Location Identification Problem

Wei Liu, wliu40@syr.edu

Abstract

In this short paper, we analyzed the possibility of using the past record of user's GPS location to predict his/her preference toward different establishments. The given data is 1) the GPS locations without timestamp and 2) the locations and types of establishments we are interested in. The given data was pro-processed and explored first, and then different algorithms were built to identify the clustering of the GPS data. Then the final algorithm was designed and implemented to calculate the ranking scores for the user based on his/her current location. At last, we discussed the conclusion and possible future work needed to prompt our work, and strategies to enhance the performance of our model.

1. Background

Wearable devices have wide applications from health monitoring system to intelligent agents. Most of wearable devices are able to collect the users GPS information, and then use those data to interactively assist the user in a variety of tasks in everyday life. Through the locations information, the intelligent agents could potentially set up a predictive model, which is able to predict the future physical location or recommend new locations to the user.

To make such a wearable device be smart, first thing is to acknowledge how it could help the users to make better decisions. The following are the scenario we could imagine when an intelligent agents, which has learnt the user's frequented location preference, could interactively help the users in daily life: 1) the device could automatically display a shopping list when a user approaches a grocery shop; 2) if the user loves Chinese cuisine, the device could recommend new restaurant nearby based on the user's past visited history; 3) the device could suggest the users jogging the next day, since the device has observed the user had routine running activity on Sunday, and the device knows it will be lovely weather the next day; 4) the device could predict the user is going to work at 7:30am in the morning, then it could remind you that there was a traffic jam on the highway, thus you could choose alternative route; 5) the device is able to speculate the user's job or identity, e.g., if the user's location mainly distributed around an university campus, he/she has high possibility to be a college student.

Learning from the past locations is the essential part to build such an intelligent system.

Most wearable devices has GPS installed and thus could gather the user's locations information at a specific frequency. In such a system, we could collection the user's physical location in a time sequence. Time is an essential part in the <location, time> pairs, since we can calculate many other information from time, for instance, the duration time the user stays at a spot, the longer time users spend may indicates he/she has more interest in that spot. We can also obtain the user's speed at each time, if we have other information (e.g., heart beat rate), we will be able to infer that whether the user is driving, running, walking, sitting or taking indoor exercises.

To build such an intelligent system, the recognition of the common people's routing life is also important since it could help our analysis of the GPS data. For most people have daily routine life, we could expect that the GPS locations data would be highly skewed distributed in both space and time domain. Most of people commute between work place and home in work days, and then in weekend, the user might has higher possibility to appear in grocery stores, parks or other entertainment places. So, we could expect several clusters on the map, while each cluster represents work place, home or the park/grocery store the user visited. Beyond that, we could also expect the distance between each clusters would be much larger than the radius of the clusters. It was reported that Americans spent 100 hours per year for commute on average. Assume the average driving speed is 50 km/h, and there are ~260 work days each year, we could infer that the average distance from home to work is about 19.2 km.

If we use bright dots to represent GPS location to represent a person's daily activity, and the brightness represents his/her staying time, then for most people, the physical appearing locations on the map would be 2~5 bright dots sparsely distributed, and several other dimmer dots represents the grocery stores, restaurants, gas station and so on. While for college students, it might be a bigger dot represents the dorm, classrooms and dining hall, which are more closely located. For a postman, the dots would be sparser and evenly distributed in a large range. From those feathers of GPS location distribution, we could infer the user's career identity.

2. Data Pre-processing and Exploring

With all the above premise and inference in mind, we could start to analyze the GPS data we were given.There are 160 points of GPS data are given, which represents the user Jone's location information in the past 48 hours, but we don't have the time stamp for those data. The format of GPS is very clean and easy to process. The information of each establishments were also given, include the name, type and the GPS location of the establishments. However, those data are not as clean as user's locations, thus we need to abstract useful information from the mess. For the extensible usage in the future, an establishment class was built, which holds all the information an establishment should have, and we could add more features if necessary.

