Multistage probabilistic classification

prepared by:	Coryn Bailer-Jones, Kester Smith
approved by:	
reference:	GAIA-C8-TN-MPIA-CBJ-037
issue:	3
revision:	0
date:	2011-07-17
status:	Issued

Abstract

We present a probabilistic framework for combining classifiers. Each classifier stage may be based on different data (e.g. astrometry, BP/RP spectrum, G-magnitude, Galactic latitude) or types of object (Galactic, extragalactic). Different data types have differing degrees of power in determining the classes. We present two types of combination. The first uses simple probability theory to combine classifications based on spectroscopy only and astrometry only. The second combines two spectral-only classifiers – one for all classes, the other only for Galactic objects (single and binary stars) – using a weighting function which depends on the astrometry. We introduce a simple approach to simulating Galactic astrometry based on the gamma distribution. We use this to train and test an astrometric-only classifier, one stage in a multistage process.

Document History

Issue	Revision	Date	Author	Comment
3	0	2011-07-17	CBJ	Corrected equations 7 and 8, removed subsequent discussion (superceded by CBJ-053)
2	0	2010-01-15	CBJ	Added note in section 2.2 to reference CBJ-053. Reformatted
1	0	2008-04-18	CBJ	First issue.
D	0	2008-04-14	CBJ	Checked. Minor correction. Conclusions and abstract added.
D	0	2008-03-03	CBJ	Added section 3.2 (Mclust on astrometric simulated data)
D	0	2008-02-14	CBJ	Working draft

Contents

1 Introduction

“Single shot” approaches to classification attempt to estimate the class of an object from a set of data in a single step. For example, we may use nonlinear regression to fit $P(C_k)$ as a function of the ensemble of BP/RP, RVS, astrometric and photometric variability data. Although simple (i.e. we just plug all of the data into the input space of a model), such an approach may not be optimal. We may instead want to build separate classifiers for different types of data, e.g. one for spectra and another for astrometry (the 2D vector of parallax and magnitude of the proper motion). There are several reasons why we may prefer such an approach.

- We can fit separate models to distinct types of data. Fitting a single model to 120 spectral variables as well as 2 astrometric bins variables may, in practice, result in the astrometric data being essentially ignored, because their impact is swamped by the presence of the numerically superior spectral data. This is especially the case when the data are noisy, when also noisy irrelevant variables (e.g. in the spectra) have an additional confusing impact.
- We can fit different classification models to distinct types of data. This may permit us to incorporate domain knowledge more effectively.
- Early in the mission we have only spectral data for classification; later we also have astrometric data. With a single shot approach we must build two entirely separate but “overlapping” models, one for spectra only, the second for spectra plus astrometry. With a multistage approach we can use single model based on spectra only and, when astrometry become available, modify its class predictions based on the astrometric classifier. This “updating” of class probabilities seems more natural.
- We achieve better understanding of the classification, because we can explicitly see the impact of the different types of data on the classification. This makes it easier to, say, disregard inferences based on some data if we have reason to be suspicious of some subset of the data (e.g. the proper motions) in certain situations (e.g. we suspect astrometric binarity has led to a wrong proper motion estimate).

In this techical note we describe two general approaches for building multistage classifiers. We apply these to the specific problem of a two stage classifier for spectra and astrometry, using simulated data.

2 Methods

2.1 Notation

e, g	subscripts to denote extragalactic and Galactic objects respectively
C_k	class k
D_s	measured spectrum
D_a	measured astrometry
P	probability
ϖ	parallax
μ	proper motion
σ_ϖ	standard deviation of the noise distribution in ϖ
σ_μ	standard deviation of the noise distribution in μ
$\Gamma(k, \theta)$	The Gamma probability distribution with shape parameter k and scale parameter θ
$N(m, \sigma)$	The Gaussian probability distribution with mean m and standard deviation σ

2.2 Combination via Bayes

17 July 2011: A more general and accurate discussion of the probabilistic combination of classifiers can be found in GAIA-C8-TN-MPIA-CBJ-053-02

This two-stage classifier sets up one classifier for the spectral data and another for the astrometric data. Here we show how to combine their predictions.

