clustr - iPhone app

Contents

1 General Overview **1**

2 Spatial Distribution **2**

3 Temporal Distribution **3**

4 Estimating N **5**

5 Putting it all together **6**
5.1 Cumulative Probability 6
5.2 Relative Probability Density 7

6 To Merge or Not To Merge **7**
6.1 Spatial Distribution 7
6.2 Temporal Distribution 8
6.3 Subtleties 10

7 Parameter Values **10**

8 Simulation Results **11**

9 Future Directions **11**
9.1 Modifying the Model 11
9.2 Additional Signals 11

10 References **13**

The clustr iPhone app will allow users to take photos and send messages (and possibly also simply announce their presence). The backend will then attempt to cluster the various contributions into clusters organized by which event they correspond to. This clustering will, in a first draft, take into consideration the temporal and spatial location of the various contributions. This document contains within it the approach that we will take in performing this clustering.

1 General Overview

Following an event there will be multiple contributions spread out over both space and time. We need to assume a certain form for the probability of a contribution at some space-time point $(t, \vec{r})$ given that an event occurs at $(\tau, \vec{\mu})$. After working with these expressions for a while, it has become very clear that in order to allow for analytic tractability, some simplifying assumptions must be made:

(i.) The distribution in space follows a two-dimensional Gaussian with mean (μ_x, μ_y), variance σ_x^2, σ_y^2 and vanishing correlation, $\sigma_{xy} = 0$. In principle we want to allow for nontrivial correlation since we can then classify interesting events that are not symmetric (e.g. parades). However, by considering symmetric situations we simplify the problem considerably. Note that a truly symmetric event would also have $\sigma_x = \sigma_y$, however this additional constraint makes the calculation below difficult and so at this initial point we allow them to vary independently.

(ii.) The distribution in time follows an exponential distribution which begins at time τ and has a decay rate of λ. In principle, one would assume that it takes some amount of time for the first contribution to happen because of reaction times etc, but by choosing a simple exponential distribution rather than a more accurate one (e.g. a recinormal distribution) we simplify the calculation considerably once again.

(iii.) There are many parameters for these models which we do not know a priori $(\mu_x, \mu_y, \sigma_x, \sigma_y, \tau, \lambda)$. In addition to these parameters the total number of contributions (which we shall call N) is also very important. The reason for this is that the temporal distribution of points is naturally ordered. As a result in order to do any kind of Bayesian analysis, we must treat N as a parameter. To see this, consider the case of three contributions occurring at times $0, 1, 2$ seconds. If the decay rate is very large, you'd expect the vast majority of the contributions to happen right away. As a result, if there are only three contributions in total, one could interpret the above data as indicating a large decay rate (otherwise they would be much more spread out). However, if we know that these are only the first three out of say one million contributions then it would be reasonably to assume a very small decay rate (otherwise you should have seen a ton of points already). So, in order to estimate the parameters one must know how many points there are. **Unfortunately, considering N a parameter in the analysis turns out to be impossibly difficult.** As a result, I have to press on with the assumption that N is known and find some reasonable way to estimate it at the end.

The general algorithm will them proceed as follows. Each time a new contribution is added to a cluster, we need to update our beliefs about the parameters (what are their most likely values? How certain are we of them?). Then, each time a new contribution appears we must consider all nearby clusters and compute the probability that this new point was generated from the same distribution. Then, given these probabilities we decide where to place the new point. If this largest probability is below some tunably threshold we don't add it at all but instead create a new cluster for it.

Finally, suppose that for some reason, two clusters were created even though they are part of the same event (perhaps the two first contributions occurred very far apart). In this case we may want to merge the two in the future. In order to determine if this should happen, we must look at the two clusters and compute the likelihood that the data in them was actually generated from the same distribution. If this probability surpasses some threshold we go ahead and merge them.

Throughout all of this analysis we must ensure that the relevant algorithms are sufficiently fast so that Parse won't complain.

2 Spatial Distribution

The spatial distribution is presumed to be a multivariate Gaussian. We are ultimately interested in the distribution $p(\vec{r}_{k+1}|\vec{r}_1, \ldots, \vec{r}_k)$. In order to compute this, we consider all possible parameters $\mu_x, \mu_y, \sigma_x, \sigma_y$ and compute the probability $p(\vec{r}_{k+1}|\mu_x, \mu_y, \sigma_x, \sigma_y, \vec{r}_1, \ldots, \vec{r}_k)$ and then integrate over all the possibly parameter sets, weighted by their posterior probability:

$$p(\vec{r}_{k+1}|\vec{r}_1, \ldots, \vec{r}_k) = \int p(\vec{r}_{k+1}|\mu_x, \mu_y, \sigma_x, \sigma_y, \vec{r}_1, \ldots, \vec{r}_k) p(\mu_x, \mu_y, \sigma_x, \sigma_y|\vec{r}_1, \ldots, \vec{r}_k) d\mu_x\, d\mu_y\, d\sigma_x\, d\sigma_y$$

Under the assumption that the points are generated independently from each other, we have

$$p(\vec{r}_{k+1}|\vec{r}_1, \ldots, \vec{r}_k) = \int p(\vec{r}_{k+1}|\mu_x, \mu_y, \sigma_x, \sigma_y) p(\mu_x, \mu_y, \sigma_x, \sigma_y|\vec{r}_1, \ldots, \vec{r}_k) d\mu_x\, d\mu_y\, d\sigma_x\, d\sigma_y$$

