

Trabalho Final – Estatística Espacial

30/01/2022

Projeto de Conclusão da Disciplina Estatística Espacial

Grupo:

- Marina Ferreira - A57186730
- Mayara Yonemura - A58337141
- Renan Moisés Bonfá - A58342926
- Renato Hirata - A58339669

BRIEFING

Os datasets, disponibilizados pela disciplina de MBA Estatística Espacial, possuem:

- Base Filadelfia: Contem informações sobre violações ocorridas na Filadélfia - EUA;
- Base Municípios Mensal: contém informações sobre os casos de lesões corporais dolosas e roubo de rua de um determinado período para respectivos municípios;
- Base Evolução Populacional Mensal: Contém informações sobre a população estimada para respectivos municípios, no ano de 2019;

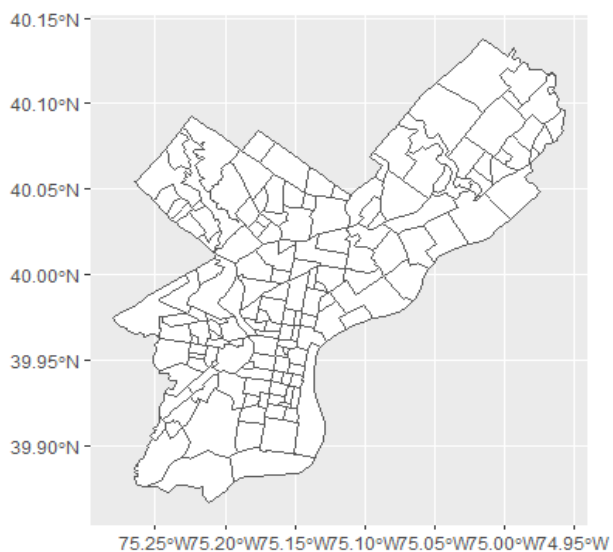
1. BASE FILADÉLFIA

i. Questão 01

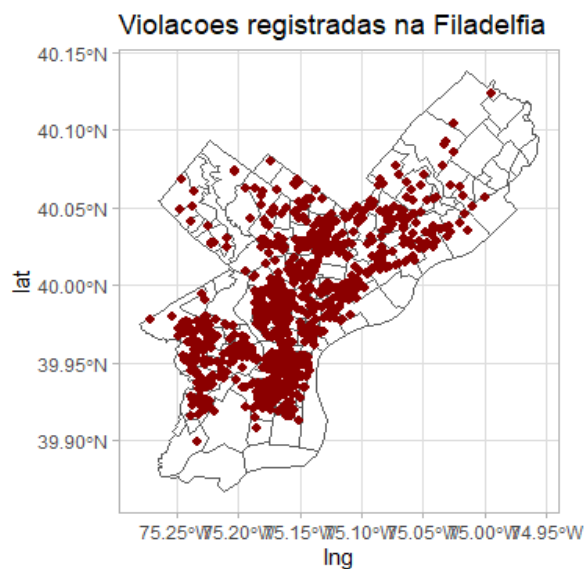
Escolha três das violações cometidas. Inicialmente a empresa gostaria de avaliar visualmente o comportamento das diferentes violações cometidas (violation_description). Discuta possíveis diferenças observadas nestes cenários.

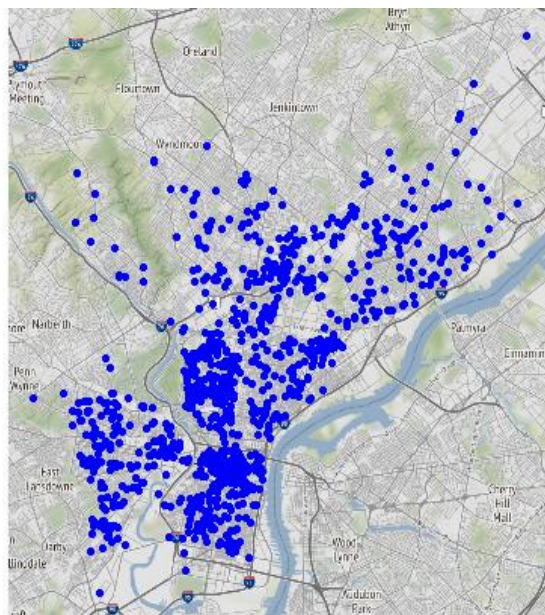
Plotagem das Violações Registradas na Filadélfia

Plot do Shape Philadelphia



Plot do mapa com os pontos da base de violações na Philadelphia:





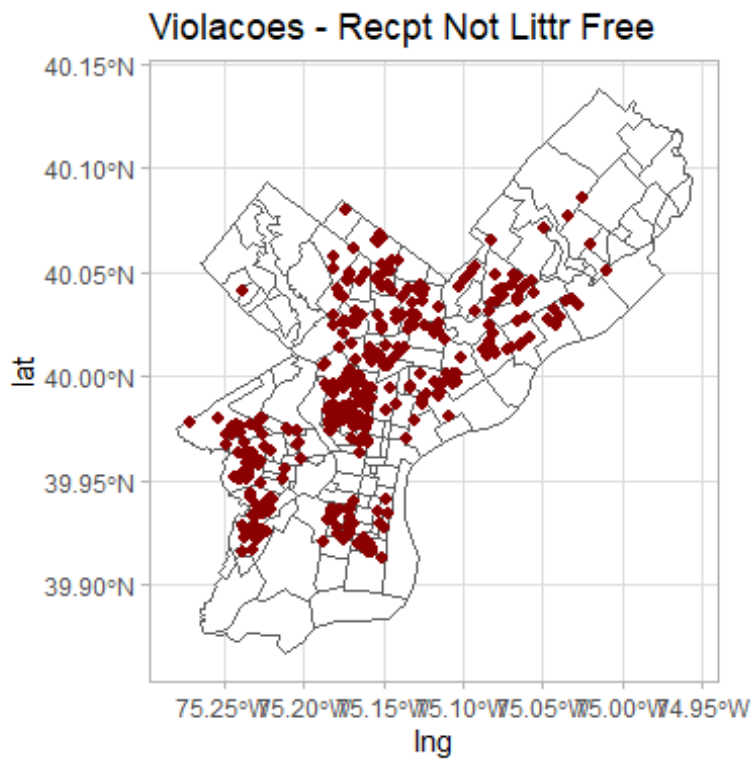
Seleção para Violações para Plots

Para selecionar as Violações, elencamos a contagem de cada violação. Como são apenas 5, escolhemos a com maior incidência (“Recept Not Littr Free”), com menor incidência (“No Permit Fixture”) e a mediana (“Unlicnsd Cstr Dumps”).

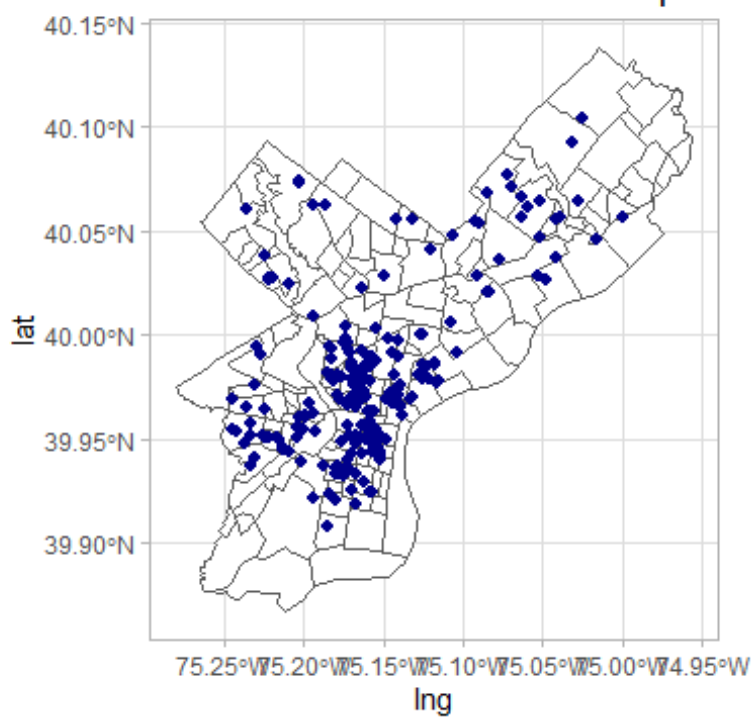
```
## # A tibble: 5 x 3
##   violation_code violation_description      n
##   <chr>          <chr>                <int>
## 1 106111K        NO PERMIT FIXTURE             138
## 2 107142        RECPY NOT LITTR FREE          477
## 3 107226D        COLLECTIONS 9PM-7AM PROHIBITED 453
## 4 10801         ABAND REFRIG (DOOR)           165
## 5 116102        UNLICNSD CSTR DUMPS           256
```

Em seguida, foram plotados os mapas com as Violações escolhidas:

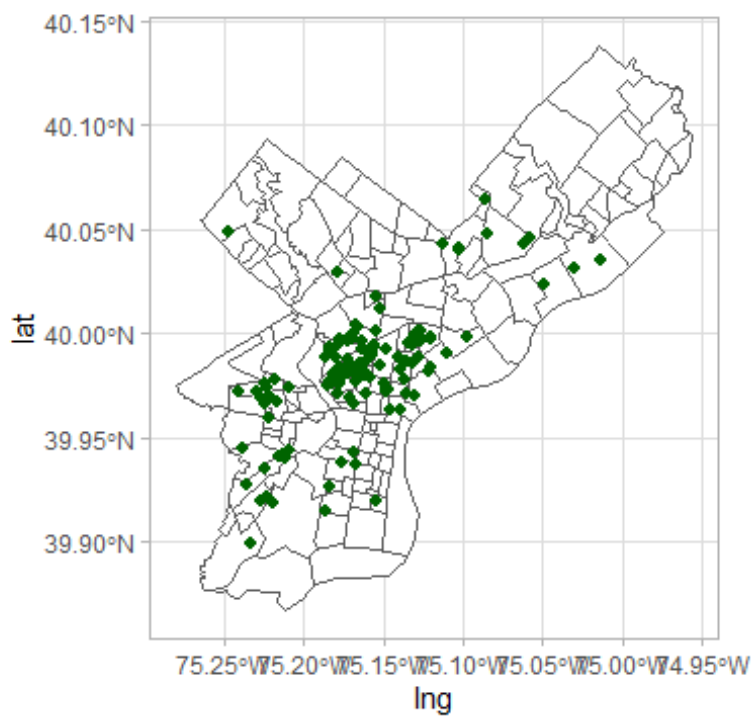
- “Recpt Not Littr Free”,
- “Unlicensd Cstr Dumps”;
- “No Permit Fixture”.



