

Smooth Pycnophylactic Interpolation Produced by Density-Equalising Map Projections

Michael T. GASTNER¹, Nihal Z. MIAJI, Adi SINGHANIA

Yale-NUS College, 16 College Avenue West, #01-220, Singapore 138527

¹michael.gastner@yale-nus.edu.sg (ORCID: 0000-0002-1097-8833)

Abstract. A large amount of quantitative geospatial data is collected and aggregated in discrete enumeration units (e.g. countries or states). Smooth pycnophylactic interpolation aims to find a smooth, nonnegative function such that the area integral over each enumeration unit is equal to the aggregated data. Conventionally, smooth pycnophylactic interpolation is achieved by a cellular automaton algorithm that converts a piecewise constant function into an approximately smooth function defined on a grid of coordinates on an equal-area map. An alternative approach, proposed by Tobler in 1976, is to construct a density-equalising map projection in which areas of enumeration units are proportional to the aggregated data. A pycnophylactic interpolation can be obtained from the Jacobian of this projection. Here, we describe a software implementation of this method. Although solutions are not necessarily optimal in terms of predefined quantitative measures of smoothness, our method is computationally efficient and can potentially be used in tandem with other methods to accelerate convergence towards an optimal solution.

Keywords: spatial interpolation, pycnophylactic density, contiguous cartogram, flow-based algorithm, fast Fourier transform, Gaussian convolution, Jacobian determinant

1 Introduction

Quantitative geospatial data are often available only as aggregated numbers for discrete enumeration units. For example, national statistics agencies usually report the number of individuals living in each administrative division of a country (e.g. a census block in the United States and an Output Area in the United Kingdom) but do not release information about each individual's exact location. Because it is impossible to infer exact locations, the aggregated data are often converted into a density function (in units of people per square kilometre) that assigns a real-valued number to each point belonging to the continuum of coordinates in the country. Let us assume that a country is divided into enu-

meration units U_1, \dots, U_n . We denote the population count in U_i by P_i and the area by A_i . Furthermore, we assume that coordinates have already been converted from longitude and latitude to Cartesian coordinates (x, y) using an equal-area map projection. A bounded, non-negative function $\rho(x, y)$ is referred to as a pycnophylactic density (from the Greek words πυκνός, meaning 'dense', and φύλαξ, meaning 'guard') if the aggregate density in U_i is equal to the observed population P_i :

$$\iint_{U_i} \rho(x, y) dx dy = P_i \text{ for } i = 1, \dots, n. \quad (1)$$

The variable P_i does not need to be population in a narrow sense; it may also refer to other data that are

Glatka piknofilaktička interpolacija dobivena s pomoću kartografske projekcije s ujednačavanjem gustoće

Michael T. GASTNER¹, Nihal Z. MIAJI, Adi SINGHANIA

Yale-NUS College, 16 College Avenue West, #01-220, Singapore 138527

¹michael.gastner@yale-nus.edu.sg (ORCID: 0000-0002-1097-8833)

Sažetak. Velika količina kvantitativnih geoprostornih podataka prikuplja se i spaja u diskretne popisne jedinice (npr. zemlje ili države). Glatka piknofilaktička interpolacija ima za cilj pronaći glatku, nenegativnu funkciju tako da je integral površine nad svakom popisnom jedinicom jednak agregiranim podacima. Konvencionalno, glatka piknofilaktička interpolacija dobiva se algoritmom staničnog automata koji pretvara po dijelovima konstantnu funkciju u približno glatku funkciju definiranu na mreži koordinata na karti u ekvivalentnoj projekciji. Alternativni pristup, kojeg je predložio Tobler (1976), je konstruirati kartografsku projekciju s ujednačavanjem gustoće u kojoj su područja popisnih jedinica proporcionalna agregiranim podacima. Piknofilaktička interpolacija može se dobiti iz Jakobijana te projekcije. U ovom radu opisujemo primjenu softvera za tu metodu. Iako rješenja nisu nužno optimalna u smislu unaprijed definiranih kvantitativnih mjera glatkoće, naša metoda je računski učinkovita i potencijalno se može koristiti zajedno s drugim metodama da bi se ubrzala konvergencija prema optimalnom rješenju.

Ključne riječi: prostorna interpolacija, piknofilaktička funkcija, kartogram susjedstva, algoritam utemeljen na protoku, brza Fourierova transformacija, Gaussova konvolucija, Jacobijan

1. Uvod

Kvantitativni geoprostorni podatci često su dostupni samo kao agregirani brojevi za diskretne popisne jedinice. Na primjer, nacionalne statističke agencije obično izvještavaju o broju pojedinaca koji žive u svakoj administrativnoj podjeli zemlje (npr. popisni blok u Sjedinjenim Državama i izlazno područje u Ujedinjenom Kraljevstvu), ali ne objavljuju informacije o točnoj lokaciji svakog pojedinca. Budući da je nemoguće imati točne lokacije, agregirani podatci često se pretvaraju u funkciju gustoće (u jedinicama ljudi po kvadratnom kilometru) koja svakoj točki koja pripada kontinuumu koordinata dodjeljuje realni broj. Pretpostavimo da je država podijeljena

na popisne jedinice U_1, \dots, U_n . Označimo broj stanovnika u U_i s P_i , a površinu s A_i . Nadalje, pretpostavljamo da su koordinate već konvertirane iz geografskih u kartezijeve (x, y) upotrebom neke ekvivalentne projekcije. Omeđena, nenegativna funkcija $\rho(x, y)$ je piknofilaktička ili piknofilaksa (od grčke riječi $\pi\kappa\nu\nu\acute{o}\varsigma$, što znači 'gust', i $\phi\acute{o}\lambda\alpha\acute{\xi}$, što znači 'čuvar') ako je agregirana gustoća u U_i jednaka opažanoj populaciji P_i :

$$\iint_{U_i} \rho(x, y) dx dy = P_i \text{ za } i = 1, \dots, n. \quad (1)$$

Varijabla P_i ne mora biti populacija u uskom smislu; može se odnositi na druge podatke koji su dostupni u agregiranom obliku (npr. bruto

only available in aggregated form (e.g. gross regional product or CO₂ emissions by enumeration unit). Henceforth, we use the term ‘population’ for any type of nonnegative, spatially extensive aggregated data.

