Inferring a nonparametric dust model via extinction estimates

Coryn A.L. Bailer-Jones
Max Planck Institute for Astronomy, Heidelberg (calj@mpia.de)
Last updated: 13 April 2015 (first version 5 February 2014)

Summary

I examine the problem of using a nonparametric model to infer the three-dimensional distribution of dust in the Galaxy using measurements of the line-of-sight extinction to number of stars. The model employs a Gaussian process to couple dust densities across space in a covariant sense. We can then estimate, probabilistically, the dust density at any point in space, even where we have no extinction measures. There exists a straight forward, but impractical, numeric solution. I show that an analytic solution exists.

1 Definitions

- scalar variables are written in lower case normal maths font; a subscript may be used to provide further information. E.g. μ, ρ_{J+1}.
- scalar sizes of vectors and matrices are written in upper case normal maths font, e.g. J.
- vectors are written in bold upright font (either upper case or lower case); the subscript indicates the number of elements. E.g. $\boldsymbol{\rho}_J$, $\mathbf{A}_N$. These are column vectors.
- matrices are written in upper case upright font, e.g. G; if the matrix is always square, then a subscript is used to indicate the size of the matrix, e.g. C_{J+1}.
- the transpose of a vector or matrix is denoted by $^\mathsf{T}$, e.g. $\boldsymbol{\rho}_J^\mathsf{T}$, G^T. A transposed vector is a row vector.
- products such as $\mathbf{m}_J^\mathsf{T}\boldsymbol{\rho}_J$ are scalar products, and products such as $\mathbf{k}_J\mathbf{k}_J^\mathsf{T}$ are outer products, i.e. with size $J \times J$.
- E[] and Cov[] are the expectation and covariance operators respectively.

2 Problem setup

We want to determine the 3D spatial distribution of dust given measurements of the line-of-sight extinction caused by this dust to a number of stars. Specifically, given the extinction toward a number of stars at different points in the Galaxy, what is our best probabilistic estimate for the dust density at any point in the Galaxy (not necessarily along the line-of-sight to one of these stars)?

Let the extinction (absorption) measured to star n at position $\mathbf{r}_n$ be A_n. A generative model for this is that extinction is caused by the total absorption of dust along the line of sight

$$f_n \propto \int_{r=0}^{r=r_n} \rho(\mathbf{r})\, dr \tag{1}$$

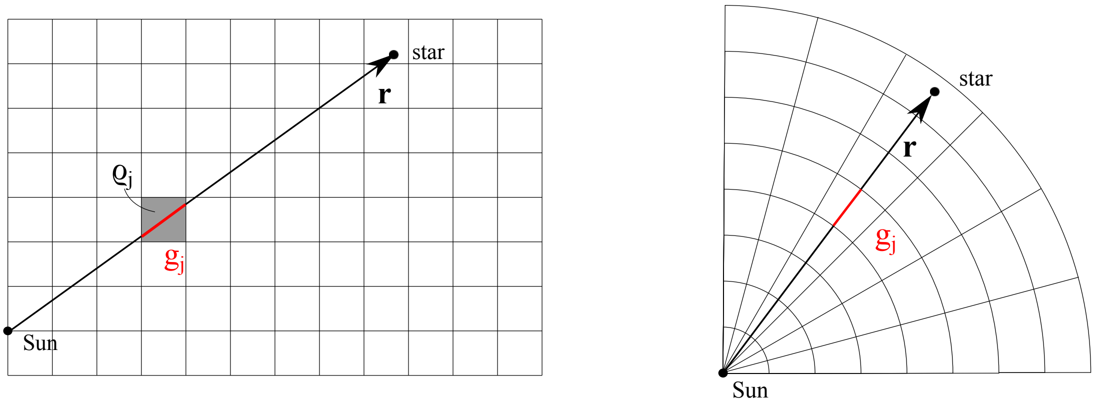

Figure 1: Two possible general geometries for the dust cells: Cartesian (left) and polar (right). In reality these are both in three-dimensions.

where $\rho(\mathbf{r})$ is the dust density at position $\mathbf{r}$, and $r = |\mathbf{r}|$. The observer is at the origin. If we were to set up a parametric model for $\rho(\mathbf{r})$, then we could solve for its parameters based on a set of measurements $\{A_n\}$ for stars with known positions. This is described in Bailer-Jones (2011).

Here we set up instead a nonparametric model. Such a model is free from the strong and/or simplistic assumptions which must be made in a parametric model, and so is potentially better able to capture the likely complex distribution of dust in the Galaxy.

In place of a parametric model for $\rho(\mathbf{r})$, we divide 3D space into a large number of fixed cells, the dust density in each of which is $\rho_{n,j}$. These are unknown. The integral in equation 1 can now be replaced by a sum

$$f_n = \kappa \sum_j g_{n,j} \rho_{n,j} \; . \tag{2}$$

$g_{n,j}$ is a geometric factor which determines the amount of the cell which contributes to the measured extinction. Typically f_n is in magnitudes, $\rho_{n,j}$ is mass per unit volume, and $g_{n,j}$ is a volume. κ is a property of the dust: the magnitude of extinction per unit mass of dust.

We are free to choose the geometry – the size and shape of the cells – according to our needs. Figure 1 shows two examples, a Cartesian grid and a polar grid. Our choice will effect how we determine the geometric factors. We will return later to this issue later. We assume that the 3D positions of the stars are known, so that the geometric factors are uniquely determined by these and must not be inferred.

If the measured extinction is A_n, and this measurement follows a Gaussian noise model with standard deviation σ_n, then the likelihood – the probability of the data given the model parameters – is

$$P(A_n | \{\rho_{n,j}\}_n) = \frac{1}{\sqrt{2\pi}\sigma_n} \exp\left[-\frac{1}{2\sigma_n^2}(A_n - f_n)^2 \right] \tag{3}$$

assuming f_n has no variance.