	establishment_latitude	establishment_longitude	establishment_name	establishment_types
0	35.7795897	-78.6381787	Raleigh	locality
1	35.7992765	-78.6899516	Carlyle Campbell Library	library
2	35.79958279999999	-78.6908077	Frankie G. Weems Art Gallery	art_gallery
3	35.7989103	-78.6909706	Martin Lot Staff	premise
4	35.823483	-78.8255621	Morrisville	locality
5	35.8577575	-78.8359571	UNC Health Care's Morrisville campus	point_of_interest
6	35.8577575	-78.8359571	ISD UNC Health Care	point_of_interest
7	35.79154	-78.7811169	Cary	locality
8	35.7896757	-78.8700624	Starbucks	cafe
9	35.7892183	-78.870043	Biscuitville	cafe
10	35.79154	-78.7811169	Cary	locality

Fig.1 The pre-processed establishments dataframe

Fig.1 shows the Dataframe after clearning. From the Dataframe, we observed some of the points have overlapped. This could be treated in the following three cases:

1) Data input error, one of the duplicated locations has a false GPS location or false name, this case needs to be corrected.

2) Some establishments are in the same building, and this is very common in reality. In this case, we could only use the establishment's GPS information rather than names, and types, because we have no way to come up which establishment the user have been, unless we could infer the establishment from other clues or the user tells us directly. This case would be common in big city regions, e.g., Manhattan of NYC, where many companies and facilities share the same building.

3) Some of the establishment could be included into its higher level name, for example, a

GPS of a university could covers its GPS of library, or a GPS of a branch of a company would be the same with its upper level company name. In this case, we can zoom in or zoom out the structure of the establishment and pick the name or the type that most interest us.

```
list_estabs = reader.get_establishments()
print(str(list_estabs[0]))
```

```
duplicated cords found: Estab_name: ISD UNC Health Care; types: point_of_interest; GPS: (35.8577575, -78.8359571)
duplicated cords found: Estab_name: Cary; types: locality; GPS: (35.79154, -78.7811169)
duplicated cords found: Estab_name: Morrisville; types: locality; GPS: (35.823483, -78.8255621)
Estab_name: Raleigh; types: locality; GPS: (35.7795897, -78.6381787)
```

Fig. 2. Duplicated GPS of establishment detected and displayed

Fig.2 shows the output of those duplicated GPS locations in the establishements collections. Further treatment is needed depends on different cases. In this paper, those duplicated data was deleted and only one copy was left randomly.