An example of a pycnophylactic density is the following piecewise constant function:

$$\rho_{\text{plateau}}(x, y) = \begin{cases} P_i / A_i & \text{if } (x, y) \in U_i, \\ \bar{\rho} & \text{if } (x, y) \notin \bigcup_i U_i, \end{cases} \quad (2)$$

where

$$\bar{\rho} = \frac{\sum_i P_i}{\sum_i A_i} \quad (3)$$

is the spatially averaged density. In principle, the density outside all enumeration units can be chosen arbitrarily because it cannot be inferred from the available data. However, it turns out to be mathematically convenient to impose the condition

$$\forall (x, y) : (x, y) \notin \bigcup_i U_i \rightarrow \rho(x, y) = \bar{\rho}, \quad (4)$$

which can be statistically interpreted as imputation of missing data by substituting the mean. Henceforth, we apply equation (4) as a condition on any pycnophylactic density. Figure 1(a) illustrates the definition of ρ_{plateau} using COVID-19 cases in Croatia between 25 February 2020 and 24 March 2022 (Croatian Institute of Public Health, 2022). In this example, the density is obtained by dividing the number of COVID-19 cases in a county by the county’s area. Although ρ_{plateau} is easily calculated, piecewise constant densities are unsuitable statistical models for COVID-19 cases and many other geospatial data because of the discontinuities at the boundaries of enumeration units. As pointed out by Openshaw (1984): ‘The areal units used to report census data (enumeration districts, census tracts, wards, local government units) have no natural or meaningful geographical identity.’ Thus, to reduce the artefacts introduced by arbitrary boundaries of enumeration units, it is generally preferable to work with a smooth density, as illustrated in Figure 1(b), instead of ρ_{plateau} , whose shape strongly depends on the location of the boundaries.

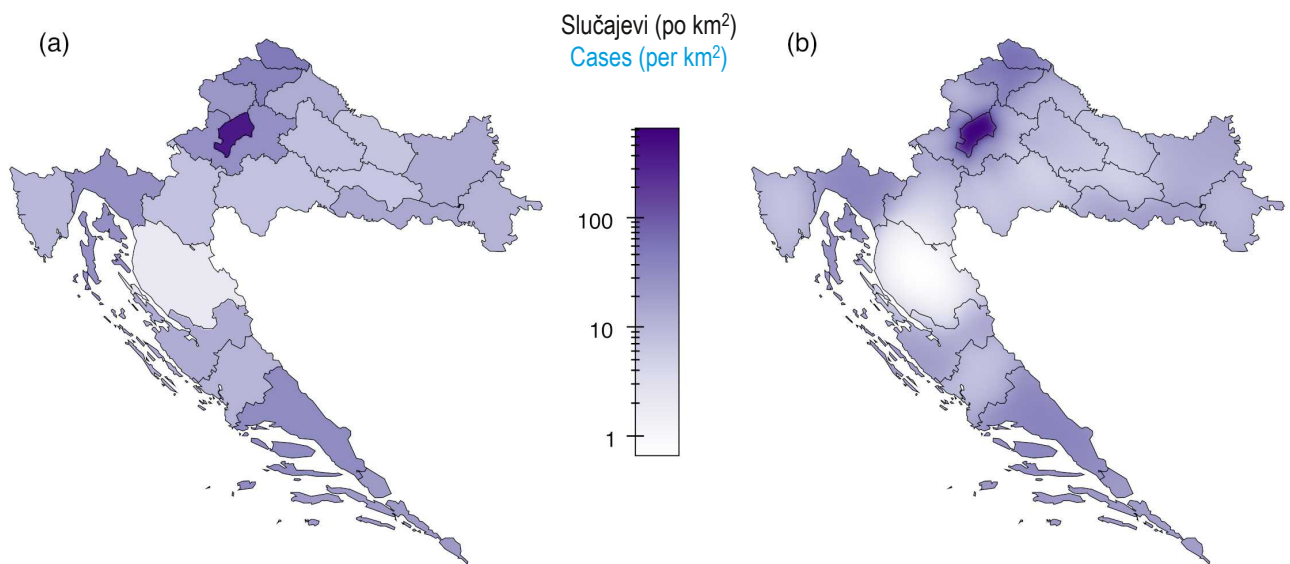
Tobler (1979a) introduced a cellular automaton algorithm for generating smooth pycnophylactic densities, which approximates the continuum of space with a fine-grained square grid. Each point (x, y) on the grid is initially assigned the density $\rho_{\text{plateau}}(x, y)$. Thereafter, the density associated with each grid point is adjusted

so that the absolute value of the discrete Laplacian is reduced, subject to the constraints that the sum of densities in each enumeration unit is conserved and density remains nonnegative. This adjustment is iterated until the changes are below a small threshold value or until the number of iterations reaches a predefined limit.

Tobler’s (1979a) algorithm has become the standard technique for smooth pycnophylactic interpolation. The algorithm has been implemented in R (Brunsdon, 2014) and Python (Pysal Developers, 2021). For users of Geographic Information Systems software, the algorithm is also available via extensions of ArcGIS (Qiu et al., 2012) and GRASS (Metz, 2013). A variant of Tobler’s (1979a) algorithm was developed by Rase (2001), in which the regular square grid is replaced by an irregular triangular network. However, the algorithm by Rase (2001) and its later refinement by the same author (Rase, 2007) keep the essential features of Tobler’s (1979a) algorithm: iterative local averaging and subsequent redistribution of density differences to enforce the pycnophylactic condition of equation (1). Despite the widespread use of Tobler’s (1979a) algorithm, it was not the first method proposed by him for smooth pycnophylactic interpolation. In an earlier publication, Tobler (1976) described an alternative approach in which the boundaries of enumeration units are transformed into an area cartogram (i.e. a map in which all enumeration units are depicted with an area proportional to their population). Tobler’s (1976) proposed method requires the area cartogram to be contiguous; that is, neighbouring enumeration units on the surface of the earth must be neighbours in the cartogram. As noted by Tobler (2017), contiguous cartograms are closely related to density-equalising map projections. In this study, we briefly review the connections between contiguous cartograms, density-equalising map projections and smooth pycnophylactic interpolation. Thereafter, we explain how to achieve smooth pycnophylactic interpolation using a recently developed algorithm that generates contiguous cartograms.

2 Relationship Between Pycnophylactic Interpolation and Density-Equalising Map Projections

To construct a contiguous cartogram, the boundaries of enumeration units are modelled as polylines. We denote the vertices of the polylines on an equal-area map projection by $(v_{1x}, v_{1y}), (v_{2x}, v_{2y}), \dots$. The vertices are then shifted to new positions $(w_{1x}, w_{1y}), (w_{2x}, w_{2y}), \dots$ such that the regions demarcated by the transformed boundaries have an area proportional to the population of the corresponding enumeration units. This transformation can be regarded as the result of a map projection $\mathbf{t} = (t_x, t_y)$ that satisfies the conditions



Slika 1. Dvije piknofilaktičke funkcije koje prikazuju razdiobu slučajeva bolesti COVID-19 u Hrvatskoj između 25. veljače 2020. i 24. ožujka 2022. (a) po dijelovima konstantna gustoća (b) izgladena gustoća dobivena s pomoću projekcije koja ujednačava gustoću.