Let us suppose that we measure the extinction to N stars. We now adopt a 3D polar grid, so that each line-of-sight includes a unique set of dust cells which are in no other line-of-sight. Furthermore, we do not adopt a complete grid: we only define cells along the line-of-sights to the N stars (see Figure 2). Ignoring the cross-sectional area for now, each line-of-sight is a 1D pencil beam split up into a number of cells, and

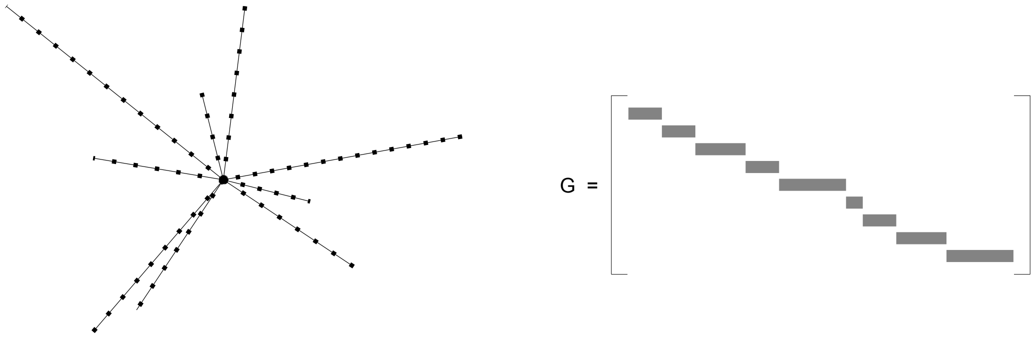

Figure 2: Practical dust cell geometry. We only need to define cells along the lines-of-sight to stars for which we have extinction measurements. These can be considered as long thin pencil beams, as shown on the left, where the dots mark the centre of the cells (the observer is where the beams join). Right: a sketch of the corresponding $N \times J$ geometric matrix G.

$g_{n,j}$ is determined by the length of the cell along the line-of-sight. For the N different lines-of-sight, let there be a total of J dust cells. We define G as the $N \times J$ matrix with elements $\kappa \, g_{n,j}$. That is, row n of G contains (κ times) the geometric factors for that line-of-sight. Most of the elements in this row will be zero, namely those elements which corresponds to dust cells in all the other lines-of-sight. For example, if $N = 100$ and we have 10 cells per line-of-sight, then $J = 1000$, and 990 elements in every row will be zero. G will always be a sparse matrix. With this particular polar coordinate geometry, it will also only have one non-zero element per column. It is even be possible to define a single cell for each line of sight, in which case $N = J$ and G is a diagonal matrix.

If we use $\boldsymbol{\rho}_J$ to denote the vector containing $\rho_{n,j}$ for all cells in all lines-of-sight, we can now write equation 2 for all N lines-of-sight as

$$\mathbf{f}_N \; = \; \mathrm{G}\boldsymbol{\rho}_J \, . \tag{4}$$

Writing the N extinction measurements as the vector $\mathbf{A}_N$, and the covariance in $(\mathbf{A}_N - \mathrm{G}\boldsymbol{\rho}_J)$ as V_N, then we can write the likelihood as the N-dimensional Gaussian

$$P(\mathbf{A}_N|\boldsymbol{\rho}_J) \; = \; \frac{1}{(2\pi)^{N/2}|\mathrm{V}_N|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{A}_N - \mathrm{G}\boldsymbol{\rho}_J)^\mathsf{T}\mathrm{V}_N^{-1}(\mathbf{A}_N - \mathrm{G}\boldsymbol{\rho}_J)\right] \, . \tag{5}$$

Let the covariance in $\mathbf{A}_N$ be the matrix Σ_N (which will be diagonal if the extinction measurements are independent of one another). If the dust model is deterministic (i.e. has no covariance), then $\mathrm{V}_N = \Sigma_N$. But we will consider stochastic models in which the elements of $\boldsymbol{\rho}_J$ are covariate, in which case $\mathbf{A}_N$ must include the covariance in $\mathrm{G}\boldsymbol{\rho}_J$ also.[1] Assuming $\mathbf{A}_N$ and $\mathrm{G}\boldsymbol{\rho}_J$ are uncorrelated[2] we can sum the individual covariances so that

$$\begin{aligned} \mathrm{V}_N \; &= \; Cov(\mathbf{A}_N) + Cov(\mathrm{G}\boldsymbol{\rho}_J) \\ &= \; \Sigma_N + \mathrm{E}[\mathrm{G}\boldsymbol{\rho}_J(\mathrm{G}\boldsymbol{\rho}_J)^\mathsf{T}] - (\mathrm{E}[\mathrm{G}\boldsymbol{\rho}_J])^2 \\ &= \; \Sigma_N + \mathrm{E}[\mathrm{G}\boldsymbol{\rho}_J\boldsymbol{\rho}_J^\mathsf{T}\mathrm{G}^\mathsf{T}] - (\mathrm{E}[\mathrm{G}\boldsymbol{\rho}_J])^2 \end{aligned} \tag{6}$$

[1] Thanks to Rene Andrae for pointing this out.

[2] This will not be the case if the extinctions and distances have both been derived from the same data, e.g. (spectro)photometric observations.

where $\boldsymbol{\rho}_J\boldsymbol{\rho}_J^\mathsf{T}$ is an outer product. If G is constant (which requires that we have no distance errors to the stars), then

$$\begin{aligned}\mathrm{E}[\mathrm{G}\boldsymbol{\rho}_J\boldsymbol{\rho}_J^\mathsf{T}\mathrm{G}^\mathsf{T}] &= \mathrm{G}\mathrm{E}[\boldsymbol{\rho}_J\boldsymbol{\rho}_J^\mathsf{T}]\mathrm{G}^\mathsf{T} \\ &= \mathrm{G}(\mathrm{C}_J + \mathrm{E}[\boldsymbol{\rho}_J]\mathrm{E}[\boldsymbol{\rho}_J^\mathsf{T}]))\mathrm{G}^\mathsf{T} \qquad (7)\end{aligned}$$

where C_J is the covariance of $\boldsymbol{\rho}_J$ and we have used the definition of covariance. If $\mathrm{E}[\boldsymbol{\rho}_J] = 0$ (which corresponds to a zero mean Gaussian Process in the next section), then $\mathrm{E}[\mathrm{G}\boldsymbol{\rho}_J] = 0$ too so

$$\mathrm{V}_N = \Sigma_N + \mathrm{G}\mathrm{C}_J\mathrm{G}^\mathsf{T} \ . \qquad (8)$$

Note that if Σ_N is symmetric then so is V_N (because C_J always is).

