RTM Software

Steps

Examples

Preparing RTM data using R

Pilar Delpino Marimon

March 2022

A note on crime data:

Not all cities name or keep track of crime data in similar ways.

Not all cities call crime data in the same way.

Risk factors are case specific, what works for one city at one time doesn’t work for another.

RTM Software

Important things to know about RMT software:

- If the model doesn’t find risk factors relevant (correlative), the analysis will omit them.
- Use NAD 1983 State Plane projection, otherwise the software won’t identify the unit of analysis. For information on coordinate systems click here (https://epsg.io/)

Steps

1. Read in boundary shapefile
2. Clean up crime data (historic complaint datasets). Select year and type of crime.
3. Prepare data for temporal heat map
4. Read in / process built environment data
5. Make sure everything is in the right projection
6. Clip features within boundary
7. Export to shapefile and review in Qgis
8. Get features from existing vector data or Google Earth Pro. Instructions here (https://www.riskterrainmodeling.com/blog/risk-factors-for-those-in-need-of-risk-factors)
9. *Optional* Open R, import klm to create shp. Or leave as is, RTM software reads in klm/kmz files
10. Make sure everything is in the right projection!!
11. Load into RTM software.

Examples

RTM for NYC, Brooklyn 2007

For specifics on how to create images, deal with the data in a more granular way, see `scripts` within the USF package. ***#rename folder***

Load libraries

```
library(USF)
library(tidyverse)
library(readr)
library(lubridate)
library(writexl)
library(chron)
library(sp)
library(sf)
library(pivottabler)
library(here)
library(ggplot2)
library(gridExtra)
```

1. Boundary

Load boundary shapefile (Source (https://data.cityofnewyork.us/City-Government/Borough-Boundaries/tqmj-j8zm))

```
brooklyn <- st_read(here("boroughboundary.shp")) %>%
  filter(boro_name == "Brooklyn") %>% # filter for Brooklyn
  # st_transform(., crs = 2260) # keep original projection _need to double check error_
```

2. Crime data

Load historic crime data (NYPD complaints) (Source (https://data.cityofnewyork.us/Public-Safety/NYPD-Complaint-Data-Historic/qgea-i56i))

```
nychcrime <- read.csv("~/Clark/RA-ing/SummerInstitute/GIS/nyc/NYPD_Complaint_Data_Histori
c.csv") %>%
  filter(!Latitude %in% NA | !Longitude %in% NA) %>% # remove NA values
  as.data.frame(.) %>%
  st_as_sf(., coords = c("Longitude", "Latitude"), crs = 9001) %>% # to match brooklyn
  rename(., c( "Full_Date" = "CMPLNT_FR_DT", "Time" ="CMPLNT_FR_TM")) # rename columns

st_crs(nychcrime) <- st_crs(brooklyn) # make sure both are in the same projection
```

Extract data within Brooklyn for year = 2007. In this NYPD complaints dataset the year is not an independent field but is found within the `Full_Date` variable.

```
nychcrime$Full_Date %>% head()

# [1] "12/31/2019" "12/29/2019" "12/15/2019" "12/28/2019" "09/05/2008" "12/27/2019"
```

To filter by year we need to create a new column and populate it with year information.

```
bk07crime <- nychcrime %>% filter(BORO_NM == "BROOKLYN") %>% # extract Brooklyn data
  separate(Full_Date, into = c("Month", "Day", "Year"), # separate field Full_Date into new columns
           sep = "/", remove = FALSE) %>%
  filter(Year == "2007") # extract 2007 data
```

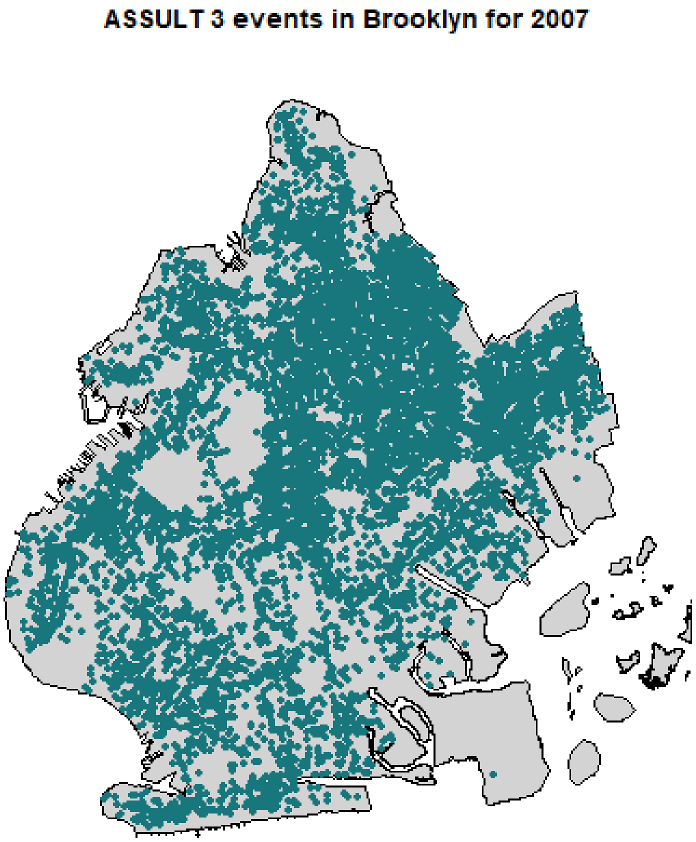
We'll select "ASSAULT 3" as the crime to be tested, but first make sure all points fall within Brooklyn.

```
bk07assault <- bk07crime %>% filter(PD_DESC == "ASSAULT 3") # select crime

bk07assault_clip <- bk07assault[brooklyn, ] # select points within polygon

bk07assault_clip <- bk07assault_clip[-c(3:4, 6:13, 14:18, 21:31,34:36)] # drop unwanted columns

# > colnames(bk07assault_clip)
# [1] "CMPLNT_NUM"    "Date"          "Year"          "Time"          "PREM_TYP_DESC"
# [6] "JURIS_DESC"    "PATROL_BORO"   "STATION_NAME"  "geometry"
```