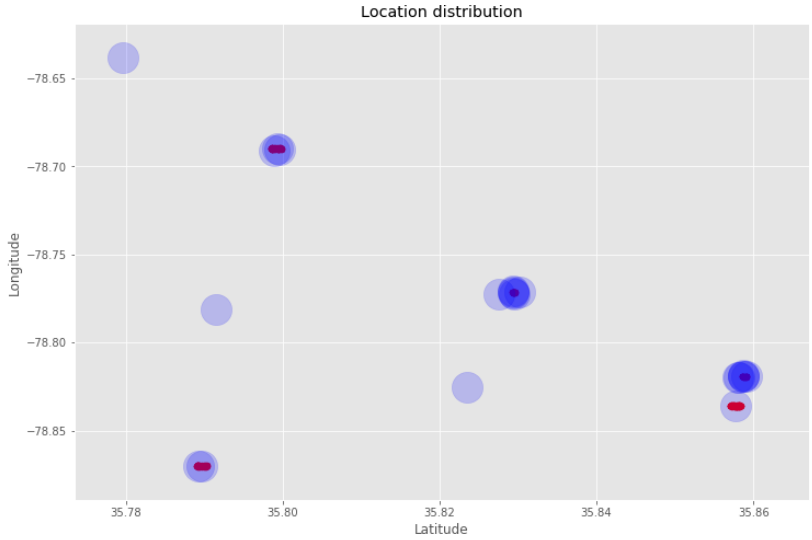

Fig.3 Data clusters on 2D map

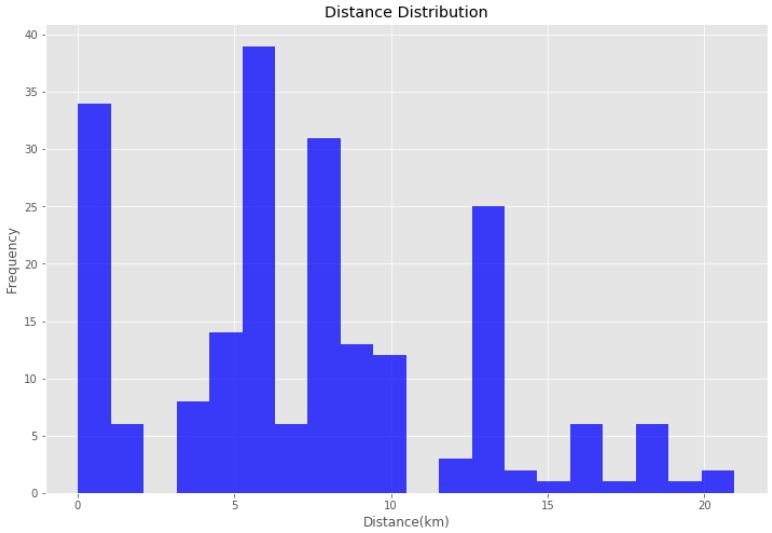

Fig. 4 Histogram of the distances between establishmens in dataset

Fig.3 is the plot of all the data points on the map. The bigger translucent blue dots represent the establishments we are interested in, while the red dots represent the GPS locations of the user. We could see that our preliminary assumption on the GPS point's distribution is validated. The data was sparsely divided into clusters and the distance between clusters is magnitude larger compares with the radius of the cluster.

Fig.4 is the distance distribution of the different establishments we are interested in. We can see the data is skewed distributed to the left of the center. We could imagine that the data would be more skewed to the left in big cities, and skewed to the right in rural area. If the distances between establishments are mostly distributed to the left, then this may bring extra difficulties to our clustering algorithms, e.g., k-means.