Problem statement: Our goal is to estimate the dust density ρ_{J+1} at an arbitrary point $\mathbf{r}_{J+1}$ in 3D space, given N measurements of of the extinction, $\mathbf{A}_N$, at known positions. Put probabilistically, we want to find $P(\rho_{J+1}|\mathbf{A}_N)$.

While each element of $\boldsymbol{\rho}_J$ refers to the average density in the corresponding cell, however large it may be, ρ_{J+1} is the density at the *point* $\mathbf{r}_{J+1}$. There is no concept of a cell for that point. Only when we calculate extinctions do we need to introduce the concept of cells.

3 Gaussian process model

In the above formulation $J \geq N$ and $N \gg 1$. A typical problem may involve $N = 10^4$ and $J = 10^5$. In order to infer the dust densities, we need to introduce some connection between the lines-of-sight, otherwise we just have N independent equations like equation 2. We do this using a Gaussian process (e.g. Gibbs 1997. Rasmussen & Williams 2006). This says that the joint distribution of the dust density at any J different points is a J-dimensional zero mean Gaussian, with a covariance matrix which depends on the distance between the points, i.e.

$$P(\boldsymbol{\rho}_J) = \frac{1}{(2\pi)^{J/2}|\mathrm{C}_J|^{1/2}} \exp\left[-\frac{1}{2}\boldsymbol{\rho}_J^\mathsf{T}\mathrm{C}_J^{-1}\boldsymbol{\rho}_J\right] \qquad (9)$$

The key aspect of the Gaussian processes is to choose an appropriate covariance function, the function which determines the elements, $c_{i,j}$, of the covariance matrix C_J. Here we use

$$c_{i,j} = \theta_1 \exp\left(-\frac{|\mathbf{r}_i - \mathbf{r}_j|^2}{\lambda^2}\right) + \theta_2 \ . \qquad (10)$$

This gives the covariance between the dust densities at two points $\mathbf{r}_i$ and $\mathbf{r}_j$. When these refer to the J dust cells, these coordinates are of the centre of the dust cells. λ is the scale length of the correlation. The above covariance function simply expresses the physical principle that the closer two points in space are (relative to λ), the more correlated their dust densities will be. Thus the larger λ, the smoother the variation of dust density over a fixed length scale. A Gaussian process is a way of specifying a prior on the covariances, as opposed to specifying the functional form of the dust variation in physical space (which is what we would normally do with a parametric model). θ_1 specifies the overall scale of variations of the dust density. θ_2 allows for a non-zero mean of the Gaussian process (which is why we can adopt a zero mean in equation 9). Here we consider all the θ (hyper)parameters to be fixed, although they can be inferred from the data too.

The function in equation 10 drops off very rapidly with increasing separation of two points, meaning that two point separated by much more than λ will hardly influence each other. Yet the covariance is still non-zero. In order to speed up the inversion of the covariance matrix, it is useful to adopt instead a truncated covariance function (thereby giving the covariance matrix compact support). In doing so we must ensure that the matrix remains positive semi-definite. This can be achieved by convolving any symmetric kernel with itself, for example.

A useful property of the Gaussian processes is that the conditional distribution of the dust density at any point $\mathbf{r}_{J+1}$, $P(\rho_{J+1}|\boldsymbol{\rho}_J)$, is also Gaussian.

Our goal is to determine at least $J+1$ parameters from N measurements. This presents no problem for inference methods, but of course means that the resulting parameters will not be independent. Indeed, the whole point of the Gaussian process is to introduce a correlation between the dust cells to make the problem tractable. This Gaussian model of course allows the dust density, ρ, to be negative. This is unphysical, but can be tolerated. This will be discussed later.

4 Numeric solution

Using the law of marginalization, we can write the quantity of interest as

$$\begin{aligned}
P(\rho_{J+1}|\mathbf{A}_N) &= \int_{\boldsymbol{\rho}_J} P(\rho_{J+1}, \boldsymbol{\rho}_J|\mathbf{A}_N) d\boldsymbol{\rho}_J \\
&= \int_{\boldsymbol{\rho}_J} P(\rho_{J+1}|\boldsymbol{\rho}_J, \mathbf{A}_N) P(\boldsymbol{\rho}_J|\mathbf{A}_N)\, d\boldsymbol{\rho}_J \\
&= \int_{\boldsymbol{\rho}_J} P(\rho_{J+1}|\boldsymbol{\rho}_J) P(\boldsymbol{\rho}_J|\mathbf{A}_N)\, d\boldsymbol{\rho}_J \qquad (11)
\end{aligned}$$

where in the last step we have used conditional independence of ρ_{J+1} on $\mathbf{A}_N$ once $\boldsymbol{\rho}_J$ is specified (e.g. determined from the data). The first term under the integral is given by the Gaussian processes. The second term is the posterior PDF over the dust densities, given by Bayes theorem from the prior and the likelihood

$$P(\boldsymbol{\rho}_J|\mathbf{A}_N) = \frac{P(\mathbf{A}_N|\boldsymbol{\rho}_J)P(\boldsymbol{\rho}_J)}{P(\mathbf{A}_N)} \qquad (12)$$

where the denominator, which is independent of $\boldsymbol{\rho}_J$, can be absorbed into a normalization constant. Using the Monte Carlo principle, we can approximate equation 11 as

$$P(\rho_{J+1}|\mathbf{A}_N) \simeq \frac{1}{K}\sum_k P(\rho_{J+1}|(\boldsymbol{\rho}_J)_k) \qquad (13)$$

where $(\boldsymbol{\rho}_J)_k$ indicates a sample of the dust density vector drawn from the posterior PDF given by equation 12. Thus the posterior PDF over ρ_{J+1} is a sum of 1D Gaussian process prior PDFs, each with a different value of $\boldsymbol{\rho}_J$ drawn from its J-dimensional posterior. Note that this is not, in general, a Gaussian itself.

