XGBOOST MODEL FOR PARK VISITATION PREDICTION IN A MID-SIZE CITY

A Professional Paper

by

XINKE HUANG

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF URBAN PLANNING

Chair of Committee, Xinyue Ye
Committee Members, Yang Song
Xiaofeng Nie
Head of Department, Galen Newman

April 2022

Major Subject: Urban Planning

Abstract

Parks have a significant impact on residents' health and social activities. By using smartphone mobility data tracking the activities of 28 parks in the College Station and Bryan Metropolitan area of Texas, USA, I present the temporal and spatial patterns of park usage within a two-year timeframe. I model the effects of the socio-economic, built environment, climate, surrounding points of interest (POI), and spatial/accessibility factors on park visitations through a machine learning model. The results show that climate change and nearby POIs such as restaurants and gas stations are significant factors enhancing park visitations while having hotels and apartment complexes are not. The study also reveals how smartphone mobility data can be applied to case studies investigating urban design/planning and understanding the social and adjacent points of interest associated with urban greenspaces. It provides empirical evidence on park visitations as well as what factors future planners, landscape architects, and park managers should consider when deciding on park investment and planning decisions for mid-sized cities.

Acknowledgment

I would like to express my special thanks and gratitude to Dr. Yang Song, who provided the dataset for my study as well as my committee chair, Dr. Xinyue Ye, and committee member Dr. Xiaofeng Nie who gave me the opportunity to proceed with this study on the park visitation prediction in College Station - Bryan Metropolitan area.

Table of Contents

Introduction **4**

Methodology **6**

Data Collection 6

XGBoost 14

Mathematics Explanation 15

Analysis and Results **18**

Data Description and Processing 18

XGBoost 20

Hyperparameter tuning 24

Results and Findings 26

Conclusion **32**

Discussion **33**

References **35**

Introduction

People visit natural areas to appreciate the landscape and wildlife to participate in a variety of leisure activities. These regions are important to society, as evidenced by the large number of people that visit parks, walking trails, and other natural areas, as well as their major economic impact (Leeworthy & Wiley, 2001). It is difficult to quantify these visits. Using observations is expensive and time-consuming, and it is hard to count every visit to specific locations. Big data, also called passive data, refers to the data not collected actively (small data). Instead, it was unintentionally collected with little cost. Examples include users' social media data, smartphone data representing users' locations, and metro card data collected at transit systems. The use of big data (passive) plays a huge role in human mobility prediction and monitoring long-term behaviors not only in park utilization but also in traffic prediction. Big data can also determine the Annual Average Daily Traffic (AADT) data. Instead of using traditional roadways sensors, from vehicles operating on the roads, such as smartphones and vehicle tracking devices (Non-Traditional Methods to Obtain Annual Average Daily Traffic (Aadt) Evaluation and Analysis | Washington State Transportation Center n.d.).

This paper uses Safegraph, the company that anonymously aggregates cell phone location data (Safegraph, 2020) and has gained attraction to monitoring urban mobility behaviors. The application highlights the potential of this growing source of big data to give detailed visiting statistics across several locations and periods, which would be impossible to do using traditional techniques. Cell phone location data's high-resolution information in

space and time expands opportunities for developing next-generation models of human interactions with the parks and other natural environments. Furthermore, cell phone location data allows us to know how many people visit certain areas and where visitors are coming from inside aggregated regions. The existing studies regarding park visitation are mainly focused on characteristics of parks, including park facilities, the environment, and socio-demographics. These are key elements that affect park visitation. However, parks nearby the points of interest also have the potential to influence park visitation.

Using the College Station and Bryan Metro area in Texas, the USA as our focus area, I collect and analyze smartphone data to quantify weekly park visits by establishing origin & destination patterns from park-related SafeGraph trajectory data, and using Google Places API to collect nearby POIs of each park. The result of this study provides important empirical evidence and insights into the future planning, design, management, and site selection of city parks.

Methodology

Data Collection

The data used in the study is Safegraph's Places data. This data contains three datasets in total.

- The "Core Point of Interest" dataset contains x.x million POIs in the U.S. conclude information about location name, address, brands, category, open hours, and other geographic and opening information.
- The "Geometry" dataset contains POI footprints and spatial hierarchy information for over 9 million locations. They use polygons to map the size and area of actual places associated with the POIs in their databases. This data offers information about the boundaries and relationships between locations. Available in the United States of America, Canada, and the United Kingdom for POIs.
- The "Patterns" dataset contains aggregations of visitor and demographic data for US sites of interest (POIs). This dataset comprises aggregated raw counts of visits to POIs from a panel of mobile devices, providing information about how frequently people visit, the dwell time, visitors' home block groups (BG), and where visitors travel next, among other things.

Due to privacy concerns, Safegraph applies differential private techniques. By adding Laplacian noises, visitors' home block groups with less than two devices are not included in the data. If there are two to four visitors for either a POI (destination) or a BG (origin), SafeGraph will report as four. To address this overcounting problem, I eliminated

SafeGraph records with a value of four, and I included those records with a weekly BG to POI user count of more than four in the final analysis.

This study focuses on the BGs and POIs in the College Station & Bryan (CSB) metro area. One hundred and eighteen BGs that overlap with CSB boundaries were included, with a median household income of $38,341(Souce: ACS Table B19001, U.S. Census Bureau)[1]. The total population of the CSB metro area is 268,248[2]. Texas A&M University is located in the center of the study area with a student population of 73,284[3]. Our data collection constructs a dataset of mobility patterns for this target area related to park visitation and outdoor recreation from 2018/12/31 to 2020/11/30. I first downloaded all available SafeGraph POIs in the "Nature Parks and Other Similar Institutions" sub-category of the SafeGraph 'Core POI' dataset and linked them with "Geometry" and "Places Patterns" datasets. Parent POIs that geographically overlap their 'child' POIs were deleted to avoid double counting. For example, if there is a POI (parent) of a mall that includes many store POIs (child), I did not count this mall POI (parent) into our dataset. I modified the BG-level visitor numbers for each week of our research period to account for variations in device sampling ratios.