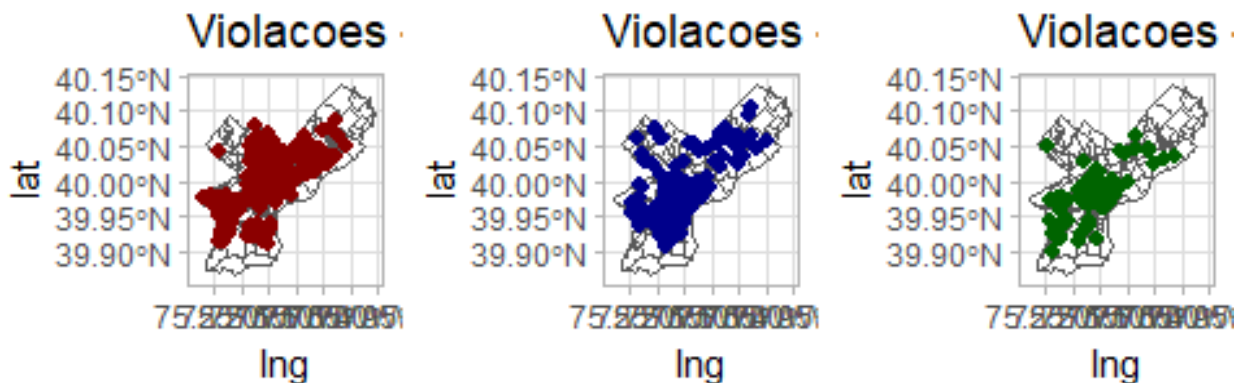
Violacoes - Unlicensd Cstr Dumps



Violacoes - No Permit Fixture



Plot das violacoes escolhidas



Comentários e Conclusões

Pode-se notar que a concentração de pontos está localizada, no caso das 3 violações selecionadas, na parte central de Filadélfia e na região Sudoeste da mesma. No caso da violação de Receipt, também existe uma alta concentração de pontos na região mais ao norte (40.05° latitude), bem como na região Sul (39.95° latitude). No demais, as 3 violações apresentam distribuições de pontos relativamente semelhantes.

ii. Questão 02 – Análise Exploratória

Investigar os efeitos de 1ª e 2ª ordem das localizações das violações cometidas. Interprete todos os resultados apresentados. Discutam as possíveis diferenças observadas entre os efeitos estimados dos 3 cenários avaliados, isto é, o comportamento da intensidade das violações parece ocorrer de formar similar para todos os seus tipos para a cidade da Philadelphia ou padrões tendem a ser diferentes?

Efeitos de 1ª Ordem

Criando o padrão de pontos a ser plotado *Plots das violações “Recpt Not Littr Free”, “Unlicensd Cstr Dumps” e “No Permit Fixture”, respectivamente para todas as análises.

```
Receptppp = ppp(x = coordenadas_recept[,1],
                y = coordenadas_recept[,2],
                window = shapeH0)

Unlicensedppp = ppp(x = coordenadas_unlicensed[,1],
                   y = coordenadas_unlicensed[,2],
                   window = shapeH0)

Fixtureppp = ppp(x = coordenadas_fixture[,1],
                 y = coordenadas_fixture[,2],
                 window = shapeH0)

Filadelfiappp = ppp(x = coordenadas_filadelfia[,1],
                    y = coordenadas_filadelfia[,2],
                    window = shapeH0)
```

Em seguida, foi realizado a estimação do Raio, os quais cada resultado são apresentados abaixo.

```
### Raio Recept
##      sigma
## 115.0609

### Raio Unlicensed
##      sigma
## 142.1341

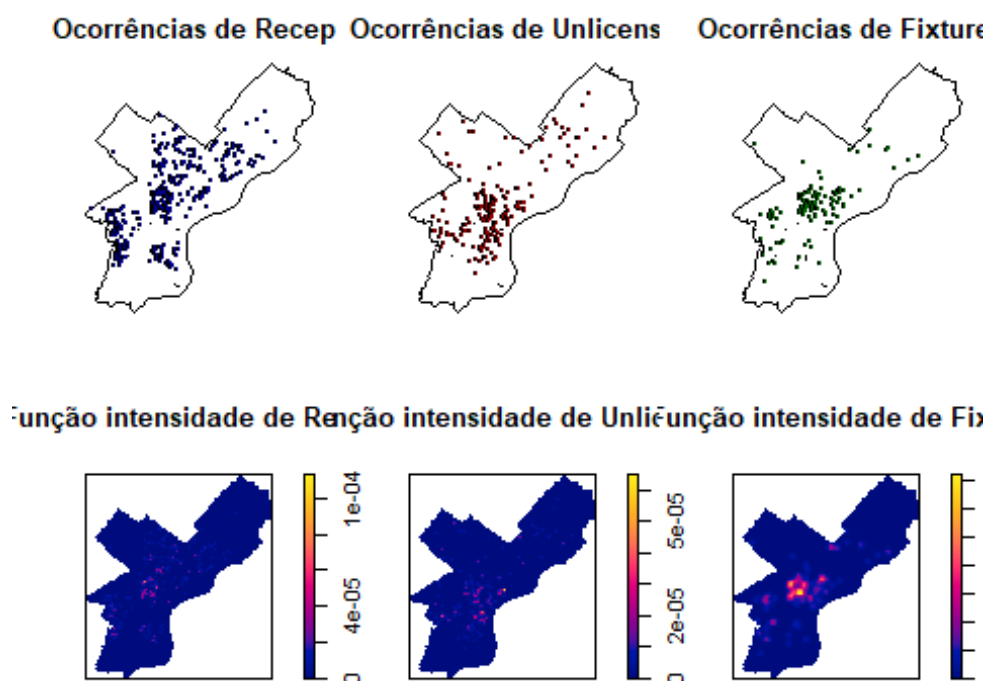
### Raio Fixture
##      sigma
## 311.3413

### Raio Filadelfia
##      sigma
##  6.76829
```

Verificação da unidade de medida utilizada e estimando a intensidade com Kernel Gaussiano.

```
## Coordinate Reference System:
##   User input: EPSG:32718
##   wkt:
## PROJCRS["WGS 84 / UTM zone 18S",
##   BASEGEOGCRS["WGS 84",
##     DATUM["World Geodetic System 1984",
##       ELLIPSOID["WGS 84",6378137,298.257223563,
##         LENGTHUNIT["metre",1]]],
##     PRIMEM["Greenwich",0,
##       ANGLEUNIT["degree",0.0174532925199433]],
##     ID["EPSG",4326]],
##   CONVERSION["UTM zone 18S",
##     METHOD["Transverse Mercator",
##       ID["EPSG",9807]],
##     PARAMETER["Latitude of natural origin",0,
##       ANGLEUNIT["degree",0.0174532925199433],
##       ID["EPSG",8801]],
##     PARAMETER["Longitude of natural origin",-75,
##       ANGLEUNIT["degree",0.0174532925199433],
##       ID["EPSG",8802]],
##     PARAMETER["Scale factor at natural origin",0.9996,
##       SCALEUNIT["unity",1],
##       ID["EPSG",8805]],
##     PARAMETER["False easting",500000,
##       LENGTHUNIT["metre",1],
##       ID["EPSG",8806]],
##     PARAMETER["False northing",10000000,
##       LENGTHUNIT["metre",1],
##       ID["EPSG",8807]]],
##   CS[Cartesian,2],
##     AXIS["(E)",east,
##       ORDER[1],
##       LENGTHUNIT["metre",1]],
##     AXIS["(N)",north,
##       ORDER[2],
##       LENGTHUNIT["metre",1]],
##   USAGE[
##     SCOPE["Engineering survey, topographic mapping."],
##     AREA["Between 78°W and 72°W, southern hemisphere between 80°S and equator, onshore and offshore. Argentina. Brazil. Chile. Colombia. Ecuador . Peru."],
##     BBOX[-80,-78,0,-72]],
##   ID["EPSG",32718]]
```


Plotando os dados e as funções intensidades estimadas pela função kernel Normal.



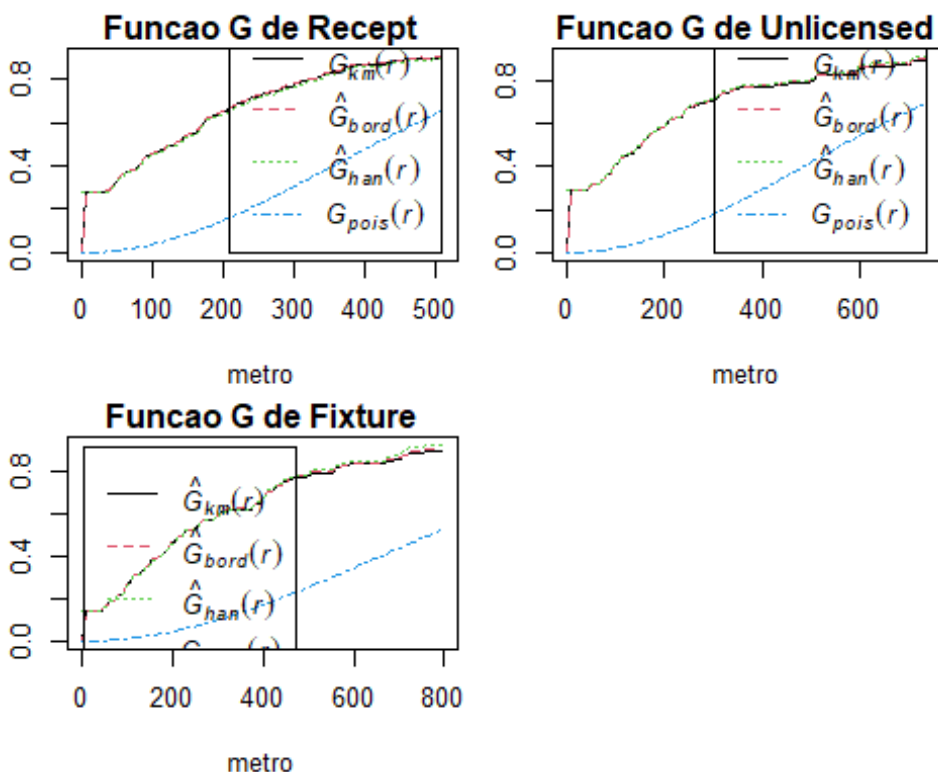
Comentários e Conclusões

Ao se analisar o efeito de 1ª ordem de cada tipo de violação analisada, em que a intensidade dos pontos espaciais, resultantes das densidades desses pontos, são levados em consideração. Ao se analisar os gráficos de intensidade obtidos, nota-se que os 3 tipos de violação ("Recept", "Unlicensed" e "Fixture") não podem ser denominados como processos pontuais estacionários, uma vez que suas propriedades dependem de sua localização espacial, isso pode ser notado através da não homogeneidade das intensidades de seus gráficos espaciais ao longo de todo seu território. De modo geral, temos um comportamento, relativamente, semelhante para o efeito de 1ª ordem de cada tipo de ocorrência, em que as regiões de alta intensidade/densidade de pontos espaciais está concentrados na região Centro-Sudoeste da Filadélfia. Entretanto, nota-se que a ocorrência "Fixture" possui uma maior intensidade na região mencionada anteriormente, quando comparado com as ocorrências "Recept" e "Unlicensed".

