

Area-to-Point Kernel Regression on Streaming Data

Alexei Pozdnoukhov
National Centre for Geocomputation
National University of Ireland Maynooth
Maynooth, Co. Kildare, Ireland
Alexei.Pozdnoukhov@nuim.ie

Christian Kaiser
National Centre for Geocomputation
National University of Ireland Maynooth
Maynooth, Co. Kildare, Ireland
Christian.Kaiser@nuim.ie

ABSTRACT

Spatial data streams are often referenced to an areal spatial unit such as a polygon rather than to a precise point location. This is the case when geo-referencing is done by user IP addresses or from a mobile phone cell ID in various location-based service applications. One problem of interest in this case is spatial modelling of various spatially continuous quantities, such as an intensity of the usage of particular service in the area. This paper investigates a machine learning framework that account for area-to-point data processing. The approach is based on so-called vicinal risk minimization principle. It is elaborated in detail for a class of kernel recursive algorithms developed for distributed processing of streaming data. Concrete examples of kernel computations are provided and the method performance is investigated experimentally.

Categories and Subject Descriptors

H.2.8 [Database Management]: Database Applications—*data mining, mining methods and algorithms, interactive data exploration and discovery*

General Terms

ALGORITHMS

Keywords

geostreaming, spatial statistics, machine learning

1. INTRODUCTION

Location aware applications and location-based services operate with data feeds that have a geographic element. This data can be exploited in various mashup applications which process such feeds and provide web maps or services to deliver modelling results for some geographical location in real-time. However, spatial data streams are often aggregated and referenced to an areal spatial unit such as a polygon rather than to a precise point in space. This is the case

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

IWGS'11 workshop at ACM GIS '11 Chicago, IL, USA
Copyright 2011 ACM 978-1-60558-649-6/09/11 ...\$10.00.

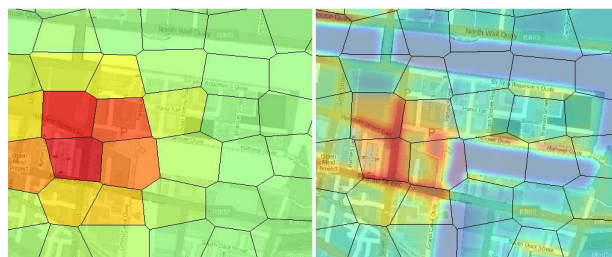


Figure 1: An example of the area-to-point regression: people density interpolation from data aggregated over cell polygons.

when geo-referencing is done by user IP addresses or from a mobile phone cell ID covering some geographical area, or data are initially aggregated within an area or attributed to an extended region due to privacy issues.

In these cases it is important to adapt the processing methods to correctly account for an extended spatial support. A typical example is a spatial interpolation problem when a continuous surface (Figure 1, right) has to be produced from areal data (Figure 1, left) for downscaling, data homogenization and interoperability in further processing, or simply for visualization. Many recent applications have demonstrated interpolated human activity heat-maps from areal data such as mobile phone cells [4], however, dealt with such data in an ad-hoc way with no regard to its areal support at interpolation step. It limits the usefulness of such maps for rigorous high-fidelity spatial analysis and decision making.

Spatial statistics offers various approaches to overcome this problem, including the so-called pycnophylactic interpolation [6] and area-to-point kriging [3]. Surprisingly, there is a theoretical framework that can provide similar functionality for a wider class of methods in machine learning, and kernel methods in particular. It is known as the Vicinal Risk Minimisation principle [7].

We develop this approach for a class of kernel algorithms in Section 2. The baseline method of least squares regression is elaborated in more detail in Section 3. We then derive analytical expressions for Gaussian kernels and Gaussian-like vicinities. Generally, an efficient numerical integration scheme developed in [5] can be used for arbitrary kernels and general polygonal units. This is discussed in Section 4.

Importantly, we implement these ideas in an incremental learning algorithm based on Kernel Recursive Least Squares (KRLS) [1] to apply the derived methods on streaming data.

We further note that a scalability of the method can be enhanced with a MapReduce distributed implementation [2] available as an open source project¹. These findings are investigated experimentally in Section 5.