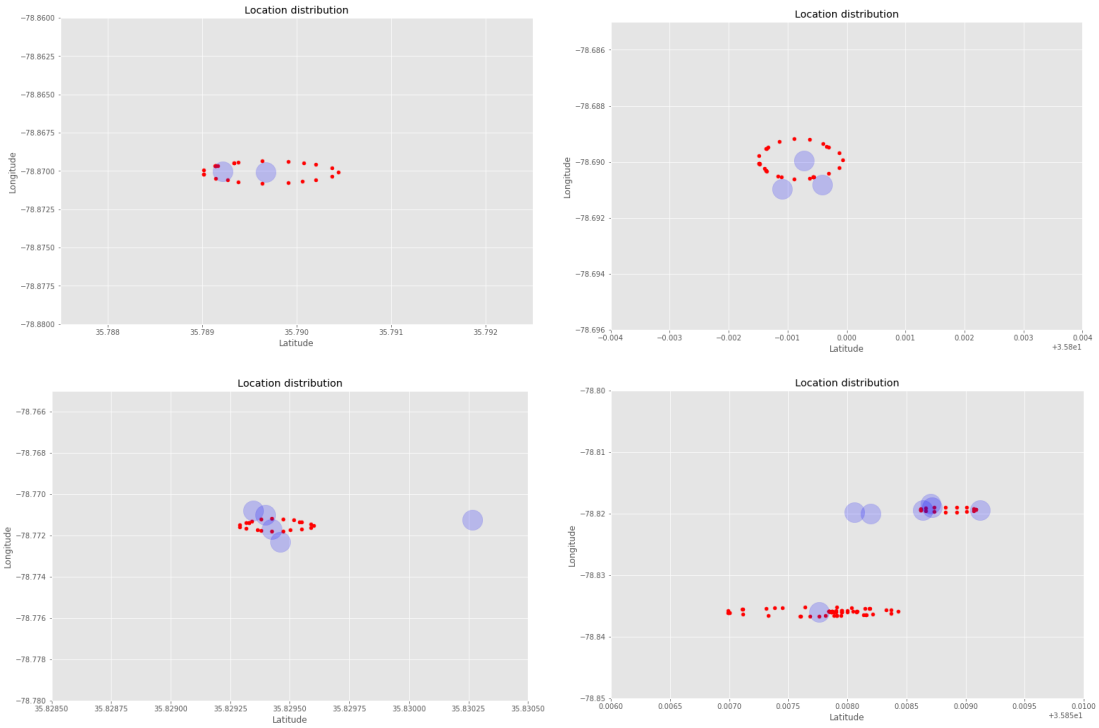

Fig. 4 Zoomed in the data cluster

Fig.4 is the zoomed in distributions of the GPS record of the user John. It is strange that the GPS shows that John seems have been walking around in a circular way. If the GPS data comes from wearable devices, e.g., a watch or phone, and John spent most of his time in his office, then the data may have a higher chance to be in Gaussian distribution considering the GPS drift noise. From the fig.4 we also observed that the clusters are clearly separable, even in this fine granularity level.

3. Algorithm Design

After the preliminary understanding of the data, we can start exploring the algorithms we can use. The model should be able to predict the next stop of the user, solely based on those 160 data points in the past 48 hours. Without the timestamp for each of GPS data, we will lose a lot of valuable information, but some of rules can still lead our analysis. The first thing we want to know is what establishments John has been to, the higher visited times and longer stay time in the establishment, and the higher possibility of John will re-visit it again. For example, it is most likely John would visit his top favorite store rather than other stores he seldom entered. Secondly, the distance of John's position to each of those establishments is another factor for consideration. For example, if John is at a location very near a grocery store he seldom visited, than it is highly likely John is going to visit that store rather than his favorite store far away. Thirdly, if John has the same favorable score toward a café and a library, and now John stands in the middle between those two spots, the algorithms could give the same score to the café and the library. If we were given the current time, e.g., it is lunch time, and then the chance of entering the café would be much higher. Thus, the type of the establishment can provide little information for our decision making process without knowing the current time.

To find the most significant establishments for John, we built a function based on the distances of each of the establishment with all the GPS points. In this method, we assign each of the GPS point to its closest establishment. This is the same idea with the k-means method, where we only do the first step of k-means without the centroid update. Table 1 depicted this algorithm in pseudo code.

Table 1 Algorithm 1 to find the most important establishment for the user

Algorithm 1
initialize cluster using the establishment's location *for each GPS record:* *compute the distances of this point to all the centroids* *assign this point to the centroid which has the minimum distance* *rank the centroids with the number of those points*

Table 2 Establishments and visited time

Name of the establishments	Cluster size
UNC Health Care's Morrisville campus	63
Extended Stay America Raleigh - Cary - Harrison Ave.	17
Carlyle Campbell Library	16
Biscuitville	14
Starbucks	11
Waffle House	8
Frankie G. Weems Art Gallery	7
Airport Blvd at Aerial Center Pkwy (Waffle House)	7
Martin Lot Staff	7
CapriFlavors	5
Hampton Inn Raleigh-Durham Airport	5