Performing this Monte Carlo integration will be almost impossible in practice. The posterior is defined in J dimensions, and for real problems we expect $J > 10^4$. It would be extremely hard to draw reliable samples from this, and/or the sum would have to be over an impossibly large number of samples.

5 Analytic solution

Using the law of marginalization and then applying Bayes theorem, we can write the quantity of interest as

$$\begin{aligned} P(\rho_{J+1}|\mathbf{A}_N) &= \int_{\boldsymbol{\rho}_J} P(\rho_{J+1}, \boldsymbol{\rho}_J|\mathbf{A}_N)\, d\boldsymbol{\rho}_J \\ &= \int_{\boldsymbol{\rho}_J} \frac{P(\rho_{J+1}, \boldsymbol{\rho}_J)P(\mathbf{A}_N|\rho_{J+1}, \boldsymbol{\rho}_J)}{P(\mathbf{A}_N)}\, d\boldsymbol{\rho}_J \\ &= \frac{1}{P(\mathbf{A}_N)} \int_{\boldsymbol{\rho}_J} P(\rho_{J+1}, \boldsymbol{\rho}_J)P(\mathbf{A}_N|\boldsymbol{\rho}_J)\, d\boldsymbol{\rho}_J \qquad (14) \end{aligned}$$

where in the last line we use the fact that $\mathbf{A}_N$ is independent of ρ_{J+1} once conditioned on $\boldsymbol{\rho}_J$. The integral is evaluated over the whole space, i.e. from $-\infty$ to $+\infty$ for each component of $\boldsymbol{\rho}_J$. The term outside the integral is independent of ρ_{J+1} so is just part of the normalization constant. The first term under the integral is the GP prior (equation 9), now in $J+1$ dimensions. The second term is the likelihood (equation 5). Both are Gaussians, albeit not in ρ_{J+1}. But their arguments are linear functions of ρ_{J+1}, which suggests that this integral may have an analytic solution. In the next section we show that it does.

5.1 Analytic solution of the integral

For brevity we write equation 14 as

$$P(\rho_{J+1}|\mathbf{A}_N) = \frac{1}{Z} \int_{\boldsymbol{\rho}_J} e^{-\psi/2}\, d\boldsymbol{\rho}_J \qquad (15)$$

where Z is a normalization constant, and where from equation 9 (with $J \to J+1$) and equation 5

$$\begin{aligned} \psi &= \boldsymbol{\rho}_{J+1}^{\mathsf{T}} \mathrm{C}_{J+1}^{-1} \boldsymbol{\rho}_{J+1} + (\mathbf{A}_N - G\boldsymbol{\rho}_J)^{\mathsf{T}} \mathrm{V}_N^{-1} (\mathbf{A}_N - G\boldsymbol{\rho}_J) \\ &= \boldsymbol{\rho}_{J+1}^{\mathsf{T}} \mathrm{C}_{J+1}^{-1} \boldsymbol{\rho}_{J+1} + \mathbf{A}_N^{\mathsf{T}} \mathrm{V}_N^{-1} \mathbf{A}_N - \boldsymbol{\rho}_J^{\mathsf{T}} \mathrm{G}^{\mathsf{T}} \mathrm{V}_N^{-1} \mathbf{A}_N + \boldsymbol{\rho}_J^{\mathsf{T}} \mathrm{G}^{\mathsf{T}} \mathrm{V}_N^{-1} \mathrm{G} \boldsymbol{\rho}_J - \mathbf{A}_N^{\mathsf{T}} \mathrm{V}_N^{-1} \mathrm{G} \boldsymbol{\rho}_J \qquad (16) \end{aligned}$$

Each term in the above equation is a scalar, and transposing a scalar leaves it unchanged. Transposing the third term:

$$(\boldsymbol{\rho}_J^{\mathsf{T}} \mathrm{G}^{\mathsf{T}} \mathrm{V}_N^{-1} \mathbf{A}_N)^{\mathsf{T}} = \mathbf{A}_N^{\mathsf{T}} \mathrm{V}_N^{-1} \mathrm{G} \boldsymbol{\rho}_J \qquad (17)$$

where we have used the fact that V_N is symmetric, and therefore its inverse too, so transposing it has no effect. Thus the third and fifth terms are identical.

In order to perform the integral we need to separate $\boldsymbol{\rho}_J$ from ρ_{J+1} in $\boldsymbol{\rho}_{J+1}$. We achieve this by partitioning the vector $\boldsymbol{\rho}_{J+1}$ and the matrix C_{J+1}^{-1} in the following way

$$\boldsymbol{\rho}_{J+1} = \begin{bmatrix} \boldsymbol{\rho}_J \\ \rho_{J+1} \end{bmatrix} \qquad (18)$$

$$\mathrm{C}_{J+1}^{-1} = \begin{bmatrix} \mathrm{M}_J & \mathbf{m}_J \\ \mathbf{m}_J^{\mathsf{T}} & \mu \end{bmatrix} \qquad (19)$$

where M_J is a $J \times J$ matrix, $\mathbf{m}_J$ is a $J \times 1$ vector, and μ is a scalar. Note that these are components of the inverted matrix, not components of C_{J+1} which are then inverted! We can then write

$$\begin{aligned}
\boldsymbol{\rho}_{J+1}^{\mathsf{T}} \mathrm{C}_{J+1}^{-1} \boldsymbol{\rho}_{J+1} &= \left[\begin{array}{cc} \boldsymbol{\rho}_J^{\mathsf{T}} & \rho_{J+1} \end{array}\right] \begin{bmatrix} \mathrm{M}_J & \mathbf{m}_J \\ \mathbf{m}_J^{\mathsf{T}} & \mu \end{bmatrix} \left[\begin{array}{c} \boldsymbol{\rho}_J \\ \rho_{J+1} \end{array}\right] \\
&= \left[\begin{array}{cc} \boldsymbol{\rho}_J^{\mathsf{T}} & \rho_{J+1} \end{array}\right] \left[\begin{array}{c} \mathrm{M}_J \boldsymbol{\rho}_J + \ \mathbf{m}_J \rho_{J+1} \\ \mathbf{m}_J^{\mathsf{T}} \boldsymbol{\rho}_J + \ \mu \rho_{J+1} \end{array}\right] \\
&= \boldsymbol{\rho}_J^{\mathsf{T}} \mathrm{M}_J \boldsymbol{\rho}_J + \boldsymbol{\rho}_J^{\mathsf{T}} \mathbf{m}_J \rho_{J+1} + \rho_{J+1} \mathbf{m}_J^{\mathsf{T}} \boldsymbol{\rho}_J + \mu \rho_{J+1}^2 \\
&= \boldsymbol{\rho}_J^{\mathsf{T}} \mathrm{M}_J \boldsymbol{\rho}_J + 2\rho_{J+1} \mathbf{m}_J^{\mathsf{T}} \boldsymbol{\rho}_J + \mu \rho_{J+1}^2 \ . \qquad (20)
\end{aligned}$$