We conclude this introduction with a brief description of state-of-the-art area-to-point interpolation methods.

1.1 Pycnophylactic Interpolation

Pycnophylactic interpolation is a classical method originally developed for mapping density values of socio-economic attributes in data aggregated over administrative units [6]. Its main idea is to preserve the total density mass such that the interpolated density aggregates over the areal units match the data values collected within those. Tobler proposed to solve this problem as solution to Laplace equation which provides smoothness to $y(u)$, under the constraints to reproduce the mass density in the areal units and Dirichlet or Neumann boundary conditions depending on the prior assumptions or data availability at the borders of the study region.

1.2 Area-to-point Kriging

Area-to-point kriging is a method that makes use of area-to-area and area-to-point covariance models and exploits a standard geostatistical scheme to derive spatial predictions [3]. For a set of K polygonal areal units s_k it introduces sampling functions $g_k(u)$ such that

$$\int_{s_k} g_k(\mathbf{u})y(\mathbf{u})d\mathbf{u} = z_k, \quad k = 1, \dots, K, \quad (1)$$

where the measured areal values are denoted as z_k and the continuous density we are interested to derive is $y(\mathbf{u})$. Area-to-area and area-to-point covariances are then defined as

$$c_{\mathbf{v},k'} = \int_{s_k} g_k(\mathbf{u})c(\mathbf{v} - \mathbf{u})d\mathbf{u}, \quad (2)$$

$$c_{k,k'} = \int \int_{s_k, s_{k'}} g_k(\mathbf{u})g_{k'}(\mathbf{u}')c(\mathbf{u} - \mathbf{u}')d\mathbf{u}d\mathbf{u}' \quad (3)$$

for a given covariance model $c(\mathbf{u} - \mathbf{u}')$. The solution to the kriging prediction at some location \mathbf{v} is then

$$\hat{y}(\mathbf{v}) = \mathbf{Y}^T \mathbf{C}^{-1} \mathbf{c} \quad (4)$$

which we here considered in its dual form to facilitate further presentation. It implies that \mathbf{C} , an area-to-area covariance matrix with entries computed with (3) is positive definite and of full rank. Note that pycnophylactic method can be derived as a particular case of the latter kriging approach given specific covariance model [3].

2. VICINAL RISK MINIMISATION

Machine learning considers estimating dependencies from empirical data as to find $f(\mathbf{x}) : \mathbf{x} \mapsto y$ from a sufficiently rich set of functions $\{\mathbf{F} = f(\mathbf{x}, \alpha \in \Lambda)\}$, which provides the minimum value of the risk functional

$$\int L(y, f(\mathbf{x}, \alpha))dP(\mathbf{x}, y) \approx \frac{1}{\ell} \sum_{i=1}^{\ell} L(y_i, f(\mathbf{x}_i, \alpha)). \quad (5)$$

Here, $L(y, f(\mathbf{x}, \alpha))$ is the loss function, or a measure of discrepancy between the estimate and the actual value y . The

¹<http://ncg.nuim.ie/i2maps/>

objective thus is to find a model that minimizes the expected average loss for a given problem. The joint distribution function $P(\mathbf{x}, y)$ is considered unknown and only the training data $\{\mathbf{x}_1, y_1, \dots, \mathbf{x}_\ell, y_\ell\}$ are available. This is why the empirical risk is usually minimised, implicitly assuming an empirical distribution $p(\mathbf{x}, y) = \frac{1}{\ell} \sum_{i=1}^{\ell} \delta(\mathbf{x} - \mathbf{x}_i) \delta(y - y_i)$. An idea to use more sophisticated distributions gives rise to Vicinal Risk Minimization principle [7]. Considering local distributions $p(\mathbf{x}|\mathbf{x}_i, r_i)$ instead of delta functions, one obtains the following Vicinal Risk functional:

$$R_{vic}(\alpha) = \frac{1}{\ell} \sum_{i=1}^{\ell} L\left(y_i - \int f(\mathbf{x}, \alpha)p(\mathbf{x}|\mathbf{x}_i, r_i)d\mathbf{x}\right), \quad (6)$$

where \mathbf{x}_i is a training sample and r_i is its vicinity parameter. Minimizing (6) instead of empirical risk is called the Vicinal Risk Minimization (VRM) principle.