Fig. 1 Two pycnophylactic densities that represent the distribution of COVID-19 cases in Croatia between 25 February 2020 and 24 March 2022. (a) Piecewise constant density. (b) Smooth density obtained from a density-equalising map projection.

regionalni proizvod ili emisije CO₂ po popisnim jedinicama). Zbog toga se u nastavku koristimo nazivom ‘populacija’ za bilo koji tip nenegativnih, prostorno agregiranih podataka.

Primjer piknofilaktičke funkcije je ova po dijelovima konstantna funkcija:

$$\rho_{\text{plateau}}(x, y) = \begin{cases} P_i / A_i & \text{ako je } (x, y) \in U_i, \\ \bar{\rho} & \text{ako je } (x, y) \notin \bigcup_i U_i, \end{cases} \quad (2)$$

gdje je

$$\bar{\rho} = \frac{\sum_i P_i}{\sum_i A_i} \quad (3)$$

prostorno osrednjena gustoća. U principu, gustoća izvan svih popisnih jedinica može se odabrati proizvoljno jer se ne može zaključiti iz dostupnih podataka. Međutim, pokazalo se da je matematički zgodno zadati uvjet

$$\forall (x, y) : (x, y) \notin \bigcup_i U_i \rightarrow \rho(x, y) = \bar{\rho}, \quad (4)$$

što se može statistički protumačiti kao imputacija podataka koji nedostaju zamjenom srednje vrijednosti. Od sada nadalje primjenjujemo jednadžbu (4) kao uvjet na bilo koju piknofilaktičku funkciju. Slika 1(a) ilustrira definiciju ρ_{plateau} , koristeći slučajeve bolesti

COVID-19 u Hrvatskoj u razdoblju od 25. veljače 2020. do 24. ožujka 2022. (Hrvatski zavod za javno zdravstvo, 2022). U tom primjeru, gustoća se dobiva dijeljenjem broja slučajeva bolesti COVID-19 u županiji s površinom županije. Iako se ρ_{plateau} lako izračunava, po dijelovima konstantne gustoće neprikladni su statistički modeli za slučajeve bolesti COVID-19 i mnoge druge geoprostorne podatke zbog diskontinuiteta na granicama popisnih jedinica. Kao što je istaknuo Openshaw (1984): "Površinske jedinice koje se koriste za izvješćivanje popisnih podataka (popisna područja, okruzi, jedinice lokalne samouprave) nemaju prirodni ili smisleni geografski identitet." Dakle, da bi se smanjili artefakti koji se unose zbog proizvoljnih granica popisnih jedinica, općenito je poželjno raditi s glatkom gustoćom, kao što je prikazano na slici 1(b), umjesto ρ_{plateau} , čiji oblik jako ovisi o položaju granica.

Tobler (1979a) je uveo algoritam staničnog automata za generiranje glatkih piknofilaktičkih funkcija koje aproksimiraju kontinuum prostora s finom kvadratnom mrežom. Svakoj točki (x, y) te mreže pridružena je inicijalno gustoća $\rho_{\text{plateau}}(x, y)$. Nakon toga, gustoća povezana sa svakom točkom mreže prilagođava se tako da se apsolutna vrijednost diskretnog Laplaceana smanjuje, podložno ograničenjima da se zbroj gustoća u svakoj popisnoj jedinici čuva i da gustoća ostaje nenegativna. Ta se prilagodba ponavlja sve dok promjene nisu ispod male vrijednosti praga ili dok broj iteracija ne dosegne unaprijed definirano ograničenje.

$t_x(v_{jx}, v_{jy}) = w_{jx}$ and $t_y(v_{jx}, v_{jy}) = w_{jy}$ for all $j = 1, 2, \dots$. A map projection with this property can be obtained, for example, by solving the following equation:

$$\det J_t(x, y) = \frac{\rho(x, y)}{\bar{\rho}}, \tag{5}$$

where $\det J_t = (\partial t_x / \partial x)(\partial t_y / \partial y) - (\partial t_y / \partial x)(\partial t_x / \partial y)$ is the Jacobian determinant of \mathbf{t} , $\rho(x, y)$ is a pycnophylactic density [i.e. it satisfies equation (1)] and $\bar{\rho}$ is its spatial average given by equation (3). The quantity

$$\rho_{\text{res}}(x, y) = \rho(x, y) - \bar{\rho} \det J_t(x, y) \tag{6}$$

can then be interpreted as ‘residual density’ (i.e. the difference from the mean density that remains unexplained by the projection \mathbf{t}), and the objective is to find a solution \mathbf{t} such that $\rho_{\text{res}}(x, y) = 0$ for all (x, y) .

Contiguous cartograms and density-equalising map projections are related to each other in two ways. First, if ρ and (v_{jx}, v_{jy}) are known, it is possible to solve equation (5) to obtain $\mathbf{t}(x, y)$ and then obtain the polyline vertices (w_{jx}, w_{jy}) of a contiguous cartogram by applying $\mathbf{t}(x, y)$ to (v_{jx}, v_{jy}) for all $j=1, 2, \dots$. Second, if (v_{jx}, v_{jy}) and (w_{jx}, w_{jy}) are known, it is possible to construct a density-equalising map projection $\mathbf{t}(x, y)$ with the property $\mathbf{t}(v_{jx}, v_{jy}) = (w_{jx}, w_{jy})$ and then obtain a pycnophylactic density by inverting equation (5):

$$\rho(x, y) = \bar{\rho} \det J_t(x, y). \tag{7}$$

In both cases, solutions are not unique. If $\rho(x, y)$ is given, equation (5) allows infinitely many solutions $\mathbf{t}(x, y)$ because the number of constraints implicit in equation (5) is 1, which is less than the number of dimensions (i.e. 2) of a geographic map. Conversely, if all (w_{jx}, w_{jy}) are given, there are infinitely many ways to extend a function \mathbf{t} with the property $\mathbf{t}(v_{jx}, v_{jy}) = (w_{jx}, w_{jy})$ to the entire mapping domain. One could impose additional constraints on $\mathbf{t}(x, y)$ to make the solution unique. At first glance, an obvious constraint would be to demand that $\mathbf{t}(x, y)$ be conformal. However, conformality adds two constraints to the problem in the form of the Cauchy-Riemann equations; hence, together with the constraint of satisfying equation (5), the problem of finding $\mathbf{t}(x, y)$ would be overdetermined (Gastner and Newman, 2004). Instead of demanding strict conformality, Tobler (1973) proposed that deviations from conformality should, at least, be minimised when constructing cartograms. However, he reported that a computer program, designed to find nearly conformal density-equalising map

projections, failed to converge. Furthermore, it is not evident that a conformal map projection $\mathbf{t}(x, y)$ necessarily generates desirable properties for a pycnophylactic density $\rho(x, y)$, calculated using equation (7). Therefore, we describe a method that is guaranteed to find a density-equalising map projection $\mathbf{t}(x, y)$ and a smooth pycnophylactic density $\rho(x, y)$, even if neither $\mathbf{t}(x, y)$ nor $\rho(x, y)$ satisfy predefined criteria of optimality.