[1] U.S. Census Bureau (2019). American Community Survey 2019 5-Year Estimates: Table B19001. Retrieved from https://data.census.gov/cedsci/table?q=B19001&g=310XX00US17780&tid=ACSDT1Y2019.B19001&moe=false.
[2] U.S. Census Bureau (2020). Retrieved From https://www.census.gov/.
[3] Total Enrollment Fall 2021. Texas A&M University. Retrieved from https://www.tamu.edu/about/at-a-glance.html.

$$VBG_{i,k,t} = \frac{SBGV_{i,k,t} * POP_i}{SGD_{i,t}} \quad (1)$$

$VBG_{i,k,t}$ refers to the visitor number from BG *i* to POI *k* during the *t* th week of our study time period, $SBGV_{i,k,t}$ refers to the SafeGraph visitors from BG *i* to POI *k* during the *t* th week of our study time period, POP_i refers to the total population from the BG *i*, and $SGD_{i,k}$ refers to the total number of devices sampled for BG *i* throughout our *t* th week of our study period (Song et al., 2022).

$$VP_{k,t} = \sum_{i=1}^{n} VBG_{i,k,t} \quad (2)$$

$VP_{k,t}$ refers to the total visitors of POI *k* during *t* th week of our study time period, and *n* refers to the total number of BGs that have traveled to the POI *k* during *t* th week of our study period (Song et al., 2022). The $VP_{k,t}$ of POIs during the *k*th week will be used in the later machine learning model.

Additionally, I used social-demographic controls with demographic data (e.g., income and age) from the American Community Survey 5-year estimate and climatic controls from the

CSB, such as weekly temperature and average rainfall data from NOAA, for subsequent analysis (Song et al., 2022).

Furthermore, based on guidelines provided by the National Recreation and Park Association (NRPA), I classified the parks as community parks and neighborhood parks based on their facilities. If a park contains recreational amenities and is greater than 16 acres, I classify it as a community park. All other parks are considered neighborhood parks (Park Classifications | Dallas Parks, Tx - Official Website, n.d.). Moreover, I indicate the number of sporting facilities at each level. For instance, an additional basketball court may serve twenty people (4 teams). To account for this variation, I categorize sports facilities as low (value 0, as no sports facilities), medium (value 1, as one to three sports facilities), or high (value 2, as more than three sports facilities).

In order to find the nearby POIs of each park, I used Google Places API to achieve the goal. Google Places API is a software library that provides a nearby search tool capable of reporting on various types of organizations, establishments, and other things located within a specified radius. In depth, the API enables developers to create applications that include Place Search, Place Details, Place Actions, Place Photos, Place Autocomplete, and Query Autocomplete features[4]. In the present version, the nearby search function provides a list of items with the supported categories listed in **Table 1**. It is possible to retrieve the number

[4] Core Features of Places API. Places API. Retrieved from https://developers.google.com/maps/documentation/places/web-service.

of items in a circumferential region for a certain radius using this API. The results are returned in either JSON or XML, both of which are easily parsed.

Table 1. Supported place types by Google Places API for Place Searches[5]

Place Types		
accounting	electronics_store	park
airport	embassy	parking
amusement_park	fire_station	pet_store
aquarium	florist	pharmacy
art_gallery	funeral_home	physiotherapist
atm	furniture_store	plumber
bakery	gas_station	police
bank	gym	post_office
bar	hair_care	primary_school
beauty_salon	hardware_store	real_estate_agency
bicycle_store	hindu_temple	restaurant
book_store	home_goods_store	roofing_contractor
bowling_alley	hospital	rv_park
bus_station	insurance_agency	school
cafe	jewelry_store	secondary_school
campground	laundry	shoe_store

[5] Supported Place Types. Retrieved from https://developers.google.com/maps/documentation/places/web-service/supported_types.

car_dealer	lawyer	shopping_mall
car_rental	library	spa
car_repair	light_rail_station	stadium
car_wash	liquor_store	storage
casino	local_government_office	store
cemetery	locksmith	subway_station
church	lodging	supermarket
city_hall	meal_delivery	synagogue
clothing_store	meal_takeaway	taxi_stand
convenience_store	mosque	tourist_attraction
courthouse	movie_rental	train_station
dentist	movie_theater	transit_station
department_store	moving_company	travel_agency
doctor	museum	university
drugstore	night_club	veterinary_care
electrician	painter	zoo

The study I used Nearby Search provided by Google Places API. The Nearby Search requests would provide direct access to the response parameters we want, but with one significant restriction: they will return no more than 20 results per query. Each search can return up to 60 results spread across three pages, which implies that we can use the

next_page_token data to retrieve a total of 60 results[6]. I did not use the next_page_token for our study, which means the highest number of results for nearby POI is 20.

The chosen radius is 1000 meters (0.62 miles), and the nearby place types restricted to the study are restaurants, department stores, apartments, secondary schools, primary schools, hotels, gas stations, hospitals, and churches. The nearby number of corresponding place types of each park is in **Table 2**. As there is no place type "school," I sum up the number of primary and secondary schools as " schools" in the result.

Table 2. The Number of Nearby Places of Each Park within 1000 meters radius.

Park Name	Restaurant	Department store	Apartment	Hotel	Gas station	Hospital	Church	School
Brazos Valley Veterans Memorial	2	0	20	20	0	3	0	0
Southwood Athletic Park	1	0	20	20	0	4	1	2
Anderson Park	20	4	20	20	2	1	3	4
Travis Athletic Complex	10	1	20	20	8	0	9	0

[6] Accessing Additional Results. Nearby Search. Retrieved from https://developers.google.com/maps/documentation/places/web-service/search-nearby#PlaceSearchPaging

John Crompton Park	10	1	20	20	9	0	3	0
Stephen C Beachy Central Park	2	0	20	20	2	1	9	0
Gloria Stephen Sale Park	20	0	20	20	7	1	20	1
Henderson Park	9	0	20	20	2	0	8	2
Copperfield Park	0	0	20	20	0	0	0	1
Thomas Pool	20	1	20	20	1	1	4	1
Brison Park	4	0	20	20	3	0	7	4
Camelot Park	9	0	20	20	7	5	3	0
Woodcreek Park	10	0	20	20	2	0	2	1
Merry Oaks Park	19	6	20	20	4	0	4	1
Sue Haswell Memorial Park	1	0	20	20	0	1	12	1
Southern Oaks Park	2	0	20	20	3	0	5	0
Lick Creek Park	0	0	10	10	0	0	0	0

