Geospatial Point Density

by Paul F. Evangelista and David Beskow

Czego dotyczy i czy wciąż jest aktualne

Wprowadzenie

Cel algorytmu

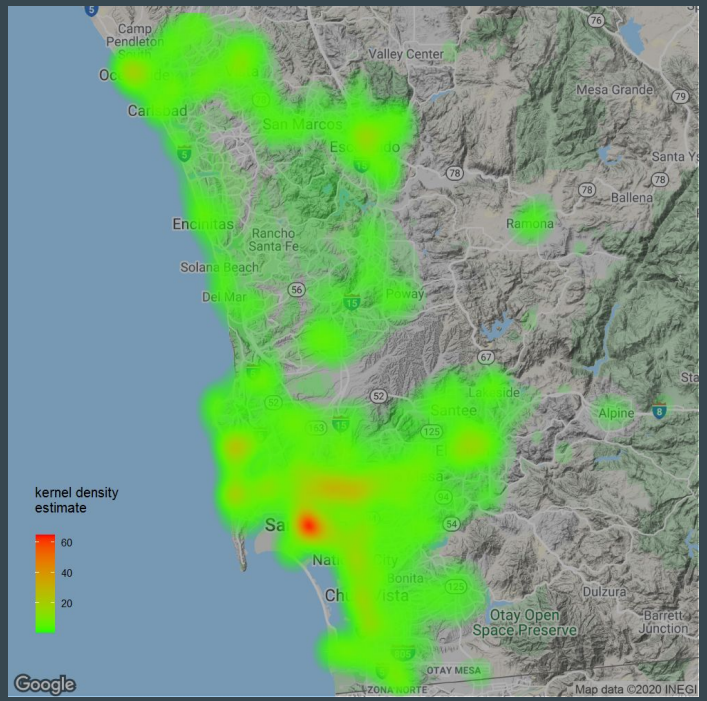
Po lewej znajduje się jedna z pierwszych map do analizy danych przestrzennych.

Pozwoliła na wykrycie źródła cholery.

Celem algorytmu autorów artykułu jest szybkie tworzenie rzetelnych wizualizacji danych przestrzennych.

The original map drawn by Dr. John Snow (1813-1858), a British physician who is one of the founders of medical epidemiology, showing cases of cholera in the London epidemics of 1854, clustered around the locations of water pumps.

Problem poprzednich algorytmów



Algorytmy były nastawione na dane ciągłe.

Dane dyskretne były wygładzane/rozmazywane.

Przez to obszar *przestępczy* był pokazywany poza miejscami występowania, m.in. na morzu.

Figure 1 from *Geospatial Point Density* by Paul F. Evangelista and David Beskow, The R Journal, 2018

Algorytm

Hash-based algorithm

Store n points and let lat_i, lon_i, and $date_i$ represent the latitude, longitude, and date of each point, respectively. Let g represent the grid size measured in degrees latitude, and r represent the radius measured in grid steps. For each lat_i and lon_i, round each to the nearest grid point, and store the rounded points as $tlat_i$ and $tlon_i$. Set $m = 0$.
for $i = 1$ to n **do**
  if bin_density_hash$(tlat_i, tlon_i)$ exists **then**
    bin_density_hash$(tlat_i, tlon_i) + +$
    bin_temporal_hash$(tlat_i, tlon_i) += date_i$
  else
    bin_density_hash$(tlat_i, tlon_i) = 1$
    bin_temporal_hash$(tlat_i, tlon_i) = date_i$
    active_grid_hash$(m) = (tlat_i, tlon_i)$
    $m + +$
  end if
end for
for $j = 1$ to m **do**
  retrieve $tlat_j$ and $tlon_j$ from active_grid_hash(j)
  for $lat_t = tlat_j - rg$ to $tlat_j + rg$ **do**
    $t = \arccos(\cos(rg)/\cos(lat_t - tlat_j))/g$
    round t to the nearest integer
    $t = t * g$
    for $lon_t = tlon_j - t$ to $tlon_j + t$ **do**
      density_hash $(lat_t, lon_t) + +$
      temporal_hash$(lat_t, lon_t) +=$ temporal_hash$(tlat_j, tlon_j)$
      $lon_t = lon_t + g$
    end for
    $lat_t = lat_t + g$
  end for
end for
for $i = 1$ to n **do**
  round $(lat_t = lat_i/g)$ to the nearest integer
  $lat_t = lat_t * g$
  round $(lon_t = lon_i/g)$ to the nearest integer
  $lon_t = lon_t * g$
  temporal_hash$(lat_t, lon_t) =$ temporal_hash$(lat_t, lon_t)/$density_hash(lat_t, lon_t)
  print $lat_i, lon_i,$ density_hash(lat_t, lon_t), temporal_hash(lat_t, lon_t)
end for