The first factor here is simply the Gaussian distribution itself while the second one is the posterior distribution over the parameters. Using Bayes' rule, we have

$$p(\mu_x, \mu_y, \sigma_x, \sigma_y|\vec{r}_1, \vec{r}_2, \ldots \vec{r}_k) = \frac{1}{\mathcal{N}} p(\vec{r}_1, \vec{r}_2, \ldots \vec{r}_k|\mu_x, \mu_y, \sigma_x, \sigma_y) p(\mu_x, \mu_y, \sigma_x, \sigma_y)$$

The final factor here is the prior that we place on the parameters. Since the event can happen anywhere in the world, we will have a completely uninformative prior on μ_x, μ_y. However, the variance sets the size of the event and will therefore have a nontrivial prior. Since we're assuming that the x, y components are generated independently we need only consider a univariate Gaussian analysis. As a result, we want to compute the posterior probability $p(\mu, \sigma|D)$ where D signifies the set of data observed so far, $x_1, \ldots, x_n$.

There are three well-studied versions of Bayesian inference of the Gaussian: unknown mean, unknown variance, and unknown mean and variance. We are interested in the latter of these. In this case, the conjugate prior is a Normal-Gamma distribution. Pulling from the third reference below, the posterior distribution (for a one-dimensional Gaussian) is given by

$$\begin{aligned}
p(\mu, \lambda|D) &= NG(\mu, \lambda|\mu_n, \kappa_n, \alpha_n, \beta_n) \\
\mu_n &= \frac{\kappa_0\mu_0 + n\bar{x}}{\kappa_0 + n} \\
\kappa_n &= \kappa_0 + n \\
\alpha_n &= \alpha_0 + n/2 \\
\beta_n &= \beta_0 + \frac{1}{2}\sum_{i=1}^{n}(x_i - \bar{x})^2 + \frac{\kappa_0 n(\bar{x} - \mu_0)^2}{2(\kappa_0 + n)}
\end{aligned}$$

Here the parameters $\mu_0, \kappa_0, \alpha_0, \beta_0$ are the parameters governing the priors over the mean and precision ($\lambda = 1/\sigma^2$). Note that the definition of the NG distribution is as follows:

$$NG(\mu, \lambda|\mu_0, \kappa_0, \alpha_0, \beta_0) = \mathcal{N}(\mu|\mu_0, (\kappa_0\lambda)^{-1})\Gamma(\lambda|\alpha_0, \beta_0)$$

Finally what we are really interested in is not so much what the parameters are, but rather what the likelihood of a new observation is within a certain cluster. With this information, we can clearly classify an event into the appropriate cluster. Integrating out the parameters, we are (according to eqn 100 in the third reference below) left with a t-distribution:

$$p(x_{new}|x_1, \ldots x_n) = t_{2\alpha_n}(x|\mu_n, \frac{\beta_n(\kappa_n + 1)}{\alpha_n\kappa_n})$$

Note that an uninformative prior on μ requires us to set $\kappa_0 = 0$. In this case, we're left with

$$\boxed{\begin{aligned}
p(x_{new}|x_1, \ldots x_n) &= t_{2\alpha_n}\left(x|\bar{x}, \frac{n+1}{n}\frac{\beta_n}{\alpha_n}\right) \\
\alpha_n &= \alpha_0 + n/2 \\
\beta_n &= \beta_0 + \frac{1}{2}\sum_{i=1}^{n}(x_i - \bar{x})^2
\end{aligned}}$$

Perhaps a way to force the x, y dimensions to be treated symmetrically would be to simply keep track of one α_n and one β_n and allow both x, y to contribute to them.

3 Temporal Distribution

The temporal distribution of timestamps is presumed to be exponential

$$p(t|\tau, \lambda) = \lambda e^{-\lambda(t-\tau)}\theta(t - \tau)$$

This seems like a reasonable (yet simple) model: the contributions spike immediately following the event and then decay exponentially. By tuning the decay rate, λ, we can consider both very short events and long events (like concerts). We are ultimately interested in answering the same question as for the spatial distribution - given the first k observations, what is the probability that the next one has a certain value: $p(t_{k+1}|t_1, \ldots, t_k)$. However, this time there are some interesting aspects that we need to take into consideration. First of all, the data points that we observe are not independent of each other since they are naturally sorted (the smallest time is revealed to us first followed by the second smallest and so on).

Contrast this with the spatial distribution for which each new point can be either farther away or closer to the mean.

