Amenity_index_Function

Yuxuan

25/05/2021

Import libraries

```
library(dplyr)
library(readr)
library(ggplot2)
library(tidyr)
library(imputeTS)
library(corrplot)
library(qwraps2)
options(qwraps2_markup='markdown')
library(hablar)
```

Import data

Requires two dataset; *google_reviews_poi_with_hours.csv* and *vancouver_facilities_2.csv*

```
review_poi<- read_csv("~/Desktop/MDS/data599/google-reviews-arts/google_reviews_poi_with_hours.csv")
van_poi<-read_csv("~/Desktop/MDS/data599/w2020-data599-capstone-projects-statistics-canada-transit/data/
```

```
#Merge review dataset with vancouver point of interest
left_join(review_poi,van_poi,by=c("poi_name"="name"))%>%distinct()->merged_data
```

Convert the data to numeric

```
merged_data%>% convert(num(Rating, Total_Review,open_days,Total_hours))->merged_data
```

Number of amenity in each type of arts facility

```
merged_data%>%group_by(type)%>%count()
```

```
## # A tibble: 9 x 2
## # Groups:   type [9]
##   type                                  n
##   <chr>                             <int>
## 1 art or cultural centre                5
## 2 artist                               48
## 3 festival site                         2
## 4 gallery                              99
## 5 heritage or historic site            28
## 6 library or archives                  86
## 7 miscellaneous                         6
## 8 museum                               92
## 9 theatre/performance and concert hall  75
```

Our primary interest would be gallery(n=99),library(n=86),museum(n=92) and theatre(n=75)

Weight Index function

Weight index function takes two arguments, *data* and *type*. *data* should contain the info of amenities such as poi_name,pid,Rating,Total_review,opening_hours,opening_days and total hours. The second argument is the amenity type of interest, for example, museum, gallery, library or archives,etc. Once the function has been called, it returns a list that contains the dataset with weighted amenity index,density plots, correlation plot with corresponding features.

```
weight_index<-function(data,Amenity="museuem"){
  data%>%filter(type==Amenity)->poi_type
  # select relevent features
  poi_type%>%select(poi_name,open_days,Total_hours,Rating,Total_Review)->poi_type
  # check number of missing data
  poi_type[poi_type == 0] <- NA
  missing_percentage<-colMeans(is.na(poi_type))

  # fill NA with column mean
  poi_type<-na_mean(poi_type)
  # summary table of interest
  summary<- list(
  "Rating"=list(
    "min"= ~ min(Rating,na.rm = TRUE),
    "max"= ~ max(Rating,na.rm = TRUE),
    "mean"= ~ mean(Rating,na.rm = TRUE)),

  "Total_Review"=list(
    "min"= ~ min(Total_Review,na.rm = TRUE),
    "max"= ~ max(Total_Review,na.rm = TRUE),
    "mean"= ~ mean(Total_Review,na.rm = TRUE)),

  "Total_hours"=list(
    "min"= ~ min(Total_hours,na.rm = TRUE),
    "max"= ~ max(Total_hours,na.rm = TRUE),
    "standard deviation"= ~ sd(Total_hours,na.rm = TRUE),
    "mean"= ~ mean(Total_hours,na.rm = TRUE)),

  "Open_days"=list(
    "min"= ~ min(open_days,na.rm = TRUE),
    "max"= ~ max(open_days,na.rm = TRUE),
     "standard deviation"= ~ sd(open_days,na.rm = TRUE),
    "mean"= ~ mean(open_days,na.rm = TRUE))
  )


  whole<-summary_table(poi_type,summary)

  # compute correlation matrix for numeric features
  cor_matrix<-cor(poi_type[-1], use = "complete.obs")
  corrplot<-corrplot(cor_matrix, method =  "number")

  # normized the features
  normalize <- function(x) {
return ((x - min(x)) / (max(x) - min(x)))
}
  Norm_poi<-poi_type%>%mutate_if(is.numeric, normalize)
```

```
  # Navie weighted index
Norm_poi%>%mutate(Index=(open_days+Total_hours+Rating+Total_Review)/4)->Norm_poi

  # plot density of each feature

  p1<-plot(density(unlist(Norm_poi[,2])), main = "Normalized Open Days Distribution")
  p1_1<-plot(hist(unlist(poi_type[,2])), main = "Unnormalized Open Days Distribution")
  p2<-plot(density(unlist(Norm_poi[,3])), main = 'Normalized Operation Hours Distribution')
  p3<-plot(density(unlist(Norm_poi[,4])), main = 'Nnormalized Rating Distribution')
  p4<-plot(density(unlist(Norm_poi[,5])), main = 'Nnormalized Total Review Distribution')
  p5<-plot(density(unlist(Norm_poi[,6])), main = 'Nnormalized Index Distribution')


  #
 result<-list(missing_percentage=missing_percentage,corrplot=corrplot,summary=whole,data=Norm_poi,plot_
 return(result)

}
```

```
# we are only interested in museum
poi_int<-c("museum","library or archives","gallery","theatre/performance and concert hall")


df<-weight_index(merged_data,Amenity="museum")
```

```
df<-df$data[FALSE,]

for(name in poi_int){
  tem<-invisible(weight_index(merged_data,Amenity=name))
  df<-rbind(invisible(tem$data),df)
}
```

```
head(df)
```

Select the point of interest

```
## # A tibble: 6 x 6
##   poi_name                       open_days Total_hours Rating Total_Review Index
##   <chr>                              <dbl>       <dbl>  <dbl>        <dbl> <dbl>
## 1 Carnegie Centre Theatre                1       0.274  0.471      0.142   0.472
## 2 Cineplex Odeon International ~         1       0.568  0.176      1       0.686
## 3 Havana Theatre                         1       0.737  0.529      0.419   0.671
## 4 Music Box Music And Theatre A~         1       0.389  0.882      0.00504 0.569
## 5 Orpheum Theatre                        1       0.716  0.824      0.673   0.803
## 6 Rickshaw Theatre                       1       0.421  0.647      0.210   0.569
```

Export csv poi_index csv file contains geo information (lat and lon) as well the accessibility index. To be noticed, it only contians 4 amenity type ("museum","library or archives","gallery","theatre/performance and concert hall")

```
poi_index<-left_join(df,van_poi,by=c("poi_name"="name"))
write.csv(poi_index,'/Users/yuxuancui/Desktop/MDS/data599/Amenity_index/poi_index.csv',row.names = FALSE
```