Chapter 1

Simulation and Estimation of Gibbs-Delaunay

1.1 Preliminaries

For a bounded observation window $\Lambda_0 \in \mathcal{B}(\mathbb{R}^3)$ we define $\Lambda := \Lambda_0 \times W \in \mathcal{B}(\mathbb{R}^3 \times W)$, where $W := (0, w_0] \subset \mathbb{R}$ is the set of possible weights. For a locally finite set $\gamma \subset \mathbb{R}^3 \times W$, denote its Delaunay (Regular) triangulation $Del(\gamma)$ and the restriction to Λ as $\gamma_\Lambda = \gamma \cap \Lambda$. Denote π^z the distribution of a homogenous marked Poisson process with intensity $z > 0$ and uniform mark distribution on W, and π^z_Λ its restriction to the observation window Λ.

Probably make this a throughout-this-paper notation, since it will probably be useful often

Define a function to get points/weights from γ? Might be easier

Define $\mathcal{M}_\infty$ etc?

The set $Del(\gamma)$ contains tetrahedra which are here defined as quadruples of points from γ, i.e. $T = \{a, b, c, d\} \subset \gamma$. Denote $B(T)$ the circumsphere of a tetrahedron T. A tetrahedron T is said to be inside the window Λ if $B(T) \cap \Lambda_0 \neq \emptyset$. Denote $Del_\Lambda(\gamma)$ the set of all tetrahedra in $Del(\gamma)$ which are in Λ. For two tetrahedra $T, T' \in Del(\gamma)$ we say $T \sim_{Del} T'$ if T and T' share a face, that is if $\text{card}(T \cap T') \geq 3$.

Is this the right way to go about it?

Program uses barycenter: rewrite?

Define Gibbs properly - DLR + finite volume, but this will do for now

To obtain a Gibbs point process from the Poisson process, its density with respect to the Poisson process needs to be defined. For every $\Lambda \in \mathcal{B}_b(\mathbb{R}^3 \times W)$, we consider the conditional density f_Λ with respect to the Poisson process π^z_Λ:

$$f_\Lambda(\gamma_\Lambda, \gamma_{\Lambda^c}) = \frac{1}{Z_\Lambda(\gamma_{\Lambda^c})} e^{-E_\Lambda(\gamma_\Lambda, \gamma_{\Lambda^c})},$$

Make sure I understand in what sense the configuration is w.r.t. γ_{Λ^c}

where $E_\Lambda(\gamma_\Lambda, \gamma_{\Lambda^c})$ is the energy function of γ_Λ given the outside configuration γ_{Λ^c}. The energy function will be specified in the next section. $Z_\Lambda(\gamma_{\Lambda^c}) := \int e^{-E_\Lambda(\gamma_\Lambda, \gamma_{\Lambda^c})} \pi^z_\Lambda(d\gamma_\Lambda)$ is the normalizing constant. For a comprehensive introduction to Gibbs point processes, see [2].

1.1.1 Model

Here we define the model through its energy function. In general, the model specification is of the following form:

$$E_\Lambda(\gamma_\Lambda, \gamma_\Lambda^c) = \sum_{T \in Del_\Lambda(\gamma)} V_1(T) + \sum_{\substack{\{T,T'\} \subset Del(\gamma) \\ T \sim_{Del} T' \\ T \text{ or } T' \text{ in } Del_\Lambda(\gamma)}} V_2(T, T').$$

In our case, we introduce hardcore parameters ϵ and α to limit the area of the smallest face of T, and its circumradius respectivelly. The intensity of the Poisson process is z and the functions V_1, V_2 are of the form:

$$V_1(T) = \begin{cases} \infty & \text{if } a(T) \leq \epsilon, \\ \infty & \text{if } R(T) \geq \alpha, \\ \theta Sur(T) & \text{otherwise,} \end{cases} \quad \text{and} \quad V_2 = 0. \tag{1.1}$$

Where $a(T)$ is the area of the smallest face of the tetrahedron T, $R(T)$ is the circumradius of T, and $Sur(T)$ is the surface area of the tetrahedron.

Note that the point weights do not play an explicit role in the energy calculation. However they do change the structure of $Del(\gamma)$ and through that the energy too.

1.2 Simulation

The simulations were done in CGAL [7], [5].

1.2.1 Initial configuration

If we used the empty outside configuration, why not just start from an entirely empty position? Would that be wrong?

