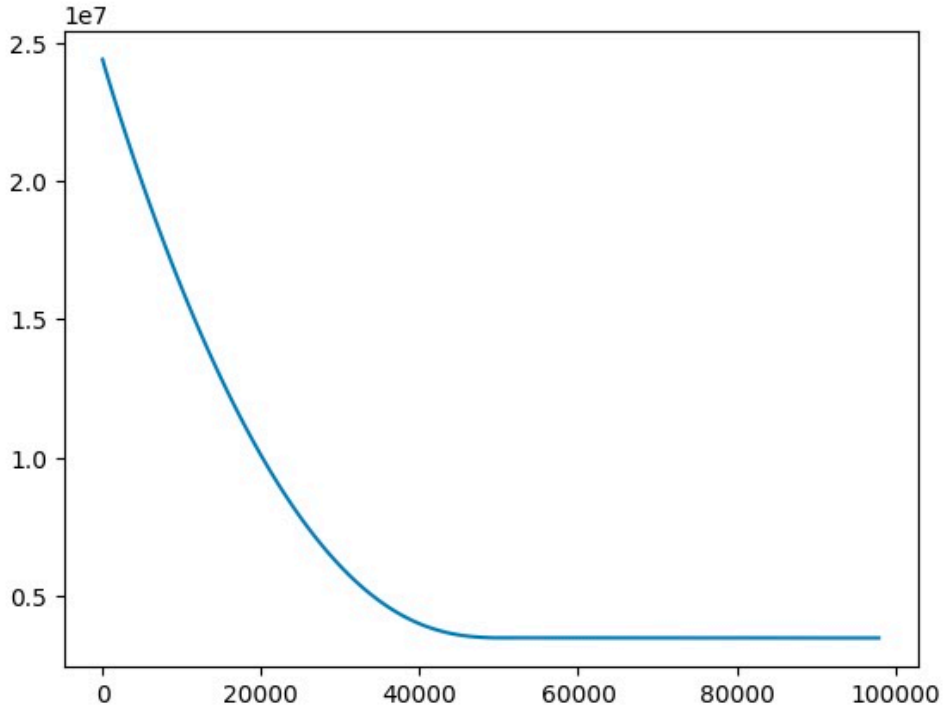


图 14: ANN Loss

5.5 RNN, LSTM

考虑循环神经网络，能够有效的处理时序信息，我们尝试了直接使用 LSTM 网络进行直接的训练，发现效果比较不具有解释性。此处猜测是因为每一个月仅仅才用了前 19 天的数据作为训练集的缘故，并且受限于时间和计算力的因素，没能继续优化训练，进而不会学习到长期行为。而采用全数据集进行 batch sampling 会学习到和后文 Extract Model 视觉上近似的效果。

6 模型构造

day.csv 数据集中，包含了时序数据，因此可以考虑对时序数据进行额外的处理。

6.1 Model 1: Direct ML

我们使用了随机森林、决策树、线性回归、Lasso 回归、岭回归、集成学习方法这些机器学习方法进行了回归。在多次迭代、调节参数之后，得到了以下结果。在 day.csv

数据集的预测中，各个方法均未表现出良好的效果。在 hour.csv 数据集的预测中，随机森林和集成学习方法表现较好，但是也都没有取得非常好的效果。

File	*RandForest*	*DecTree*	*Linear*	*Ridge*	*RidgeCV*	*Lasso*	*GradBoost*
day	0.717	0.659	0.643	0.628	0.641	0.641	0.792
hour	0.870	0.824	0.365	0.365	0.365	0.362	0.888