Classification algorithms are trained on a set of data in order to predict the class of an object. $P(D_s|C_k)$ is the likelihood function returned by a typical classifier for that object. Formally it is the conditional probability of observing spectrum D_s given that the source is class C_k, but we tend to think of it as the likelihood that the data D_s are drawn from C_k, and thus as an explicit function of C_k. This is related to the quantity we are really interested in, namely the (posterior) probability that the object is of class C_k given the observation of data D_s, via *Bayes' theorem*

$$P(C_k|D_s) = \frac{P(D_s|C_k)P(C_k)}{P(D_s)} \tag{1}$$

We may consider the denominator as a normalization factor because

$$P(D_s) = \sum_k P(D_s|C_k)P(C_k) \tag{2}$$

This is the *marginalisation equation*, which together with Bayes' theorem form the two basic equations of probability (Sivia 1996). Often a classifier provides a likelihood and we automatically equate this with $P(C_k|D_s)$, in which case we are implicitly assuming equal prior probabilities, $P(C_k)$, for the classes.[1] In general one can (and should) control the priors and thus explicitly derive the posterior probability. We do this explicitly in another technical note to accommodate the relative rareness of quasars compared to stars (CBJ-036).

Equation 1 represents a classifier based on spectral data, but we can make a similar statement for a classifier based on astrometric data

$$P(C_k|D_a) = \frac{P(D_a|C_k)P(C_k)}{P(D_a)} \tag{3}$$

How do we combine these data, or rather the outputs from the two classifiers? In other words, what is $P(C_k|D_s, D_a)$? We write Bayes' theorem for the combined data

$$P(C_k|D_s, D_a) = \frac{P(D_s, D_a|C_k)P(C_k)}{P(D_s, D_a)} \tag{4}$$

The likelihood term on the RHS is

$$P(D_s, D_a|C_k) = P(D_s|D_a, C_k)P(D_a|C_k) = P(D_a|D_s, C_k)P(D_s|C_k) \tag{5}$$

If we assume that a measurement of D_s and that of D_a are independent conditional on the class C_k

$$P(D_s, D_a|C_k) = P(D_s|C_k)P(D_a|C_k) \tag{6}$$

but in general

$$P(D_s, D_a) \neq P(D_s)P(D_a) \tag{7}$$

(see CBJ-053 for an explanation). Substituting these last two equations into equation 4 gives

$$P(C_k|D_s, D_a) = a\, P(D_s|C_k)P(D_a|C_k)P(C_k) \tag{8}$$

where a is a class-independent normalization factor. This is intuitive (note the essential symmetry between D_s and D_a), but the derivation serves to highlight the assumed independence of D_s and D_a.

For further discussion of the independence assumption (or what we even mean by "indepenedence"), see CBJ-053. The rest of the text in this section is from version 2 of this document and is just retained for historical interest.

Is this independence assumption valid? Given a measured spectrum, can we make better-than-random predictions about the astrometry? In general we can, provided we have sufficient SNR. Interestingly, the converse (can we predict the spectrum from the astrometry?) is not always true. Given a measurement of a significantly non-zero parallax and proper motion, we can say *something* about the expected spectrum (e.g. it's unlikely to be a quasar or galaxy). However, at very faint magnitudes, Gaia will measure many parallaxes and proper motions consistent with zero (within the noise). These could be either distant stars or extragalactic objects, so in this case D_a does

[1] You can't help but be a Bayesian, because there are always priors, whether you like it or not. It's better to be a Bayesian and be explicit and honest about your priors. See CBJ-036 for more discussion.

not tell us anything about D_s: the data are independent.[2] In general, the assumption of independence is not strictly true, at least not when conditional upon the object magnitude (which we will usually know). In that case, we would have to set up a model for for $P(D_a, D_s|C_k)$. This is difficult. We could instead use equation 5 and define a model for $P(D_a|D_s, C_k)$ or $P(D_s|D_a, C_k)$.

For now, however, we will assume independence. This permits us to build separate classifiers based on astrometry and spectra and to combine their predictions in the simple way given in equation 8.