3. Temporal heatmap

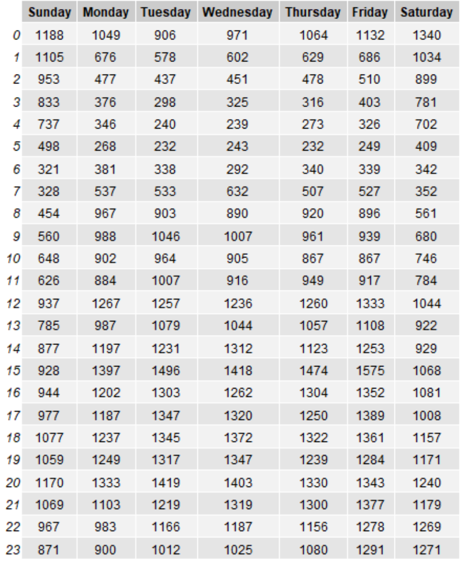
For temporal heat map for *all* crime (RTM how to here (http://www.riskterrainmodeling.com/uploads/2/6/2/0/26205659/tempheatmap_tutorial.pdf))

```
heatmap07 <- bk07crime %>%
  mutate(timestamp = paste(Full_Date, Time)) %>% # create new column with date and time
  mutate(date1 = strptime(.$timestamp, format = "%m/%d/%Y %H:%M:%S")) %>% # format to time
stamp
  mutate(DayFormat = weekdays(date1)) %>% # transform date to Weekdays
  mutate(DayFormatText = as.character(DayFormat)) %>%
  mutate(HourFormat = hour(date1)) # transform time to 0-23 hour of the day
```

Create pivot table ***small zoom, fix!***

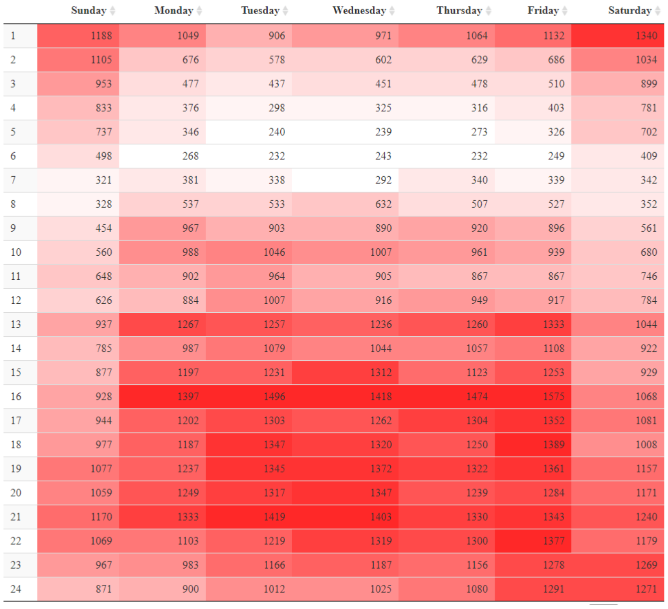
```
hmtable07 <- t(table(heatmap07$DayFormatText, heatmap07$HourFormat)) # pivot count and day

heatmap07$DayFormatText <- factor(heatmap07$DayFormatText, # Reorder factor levels by day
 of the week
  levels= c("Sunday", "Monday", "Tuesday", "Wednesday", "Thursday",
            "Friday", "Saturday"))
```



	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
0	1188	1049	906	971	1064	1132	1340
1	1105	676	578	602	629	686	1034
2	953	477	437	451	478	510	899
3	833	376	298	325	316	403	781
4	737	346	240	239	273	326	702
5	498	268	232	243	232	249	409
6	321	381	338	292	340	339	342
7	328	537	533	632	507	527	352
8	454	967	903	890	920	896	561
9	560	988	1046	1007	961	939	680
10	648	902	964	905	867	867	746
11	626	884	1007	916	949	917	784
12	937	1267	1257	1236	1260	1333	1044
13	785	987	1079	1044	1057	1108	922
14	877	1197	1231	1312	1123	1253	929
15	928	1397	1496	1418	1474	1575	1068
16	944	1202	1303	1262	1304	1352	1081
17	977	1187	1347	1320	1250	1389	1008
18	1077	1237	1345	1372	1322	1361	1157
19	1059	1249	1317	1347	1239	1284	1171
20	1170	1333	1419	1403	1330	1343	1240
21	1069	1103	1219	1319	1300	1377	1179
22	967	983	1166	1187	1156	1278	1269
23	871	900	1012	1025	1080	1291	1271

