Deselby distributions: A family of distributions for probabilistic agent based modelling

Daniel Tang

Leeds Institute for Data Analytics

Abstract

In the accompanying "Mathematical foundations" document I described how a broad class of agent based models can be described in terms of creation and annihilation operators. Here I describe a probability distribution that has a particularly simple transformation when operated on by the creation and annihilation operators. I show how this can be used to perform data assimilation on agent based models.

1 The discrete Deselby distribution

When the state of an agent in an ABM takes on a finite number of discrete states, such as in a compartmental model, then it is convenient to represent the state of the whole system in terms of the number of agents in each state (the "occupation number")[1]. So, a probability distribution over system states is a distribution over the space of mappings from agent state to occupation number[2]. If we label the agent states $1...N$ and let $k_1...k_N$ be the occupation numbers of each state, then we define the discrete Deselby distribution to be

$$D_\Delta(k_1...k_N) = \prod_{i=1}^{N}(k_i)_{\Delta_i}\frac{\lambda_i^{k_i-\Delta_i}e^{-\lambda_i}}{k_i!} \qquad (1)$$

where $\Delta = \langle \Delta_1,...,\Delta_N \rangle$ is a vector of integers and we use the notation $(k_i)_{\Delta_i}$ to signify the falling factorial.

Note that D_0 is just a multivariate Poisson distribution.

---

[1]This representation is preferable to an ordered list of states, one for each agent, since it does not impose an arbitrary ordering on the agents

[2]This is a Fock space

1.1 Discrete Deselby under creation and annihilation operators

For simplicity of notation, we will first consider a set of agents with just one state, then extend to agents with many states.

The creation and annihilation operators should, by definition, have the following properties:

$$(a^\dagger D_\Delta)(k) = D_\Delta(k-1) \qquad (2)$$

$$(a^\dagger D_\Delta)(0) = 0 \qquad (3)$$

and

$$(aD_\Delta)(k) = (k+1)D_\Delta(k+1) \qquad (4)$$

Starting with the creation operator, from equation 1

$$D_\Delta(k-1) = (k-1)_\Delta\frac{\lambda^{k-1-\Delta}e^{-\lambda}}{(k-1)!}$$

$$= \frac{(k)_{\Delta+1}}{k}\frac{\lambda^{k-(\Delta+1)}e^{-\lambda}}{k^{-1}k!} = D_{\Delta+1}(k)$$

and the falling factorial ensures that $D_{\Delta+1}(0) = 0$.

Similarly for the annihilation operator

$$(k+1)D_\Delta(k+1) = (k+1)(k+1)_\Delta\frac{\lambda^{k+1-\Delta}e^{-\lambda}}{(k+1)!}$$

$$= (k+1)(k)_{\Delta-1}\frac{\lambda^{k-(\Delta-1)}e^{-\lambda}}{k!}$$

but

$$(k+1)(k)_{\Delta-1} = (k-(\Delta-1)+\Delta)(k)_{\Delta-1} = (k)_\Delta + \Delta(k)_{\Delta-1}$$

so

$$(k+1)D_\Delta(k+1) = ((k)_\Delta + \Delta(k)_{\Delta-1})\frac{\lambda^{k-(\Delta-1)}e^{-\lambda}}{k!}$$

$$= \lambda D_\Delta(k) + \Delta D_{\Delta-1}(k)$$

So, under creation and annihilation operators, the Deselby distributions transform according to

$$a^\dagger D_\Delta = D_{\Delta+1} \qquad (5)$$

and

$$a D_\Delta = \lambda D_\Delta + \Delta D_{\Delta-1} \qquad (6)$$

It can be seen immediately that this can be extend to the multivariate case:

$$a_i^\dagger D_\Delta = D_{\langle \Delta_1 \ldots \Delta_i+1 \ldots \Delta_N \rangle} \qquad (7)$$

and

$$a_i D_\Delta = \lambda_i D_\Delta + \Delta_i D_{\langle \Delta_1 \ldots \Delta_i-1 \ldots \Delta_N \rangle} \qquad (8)$$

1.2   Lambda optimisation

Using the above realationships, we can express the rate of change of the distribution over system states in terms of a sum of Deselby distributions. So upon integration, the distribution of the system will always be exactly expressible as a sum of Deselby distributions. However, because the annihilation operator results in a sum of two distributions, the number of non-negligible terms will, in the worst case, grow with $O(t^N)$ where N is the number of agent states.