We are simulating finite volume approximation, how is it connected to the infinite process, exactly?

In [4], three options for the initial configuration are suggested: the empty configuration, a specific fixed outside configuration, and periodic configuration. [4] rejects the empty configuration because it "produces non bounded Delaunay-Voronoi cells". While this is true for a Voronoi diagram, it does not hold for the Delaunay case and so such configuration would be possible. The method chosen was to fix a regular grid of points in and out of Λ such that the resulting tessellation fulfills the hardcore conditions. This does mean that the initial configuration is dependent on the values of α and ϵ. The reason for this choice is simple: the CGAL implementation of 3d periodic triangulations [1] runs much slower than in the non-periodic case.

Yeah, but why not empty?

Make this less vague

1.2.2 MCMC

The algorithm is based on a classic Birth-Death-Move algorithm from [6]. However, one crucial alteration has to be made. One of the main differences between the Regular and Delaunay triangulation is that in a regular triangulation, points may be redundant and thus not be included in the triangulation. A new point may make other points redundant, or be itself redundant. To remove this effect,

Why is it of this form

we decided to only add points from a subset $A \subset [0,1]^3 \times W$ of those possible points that will not result in any redundant points.
First, start from a permissible initial configuration γ_0.

is this a good approach? The space for adding new points changes each step, is that a problem?

1. Let $n = \mathrm{card}(\gamma_0 \cap \Lambda)$.
2. Draw independently a and b uniformly on $[0,1]$.
3. If $a < 1/3$, then generate x uniformly on A and set

$$\gamma_1 = \begin{cases} \gamma_0 \cup \{x\} & \text{if } b < \frac{zf(\gamma_0 \cup \{x\})}{(n+1)f(\gamma_0)}, \\ \gamma_0 & \text{otherwise.} \end{cases} \tag{1.2}$$

4. If $a > 2/3$, then generate x uniformly on γ_0 and set

$$\gamma_1 = \begin{cases} \gamma_0 \setminus \{x\} & \text{if } b < \frac{nf(\gamma_0 \setminus \{x\})}{zf(\gamma_0)}, \\ \gamma_0 & \text{otherwise.} \end{cases} \tag{1.3}$$

5. If $1/3 < a < 2/3$, then generate x uniformly on γ_0, generate $y \sim \mathcal{N}(x, \sigma^2 I)$ such that $y \in A$ and set

$$\gamma_1 = \begin{cases} \gamma_0 \setminus \{x\} \cup \{y\} & \text{if } b < \frac{f(\gamma_0 \setminus \{x\} \cup \{y\})}{f(\gamma_0)}, \\ \gamma_0 & \text{otherwise.} \end{cases} \tag{1.4}$$

6. Set $\gamma_0 \leftarrow \gamma_1$ and go to 1.

This is not true at the moment, currently the algorithm just skips points in conflict with other points.

Why does this work? Can we choose different proposal density to improve the convergence?

If the moved point would fall outside $[0,1]^3$, it is 'bounced back' from the boundary of $[0,1]^3$ as if the boundary of the unit box was a solid wall. This differs from [4], where the point would be replaced inside $[0,1]^2$ by the periodic property. The idea is that a small perturbation to the point's position should not result in a radically different positionof the point.

Is this a good approach? I makes the density of moved points next to boundary concentrate more in some places

1.2.3 Explicit expression for the proposal densities

This section is poorly described. Notation and some new terms needs to be defined/improved for this to be understandable and precise. Improve after writing a chapter on Delaunay

Describe this properly

Since the energy function is a simple sum over the individual tetrahedra, the density ratios 1.2, 1.3, 1.4, can be further simplified in all three cases. Since all the density ratios in contain both density of γ and density of γ with point added or removed in a ratio, most terms in the sum will be canceled out. In each step some tetrahedra are removed and some added - the key is in finding which tetrahedra those are.

When a point x is added into $Del(\gamma)$, the tetrahedra in conflict with x are precisely those that will be deleted. Likewise, the tetrahedra in $Del(\gamma \cup \{x\})$ that are adjacent to x (contain x as one of its vertices) are precisely those that were newly created.

For $x \in \gamma$, denote $\mathrm{DT}^{\ell}(x, \gamma)$ the set of tetrahedra adjacent to x in $Del(\gamma)$. For $x \notin \gamma$, denote $\mathrm{DT}^{\otimes}(x, \gamma)$ the set of tetrahedra conflicting with point x.