As before, we will compute

$$p(t_{k+1}|t_1,\dots,t_k) = \int p(t_{k+1}|\tau,\lambda,t_1,\dots,t_k)p(\tau,\lambda|t_1,\dots,t_k)d\tau d\lambda$$

Once again, the final factor above is the posterior distribution over the parameters given the first k observations. In order to compute this, we will once again use Bayes' theorem

$$p(\tau,\lambda|t_1,\dots,t_k) = \frac{1}{N}p(t_1,\dots,t_k|\tau,\lambda)p(\tau,\lambda|\emptyset)$$

We will decide on a prior distribution, $p(\tau,\lambda|\emptyset)$ later, so for now let's focus on the likelihood $p(t_1,\dots,t_k|\tau,\lambda)$. In order to address this problem, we must assume that there are a fixed number of contributions, N. Clearly $N \geq k$ since we've already observed k contributions. In order for the first k contributions to occur at times $t_1,\dots,t_k$, the remaining $N-k$ contributions must all occur at times $t > t_k$. The probability for any given datapoint to occur in this interval is

$$\tilde{\Gamma}(t_k) = \int_{t_k}^{\infty} \lambda e^{-\lambda(t-\tau)}\theta(t-\tau)dt = e^{-\lambda(max(t_k-\tau,0))}$$

Then, since $N-k$ points have to fall in this interval while the other k points occur at $t_1,\dots,t_k$ we have (excluding an overall normalization constant)

$$\begin{aligned}
p(t_1,\dots,t_k|\tau,\lambda) &\propto \lambda^k e^{-\lambda(t_k-\tau)(N-k)}e^{-\lambda(t_1+\ldots+t_k-k\tau)}\theta(t_1-\tau) \\
&= \lambda^k e^{-\lambda\{(t_k-\tau)(N-k)+k\bar{t}-k\tau\}}\theta(t_1-\tau) \\
&= \lambda^k e^{-\lambda\{N(t_k-\tau)+k(\bar{t}-t_k)\}}\theta(t_1-\tau)
\end{aligned}$$

Since we don't have any prior information on when this particular event occurred, we must assume a completely non-informative prior on τ. However, for the decay rate, λ, we will assume a gamma distribution

$$p(\tau,\lambda|\emptyset) \propto \lambda^\alpha e^{-\beta\lambda}$$

Putting these together, we then have

$$\boxed{p(\tau,\lambda|t_1,\dots,t_k) \propto \lambda^{k+\alpha}e^{-\lambda\{N(t_k-\tau)+k(\bar{t}-t_k)+\beta\}}\theta(t_1-\tau)}$$

Now, we want to determine the other ingredient in the predictive distribution above, $p(t_{k+1}|\tau,\lambda,t_1,\dots,t_k)$. In order for the $(k+1)$th contribution to occur at time t_{k+1}, the remaining $N-k-1$ contributions must happen at later times. Furthermore, since the prior contributions $(t_1,\dots,t_k)$ are being conditioned upon, they don't factor into the distribution. As a result, we find

$$\begin{aligned}
p(t_{k+1}|\tau,\lambda,t_1,\dots,t_k) &\propto \tilde{\Gamma}(t_{k+1})^{N-k-1}\lambda e^{-\lambda(t_{k+1}-\tau)}\theta(t_1-\tau) \\
&= e^{-\lambda(t_{k+1}-\tau)(N-k-1)}\lambda e^{-\lambda(t_{k+1}-\tau)}\theta(t_1-\tau) \\
&= \lambda e^{-\lambda(t_{k+1}-\tau)(N-k)}\theta(t_1-\tau)
\end{aligned}$$

Putting these together we then have

$$\begin{aligned}
p(t_{k+1}|t_1,\dots,t_k) &\propto \int_{-\infty}^{\infty} d\tau \int_0^{\infty} d\lambda\, \lambda^{k+\alpha+1}e^{-\lambda\{N(t_k-\tau)+k(\bar{t}-t_k)+\beta\ \ +(t_{k+1}-\tau)(N-k)\}}\theta(t_1-\tau) \\
&= \int_{-\infty}^{t_1} d\tau \int_0^{\infty} d\lambda\, \lambda^{k+\alpha+1}e^{-\lambda\{N(t_k+t_{k+1})+k(\bar{t}-t_k-t_{k+1})+\beta+\tau(k-2N)\}} \\
&= \int_0^{\infty} d\lambda\, \lambda^{k+\alpha+1}e^{-\lambda\{N(t_k+t_{k+1})+k(\bar{t}-t_k-t_{k+1})+\beta\}}\frac{1}{\lambda(k-2N)}e^{-\lambda t_1(k-2N)}
\end{aligned}$$

Removing normalization constants, we are left with evaluating the integral

$$\begin{aligned}
p(t_{k+1}|t_1,\dots,t_k) &\propto \int_0^\infty d\lambda\, \lambda^{k+\alpha} e^{-\lambda\{N(t_k+t_{k+1})+k(\bar{t}-t_k-t_{k+1})+\beta+t_1(k-2N)\}}\theta(t_{k+1}-t_k) \\
&= \frac{\Gamma(k+\alpha+1)\theta(t_{k+1}-t_k)}{(N(t_k+t_{k+1})+k(\bar{t}-t_k-t_{k+1})+\beta+t_1(k-2N))^{k+\alpha+1}} \\
&= \frac{\Gamma(k+\alpha+1)\theta(t_{k+1}-t_k)}{(N(t_k+t_{k+1}-2t_1)+k(\bar{t}+t_1-t_k-t_{k+1})+\beta)^{k+\alpha+1}} \\
&= \frac{\Gamma(k+\alpha+1)\theta(t_{k+1}-t_k)}{((N-k)(t_k+t_{k+1}-2t_1)+k(\bar{t}-t_1)+\beta)^{k+\alpha+1}}
\end{aligned}$$

Notice that if N is very large, the sensitivity to t_{k+1} in higher. In other words, one would find the probability to be substantially larger for smaller t_{k+1}. This makes a lot of sense since if there are a lot of contributions, then clearly the smallest of these will be rather small. This last way of writing the distribution makes it clear that the denominator is always positive, something that's clearly important.