表 2: 各个机器学习方法在两个数据集上的得分

6.2 Model 2: STL

考虑到时序数据在 ML 中，并不能被很好的学习到，我们尝试了将测试集中的时序数据进行 STL 分解，尝试获取其与时间相对无关的趋势。

而对于时序数据，我们可以用加性模型或者乘性模型，可以认为：

$$\hat{y} = f(\mathbf{x}) = T(\mathbf{x}) + S(\mathbf{x}) + P(\mathbf{x})$$

或者认为：

$$\hat{y} = f(\mathbf{x}) = T(\mathbf{x})S(\mathbf{x})P(\mathbf{x})$$

但是通过观察我们的数据集，我们认为前者作为更平稳的模型。这里以 2011 Feb 为例，列出了对其进行 STL 分解后的效果。

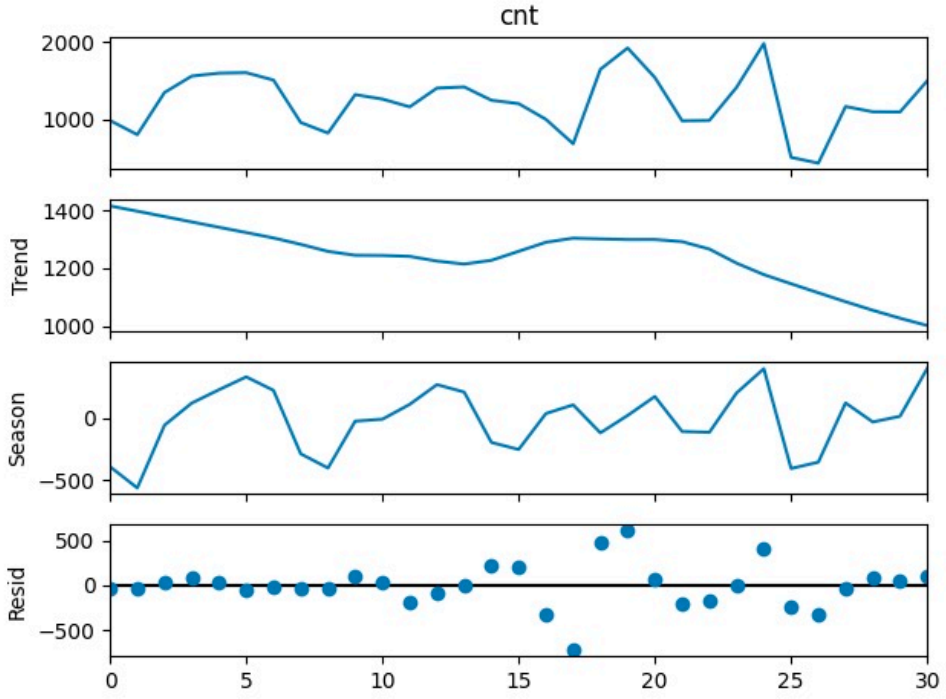


图 15: STL 分解示意图

可以注意到，cnt 被分解成了 trend, season, period。采用此模型，可以通过移动平均法等进行预测，会发现预测结果对于 6-10 月的结果相对稳定，但是会发现受到前 19 天数据的影响较大，会对预测的结果起到决定性作用。