Add colour to reveal heatmap ***don’t know how to do this in regular table(). Used library DT and screencapture of html***



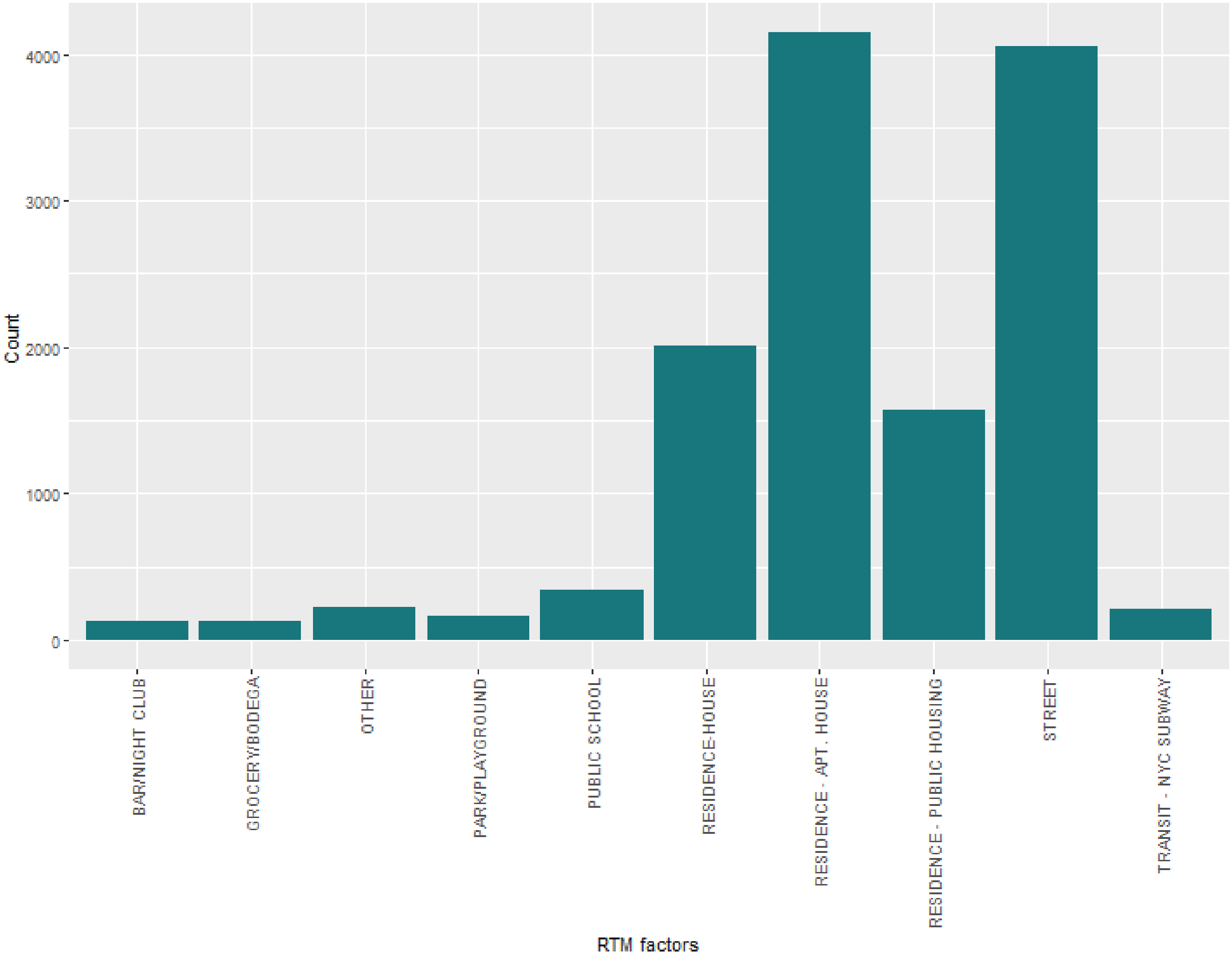
	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
1	1188	1049	906	971	1064	1132	1340
2	1105	676	578	602	629	686	1034
3	953	477	437	451	478	510	899
4	833	376	298	325	316	403	781
5	737	346	240	239	273	326	702
6	498	268	232	243	232	249	409
7	321	381	338	292	340	339	342
8	328	537	533	632	507	527	352
9	454	967	903	890	920	896	561
10	560	988	1046	1007	961	939	680
11	648	902	964	905	867	867	746
12	626	884	1007	916	949	917	784
13	937	1267	1257	1236	1260	1333	1044
14	785	987	1079	1044	1057	1108	922
15	877	1197	1231	1312	1123	1253	929
16	928	1397	1496	1418	1474	1575	1068
17	944	1202	1303	1262	1304	1352	1081
18	977	1187	1347	1320	1250	1389	1008
19	1077	1237	1345	1372	1322	1361	1157
20	1059	1249	1317	1347	1239	1284	1171
21	1170	1333	1419	1403	1330	1343	1240
22	1069	1103	1219	1319	1300	1377	1179
23	967	983	1166	1187	1156	1278	1269
24	871	900	1012	1025	1080	1291	1271

4. Risky locations

For ASSAULT 3, use `PREM_TYP_DESC` to get a sense of what locations appear as risky.