3. RECURSIVE KERNEL REGRESSION

Consider the minimization of vicinal risk for the squared loss

$$R_{vic}(f) = \frac{1}{\ell} \sum_{i=1}^{\ell} \left(y_i - \int f(\mathbf{x})p(\mathbf{x}|\mathbf{x}_i, r_i)d\mathbf{x}\right)^2, \quad (7)$$

where r_i is a parameter vector defining the vicinity of a sample \mathbf{x}_i . We are going to find the solution to this problem in a form of kernel expansion

$$f(\mathbf{x}) = \sum_{j=1}^{\ell} \alpha_j \int K(\mathbf{x}, \mathbf{x}')p(\mathbf{x}'|\mathbf{x}_j, r_j)d\mathbf{x}', \quad (8)$$

i.e. as a weighted combination of the average responses of each vicinity. Substitution to (7) gives:

$$R_{vic}(f) = \frac{1}{\ell} \sum_{i=1}^{\ell} \left(y_i - \sum_{j=1}^{\ell} \alpha_j m_{ij}\right)^2 \quad (9)$$

where m_{ij} is the values of a so called two-vicinal kernel

$$m_{ij} = \int \int K(\mathbf{x}, \mathbf{x}')p(\mathbf{x}'|\mathbf{x}_j, r_j)p(\mathbf{x}|\mathbf{x}_i, r_i)d\mathbf{x}d\mathbf{x}'. \quad (10)$$

In matrix notation, with \mathbf{Y} as a vector of training values y_i and \mathbf{M} is an $\ell \times \ell$ matrix with elements m_{ij} , (9) is simply the $\|\mathbf{Y} - \mathbf{M}\alpha\|^2$ and hence is minimized by $\alpha = \mathbf{Y}^T \mathbf{M}^{-1}$. The predictions are then made with

$$\mathbf{f} = \mathbf{Y}^T \mathbf{M}^{-1} \mathbf{L} \quad (11)$$

where \mathbf{L} is an $\ell \times 1$ vector of one-vicinal kernels (refer to Eq. (8)).

Comparing Eqs. (11) and (4) one can see that in the particular case of one-vicinal models (8) the relation between the dual form of kriging and kernel methods also holds for the areal data problems.

3.1 Why areal aggregates are preserved?

Our initial motivation was to develop an area-to-point interpolation method that preserves areal aggregates. Let us compute an average values of the predictive model (8) over k^{th} area, $E_k[f(\mathbf{x})]$. It is given by

$$\begin{aligned} E_k[f(\mathbf{x})] &= \int \sum_{j=1}^{\ell} \alpha_j \int K(\mathbf{x}, \mathbf{x}')p(\mathbf{x}'|\mathbf{x}_j, r_j)p(\mathbf{x}|\mathbf{x}_k, r_k)d\mathbf{x}'d\mathbf{x} \\ &= \sum_{j=1}^{\ell} \alpha_j m_{kj} = \alpha \mathbf{m}_k = \mathbf{Y}_k, \end{aligned} \quad (12)$$

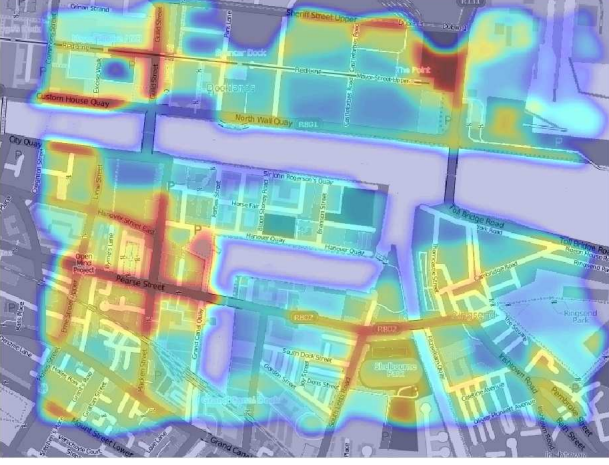


Figure 2: Modelled people density as sensed from a simulated cell phone network coverage and prior probability of space occupancy.

as directly follows from coefficient equation $\alpha = \mathbf{Y}^T \mathbf{M}^{-1}$ thus confirming pycnophylactic properties of the model.