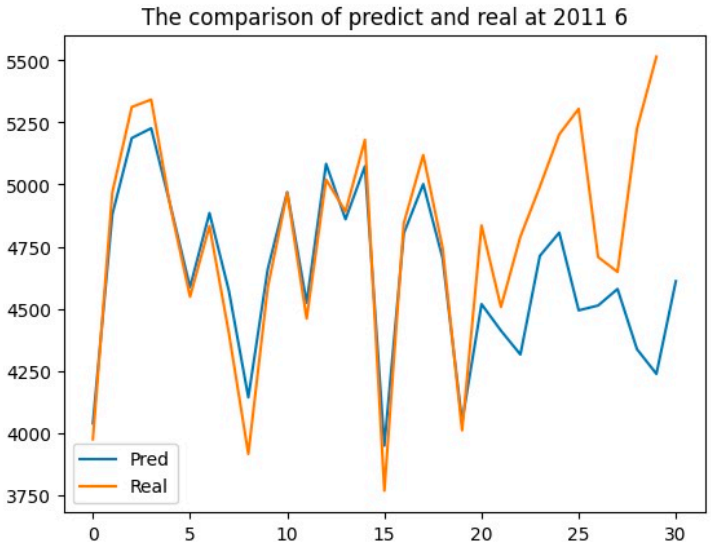


图 16: 2011 06 Prediction

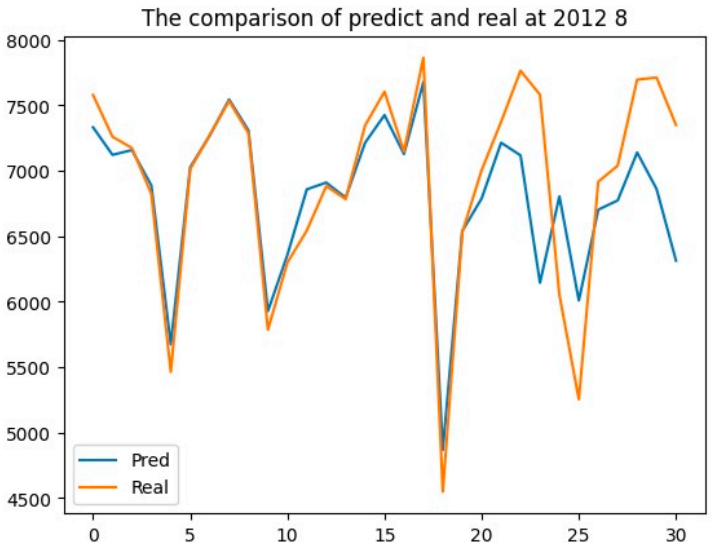


图 17: 2012 08 Prediction

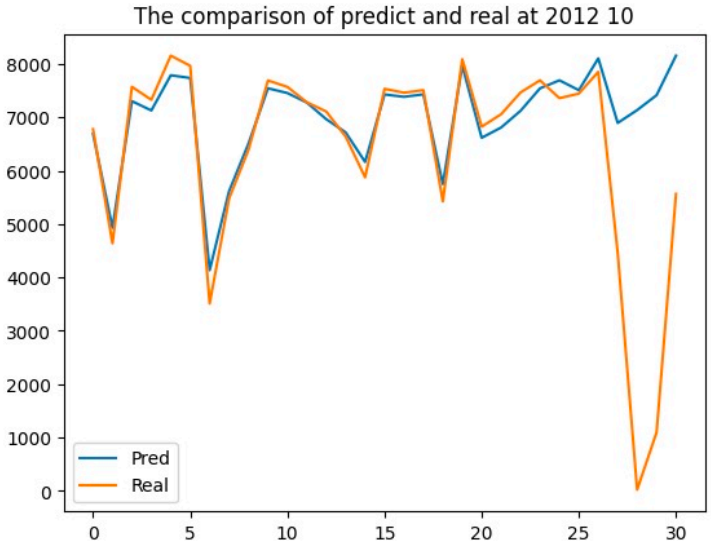


图 18: 2012 10 Prediction

6.3 Model 3: STL + ML

由于 Model 2 的结果中，我们感受到了采样造成的预测困难，于是我们质疑了题目要求的数据采样效果会很差劲，会造成预测的巨大困难。

由于 STL 模型的预测着重采用过去的统计量，为了解决前文基于统计的 STL 模型在预测上的“预测性”不足，我们引入了简单的回归模型，记做 ML。而 ML 模型在预测效果上和 STL 应当是对等的，即我们并不能直接声明哪个更好哪个更坏。因此，我