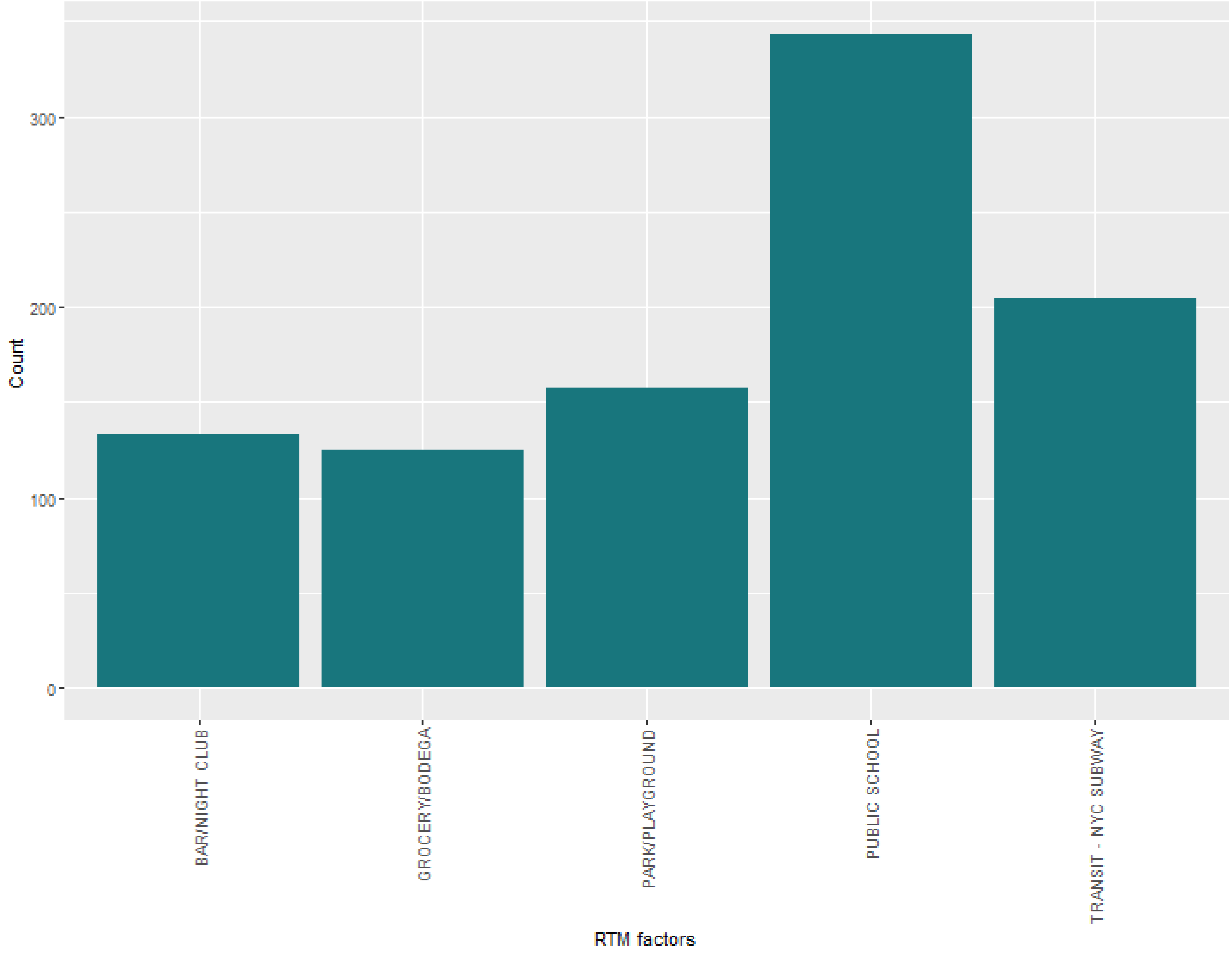
```
rtmbk07 <- bk07assault_clip %>% group_by(bk07assault_clip$PREM_TYP_DESC) %>% # group by categories of premises
  count() %>% # frequency per group
  rename(., RTM_factors = "bk07assault_clip$PREM_TYP_DESC", Count = "n") # rename
st_geometry(rtmbk07) <- NULL # drop geometry to make processing easier
rtmbk07 <- rtmbk07 %>% as_tibble() # create table
t <- rtmbk07%>% arrange(-Count) %>% slice(1:10) # arrange highest first, then select top 10 results
```

These are the top 10 locations according to the dataset. Once identified, look for datasets and other spatial data that match the results. Use Google Earth Pro, OpenData portals, Community based datahubs, etc.