Substituting this into equation 16 and gathering together terms gives

$$\begin{aligned}
\psi &= \boldsymbol{\rho}_J^{\mathsf{T}} \mathrm{M}_J \boldsymbol{\rho}_J + \boldsymbol{\rho}_J^{\mathsf{T}} \mathrm{G}^{\mathsf{T}} \mathrm{V}_N^{-1} \mathrm{G} \boldsymbol{\rho}_J + 2\rho_{J+1} \mathbf{m}_J^{\mathsf{T}} \boldsymbol{\rho}_J - 2\mathbf{A}_N^{\mathsf{T}} \mathrm{V}_N^{-1} \mathrm{G} \boldsymbol{\rho}_J + \mathbf{A}_N^{\mathsf{T}} \mathrm{V}_N^{-1} \mathbf{A}_N + \mu \rho_{J+1}^2 \\
&= \boldsymbol{\rho}_J^{\mathsf{T}} (\mathrm{M}_J + \mathrm{G}^{\mathsf{T}} \mathrm{V}_N^{-1} \mathrm{G}) \boldsymbol{\rho}_J + 2(\rho_{J+1} \mathbf{m}_J^{\mathsf{T}} - \mathbf{A}_N^{\mathsf{T}} \mathrm{V}_N^{-1} \mathrm{G}) \boldsymbol{\rho}_J + (\mathbf{A}_N^{\mathsf{T}} \mathrm{V}_N^{-1} \mathbf{A}_N + \mu \rho_{J+1}^2) \qquad (21)
\end{aligned}$$

This is a quadratic expression in $\boldsymbol{\rho}_J$. The last term is independent of $\boldsymbol{\rho}_J$ so can be taken out of the integral, allowing us to write equation 15 as

$$P(\rho_{J+1}|\mathbf{A}_N) = \frac{1}{Z} \exp\left[-\frac{1}{2}\mathbf{A}_N^{\mathsf{T}} \mathrm{V}_N^{-1} \mathbf{A}_N\right] \exp\left[-\frac{1}{2}\mu \rho_{J+1}^2\right] \int_{\boldsymbol{\rho}_J} \exp\left[-\frac{1}{2}\boldsymbol{\rho}_J^{\mathsf{T}} \mathrm{R}_J \boldsymbol{\rho}_J + \mathbf{b}_J^{\mathsf{T}} \boldsymbol{\rho}_J\right] d\boldsymbol{\rho}_J \qquad (22)$$

where

$$\begin{aligned}
\mathrm{R}_J &= \mathrm{M}_J + \mathrm{G}^{\mathsf{T}} \mathrm{V}_N^{-1} \mathrm{G} \qquad (23) \\
\mathbf{b}_J^{\mathsf{T}} &= \mathbf{A}_N^{\mathsf{T}} \mathrm{V}_N^{-1} \mathrm{G} - \rho_{J+1} \mathbf{m}_J^{\mathsf{T}} \qquad (24) \\
\mathbf{b}_J &= \mathrm{G}^{\mathsf{T}} \mathrm{V}_N^{-1} \mathbf{A}_N - \rho_{J+1} \mathbf{m}_J \qquad (25)
\end{aligned}$$

The integral, performed over the whole space, is a standard one

$$\int_{\boldsymbol{\rho}_J} \exp\left[-\frac{1}{2}\boldsymbol{\rho}_J^{\mathsf{T}} \mathrm{R}_J \boldsymbol{\rho}_J + \mathbf{b}_J^{\mathsf{T}} \boldsymbol{\rho}_J\right] d\boldsymbol{\rho}_J = \frac{(2\pi)^{J/2}}{|\mathrm{R}_J|^{1/2}} \exp\left[\frac{1}{2}\mathbf{b}_J^{\mathsf{T}} \mathrm{R}_J^{-1} \mathbf{b}_J\right] \qquad (26)$$

provided $|\mathrm{R}_J| > 0$. Using this we can write equation 22 as

$$\begin{aligned}
P(\rho_{J+1}|\mathbf{A}_N) &= \frac{1}{Z} e^{-\phi/2} \quad \text{where} \\
\phi &= \mu \rho_{J+1}^2 - \mathbf{b}_J^{\mathsf{T}} \mathrm{R}_J^{-1} \mathbf{b}_J \qquad (27)
\end{aligned}$$

and we have absorbed all factors which do not depend on ρ_{J+1} into the normalization constant. Substituting for $\mathbf{b}_J$ this becomes