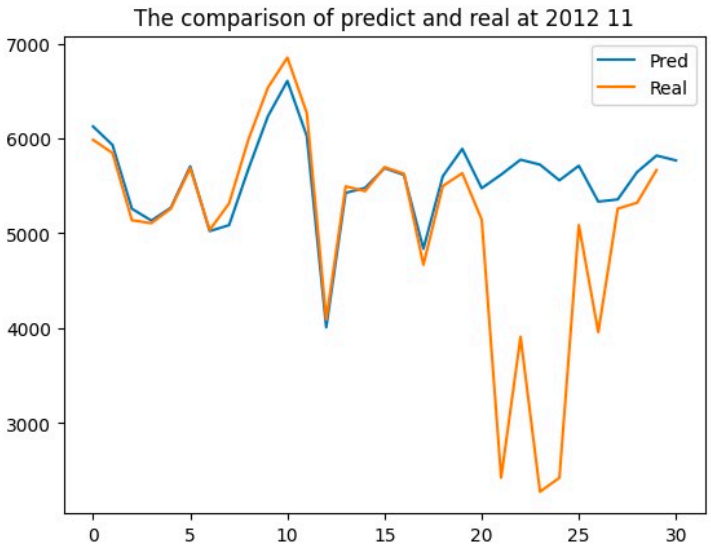


图 19: 2012 11 Prediction

们引入一个权重，最终有以下形式：

$$\hat{y} = \alpha STL(\mathbf{x}) + (1-\alpha)ML(\mathbf{x})$$

为了寻求最优解，我们首先要观察系数的配比，可以假设 trend, season, noise 分别对应三个系数为 a_t, a_s, a_e，由于两种模型预测产生的价值可以认为是相通的，需要让其系数和为 1。考虑到系数的和是定值，因此三个部分的六个参数可以简化成 3 个，此外，考虑到噪声部分是不具有明显意义的，因此我额外讲 trend 和 season 的两部分进行加权，以 $1+\alpha$ 作为 trend 的系数，以 $1-\alpha$ 作为 season 的系数。通过暴力枚举法，我们可以近似找到一个较优的参数，即：

$$(a_t = 0.8, a_s = 0.6, a_r = 0, alpha = 0.8)$$

此最优参数下，测试集总体的 $L-1$ 误差为 3912.37。

6.4 Model 4: Extract Model

Model 3 的设计和前文数据集的难点给我们提供了一种新的思路和可能，我们认为，可以通过抓取其他年度的相同月份甚至仅仅是相同 weekday 的数据进行模型学习和预测。我们首先针对每个 weekday 训练一个确切的小的模型，而预测的时候，需要根据数据的 weekday 选择对应的 Model，进行直接预测，或者额外采用相邻日期的 Model 进行预测后的加权平均，即：

$$\hat{y} = Model_{Wk(\mathbf{x})}(\mathbf{x})$$

或者

$$\hat{y} = \sum_k \alpha_K Model_k(\mathbf{x}), \quad |k - Wk(x)| \leq 1$$

此处，经过简单测试使用了 1.2, 1.5, 0.3 的参数取前后两天的模型作为辅助，得到测试集上 L_1 norm 下的绝对误差为：789.04.

6.5 Model 5: NN Model*

此部分，我们简述一下我们没有成功训练出来的 NN Model 的部分，我们使用了 pytorch 框架，其内部实现了很简易的 LSTM 接口，我们使用 100 个 hidden layers。训练了约 40000 代，但限于时间原因，没有办法对效果进行检测。这里仅仅贴上我们的训练的 loss 历史数据的折线图。