To avoid skewed results, we filter out residence, street, and other to see what other built environment options could be classified as risk factors. We select the top 6 places.

```
ds07 <- rtmbk07 %>%
  filter(!RTM_factors %in% c("RESIDENCE - APT. HOUSE", "RESIDENCE-HOUSE",
                   "RESIDENCE - PUBLIC HOUSING", "STREET", "OTHER", " ")) %>%
  arrange(-Count) %>% slice(1:6)
```



5. Projections

Project everything to EPSG:2260, NAD83 / New York East (ftUS) and export as shapefile

```
brooklyn <- brooklyn %>% st_transform(., crs = 2260)
bk07assault_clip <- bk07assault_clip %>% st_transform(., crs = 2260)
```

Do the same for other data gathered.

6. Export

Export to shapefile format.

This is compatible with QGIs and ArcGis

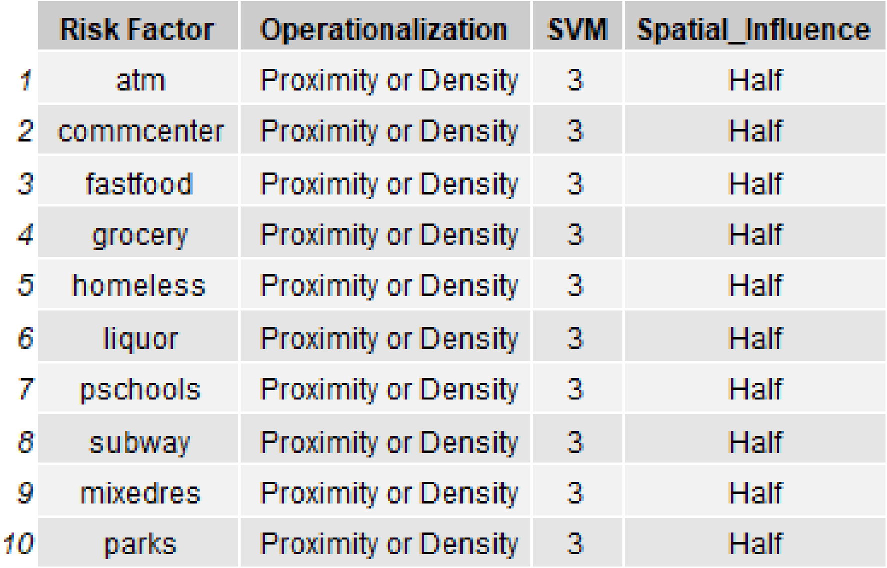
```
st_write(bk07assault_clip,
         here("bk07assault.shp"),
         driver = "ESRI Shapefile")

st_write(brooklyn,
         here("brooklyn.shp"),
         driver = "ESRI Shapefile")
```

7. RTM software

Load data to RTM software, run model, get results.

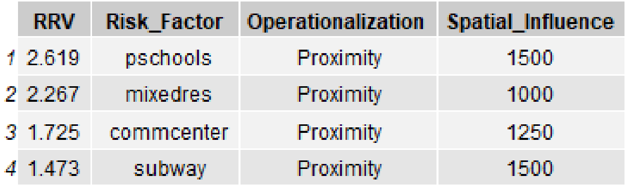
All Risk Factors Tested:



	Risk Factor	Operationalization	SVM	Spatial_Influence
1	atm	Proximity or Density	3	Half
2	commcenter	Proximity or Density	3	Half
3	fastfood	Proximity or Density	3	Half
4	grocery	Proximity or Density	3	Half
5	homeless	Proximity or Density	3	Half
6	liquor	Proximity or Density	3	Half
7	pschools	Proximity or Density	3	Half
8	subway	Proximity or Density	3	Half
9	mixedres	Proximity or Density	3	Half
10	parks	Proximity or Density	3	Half

Results
From the report:

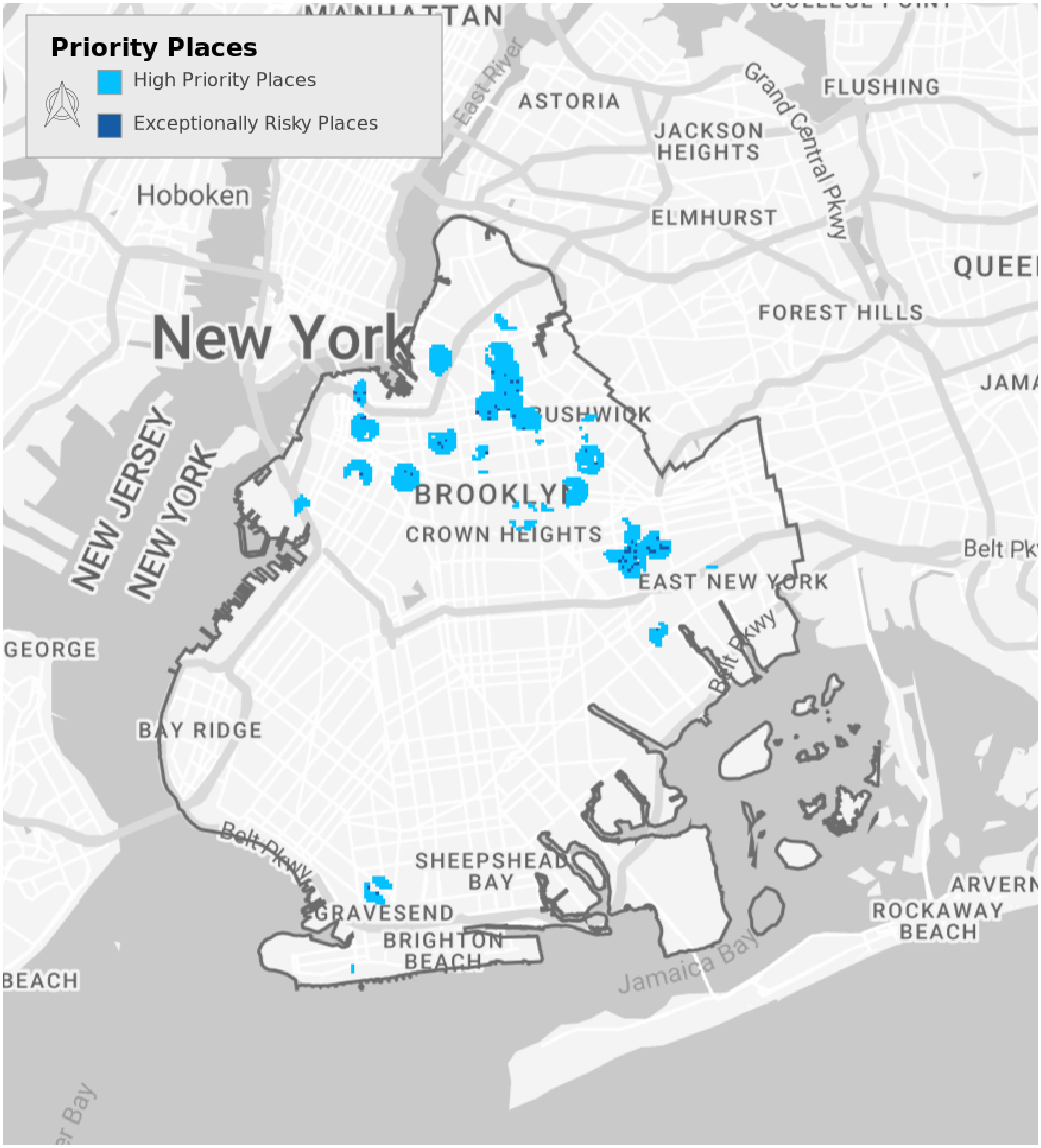
"Understand risk factors in the risk terrain model according to the operationalization, spatial influence, and relative risk value (RRV). Interpret RRVs as risk factor weights; places affected by a risk factor with a RRV of 6 are twice as risky compared to places affected by risk factor with a RRV of 3. Develop risk narratives for the study topic based on the RTM Results Table. You may choose to prioritize the risk factors for mitigation based on the RRVs and/or your risk narratives."



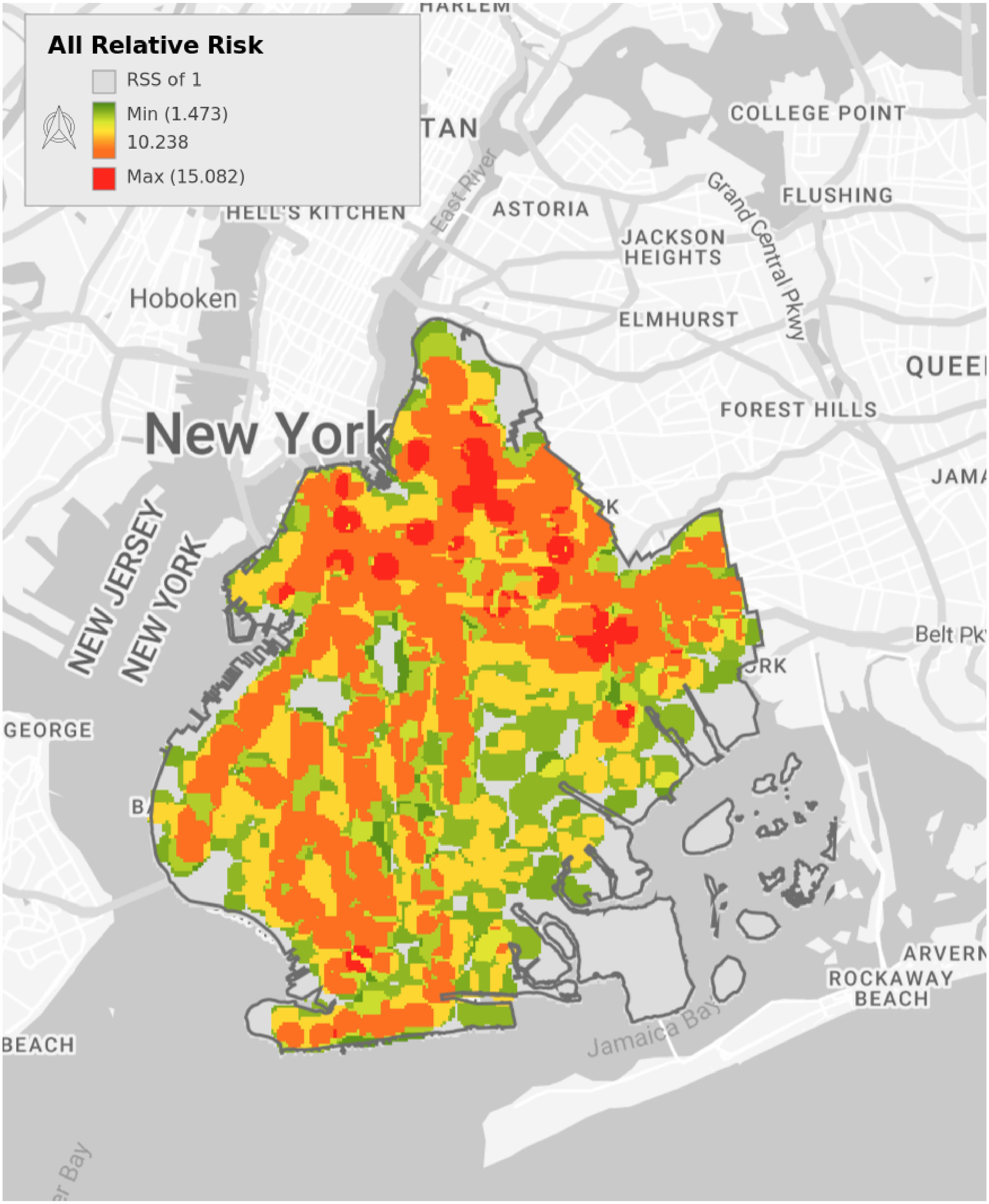
	RRV	Risk_Factor	Operationalization	Spatial_Influence
1	2.619	pschools	Proximity	1500
2	2.267	mixedres	Proximity	1000
3	1.725	commcenter	Proximity	1250
4	1.473	subway	Proximity	1500