Dominino Oaks Park	12	1	20	20	3	0	2	1
Steeplechase Park	14	0	20	20	2	0	4	1
Austins Colony Park	14	0	20	20	1	0	2	2
Richard Carter Park	16	0	20	20	7	1	1	0
Longmire Park	15	1	20	20	8	1	4	2
Southwest Park	17	1	20	20	6	1	5	0
Siena Estates Park	3	0	20	20	0	0	0	0
Crescent Park	20	0	20	20	3	0	9	1
Bee Creek Park	20	1	20	20	7	1	7	3
University Dog Park	3	1	20	20	4	0	9	1
Dr David E Schob Nature Preserve	20	0	20	20	1	1	6	1

XGBoost

The machine learning model used for this park visitation study is the XGBoost (Chen & Guestrin, 2016), it is flexible and cutting-edge use of gradient boosting machines that have

demonstrated the ability to push the computational limitations of boosted tree algorithms. It was created solely for the goal of optimizing model performance and computational speed. Boosting is an ensemble method in which new models are added to compensate for current models' flaws. Models are introduced in a cyclical fashion until no obvious improvement is observed. Gradient boosting is a technique in which new models are developed to forecast the residuals of previous models and then combined to make the final prediction. It employs a gradient descent approach to minimize the loss associated with the new model. This strategy is applicable to both regression and classification. The speed was greatly improved by using several CPU cores and decreasing the lookup durations of individual trees produced in XGBoost. This method is developed in R and Python using the SciKit-learn (Pedregosa et al., 2011) library, and it incorporates unique regularisation approaches.

Mathematics Explanation

XGBoost is a supervised learning methodology that refers to the mathematical structure that is used to make the prediction x_i from the input y_i. For instance, in a linear model, the prediction is a linear combination of weighted input features $\widehat{y_i} = \Sigma_j \theta_j x_{ij}$ (Introduction to Boosted Trees — Xgboost 1.5.2 Documentation, n.d.), where the parameters θ need to learn from the data and it varied from different datasets. We may describe a number of tasks using careful selections for y_i, including regression, classification, and ranking.

The goal of the training model is to find the best parameters θ that fit the prediction x_i with the input value y_i. The objective function (Introduction to Boosted Trees — Xgboost 1.5.2 Documentation, n.d.) (3) is to determine the performance of the model, that is, evaluate how the model fits with the training data,

$$obj(\theta) = L(\theta) + \Omega(\theta) \quad (3)$$

, where L is training loss and Ω is the regularization term.

The training loss quantifies how accurate our model is in relation to the training data. The regularization term is to keep the model's complexity within acceptable boundaries, preventing issues like over-stacking or overfitting of data, which can result in a less accurate model. In order to balance the training loss and regularization term, which means the model should be predictive and straightforward, the technique in machine learning for this tradeoff is the bias-variance tradeoff.

Random Forest and boosted trees are the same models. The distinction between random forests and boosted trees lies in the training process. A predictive service works for random forests, and gradient boosted trees; this is the advantage of supervised learning, defining an objective function and optimizing it (Introduction to Boosted Trees — Xgboost 1.5.2 Documentation, n.d.). The first step is to set up the parameters of trees using functions f_i that have the structure of the tree and leaf scores. Learning tree structure is delicate and is

not straightforward to train all the trees simultaneously. Rather, XGBoost uses the additive strategy, meaning optimizing the existing tree and adding a new tree at every step, where a new tree is the one that helps to the optimization of our objective function. The objective function (4) (Introduction to Boosted Trees — Xgboost 1.5.2 Documentation, n.d.) takes the second order of Taylor expansion of the loss function at step t

$$obj_t = \sum_{i=1}^{n} [g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \Omega(f_t) \tag{4}$$

, where g_i and h_i are inputs. The major advantage of this new objective function is it only depends on the inputs g_i and h_i.

Furthermore, regularization is critical in determining the complexity of the tree $\Omega(f)$. To begin, the clarification of the definition of the tree is (Introduction to Boosted Trees — Xgboost 1.5.2 Documentation, n.d.)

$$f_t(x) = w_{q(x)},\ w \in R^T,\ q:\ R^d \rightarrow \{1,\ 2,\ \ldots,\ T\} \tag{5}$$

, where w is the vector score (weight) on the leaves, q is a function that corresponds each data point to a leaf, and T is the number of leaves. In XGBoost, the model complexity is defined as (Introduction to Boosted Trees — Xgboost 1.5.2 Documentation, n.d.)

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \quad (6)$$

Hence, the equation (6) can plug into equation (3) to generate a new objective function at *t* th step or *t* th tree. The tree structure of XGBoost is determined by computing the regularization, leaf scores, and objective function at each level, as it is impractical to calculate all tree combinations simultaneously. The gain is computed at each level as a leaf is divided into left and right leaves, and the gain is calculated at the present leaf after any additional leaves have been regularized. If the benefit is insufficient to compensate for the increased regularization value, the according branch is abandoned. This is how XGBoost penetrates deep into trees and classifies data, resulting in the calculation of accuracy and other metrics (Dhaliwal et al., 2018).

Analysis and Results

Data Description and Processing

In the study, there are 28 parks in total in the College Station -Bryan Metropolitan Area. The travel patterns of all park points of interest cover an average of 87 weeks. The average daily visitors varies from 21 to 305. There are several reasons to keep us including all parks: 1) SafeGraph POI datasets are still not complete, many smaller and newer parks are omitted; 2) I excluded park types that do not fit the study, such as natural reserve areas, forests, greenway trails, aquatic centers, school parks, and empty lands. 3) I deleted parks with less than 40 weeks of data coverage as I focused on studying long-term visitation

patterns.