3.2 Sparse Incremental Training

An inverse of the kernel matrix \mathbf{M}^{-1} can be computed incrementally in $O(\ell^2)$ at each step with a single pass over data, giving rise to incremental update equations for the set of coefficients $\alpha = \{\alpha_i\}_{i=1,\ell}$. This is the key idea from kernel recursive least squares regression [1] that we adopt to apply the method for streaming data. The second important idea is a sparsification of entries in \mathbf{M} and the corresponding α (the *dictionary*) using a so-called Approximate Linear Dependence (ALD) test [1]. For every new incoming sample $\mathbf{x}_{\ell+1}$ in a stream one computes kernel values $\mathbf{m}_{\ell+1} = \{m(\mathbf{x}_i, \mathbf{x}_{\ell+1})\}_{i=1,\ell}$ and checks if it can be represented by the samples contained in dictionary and the current model α with sufficient precision, defined by a user-selected threshold η . The ALD test computation is:

$$\delta = m(\mathbf{x}_{\ell+1}, \mathbf{x}_{\ell+1}) - \mathbf{m}_{\ell+1}^T \mathbf{a}, \quad \mathbf{a} = \mathbf{M}_{\ell}^{-1} \mathbf{m}_{\ell+1}. \quad (13)$$

If $\delta < \eta$, the sample is added to the dictionary, the \mathbf{M}^{-1} extension and an update of α is computed as:

$$\mathbf{M}_{\ell+1}^{-1} = \frac{1}{\delta_{\ell}} \begin{bmatrix} \delta_{\ell} \mathbf{M}_{\ell}^{-1} + \mathbf{a} \mathbf{a}^T & -\mathbf{a} \\ -\mathbf{a}^T & 1 \end{bmatrix}, \quad (14)$$

$$\alpha_{\ell+1} = \begin{bmatrix} \alpha_{\ell} - \frac{\mathbf{a}}{\delta_{\ell}} (y_{\ell+1} - \mathbf{m}_{\ell+1}^T \alpha_{\ell}) \\ \frac{1}{\delta_{\ell}} (y_{\ell+1} - \mathbf{m}_{\ell+1}^T \alpha_{\ell}) \end{bmatrix}. \quad (15)$$

Otherwise, $\mathbf{M}_{\ell+1}^{-1} = \mathbf{M}_{\ell}^{-1}$ and coefficients updates are:

$$\alpha_{\ell+1} = \alpha_{\ell} + \mathbf{M}_{\ell}^{-1} \frac{\mathbf{P}_{\ell}}{1 + \mathbf{a}^T \mathbf{P}_{\ell} \mathbf{a}} (y_{\ell+1} - \mathbf{m}_{\ell+1}^T \alpha_{\ell}), \quad (16)$$

$$\mathbf{P}_{\ell} = \mathbf{P}_{\ell-1} - \frac{\mathbf{P}_{\ell-1} \mathbf{a} \mathbf{a}^T \mathbf{P}_{\ell-1}}{1 + \mathbf{a}^T \mathbf{P}_{\ell-1} \mathbf{a}}. \quad (17)$$

The ‘‘oldest’’ or ‘‘least relevant’’ samples can be eliminated from the current dictionary at the cost of $O(\ell^2)$ in order to constrain the dictionary size. It involves simple matrix operations of ‘‘downsizing’’ \mathbf{M} and \mathbf{M}^{-1} .

4. KERNEL COMPUTATION

Integration involved in kernel definitions (8) and (10) can be computationally expensive. However, both \mathbf{L} and \mathbf{M} can be pre-computed for a particular sensing infrastructure and prediction locations if vicinity distributions $p(\mathbf{x}|\mathbf{x}_i, r_i)$ can be assumed constant in time. In this case, \mathbf{M} is growing as per Eqs. (14)-(15) and ALD criterion until each areal unit is represented in a dictionary. Thereafter, stream processing only requires incremental updates (16)-(17).