Possible an easier way would be to denote those $T \in Del(\gamma) : x \in T$ and $T \in Del(\gamma) : x \in B(T)$

Birth step 1.2 then becomes:

$$\begin{aligned}\frac{f(\gamma_0 \cup \{x\})}{f(\gamma_0)} &= \exp\left(\sum_{T \in Del_\Lambda(\gamma_0 \cup \{x\})} V_1(T) - \sum_{T \in Del_\Lambda(\gamma_0)} V_1(T)\right) \\ &= \exp\left(\sum_{T \in \mathrm{DT}^{\otimes}(x, \gamma_0)} V_1(T) - \sum_{T \in \mathrm{DT}^{\ell}(x, \gamma_0 \cup \{x\})} V_1(T)\right)\end{aligned}$$

Death step 1.3 becomes:

$$\begin{aligned}\frac{f(\gamma_0 \setminus \{x\})}{f(\gamma_0)} &= \exp\left(\sum_{T \in Del_\Lambda(\gamma \setminus \{x\})} V_1(T) - \sum_{T \in Del_\Lambda(\gamma)} V_1(T)\right) \\ &= \exp\left(\sum_{T \in \mathrm{DT}^{\ell}(x, \gamma_0)} V_1(T) - \sum_{T \in \mathrm{DT}^{\otimes}(x, \gamma_0 \setminus \{x\})} V_1(T)\right)\end{aligned}$$

Move step 1.4 becomes:

$$\begin{aligned}\frac{f(\gamma_0 \setminus \{x\} \cup \{y\})}{f(\gamma_0)} &= \frac{f(\gamma_0 \setminus \{x\} \cup \{y\})}{f(\gamma_0 \setminus \{x\})} \frac{f(\gamma_0 \setminus \{x\})}{f(\gamma_0)} \\ &= \exp\left(\sum_{T \in \mathrm{DT}^{\otimes}(x, \gamma_0 \setminus \{x\})} V_1(T) - \sum_{T \in \mathrm{DT}^{\ell}(x, \gamma_0 \setminus \{x\} \cup \{y\})} V_1(T)\right. \\ &\quad \left. + \sum_{T \in \mathrm{DT}^{\ell}(x, \gamma_0)} V_1(T) - \sum_{T \in \mathrm{DT}^{\otimes}(x, \gamma_0 \setminus \{x\})} V_1(T)\right)\end{aligned}$$

These expressions simplify the energy calculation immensely. Whereas calculating the energy for the whole tessellation requires all the tetrahedra, and thus depends on $\mathrm{card}(\gamma \cap \Lambda)$, the final expressions only contain the tetrahedra local to x, and thus the energy can be calculated in constant time.

1.3 Estimation

Assume now that we obtain the point configuration γ on the observation window $\Lambda_n = [-n, n]^3 \times W$ and wish to estimate the model parameters.

Boundary problems. Do they simply exist because we're assuming to *only* know the configuration on Λ_n?

The estimation procedure closely follows that from [4]. That is a two-step approach, first estimating the hardcore parameters $\beta = (\epsilon, \alpha)$ and then using the estimates to obtain the estimate of θ through maximum pseudolikelihood (MPLE).

What exactly is the role of the growing window in [4]?

1.3.1 Estimation of the hardcore parameters

Thanks to the fact that the hardcore parameter ϵ satisfies

[4] has this the other way around?

$$\text{if } \epsilon < \epsilon' \text{ then } \forall \Lambda,\ E_\Lambda^{\epsilon,\alpha,\theta}(\gamma_\Lambda, \gamma_{\Lambda^c}) < \infty \Rightarrow E_\Lambda^{\epsilon',\alpha,\theta}(\gamma_\Lambda, \gamma_{\Lambda^c}) < \infty,$$

and the hardcore parameter α satisfies

$$\text{if } \alpha > \alpha' \text{ then } \forall \Lambda,\ E_\Lambda^{\epsilon,\alpha,\theta}(\gamma_\Lambda, \gamma_{\Lambda^c}) < \infty \Rightarrow E_\Lambda^{\epsilon,\alpha',\theta}(\gamma_\Lambda, \gamma_{\Lambda^c}) < \infty,$$

their consistent estimators are:

$$\hat{\epsilon} = \inf\{\epsilon > 0, E_\Lambda(\gamma_\Lambda, \gamma_\Lambda^c) < \infty\},$$

$$\hat{\alpha} = \sup\{\alpha > 0, E_\Lambda(\gamma_\Lambda, \gamma_\Lambda^c) < \infty\}.$$