Maps

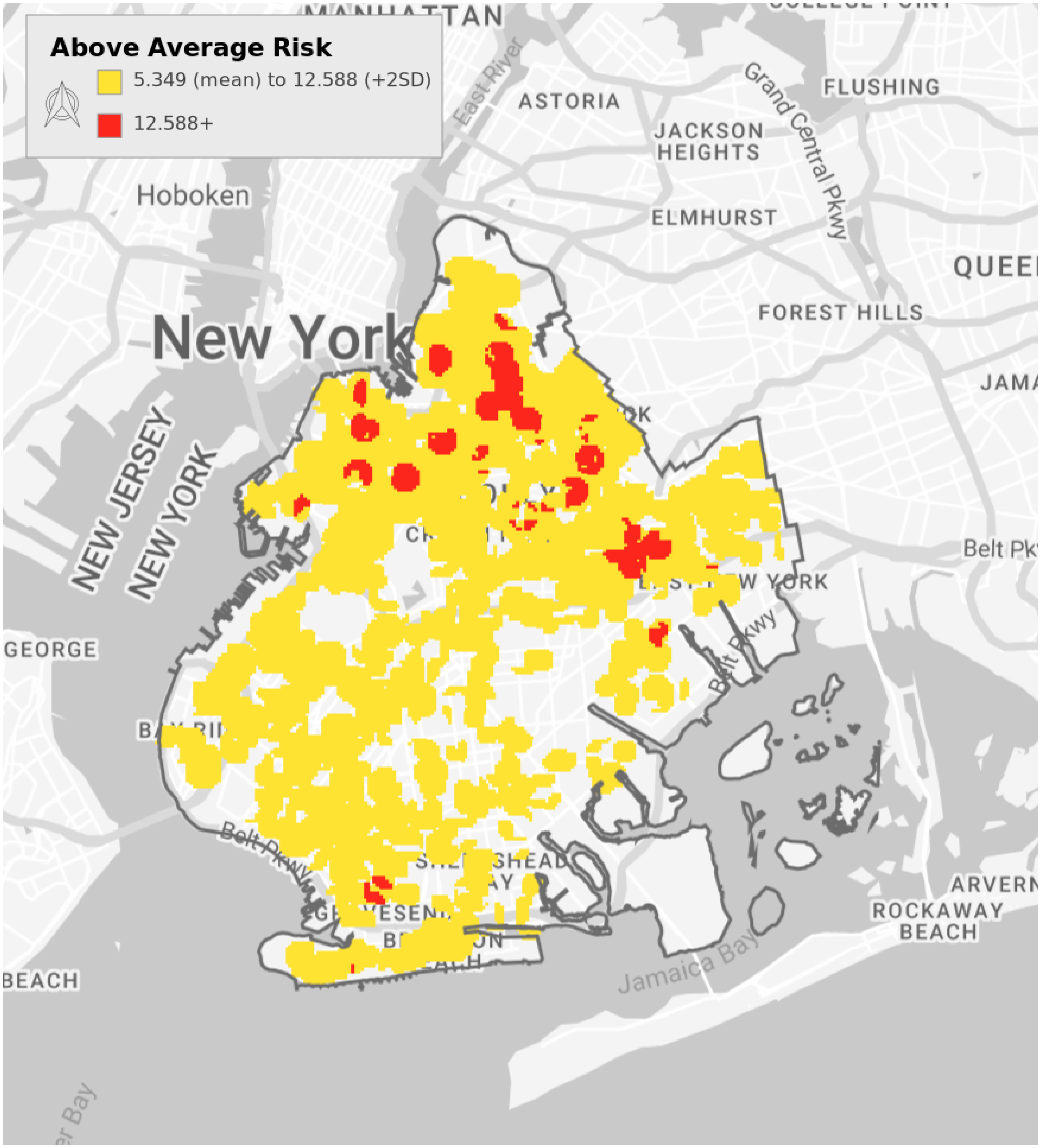
Priority places as identified by RTM software