3 Obtaining a Pycnophylactic Interpolation from a Density-Equalising Map Projection

Gastner et al. (2018) introduced a flow-based algorithm that generates density-equalising map projections. In this algorithm, $\rho(x, y)$ in equation (5) is treated as the initial density of a fluid. The algorithm proceeds by constructing a velocity field that conserves the mass of the fluid, is free of vortices and equilibrates the density over time. By integrating the velocity, the algorithm determines the final displacement $\mathbf{t}(x, y)$ of any arbitrary point that is initially at position (x, y) . It can be shown that $\mathbf{t}(x, y)$ satisfies equation (5); thus, $\mathbf{t}(x, y)$ is a density-equalising map projection.

For the boundary conditions chosen by Gastner et al. (2018), $\mathbf{t}(x, y)$ can be calculated efficiently by applying a Fourier transform to the residual density $\rho_{\text{res}}(x, y)$. Suppose that $\rho(x, y)$ is initially chosen to be the piecewise constant function ρ_{plateau} given by equation (2) on the unprojected map, where $\det J_t = 1 \forall (x, y)$. Hence, the initial residual density can be expressed as follows:

$$\rho_{\text{res},1}(x, y) = \begin{cases} P_i / A_i - \bar{\rho} & \text{if } (x, y) \in U_i, \\ 0 & \text{if } (x, y) \notin \bigcup_i U_i, \end{cases} \tag{8}$$

where $\bar{\rho}$ is the spatial average, defined in equation (3). Figure 2(a) shows $\rho_{\text{res},1}(x, y)$, using COVID-19 cases in Croatia as an example. Because $\rho_{\text{res},1}(x, y)$ has discontinuities at the boundaries of enumeration units, the Fourier transform of $\rho_{\text{res},1}(x, y)$ exhibits the Gibbs phenomenon (Carslaw 1925); that is, the approximation of $\rho_{\text{res},1}(x, y)$ by a finite Fourier series exhibits large oscillations at the boundaries. These oscillations can cause numerical artefacts in subsequent calculations. To circumvent this problem, Gastner’s (2022) computer implementation of the flow-based algorithm removes discontinuities from $\rho_{\text{res},1}(x, y)$ by applying Gaussian smoothing as a low-pass filter. Figure 2(b) illustrates the effect of Gaussian smoothing. The density after Gaussian smoothing can be calculated rapidly in Fourier space. However, Gaussian smoothing tends

Toblerov (1979a) algoritam postao je standardna tehnika za glatku piknofilaktičku interpolaciju. Algoritam je primijenjen u R-u (Brunsdon 2014) i Pythonu (Pysal Developers 2021). Za korisnike softvera za GIS algoritam je također dostupan preko proširenja ArcGIS-a (Qiu i dr. 2012) i GRASS-a (Metz 2013). Varijantu Toblerova (1979a) algoritma razvio je Rase (2001), pri čemu je pravilna kvadratna mreža zamijenjena neregularnom mrežom trokuta. Međutim, Raseov algoritam (2001) i kasnija profinjenja istog autora (Rase 2007) zadržala su bitna svojstva Toblerova (1979a) algoritma: iterativno lokalno usrednjavanje i naknadna preraspodjela razlika u gustoći kako bi se učvrstio piknofilaktički uvjet (1). Unatoč raširenoj upotrebi Toblerovog (1979a) algoritma, to nije bila prva metoda koju je on predložio za glatku piknofilaktičku interpolaciju. U ranijoj publikaciji, Tobler (1976) je opisao alternativni pristup u kojem se granice popisnih jedinica pretvaraju u površinski kartogram (tj. kartu u kojoj su sve popisne jedinice prikazane s površinom proporcionalnom njihovoj populaciji). Predložena Toblerova metoda (1976) zahtijeva da površinski kartogram područja bude susjedan; odnosno susjedne popisne jedinice na Zemljinoj površini moraju biti susjedi na kartogramu. Kao što je primijetio Tobler (2017), kartogrami susjedstva usko su povezani s kartografskim projekcijama s ujednačavanjem gustoće. U ovom članku ukratko opisujemo veze između kartograma susjedstva, kartografskih projekcija s ujednačavanjem gustoće i glatke piknofilaktičke interpolacije. Nakon toga objašnjavamo kako postići glatku piknofilaktičku interpolaciju koristeći nedavno razvijeni algoritam koji generira kartogram susjedstva.

2. Odnos između piknofilaktičke interpolacije i kartografske projekcije s ujednačavanjem gustoće

Da bi se konstruirao kartogram susjedstva granice popisnih jedinica se modeliraju kao polilinije. Označimo vrhove polilinija u nekoj ekvivalentnoj projekciji s $(v_{1x}, v_{1y}), (v_{2x}, v_{2y}), \dots$. Vrhovi se onda pomaknu u nove položaje $(w_{1x}, w_{1y}), (w_{2x}, w_{2y}), \dots$ tako da regije razgraničene transformiranim granicama imaju površinu proporcionalnu populaciji odgovarajućih popisnih jedinica. Ta transformacija može se doživjeti kao kartografska projekcija $\mathbf{t} = (t_x, t_y)$ koja zadovoljava uvjete $t_x(v_{jx}, v_{jy}) = w_{jx}$ i $t_y(v_{jx}, v_{jy}) = w_{jy}$ za svaki $j = 1, 2, \dots$. Kartografska projekcija s takvim svojstvom može se dobiti, na primjer, rješavanjem jednadžbe:

$$\det J_{\mathbf{t}}(x, y) = \frac{\rho(x, y)}{\bar{\rho}}, \quad (5)$$

gdje je $\det J_{\mathbf{t}} = ((\partial t_x)/\partial x)((\partial t_y)/\partial y) - ((\partial t_y)/\partial x)((\partial t_x)/\partial y)$ Jacobijeva determinanta od \mathbf{t} , $\rho(x, y)$ je piknofilaktička funkcija [zadovoljava jednadžbu (1)] i $\bar{\rho}$ je njezina prostorna sredina dana jednadžbom (3). Veličina

$$\rho_{\text{res}}(x, y) = \rho(x, y) - \bar{\rho} \det J_{\mathbf{t}}(x, y) \quad (6)$$

može se interpretirati kao "rezidual gustoće" (tj. razlika između srednje gustoće koja ostaje neobjašnjena projekcijom \mathbf{t}), a cilj je naći rješenje \mathbf{t} tako da bude $\rho_{\text{res}}(x, y) = 0$ za sve (x, y) .