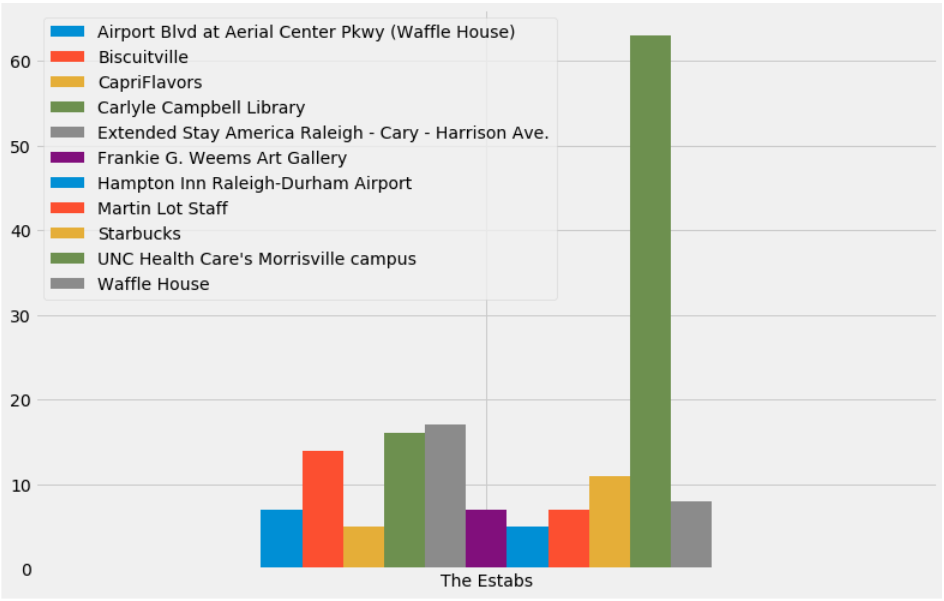

Fig.5 Histogram of the establishments and their cluster size

The most intuitive way of interpreting the cluster size is the importance of that cluster. Because we don't know the timestamp, we assume those data points were sampled evenly at same time interval in the last 48 hours. So, the importance of an establishment would be positively correlated to its cluster size. Table 4 is the output of the algorithm 1. Fig.5 is the histogram of those establishments with their cluster size.

Table 3 Algorithm 2 to find the most important establishment for the user

Algorithm 2 customized k-means
initialize cluster using the establishment's location
repeat
for each GPS record:
compute the distances of this point to all the centroids
assign this point to the centroid with the minimum distance
recalculate the centroid based the new cluster
until the centroid not change
assign all the points in one cluster to its nearest neighbor of establishment

Table 3 is the pseudo code of applying complete k-means algorithm on the same datset. As we know, k-means working not well if the data distribution has the following features:

1) The clusters are not sphere shape;
2) The points are distributed in the way that cluster has large variety of density;
3) The initial seed points are poorly chosen;

For our algorithm, we choose the establishments centers as the seed, thus we could greatly improve the robust of the program. This will be essential, since for this specific problem, the GPS points would be magnitude larger than the establishment points. We will have a higher chance to select the initial seed badly. In this program, a function was added to double check each radius of the clusters, thus to make sure we had the global optimized solution. Also, we use the physical distance of two GPS points as our distance function rather than the Euclidian distance of latitude and longitude pairs in algorithm 2.

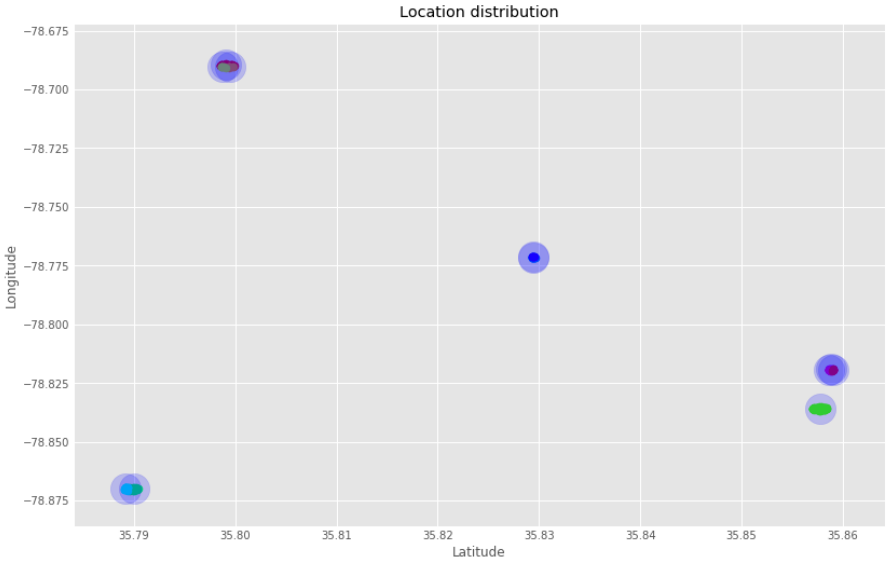

Fig. 6 Clusters after running k-means algorithm

Fig.6 is the plot after running the algorithms 2, we noticed that each cluster was marked with different colors. In Fig.6, we also deleted the establishments that cluster size equals zero.