We now turn to normalizing this distribution. This is important so that we can compare different clusters with each other. The distribution above has the form

$$p(t_{k+1}|t_1,\dots,t_k) = \frac{1}{\mathcal{N}}\frac{1}{(t_{k+1}+a)^b}\theta(t_{k+1}-t_k)$$

where $a = t_k - 2t_1 + (k(\bar{t}-t_1)+\beta)/(N-k)$ and $b = k+\alpha+1$. In order to find the normalization constant $\mathcal{N}$, we must integrate this expression and set it equal to 1. For simplicity we will switch variables to $x = t_{k+1} - t_k$. Defining $w = a + t_k$ we must then have

$$\frac{1}{\mathcal{N}}\int_0^\infty \frac{dx}{(x+w)^b} = 1$$

Note that since $w > 0$, this integral does not encounter any divergences. Furthermore, since also $b \geq 2$ (since $k > 1$ for any cluster that already exists), this integral converges. We then find

$$\frac{1}{\mathcal{N}} = (b-1)w^{b-1}$$

The probability distribution is then

$$\begin{aligned}
p(t_{k+1}|t_1,\dots,t_k) &= (b-1)w^{b-1}\frac{1}{(t_{k+1}+a)^b}\theta(t_{k+1}-t_k) \\
&= (b-1)\frac{(t_k+a)^{b-1}}{(t_{k+1}+a)^b}\theta(t_{k+1}-t_k)
\end{aligned}$$

We then finally arrive at

$$\boxed{p(t_{k+1}|t_1,\dots,t_k) = (k+\alpha)(N-k)\frac{\{2(N-k)(t_k-t_1)+k(\bar{t}-t_1)+\beta\}^{k+\alpha}}{\{(N-k)(t_{k+1}+t_k-2t_1)+k(\bar{t}-t_1)+\beta\}^{k+\alpha+1}}\theta(t_{k+1}-t_k)}$$

What remains is now a way of estimating N. Ideally we would treat N just like the other parameters, and update our beliefs of it over time and then sum over the various values just like we integrated over τ, λ. However, the sums you end up with are so nontrivial that one cannot find any closed form solution for them. As a result, we will have to estimate N in some other way and then use it in the above formula. At best, we might consider a few different N values (perhaps various orders of magnitude) and perform a simple three term summation over them.

4 Estimating N

At this point, I estimate N as

$$N = k + \sqrt{k}$$

This is loosely based on the error of the Poisson distribution which goes like the square root of the number of observations. This estimate may very well be updated in the future. This is just a reasonable guess placeholder for now.

5 Putting it all together

Given both a probability density over the next timestamp, t_{k+1} and the next positions, x_{k+1}, y_{k+1} we can now give the full probability as the product of these. Each time a new contribution is added we find the nearby clusters and compute this probability for each of the clusters. We then pick the cluster that gives us the largest probability as long as this largest probability surpasses some tunable threshold. If this threshold is not met we instead create a new cluster and set its parameters according to the above discussion.

There's a subtlety with this approach that we must resolve. Fundamentally it stems from the fact that the expressions we've arrived at are not truly probabilities but rather probability densities. To see where this issue would creep up, suppose that we have a simpler situation with two nearby clusters. Forget about the time component for a moment, and suppose that one cluster is centered at $x = 0$ with a very large size while another one is centered at $x = 1$ with a very narrow size. Furthermore, suppose that a new contribution appears at $x = 0$. Clearly, this contribution should be clustered in with the first cluster. However, let's see what the probability distributions look like:

$$\begin{aligned} p_1(0) &= \frac{1}{\sqrt{2\pi\sigma_1^2}} \\ p_2(0) &= \frac{1}{\sqrt{2\pi\sigma_2^2}} e^{-1/2\sigma_2^2} \end{aligned}$$

If we choose σ_1 sufficiently large for any given σ_2 we can make the first probability arbitrarily small. In other words, while we should most certainly categorize this datapoint into the first cluster, we may incorrectly classify it into the second cluster. Like I mentioned above, this fundamentally stems from the fact that these are not actual probabilities but rather probability densities. As far as I currently see, there are two ways around this that I'll discuss below

5.1 Cumulative Probability

One way to turn these into actual probabilities is to compute what the probability is that the next datapoint occurs at x_{k+1} *or more extreme* values and similarly for the time. Assuming the covariance is symmetric we would then want to compute (here Δ is the distance the $(k+1)$th contribution is away from the mean).

$$p(r \geq \Delta | x, y_1, \ldots, x_k, y_k) = \int_{r>\Delta}^{\infty} \int_0^{2\pi} \frac{1}{2\pi\sigma^2} e^{-r^2/2\sigma^2} r dr d\theta$$

Computing the integral gives

$$\boxed{p(r \geq \Delta | x, y_1, \ldots, x_k, y_k) = e^{-\Delta^2/2\sigma^2}}$$

Note that for $\Delta = 0$ this return the full possible probability of 1 while as $\Delta \gg \sigma$, the probability decreases.