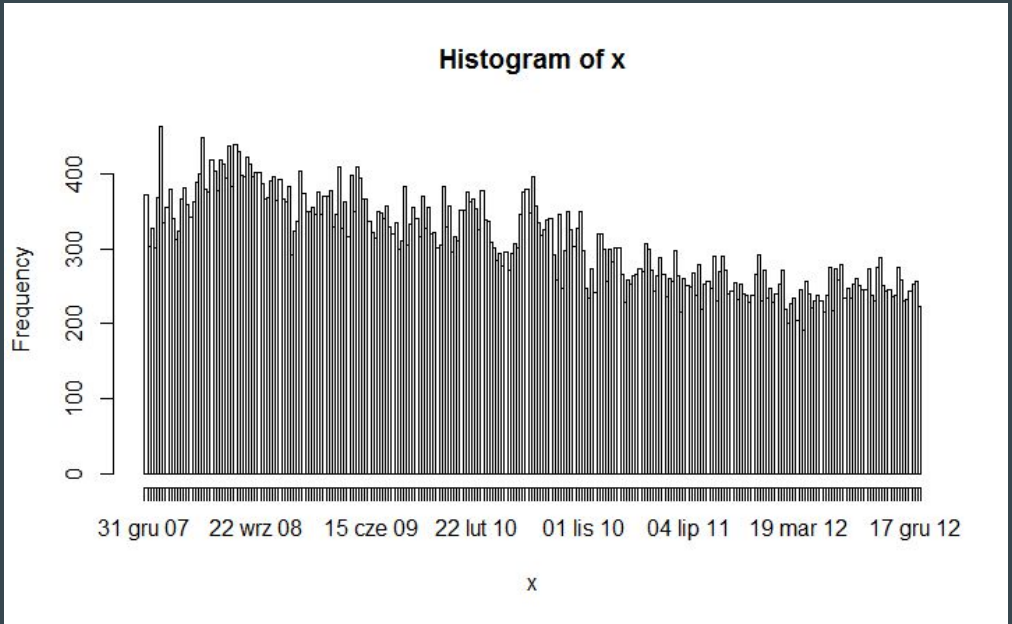
Początkowo okazał się sukcesem, bo był szybszy od poprzednich algorytmów.

Jednak jest wolniejszy przy większych datasetach.

Matrix-based algorithm

https://awwapp.com/b/uokt9aovd/

Trend



Produktem ubocznym tworzenia algorytmu była koncepcja obliczania tendencji występowania eventów z pewnego sąsiedztwa w czasie.

W bardzo prosty sposób można wyeksponować wcześniej niezauważone trendy i schematy.

Figure 6 from *Geospatial Point Density* by Paul F. Evangelista and David Beskow, The R Journal, 2018

Wyniki

Reprodukowalność - przyjęta definicja

LeVeque (2009):

The idea of `reproducible research` in scientific computing is to archive and make publicly available all the codes used to create paper's figures or tables, preferably in such a manner that readers can download the codes and run them to reproduce the results.

From: DATA 598 A Wi 20: Special Topics In Data Science: Reproducibility for Data Science

Reprodukowalność

Wyniki udało się zreprodukować.

Wszystkie kody udostępnione przez autorów są wciąż aktualne.

Uzyskane wyniki zgadzają się z tymi przedstawionymi w artykule.