Efeitos de 2ª Ordem

Funções G

Plotando as funções G de cada ocorrência *Plots das violações “Recpt Not Littr Free”, “Unlicensd Cstr Dumps” e “No Permit Fixture”, respectivamente para todas as análises



Comentários e Conclusões

Ao se analisar o efeito de 2ª ordem de cada tipo de violação analisada, em que esse efeito fornece informações sobre a interação de eventos em pequena escala. Ao se analisar a função G de cada tipo de ocorrência, nota-se que os pontos das 3 ocorrências indicam uma tendência de agrupamento de pontos, uma vez que a curva G estimada se encontra acima da curva de aleatoriedade de pontos e apresenta um comportamento de crescimento contínuo ao longo do eixo X. Entretanto, para garantir que os pontos apresentem realmente uma tendência de agrupamento e sair do campo da intuição, deve-se realizar o teste de CSR, o qual é apresentado a seguir.

Teste de Clark-Evans

Realizando o teste de Clark-Evans para verificar agregação espacial.

```
##
## Clark-Evans test
## No edge correction
## Z-test
##
## data: Receptppp
## R = 0.47956, p-value < 2.2e-16
## alternative hypothesis: clustered (R < 1)

##
## Clark-Evans test
## No edge correction
## Z-test
##
## data: Unlicensedppp
## R = 0.4722, p-value < 2.2e-16
## alternative hypothesis: clustered (R < 1)

##
## Clark-Evans test
## No edge correction
## Z-test
##
## data: Fixtureppp
## R = 0.52496, p-value < 2.2e-16
## alternative hypothesis: clustered (R < 1)
```

Comentários e Conclusões

Ao se realizar o teste de Clark-Evans em conjunto com a adoção da premissa de que o nível de significância do teste seja de 5%, nota-se que para as 3 ocorrências a hipótese nula é rejeitada, uma vez que o p-valor do teste de Clark-Evans para cada tipo de ocorrência é menor que 5%. Logo, pode-se assumir que existe uma relação de agrupamento entre os pontos de cada ocorrência. Logo, o efeito de 2a ordem para cada tipo de ocorrência resulta no apontamento de agrupamento de pontos. Por fim, pode-se concluir que nos 3 tipos de ocorrência selecionados, o comportamento da densidade de pontos ao longo da superfície geográfica e as interações entre esses pontos com seus respectivos pontos vizinhos apresentam um certo padrão, uma vez que os pontos nas três ocorrências selecionada encontram-se com elevada intensidade na região Centro-Sudoeste da Filadélfia, não

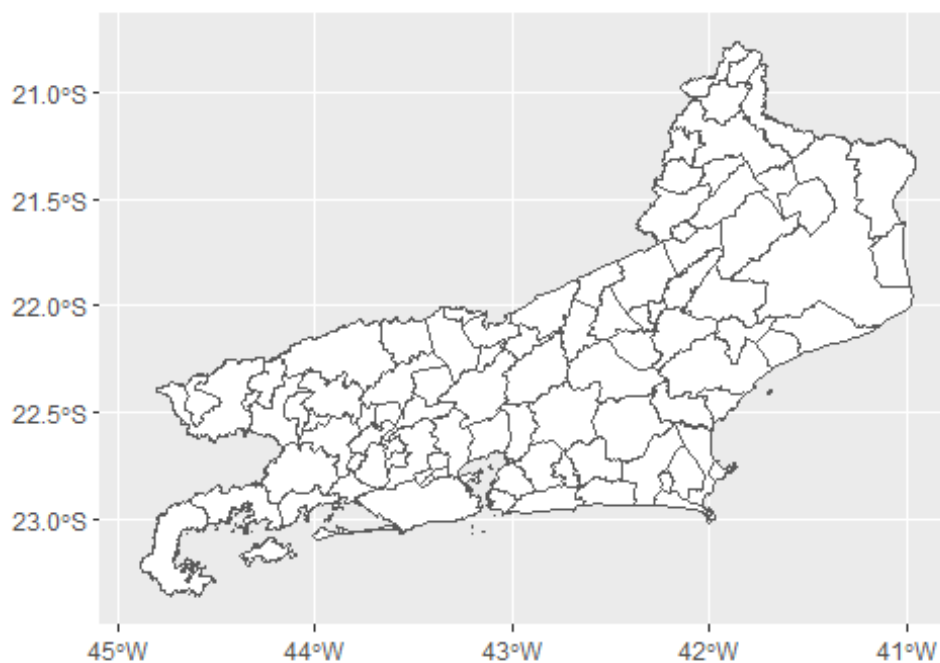
apresentando características de processos estacionários ou homogêneos. Além disso, nas 3 ocorrências foram identificados padrões de agrupamento de pontos, os quais são evidenciados pela curva estimada da função G e pelo teste de Clark-Evans.

2. BASE RIO DE JANEIRO

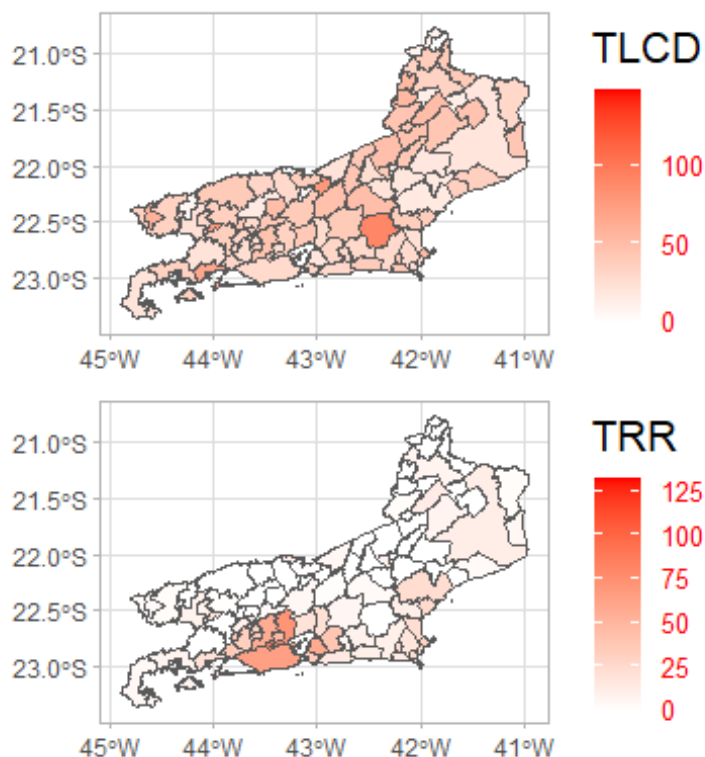
i. Questão 01

Crie uma visualização apropriada para a taxa de lesão corporal dolosa por 100.000 habitantes em 2019 (TLCD) e para a taxa de roubo de rua por 100.000 habitantes no ano de 2019 (TRR) considerando os municípios do Estado do Rio de Janeiro. Discuta a superfície de taxas estimadas.

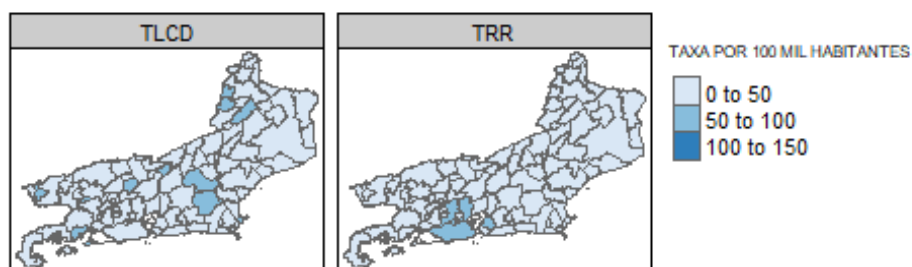
Plot do Shape Rio de Janeiro.



Mapas (GGPLOT) por Intensidade TLCD e TRR :



Mapa (TMAP) por Intensidade de Taxa na Mesma Escala



Comentários e Conclusões

Podemos observar que a Taxa de lesão corporal dolosa (TLCD) comparada com a Taxa de roubo de rua (TRR) possui maior incidência no estado do RJ-Dados da base de 2019 por 100 mil habitantes.

TLCD possui maior dispersão ao longo do estado ao passo que que TRR está mais concentrado na região da capital e baixada fluminense.

i. Questão 02

Verifique se existe dependência espacial para as duas taxas de interesse. Avalie diferentes critérios de vizinhança e diferentes pesos. Conclua usando um nível de significância de 5%.

Matriz de Proximidade Espacial

Duas principais abordagens:

- Contiguidade;
- Distância.

Contiguidade Binária

- Critério de Rook (Borda Comum);
- Critério Bishop (Vértice Comum);
- Critério Queen (Borda ou Vértice Comum).

Critério de Contiguidade:

Definição da matriz de vizinhança (W): Contiguidade (bishop, rook e queen) ou distância (k-vizinhos mais próximos).

W define quem é vizinho de quem e a intensidade (peso) dessa relação.

Criando a matriz de vizinhança W baseada no critério de contiguidade Queen e Rook

```
summary(W.rook)

## Neighbour list object:
## Number of regions: 1104
## Number of nonzero links: 76080
## Percentage nonzero weights: 6.242124
## Average number of links: 68.91304
## Link number distribution:
##
```

```
## 23 35 47 59 71 83 95 107 119 131
## 36 84 192 216 192 156 96 96 12 24
## 36 least connected regions:
## 49 50 51 52 53 54 55 56 57 58 59 60 397 398 399 400 401 402 403 404 405 40
6 407 408 625 626 627 628 629 630 631 632 633 634 635 636 with 23 links
## 24 most connected regions:
## 193 194 195 196 197 198 199 200 201 202 203 204 649 650 651 652 653 654 65
5 656 657 658 659 660 with 131 links
```

Definição dos pesos de W

- **Queen**

Lista de vizinhança espacial com pesos

```
recWQW <- nb2listw(neighbours = W.queen,
                    style="W") #outras opcoes: B, C, S e U
```

```
recWQB <- nb2listw(neighbours = W.queen,
                    style="B") #outras opcoes: B, C, S e U
```

- **Rook**

```
recWRW <- nb2listw(neighbours = W.rook,
                    style="W") #outras opcoes: B, C, S e U
```

```
recWRB <- nb2listw(neighbours = W.rook,
                    style="B") #outras opcoes: B, C, S e U
```

```
summary(W.rook)
```

Indicadores de autocorrelação global:

- Índice I global de Moran;
- C de Gery;
- Getis e Ord's $G(d)$.