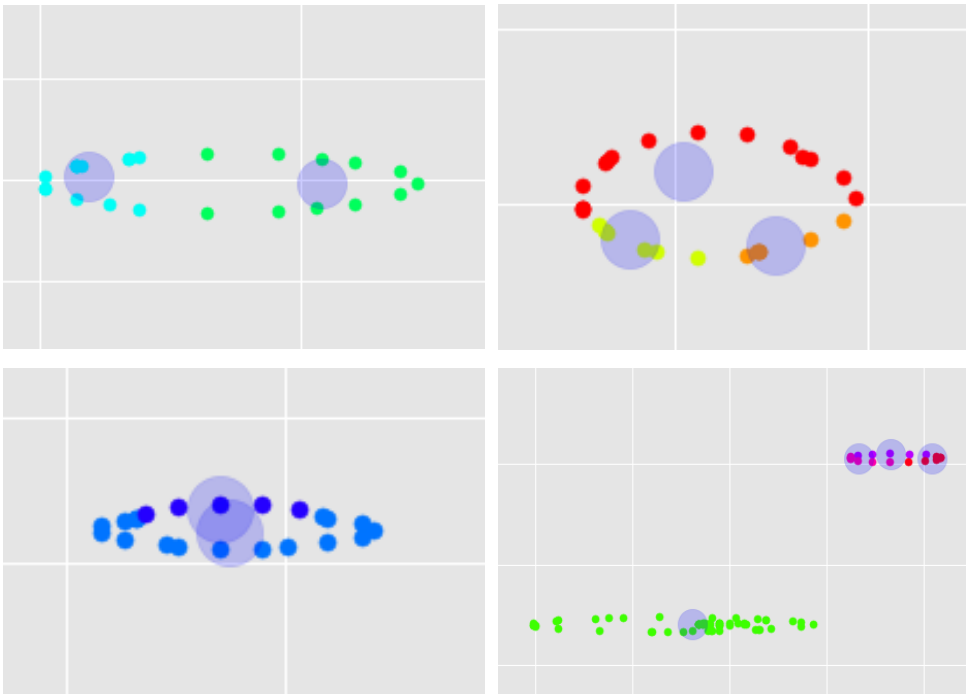

Fig. 7 Zoom in graph for each cluster

Fig.7 is the zoomed in graph for each cluster. We could see the clear boundary for each cluster around each establishment.

Table 4 Cluster radius from algorithm 2

Radius of each cluster (km)
0.10335779415809133
0.04632939844635364
0.04264085028208585
0.09233317704947942
0.08236399876431565
0.074870370812517
0.022857645183081133
0.011635469032824917
0.020711604244885863
0.031151736689348022
0.02926714038702704

Table 4 is the radius (in km) for each of those clusters; we can confirm that the algorithm has converged to the global minimum. The maximum radius found is ~100m, which makes sense in a person's daily activities.

Since we have added the centroid-update phase in algorithm 2, so, the centroid will have a shift toward the original establishment location. Through Table 5, we can see this shift is negligible.

Table 5 Comparison of the updated centroid with the original establishment centers

Updated centroids	Establishment centers
[35.79904067, -78.68963169],	[35.7795897, -78.6381787],
[35.79952484, -78.69048433],	[35.7992765, -78.6899516],
[35.7987671 , -78.69041078],	[35.7995827, -78.6908077],
[35.85780888, -78.83597282],	[35.823483, -78.8255621],
[35.79007535, -78.87007741],	[35.79154, -78.7811169],
[35.78918686, -78.86992024],	[35.7896757, -78.8700624],
[35.82943708, -78.77157353],	[35.7892183, -78.870043],
[35.82942633, -78.77123347],	[35.8302672, -78.771261],
[35.8588325 , -78.81899968],	[35.827521, -78.7724144],
[35.85866709, -78.81946191],	[35.8581983, -78.8199726],
[35.85904663, -78.81944097],	[35.8587064, -78.8184502],

The output of the algorithm 2 is exactly the same with the algorithm 1. This is not surprising, because all the clusters are dense and we have initialized the seed with the same value. But for larger dataset, the output result may different, since the centroids would shift more drastically.

After clustering, we can estimate the importance of each establishment to John. For this part, we defined a function:

$$f_1^i = \frac{C_i+\theta}{\sum_i(C_i+\theta)} \qquad \text{Equ. (1)}$$

f_1^i is the favorable degree of the user to each establishment i.

C_i is the cluster size for the establishment i.