4.1 Soft Vicinities

We refer to soft vicinities when $p(\mathbf{x}|\mathbf{x}_i, r_i)$ is of unbounded support. We assume it is decaying with distance from \mathbf{x}_i at a rate that integrals in (8) and (10) exist. An example is a Gaussian RBF type of vicinities with parameters \mathbf{x}_i, Ω and \mathbf{x}_j, Ω' , for which one- and two-vicinal kernels for a baseline $K(\mathbf{x}, \mathbf{x}') = e^{-\frac{1}{2}(\mathbf{x}-\mathbf{x}')^T |\Sigma|^{-1} (\mathbf{x}-\mathbf{x}')}$ can be computed with:

$$l(\mathbf{x}, \mathbf{x}_i) = \frac{|\Sigma|^{1/2}}{|\Sigma+\Omega|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\mathbf{x}_i)^T (\Sigma+\Omega)^{-1} (\mathbf{x}-\mathbf{x}_i)}, \quad (18)$$

$$m(\mathbf{x}_i, \mathbf{x}_j) = \frac{|\Sigma|^{1/2}}{|\Sigma+\Omega+\Omega'|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}_i-\mathbf{x}_j)^T (\Sigma+\Omega+\Omega')^{-1} (\mathbf{x}_i-\mathbf{x}_j)}. \quad (19)$$

This results in Gaussians with space-varying variances that equalize the prediction density in a sense explained in Section 3.1 above.

4.2 Hard Vicinities

In case of the functions $p(\mathbf{x}|\mathbf{x}_i, r_i)$ with finite support we deal with so-called hard vicinities. This applies when data are attributed to administrative units or regions which can be considered as bounded areas in space and modelled as polygons. A usual approach is to consider the uniform density within a polygon. More complicated vicinity functions can be used if prior knowledge about the sensing system is available, such as a signal propagation model, or a proxy on the spatial densities reflecting the physical or infrastructural constraints and land use.

This approach generally requires numerical integration. An efficient integration scheme for arbitrary 2D polygons was proposed in [5]. It is based on a Gauss-like cubature formula over convex, non-convex or even multiply connected polygons and does not need any pre-processing like triangulation of the domain. We will use this approach in our experiments below. Note that in practice when the sensing infrastructure is fixed most of the pre-processing for efficient integration can be done off-line.

5. A CITY-SENSING APPLICATION

Telecommunication systems with their high penetration into modern society provide huge volumes of streaming data on human activities. Usually the data are available in various forms of spatial aggregates streaming from fixed hardware installations which we call ‘‘sensors’’ for simplicity.

If the only knowledge available on the sensing system is the spatial extent and approximate shapes of data aggregation regions for each sensor, one can encode it with a Gaussian-like sampling function of covariance Ω_i centered at \mathbf{x}_i . The use of soft vicinities scheme described in Section 4.1 is then straightforward.

In case the polygonal spatial areas within which streaming data is collected are known, this is the case of hard vicinities (Section 4.2). One can use various methods to define den-

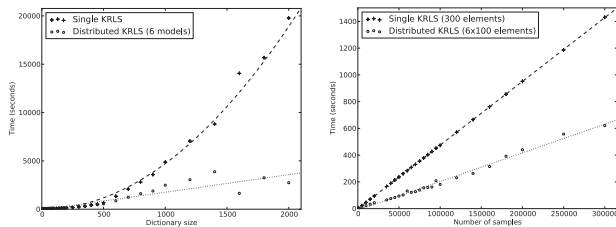


Figure 3: Processing time with respect to the number of entries in a dictionary ℓ and a number of samples processed in a stream.

sity $p(\mathbf{x}|\mathbf{x}_i, r_i)$ within polygons depending on the type of a measurement system at hand:

- $p(\mathbf{x}|\mathbf{x}_i, r_i) \sim \text{const}$ corresponds to simple aggregation over the polygon area. A constant input will be transformed into $1/S_i$, where S_i is the area of the polygon r_i .
- $p(\mathbf{x}|\mathbf{x}_i, r_i) \sim 1/S_i$ corresponds to intensity-type measurements. A constant input will be kept constant.

5.1 Population density sensing

We considered an application when an aggregated number of cell phone users within each cell is known at regular intervals in time and the task is to estimate the real-time dynamics of population density in a city.