Kartogrami susjedstva i kartografske projekcije koje ujednačuju gustoću međusobno se odnose na dva načina. Prvo, ako su ρ i (v_{jx}, v_{jy}) poznati, moguće je riješiti jednadžbu (5) da se dobije $\mathbf{t}(x, y)$ i onda dobiti vrhove polilinije (w_{jx}, w_{jy}) kartograma susjedstva primjenjujući $\mathbf{t}(x, y)$ na (v_{jx}, v_{jy}) za sve $j = 1, 2, \dots$. Drugo, ako (v_{jx}, v_{jy}) i (w_{jx}, w_{jy}) nisu poznati, moguće je konstruirati kartografsku projekciju s ujednačenom gustoćom $\mathbf{t}(x, y)$ sa svojstvom $\mathbf{t}(v_{jx}, v_{jy}) = (w_{jx}, w_{jy})$ i onda dobiti piknofilaktičku funkciju invertiranjem jednadžbe (5):

$$\rho(x, y) = \bar{\rho} \det J_{\mathbf{t}}(x, y). \quad (7)$$

U oba slučaja rješenje nije jedinstveno. Ako je zadano $\rho(x, y)$, jednadžba (5) daje beskonačno mnogo rješenja $\mathbf{t}(x, y)$ jer je broj uvjeta sadržanih u jednadžbi (5) jednak 1, što je manje od dimenzija karte (tj. 2). Obratno, ako su zadani svi (w_{jx}, w_{jy}) , tada postoji beskonačno mnogo načina za proširenje funkcije \mathbf{t} sa svojstvom $\mathbf{t}(v_{jx}, v_{jy}) = (w_{jx}, w_{jy})$ na cijelom području definicije preslikavanja. Da bi se dobilo jedinstveno rješenje mogu se na $\mathbf{t}(x, y)$ postaviti dodatni uvjeti. Na prvi pogled, jedan očiti uvjet mogao bi biti zahtjev da je preslikavanje $\mathbf{t}(x, y)$ konformno. Međutim, konformnost dodaje dva uvjeta u obliku Cauchy-Riemannovih jednadžbi; zbog toga, zajedno s uvjetom zadovoljavanja jednadžbe (5), problem pronalaženja $\mathbf{t}(x, y)$ bio bi preodređen (Gastner i Newman 2004). Umjesto traženja striktno konformnosti, Tobler (1973) je predložio da odstupanja od konformnosti trebaju barem biti minimizirana pri konstrukciji kartograma. Međutim, on je izvijestio da računalni program oblikovan za pronalaženje skoro konformnih projekcija s ujednačenjem gustoće ne konvergira. Nadalje, nije očito da konformna projekcija $\mathbf{t}(x, y)$ nužno generira poželjna svojstva piknofilaktičke funkcije $\rho(x, y)$, izračunane s pomoću jednadžbe (7). Dakle, mi opisujemo metodu koja garantira pronalaženje projekcije $\mathbf{t}(x, y)$ koja ujednačuje gustoće i glatku piknofilaktičku funkciju $\rho(x, y)$, čak i ako ni $\mathbf{t}(x, y)$ ni $\rho(x, y)$ ne zadovoljavaju unaprijed definiran kriterij optimalnosti.