Indicadores de autocorrelação local:

- Índice I local de Moran;

Indicadores de autocorrelação global

Calculando o índice de Moran considerando uma estrutura de vizinhança Queen com padronização pelas linhas

```
##
## Moran I test under randomisation
##
## data: Rio_final$TLCD
## weights: recWQW
##
## Moran I statistic standard deviate = 9.8535, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic      Expectation      Variance
##      5.027501e-02      -9.066183e-04      2.698006e-05

##
## Moran I test under randomisation
##
## data: Rio_final$TRR
## weights: recWQW
##
## Moran I statistic standard deviate = 146.47, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic      Expectation      Variance
##      7.601207e-01      -9.066183e-04      2.699644e-05
```

Calculando o índice de Moran considerando uma estrutura de vizinhança Queen com pesos iguais

```
##
## Moran I test under randomisation
##
## data: Rio_final$TLCD
## weights: recWQB
##
## Moran I statistic standard deviate = 10.11, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
```



```
## Moran I statistic      Expectation      Variance
##      4.857424e-02      -9.066183e-04      2.395431e-05

##
## Moran I test under randomisation
##
## data: Rio_final$TRR
## weights: recWQB
##
## Moran I statistic standard deviate = 151.77, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic      Expectation      Variance
##      7.421386e-01      -9.066183e-04      2.396864e-05
```

Calculando o índice de Moran considerando uma estrutura de vizinhança Rook com padronização pelas linhas

```
##
## Moran I test under randomisation
##
## data: Rio_final$TLCD
## weights: recWRW
##
## Moran I statistic standard deviate = 9.712, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic      Expectation      Variance
##      4.970105e-02      -9.066183e-04      2.715287e-05

##
## Moran I test under randomisation
##
## data: Rio_final$TRR
## weights: recWRW
##
## Moran I statistic standard deviate = 146.5, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic      Expectation      Variance
##      7.626964e-01      -9.066183e-04      2.716935e-05
```

Calculando o índice de Moran considerando uma estrutura de vizinhança Rook com pesos iguais

```
##
## Moran I test under randomisation
##
## data: Rio_final$TLCD
## weights: recWRB
##
## Moran I statistic standard deviate = 9.76, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic      Expectation      Variance
##      0.0470571843      -0.0009066183      0.0000241506
##
## Moran I test under randomisation
##
## data: Rio_final$TRR
## weights: recWRB
##
## Moran I statistic standard deviate = 152.43, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic      Expectation      Variance
##      7.483858e-01      -9.066183e-04      2.416505e-05
```

Indicadores de autocorrelação local

Calculando o I local de Moran e apresentando as 100 primeiras linhas.

	Ii	E.Ii	Var.Ii	Z.Ii	Pr(z != E(Ii))
## 1	4.455463e-04	-1.766102e-08	3.975294e-07	0.7066842744	4.797627e-01
## 2	-8.131865e-02	-4.458338e-04	1.003074e-02	-0.8074880760	4.193853e-01
## 3	-4.856490e-02	-1.754372e-04	3.948198e-03	-0.7701075404	4.412361e-01
## 4	-5.170948e-02	-1.969032e-04	4.431194e-03	-0.7738429814	4.390237e-01
## 5	-1.023661e-01	-6.675238e-04	1.501517e-02	-0.8299459449	4.065693e-01
## 6	-1.358994e-01	-1.083663e-03	2.436559e-02	-0.8636782510	3.877647e-01
## 7	-1.320572e-01	-1.032471e-03	2.321576e-02	-0.8599273889	3.898291e-01
## 8	-6.131760e-02	-2.687328e-04	6.047245e-03	-0.7850527039	4.324227e-01
## 9	-6.786835e-02	-3.228122e-04	7.263789e-03	-0.7925287551	4.280524e-01
## 10	-1.282441e-01	-9.825177e-04	2.209363e-02	-0.8561772503	3.918997e-01
## 11	-7.453536e-02	-3.818458e-04	8.591633e-03	-0.8000071938	4.237066e-01
## 12	-2.204884e-03	-4.277893e-07	9.629047e-06	-0.7104112270	4.774492e-01
## 13	4.202911e-02	-9.891505e-05	2.226245e-03	0.8928626472	3.719307e-01
## 14	-2.448596e-01	-1.979543e-03	4.446905e-02	-1.1517624762	2.494187e-01
## 15	4.202911e-02	-9.891505e-05	2.226245e-03	0.8928626472	3.719307e-01

## 16	1.333134e-01	-1.381195e-03	3.104619e-02	0.7644448721	4.446022e-01
## 17	2.050399e-01	-6.082857e-03	1.360853e-01	0.5723074714	5.671137e-01
## 18	-1.665306e-02	-1.342350e-05	3.021437e-04	-0.9572757730	3.384281e-01
## 19	9.201795e-02	-5.548389e-04	1.248186e-02	0.8285975276	4.073322e-01
## 20	-1.600974e-01	-9.537373e-04	2.144707e-02	-1.0866894010	2.771741e-01
## 21	1.898244e-01	-4.145204e-03	9.291707e-02	0.6363352841	5.245579e-01
## 22	-1.600974e-01	-9.537373e-04	2.144707e-02	-1.0866894010	2.771741e-01
## 23	-3.383151e-01	-3.375780e-03	7.572847e-02	-1.2171289073	2.235552e-01
## 24	4.202911e-02	-9.891505e-05	2.226245e-03	0.8928626472	3.719307e-01
## 25	-2.931831e-02	-5.252383e-05	4.897750e-04	-1.3223973124	1.860359e-01
## 26	-3.935069e-02	-9.346362e-05	8.714953e-04	-1.3298023760	1.835834e-01
## 27	-9.132417e-02	-4.739019e-04	4.417185e-03	-1.3669524058	1.716402e-01
## 28	-2.434742e-02	-3.644742e-05	3.398709e-04	-1.3186977914	1.872702e-01
## 29	5.417825e-05	-1.861872e-10	1.736252e-09	1.3002296416	1.935223e-01
## 30	5.108068e-02	-1.775673e-04	1.655575e-03	1.2597644330	2.077544e-01
## 31	1.903198e-02	-2.356420e-05	2.197381e-04	1.2854894928	1.986213e-01
## 32	4.207488e-02	-1.189215e-04	1.108847e-03	1.2671055741	2.051175e-01
## 33	5.108068e-02	-1.775673e-04	1.655575e-03	1.2597644330	2.077544e-01
## 34	-1.940671e-02	-2.329999e-05	2.172745e-04	-1.3150002558	1.885098e-01
## 35	5.553830e-02	-2.112837e-04	1.969868e-03	1.2560964613	2.090810e-01
## 36	4.207488e-02	-1.189215e-04	1.108847e-03	1.2671055741	2.051175e-01
## 37	-4.449424e-02	-1.069279e-03	2.404252e-02	-0.2800589111	7.794323e-01
## 38	2.520294e-03	-2.918822e-05	6.569744e-04	0.0994667115	9.207677e-01
## 39	-2.125142e-02	-3.999608e-04	8.999061e-03	-0.2198051414	8.260229e-01
## 40	-2.125142e-02	-3.999608e-04	8.999061e-03	-0.2198051414	8.260229e-01
## 41	-4.449424e-02	-1.069279e-03	2.404252e-02	-0.2800589111	7.794323e-01
## 42	-1.136938e-01	-3.375780e-03	7.572847e-02	-0.4008822782	6.885068e-01
## 43	-3.993222e-03	-9.489014e-04	2.133842e-02	-0.0208405507	9.833728e-01
## 44	3.049190e-03	-3.277341e-04	7.374504e-03	0.0393237440	9.686323e-01
## 45	-7.530836e-02	-2.061219e-03	4.630006e-02	-0.3404081095	7.335492e-01
## 46	-7.530836e-02	-2.061219e-03	4.630006e-02	-0.3404081095	7.335492e-01
## 47	-7.530836e-02	-2.061219e-03	4.630006e-02	-0.3404081095	7.335492e-01
## 48	-7.530836e-02	-2.061219e-03	4.630006e-02	-0.3404081095	7.335492e-01
## 49	5.474224e-01	-1.266224e-02	5.881117e-01	0.7303377387	4.651838e-01
## 50	-6.291557e-02	-3.558242e-05	1.673800e-03	-1.5369533111	1.243047e-01
## 51	2.789078e-01	-1.020947e-03	4.797808e-02	1.2779859298	2.012544e-01
## 52	5.237555e-01	-7.070896e-03	3.302753e-01	0.9236649491	3.556608e-01
## 53	1.668838e-01	-3.144681e-04	1.478847e-02	1.3748968010	1.691634e-01
## 54	3.428173e-01	-1.716577e-03	8.061216e-02	1.2134772012	2.249474e-01
## 55	3.546809e-02	-1.234987e-05	5.809524e-04	1.4720369111	1.410109e-01
## 56	3.546809e-02	-1.234987e-05	5.809524e-04	1.4720369111	1.410109e-01
## 57	2.063798e-01	-5.050320e-04	2.374559e-02	1.3425705942	1.794111e-01
## 58	8.142796e-02	-6.812700e-05	3.204595e-03	1.4396277883	1.499727e-01
## 59	3.981082e-01	-2.591923e-03	1.216126e-01	1.1490271562	2.505448e-01
## 60	2.437211e-01	-7.405249e-04	3.480978e-02	1.3102676909	1.901053e-01
## 61	-2.114801e-01	-9.831996e-04	1.741207e-02	-1.5952192769	1.106632e-01
## 62	-1.734222e-01	-6.808869e-04	1.206188e-02	-1.5728548584	1.157524e-01
## 63	-1.004125e-01	-2.424062e-04	4.296101e-03	-1.5282733908	1.264447e-01
## 64	6.399290e-02	-1.153825e-04	2.045153e-03	1.4175920555	1.563099e-01
## 65	1.225076e-01	-4.522442e-04	8.013320e-03	1.3735897536	1.695691e-01

```
## 66 -2.907018e-01 -1.753970e-03 3.103814e-02 -1.6401050043 1.009833e-01
## 67 1.225076e-01 -4.522442e-04 8.013320e-03 1.3735897536 1.695691e-01
## 68 -6.546075e-02 -1.062383e-04 1.883089e-03 -1.5060525749 1.320537e-01
## 69 1.225076e-01 -4.522442e-04 8.013320e-03 1.3735897536 1.695691e-01
## 70 -1.363997e-01 -4.339558e-04 7.689408e-03 -1.5505401930 1.210119e-01
## 71 -1.734222e-01 -6.808869e-04 1.206188e-02 -1.5728548584 1.157524e-01
## 72 -6.546075e-02 -1.062383e-04 1.883089e-03 -1.5060525749 1.320537e-01
## 73 1.052922e-01 -1.033912e-04 1.098912e-03 3.1793708175 1.475951e-03
## 74 2.491988e-01 -5.981513e-04 6.354416e-03 3.1336413006 1.726518e-03
## 75 5.954198e-01 -3.717760e-03 3.937206e-02 3.0194823514 2.532070e-03
## 76 -9.395827e-02 -7.892262e-05 8.388641e-04 -3.2413376828 1.189702e-03
## 77 -9.912901e-03 -8.939958e-07 9.502973e-06 -3.2153764938 1.302735e-03
## 78 7.266915e-02 -4.889944e-05 5.197650e-04 3.1896183265 1.424608e-03
## 79 2.017567e-01 -3.878590e-04 4.121258e-03 3.1488155017 1.639337e-03
## 80 5.626981e-02 -2.921562e-05 3.105467e-04 3.1947539523 1.399501e-03
## 81 -1.279862e-01 -1.454236e-04 1.545597e-03 -3.2517801107 1.146847e-03
## 82 5.626981e-02 -2.921562e-05 3.105467e-04 3.1947539523 1.399501e-03
## 83 -9.912901e-03 -8.939958e-07 9.502973e-06 -3.2153764938 1.302735e-03
## 84 6.720576e-03 -4.123590e-07 4.383285e-06 3.2102087752 1.326386e-03
## 85 -3.865763e-03 -3.747137e-07 3.494318e-06 -2.0678159074 3.865734e-02
## 86 7.104267e-02 -1.316976e-04 1.227958e-03 2.0311022891 4.224462e-02
## 87 1.973147e-01 -1.093085e-03 1.018221e-02 1.9662452895 4.927028e-02
## 88 1.124949e-01 -3.379508e-04 3.150430e-03 2.0102528334 4.440444e-02
## 89 1.073672e-01 -3.069505e-04 2.861529e-03 2.0128539956 4.412999e-02
## 90 1.429385e-01 -5.552645e-04 5.175136e-03 1.9946754212 4.607829e-02
## 91 2.214164e-01 -1.397191e-03 1.301103e-02 1.9533758192 5.077508e-02
## 92 1.479587e-01 -5.967021e-04 5.561109e-03 1.9920840599 4.636184e-02
## 93 1.176073e-01 -3.704422e-04 3.453207e-03 2.0076530883 4.468017e-02
## 94 9.706559e-02 -2.494231e-04 2.325366e-03 2.0180605941 4.358495e-02
## 95 6.579197e-02 -1.126256e-04 1.050150e-03 2.0337149937 4.198034e-02
## 96 1.176073e-01 -3.704422e-04 3.453207e-03 2.0076530883 4.468017e-02
## 97 9.450330e-02 -1.074772e-04 1.905046e-03 2.1676437108 3.018581e-02
## 98 1.467334e-01 -2.655242e-04 4.705706e-03 2.1428988719 3.212122e-02
## 99 1.736952e-01 -3.769347e-04 6.679413e-03 2.1299037785 3.317956e-02
## 100 1.763713e-01 -3.891464e-04 6.895726e-03 2.1286053299 3.328693e-02
```

```
## attr(,"call")
## localmoran(x = Rio_final$TLCD, listw = recWQW, zero.policy = TRUE,
##   na.action = na.exclude)
## attr(,"class")
## [1] "localmoran" "matrix"      "array"
## attr(,"quadr")
##      mean      median      pysal
## 1 High-High High-High High-High
## 2 Low-High Low-High Low-High
## 3 Low-High Low-High Low-High
## 4 Low-High Low-High Low-High
## 5 Low-High Low-High Low-High
## 6 Low-High Low-High Low-High
## 7 Low-High Low-High Low-High
```