The rate of growth in the number of terms can be slowed by absorbing some of the rate of change of the higher order terms (i.e. the terms with Δs with a higher L_1 norm) into a rate of change of the λ_is.

From equation 1, in the monovariate case

$$\frac{\partial D_\Delta}{\partial \lambda} = \left(\frac{k-\Delta}{\lambda} - 1 \right) (k)_\Delta \frac{\lambda^{k-\Delta} e^{-\lambda}}{k!}$$

$$= \left(\frac{(k)_{\Delta+1}}{\lambda} - (k)_\Delta \right) \frac{\lambda^{k-\Delta} e^\lambda}{k!}$$

$$= D_{\Delta+1} - D_\Delta$$

Numerical experiments show that minimising the cartesian norm of the coefficients of the rate of change is quite effective at reducing the rate of growth of the number of non-negligible terms (i.e. the number of terms with probability above some small cutoff value).

Note that, in the case that there is no interaction between agents, the number of λ_is is the same as the number of higher order terms in the rate of change of a single Deselby disrtibution, so if the system starts in a pure Deselby distribution (e.g. D_0, the Poisson state) we can use Lambda optimisation to maintain the number of terms at 1, and integrate the system state over time analytically.

1.3   SIR model with data assimilation

Assimilation of observations into a model proceeds using Baye's formula:

$$P(S|O) = \frac{P(O|S)P(S)}{P(O)}$$

where S is the state of the system, O is the observation, $P(S)$ is the prior over the states, which in the case of data assimilation is the forecast distribution, and $P(O|S)$ is the likelihood of the observation, which is usually relatively easy to compute. $P(O)$ is a normalising factor that is often ignored.

As an example of data assimilation with Deselby distributions, we ran numerical experiments with a compartmental SIR epidemiological model. The observation to assimilate was the number of hospital admissions. If we assume that each agent, upon infection, has a fixed probability, r, of being hospitalised, then the likelihood function is just the binomial distribution. Multiplying a Deselby distribution by a Binomial gives

$$P(k|m) \propto \frac{k!}{m!(k-m)!} r^m (1-r)^{k-m} (k)_\Delta \frac{\lambda^{(k-\Delta)} e^{-\lambda}}{k!}$$

which, after some rearranging and dropping of constants gives

$$P(k|m) \propto (1-r)^\Delta (k)_m D_{\Delta,(1-r)\lambda}(k) \qquad (9)$$

where we now make explicit the λ dependency of $D_{\Delta,\lambda}$.

The product of a Deselby distribution and a falling factorial in k can be caluated from the product of two falling factorials, which is given by

$$(k)_m (k)_n = \sum_{q=0}^{min(m,n)} \binom{m}{q} \binom{n}{q} q! (k)_{m+n-q}$$

So

$$(k)_m D_\Delta = \sum_{q=0}^{min(m,\Delta)} \frac{m!\Delta!\lambda'^{(m-q)}}{q!(m-q)!(\Delta-q)!} D_{m+\Delta-q}$$

where $\lambda' = (1-r)\lambda$. The multiplying factors can be efficiently computed using a recursive form. Let

$$c_q = \frac{m!\Delta!((1-r)\lambda)^{(m-q)}}{q!(m-q)!(\Delta-q)!}$$

then

$$c_0 = ((1-r)\lambda)^m$$

and

$$c_{q+1} = \frac{(m-q)(\Delta-q)}{(q+1)(1-r)\lambda} c_q$$

Substituting this back into equation 9 gives the final form of the posterior

$$P(k|m) \propto (1-r)^\Delta \sum_{q=0}^{\min(\Delta,m)} c_q D_{\Delta+m-q,(1-r)\lambda} \qquad (10)$$

Interestingly, this is equivalent to

$$P(k|m) \propto (1-r)^\Delta a^{\dagger m} a^m D_{\Delta,(1-r)\lambda} \qquad (11)$$

Numerical experiemtns showed that data assimilation into an SIR model using this technique was faster than using Metropolis-Hastings MCMC for the same level of accuracy.