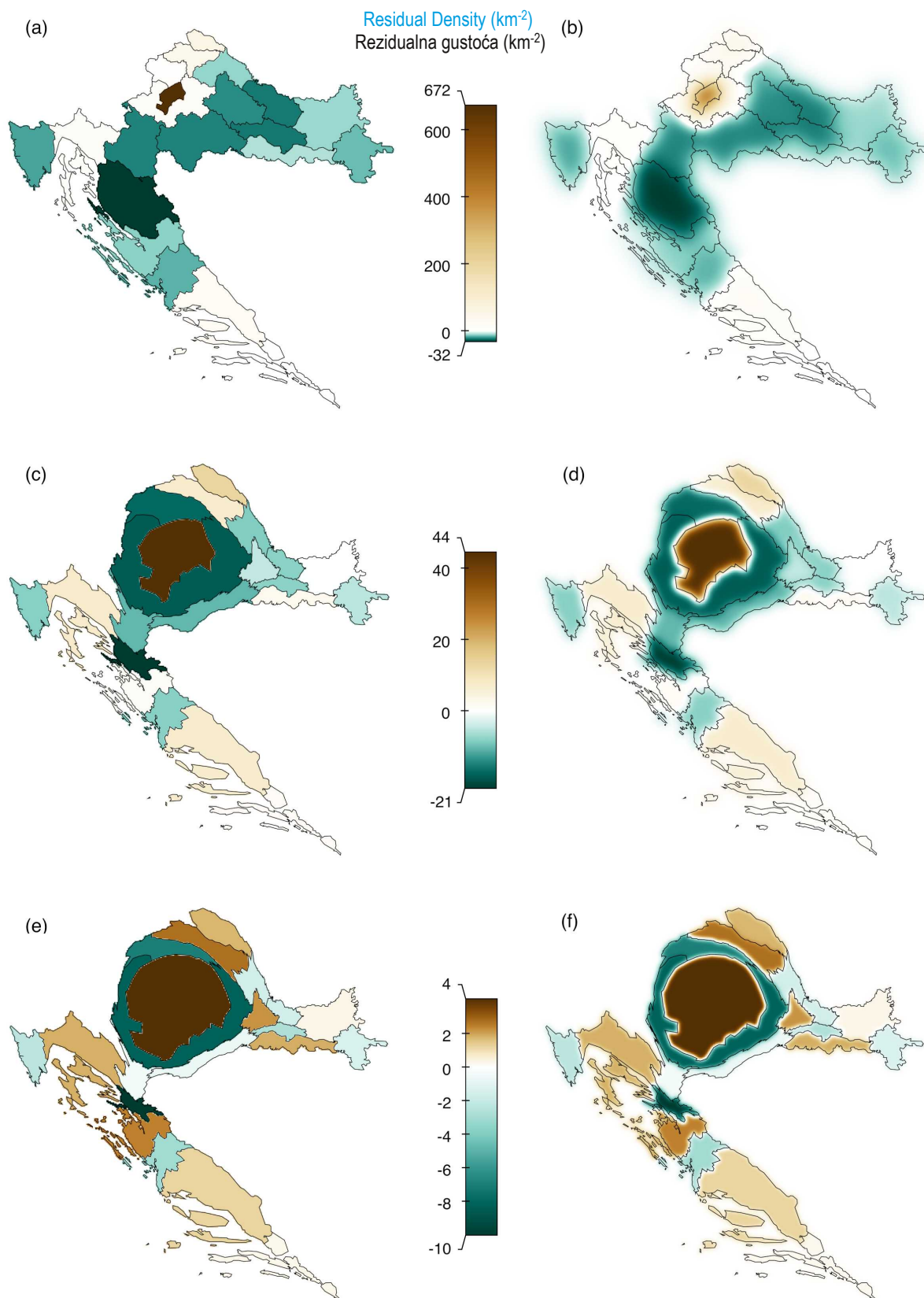


Fig. 2 Steps in the calculation of a pycnophylactic density of COVID-19 cases in Croatia. Starting from an equal-area map in (a), a contiguous cartogram is iteratively calculated in (b) to (f). Panels (b),(d) and (f) show the residual density after Gaussian smoothing of the densities shown in (a), (c) and (e), respectively.

Slika 2. Koraci pri računanju piknofilaktičke funkcije za slučajeve bolesti COVID-19 u Hrvatskoj. Počevši s kartom u ekvidistantnoj projekciji (a), kartogrami susjedstva se iterativno računaju (b) – (f). Slike (b),(d) i (f) pokazuju rezidualnu gustoću nakon Gaussova izgladivanja gustoća prikazanih na slikama (a), (c) i (e).

3. Dobivanje piknofilaktičke interpolacije iz projekcije koja ujednačuje gustoću

Gastner i dr. (2018) uveli su algoritam utemeljen na toku koji generira projekcije koje ujednačuju gustoću. U tom algoritmu, $\rho(x, y)$ u jednadžbi (5) tretira se kao početna gustoća fluida. Algoritam nastavlja s konstrukcijom polja brzina koje čuva masu fluida, bez vrtloga je i uravnotežuje gustoću tijekom vremena. Integriranjem brzine algoritam određuje konačni pomak $\mathbf{t}(x, y)$ proizvoljne točke koja je u početnom položaju (x, y) . Može se pokazati da $\mathbf{t}(x, y)$ zadovoljava jednadžbu (5); dakle $\mathbf{t}(x, y)$ je projekcija koja ujednačuje gustoću.

Za rubne uvjete koje su izabrali Gastner i dr. (2018), $\mathbf{t}(x, y)$ se može učinkovito izračunati primjenom Fourierove transformacije na rezidualnu gustoću $\rho_{\text{res}}(x, y)$. Pretpostavimo da je $\rho(x, y)$ početno odabrana kao po dijelovima konstantna funkcija ρ_{plateau} zadana jednadžbom (2) na neprojeciranoj karti, gdje je $\det J_t = 1, \forall(x, y)$.

$$\rho_{\text{res},1}(x, y) = \begin{cases} P_i / A_i - \bar{\rho} & \text{ako je } (x, y) \in U_i, \\ 0 & \text{ako je } (x, y) \notin \bigcup_i U_i, \end{cases} \quad (8)$$

Zato se početna rezidualna gustoća može izraziti ovako: gdje je ρ prostorna sredina definirana jednadžbom (3). Slika 2(a) pokazuje primjer $\rho_{\text{res},1}(x, y)$ za slučajeve bolesti COVID-19 u Hrvatskoj. Budući da $\rho_{\text{res},1}(x, y)$ ima diskontinuitete na granicama popisnih jedinica, Fourierova transformacija od $\rho_{\text{res},1}(x, y)$ pokazuje Gibbsov fenomen (Carslaw 1925); tj. aproksimacija od $\rho_{\text{res},1}(x, y)$ primjenom konačnih Fourierovih redova daje velike oscilacije na granicama. Te oscilacije mogu prouzročiti numeričke artefakte pri računanjima koja slijede. Da bi se to izbjeglo, Gastnerova (2022) računalna primjena algoritma utemeljenog na toku uklanja diskontinuitete od $\rho_{\text{res},1}(x, y)$ primjenjujući Gaussovo izgladivanje kao niskopropusni filter. Slika 2(b) ilustrira učinak Gaussovog izgladivanja. Nakon Gaussova izgladivanja gustoća se može brzo izračunati u Fourierovu prostoru. Međutim, Gaussovo izgladivanje ima tendenciju pomaka gustoće prema slabije popunjenim područjima; dakle, ono ne daje piknofilaktičku funkciju. Prema tome, izračunana projekcija $\mathbf{t}_1(x, y)$ ne ujednačuje gustoću. Bez obzira na to, projiciranjem granica, gusto naseljene popisne jedinice imaju tendenciju širenja, a rijetko naseljene popisne jedinice imaju tendenciju smanjivanja, kao što pokazuju granice županija na slici 2(c). Uočimo da algoritam nadopunjuje gustoću koja curi iz unutrašnjosti Hrvatske na slici 2(b) prije nego što izračuna zaostalu gustoću na slici 2(c).

Kako bi poboljšao ujednačavanje gustoće, Gastnerov računalni program (2022) umeće granice dobivene

iz početnog pokretanja algoritma temeljenog na protoku kao ulaz u drugu iteraciju; tj. konstruirana je nova po dijelovima konstantna rezidualna gustoća $\rho_{\text{res},2}(x, y)$; Gaussovo izgladivanje je primijenjeno na $\rho_{\text{res},2}(x, y)$, kao što je prikazano na slici 2(d); i izračuna se nova projekcija $\mathbf{t}_2(x, y)$. Nove granice, prikazane na slici 2(e), mogu se ponovno staviti u algoritam, koji ponovno računa po dijelovima konstantnu rezidualnu gustoću $\rho_{\text{res},3}(x, y)$ prije primjene Gaussova izgladivanja, kako je prikazano na slici 2(f). Sa svakom iteracijom k , diskontinuiteti od $\rho_{\text{res},k}(x, y)$ postaju sve manji. Dakle, širina Gaussove jezgre koja se koristi za izgladivanje može se nakon svake iteracije smanjiti sve dok širina ne postane nerazlučiva od nule. Trake u boji na slici 2 pokazuju da se zaostala gustoća približava nuli kako se postupak ponavlja.