Uzyskane wykresy - mapy

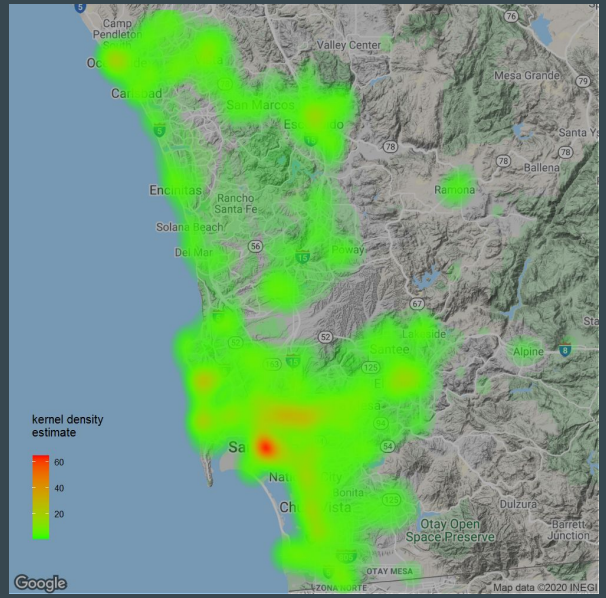


Figure 1 - bkde2D

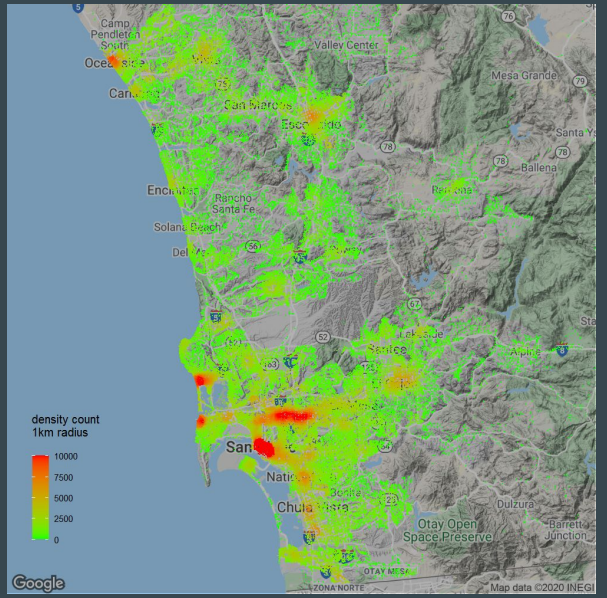


Figure 4 - pointdensity

Uzyskane wykresy - trend

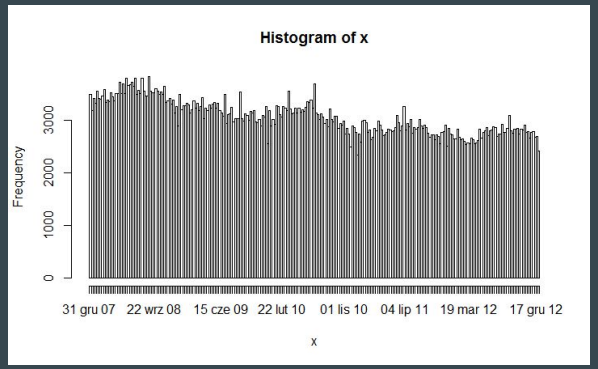


Figure 6 - San Diego

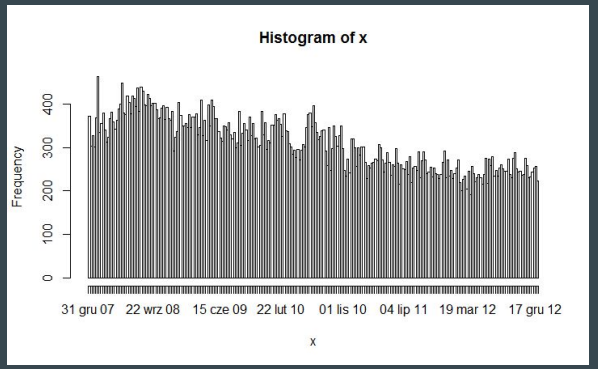


Figure 6 - Mid-city

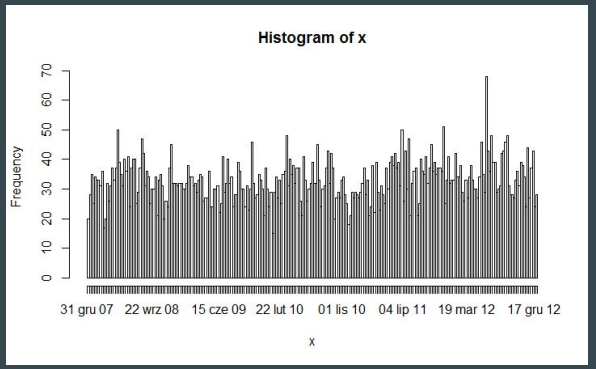


Figure 6 - Encinitas

Statystyki modeli

	Estimate	Std. Error	t value	Pr(>|t|)
San Diego	-0.5077	0.0204	-24.89	<2e-16
Mid-city	-9.577e-02	3.711e-03	-25.81	<2e-16
Encinitas	0.0023281	0.0008093	2.877	0.00435

Dziękujemy za uwagę

...

Wojciech Szczypek & Anna Urbala