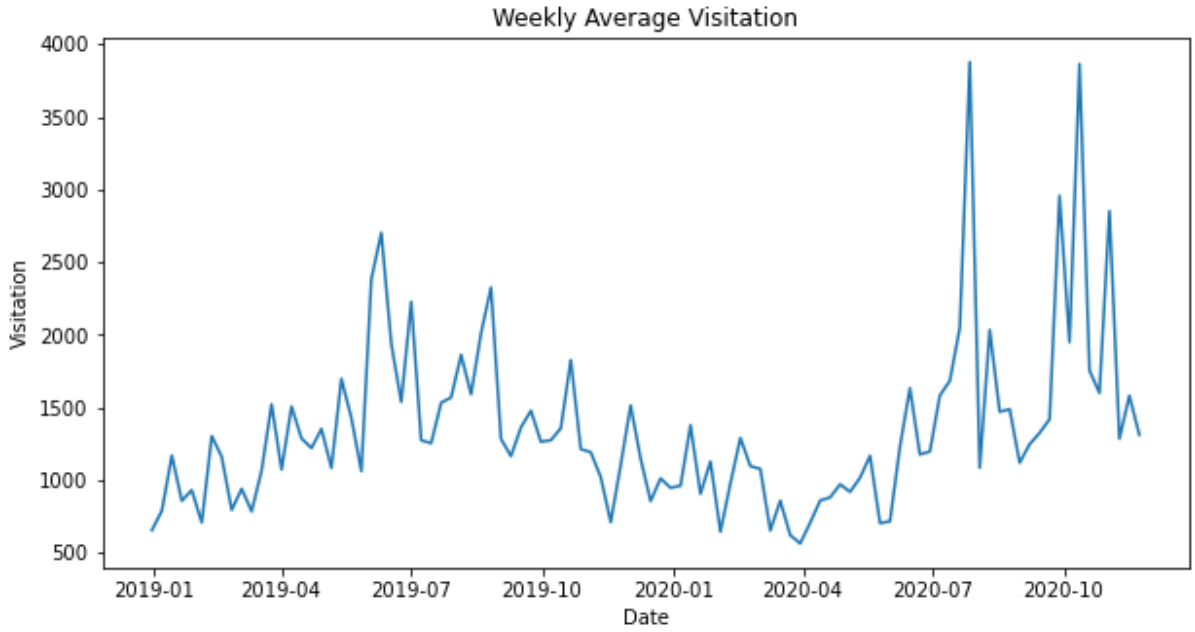

Figure 1. Weekly Average Visitation

Figure 1. shows the changes in these park categories in median visitations each week. The large visitor of park usage generally occurs in the summer months, from July to late October in 2019 and 2020. Significant increases were shown during the late summer and early fall of 2020 (July to Nov) for all parks. COVID appears to have played a substantial effect in the increase in summer visitor numbers. Many residents in the neighborhood work in higher education and have summer vacation plans with their families. More people had to reschedule their plans for 2020 due to COVID restrictions, and instead, they stayed in town.

XGBoost

First, the data was set up and checked for missing and duplicate values. In order to prevent negative visitation prediction, the numpy.log1p[7] technique is used for the target value (y), which returns the natural logarithmic value of one plus the input array. After obtaining the predicted values after hyperparameter tuning and running the XGBoost model, turning the natural log plus one of the predicted values into the actual prediction value using numpy.expm1[8]. **Table 3.** and **Table 4.** are the variable description for the dataset.

Table 3. Feature Variable (X) Description in the Dataset.

Column Name	Description	Data Type
PRCP	The weekly average precipitation of the point of interest	float
TAVG	The weekly average temperature of the point of interest	integer
includes_parking_lot	Whether the point of interest has parking lots	boolean

[7] https://numpy.org/doc/stable/reference/generated/numpy.log1p.html
[8] https://numpy.org/doc/stable/reference/generated/numpy.expm1.html#numpy.expm1

area_square_feet	The area square feet of the point of interest	integer
trails	Whether the point of interest has trails	boolean
sports facility	The level of sports facility numbers of the point of interest. Low (value 0, no sports facilities), medium (value 1, 1-3 sports facilities), and high (value 2, >3 sports facilities)	float
playgrounds	Whether the point of interest has playgrounds	boolean
water body or streams	Whether the point of interest has waterbody or streams	boolean
pavillion/ seating area	Whether the point of interest has avillion/ seating area	boolean
community park	Whether the point of interest is a community park	boolean

neighrborhood park	Whether the point of interest is a neighrborhood park	boolean
month	The month of the year	integer
restaurant	The number of restaurant within 1000m radius of the point of interest	float
department_store	The number of department store within 1000m radius of the point of interest	float
apartment	The number of residential building within 1000m radius of the point of interest	float
hotel	The number of hotel within 1000m radius of the point of interest	float

gas_station	The number of gas station within 1000m radius of the point of interest	float
hospital	The number of hospital within 1000m radius of the point of interest	float
church	The number of church within 1000m radius of the point of interest	float
school	The number of school within 1000m radius of the point of interest	float
landscape design effort	The level of landscape design features in the park	integer

Table 4. Target Variable (y) Description in the Dataset.

Column Name	Description	Data Type
real_visitor_poi	Weekly total visitation of the point of interest	float

Hyperparameter tuning

To attain peak performance, the model must be fine-tuned. Due to the large number of hyperparameters in XGBoost, tuning it might be challenging. These settings are divided into four categories: general, booster, learning task, and command line. Tuning can be performed using either a grid or a random search. This paper makes use of the grid search. Grid search finding the optimal solution might be challenging when the parameter dimension is large. This is easily accomplished by focusing on a smaller set of parameters with suitable parametric ranges at a time. During the model selection step, K-fold cross-validation is used to evaluate the model's performance. The grid search is performed in the following manner.

- Keep 30 percent of the dataset as the test set and the remaining 70 percent to our XGBoost model (**Table 5.**). The “n_estimators” determines the epoche of the model is set to 100 and 500.
- The grid values for the optimal “learning_rate” are 0.03, 0.05, and 0.07, which are set to eliminate overfitting problems. All the values run for the model tuning, and one with the best performance is retained as the optimal value.
- After obtaining the optimal value of “learning_rate”, perform the grid search of “max_depth” and “min_child_weight” in the range from 1 to 10.
- Perform a grid search for “sub_sample” with values from 0 to 1. This affects the subsample ratio of the training instances and prevents overfitting.
- Last do a grid search of “colsample_bytree” with values from 0 to 1.

Table 5. The number of entries of the data.

