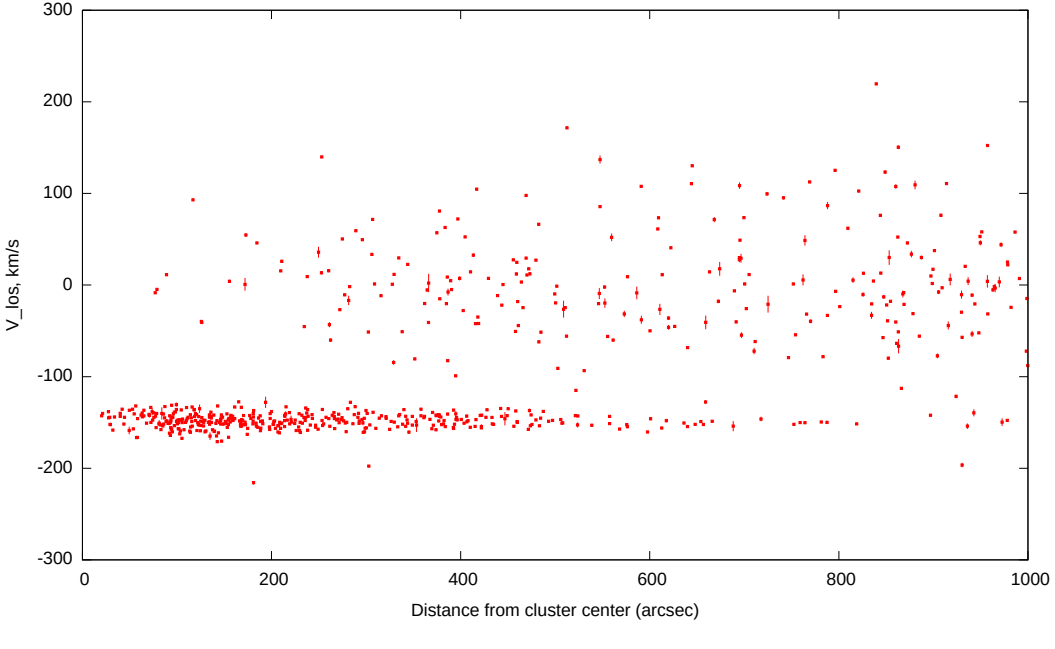
Mixture models and classification

Eugene Vasiliev

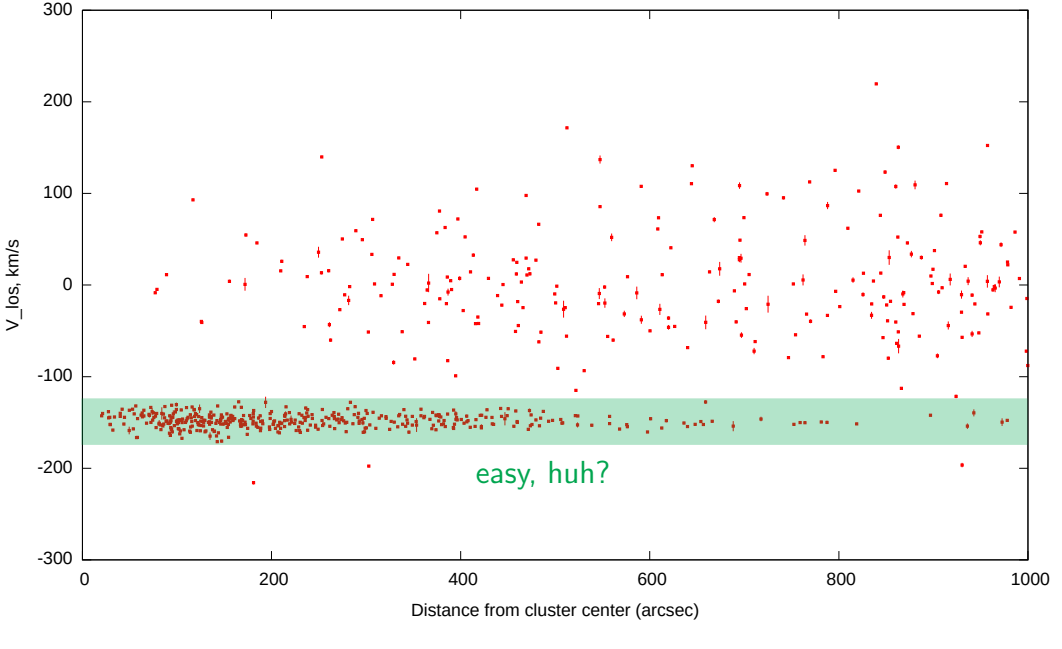
Institute of Astronomy, Cambridge

Heidelberg Physics Graduate Days, October 2020

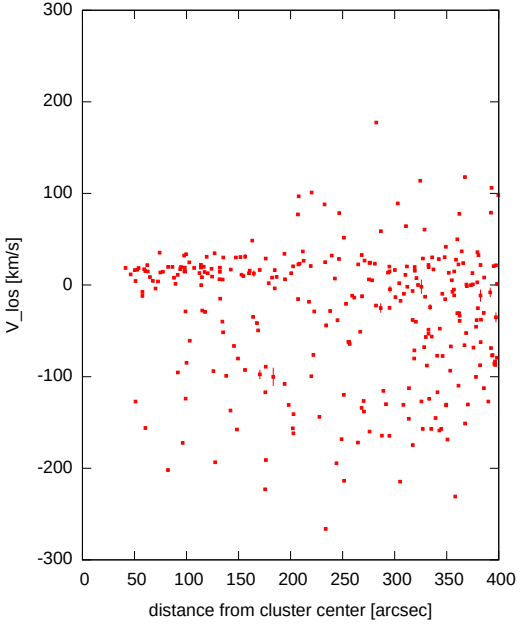
Line-of-sight velocities of globular clusters

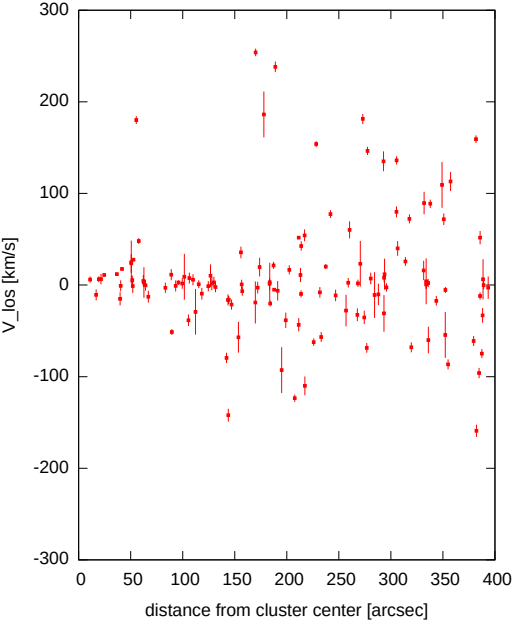


Line-of-sight velocities of globular clusters



Line-of-sight velocities of globular clusters





Outline

1. Likelihood-based inference on model parameters
2. Single-component model with measurement errors
3. Mixture model and outlier rejection
4. Mixture model with errors
5. Nonparametric models and further complexities

Food for thought:

Hogg, Bovy & Lang, "Data analysis recipes: fitting a model to data" (arXiv:1008.4686)

Kuhn & Feigelson, "Mixture models in astronomy" (arXiv:1711.11101)

Likelihood-based analysis: fit a Gaussian to the data

1. The model: a normal distribution with mean μ and variance σ^2:

$$x_i \sim \mathcal{N}(\mu, \sigma^2) \equiv \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left[-\frac{(x_i-\mu)^2}{2\,\sigma^2}\right]$$

2. The data: N precisely measured values x_i drawn from this distribution.
3. The likelihood function for the observed dataset given the model :

$$\mathscr{L}\big(\{x_i\} \mid \mu, \sigma\big) = \prod_{i=1}^{N} \mathcal{N}(x_i \mid \mu, \sigma^2), \quad \text{or}$$

$$\ln \mathscr{L} = -\tfrac{1}{2} \sum_{i=1}^{N} \frac{(x_i-\mu)^2}{\sigma^2} - N \ln \sigma - \tfrac{N}{2} \ln 2\pi$$

4. Vary the parameters (μ, σ) to maximize $\ln \mathscr{L}$: solve $\left\{ \dfrac{\partial \mathscr{L}}{\partial \mu} = 0, \dfrac{\partial \mathscr{L}}{\partial \sigma} = 0 \right\}$