We could generalize equation 8 to include additional terms for data other than D_s and D_a. Following the above discussion, we may want to include a classifier based only on the object's magnitude or its Galactic latitude. For bright objects the magnitude is a powerful Galactic/extragalactic discriminator, yet we would need to be careful because such a classifier must be built assuming a universe model.[3]

2.3 Combination via explicit weighting

In this second approach, the two classifiers are both built only using spectral data. The first, A, is built to classify all types of objects (Galactic and extragalactic). The second, B, learns only to classify different types of Galactic objects. The reason for this distinction is that astrometry is only useful inside A: in model B the parallax and proper motion are a priori consistent with zero.

Let $P_A(C_k|D_s)$ be the probability from classifier A and $P_B(C_k|D_s)$ the probability from classifier B. These are posterior probabilities in the sense defined in section 2.2, so potentially non-equal class priors have been taken into account. We now define a weighting function, $f(D_a)$, with the property $0 \leq f(D_a) \leq 1$. This function is large if the data indicate Galactic status, i.e. give more evidence for model B. This would be the case if the astrometry is significantly greater than zero. We use this to combine the outputs from the two models thus

$$\bar{P}(C_k|D_s, D_a) = [1 - f(D_a)]P_A(C_k|D_s) + f(D_a)P_B(C_k|D_s) \tag{9}$$

Function $f(D_a)$ should become significantly above zero only when the measured parallax and proper motion are greater than their expected errors. One possible function is

$$f(D_a) = f(\varpi, \mu) = tanh\left(\lambda_\varpi \frac{\varpi}{\sigma_\varpi}\right) tanh\left(\lambda_\mu \frac{\mu}{\sigma_\mu}\right) \tag{10}$$

where the λ terms control the sharpness of the transition from 0 to 1 for each astrometric parameter separately. This function is shown in Fig. 1.

[2] You can still say something about the relative probability of the object being Galactic or extragalactic, but this is (a) based only on your prior knowledge of the universe and the object magnitude, not on the specific measurement of the astrometry, D_s, and (b) a statement about the classes, not the data.

[3] This is just another type of prior information affecting our classifications. We could build the model to make stronger or weaker statements about class probability as we so desire, by controlling the shape of the one-dimensional probability density functions, $P(C_k|D_m)$, where $D_m = G$, the Gaia G-band magnitude.