In practice, the parameters are estimated as

Are these consistent? Why?

$$\hat{\epsilon} = \min\{a(T), T \in Del_\Lambda(\gamma)\},$$

$$\hat{\alpha} = \max\{r(T), T \in Del_\Lambda(\gamma)\}.$$

The estimate $\hat{\beta} = (\hat{\epsilon}, \hat{\alpha})$ is then used in the pseudo-likelihood function in the second estimation step.

1.3.2 Estimation of the smooth interaction parameters

Equation references

The classical version of MPLE requires hereditarity of the interactions. Hereditarity means that for every permissible γ, the point pattern $\gamma \setminus \{x\}$ remains permissible for every $x \in \gamma$, that is any point can be removed from the point pattern. The hardcore interaction in the model 1.1 does not satisfy this condition. However, [3] extends MPLE to the non-hereditary case.

How does it relate to my case, exactly?

Since some points cannot be removed from the tessellation, we need to introduce the notion of a removable points. A point $x \in \gamma$ is removable in γ iff $\gamma \setminus \{x\}$ is permissible. We denote $\mathcal{R}^\beta(\gamma)$ the set of removable points of γ. Similarly the notion of an addable point will be useful. A point $x \in \gamma$ is addable in γ iff $\gamma \cup \{x\}$ is permissible.

The definition in [3] is actually different and this is given as a proposition.

In the non-hereditary case, the pseudo-likelihood function then becomes:

$$PLL_{\Lambda_n}(\gamma, z, \beta, \theta) = \int_{\Lambda_n'} z \exp(-h^{\beta,\theta}(x, \gamma))dx + \sum_{x \in \mathcal{R}^\beta(\gamma) \cap \Lambda_n} \big(h^{\beta,\theta}(x, \gamma \setminus \{x\}) - \ln(z)\big), \tag{1.5}$$

where Λ_n' is the set of all addable points in Λ_n and $h^{\beta,\theta}(x, \gamma \setminus \{x\})$ is local energy of x in γ defined for every $x \in \mathcal{R}^\beta(\gamma)$ by:

Is there any reason to define it the way [3] does?

$$h^{\beta,\theta}(x, \gamma \setminus \{x\}) = E_\Lambda^{\beta,\theta}(\gamma_\Lambda, \gamma_{\Lambda^c}) - E_\Lambda^{\beta,\theta}(\gamma_\Lambda \setminus \{x\}, \gamma_{\Lambda^c}).$$

The estimates $\hat{\theta}$ and $\hat{z}$ are obtained through minimizing the PLL_{Λ_n} function 1.5:

Connection between this and Papangelou could be useful

$$(\hat{z}, \hat{\theta}) = \text{argmin}_{z,\theta} PLL_{\Lambda_n}(\gamma, z, \hat{\beta}, \theta).$$

By differentiating the PLL function 1.5 with respect to z, respectivelly θ, and setting them equal to zero, we obtain the estimate for $\hat{z}$,

$$\hat{z} = \frac{\text{card}(\mathcal{R}^{\beta}(\gamma) \cap \Lambda_n)}{\int_{\Lambda_n} \exp\left(-h^{\hat{\beta},\theta}(x,\gamma)\right) dx}, \tag{1.6}$$

and the estimate $\hat{\theta}$ as the solution of

$$z \int_{\Lambda'_n} (h^{\hat{\beta},1}(x,\gamma) \exp\left(-h^{\hat{\beta},\theta}(x,\gamma)\right)) dx = \sum_{x \in \mathcal{R}^{\hat{\beta}}(\gamma) \cap \Lambda_n} h^{\hat{\beta},1}(x, \gamma \setminus \{x\}), \tag{1.7}$$

where we have used the fact that the local energy depends on θ linearly, yielding

$$\frac{\partial h^{\hat{\beta},\theta}}{\partial \theta}(x,\gamma) = h^{\hat{\beta},1}(x,\gamma).$$