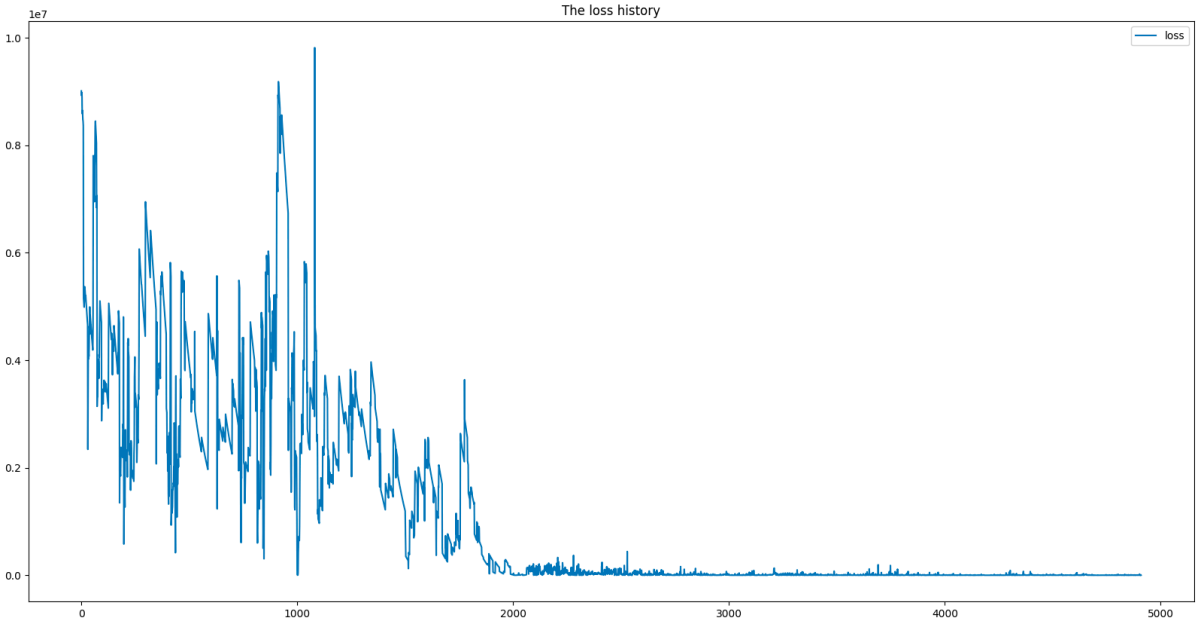


图 20: The loss history of our self-designed NN

以及我们没办法进行合理解释的 LSTM 模型：

可以注意到，LSTM 模型仍会有较多差距较大的值，经过手动比对验证，我们发现节假日相关的利群值构成了这些具有明显差距的点的主要成分。

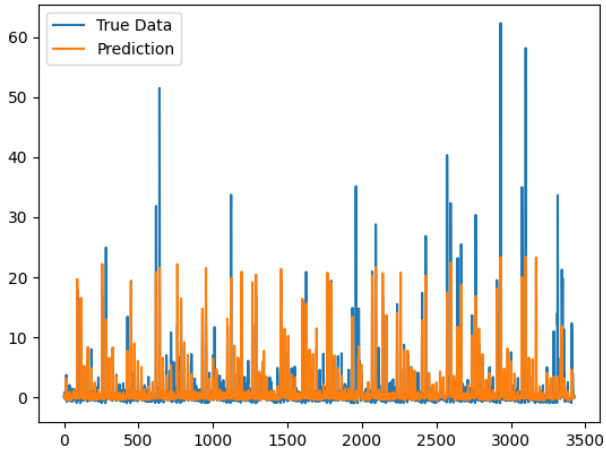


图 21: Caption

7 特征选择

7.1 特征添加

我们调查了在 2011-2012 年美国华盛顿地区发生的重要事件，具体如下：

type	*date*	*event*
节日	2011-01-17	马丁 · 路德 · 金纪念日
极端天气	2011-01-26	暴风雪
极端天气	2011-01-27	暴风雪
节日	2011-02-21	总统节
极端天气	2011-04-16	龙卷风
节日	2011-05-30	阵亡将士纪念日
节日	2011-07-04	独立日
节日	2011-09-05	劳动节
政治事件	2011-10-08	占领华尔街游行
节日	2011-10-10	哥伦布日
节日	2011-10-31	万圣节
节日	2011-11-11	退伍军人日
节日	2011-11-24	感恩节
节日	2011-12-26	圣诞节
节日	2012-01-16	马丁 · 路德 · 金纪念日
节日	2012-02-20	总统节
节日	2012-05-28	阵亡将士纪念日
节日	2012-07-04	独立日
节日	2012-09-03	劳动节
节日	2012-10-8	哥伦布日
极端天气	2012-10-29	飓风桑迪
节日	2012-10-31	万圣节
政治事件	2012-11-06	美国总统大选正式选举日
节日	2012-11-12	退伍军人日
节日	2011-11-22	感恩节
节日	2011-12-25	圣诞节