## 8	Low-High	Low-High	Low-High
## 9	Low-High	Low-High	Low-High
## 10	Low-High	Low-High	Low-High
## 11	Low-High	Low-High	Low-High
## 12	Low-High	High-High	Low-High
## 13	High-High	High-High	High-High
## 14	Low-High	Low-High	Low-High
## 15	High-High	High-High	High-High
## 16	High-High	High-High	High-High
## 17	High-High	High-High	High-High
## 18	Low-High	High-High	Low-High
## 19	High-High	High-High	High-High
## 20	Low-High	Low-High	Low-High
## 21	High-High	High-High	High-High
## 22	Low-High	Low-High	Low-High
## 23	Low-High	Low-High	Low-High
## 24	High-High	High-High	High-High
## 25	High-Low	High-Low	High-Low
## 26	High-Low	High-Low	High-Low
## 27	High-Low	High-Low	High-Low
## 28	High-Low	High-Low	High-Low
## 29	Low-Low	High-Low	Low-Low
## 30	Low-Low	Low-Low	Low-Low
## 31	Low-Low	Low-Low	Low-Low
## 32	Low-Low	Low-Low	Low-Low
## 33	Low-Low	Low-Low	Low-Low
## 34	High-Low	High-Low	High-Low
## 35	Low-Low	Low-Low	Low-Low
## 36	Low-Low	Low-Low	Low-Low
## 37	Low-High	Low-High	Low-High
## 38	High-High	High-High	High-High
## 39	Low-High	Low-High	Low-High
## 40	Low-High	Low-High	Low-High
## 41	Low-High	Low-High	Low-High
## 42	Low-High	Low-High	Low-High
## 43	High-High	High-High	High-Low
## 44	High-High	High-High	High-High
## 45	Low-High	Low-High	Low-High
## 46	Low-High	Low-High	Low-High
## 47	Low-High	Low-High	Low-High
## 48	Low-High	Low-High	Low-High
## 49	High-High	High-High	High-High
## 50	Low-High	Low-High	Low-High
## 51	High-High	High-High	High-High
## 52	High-High	High-High	High-High
## 53	High-High	High-High	High-High
## 54	High-High	High-High	High-High
## 55	High-High	High-High	High-High
## 56	High-High	High-High	High-High
## 57	High-High	High-High	High-High

```

## 58 High-High High-High High-High
## 59 High-High High-High High-High
## 60 High-High High-High High-High
## 61 High-Low High-Low High-Low
## 62 High-Low High-Low High-Low
## 63 High-Low High-Low High-Low
## 64 Low-Low Low-Low Low-Low
## 65 Low-Low Low-Low Low-Low
## 66 High-Low High-Low High-Low
## 67 Low-Low Low-Low Low-Low
## 68 High-Low High-Low High-Low
## 69 Low-Low Low-Low Low-Low
## 70 High-Low High-Low High-Low
## 71 High-Low High-Low High-Low
## 72 High-Low High-Low High-Low
## 73 High-High High-High High-High
## 74 High-High High-High High-High
## 75 High-High High-High High-High
## 76 Low-High Low-High Low-High
## 77 Low-High High-High Low-High
## 78 High-High High-High High-High
## 79 High-High High-High High-High
## 80 High-High High-High High-High
## 81 Low-High Low-High Low-High
## 82 High-High High-High High-High
## 83 Low-High High-High Low-High
## 84 High-High High-High High-High
## 85 High-Low High-Low High-Low
## 86 Low-Low Low-Low Low-Low
## 87 Low-Low Low-Low Low-Low
## 88 Low-Low Low-Low Low-Low
## 89 Low-Low Low-Low Low-Low
## 90 Low-Low Low-Low Low-Low
## 91 Low-Low Low-Low Low-Low
## 92 Low-Low Low-Low Low-Low
## 93 Low-Low Low-Low Low-Low
## 94 Low-Low Low-Low Low-Low
## 95 Low-Low Low-Low Low-Low
## 96 Low-Low Low-Low Low-Low
## 97 Low-Low Low-Low Low-Low
## 98 Low-Low Low-Low Low-Low
## 99 Low-Low Low-Low Low-Low
## 100 Low-Low Low-Low Low-Low

```

##		Ii	E.Ii	Var.Ii	Z.Ii	Pr(z != E(Ii))
## 1	3.087226e-02	-1.031160e-05	2.321000e-04	2.0271034	4.265184e-02	
## 2	5.275447e-02	-3.045182e-05	6.854150e-04	2.0161976	4.377932e-02	
## 3	3.087226e-02	-1.031160e-05	2.321000e-04	2.0271034	4.265184e-02	
## 4	4.184460e-02	-1.905098e-05	4.288077e-04	2.0216492	4.321261e-02	