$$\begin{aligned}
\phi &= \mu \rho_{J+1}^2 - (\mathbf{A}_N^{\mathsf{T}} \mathrm{V}_N^{-1} \mathrm{G} - \rho_{J+1} \mathbf{m}_J^{\mathsf{T}}) \mathrm{R}_J^{-1} (\mathrm{G}^{\mathsf{T}} \mathrm{V}_N^{-1} \mathbf{A}_N - \rho_{J+1} \mathbf{m}_J) \\
&= -\mathbf{A}_N^{\mathsf{T}} \mathrm{V}_N^{-1} \mathrm{G} \mathrm{R}_J^{-1} \mathrm{G}^{\mathsf{T}} \mathrm{V}_N^{-1} \mathbf{A}_N + (\mu - \mathbf{m}_J^{\mathsf{T}} \mathrm{R}_J^{-1} \mathbf{m}_J) \rho_{J+1}^2 + \\
&\qquad \mathbf{A}_N^{\mathsf{T}} \mathrm{V}_N^{-1} \mathrm{G} \mathrm{R}_J^{-1} \rho_{J+1} \mathbf{m}_J + \rho_{J+1} \mathbf{m}_J^{\mathsf{T}} \mathrm{R}_J^{-1} \mathrm{G}^{\mathsf{T}} \mathrm{V}_N^{-1} \mathbf{A}_N \\
&= -\mathbf{A}_N^{\mathsf{T}} \mathrm{V}_N^{-1} \mathrm{G} \mathrm{R}_J^{-1} \mathrm{G}^{\mathsf{T}} \mathrm{V}_N^{-1} \mathbf{A}_N + (\mu - \mathbf{m}_J^{\mathsf{T}} \mathrm{R}_J^{-1} \mathbf{m}_J) \rho_{J+1}^2 + 2\mathbf{A}_N^{\mathsf{T}} \mathrm{V}_N^{-1} \mathrm{G} \mathrm{R}_J^{-1} \mathbf{m}_J \rho_{J+1} \ . \qquad (28)
\end{aligned}$$

The first term does not depend on ρ_{J+1} so can be absorbed into the normalization constant. Putting this into equation 27 gives us

$$P(\rho_{J+1}|\mathbf{A}_N) = \frac{1}{Z}\exp\left[-\frac{1}{2}\alpha\rho_{J+1}^2 - \beta\rho_{J+1}\right] \tag{29}$$

where

$$\begin{aligned}\alpha &= \mu - \mathbf{m}_J^{\mathsf{T}}\mathrm{R}_J^{-1}\mathbf{m}_J \\ \beta &= \mathbf{A}_N^{\mathsf{T}}\mathrm{V}_N^{-1}\mathrm{G}\mathrm{R}_J^{-1}\mathbf{m}_J \, .\end{aligned} \tag{30}$$

By completing the square in the exponent

$$-\frac{1}{2}\alpha\rho_{J+1}^2 - \beta\rho_{J+1} = -\frac{\alpha}{2}\left(\rho_{J+1} + \frac{\beta}{\alpha}\right)^2 + \frac{\beta^2}{2\alpha} \tag{31}$$

and taking the term $\exp(\frac{\beta^2}{2\alpha})$ outside of the integral (where we may absorb it into the) normalization constant, we see that $P(\rho_{J+1}|\mathbf{A}_N)$ is just a Gaussian with mean $-\beta/\alpha$ and variance $1/\alpha$, i.e.

$$P(\rho_{J+1}|\mathbf{A}_N) = \sqrt{\frac{\alpha}{2\pi}}\exp\left[-\frac{\alpha}{2}\left(\rho_{J+1} + \frac{\beta}{\alpha}\right)^2\right] \, . \tag{32}$$

This is our analytic solution.[3] In order to calculate the 1D PDF over any single ρ_{J+1} for position $\mathbf{r}_{J+1}$, we must evaluate all of the matrices and vectors for that given point. The terms in the exponent (apart from ρ_{J+1}) comprise two types. The first depend only on the fixed data. These are $\mathbf{A}_N$ and G. They therefore only need to be calculated once for any number of new points. The other terms involve the covariance function, or more specifically the components of the inverted covariance matrix in equation 19. These are V_N, M_J (which is part of R_J), $\mathbf{m}_J$, and μ. As they depend on $\mathbf{r}_{J+1}$ they must be calculated for every new point. The matrix R_J has to be inverted every time and as this takes time $\mathcal{O}(J^3)$, this is potentially very slow. In the next section we will see how we can accelerate this.

It is clear from this that the PDF is only normalizable if $\alpha > 0$. Presumably it must be, although it is not obvious that this is enforced by the positive semi-definite property of C_J.[4] Numerical errors might cause it to be negative in practice.

5.2 Accelerating the matrix evaluations via matrix identities

Let us first consider the speed of operations. Multiplying a matrix of size $J \times J$ with a vector of size $J \times 1$, or taking the outer product of two such vectors, takes time $\mathcal{O}(J^2)$. Multiplying two $J \times J$ matrices takes time $\mathcal{O}(J^3)$ (unless one of them is diagonal, in which case it is $\mathcal{O}(J^2)$). A chain of three matrix multiplications takes time $\mathcal{O}(2J^3)$, but we will neglect constant numeric factors in these order of magnitude estimates.

[3] One can check that the covariance elements enter correctly into expressions, and that the dimensions agree (the argument of the exponent must have no dimensions).

[4] We can investigate this by referring to the version of our solution in equation 42 in section 5.2. C_J is positive semi-definite (PSD), and therefore C_J^{-1} is too. It follows that $\mathbf{k}_J^{\mathsf{T}}\mathrm{C}_J^{-1}\mathbf{k}_J > 0$ provided $\mathbf{k}_J \neq 0$ (which it will be with a covariance matrix with infinite support). But in general we do not know the size of this compared to k, so we cannot be sure that $\mu > 0$. Any principal submatrix of a PSD-matrix is also PSD. Therefore M_J and hence M_J^{-1} are PSD. If X is PSD, then $X\mathrm{M}_J^{-1}X$ is too. If $J = N$, then G is diagonal and all elements are positive, so it is PSD, and hence $\mathrm{G}\mathrm{M}_J^{-1}\mathrm{G}^{\mathsf{T}}$ is also PSD. The sum of two PSD matrices is also PSD. It follows from all of this that when $J = N$, R_J^{-1} is PSD. Hence $\mathbf{m}_J^{\mathsf{T}}\mathrm{R}_J^{-1}\mathbf{m}_J > 0$. But this still does not tell us whether it is smaller than μ and thus whether $\alpha > 0$.

Inverting the matrix also takes time $\mathcal{O}(J^3)$. Multiplying a $J \times J$ matrix with a $J \times N$ matrix takes time $\mathcal{O}(J^2 N)$.