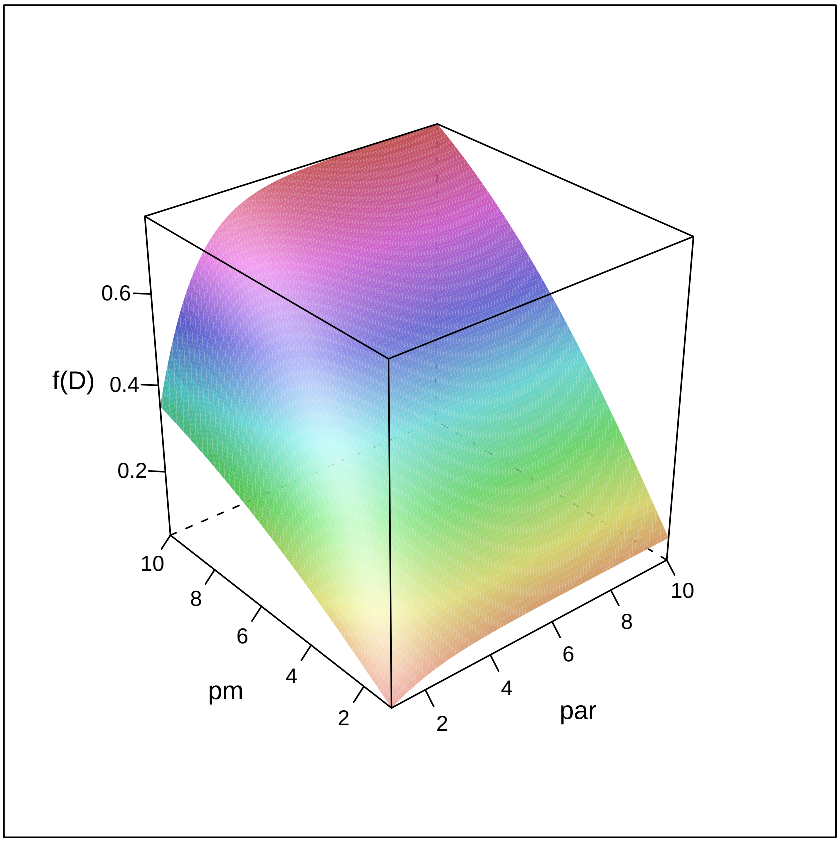

FIGURE 1: Illustration of a 2D weighting function, $f(D_a)$, used in the model in section 2.3, equation 10. It is shown for $\lambda_\varpi = 0.5$ and $\lambda_\mu = 0.1$.

3 Experiments

The two-stage approach in section 2.2 allows one to use a simple two parameter classifier for the astrometry. This two-dimensional parallax and proper motion plane can be fit easily using standard statistical methods, without needing to resort to specific high dimensional methods (as we do for the spectral classifier). This is demonstrated in section 3.2 using simulated astrometric data, which are first described in section 3.1.

3.1 Astrometry simulations

Astrometric data are simulated for Galactic and extragalactic objects. The philosophy we adopt is that while the astrometry for these two classes must of course reflect the differences between these populations, it should not be tied too closely to a specific Galaxy or Universe model. The purpose of Gaia is to learn about the relationship between kinematics, structure and intrinsic stellar properties, so the classifier should not have prohibitive or extreme biases concerning these properties. Here, therefore, we use very simple distributions for the proper motions and parallaxes of the two classes of objects, and simulate data by drawing from these at random.

Extragalactic objects. The intrinsic proper motion and parallax are both zero, so the observed values are due to noise:

$$\varpi_e = N(0, \sigma_\varpi) \tag{11}$$

$$\mu_e = \sqrt{N(0, \sigma_\mu)^2 + N(0, \sigma_\mu)^2} \tag{12}$$

where $\sigma_\varpi = 0.250$ mas and $\sigma_\mu = 0.250$ mas/yr are the approximate errors at G=20. Note that we are using the magnitude of the proper motion, which is the quadrature sum of two independent Gaussian random variables.

Galactic objects. The intrinsic parallaxes, ϖ'_g, must be positive, so a Gaussian distribution cannot be used. We use a Gamma distribution, $\Gamma(k, \theta)$, with shape $k = 1.5$ and scale $\theta = 1.0$. The density function is show in Fig 2. To this is added the parallax noise, a Gaussian random variable to achieve the measured parallax, ϖ_g. Proper motions we model by assuming that stars have a space velocity distribution (relative to the Gaia reference frame, which is an inertial frame formed by the quasars) which is independent of distance from the Sun (CBJ-029). We model this with a Gaussian random variable for each proper motion component which has zero mean and a standard deviation which increases linearly with the (intrinsic) parallax.

$$\varpi'_g = \Gamma(k, \theta) \tag{13}$$

$$\varpi_g = \varpi'_g + N(0, \sigma_\varpi) \tag{14}$$

$$\mu_g = \sqrt{N(0, f * \varpi'_g)^2 + N(0, f * \varpi'_g)^2} \tag{15}$$

where we use $f =$ 7/yr, which is a fit from Hipparcos data.

The simulations for a sample of 1000 extragalactic and 1000 extragalactic objects are shown in Fig. 3. A density plot of the astrometry for the Galactic sample is shown in Fig. 4.

3.2 Astrometric classification using mixture models

We used the R package Mclust (Fraley & Raftery 2006) to perform discriminant analysis using a Gaussian mixture model. The density over the astrometry for each class, C_k, is modelled using a mixture (sum) of Gaussians. This density, or likelihood, we write as $P(\varpi, \mu | C_k, \beta_k, H)$, where β_k is the set of the parameters of the Gaussian(s) and H denotes the type of model. In this case, H specifies the parametrization of the two-dimensional (parallax, proper motion) covariance function for each Gaussian, plus the number of Gaussians used for each class. (That is, some of the models tested have a restricted covariance, e.g. no off-diagonal elements.) For a given H the