We do the same for the temporal distribution. We want to compute the probability that t_{k+1} exceeds some cutoff that we will call Δ for now. Using the expression above, we need to integrate

$$\int_{\Delta}^{\infty} p(t_{k+1}|t_1, \ldots, t_k) dt_{k+1} = (k+\alpha)(N-k) \int_{\Delta}^{\infty} \frac{\{2(N-k)(t_k - t_1) + k(\bar{t} - t_1) + \beta\}^{k+\alpha}}{\{(N-k)(t_{k+1} + t_k - 2t_1) + k(\bar{t} - t_1) + \beta\}^{k+\alpha+1}} \theta(t_{k+1} - t_k) dt_{k+1}$$

This equates to

$$\boxed{p(t_{k+1} \geq \Delta | t_1, \ldots, t_k) = \left\{ \frac{2(N-k)(t_k - t_1) + k(\bar{t} - t_1) + \beta}{(N-k)(\Delta + t_k - 2t_1) + k(\bar{t} - t_1) + \beta} \right\}^{k+\alpha}}$$

5.2 Relative Probability Density

The other approach is to compare the density at a given point to the maximum value of the distribution. This allows you you make comparisons between the various distributions in a normalized way. The maximum for the spatial distribution occurs at the location of the mean. Approximating the t-distribution by a Gaussian we find

$$\boxed{f_{spatial}(r_{k+1}|r_1,\ldots,r_k) = e^{-\Delta^2/2\sigma^2}}$$

where σ is the variance we found above $((k+1)\beta_n/k\alpha_n)$. Interestingly this gives precisely the same expression as the cumulative expression. As for the temporal expression, we're left with

$$\boxed{f_{temporal}(t_{k+1}|t_1,\ldots,t_k) = \left\{\frac{2(N-k)(t_k-t_1)+k(\bar{t}-t_1)+\beta}{(N-k)(\Delta+t_k-2t_1)+k(\bar{t}-t_1)+\beta}\right\}^{k+\alpha+1}}$$

6 To Merge or Not To Merge

As this algorithm runs, it may sometimes start two separate clusters that should really be one and the same. Suppose e.g. that very first two contributions which are really part of a very large cluster (such as a concert), occur far apart. It is then reasonable that they create two separate clusters. The various nearby contributions will then be added to each of these, creating in the process two clusters that will begin to almost overlap. At this point, the algorithm needs to identify that it initially made a mistake and now go ahead and merge them into one single cluster.

Given two clusters with data $(t_1^{(1)}, x_1^{(1)}, y_1^{(1)}),\ldots,(t_k^{(1)}, x_k^{(1)}, y_k^{(1)})$ and $(t_1^{(2)}, x_1^{(2)}, y_1^{(2)}),\ldots,(t_p^{(2)}, x_p^{(2)}, y_p^{(2)})$, we want to determine if they should be merged. We will consider two sides of this story, the spatial and temporal data. Let's begin with the spatial values since these are independent and identically distribution.

6.1 Spatial Distribution

Let's for a moment focus exclusively on the x coordinates of the two events. At the end of the day we will just end up performing a simple z-test on this data, but it's instructive to see how we end up there as we will be re-doing much of the analysis for the temporal part later on (in which case we will not simply be performing a z-test). Assuming that the x-coordinates follows a simple Gaussian distribution, we have that the posterior distribution over the parameters, μ, λ is

$$p(\mu,\lambda|\vec{x}) = \frac{1}{\mathcal{N}}\frac{\beta_0^{\alpha_0}}{\Gamma(\alpha_0)}\left(\frac{\lambda}{2\pi}\right)^{k/2}\lambda^{\alpha_0-1}e^{-\lambda\left(\frac{1}{2}\sum_{i=1}^{k}(x_i-\mu)^2+\beta_0\right)}$$

Here we have used Bayes' theorem to write the posterior in terms of $p(\vec{x}|\mu,\lambda)$ and the prior $p(\mu,\lambda|\emptyset)$ which we choose to be a Gamma distribution over the precision, and uninformative over the mean just as before. Since we'll need it later, we quickly marginalize over the mean to obtain the distribution just over the precision,

$$\boxed{p(\lambda|\vec{x}) \sim \lambda^{\alpha_0+\frac{k-1}{2}-1}e^{-\lambda\left(\frac{1}{2}\sum_{i=1}^{k}(x_i-\langle x\rangle)^2+\beta_0\right)}}$$

We can now look at the two samples, $x^{(1)}$ and $x^{(2)}$ and obtain a distribution over the two means $\mu^{(1)}$ and $\mu^{(2)}$.

$$p(\mu^{(1)},\mu^{(2)},\lambda^{(1)},\lambda^{(2)}|\vec{x}^{(1)},\vec{x}^{(2)}) = p(\mu^{(1)},\lambda^{(1)}|\vec{x}^{(1)})p(\mu^{(2)},\lambda^{(2)}|\vec{x}^{(2)})$$

We then switch to the variables $s = \mu^{(1)} + \mu^{(2)}$ and $\Delta = \mu^{(1)} - \mu^{(2)}$ and follow that by an integration over the sum, s. We're left with the marginal distribution over the difference between the two means

$$p(\Delta,\lambda^{(1)},\lambda^{(2)}|\vec{x}^{(1)},\vec{x}^{(2)}) \sim \mathcal{N}(\langle x^{(1)}\rangle - \langle x^{(2)}\rangle, \frac{\lambda^{(1)}\lambda^{(2)}kp}{\lambda^{(1)}k+\lambda^{(2)}p})$$