Dataset	Number of Rows	Number of Columns
Train	19,800	21
Test	8,487	21

The number of cross-validation of the grid search is set to 2, which means two cross-validations will perform for each selected set of hyperparameters. The "n_jobs" is set to 5, meaning that five processes will run in parallel. After all the grid search steps, the optimal hyperparameters are generated in **Table 6**. Note "nthreats" is set to 4, meaning the XGBoost model will run four parallel threats; "objective " means the objective function is set to "reg:squarederror" since our prediction is predictive regression modeling.

Table 6. Optimal hyperparameters.

Hyperparameters	Optimal Values
n_estimators	500
learning_rate	0.07
max_depth	10
min_child_weight	2
subsample	0.5
colsample_bytree	0.7

Results and Findings

After implementing the optimal hyperparameters from the tuning step, the XGBoost model carried out the prediction of park visitation. The coefficient of determination score (R2 score) on the testing data is 0.8495, which means that 85 percent of the changeability of the dependent output attribute can be explained by the model, while the remaining 15 percent of variability stays unaccounted. The difference between actual park visitation and predicted visitation value is in **Figure 2**, where the x-axis is the number of rows of the whole dataset and the y-axis is the difference. The scatterplot shows most of the differences lay between negative 2,000 and 1,000, with several outliers on the bottom of the plot, where the predicted values are way larger than the actual values.

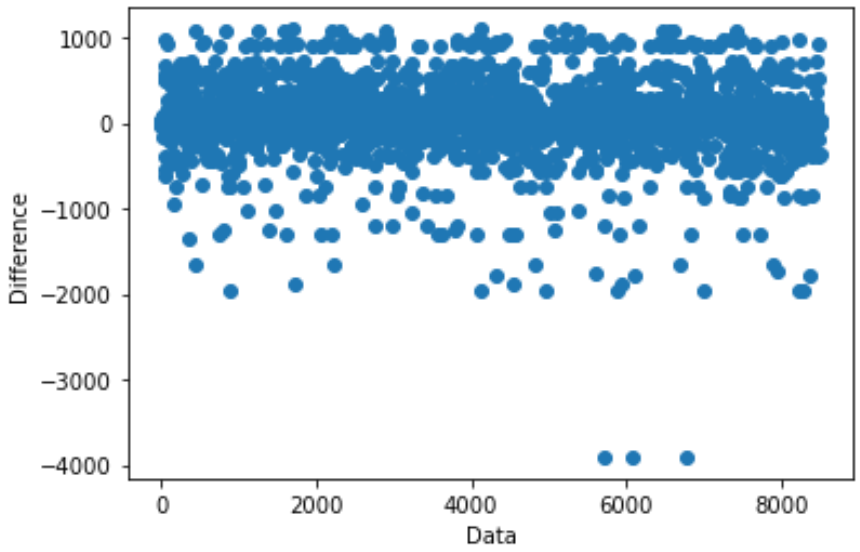

Figure 2. Difference between real visitation and predicted visitation

By looking at the errors greater than 3,000, the park was located. It was Stephen C Beachy Central Park in the week of June 29, 2020, the middle of summer vacation at Texas A&M

University and Blinn College. Many students and families travel out of the College Station - Bryan metropolitan area.

The feature selection result is accomplished by mapping feature importance to the XGBoost feature importance plot, see **Figure 3.** F-scores in the feature importance context simply means the number of times a feature is used to split the data across all trees. In Figure x, the F-scores of the features neighborhood park, apartment, and hotel are close to zero. So these features are excluded from the model. The new selected feature importance was mapped again in **Figure 4**, with a new R2 score of 0.8498. After performing the feature selection, the R2 score didn't change significantly. **Table 7.** shows the comparison of the R2 score based on the feature selection.