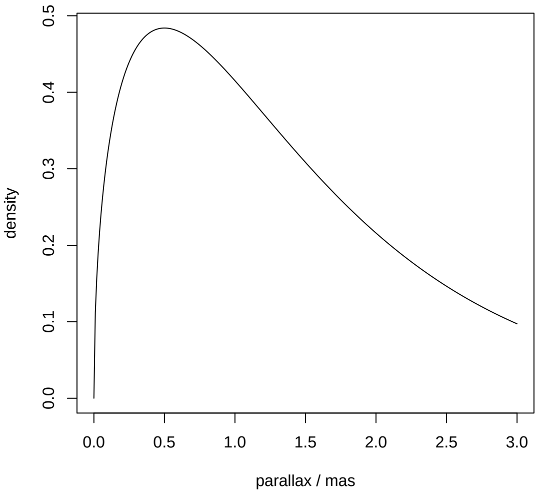

FIGURE 2: The density function of the intrinsic (noise free) parallaxes for Galactic objects. It is a Gamma distribution, $\Gamma(k, \theta)$, with shape $k = 1.5$ and scale $\theta = 1.0$.

β_k values are learned from a training set using the EM algorithm to achieve optimal class separation. The performance is tested on a separate set drawn from the same distributions described in section 3.1. Train and test sets each comprised 1000 stars and 1000 extragalactic objects. Mclust selects the optimum model, H, by calculating the Bayesian Information Criterion (BIC) for each model (after determining the $\{\beta_k\}$ for that model). The BIC is a metric which trades off the likelihood of the model with its complexity. It adopts the Bayesian view that more complex models are a priori less supported because they make less specific predictions.

We used Mclust to fit models with between 1 and 9 Gaussians per class, each with a range of covariance parametrizations (see Fraley & Raftery 2006 for more details). The best fit (maximum BIC) has three Gaussians for the extragalactic class and five for the galactic objects, in both cases with the VVI covariance function. This is an Mclust coding which means that the axes of the Gaussian ellipses are oriented parallel to the variable axes (no off-diagonal elements in the covariance function) but that their sizes were unconstrained (different on-diagonal elements). Gaussians with arbitrary orientation offer more flexiblity but require more parameters, i.e. more degrees of freedom, and such a model is more heavily penalized by the BIC. We might expect more Gaussians needed for fitting the stars because of their non-Gaussian generative distribution. Of the 2000 objects in the test set there were 36 misclassifications ($36/2000 = 1.8\,\%$) of which 2 were misclassified stars and the rest misclassified extragalactic objects. These are plotted in Fig. 5. It was already obvious from Fig. 3 (the training data) that a perfect classification is not possible.