Given that the mean of the normal distribution occurs at $\langle x^{(1)}\rangle - \langle x^{(2)}\rangle$, the value zero (i.e. no difference between the two samples) lies z_x standard deviations away from the mean, where z_x is given by

$$\boxed{z_x = \frac{|\langle x^{(1)}\rangle - \langle x^{(2)}\rangle|}{\sqrt{(\sigma^{(1)})^2/k + (\sigma^{(2)})^2/p}}}$$

Here, we'll use our best guess for the values of $\sigma^{(1)}$ and $\sigma^{(2)}$. These are obtained by locating the mean of the marginal distribution over λ from above. This occurs at

$$\bar{\lambda} = \frac{\alpha_0 + \frac{k-1}{2}}{\frac{1}{2}\sum_{i=1}^{N}(x_i - \langle x\rangle)^2 + \beta_0}$$

In terms of the variance, $\sigma^2 = 1/\lambda$ we thus have

$$\boxed{\sigma^2 = \frac{1}{k + 2\alpha_0 - 1}\left\{\sum_{i=1}^{k}(x_i - \langle x\rangle)^2 + 2\beta_0\right\}}$$

We now clearly see how the prior should be interpreted: There are $2\alpha_0$ auxiliary observations with a total variance of β_0 prior to the first measurement.

So, the story goes as follows. Given two clusters, look at their x-values and use the above two formulas to compute the z score between them. Then, as long as this z-score is above some threshold (perhaps $1-2$) we realize that the means are significantly different and thus avoid merging them. Otherwise we decide to merge them. That being said, we of course also have a similar metric for the y-coordinates. In order to make a decision we should thus consider both of them, or more precisely the distance between them. To be even more precise, given two independent z-scores, z_x and z_y their combined distribution is also a Gaussian

$$p(z_x, z_y) = \mathcal{N}\left(\vec{0}, 1\right)$$

In terms of the "Euclidean distance", $r^2 = z_x^2 + z_y^2$, we have

$$p(r) = re^{-r^2/2}$$

Suppose we want to know how improbable a value $r \geq \gamma$ is, then we're left with evaluating

$$p(r \geq \gamma) = \int_{\gamma}^{\infty} p(r)dr = e^{-\gamma^2/2}$$

Note that as $\gamma \to 0$, this probability approaches 1. Similarly, for infinite r it approaches zero. We must then in our model place some cutoff probability, p_0 and require

$$\boxed{z_x^2 + z_y^2 \geq 2|\ln p_0|}$$

in order to not merge the two clusters. If we put a very strict cutoff and say e.g. that $p_0 = 0.01$, i.e. we want to merge all clusters unless they are statistically very unlikely to be the same (1%). Then, we would have to have $z_x^2 + z_y^2 \geq 9.21$. Even if the x means were identical, the y means would have to differ by at least 3 standard deviations for us to keep them separate. Since p_0 is a tunable parameter, we can adjust it as we see fit.

6.2 Temporal Distribution

We now re-do the analysis above for the temporal distribution. In particular, the posterior distribution over the parameters τ, λ is given by

$$p(\tau, \lambda|\vec{t}) \sim \lambda^{k+\alpha} e^{-\lambda\{N(t_k-\tau)+k(\bar{t}-t_k)+\beta\}}$$

Just like above we will later need a marginal distribution over the width. We thus begin by integrating over τ

$$p(\lambda|\vec{t}) \sim \lambda^{k+\alpha-1}e^{-\lambda\{N(t_k-t_1)+k(\bar{t}-t_k)+\beta\}}$$

The mean value for the width is thus

$$\boxed{\bar{\lambda} = \frac{k+\alpha}{N(t_k-t_1)+k(\bar{t}-t_k)+\beta}}$$

Continue by considering two different sets of time measurements, $\vec{t} = t_1, \ldots t_k$ and $\vec{\phi} = \phi_1, \ldots, \phi_p$. We will consider the joint distribution over $\tau_1, \tau_2, \lambda_1, \lambda_2$ and just like before integrate out the sum $s = \tau_1 + \tau_2$. With λ_1, λ_2 fixed for now (we will later just plug in their average values from above), we have

$$p(s, \Delta|\vec{t}, \vec{\phi}) \sim e^{\frac{1}{2}(\lambda_1 N - \lambda_2 M)\Delta} e^{\frac{1}{2}(\lambda_1 N + \lambda_2 M)s}$$

In order to integrate out s, we must be careful with the limits of integration. Since $s + \Delta = \tau_1 \leq t_1$ and $s - \Delta = \tau_2 \leq \phi_1$, we must always have $s \leq \text{Min}(2t_1 - \Delta, 2\phi_1 + \Delta)$. Integrating out s then gives us

$$p(\Delta|\vec{t}, \vec{\phi}) = e^{\frac{1}{2}(\lambda_1 N - \lambda_2 M)\Delta + \frac{1}{2}(\lambda_1 N + \lambda_2 M)\text{Min}(2t_1 - \Delta, 2\phi_1 + \Delta)}$$