Likelihood-based analysis: fit a Gaussian to the data

4. solve $\left\{ \dfrac{\partial \mathscr{L}}{\partial \mu} = 0, \dfrac{\partial \mathscr{L}}{\partial \sigma} = 0 \right\}$:

$$\frac{\partial \mathscr{L}}{\partial \mu} = \sum_{i=1}^{N} \frac{x_i - \mu}{\sigma^2} = 0 \quad \Longrightarrow \quad \mu = \frac{1}{N} \sum_{i=1}^{N} x_i,$$

$$\frac{\partial \mathscr{L}}{\partial \sigma} = \sum_{i=1}^{N} \frac{(x_i - \mu)^2}{\sigma^3} - \frac{N}{\sigma} = 0 \quad \Longrightarrow \quad \sigma = \frac{1}{N} \sum_{i=1}^{N} (x_i - \mu)^2.$$

5. The likelihood surface near its maximum is a paraboloid, and the covariance matrix of the uncertainties on model parameters is

$$C = \begin{pmatrix} \delta\mu^2 & \rho\, \delta\mu\, \delta\sigma \\ \rho\, \delta\mu\, \delta\sigma & \delta\sigma^2 \end{pmatrix} = - \begin{pmatrix} \dfrac{\partial^2 \ln \mathscr{L}}{\partial \mu^2} & \dfrac{\partial^2 \ln \mathscr{L}}{\partial \mu\, \partial \sigma} \\ \dfrac{\partial^2 \ln \mathscr{L}}{\partial \mu\, \partial \sigma} & \dfrac{\partial^2 \ln \mathscr{L}}{\partial \sigma^2} \end{pmatrix}^{-1}$$

Inferring the intrinsic dispersion from imprecise measurements

1. The model for the intrinsic distribution and the measurement process: the true values are drawn from a Gaussian with mean μ and variance σ^2, the observed values x_i are further perturbed by measurement errors δx_i

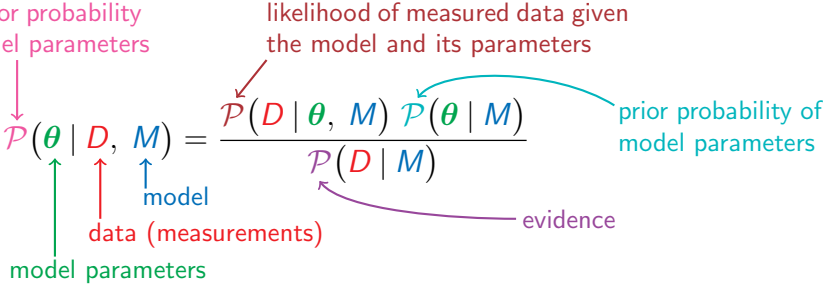
$$x_i \sim \underbrace{\mathcal{N}(\mu, \sigma^2) * \mathcal{N}(0, \delta x_i^2)}_{\text{convolution of two Gaussians is also a Gaussian}} = \mathcal{N}(\mu, \sigma_i^2), \quad \sigma_i^2 = \sigma^2 + {\delta x_i}^2$$

2. Write down the likelihood function $\mathscr{L}\big(\mu, \sigma \mid \{x_i, \delta x_i\}\big)$:

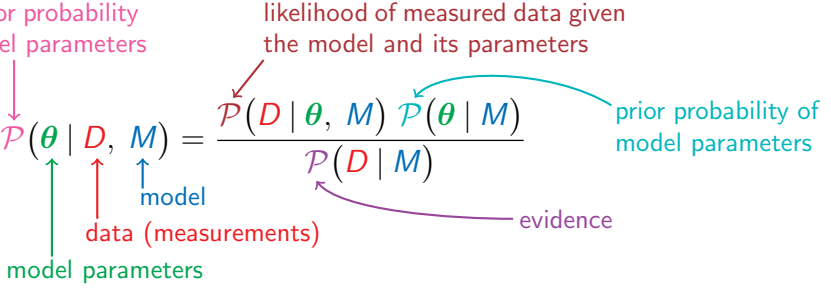
$$\ln \mathscr{L} = -\tfrac{1}{2} \sum_{i=1}^{N} \frac{(x_i - \mu)^2}{\sigma^2 + {\delta x_i}^2} - \tfrac{1}{2} \sum_{i=1}^{N} \ln\left(\sigma^2 + {\delta x_i}^2\right) - \tfrac{N}{2} \ln 2\pi$$

3. Vary the parameters (μ, σ) to maximize $\ln \mathscr{L}$: solve $\left\{ \dfrac{\partial \mathscr{L}}{\partial \mu} = 0, \dfrac{\partial \mathscr{L}}{\partial \sigma} = 0 \right\}$ – no explicit analytic expression, but straightforward to solve numerically
(note that the best-fit σ may be zero if the actual spread of measured values is smaller than the typical measurement error)