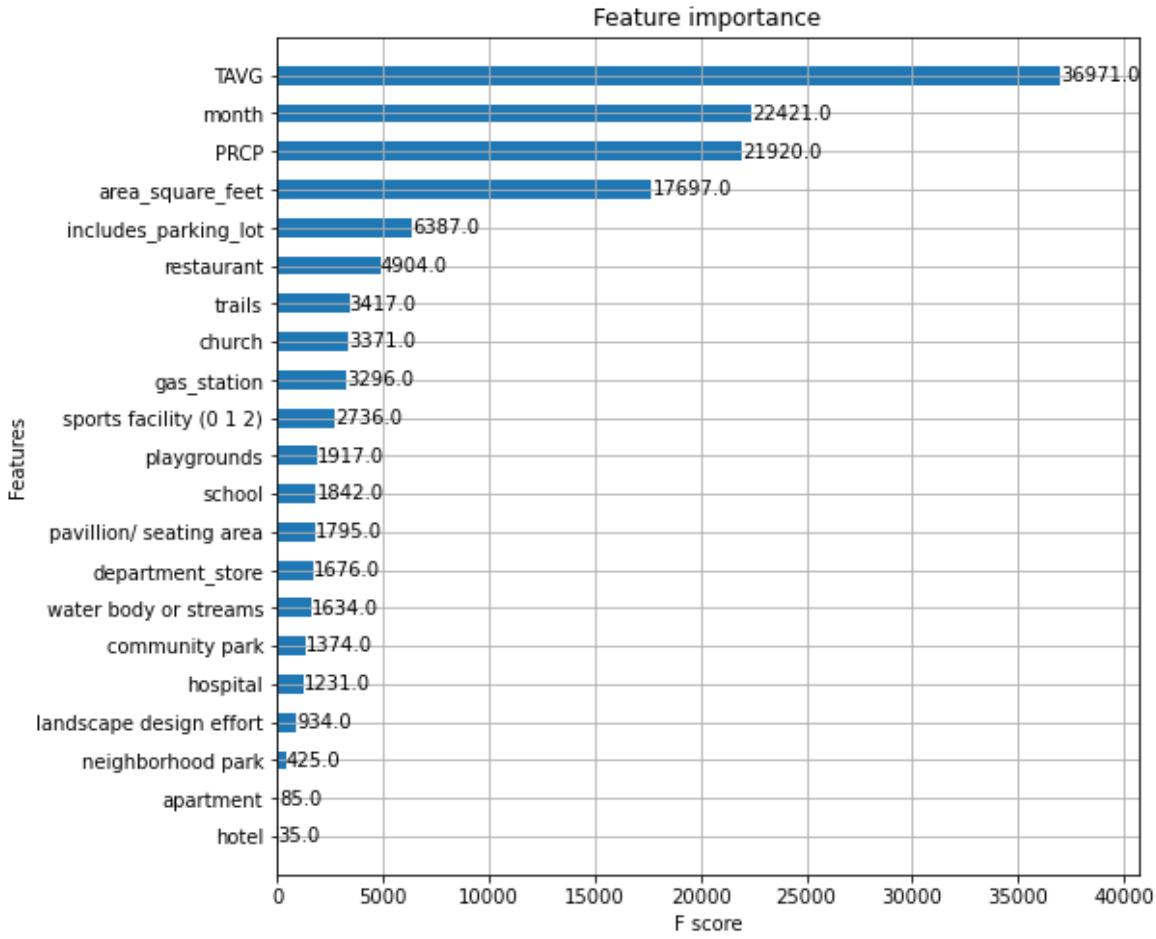

Figure 3. Map of feature importance

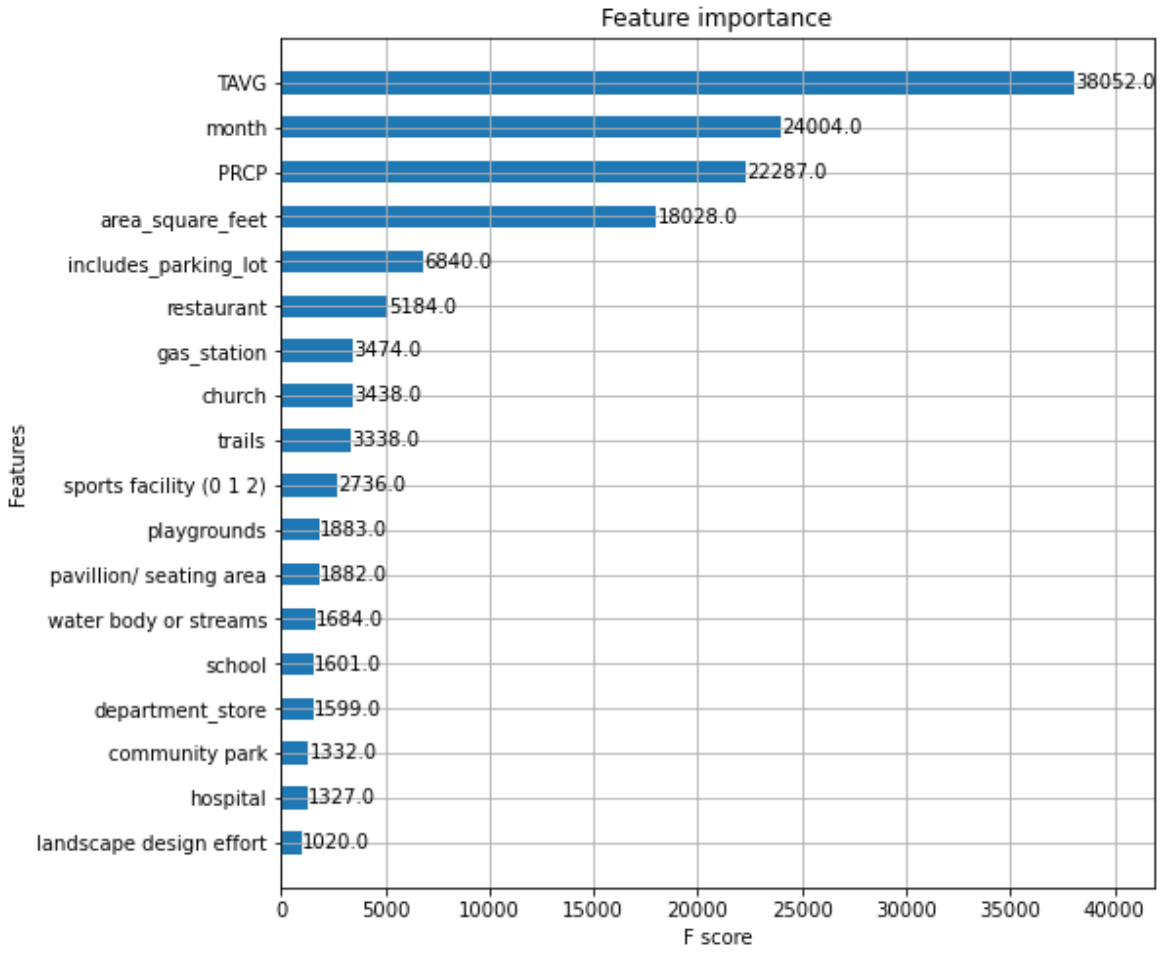

Figure 4. Map of new feature importance

Table 7. R2 score based on feature selection.

Model	Feature excluded	R2 score
All baseline features	-	0.8495
Feature selection	neighborhood park, apartment and hotel	0.8498

Figures 5, 6, 7, 8, and **9** are the most important features related to the park visitation.