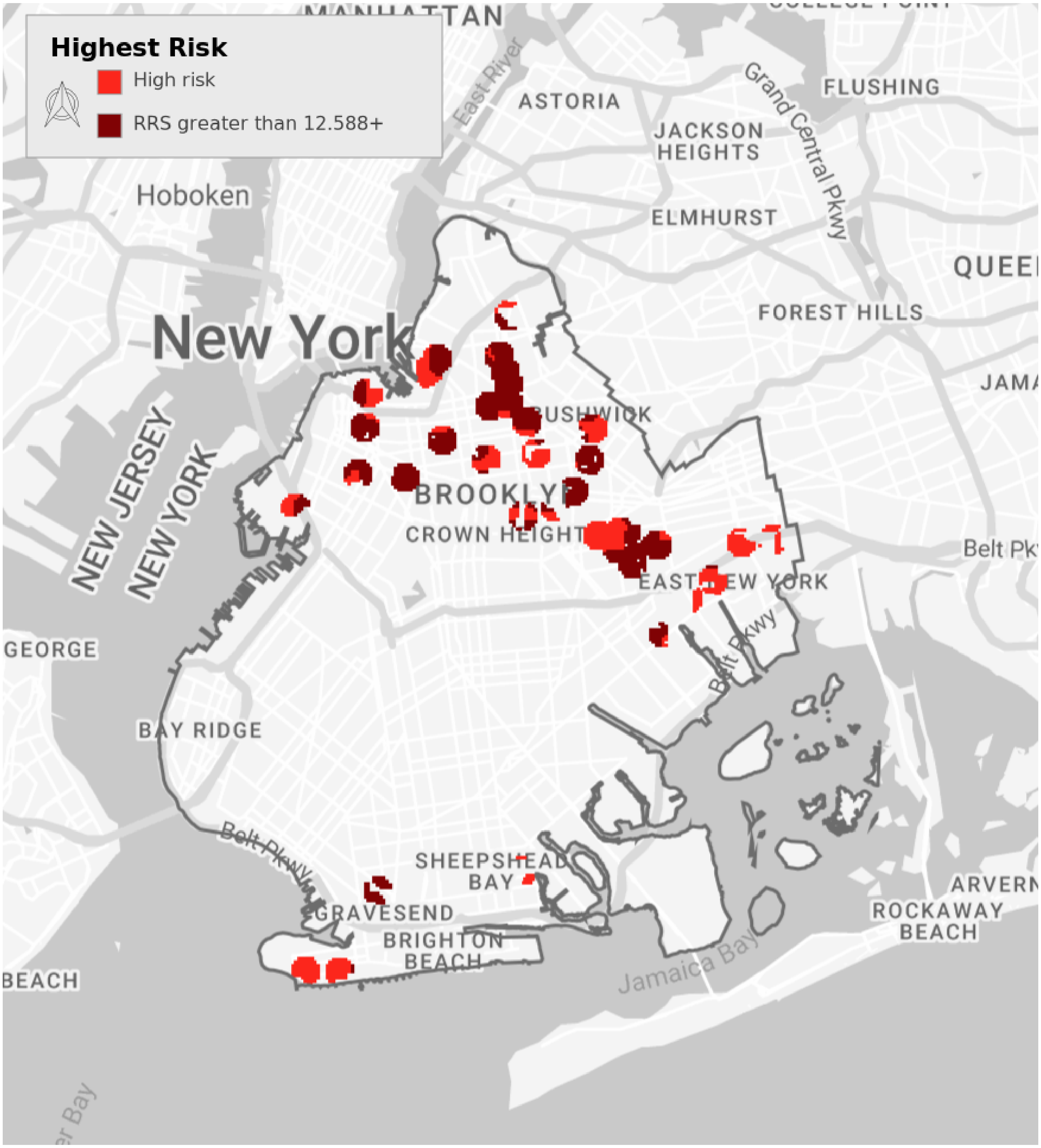
Relative risk as identified by RTM software



Above Average Risk places as identified by RTM software



Highest Risk places as identified by RTM software


Highest Risk
🟥 High risk
🟥 RRS greater than 12.588+
MANHATTAN
East River
ASTORIA
FLUSHING
JACKSON HEIGHTS
Grand Central Pkwy
ELMHURST
Hoboken
QUEE
FOREST HILLS
New York
JAMA
BUSHWICK
NEW JERSEY
NEW YORK
BROOKLYN
CROWN HEIGHTS
Belt Pky
EAST NEW YORK
Belt Pkwy
GEORGE
BAY RIDGE
Belt Pkwy
SHEEPSHEAD BAY
GRAVESEND
BRIGHTON BEACH
ARVERN
ROCKAWAY BEACH
Jamaica Bay
BEACH
r Bay