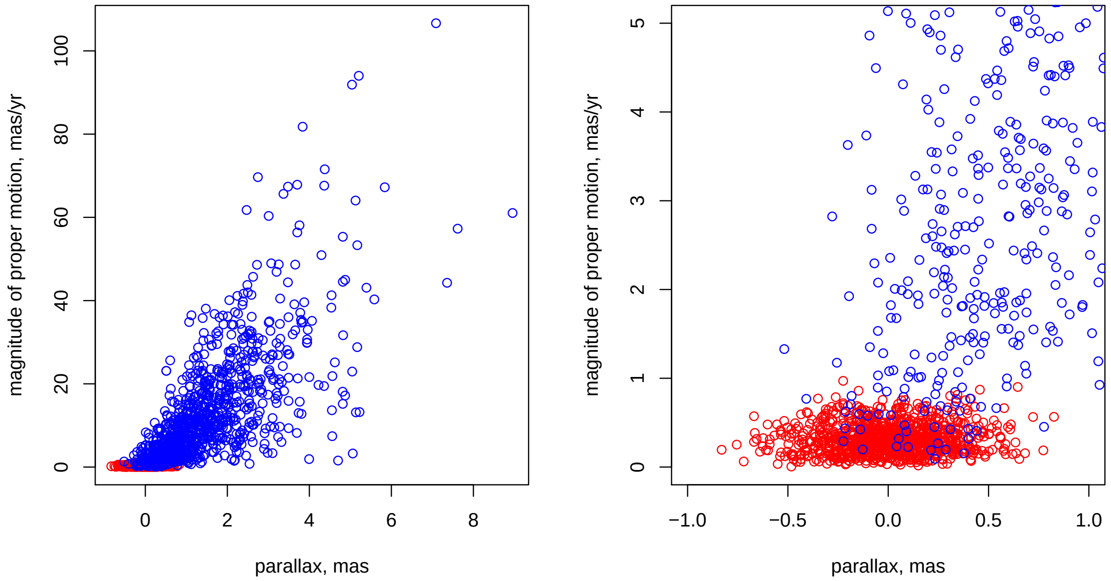

FIGURE 3: Simulated astrometry for 1000 Galactic objects (blue) and 1000 extragalactic objects (red), drawn from the distributions described in section 3.1. The right panel is a zoom.

4 Conclusions

Adopting a probabilistic framework, we have outlined a general way of combining classifiers. Each classifier could focus on one type of data (e.g. spectroscopy, astrometry) or a subset of classes (e.g. extragalactic, galactic). Each such classifier is a "stage" in a classification chain, the probabilities from each of which are multiplied (and normalized) to form the final class probability. Additional stages could be based simply on the magnitude or Galactic latitute of the source. Using a reasonable simulation for stellar and extragalactic source astrometry, we demonstrate that a purely astrometric classifier is a useful "stage". Of course, these data are only a useful discriminant in this case when the parallax or proper motion are significantly larger than zero. Otherwise the probability simply reduced to the prior class probabilities for measurements which are consistent with zero.

5 References

Fraley & Raftery, 2006, MCLUST Version 3 for R: Normal mixture modelling and model-based clustering `http://www.stat.washington.edu/mclust/`

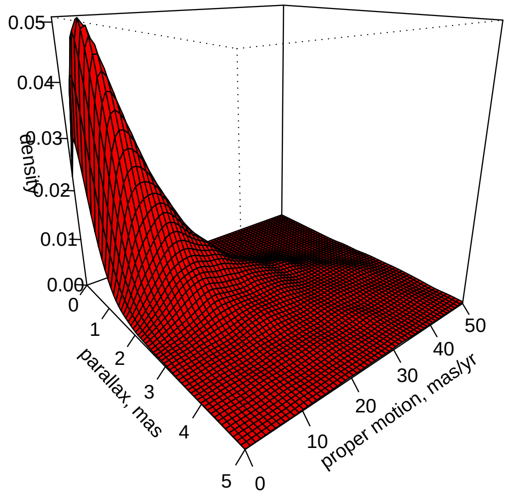

FIGURE 4: Smoothed density plot of the simulated astrometry for the 1000 Galactic objects

GAIA-C8-SP-MPIA-CBJ-029, Bailer-Jones 2007, Cycle 3 simulation specifications, on Livelink (CU8 folder)

GAIA-C8-TN-MPIA-CBJ-036, Bailer-Jones & Smith 2008, Quasar classification with DSC. Contamination, completeness and modified priors, on Livelink (CU8 folder)
15 Jan. 2010: *See instead Bailer-Jones et al. 2008, MNRAS 391, 1838*

GAIA-C8-SP-MPIA-CBJ-053, Bailer-Jones & Smith 2011, Combining probabilities

GAIA-C8-DA-OAPD-RS-002, Sordo & Vallenari 2008, Cycle 3 simulations description, on Livelink (CU8 folder)