We first need to relate the components of the inverted covariance matrix in equation 19 to the components of the covariance matrix itself, which we partition as

$$\mathrm{C}_{J+1} = \begin{bmatrix} \mathrm{C}_J & \mathbf{k}_J \\ \mathbf{k}_J^\mathsf{T} & k \end{bmatrix} . \tag{33}$$

C_J is the $J{\times}J$ covariance matrix involving only the fixed data. $\mathbf{k}_J$ is the $J{\times}1$ vector with elements given by $c_{j,J+1}$, i.e. the covariance between the fixed data and the new point. k is the scalar $c_{J+1,J+1}$, the covariance of the new point with itself (i.e. its variance). Recall that each element of the covariance matrix is determined by the covariance function, such as equation 10.

Given the partitioning defined in equations 19 and 33, a standard result of matrix algebra (e.g. Press et al. 1992) is that the components of the inverted matrix can be written as

$$\mathrm{M}_J = \mathrm{C}_J^{-1} + (\mathrm{C}_J^{-1}\mathbf{k}_J)(k - \mathbf{k}_J^\mathsf{T}\mathrm{C}_J^{-1}\mathbf{k}_J)^{-1}(\mathbf{k}_J^\mathsf{T}\mathrm{C}_J^{-1}) \tag{34}$$

$$\mathbf{m}_J = -(\mathrm{C}_J^{-1}\mathbf{k}_J)(k - \mathbf{k}_J^\mathsf{T}\mathrm{C}_J^{-1}\mathbf{k}_J)^{-1} \tag{35}$$

$$\mu = (k - \mathbf{k}_J^\mathsf{T}\mathrm{C}_J^{-1}\mathbf{k}_J)^{-1} . \tag{36}$$

Some of the parentheses are used to show the common elements (they are otherwise unnecessary, because matrix multiplication is associative). Defining

$$\mathbf{h}_J = \mathrm{C}_J^{-1}\mathbf{k}_J \tag{37}$$

we can write the matrix and vector parts of the inverted covariance matrix as

$$\mathrm{M}_J = \mathrm{C}_J^{-1} + \mu\mathbf{h}_J\mathbf{h}_J^\mathsf{T} \tag{38}$$

$$\mathbf{m}_J = -\mu\mathbf{h}_J \tag{39}$$

Note that $\mathbf{h}_J\mathbf{h}_J^\mathsf{T}$ is an outer product, so produces a matrix of size $J \times J$. We now use the matrix inversion lemma (also known as the Woodbury matrix identity; e.g. Press et al. 1992) to write, from equation 23,

$$\begin{aligned} \mathrm{R}_J^{-1} &= (\mathrm{M}_J + \mathrm{G}^\mathsf{T}\mathrm{V}_N^{-1}\mathrm{G})^{-1} \\ &= \mathrm{M}_J^{-1} - \mathrm{M}_J^{-1}\mathrm{G}^\mathsf{T}(\mathrm{V}_N + \mathrm{G}\mathrm{M}_J^{-1}\mathrm{G}^\mathsf{T})^{-1}\mathrm{G}\mathrm{M}_J^{-1} . \end{aligned} \tag{40}$$

The Sherman–Morrison formula (which is a special case of the matrix inversion lemma) allows us to write

$$\begin{aligned} \mathrm{M}_J^{-1} &= (\mathrm{C}_J^{-1} + \mu\mathbf{h}_J\mathbf{h}_J^\mathsf{T})^{-1} \\ &= \mathrm{C}_J - (1 + \mu\mathbf{h}_J^\mathsf{T}\mathrm{C}_J\mathbf{h}_J)^{-1}\mu\mathrm{C}_J\mathbf{h}_J\mathbf{h}_J^\mathsf{T}\mathrm{C}_J \\ &= \mathrm{C}_J - (1 + \mu\mathbf{k}_J^\mathsf{T}\mathrm{C}_J^{-1}\mathbf{k}_J)^{-1}\mu\mathbf{k}_J\mathbf{k}_J^\mathsf{T} \end{aligned} \tag{41}$$

where the last step has been obtained by substituting for $\mathbf{h}_J$ from equation 37.

Thus we can simplify our expressions for α and β in equation 30 using

$$\begin{aligned}
\mathrm{V}_N &= \Sigma_N + G\mathrm{C}_J G^\mathsf{T} \\
\mathrm{R}_J^{-1} &= \mathrm{M}_J^{-1} - \mathrm{M}_J^{-1}\mathrm{G}^\mathsf{T}(\mathrm{V}_N + \mathrm{G}\mathrm{M}_J^{-1}\mathrm{G}^\mathsf{T})^{-1}\mathrm{G}\mathrm{M}_J^{-1} \\
\mathrm{M}_J^{-1} &= \mathrm{C}_J - (1 + \mu\mathbf{k}_J^\mathsf{T}\mathrm{C}_J^{-1}\mathbf{k}_J)^{-1}\mu\mathbf{k}_J\mathbf{k}_J^\mathsf{T} \\
\mathbf{m}_J &= -\mu\mathrm{C}_J^{-1}\mathbf{k}_J \\
\mu &= (k - \mathbf{k}_J^\mathsf{T}\mathrm{C}_J^{-1}\mathbf{k}_J)^{-1} \\
k &= c_{J+1,J+1} \\
\mathbf{k}_J &= (c_{1,J+1}, c_{2,J+1}, \ldots, c_{J,J+1})^\mathsf{T}
\end{aligned} \tag{42}$$

C_J^{-1} takes time $\mathcal{O}(J^3)$ to compute, but must only be done once. If we were to truncate the covariance function to zero at larger distances, then C_J will be a sparse matrix, a fact which presumably can be exploited in its inversion.