Practical implementation

We obtain the estimate of θ by substituting the expression for $\hat{z}$ 1.6 into 1.7. This leads to the equation

$$\frac{\int_{\Lambda'_n} (h^{\hat{\beta},1}(x,\gamma) \exp\left(-h^{\hat{\beta},\theta}(x,\gamma)\right)) dx}{\int_{\Lambda_n} \exp\left(-h^{\hat{\beta},\theta}(x,\gamma)\right) dx} = \frac{\sum_{x \in \mathcal{R}^{\hat{\beta}}(\gamma) \cap \Lambda_n} h^{\hat{\beta},1}(x, \gamma \setminus \{x\})}{\text{card}(\mathcal{R}^{\beta}(\gamma) \cap \Lambda_n)}.$$

In order to simplify the estimation of θ, we can simplify this equation further. First, we denote the righ-hand-side of the equation as c as it is constant with respect to θ. Second, we note that $x \notin \Lambda'_n \Rightarrow \exp\left(-h^{\hat{\beta},\theta}(x,\gamma)\right) = 0$ which enables us to integrate over Λ'_n instead of the whole Λ_n. Lastly we denote the local energy $h^{\hat{\beta},1}(x,\gamma) =: h(x)$, yielding the expression

$$\int_{\Lambda'_n} h(x) \exp\left(-\theta h(x)\right) dx = c \int_{\Lambda'_n} \exp\left(-\theta h(x)\right),$$

leading into the final expression

$$\int_{\Lambda'_n} \exp\left(-\theta h(x)\right)(h(x) - c) dx. \tag{1.8}$$

The integral 1.8 is estimated using Monte-Carlo integration, i.e. is approximately equal to

$$\frac{1}{N} \sum_{i=0}^{N} 1_{\Lambda'_n}(x_i) \exp\left(-\theta h_i\right)(h_i - c) dx$$

where $h_i = h^{\hat{\beta},1}(x_i,\gamma)$ and $x_1, \ldots, x_N$ is a random sample from the uniform distribution on Λ'_n

After $\hat{\theta}$ is estimated, we then obtain the estimate $\hat{z}$ with $\hat{\theta}$ instead of θ and the integral replaced by a MC-integration approximation.

Do we need the indicator function if we're only sampling from Λ'_n ?

Citations: Make sure I cite the published article, not arxiv

Bibliography

[1] Manuel Caroli, Aymeric Pellé, Mael Rouxel-Labbé, and Monique Teillaud. 3D periodic triangulations. In *CGAL User and Reference Manual*. CGAL Editorial Board, 4.12 edition, 2018.

[2] D. Dereudre. Introduction to the theory of Gibbs point processes. *ArXiv e-prints*, January 2017.

[3] D. Dereudre and F. Lavancier. Campbell equilibrium equation and pseudo-likelihood estimation for non-hereditary Gibbs point processes. *ArXiv e-prints*, September 2007.

[4] D. Dereudre and F. Lavancier. Practical simulation and estimation for Gibbs Delaunay-Voronoi tessellations with geometric hardcore interaction. *ArXiv e-prints*, May 2010.

[5] Clément Jamin, Sylvain Pion, and Monique Teillaud. 3D triangulations. In *CGAL User and Reference Manual*. CGAL Editorial Board, 4.12 edition, 2018.

[6] J. Moller and R.P. Waagepetersen. *Statistical Inference and Simulation for Spatial Point Processes*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. CRC Press, 2003.

[7] The CGAL Project. *CGAL User and Reference Manual*. CGAL Editorial Board, 4.12 edition, 2018.

Todo list

List of Abbreviations

1. $\mathcal{B}_b(X)$ bounded Borel sets in the space X
2. card(A) cardinality of the set A

Appendix: Notes on existing theory

This section has two goals

- To track the existing theory behind [4] and
- To see if extensions to our case ($d = 3$, regular triangulation) are possible

1.3.3 Logic of the procedure

Are we simulating a finite volume Gibbs? If so, why does that allow us to talk about infinite volume Gibbs?

1.3.4 Existence

1.3.5 Uniqueness

1.3.6 Irreducibility

For $d = 2$, it was proved in [4] through the saturation technique. Could it be proved for $d = 3$, or even in the regular case? Perhaps if one of the parameters is removed?

1.3.7 Constincency

In [3], consistency is proven for the two-step approach in the non-hereditary case. For $d = 2$, the assumptions are checked for the Delaunay model, but only in case of one hardcore parameter - two would require "some further assumptions and proofs"