There are obviously two branches to this distribution depending on which side of $t_1 - \phi_1$, Δ falls on. Since we now have a distribution over Δ, we want to figure out, given all the data, how improbably $\Delta = 0$ is. More precisely let's compute the probability that Δ is at least as extreme as 0. In order to do this, we must first normalize the distribution above. Doing this gives us

$$p(\Delta|\vec{t}, \vec{\phi}) = \frac{\lambda_1 \lambda_2 NM}{\lambda_1 N + \lambda_2 M} e^{-\frac{1}{2}(\lambda_1 N - \lambda_2 M)(t_1 - \phi_1)} \begin{cases} e^{\lambda_1 N \Delta} e^{-\frac{1}{2}(\lambda_1 N + \lambda_2 M)(t_1 - \phi_1)} & \Delta \leq t_1 - \phi_1 \\ e^{-\lambda_2 M \Delta} e^{\frac{1}{2}(\lambda_1 N + \lambda_2 M)(t_1 - \phi_1)} & \Delta > t_1 - \phi_1 \end{cases}$$

Let's assume, without loss of generality, that $t_1 > \phi_1$. If this is not the case, just interchange the meaning of cluster 1 and cluster 2. In that case, $\Delta = 0$ falls in the region $\Delta < t_1 - \phi_1$. The probability that one obtains at least as extreme a difference as 0 is then

$$p(\Delta \leq 0) = \frac{\lambda_1 \lambda_2 NM}{\lambda_1 N + \lambda_2 M} e^{-\lambda_1 N(t_1 - \phi_1)} \int_{-\infty}^{0} e^{\lambda_1 N \Delta} d\Delta = \frac{\lambda_2 M}{\lambda_1 N + \lambda_2 M} e^{-\lambda_1 N(t_1 - \phi_1)}$$

Notice that as $\lambda_2 \to \infty$, i.e. all the probability is focused on the left side of $t_1 - \phi_1$, and simultaneously $t_1 - \phi_1 = 0$, we obtain a total probability of 1 which is what we'd expect. So, all seems good so far. Now, if this probability is very small it's indicative that the difference between he two τ values is probably rather large. As a result, we should probably not merge them. If we again place some small cutoff probability $\tilde{p}_0$ that this must be smaller than, we find that the difference, $t_1 - \phi_1$ must exceed

$$\boxed{t_1 - \phi_1 \geq \frac{1}{\lambda_1 N} \left| \ln \left(\frac{\lambda_1 N + \lambda_2 M}{\lambda_2 M} \tilde{p}_0 \right) \right|}$$

Recall once again the we have enforced the ordering $t_1 > \phi_1$ here. The estimates for λ_1 and λ_2 are once again given by their posterior mean.

$$\boxed{\begin{aligned} \lambda_1 &= \frac{k+\alpha}{N(t_k-t_1)+k(\bar{t}-t_k)+\beta} \\ \lambda_2 &= \frac{p+\alpha}{M(\phi_p-\phi_1)+p(\bar{\phi}-\phi_p)+\beta} \end{aligned}}$$

The idea is now to first consider the spatial part of the story. If we decide that the clusters are not sufficiently far apart, we then consider the temporal part. If the temporal part also passes the test (i.e. they are not sufficiently different) we merge them. This way we avoid having to look at the thorny (see below) temporal part unless we're already considering merging the two clusters.

6.3 Subtleties

There is one slight subtlety in this (temporal) analysis. Suppose that for some reason after a cluster has already been started, a small off-shoot of it starts its own mini cluster very nearby (obviously, since it's really the same cluster). The time stamps will then be e.g. $1, 2, 3, 4, 5, 6, 7, 8, 9, 10$ for the first cluster and $4, 5, 6$ for the second cluster. The estimate for τ_1 will be 1 and τ_2 will be 4. These are of course very far apart and thus, temporally the clusters shouldn't be merged. So, while the timestamps seem very similar to each other, they will not be merged. Perhaps this could be really cool: suppose you're at a big event and step outside during it and take a few photos. Perhaps having two separate clusters is kind of cool. Also, suppose that two nearby events really did occur and that these two clusters really do correspond to two events (maybe two nearby happy hours, one at 6pm and one at 6:30pm). Not merging them could really be a benefit then.

Fundamentally it's always better to merge too little than too much since too many merges will conflate independent events. As a result, I'll stick with the above analysis for now and think deeper about this as we move forward.

7 Parameter Values

We now have a decent understanding of the Bayesian problem at hand, however we still have a few parameters that we need particular values for. These parameters are $\alpha, \beta, \alpha_0, \beta_0$. We also need a way of estimating N, the total number of expected contributions. Let's begin by looking at the temporal distribution. The decay rate should then be in some range with mean of λ_* and width of $\Delta\lambda$. Since the decay rate follows the distribution $\sim \lambda^\alpha e^{-\beta\lambda}$, the mean and width are given by

$$\lambda_* = \frac{\alpha+1}{\beta}$$
$$\Delta\lambda^2 = \frac{\alpha+1}{\beta^2}$$

Solving these for α, β we find

$$\alpha = \lambda_*^2/\Delta\lambda^2 - 1$$
$$\beta = \lambda_*/\Delta\lambda^2$$