Bayesian formulation



Bayesian formulation



Posterior is a normalized probability distribution: $\int \mathcal{P}(\boldsymbol{\theta} \mid D,\ M)\ d\boldsymbol{\theta} = 1$,

hence the evidence is a "normalization factor":

$$\mathcal{P}(D \mid M) = \int \mathcal{P}(D \mid \boldsymbol{\theta},\ M)\ \underbrace{\mathcal{P}(\boldsymbol{\theta} \mid M)}_{\text{often this is a flat prior, i.e. constant [in some range]}}\ d\boldsymbol{\theta}$$

Evidence is useful only when choosing between alternative models M_1, M_2.

Treatment of measurement errors

D are measured (observed) data;

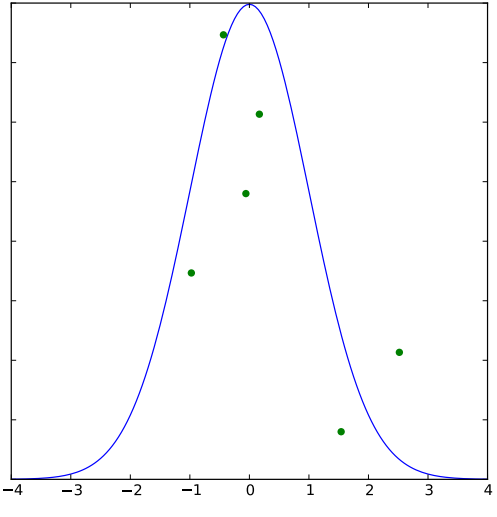
T are "true" (intrinsic) values predicted by the model M with parameters $\boldsymbol{\theta}$: $\mathcal{P}(T \mid \boldsymbol{\theta}, M)$ is the predicted distribution of true values;

$\mathcal{P}(D \mid T)$ is the measurement model: predicted distribution of D given T;

predicted distribution of observables is a marginalization over the [unknown] true values: $\mathcal{P}(D \mid \boldsymbol{\theta}, M) = \underbrace{\int \mathcal{P}(D \mid T)\, \mathcal{P}(T \mid \boldsymbol{\theta}, M)\, dT}_{\text{convolution with error distribution}}$

In the previous example, the model prediction was a normal distribution with two free parameters $\boldsymbol{\theta} = \{\mu, \sigma\}$: $\mathcal{P}(T \mid \boldsymbol{\theta}, M) = \mathcal{N}(T \mid \mu, \sigma)$, and the measurement model was a normal distribution with width δx_i: $\mathcal{P}(D \mid T) = \mathcal{N}(D \mid T, \delta x_i)$.