## 5	1.273747e-01	-1.847788e-04	4.158392e-03	1.9781083	4.791649e-02
## 6	7.975589e-02	-7.059785e-05	1.588965e-03	2.0025799	4.522239e-02
## 7	1.010760e-01	-1.146913e-04	2.581274e-03	1.9916974	4.640427e-02
## 8	5.275447e-02	-3.045182e-05	6.854150e-04	2.0161976	4.377932e-02
## 9	7.438684e-02	-6.123791e-05	1.378311e-03	2.0053022	4.493075e-02
## 10	1.063669e-01	-1.273780e-04	2.866770e-03	1.9889783	4.670360e-02
## 11	9.044716e-02	-9.131383e-05	2.055183e-03	1.9971374	4.581026e-02
## 12	1.010760e-01	-1.146913e-04	2.581274e-03	1.9916974	4.640427e-02
## 13	3.816400e-01	-3.748564e-04	8.434428e-03	4.1596082	3.187940e-05
## 14	3.816400e-01	-3.748564e-04	8.434428e-03	4.1596082	3.187940e-05
## 15	3.816400e-01	-3.748564e-04	8.434428e-03	4.1596082	3.187940e-05
## 16	3.816400e-01	-3.748564e-04	8.434428e-03	4.1596082	3.187940e-05
## 17	3.816400e-01	-3.748564e-04	8.434428e-03	4.1596082	3.187940e-05
## 18	1.872167e-01	-8.810328e-05	1.982930e-03	4.2062523	2.596403e-05
## 19	3.816400e-01	-3.748564e-04	8.434428e-03	4.1596082	3.187940e-05
## 20	3.816400e-01	-3.748564e-04	8.434428e-03	4.1596082	3.187940e-05
## 21	3.816400e-01	-3.748564e-04	8.434428e-03	4.1596082	3.187940e-05
## 22	3.816400e-01	-3.748564e-04	8.434428e-03	4.1596082	3.187940e-05
## 23	3.816400e-01	-3.748564e-04	8.434428e-03	4.1596082	3.187940e-05
## 24	3.816400e-01	-3.748564e-04	8.434428e-03	4.1596082	3.187940e-05
## 25	5.384901e-03	-2.057468e-04	1.918257e-03	0.1276465	8.984288e-01
## 26	-1.377603e-03	-6.293405e-06	5.868755e-05	-0.1790039	8.579347e-01
## 27	5.174958e-03	-1.810882e-04	1.688397e-03	0.1303488	8.962905e-01
## 28	5.578624e-03	-2.319789e-04	2.162772e-03	0.1249441	9.005678e-01
## 29	3.251202e-03	-5.253451e-05	4.898745e-04	0.1492667	8.813432e-01
## 30	2.911501e-03	-4.046351e-05	3.773193e-04	0.1519696	8.792110e-01
## 31	2.911501e-03	-4.046351e-05	3.773193e-04	0.1519696	8.792110e-01
## 32	1.390497e-03	-7.914035e-06	7.380021e-05	0.1627819	8.706902e-01
## 33	3.881946e-03	-8.139686e-05	7.589888e-04	0.1438613	8.856100e-01
## 34	-8.757033e-04	-2.629879e-06	2.452436e-05	-0.1762999	8.600584e-01
## 35	4.706412e-03	-1.364913e-04	1.272649e-03	0.1357536	8.920160e-01
## 36	-4.130397e-03	-4.821282e-05	4.495776e-04	-0.1925264	8.473299e-01
## 37	3.454325e-01	-3.748564e-04	8.434428e-03	3.7653593	1.663097e-04
## 38	3.454325e-01	-3.748564e-04	8.434428e-03	3.7653593	1.663097e-04
## 39	3.454325e-01	-3.748564e-04	8.434428e-03	3.7653593	1.663097e-04
## 40	3.454325e-01	-3.748564e-04	8.434428e-03	3.7653593	1.663097e-04
## 41	3.454325e-01	-3.748564e-04	8.434428e-03	3.7653593	1.663097e-04
## 42	3.454325e-01	-3.748564e-04	8.434428e-03	3.7653593	1.663097e-04
## 43	3.454325e-01	-3.748564e-04	8.434428e-03	3.7653593	1.663097e-04
## 44	3.454325e-01	-3.748564e-04	8.434428e-03	3.7653593	1.663097e-04
## 45	3.454325e-01	-3.748564e-04	8.434428e-03	3.7653593	1.663097e-04
## 46	3.454325e-01	-3.748564e-04	8.434428e-03	3.7653593	1.663097e-04
## 47	3.454325e-01	-3.748564e-04	8.434428e-03	3.7653593	1.663097e-04
## 48	3.454325e-01	-3.748564e-04	8.434428e-03	3.7653593	1.663097e-04
## 49	8.343320e-01	-1.311963e-03	6.163605e-02	3.3659206	7.628866e-04
## 50	7.597683e-01	-1.072375e-03	5.039230e-02	3.3893129	7.006801e-04
## 51	9.077382e-01	-1.575687e-03	7.400625e-02	3.3425632	8.300846e-04
## 52	6.071686e-01	-6.656070e-04	3.129047e-02	3.4362053	5.899238e-04
## 53	4.499390e-01	-3.553821e-04	1.671185e-02	3.4832469	4.953713e-04
## 54	1.121012e+00	-2.511673e-03	1.178567e-01	3.2726917	1.065286e-03


```
## 55 1.121012e+00 -2.511673e-03 1.178567e-01 3.2726917 1.065286e-03
## 56 7.597683e-01 -1.072375e-03 5.039230e-02 3.3893129 7.006801e-04
## 57 2.054136e-01 -7.106339e-05 3.342708e-03 3.5541038 3.792695e-04
## 58 9.077382e-01 -1.575687e-03 7.400625e-02 3.3425632 8.300846e-04
## 59 6.071686e-01 -6.656070e-04 3.129047e-02 3.4362053 5.899238e-04
## 60 3.695880e-01 -2.364734e-04 1.112149e-02 3.5068258 4.534860e-04
## 61 5.374374e-02 -5.832264e-05 1.033827e-03 1.6733046 9.426734e-02
## 62 -5.707198e-02 -6.069312e-05 1.075844e-03 -1.7381461 8.218508e-02
## 63 5.374374e-02 -5.832264e-05 1.033827e-03 1.6733046 9.426734e-02
## 64 -5.707198e-02 -6.069312e-05 1.075844e-03 -1.7381461 8.218508e-02
## 65 -5.517340e-04 -5.902309e-09 1.046304e-07 -1.7056746 8.806868e-02
## 66 1.310155e-01 -3.689302e-04 6.537623e-03 1.6249275 1.041780e-01
## 67 -5.707198e-02 -6.069312e-05 1.075844e-03 -1.7381461 8.218508e-02
## 68 -1.460238e-01 -3.748564e-04 6.642599e-03 -1.7870564 7.392837e-02
## 69 -5.707198e-02 -6.069312e-05 1.075844e-03 -1.7381461 8.218508e-02
## 70 -8.616639e-02 -1.356632e-04 2.404580e-03 -1.7544217 7.935827e-02
## 71 -8.616639e-02 -1.356632e-04 2.404580e-03 -1.7544217 7.935827e-02
## 72 -2.853376e-02 -1.547402e-05 2.743041e-04 -1.7218973 8.508813e-02
## 73 2.450923e-01 -2.518639e-04 2.676585e-03 4.7422582 2.113490e-06
## 74 2.805427e-01 -3.311507e-04 3.518894e-03 4.7348709 2.191946e-06
## 75 1.916809e-01 -1.532455e-04 1.628716e-03 4.7533864 2.000376e-06
## 76 2.628332e-01 -2.901532e-04 3.083371e-03 4.7385614 2.152408e-06
## 77 2.450923e-01 -2.518639e-04 2.676585e-03 4.7422582 2.113490e-06
## 78 1.916809e-01 -1.532455e-04 1.628716e-03 4.7533864 2.000376e-06
## 79 2.450923e-01 -2.518639e-04 2.676585e-03 4.7422582 2.113490e-06
## 80 2.450923e-01 -2.518639e-04 2.676585e-03 4.7422582 2.113490e-06
## 81 1.738142e-01 -1.257891e-04 1.336942e-03 4.7571084 1.963857e-06
## 82 2.095161e-01 -1.834101e-04 1.949252e-03 4.7496707 2.037482e-06
## 83 2.628332e-01 -2.901532e-04 3.083371e-03 4.7385614 2.152408e-06
## 84 2.805427e-01 -3.311507e-04 3.518894e-03 4.7348709 2.191946e-06
## 85 2.100927e-01 -1.698526e-04 1.583658e-03 5.2836155 1.266589e-07
## 86 1.898428e-01 -1.384637e-04 1.291038e-03 5.2873909 1.240733e-07
## 87 1.797055e-01 -1.239707e-04 1.155922e-03 5.2892818 1.227976e-07
## 88 2.404056e-01 -2.229434e-04 2.078552e-03 5.2779680 1.306243e-07
## 89 2.100927e-01 -1.698526e-04 1.583658e-03 5.2836155 1.266589e-07
## 90 2.202053e-01 -1.867486e-04 1.741162e-03 5.2817309 1.279690e-07
## 91 1.492440e-01 -8.529774e-05 7.953596e-04 5.2949669 1.190380e-07
## 92 2.303096e-01 -2.044455e-04 1.906127e-03 5.2798484 1.292908e-07
## 93 2.404056e-01 -2.229434e-04 2.078552e-03 5.2779680 1.306243e-07
## 94 2.303096e-01 -2.044455e-04 1.906127e-03 5.2798484 1.292908e-07
## 95 2.202053e-01 -1.867486e-04 1.741162e-03 5.2817309 1.279690e-07
## 96 1.492440e-01 -8.529774e-05 7.953596e-04 5.2949669 1.190380e-07
## 97 3.275282e+00 -1.589146e-03 2.812607e-02 19.5391114 5.106345e-85
## 98 3.814362e+00 -2.161832e-03 3.824000e-02 19.5168175 7.900882e-85
## 99 3.126281e+00 -1.446642e-03 2.560756e-02 19.5454110 4.513422e-85
## 100 3.554322e+00 -1.874382e-03 3.316494e-02 19.5274740 6.413418e-85
```

```
## attr(,"call")
## localmoran(x = Rio_final$TRR, listw = recWQW, zero.policy = TRUE,
```



```
##      na.action = na.exclude)
## attr(,"class")
## [1] "localmoran" "matrix"      "array"
## attr(,"quadr")
##      mean      median      pysal
## 1      Low-Low High-High Low-Low
## 2      Low-Low High-High Low-Low
## 3      Low-Low High-High Low-Low
## 4      Low-Low High-High Low-Low
## 5      Low-Low High-High Low-Low
## 6      Low-Low High-High Low-Low
## 7      Low-Low High-High Low-Low
## 8      Low-Low High-High Low-Low
## 9      Low-Low High-High Low-Low
## 10     Low-Low High-High Low-Low
## 11     Low-Low High-High Low-Low
## 12     Low-Low High-High Low-Low
## 13     Low-Low Low-Low Low-Low
## 14     Low-Low Low-Low Low-Low
## 15     Low-Low Low-Low Low-Low
## 16     Low-Low Low-Low Low-Low
## 17     Low-Low Low-Low Low-Low
## 18     Low-Low High-Low Low-Low
## 19     Low-Low Low-Low Low-Low
## 20     Low-Low Low-Low Low-Low
## 21     Low-Low Low-Low Low-Low
## 22     Low-Low Low-Low Low-Low
## 23     Low-Low Low-Low Low-Low
## 24     Low-Low Low-Low Low-Low
## 25     High-High High-High High-High
## 26     Low-High High-High Low-High
## 27     High-High High-High High-High
## 28     High-High High-High High-High
## 29     High-High High-High High-High
## 30     High-High High-High High-High
## 31     High-High High-High High-High
## 32     High-High High-High High-High
## 33     High-High High-High High-High
## 34     Low-High High-High Low-High
## 35     High-High High-High High-High
## 36     Low-High High-High Low-High
## 37     Low-Low Low-Low Low-Low
## 38     Low-Low Low-Low Low-Low
## 39     Low-Low Low-Low Low-Low
## 40     Low-Low Low-Low Low-Low
## 41     Low-Low Low-Low Low-Low
## 42     Low-Low Low-Low Low-Low
## 43     Low-Low Low-Low Low-Low
## 44     Low-Low Low-Low Low-Low
## 45     Low-Low Low-Low Low-Low
```

## 46	Low-Low	Low-Low	Low-Low
## 47	Low-Low	Low-Low	Low-Low
## 48	Low-Low	Low-Low	Low-Low
## 49	High-High	High-High	High-High
## 50	High-High	High-High	High-High
## 51	High-High	High-High	High-High
## 52	High-High	High-High	High-High
## 53	High-High	High-High	High-High
## 54	High-High	High-High	High-High
## 55	High-High	High-High	High-High
## 56	High-High	High-High	High-High
## 57	High-High	High-High	High-High
## 58	High-High	High-High	High-High
## 59	High-High	High-High	High-High
## 60	High-High	High-High	High-High
## 61	High-High	High-High	High-High
## 62	Low-High	High-High	Low-High
## 63	High-High	High-High	High-High
## 64	Low-High	High-High	Low-High
## 65	Low-High	High-High	Low-High
## 66	High-High	High-High	High-High
## 67	Low-High	High-High	Low-High
## 68	Low-High	Low-High	Low-High
## 69	Low-High	High-High	Low-High
## 70	Low-High	High-High	Low-High
## 71	Low-High	High-High	Low-High
## 72	Low-High	High-High	Low-High
## 73	Low-Low	Low-Low	Low-Low
## 74	Low-Low	Low-Low	Low-Low
## 75	Low-Low	High-Low	Low-Low
## 76	Low-Low	Low-Low	Low-Low
## 77	Low-Low	Low-Low	Low-Low
## 78	Low-Low	High-Low	Low-Low
## 79	Low-Low	Low-Low	Low-Low
## 80	Low-Low	Low-Low	Low-Low
## 81	Low-Low	High-Low	Low-Low
## 82	Low-Low	High-Low	Low-Low
## 83	Low-Low	Low-Low	Low-Low
## 84	Low-Low	Low-Low	Low-Low
## 85	Low-Low	High-Low	Low-Low
## 86	Low-Low	High-Low	Low-Low
## 87	Low-Low	High-Low	Low-Low
## 88	Low-Low	Low-Low	Low-Low
## 89	Low-Low	High-Low	Low-Low
## 90	Low-Low	Low-Low	Low-Low
## 91	Low-Low	High-Low	Low-Low
## 92	Low-Low	Low-Low	Low-Low
## 93	Low-Low	Low-Low	Low-Low
## 94	Low-Low	Low-Low	Low-Low
## 95	Low-Low	Low-Low	Low-Low

## 96	Low-Low	High-Low	Low-Low
## 97	High-High	High-High	High-High
## 98	High-High	High-High	High-High
## 99	High-High	High-High	High-High
## 100	High-High	High-High	High-High

Comentários e Conclusões

Utilizando tanto o critério de contiguidade Rook como o Queen, variando também os pesos de W, entre “W” E “R”, observa-se que os Indicadores de autocorrelação global (Índices de Moran) para TLCD e TRR são valores muito próximos de zero, e por isso requerem a análise de inferência. Como ambos os p-value para TLCD E TRR são também muito próximos de zero, rejeita-se a H_0 . Ou seja, existe dependência espacial, porém fraca.