θ is the threshold number we set, in case some of the establishment cluster size are zero.

Now based the past GPS record and the output from algorithm 1/2, we will get the following output for each establishment:

Table 6 f_i^1 score for each establishment

f_i^1 score
UNC Health Care's Morrisville campus 0.35359116022099446
Extended Stay America Raleigh - Cary - Harrison Ave. 0.09944751381215469
Carlyle Campbell Library 0.09392265193370165
Biscuitville 0.08287292817679558

```
Starbucks 0.06629834254143646
Waffle House 0.049723756906077346
Frankie G. Weems Art Gallery 0.04419889502762431
Martin Lot Staff 0.04419889502762431
Airport Blvd at Aerial Center Pkwy (Waffle House) 0.04419889502762431
CapriFlavors 0.03314917127071823
Hampton Inn Raleigh-Durham Airport 0.03314917127071823
Raleigh 0.0055248618784530384
Morrisville 0.0055248618784530384
Cary 0.0055248618784530384
Jiffy Lube 0.0055248618784530384
Bass Pro Shops 0.0055248618784530384
Harrison Oaks Blvd at Weston Pkwy 0.0055248618784530384
The Arboretum 0.0055248618784530384
Days Inn Raleigh-Airport-Research Triangle Park 0.0055248618784530384
Daly Seven Inc 0.0055248618784530384
1006 Airport Blvd Parking 0.0055248618784530384
```

Now we consider the current distance of John to each of the establishments. From the previous analysis, we can anticipate the possibility of the user entering an establishment should be negatively correlated. The following function for evaluating the influence of the distance was used:

$$f_2^i = \delta \frac{\sum_i \exp(d_i)}{d_i} \qquad \text{Equ. (2)}$$

$$f_2^i = \frac{f_2^i}{norm\langle f_2^i \rangle} \qquad \text{Equ. (3)}$$

d_i is the distance from current position to i^{th} establishment;

δ is a small number to avoid overflow of float number, we choose 0.00001 as default;

f_2^i is the score to the establishment i considering the distance between them.

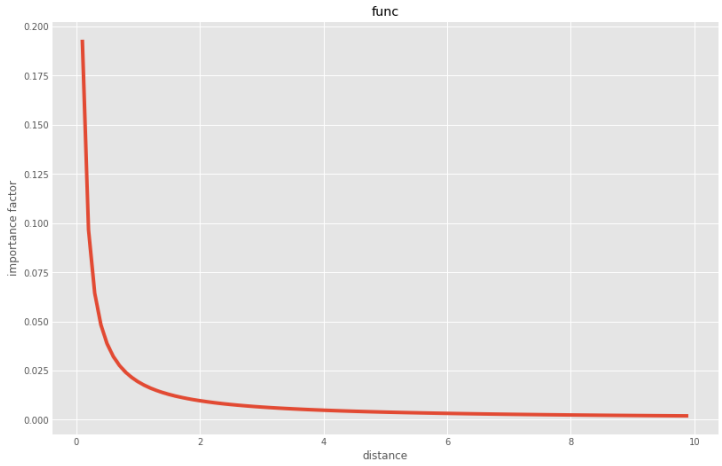

Fig.8 is the plot of f_2^i from 0.1 km to 10 km.

Fig.8 is the plot of f_2^i from 0.1 km to 10 km. This makes sense considering a large distance would makes it impossible to anticipate the user's destination, and when the user approaches an establishment very close, the chance of entering that establishment would be exponentially increased.

We combined the f_1^i and f_2^i in a linear format to get the final score:

$$f_i = \alpha f_1^i + \beta f_2^i + \gamma \qquad \text{Equ. (4)}$$

α is the coefficient number for f_1^i;

β is the coefficient number for f_2^i;

γ is the bias term.

Now we can test our algorithms with input. The following output is when position = [35, -78], the average distance to those establishments are 110 – 120km, we also set set $\alpha = 1$, $\beta = 1$, $\gamma = 0$, the following output was obtained.

From the output, we could infer that since the distances are very far, so f_2^i are basically at the same level, and then the f_1^i score would play an important role.