People flow and corresponding counts of cell phone connections were simulated for a 2×2 km region for 10000 temporal steps. One hundred cell phone towers (or “femto-cells”, according to the scale of simulation) were distributed within the area. A sample sub-region is shown in Figure 1, left. We used the OpenStreetMap geometries to compute prior occupancy probabilities and enhance spatial fidelity of population estimates. Busy streets and public areas such as squares were assigned with the weight 1, streets in quite residential zones as 0.5 and finally the areas over water bodies as 0.

The baseline kernel we used for this study is an isotropic ($\Sigma = \sigma \mathbf{I}$) Gaussian RBF. Kernel bandwidth was tuned offline using 10-fold cross-validation. The resulting population density estimate obtained with $\sigma = 0.2$ km is presented in Figure 3.2. An animation illustrating the temporal dynamics of the estimates from streaming data is available online².

5.2 Scaling properties of streaming

We used an implementation introduced in [2] to investigate the properties of the method when applied to high volume stream processing. It operates with a distributed ensemble of kernel predictors each trained incrementally and stored at local nodes. The implementation is built using the MapReduce framework. The final prediction of an ensemble is computed as a weighted linear combination of predictions of individual models. Due to the linear weighting and strict pycnophilactic property (12), this distributed scheme preserves areal aggregates as well.

Two baseline scaling properties we would like to demonstrate (see Figure 3) is a constant processing time per sample in a stream and a quadratic growth of time with respect to the number of kernels used in a dictionary (in other words,

²http://www.youtube.com/watch?v=NJB5Cv_WfMM

the size of the matrix \mathbf{M}). Kernel computation was not parallelized in this implementation, however, and it is a stage where huge time savings can be expected as fully parallel implementation of numerical integration in (8) and (10) is straightforward.

6. DISCUSSION AND CONCLUSIONS

Geospatial data analysis often requires dealing with data available at different supports. While spatial statistics provide suitable tools to approach this problem, state-of-the-art applications nevertheless often deal with such data in an ad-hoc way. For example, this concerns non-intrusive population density sensing by leveraging data streams from existing telecommunication infrastructures. The knowledge of physical environment and various constraints available from high-resolution spatial databases is often either neglected or incorporated at the post-processing stage via simple thresholding.

We introduced a mathematically rigorous and consistent framework to incorporate this knowledge into spatial modelling. This framework is applicable for a broad class of machine learning algorithms. We derived a particular area-to-point regression method and adapted an incremental training algorithm to apply it for streaming data. The validity of the method for real-time sensing of city population density from a typical data stream available from telecommunication infrastructures was demonstrated. We investigated the scaling properties of the algorithm and found it as a promising approach to be extended into a larger-scale system useful to uncover high-fidelity patterns of city dynamics.

7. ACKNOWLEDGMENTS

Research presented in this paper was funded by a Strategic Research Cluster grant (07/SRC/I1168) and Stokes Lectureship award by Science Foundation Ireland under the National Development Plan. The authors gratefully acknowledge this support.

8. REFERENCES

- [1] Y. Engel, S. Mannor, and R. Meir. The kernel recursive least-squares algorithm. *IEEE Transactions on Signal Processing*, 52(8):2275–2285, 2004.
- [2] C. Kaiser and A. Pozdnoukhov. Enabling real-time city sensing with kernel stream oracles and mapreduce, June 2011. The First Workshop on Pervasive Urban Applications (PURBA).
- [3] P. C. Kyriakidis. A geostatistical framework for Area-to-Point spatial interpolation. *Geographical Analysis*, 36(3):259–289, 2004.
- [4] C. Ratti, R. Pulselli, S. Williams, and D. Frenchman. Mobile landscapes: Using location data from cell phones for urban analysis. *Environment and Planning B*, 5(33):727–748, 2006.
- [5] A. Sommariva and M. Vianello. Product gauss cubature over polygons based on greens integration formula. *Bit Numerical Mathematics*, 47(13):441–453, 2007.
- [6] W. Tobler. Smooth pycnophilactic interpolation for geographical regions. *Journal of the American Statistical Association*, 367(74):519–530, 1979.
- [7] V. N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, September 1998.