Označimo li projekciju izračunanu u k -toj iteraciji algoritma utemeljenog na toku s $\mathbf{t}_k(x, y)$, rješenje jednadžbe (2) može se dobiti kao kompozicija funkcija $\mathbf{t}(x, y) = \mathbf{t}_l \circ \mathbf{t}_{l-1} \circ \dots \circ \mathbf{t}_1(x, y)$, uz pretpostavku da je l dovoljno velik. U praksi su vrijednosti l između 5 i 20 obično dovoljne da površine svih popisnih jedinica na kartogramu budu točne na 1%. Složena funkcija $\mathbf{t}(x, y)$ je diferencijabilna na granicama popisnih jedinica jer je svaka projekcija $\mathbf{t}_k(x, y)$ rezultat Gaussove konvolucije i stoga diferencijabilan (Gwosdek i dr. 2012). Računalni program aproksimira Jacobijevu determinantu $\det J_t$ kao faktor s pomoću kojeg površine fine kvadratne mreža povećavaju ($\det J_t > 1$) ili smanjuju ($\det J_t < 1$) zbog $\mathbf{t}(x, y)$. U slučaju Hrvatske program upotrebljava mrežu od 512 horizontalnih i 512 vertikalnih linija. Nakon toga, $\rho(x, y)$ se dobije rješavanjem jednadžbe (7) što rezultira aproksimacijom diferencijabilne piknofilaktičke funkcije. Slika 1(b) pokazuje rezultat slučajeve bolesti COVID-19 za Hrvatsku.

4. Zaključak

Gastnerov program (2022) napisan je u C++ i optimiran za brzinu računanja. Za kartu prikazanu na slici 1(b), računanje je trajalo oko 5,1 sekundi za testiranje na laptopu MacBook Pro s procesorom 2.7 GHz Quad-Core Intel i7. Potvrđujemo da bi usporedba s Toblerovim (1979a) algoritmom zahtijevala pažljivu usporedbu vremena izvođenja, što planiramo napraviti. Međutim, naše preliminarne studije sugeriraju da naša metoda zahtijeva manje ponavljanja nego Toblerov (1979a) algoritam. To opažanje objašnjava se činjenicom da se u algoritmu koji se temelji na kartogramu gustoća povećava s bilo kojom točkom u prostoru već mijenja nakon prve iteracije, čak i za točke udaljene od bilo koje granice, zbog značajne širine početne Gaussove jezgre. Nasuprot tome, u Toblerovom (1979a) algoritmu točka mreže koja je udaljena m razmaka mreže od najbliže

to shift density towards sparsely populated regions; thus, it does not produce a pycnophylactic density. Consequently, the calculated map projection $\mathbf{t}_1(x, y)$ is not density-equalising. Nevertheless, by projecting the boundaries, densely populated enumeration units tend to expand, and sparsely populated enumeration units tend to shrink, as shown by the county borders in Figure 2(c). We note that the algorithm replenishes the density that leaks out of the interior of Croatia in Figure 2(b) before calculating the residual density in Figure 2(c).

To improve density-equalisation, the computer program by Gastner (2022) inserts the boundaries obtained from the initial run of the flow-based algorithm as input into a second run; that is, a new piecewise constant residual density $\rho_{\text{res},2}(x, y)$ is constructed; Gaussian smoothing is applied to $\rho_{\text{res},2}(x, y)$, as indicated in Figure 2(d); and a new projection $\mathbf{t}_2(x, y)$ is calculated. The newly projected boundaries, shown in Figure 2(e), can be reinserted into the algorithm, which again calculates a piecewise constant residual density $\rho_{\text{res},3}(x, y)$ before applying Gaussian smoothing, as shown in Figure 2(f). With each iteration k , discontinuities of $\rho_{\text{res},k}(x, y)$ tend to become smaller. Thus, the width of the Gaussian kernel that is used for smoothing can be made smaller after each iteration until the width becomes indistinguishable from zero. The colour bars in Figure 2 show that the residual density converges towards zero as the procedure is repeated.

Denoting the projections calculated in the k -th iteration of the flow-based algorithm by $\mathbf{t}_k(x, y)$, a solution to equation (2) can be obtained by the function composition $\mathbf{t}(x, y) = \mathbf{t}_l \circ \mathbf{t}_{l-1} \circ \dots \circ \mathbf{t}_1(x, y)$, provided that l is sufficiently large. In practice, values of l between 5 and 20 are usually sufficient to make the areas of all enumeration units on the cartogram accurate to within 1%. The composed projection $\mathbf{t}(x, y)$ is differentiable at the boundaries of the enumeration units because each projection $\mathbf{t}_k(x, y)$ is the result of Gaussian convolution and, hence, differentiable (Gwosdek et al. 2012). The computer program approximates the Jacobian determinant $\det J_{\mathbf{t}}$ as the factor by which the areas of cells in a fine-grained square grid increase ($\det J_{\mathbf{t}} > 1$) or decrease ($\det J_{\mathbf{t}} < 1$) because of $\mathbf{t}(x, y)$. In the case of Croatia, the program uses a grid with 512 horizontal and 512 vertical lines. Afterwards, equation (7) is used to solve for $\rho(x, y)$, which results in a finite-size approximation of a differentiable pycnophylactic density. Figure 1(b) shows the result for COVID-19 cases in Croatia.

4 Conclusion

The program by Gastner (2022) is written in C++ and optimised for computational speed. For the map shown

in Figure 1(b), the calculation needed only an average time of 5.1 seconds for a test run on a MacBook Pro laptop with a 2.7 GHz Quad-Core Intel i7 processor. We acknowledge that a comparison with Tobler's (1979a) algorithm would require careful benchmarking of run times, which we still need to implement. However, our preliminary studies suggest that the method described above requires fewer iterations than Tobler's (1979a) algorithm. This observation is explained by the fact that, in the cartogram-based algorithm outlined above, the density associated with any point in space already changes after the first iteration, even for points far from any boundary, because of the significant width of the initial Gaussian kernel. By contrast, in Tobler's (1979a) algorithm, a grid point that is m grid spacings away from the nearest boundary requires $O(m)$ iterations until its density is affected by a neighbouring enumeration unit. This advantage of the cartogram-based algorithm is partly offset by the time needed to calculate Fourier transforms. However, we hypothesise that the substantial reduction in the number of overall iterations more than compensates for the cost of the Fourier transforms. As Tobler (1979b) himself noted, 'The use of a fast Fourier transform ... should be investigated to hasten convergence of the algorithm.'

We also acknowledge that pycnophylactic densities generated by the algorithm outlined above do not optimise any predefined criterion for smoothness, whereas Tobler's (1979a) algorithm directly minimises $\iint [\partial^2 \rho / \partial x^2 + \partial^2 \rho / \partial y^2]^2 dx dy$. However, the density ρ obtained from our algorithm could be used as input to Tobler's (1979a) algorithm, thereby combining the benefits of both methods. The potential advantages of combining the algorithms will be investigated in future work.