Table 6 f_i score when the user is far from all of the establishments

f_i score: pos= [35, -78], $\alpha = 1$, $\beta = 1$, $\gamma = 0$
UNC Health Care's Morrisville campus 0.199461047543
Extended Stay America Raleigh - Cary - Harrison Ave. 0.0735742397842
Carlyle Campbell Library 0.0723636507655
Biscuitville 0.0648334235739
Starbucks 0.0565384274083
Waffle House 0.0476864513519
Martin Lot Staff 0.0474971043901
Frankie G. Weems Art Gallery 0.0474848491495

Airport Blvd at Aerial Center Pkwy (Waffle House) 0.0449163876743
CapriFlavors 0.040433344548
Hampton Inn Raleigh-Durham Airport 0.0394026779444
Raleigh 0.0292505822731
Cary 0.0272007870095
The Arboretum 0.0266394326344
Jiffy Lube 0.0266245730317
Harrison Oaks Blvd at Weston Pkwy 0.0266053024836
Bass Pro Shops 0.0266024515821
Morrisville 0.0261081501544
Daly Seven Inc 0.0255960610255
1006 Airport Blvd Parking 0.0255925868683
Days Inn Raleigh-Airport-Research Triangle Park 0.0255884688046

The following output in Table 7 is when position = [35.8580603, -78.819797499],$\alpha = 1$, $\beta = 1$, $\gamma = 0$. This is the case when the user is very close to 1006 Airport Blvd Parking, Even this is the most unfavorable spot for the user, the algorithm correct predict he will enter the establishment.

Table 7 f_i score when the user is close to one of the establishments

f_i score: pos= [35.8580603, -78.819797499], $\alpha = 1$, $\beta = 1$, $\gamma = 0$
1006 Airport Blvd Parking 0.502762427071
UNC Health Care's Morrisville campus 0.176795580141
Extended Stay America Raleigh - Cary - Harrison Ave. 0.0497237569144
Carlyle Campbell Library 0.0469613259702
Biscuitville 0.0414364640934
Starbucks 0.0331491712758
Waffle House 0.0248618790675
Airport Blvd at Aerial Center Pkwy (Waffle House) 0.022099447875
Frankie G. Weems Art Gallery 0.0220994475172
Martin Lot Staff 0.0220994475172
Hampton Inn Raleigh-Durham Airport 0.01657458606
CapriFlavors 0.0165745856436
Days Inn Raleigh-Airport-Research Triangle Park 0.00276243297071
Daly Seven Inc 0.00276243125617
Morrisville 0.00276243095075
Harrison Oaks Blvd at Weston Pkwy 0.00276243094761
Bass Pro Shops 0.00276243094757
Jiffy Lube 0.00276243094743
The Arboretum 0.00276243094742
Cary 0.00276243094469
Raleigh 0.00276243094164

4. Discussion and Future Work

From the implement of the algorithm and testing on the data, we can draw the following conclusion:

1) Data preprocessing would be important, considering that the input establishment data are not in clean format, and there are also duplicated coordinates presents.
2) The GPS tracking history shows it has good properties for implementing k-means algorithm to find clusters.
3) The density of the establishments would affect the performance of k-means algorithms. But for most common cases, where most people just commute between home/work and other facilities, the distributions of the establishments that we are interested in are actually very sparse in space.
4) A linear combination of the favorable degree of the users toward different establishments and also the distance between them would be a good model for prediction. The correctness of the model with large/small distance was validated.

The pre-exploring of the data was proven to be essential for further algorithm design. However, since the dataset size is very small, we may expect to find more complicated patterns in the future work, thus in this way, we could optimize the algorithms accordingly.

The lacking of the timestamp brings many difficulties for our analysis. A variable of new features could be derived from the time parameter. For example, the speed of the user, the GPS signal distribution variation versus time could be informative. We could also decrease the space and time complexity of the GPS data, since we could recognize the GPS drift, and subtract these noises from the original dataset.

Since GPS signals are generated in time sequence, so we may build a powerful predictive model with deep learning algorithms, e.g., recurrent neural network. In this way, we could convert the problem from the unsupervised algorithms (k-means clustering) to supervised algorithms. Based on the user's GPS tracking record in time series, we could predict the user's next movement.