ANEXO - Códigos

```
tbphila <- read_csv2('Base_Filadelfia.csv')
head(tbphila)
```

Como a base está com medidas latitude e longitude, o padrão de pontos é NÃO-PROJETADO. Para utilizar as técnicas de coordenadas, é necessário que o Shape e a base estejam no mesmo padrão referencial de coordenadas (UTM ou WGS84).

Dimensões do Dataset - quantidade de linhas e de colunas, respectivamente:

```
dim(tbphila)
```

Datatypes do Dataset:

```
str(tbphila)
```

Removendo os dados nulos para plot do Shape

Importando o Shape

```
shapephila = read_sf("neighborhoods_Azavea.shp")
class(shapephila)
shapephila
```

```
shape_simp = ms_simplify(input = shapephila,
                          keep = 0.1,
                          keep_shapes = TRUE)
```

Trata-se de um Shape Polígono, bidimensional, no padrão de coordenadas EPSG (NAD83).

Analisando o sistema de coordenadas do Shape

```
st_crs(shape_simp)  ### EPSG:2272 = WGS84
```

```
### Transforma em coordenadas projetadas no mapa global
philatransform = st_transform(shape_simp, crs = "WGS84")
```

O Shape se encontra no sistema de coordenadas EPSG:2272 - Foi convertido para o sistema WGS84.

Plot do Shape Philadelphia

```
#Plotando o shape
ggplot(data = philatransform) +
  geom_sf(fill = "White")
```

Plot do mapa com os pontos da base de violações na Philadelphia:

```
#Mapa com os pontos - ggplot
```

```
ggplot(data = philatransform) +
  geom_sf(fill = "White") +
  geom_point(data = tbphila,
             aes(x = lng,
                 y = lat),
             colour = 'Dark Red',
             size = 1.75) +
  ggtitle("Violacoes registradas na Filadelfia") +
  theme_light()
```

#Mapa com os pontos - ggmap

```
qmapplot(x = lng,
          y = lat,
          data = tbphila,
          colour = I('blue'),
          size = I(1.5),
          darken = 0.03)
```

Seleção para Violações para Plots

```
tbphila %>% count(violation_code, violation_description)
```

Para selecionar as Violações, elencamos a contagem de cada violação; como são apenas 5: escolhemos a com maior incidência, com menor incidência e a mediana.

Plotando os mapas com as Violações escolhidas: “Recpt Not Littr Free”, “Unlicensd Cstr Dumps” e “No Permit Fixture”

Base Collections

```
tbrecept <- filter(tbphila, violation_code == "107142")
plotrecept <- ggplot(data = philatransform) +
  geom_sf(fill = "White") +
  geom_point(data = tbrecept,
             aes(x = lng,
                 y = lat),
             colour = 'Dark Red',
             size = 1.75) +
  ggtitle("Violacoes - Recpt Not Littr Free") +
  theme_light()
```

Base Recpt

```
tbunlicensed <- filter(tbphila, violation_code == "116102")
plotunlicensed <- ggplot(data = philatransform) +
  geom_sf(fill = "White") +
  geom_point(data = tbunlicensed,
             aes(x = lng,
                 y = lat),
             colour = 'Dark Blue',
             size = 1.75) +
```

```

ggtitle("Violacoes - Unlicensed Cstr Dumps") +
theme_light()

## Base Unlicensed
tbfixture <- filter(tbphila, violation_code == "106111K")
plotfixture <- ggplot(data = philatransform) +
  geom_sf(fill = "White") +
  geom_point(data = tbfixture,
             aes(x = lng,
                 y = lat),
             colour = 'Dark Green',
             size = 1.75) +
  ggtitle("Violacoes - No Permit Fixture") +
  theme_light()

plotrecept
plotunlicensed
plotfixture

plot_grid(
  plotrecept,
  plotunlicensed,
  plotfixture,
  labels="Plot das violacoes escolhidas",
  ncol = 3,
  nrow = 1)

```

Insights Pode-se notar que a concentração de pontos está localizada, no caso das 3 violações selecionadas, na parte central de Filadélfia e na região Sudoeste da mesma. No caso da violação de Receipt, também existe uma alta concentração de pontos na região mais ao norte (40.05º latitude), bem como na região Sul (39.95º latitude). No demais, as 3 violações apresentam distribuições de pontos relativamente semelhantes.

Questão 1.2 - Análise Exploratória Investigar os efeitos de 1ª e 2ª ordem das localizações das violações cometidas. Interprete todos os resultados apresentados. Discutam as possíveis diferenças observadas entre os edeitos estimados dos 3 cenários avliados, isto é, o comportamento da intensidade das violações parece ocorrer de formar similar para tosos os seus tipos para a cidade da Philadelphia ou padrões tendem a ser diferentes?

Efeitos de 1ª Ordem

```

#Transformando o CRS do shape
shapeH_32718 = st_transform(shape_simp,
                           crs = 32718)

```

```

#Criando um objeto owin (observation window)
shapeH0 <- as.owin(shapeH_32718)

```

O Shape se encontra no sistema de coordenadas EPSG:4326 - Foi convertido para o sistema WGS854.

```
#CRS do shapefile
base_4326 = st_as_sf(x = tbphila,
                    coords = c("lng", "lat"),
                    crs = 4326)

base_32718 = st_transform(x = base_4326,
                        crs = 32718)
```

Extraindo bases transformadas por Violação

```
## Base Recept
base_recept_32718 <- filter(base_32718, violation_code == "107142")

## Base Unlicensed
base_unlicensed_32718 <- filter(base_32718, violation_code == "116102")

## Base Fixture
base_fixture_32718 <- filter(base_32718, violation_code == "106111K")
```

Extraindo as coordenadas de cada área

```
#Extraindo as coordenadas de cada base
coordenadas_recept = st_coordinates(x = base_recept_32718)
coordenadas_unlicensed = st_coordinates(x = base_unlicensed_32718)
coordenadas_fixture = st_coordinates(x = base_fixture_32718)
coordenadas_filadelfia = st_coordinates(x = base_32718)

#Visualizando o objeto coordenadas
head(coordenadas_recept)
head(coordenadas_unlicensed)
head(coordenadas_fixture)
head(coordenadas_filadelfia)
```

Criando o padrão de pontos a ser plotado

```
#Criando o padrao de pontos a ser plotado
Receptppp = ppp(x = coordenadas_recept[,1],
                y = coordenadas_recept[,2],
                window = shapeH0)

Unlicensedppp = ppp(x = coordenadas_unlicensed[,1],
                    y = coordenadas_unlicensed[,2],
                    window = shapeH0)

Fixtureppp = ppp(x = coordenadas_fixture[,1],
                 y = coordenadas_fixture[,2],
```

```

        window = shapeH0)

Filadelfiapp = ppp(x = coordenadas_filadelfia[,1],
                  y = coordenadas_filadelfia[,2],
                  window = shapeH0)

### Estimando o Raio

raio.est_recept = bw.diggle(Receptppp)
raio.est_recept

raio.est_unlicensed = bw.diggle(Unlicensedppp)
raio.est_unlicensed

raio.est_fixture = bw.diggle(Fixtureppp)
raio.est_fixture

raio.est_filadelfia = bw.diggle(Filadelfiapp)
raio.est_filadelfia

### Verificação da unidade de medida utilizada e estimando a intensidade com Kernel
Gaussiano

#Checando a unidade de medida
st_crs(shapeH_32718)

#Estimando a intensidade com kernel Gaussiano
Recept_de.g = density.ppp(x = Receptppp,
                          sigma = raio.est_recept,
                          kernel = "gaussian")

Unlicensed_de.g = density.ppp(x = Unlicensedppp,
                              sigma = raio.est_unlicensed,
                              kernel = "gaussian")

Fixture_de.g = density.ppp(x = Fixtureppp,
                           sigma = raio.est_fixture,
                           kernel = "gaussian")

### Plotando os dados e as funcoes intensidades estimadas pela função kernel Normal

par(mfrow=c(2,3))

par(mar=c(2.5,2.5,1.5,.5))

plot(Receptppp,
     pch=21,
     cex=0.5,

```



```

    bg="Dark Blue", main="Ocorrências de Recept",
    cex.main=0.5)

plot(Unlicensedppp,
     pch=21,
     cex=0.5,
     bg="Dark Red", main="Ocorrências de Unlicensed",
     cex.main=0.5)

plot(Fixtureppp,
     pch=21,
     cex=0.5,
     bg="Dark green", main="Ocorrências de Fixture",
     cex.main=0.5)

plot(Recept_de.g,
     main="Função intensidade de Recept",
     cex.main=0.5)

plot(Unlicensed_de.g,
     main="Função intensidade de Unlicensed",
     cex.main=0.5)

plot(Fixture_de.g,
     main="Função intensidade de Fixture",
     cex.main=0.5)

par(mfrow=c(1,1))

```

Insights

Ao se analisar o efeito de 1ª ordem de cada tipo de violação analisada, em que a intensidade dos pontos espaciais, resultantes das densidades desses pontos, são levados em consideração. Ao se analisar os gráficos de intensidade obtidos, nota-se que os 3 tipos de violação ("Recept", "Unlicensed" e "Fixture") não podem ser denominados como processos pontuais estacionários, uma vez que suas propriedades dependem de sua localização espacial, isso pode ser notado através da não homogeneidade das intensidades de seus gráficos espaciais ao longo de todo seu território. De modo geral, temos um comportamento, relativamente, semelhante para o efeito de 1ª ordem de cada tipo de ocorrência, em que as regiões de alta intensidade/densidade de pontos espaciais estão concentrados na região Centro-Sudoeste da Filadélfia. Entretanto, nota-se que a ocorrência "Fixture" possui uma maior intensidade na região mencionada anteriormente, quando comparado com as ocorrências "Recept" e "Unlicensed".