表 3: 2011-2012 年美国华盛顿地区重要事件

我们分别针对这些事件，包括马丁 · 路德 · 金纪念日、暴风雪、总统节、龙卷风、阵亡将士纪念日、独立日、劳动节、占领华尔街游行、哥伦布日、万圣节、退伍军人日、感恩节、圣诞节、飓风桑迪，添加了一共 14 个特征。

7.2 特征筛选

我们使用添加了新特征的数据集，重新训练和测试了我们的模型，发现并没有得到什么提升，和原数据集表现无明显差异。因此，我们只保留了原数据集中的特征。

7.3 特征重要性

对于 day.csv 数据集，我们使用了线性回归的方法，统计了每个 feature 对应的系数，如下图所示。由图可知，每天共享单车的数量和 temp(温度) 的关系非常密切，除此之外，和 windspeed(风速) 以及 hum(湿度) 也有比较强的关系。

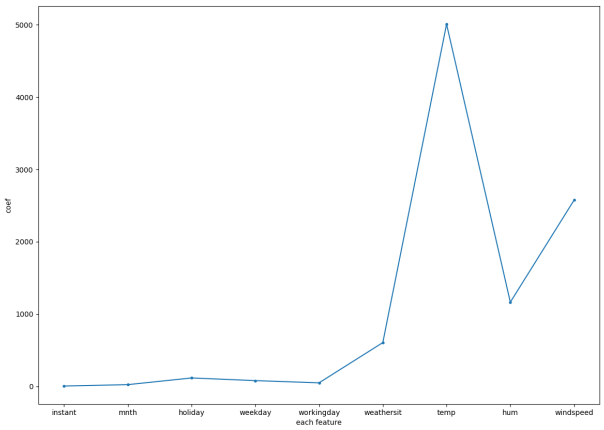


图 22: 线性回归模型中各个 feature 的系数

对于 hour.csv 数据集，我们统计了随机森林的特征重要性，如下图所示。由图可知，具体每个时间段共享单车的数量和 hr(当前具体时间) 关系非常密切，除此之外，和 instant(日期变化)、workingday(是否为工作日) 以及 temp(温度) 也有比较强的关系。

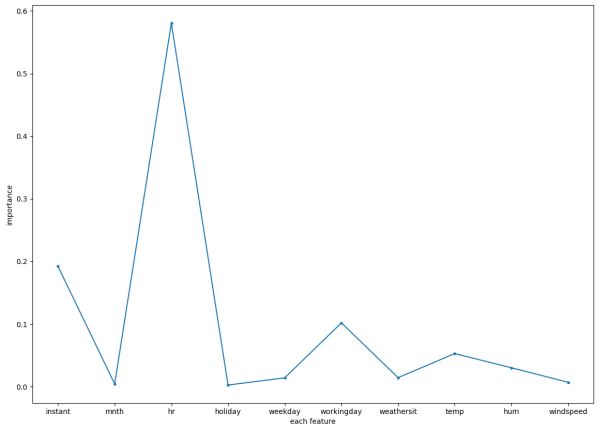


图 23: 随机森林模型中各个 feature 的重要性

8 模型评估与后期工作

8.1 STL + ML

考虑到我们提出的 STL 模型和 ML 的结合，采用 L_1 Norm，我们通过调参，可以得到一个最优的结果。但是实际上有关这个预测准确度我们认为有待商榷。

通过暴力枚举，以

8.2 Extract Model

我们观察到的午休现象等可以提醒我们，在处理的时候对工作日和休息日进行不同的建模会更加合理。而工作日中，中午的午休现象可以通过额外的针对'registered' 用户进行建模。因此实际上，我们认为如果要获得更高的准确度，应当尝试对数据集重新进行标注，改造为 Monday，Tuesday 等的时间序列，这样在学习一个新的日期的时候，根据已有的公式可以计算出其日期，进而进行更有效的学习。