Acknowledgements

This work was supported by the Singapore Ministry of Education (AcRF Tier 1 Grant IG18- PRB104, R-607-000-401-114), a Yale-NUS College research award (through grant number A-0000177-00-00), and the Yale-NUS Summer Research Programme. We would like to thank Editage (www.editage.com) for English language editing. We are grateful to Adam Tonks and two anonymous reviewers for their insightful comments and suggestions.

Conflicts of interest

All authors declare that they have no conflicts of interest.

granice zahtijeva $O(m)$ iteracija sve dok na njezinu gustoću ne utječe susjedna popisna jedinica.

Ova prednost algoritma zasnovanog na kartogramu djelomično je nadoknađena vremenom potrebnim za izračunavanje Fourierovih transformacija. Međutim, pretpostavljamo da značajno smanjenje broja ukupnih iteracija više nego kompenzira trošak Fourierovih transformacija. Kao što je sam Tobler (1979b) primijetio: "Upotrebu brze Fourierove transformacije... treba istražiti kako bi se ubrzala konvergencija algoritma."

Također potvrđujemo da piknofilaktike generirane našim algoritmom ne optimiraju nijedan unaprijed definirani kriterij za glatkoću, dok Toblerov (1979a) algoritam izravno minimizira $\iint [(\partial^2\rho/\partial x^2 + \partial^2\rho/(\partial y^2)^2)]dx dy$. Međutim, gustoća ρ dobivena iz našeg algoritma mogla bi se koristiti kao ulaz u Toblerov (1979a) algoritam, kombinirajući na taj način

prednosti obje metode. Potencijalne prednosti kombiniranja algoritama istraživat će se u budućem radu.

Zahvala

Ovaj rad podržalo je Ministarstvo obrazovanja Singapura (AcRF Tier 1 Grant IG18-PRB104, R-607-000-401-114), istraživačka nagrada Yale-NUS Collegea (kroz grant broj A-0000177-00-00), i Yale-NUS Summer Research Program. Zahvaljujemo Editageu (www.editage.com) za uređivanje engleskog jezika. Zahvaljujemo Adamu Tonksu i dvojici anonimnih recenzenata na njihovim pronicljivim komentarima i prijedlozima.

Sukob interesa

Svi autori izjavljuju da nemaju sukoba interesa.

Literatura / References

- Brunsdon C (2014) pycno: Pycnophylactic Interpolation. R package version 1.2. Available via CRAN. <https://CRAN.R-project.org/package=pycno>. Accessed 6 Feb 2022
- Carslaw H (1925) Gibbs' phenomenon in Fourier's integrals. *Nature*, vol. 116, no. 2193, 312–313. doi: 10.1038/116312c0
- Croatian Institute of Public Health (2022) Službena stranica Vlade za pravodobne i točne informacije o koronavirusu. Available via <https://www.koronavirus.hr/>. Accessed 24 Mar 2022
- Gastner MT, Newman MEJ (2004) Diffusion-based method for producing density-equalizing maps. *Proceedings of the National Academy of Sciences*, vol. 101, no. 20, 7499–7504. doi: 10.1073/pnas.0400280101
- Gastner MT, Seguy V, More P (2018) Fast flow-based algorithm for creating density-equalizing map projections. *Proceedings of the National Academy of Sciences*, vol. 115, no. 10, E2156–E2164. doi: 10.1073/pnas.1712674115
- Gastner MT (2022) cartogram-cpp [GitHub repository]. doi: 10.5281/zenodo.6346715
- Gwosdek P, Grewenig S, Bruhn A, Weickert J (2012). Theoretical Foundations of Gaussian Convolution by Extended Box Filtering. In: Bruckstein AM, ter Haar Romeny BM, Bronstein AM, Bronstein MM (eds) *Scale Space and Variational Methods in Computer Vision*. SSVM 2011. *Lecture Notes in Computer Science*, vol. 6667, Springer, Berlin, p 447–458. doi: 10.1007/978-3-642-24785-9_38
- Metz M (2013) v.surf.mass - GRASS GIS manual. Available via GRASS reference manual. <https://grass.osgeo.org/grass78/manuals/addons/v.surf.mass.html>. Accessed 6 Feb 2022
- Openshaw S (1984) Ecological fallacies and the analysis of areal census data. *Environment and Planning A*, vol. 16, no. 1, 17–31. doi: 10.1068/a160017
- PySAL Developers (2021) Tobler. Release v0.8.2. Available via GitHub. <https://github.com/pysal/tobler>. Accessed 6 Feb 2022
- Qiu F, Zhang C, Zhou Y (2012) The development of an areal interpolation ArcGIS extension and a comparative study. *GIScience & Remote Sensing*, vol. 49, no. 5, 644–663. doi: 10.2747/1548-1603.49.5.644
- Rase W-D (2001) Volume-preserving interpolation of a smooth surface from polygon-related data. *Journal of Geographical Systems*, vol. 3, no. 2, 199–213. doi: 10.1007/PL00011475
- Rase W-D (2007) Volumenerhaltende Interpolation aus polygonbezogenen Daten in einem unregelmäßigen Dreiecksnetz (TIN). In: Strobl J, Blaschke T, Griesebner G (eds) *Angewandte Geoinformatik 2007. Beiträge zum 19. AGIT-Symposium Salzburg, Wichmann, Heidelberg*, p 595–604
- Tobler WR (1973) A continuous transformation useful for districting. *Annals of the New York Academy of Sciences*, vol. 219, no. 1, 215–220. doi: 10.1111/j.1749-6632.1973.tb41401.x
- Tobler WR (1976) Cartograms and Cartosplines. In: U.S. Department of Health, Education, and Welfare (eds) *Proceedings, Workshop on Automated Cartography and Epidemiology, Arlington (Virginia), USA, 18–19 Mar 1976*, p 53–58
- Tobler WR (1979a) Smooth pycnophylactic interpolation for geographical regions. *Journal of the American Statistical Association*, vol. 74, no. 367, 519–530. doi: 10.1080/01621459.1979.10481647
- Tobler WR (1979b) Rejoinder. *Journal of the American Statistical Association*, vol. 74, no. 367, 535–536. doi: 10.1080/01621459.1979.10481649
- Tobler WR (2017) Cartograms as Map Projections. In: Lapaine M, Uesry EL (eds) *Choosing a Map Projection*, Springer, Cham, p 149–159. doi: 10.1007/978-3-319-51835-0