Efeitos de 2ª Ordem

Estimando as funções G de cada ocorrência

```
Recept.G = Gest(Receptppp)
Unlicensed.G = Gest(Unlicensedppp)
Fixture.G = Gest(Fixtureppp)

#Plotando as funcoes G de cada ocorrência
par(mfrow = c(2,2))

par(mar=c(4,2.5,1.5,.5))

plot(Recept.G,
     main="Funcao G de Recept",
     xlab = "metro")

plot(Unlicensed.G,
     main="Funcao G de Unlicensed",
     xlab = "metro")

plot(Fixture.G,
     main="Funcao G de Fixture",
     xlab = "metro")
par(mfrow = c(1,1))
```

Insights Ao se analisar o efeito de 2a ordem de cada tipo de violação analisada, em que esse efeito fornece informações sobre a interação de eventos em pequena escala. Ao se analisar a função G de cada tipo de ocorrência, nota-se que os pontos das 3 ocorrências indicam uma tendência de agrupamento de pontos, uma vez que a curva G estimada se encontra acima da curva de aleatoriedade de pontos e apresenta um comportamento de crescimento contínuo ao longo do eixo X. Entretanto, para garantir que os pontos apresentem realmente uma tendência de agrupamento e sair do campo da intuição, deve-se realizar o teste de CSR, o qual é apresentado a seguir.

Realizando o teste de Clark-Evans para verificar agregacao espacial

```
Recept_clarkevans <- clarkevans.test(X = Receptppp,
                                   alternative = "clustered")
Recept_clarkevans

Unlicensed_clarkevans <- clarkevans.test(X = Unlicensedppp,
                                       alternative = "clustered")
Unlicensed_clarkevans

Fixture_clarkevans <- clarkevans.test(X = Fixtureppp,
                                    alternative = "clustered")
Fixture_clarkevans
```

Insights Ao se realizar o teste de Clar-Evans em conjunto com a adoção da premissa de que o nível de significância do teste seja de 5%, nota-se que para as 3 ocorrências a hipótese

nula é rejeitada, uma vez que o p-valor do teste de Clark-Evans para cada tipo de ocorrência é menor que 5%. Logo, pode-se assumir que existe uma relação de agrupamento entre os pontos de cada ocorrência. Logo, o efeito de 2ª ordem para cada tipo de ocorrência resulta no apontamento de agrupamento de pontos. Por fim, pode-se concluir que nos 3 tipos de ocorrência selecionados, o comportamento da densidade de pontos ao longo da superfície geográfica e as interações entre esses pontos com seus respectivos pontos vizinhos apresentam um certo padrão, uma vez que os pontos nas 3 ocorrências selecionadas encontram-se com elevada intensidade na região Centro-Sudoeste da Filadélfia, não apresentando características de processos estacionários ou homogêneos. Além disso, nas 3 ocorrências foram identificados padrões de agrupamento de pontos, os quais são evidenciados pela curva estimada da função G e pelo teste de Clark-Evans.

Base Rio de Janeiro # Questão 1 Crie uma visualização apropriada para a taxa de lesão corporal dolosa por 100.000 habitantes em 2019 e para a taxa de roubo de rua por 100.000 habitantes no ano de 2019 considerando os municípios do Estado do Rio de Janeiro. Discuta a superfície de taxas estimadas.

– Análise exploratória: 1. Mapa coroplético, 2. Definição de uma matriz de vizinhança W (queen, rook, distância), 3. Definição dos pesos de W, 4. Estimativa da autocorrelação global e teste de hipóteses (I de Moran), 5. Estimativa da autocorrelação local e teste de hipóteses (I local de Moran).

Importando o dataset: contem os dados dos municípios do Rio de Janeiro com a quantidade de incidentes ocorridos para Lesão Corporal Dolosa (LCD) e Roubos de Rua (RR); O outro dataset, contem os dados de evolução populacional por Municípios por vários anos. #### Visualização dos 5 primeiros registros de cada Dataset: - Dataset Incidentes no Rio de Janeiro

#Importando a base

```
Base_municipio_mensal = read_csv2(file = "BaseMunicipioMensal.csv")
head(Base_municipio_mensal)
```

- Dataset de Evolução Populacional por Municípios

#Importando população ev mensal municipio

```
Pop_ev_mensal_mun = read_csv2(file = "PopulacaoEvolucaoMensalMunic.csv")
head(Pop_ev_mensal_mun)
```

União dos Datasets

#Filtrando 2019

```
Base_municipio_mensal = filter(Base_municipio_mensal, ano == "2019")
```

#Acrescentando a variável no shape

```
base_join = left_join(x = Base_municipio_mensal,
                      y = Pop_ev_mensal_mun,
                      by = c("fmun_cod" = "cod_munic"))
head(base_join)
```

Ajustando o Dataset para população por 100 mil habitantes

```
#Taxa de Lesão corporal dolosa por 100.000 habitantes em 2019 (TLCD)
#Taxa de roubo de rua por 100.000 habitantes no ano de 2019 (TRR)
base_join$TLCD <- base_join$lesao_corp_dolosa/base_join$pop_munic*100000

base_join$TRR <- base_join$roubo_rua/base_join$pop_munic*100000

#Transformando a variavel fmun_cod em character
base_join$fmun_cod <- as.character(base_join$fmun_cod)

### Importando shape

Rio = read_sf("33MUE250GC_SIR.shp")
class(Rio)
Rio
```

Simplificando o numero de pontos no shape

```
Rio_simp = ms_simplify(Rio,
                        keep_shapes = TRUE)
```

Plot do Shape Rio de Janeiro

```
ggplot(Rio_simp) +
  geom_sf(fill = "White")
```

Acrescentando a variavel no shape

```
Rio_final = left_join(Rio_simp, base_join, by = c("CD_GEOCMU" = "fmun_cod"))
head(Rio_final)
```

#1) Mapa coroplético

Mapas - GGLOT

```
# Mapa- TLCD
map_TLCD_ggplot <- ggplot(data = Rio_final) +
  geom_sf(aes(fill = TLCD)) +
  scale_fill_gradient(name = "TLCD",
                      low = "White",
                      high = "Red") +

  theme_light() +
  theme(legend.title = element_text(size = 16),
        legend.text = element_text(size = 10,
                                     colour = "Red"))

map_TLCD_ggplot
# Mapa- TRR
map_TRR_ggplot <- ggplot(data = Rio_final) +
  geom_sf(aes(fill = TRR)) +
  scale_fill_gradient(name = "TRR",
                      low = "White",
```

```

        high = "Red") +
  theme_light() +
  theme(legend.title = element_text(size = 16),
        legend.text = element_text(size = 10,
                                     colour = "Red"))
map_TRR_ggplot

#Plotando os dois mapas em uma mesma figura
grid.arrange(map_TLCD_ggplot,
              map_TRR_ggplot,
              ncol = 1,
              nrow = 2)

```

Mapas - TMAP

```

#Colocando todos os graficos em uma mesma escala
tm_shape(shp = Rio_final) +
  tm_fill(col = c("TLCD", "TRR"),
          title = c("TAXA POR 100 MIL HABITANTES"),
          palette = "Blues") +
  tm_facets(nrow = 1,
            free.scales = FALSE) +
  tm_borders() +
  tm_layout(panel.labels = c("TLCD", "TRR"))

```

INSIGHTS Podemos observar que a Taxa de lesão corporal dolosa (TLCD) comparada com a Taxa de roubo de rua (TRR) possui maior incidência no estado do RJ-Dados da base de 2019 por 100 mil habitantes.

TLCD possui maior dispersão ao longo do estado ao passo que TRR está mais concentrado na região da capital e baixada fluminense

Questão 2 Verifique se existe dependência espacial para as duas taxas de interesse. Avalie diferentes critérios de vizinhança e diferentes pesos. Conclua usando um nível de significância de 5%.

Matriz de proximidade espacial: Duas principais abordagens • Contiguidade • Distância

Contiguidade binária: • Critério rook (borda comum), • Critério bishop (vértice comum), • Critério queen (borda ou vértice comum).

Critério de Contiguidade:

Definição da matriz de vizinhança (W): contiguidade (bishop, rook e queen) ou distância (k-vizinhos mais próximos). • W define quem é vizinho de quem e a intensidade (peso) dessa relação

Criando a matriz de vizinhança W baseada no critério de contiguidade queen

```
W.queen = poly2nb(pl = Rio_final,
                  row.names = Rio_final$NM_MUNICIP,
                  queen = TRUE)

W.rook = poly2nb(pl = Rio_final,
                 row.names = Rio_final$NM_MUNICIP,
                 queen = FALSE)

summary(W.rook)
```

Definição dos pesos de W - Queen

Lista de vizinhança espacial com pesos

```
recWQW <- nb2listw(neighbours = W.queen,
                  style="W") #outras opcoes: B, C, S e U

recWQB <- nb2listw(neighbours = W.queen,
                  style="B") #outras opcoes: B, C, S e U
```

- Rook

```
recWRW <- nb2listw(neighbours = W.rook,
                  style="W") #outras opcoes: B, C, S e U

recWRB <- nb2listw(neighbours = W.rook,
                  style="B") #outras opcoes: B, C, S e U
```

Indicadores de autocorrelação global: • Índice I global de Moran, • C de Gery, • Getis e Ord's G(d). Indicadores de autocorrelação local: • Índice I local de Moran,

Indicadores de autocorrelação global

Calculando o índice de moran considerando uma estrutura de vizinhança Queen com padronizacao pelas linhas

```
#Para TLCD - Taxa de Lesão corporal dolosa por 100.000 habitantes em 2019
moran.test(x = Rio_final$TLCD,
           listw = recWQW)

#Para TRR - Taxa de roubo de rua por 100.000 habitantes no ano de 2019
moran.test(x = Rio_final$TRR,
           listw = recWQW)
```

Calculando o índice de moran considerando uma estrutura de vizinhança Queen com pesos iguais

```
#Para TLCD - Taxa de Lesão corporal dolosa por 100.000 habitantes em 2019
moran.test(x = Rio_final$TLCD,
           listw = recWQB)
```

```
#Para TRR - Taxa de roubo de rua por 100.000 habitantes no ano de 2019
moran.test(x = Rio_final$TRR,
           listw = recWQB)
```

Calculando o índice de moran considerando uma estrutura de vizinhanca Rook com padronizacao pelas linhas

```
#Para TLCD - Taxa de Lesão corporal dolosa por 100.000 habitantes em 2019
moran.test(x = Rio_final$TLCD,
           listw = recWRW)
```

```
#Para TRR - Taxa de roubo de rua por 100.000 habitantes no ano de 2019
moran.test(x = Rio_final$TRR,
           listw = recWRW)
```

Calculando o índice de moran considerando uma estrutura de vizinhanca Rook com pesos iguais

```
#Para TLCD - Taxa de Lesão corporal dolosa por 100.000 habitantes em 2019
moran.test(x = Rio_final$TLCD,
           listw = recWRB)
```

```
#Para TRR - Taxa de roubo de rua por 100.000 habitantes no ano de 2019
moran.test(x = Rio_final$TRR,
           listw = recWRB)
```

Indicadores de autocorrelação local Calculando o I local de Moran

```
#Para TLCD
moranlocREC_TLCD = localmoran(x = Rio_final$TLCD,
                              listw = recWQW,
                              na.action=na.exclude,
                              zero.policy=TRUE)
```

```
moranlocREC_TLCD
```

```
#Para TREE
moranlocREC_TRR = localmoran(x = Rio_final$TRR,
                              listw = recWQW,
                              na.action=na.exclude,
                              zero.policy=TRUE)
```

```
moranlocREC_TRR
```