此处，我们定义一个 model 类，内部包含针对单个工作日的模型，因此在 fit 的时候先对数据集进行合理的分类，得到相应的 7 个模型。

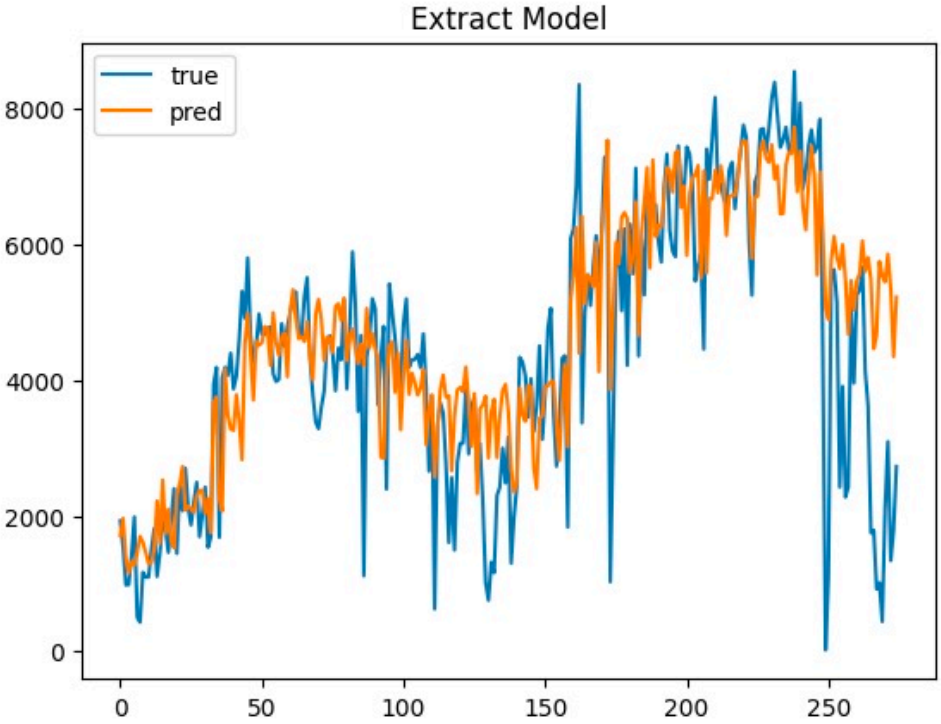


图 24: The result of Extract Model in the testset

由于时间和精力的原因，Extract Model 没有去对 hour.csv 进行额外的训练，后期可以考虑对每天的日期进行额外的分期，然后采用 day.csv 中近似的思路，抓取邻近的时间或者直接将时间分成上中下三个阶段以进行训练。应该能够很好的解决午休现象的同时，获得更高的精确率。这里不对模型的可行性进行进一步的阐释，如果需要实现模型的功能，也只需要额外的数据处理的代码部分。

8.3 Self Design Neural Network

此外，我们考虑了针对时序变量进行单独的 RNN 设计，网络结构采用单个 lstm 和线性模型结合，以 100 层隐藏层的 lstm 构建时序的关键处理。但是由于时间关系，我们没有实现这个方法，额外的尝试了预训练的模型，但是没有很明白效果。

考虑到 lstm 网络训练的速度是极其慢速的，我们将它放到了服务器（8 张 12 GB 3090），使用 cuda 加速的情况下，训练一代大概要 1 秒附近。而训练 4 万代左右的时候 loss 仍然在较高的量级，目测需要更多的时间才能得到更好的效果，这里也是未来可以做的部分。

9 分工说明

我们的项目，其中理论和建模的部分主要靠王铎磊同学提供思路，陈文雁同学负责代码实现，同时感谢张振老师的服务器的计算资源，为我们训练神经网络提供了重要遍历，虽然我们的模型的核心部分和创新点并不是神经网络的部分。

具体分析的话，Extract Model，One Class SVM 离群值检测，自定义的简单神经网络的代码，基本的数据可视化（如包含午休部分的那些）是王铎磊同学的代码。基本的数据可视化（总体分布、各项指标、单一指标）、数据预处理、特征添加选择以及重要性分析、基础机器学习模型是陈文雁同学的代码。

其余部分认为是均等分工。