2 The Deselby distribution in the continuous limit

Suppose the agent state consists of a real number in the interval $[0,1]$. We can deal with this by taking the Deselby distribution in the limit of infinite dimensions.

Starting with the N dimensional Deselby distribution

$$D_\Delta(k_{1...N}) = \prod_{i=1}^N (k_i)_{\Delta_i} \frac{\lambda_i^{k_i-\Delta_i} e^{-\lambda_i}}{k_i!}$$

Let $\delta x = \frac{1}{N}$, $\lambda_i = \rho_i \delta x$ and $K = \langle k_1...k_N \rangle$ then

$$D_\Delta(K) = \prod_{i=1}^N (k_i)_{\Delta_i} \frac{\rho_i^{k_i-\Delta_i} e^{-\rho_i \delta x}}{k_i!} \delta x^{\|K\|-\|\Delta\|} \qquad (12)$$

where $\|\langle x_1...x_N \rangle\| = \sum_i x_i$

In the limit that δx tends to zero, but the number of agents remains finite, K and Δ become finite multisets over the real numbers and our interest shifts to an integral over Deselby distributions

$$P(K)dx^{\|K\|} = \int_\Delta \tau(\Delta) D_\Delta(K) dx^{\|\Delta\|}$$

where the integral is over all finite multisets and $\tau(\Delta)$ is a PDF over the Deselby distributions.

However, if $\Delta \not\subseteq K$ then

$$\prod_i (k_i)_{\Delta_i} = 0$$

so we need only sum over Δs that are a subset of K, leaving us with

$$P(K) = \sum_{\Delta \subseteq K} \tau(\Delta) D_\Delta(K)$$

where we define a subset of a multiset in the obvious way.

Moreover, if we have m agents and N states, there are $\frac{N^m}{m!}$ possible occupancy states. However, $\binom{N}{m}$ of these states have no occupancy above one. So, the proportion of states with no multiple occupancy is $\frac{N!}{(N-m)!N^m}$. For large N we can approximate this using Stirling's approximation

$$\ln\left(\frac{N!}{(N-m)!N^m}\right) \approx (N-m)(\ln(N) - \ln(N-m)) - m$$

$$\approx -\frac{m^2}{2N}$$

So, as N tends to infinity, the proportion of states that have no multiple occupancy tends to one.

So, we can restrict our interest to the set of states that have occupancy no greater than one.

2.1 Creation and annihilation operators

From equations 12 and 2 to 4 we have, similarly to before

$$a_x^\dagger D_\Delta = D_{(\Delta \cup x)}$$

and

$$a_x D_\Delta = \rho(x) D_\Delta \delta x + (x \in \Delta) D_{(\Delta \setminus x)}$$

where $x \in \Delta$ gives the number of xs in Δ

This translates to operations on $\tau(\Delta)$

$$(a_x^\dagger \tau)(\Delta)\delta x = \delta_\Delta(x) \tau(\Delta \setminus x)\delta x$$

where δ_Δ is a sum of Dirac delta functions

$$\delta_\Delta(x) = \sum_{y \in K} \delta(x-y)$$

and, to first order in δx

$$(a_x \tau)(\Delta) = (\rho(x)\tau(\Delta) + \tau(\Delta \cup x))\,\delta x$$

Now, in the discrete case, when we wanted to apply a creation operator, we just supplied a single state to create, but in the continuous case we need to supply a PDF over the agent states in order to define which states the operator applies to, and we need to integrate over all these states. For annihilation, we should multiply by a state-dependent rate and integrate over all states.