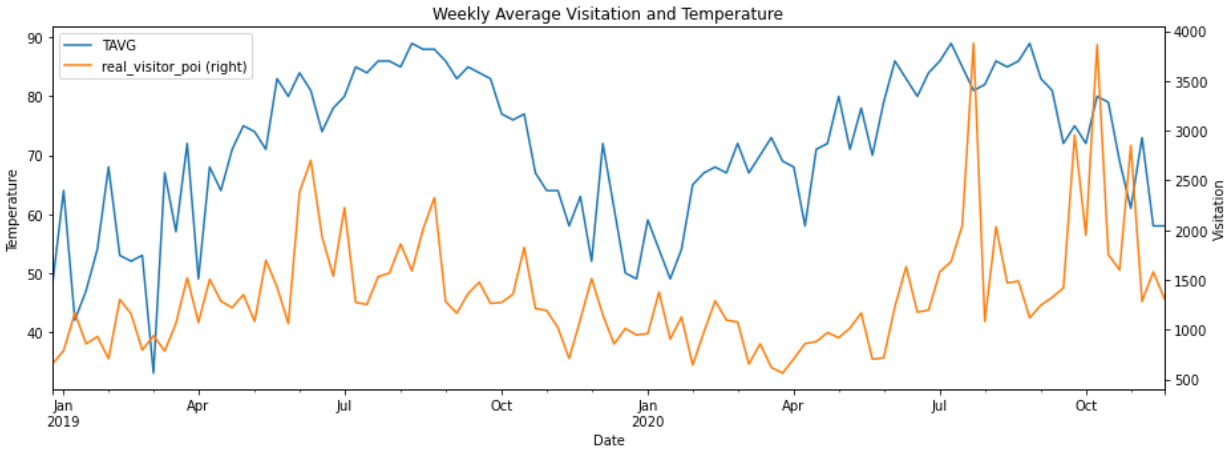

Figure 5. Weekly average visitation and temperature of all parks

Figure 5. is the weekly average visitation and temperature of all parks. The blue line is the weekly average temperature, and the orange line is the weekly average visitation. In 2019, the weekly average visitation grew when the weekly average temperature increases. When the weekly average temperature is close to 90 degrees Fahrenheit in August and September, the average visitation starts to decrease, and the decreasing trend continues throughout the rest of the year. In 2020, because the COVID 19 pandemic hit College Station & Bryan metro area in March, the average weekly visitation remained low regardless of the increase in the weekly average temperature. The visitation started increasing in June and reached a peak in August, and decreased in the following months. The growth of weekly visitation started again in October and decreased afterward due to the low temperature.

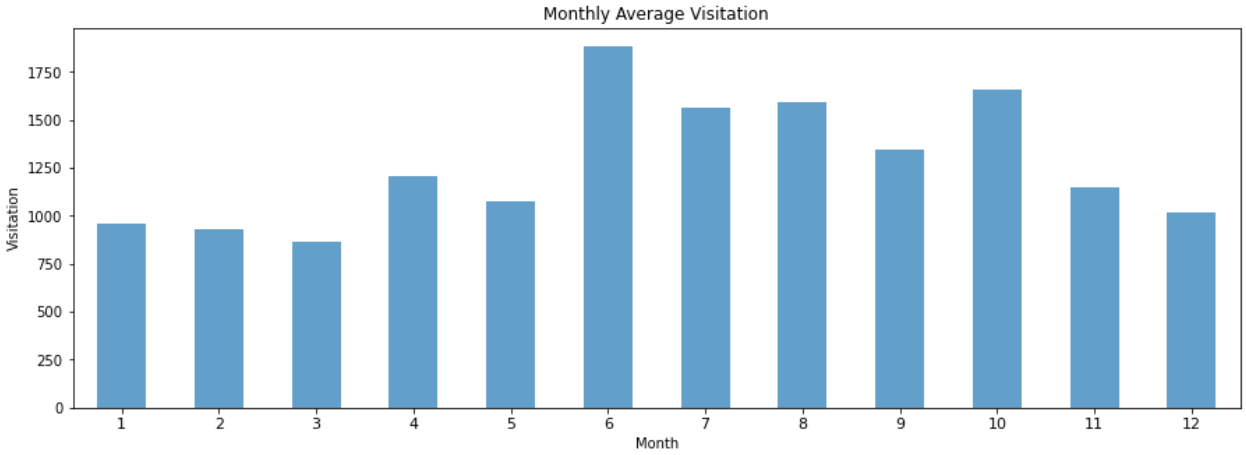

Figure 6. Monthly average visitation

Figure 6. is the monthly average visitation. From January to May, the visitation stays the same. Starting from June, the visitation starts increasing and drops in November and December.

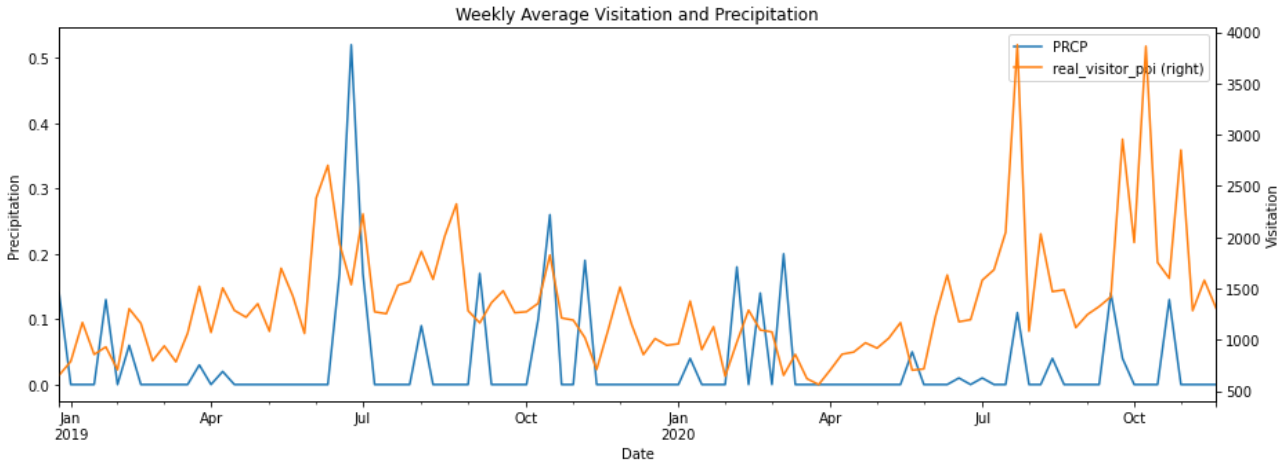

Figure 7. Weekly average visitation and precipitation of all parks

Figure 7. is the weekly average precipitation has an impact on the weekly average park

visitations. The blue line is the weekly average precipitation, and the orange line is the weekly average visitation. Generally, throughout the two years, when the average precipitation is less than 1 inch or zero, the average park visitation increases. But the visitation remains high in summer, especially in July 2019, regardless of the high average precipitation this month.