Let's suppose that the average event has a length of 600 seconds. We then need to set $\lambda_* = 1/600$. Furthermore, we want to support a large range of events so we need to have a rather large spread, $\Delta\lambda$. By setting $\Delta\lambda = 1/1000$, we can support events all the way from 6.25 to 25 minutes (as our prior!) with the mean event being 10 minutes. This gives

$$\boxed{\begin{aligned}\alpha &= 1.778\\ \beta &= 1667\end{aligned}}$$

As for the spatial distribution, α_0, β_0 refer to the parameters in the Gamma distribution which confusingly are just like the above parameters for the temporal distribution with the exception that there's a one-off difference in the α parameter. As a result, we have (here λ is the precision of the Gaussian - $\lambda = 1/\sigma^2$. Unfortunately I was stupid in how I chose the parameter names.... sorry)

$$\lambda_* = \frac{\alpha_0}{\beta_0}$$
$$\Delta\lambda^2 = \frac{\alpha_0}{\beta_0^2}$$

Inverting these, we find

$$\alpha_0 = \lambda_*^2/\Delta\lambda^2$$
$$\beta_0 = \lambda_*/\Delta\lambda^2$$

Suppose that we expect the average event to be rather small as e.g. 0.01 miles but we want to easily support events up to 0.03 miles. Then we might want to choose $\lambda_* = 5500$ and $\Delta\lambda = 4500$. This gives a range in the precision between $[1000, 10000]$ which translates into a width for the Gaussian of $[0.01, 0.03]$ with the mean falling closer to 0.01. The parameters should then be set to

$$\boxed{\begin{aligned}\alpha_0 &= 0.111\\ \beta_0 &= 0.0000444\end{aligned}}$$

8 Simulation Results

Using the above parameters and algorithm, there are still a few loose ends. In particular, what should the threshold probability be for a contribution to join a cluster? Furthermore, in addition to requiring a minimum threshold to be met, one may want the best fit cluster to be at least some multiple bigger than the second best fit. What should this multiple be? Finally, how many z scores away should two clusters be in order for them not to be merged?

9 Future Directions

There are many other really interesting ideas for how to cluster events. They fall into two subcategories: modifying the model, and adding additional signals.

9.1 Modifying the Model

While the above model seems to work rather well in simulations, there are still a few shortcomings that can be addressed in future iterations.

i. It would be interesting to re-introduce the correlation between the x and y coordinates, σ_{xy} as this will allow you to classify events that are asymmetric such as parades.

ii. The underlying assumption for the spatial part of the story is that the contributions follow a Gaussian centered at the location of the event. However, this may not be the case. Consider e.g. a baseball game where the event takes place on the field, but all the potential contributions occur in an annular region around the center. In this case, the distribution oohs completely different. The way to attack this would be to allow for a finite set of distributions, one of which is a Gaussian while others more appropriate to the baseball game are also included. Then we place a prior on the various distributions presumably strongly in favor of the Gaussian since most events would likely look like that. Then, just like we integrate out all the parameters to obtain a true posterior distribution, we also sum over the various models.

iii. The same issue as was mentioned above for the non Gaussian spatial distribution could also occur for the temporal case. One could imagine a scenario where an event takes place continuously between two strict time limits such as a concert or a sports event. In these cases you'd not really see a decay of events but rather a uniform distribution between two extreme values. The same approach as above would work – simply sum over the various models with some appropriate priors.

iv. So far we can only classify stationary events, but what about events taking place on a bus/cruise/car? I'm not sure of the best way to classify these events, but it would be an interesting problem to be sure. Perhaps the main idea would be to add an additional parameter – speed, place a prior on it and integrate it out just like everything else.

v. Using different parameters for the priors in different places, based on historical contributions. If we know that the location of a certain contribution lies at a baseball stadium, we should probably shift the prior in favor of an annular region. Similarly, events in New York city will most likely have a larger number of contributions than events in the middle of nowhere.

9.2 Additional Signals

So far the model only takes into account the location and time of each contribution, but there are also additional signals that we can look at to determine clustering.

i. User Id. Presumably two photos taken close in time and space to each other by the same user should be clustered together. So, even if the location and time are not sufficiently close together to warrant clustering, this signal might push it over the edge. This might be particularly useful in the case of moving events.

ii. I already mentioned above that we might want to update priors based on location. In other words, we might want to classify locations into a few categories, *suburban*, *urban*, *stadium*, *landmark*, . . . and use this as a signal as well.

iii. Using iBeacon we can broadcast to all those around us each time we upload a contribution. When those other people later upload a contribution they will also send a long a list of people they've been nearby recently. We can then use this information to cluster the new contribution together with the prior one if they are sufficiently near each other in space and time anyway.

10 References

Reaction times: http://cogprints.org/6326/1/moscoso-bbs.pdf
Bayesian online learning: http://www.ki.tu-berlin.de/fileadmin/fg135/publikationen/opper/Op98b.pdf
Bayesian inference of the Gaussian: http://www.cs.ubc.ca/ murphyk/Papers/bayesGauss.pdf
ML book: http://www.hua.edu.vn/khoa/fita/wp-content/uploads/2013/08/Pattern-Recognition-and-Machine-Learning-Christophe-M-Bishop.pdf
failure analysis: http://informatik.hu-berlin.de/Members/salfner/publications/salfner10survey.pdft