Sivia, D., Data Analysis: A Bayesian Tutorial, OUP, 1996

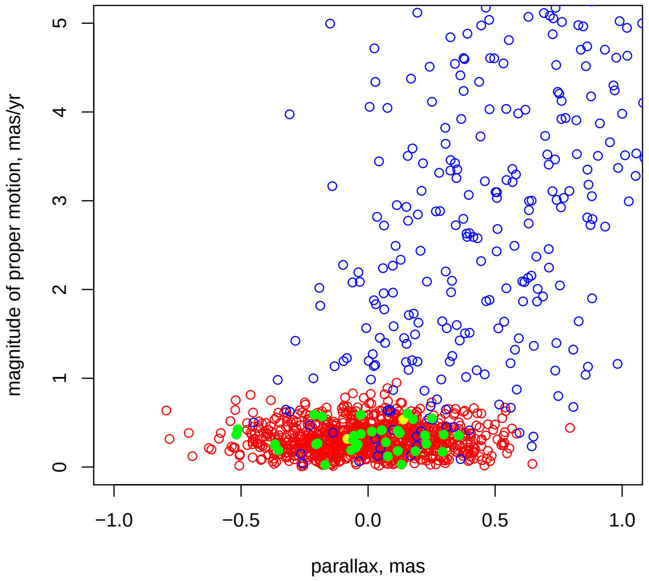

FIGURE 5: Classification results from applying a mixture model to the astrometric data. Blue and red points are correct classifications of galactic and extragalactic objects respectively, whereas green and yellow points are the corresponding misclassifications (e.g. green are galactic objects misclassified as extragalactic ones).

Appendix: R code for the astrometric simulations

```
# Simulate galactic par from sum of gamma rv and Gaussian rv. Latter
# represents noise. Gamma distribution ensures parallax is >=0
# Simulate galactic pm from sqrt of sum of square of two Gaussian
# random variables each with zero mean and sd = velfac*par where
# velfac=7 from CBJ-029.  Extragalactic par is Gaussian rv and pm is
# sqrt of sum of squares of two Gaussian rvs.  Units are mas
n.ext <- 1000
n.gal <- 1000
par.noisesd <- 0.250
pm.noisesd  <- 0.250
velfac <- 7.0
gamma.shape <- 1.5
gamma.scale <- 1
# plot gamma
# mean is shape*scale, variance is shape*(scale^2), mode is (shape-1)*scale (for shape>1)
x <- seq(0,3,0.01)
plot(x,dgamma(x, shape=gamma.shape, scale=gamma.scale), type="l", xlab="parallax / mas", ylab="density")
set.seed(100)

# EXT
par <- rnorm(n.ext, mean=0, sd=par.noisesd)
pm <- sqrt( (rnorm(n.ext, mean=0, sd=pm.noisesd))^2 + (rnorm(n.ext, mean=0, sd=pm.noisesd))^2 )
temp.ext <- data.frame(par, pm)
colnames(temp.ext) <- c("par", "pm")
temp.ext$cl <- "EXT"

# GAL
par.true  <- rgamma(n.gal, shape=gamma.shape, scale=gamma.scale)
par.noise <- rnorm(n.gal, mean=0, sd=par.noisesd)
par <- par.true + par.noise
pm <- sqrt( (rnorm(n.gal, mean=0, sd=velfac*par.true))^2 + (rnorm(n.gal, mean=0, sd=velfac*par.true))^2 )
temp.gal <- data.frame(par, pm)
colnames(temp.gal) <- c("par", "pm")
temp.gal$cl <- "GAL"
```

```
simast <- rbind(temp.ext, temp.gal)
EXT.ind <- 1:n.ext
GAL.ind <- n.ext+(1:n.gal)
rm(temp.ext, temp.gal, par, pm, par.true, par.noise)
par(mfrow=c(1,2))
plot(simast$par, simast$pm, type="n", xlab="parallax, mas", ylab="magnitude of proper motion, mas/yr")
points(simast[EXT.ind,]$par, simast[EXT.ind,]$pm, col="red")
points(simast[GAL.ind,]$par, simast[GAL.ind,]$pm, col="blue")
plot(simast$par, simast$pm, type="n", xlim=c(-1,1), ylim=c(0,5), xlab="parallax, mas", ylab="magnitude of proper motion, mas/yr")
points(simast[EXT.ind,]$par, simast[EXT.ind,]$pm, col="red")
points(simast[GAL.ind,]$par, simast[GAL.ind,]$pm, col="blue")

# 2D density plot of GAL astrometry
dense <- kde2d(simast[GAL.ind,]$par, simast[GAL.ind,]$pm, n=60, lims=c(0,5,0,50))
persp(dense, theta=50, phi=20, ltheta=100, lphi=20, shade=0.3, col="red", xlab="parallax, mas",
  ylab="proper motion, mas/yr", zlab="density", ticktype="detailed")
```