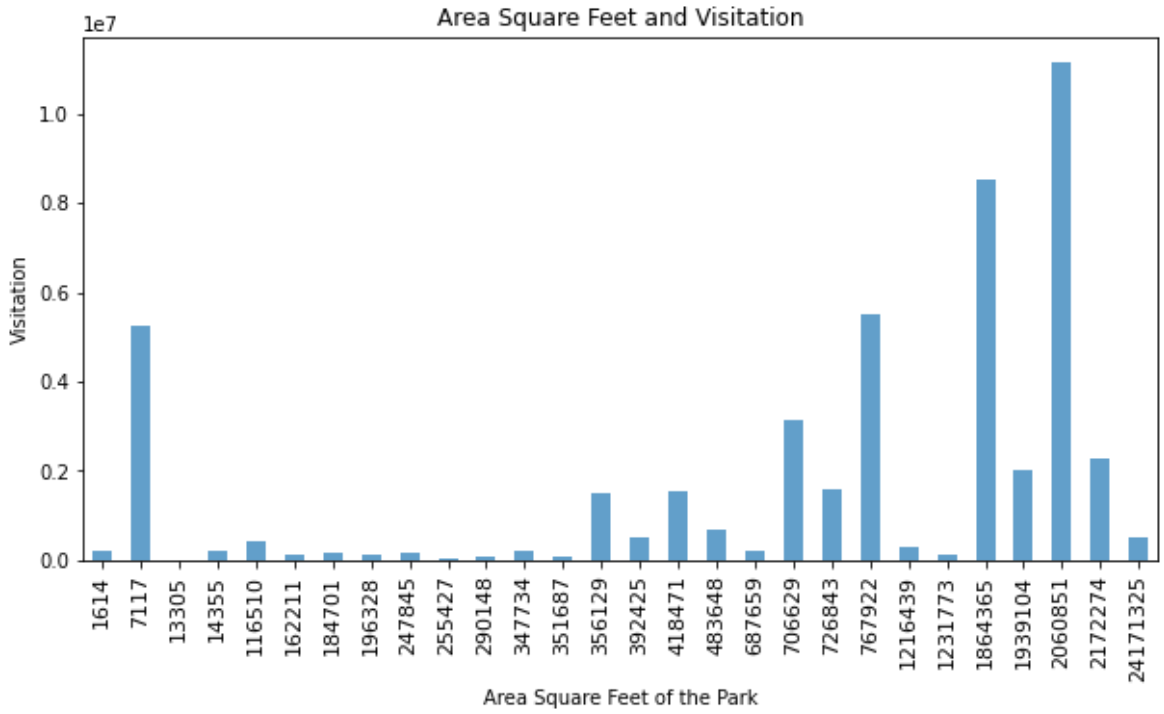

Figure 8. The area square footage and the park visitation

Figure 8. is the summation of visitation with different area square footage of the park. Generally, the smaller the parks are, the less visitation the parks have. When the park area is greater than 700,000 square feet, the visitation starts increasing.

Figure 9. Median visitation of parks with or without parking lots

Figure 9. is the median visitation of parks with or without parking lots. Apparently, parks without parking lots have lower median visitation than parks with parking lots.

Conclusion

In this study, the XGBoost model performed the prediction of park visitation in the College Station & Bryan (CSB) metropolitan area, training and testing data were extracted from the dataset, with testing data being 0.3 of the whole dataset. After the feature selection, the R2 score of the model is 0.8498. According to the feature importance, the weekly average temperature and precipitation, month of the year, area square feet, and whether the park includes parking lots have the highest importance in the visitation, followed by the number of restaurants, gas stations, churches, and trails.

Discussion

The implementation of park visitation through smartphone data prediction utilizing existing park features, social demographics, weekly temperature, and precipitation data, as well as the Google Places API for adjacent points of interest contributes to the development of a more comprehensive perspective on park visits. Using the XGBoost model will generate the feature importance which helps the understanding of how features affect the visits. Future planners, landscape architects, and park administrators should evaluate their communities' distinctive characteristics and prioritize the most important variables that may influence park visitation. Moreover, The National Recreation and Park Association (2019) report show the reduced incidence of obesity and obesity-related illnesses is closely linked to increasing physical activity. The application of this study would help the implication health cities in the United States, improve the health conditions for the residents such as lower the obesity rate.

Additionally, both College Station and Bryan are midsized cities in the United States. Their comparable sizes to the majority of other US cities make this study more reflective of the majority of US cities, making its conclusions more generalizable. From a larger perspective, the approach used in this study is simply applicable to any problem involving site selection, not only parks.

There are several limitations subject to this study. First, the study uses the mobility patterns from BGs within the CSB metropolitan area, while park visitation conducted by persons

living outside of the CSB metro area was excluded. Second, SafeGraph data primarily covers mobile device mobility; visitors who do not own smartphones, such as children and teenagers, were excluded from the analysis. Finally, the study is based on the College Station and Bryan metropolitan area, which does not reflect all US mid-sized cities. The findings may be limited to college towns due to the influence of Texas A&M University, such as academic holidays and football seasons. Outside of Texas, regions with a different climate and socioeconomic environment may potentially provide results that differ from this particular study.

References

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. https://doi.org/10.1145/2939672.2939785

Dhaliwal, S. S., Nahid, A.-A., & Abbas, R. (2018). Effective intrusion detection system using xgboost. *Information*, *9*(7), 149. https://doi.org/10.3390/info9070149

Green Infrastructure Health Literature Review. National Recreation and Park Association. (2019, July). Retrieved April 16, 2022, from https://www.nrpa.org/contentassets/0aa1178a2d944cbc8cb6fbc5ce31b266/green-infrastructure-health-literature-review.pdf

Introduction to Boosted Trees—Xgboost 1.5.2 documentation. (n.d.). Retrieved March 12, 2022, from https://xgboost.readthedocs.io/en/stable/tutorials/model.html

Leeworthy, V. R., & Wiley, P. C. (2001). Current participation patterns in marine recreation. *National oceanic and atmospheric administration*.

Non-traditional methods to obtain annual average daily traffic (Aadt) evaluation and analysis | washington state transportation center. (n.d.). Retrieved March 10, 2022, from https://depts.washington.edu/trac/current-projects/non-traditional-methods-to-obtain-annual-average-daily-traffic-aadt-evaluation-and-analysis/

Park classifications | dallas parks, tx—Official website. (n.d.). Retrieved March 11, 2022, from https://www.dallasparks.org/151/Park-Classifications

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *the Journal of machine Learning research*, *12*, 2825-2830.

Places data & foot traffic insights | safegraph. (n.d.). Retrieved March 9, 2022, from https://www.safegraph.com

Song, Y., Newman, G., Huang, X., & Ye, X. (2022). Factors influencing long-term city park visitations for mid-sized US cities: A big data study using smartphone user mobility. *Sustainable Cities and Society*, *80*, 103815. https://doi.org/10.1016/j.scs.2022.103815