The following must be calculated for every new point where we want to estimate the dust density. (We ignore the time taken to invert C_J.) $\mathrm{C}_J^{-1}\mathbf{k}_J$ takes time $\mathcal{O}(J^2)$ to compute, and therefore also the argument within the parentheses for μ. This is a scalar, so inverting it takes time $\mathcal{O}(0)$ to compute. Thus both μ and $\mathbf{m}_J$ take time $\mathcal{O}(J^2)$. The argument in the parentheses in the expressions for M_J^{-1} likewise takes time $\mathcal{O}(J^2)$ to compute (and an additional $\mathcal{O}(0)$ to invert). $\mathbf{m}_J^\mathsf{T}\boldsymbol{\rho}_J$ takes time $\mathcal{O}(J^2)$, and therefore so does M_J^{-1} overall. In the expression for R_J^{-1}, $\mathrm{G}\mathrm{M}_J^{-1}$ takes time $\mathcal{O}(NJ^2)$ to compute, so dominates over the computation of M_J^{-1}. The term $(\mathrm{V}_N + \mathrm{G}\mathrm{M}_J^{-1}\mathrm{G}^\mathsf{T})$ in R_J^{-1} is an $N \times N$ matrix so takes time $\mathcal{O}(N^3)$ to compute.[5] The multiplication of this with the other terms in that expression take time $\mathcal{O}(N^2J + NJ^2 + N^3) \sim \mathcal{O}(NJ^2)$ as $J \geq N$. The multiplications in equation 30 take no longer than this, so $\mathcal{O}(NJ^2)$ is the time required to compute the exponential argument for each new point by exact methods.

Note that when $J > N$, the calculation time is in principle dominated by the time required to calculate $\mathrm{G}\mathrm{M}_J^{-1}$, rather than to invert $(\mathrm{V}_N + \mathrm{G}\mathrm{M}_J^{-1}\mathrm{G}^\mathsf{T})$. However, some of these matrices are sparse, and this can be exploited to accelerate the operations. G will always be sparse in practice. If $J = N$ then it is even diagonal, in which case computing $\mathrm{G}\mathrm{M}_J^{-1}$ takes time $\mathcal{O}(J^2)$. In that case the time to compute equation 30 is $\mathcal{O}(J^3)$, and is dominated by the inversion of $(\mathrm{V}_N + \mathrm{G}\mathrm{M}_J^{-1}\mathrm{G}^\mathsf{T})$.

If we use a truncated covariance function, then both C_J and the outer product $\mathbf{k}_J\mathbf{k}_J^\mathsf{T}$ can be made sparse and so M_J^{-1} is sparse (but never diagonal). V_N is also sparse – it may even be diagonal.

5.3 Fast and approximate matrix operations

When matrices are sparse, knowledge of where the non-zero elements are can be exploited to accelerate matrix multiplication.

Matrices can be inverted by a number of exact methods, such as Gauss-Jordan elimination. Direct methods can run into numerical problems when the matrix is ill-conditioned (i.e. when the ratio of the largest to

[5] If we applied the matrix inversion lemma to this we get

$$(\mathrm{V}_N + \mathrm{G}\mathrm{M}_J^{-1}\mathrm{G}^\mathsf{T})^{-1} = \mathrm{V}_N^{-1} - \mathrm{V}_N^{-1}\mathrm{G}(\mathrm{M}_J + \mathrm{G}^\mathsf{T}\mathrm{V}_N^{-1}\mathrm{G})^{-1}\mathrm{G}^\mathsf{T}\mathrm{V}_N^{-1} \,. \tag{43}$$

But the term $(\mathrm{M}_J + \mathrm{G}^\mathsf{T}\mathrm{V}_N^{-1}\mathrm{G})$ takes time $\mathcal{O}(J^3)$ to invert, which is no faster then before.

smallest eigenvalues is very large). In that case indirect methods such as LU decomposition or singular valued decomposition (SVD) are preferred. There also exist approximate methods for inverting matrices. A common method is the conjugate gradient method (e.g. Press et al. 1992), which takes time $\mathcal{O}(J^2)$ and also allows us to estimate the accuracy of the inversion.

6 On-going work and open issues

1. Does equation 26 require R_J to be symmetric? (I think it is anyway: equation 23)
2. We need to ensure that $\alpha > 0$. It should be guaranteed in principle, but we need to check in practice.
3. Is a calculation time faster than $\mathcal{O}(NJ^2)$ with exact methods theoretically possible? (I think not.)
4. Can we accelerate the matrix operations (products and inverses) using approximate methods?
5. What numerical error issues do we have, and how do we mitigate them?
6. How shall we set the parameters in the covariance function?
7. What would be an appropriate (and permitted) truncated covariance function, i.e. one with compact support? Candidates include a truncated polynomial as described on p. 88 of Rasmussen & Williams (2006), or a cosine kernel suggested in Storkey (1999). Note that if point $J+1$ is so far from all data points such that $c_{i,J+1} = 0 \; \forall \; i$, then we cannot solve for that point.
8. With the Gaussian covariance function we could use different scale lengths in the radial and vertical directions (of the Galactic disk). More generally, if the ISM shows structure on different length scales, is there a more appropriate (“multi-scale”) covariance function?
9. How shall we determine the geometry? Is there any reason not to use $N = J$, i.e. one elongated cell per line-of-sight?
10. How do we deal with the negativity of the dust density? Or, can we (should we) include a positivity constraint? (Note that the integral used in equation 26 extends to minus infinity.)
11. How shall we introduce the existence of errors in the positions of the N stars, which primarily means in the distances? In principle we would need to introduce (co)variances for both the position vectors, $\mathbf{r}_j$, as well as the geometric factors, $g_{n,j}$, which means the elements of G. In the former case we might be able to accommodate this by replacing λ^2 in equation 10 with something like $\lambda^2 + \delta^2_{r_i} + \delta^2_{r_j}$. Introducing uncertainty into G looks like it would be difficult; or at least would sacrifice some of the simplicity of the model. But it may be important, because distance uncertainties are likely to be as large as extinction uncertainties, on which the likelihood is based.

References

Bailer-Jones, C.A.L., 2011, *3D extinction modelling*, Gaia technical note, CBJ-060 (draft)

Gibbs M.N., 1997, *Bayesian Gaussian processes for regression and classification*, PhD thesis, University of Cambridge, `http://www.inference.phy.cam.ac.uk/mng10/GP/`

Press et al., 1992, *Numerical Recipes*, CUP

Rasmussen C.E., Williams C.K.I., 2006, *Gaussian processes for machine learning*, MIT Press, `http://www.GaussianProcess.org/gpml`

Storkey A.J, 1999, *Truncated covariance matrices and Toeplitz methods in Gaussian processes*, `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.33.8263`