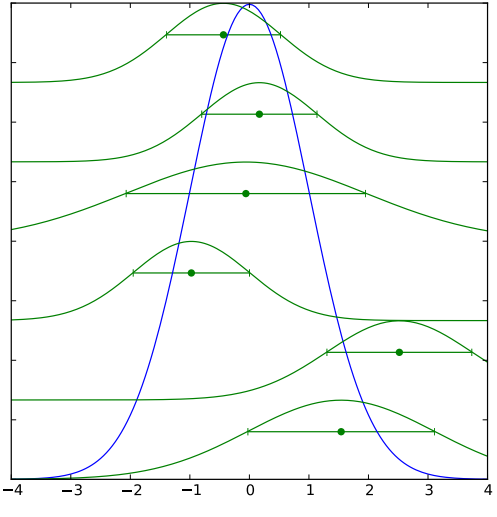
Inferring the intrinsic dispersion



blue: intrinsic distribution

green: true values of sampled points

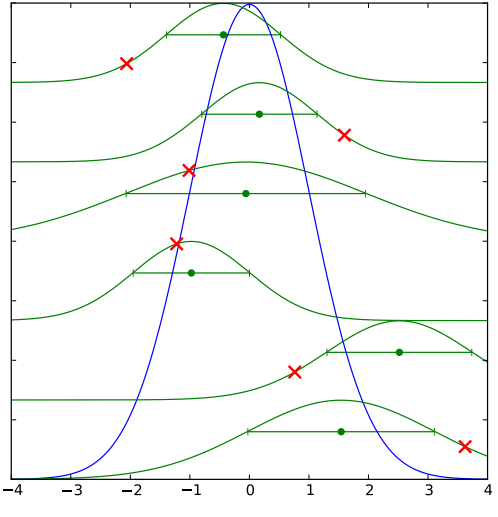
Inferring the intrinsic dispersion



blue: intrinsic distribution

green: true values of sampled points with measurement uncertainties

Inferring the intrinsic dispersion

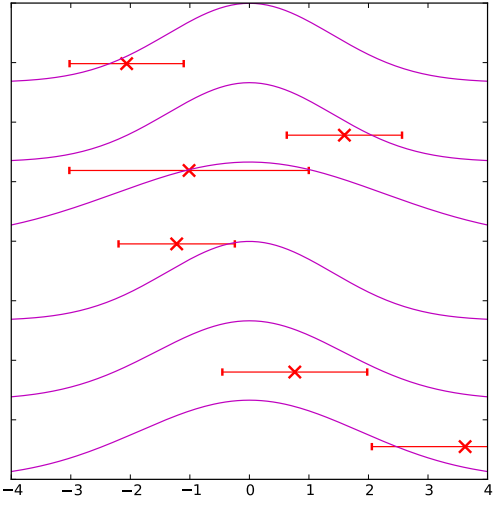


blue: intrinsic distribution

green: true values of sampled points
with measurement uncertainties

red: measured values (perturbed by errors)

Inferring the intrinsic dispersion



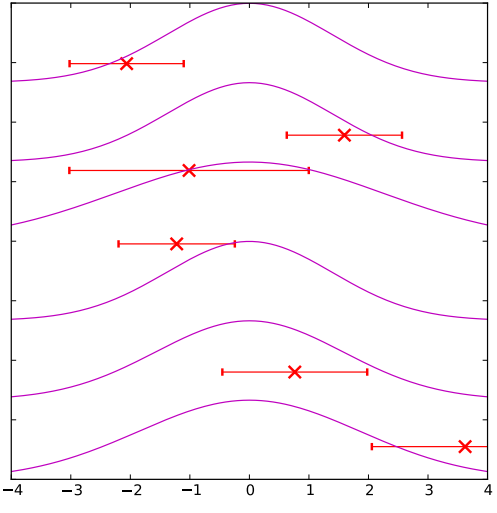
blue: intrinsic distribution

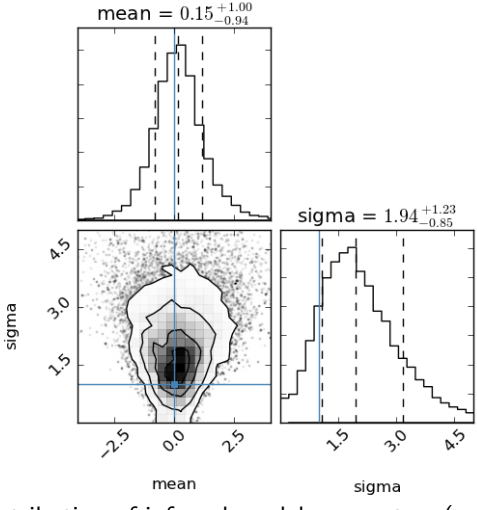
green: true values of sampled points
with measurement uncertainties

red: measured values (perturbed by errors)

magenta: prob.distrib. for each measured point

Inferring the intrinsic dispersion: deconvolution





distribution of inferred model parameters (μ, σ) obtained from a Markov Chain Monte Carlo run

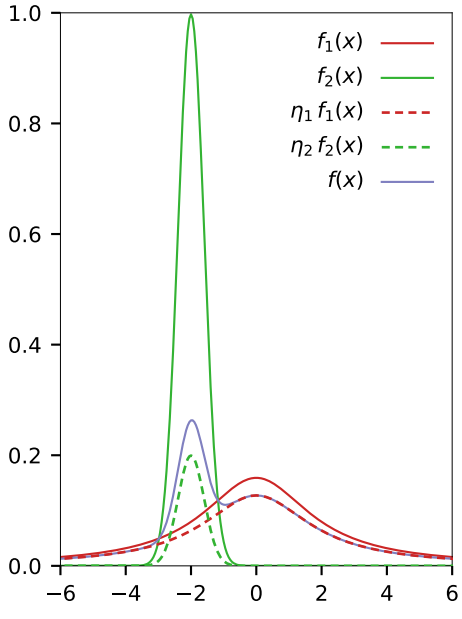
Membership determination from mixture modelling

N stars with observed properties $\mathbf{x}_i$, $i = 1..N$ (e.g., position, velocity, colours, etc.);

The entire dataset consists of C populations; i-th star belongs to the component with index a_i.

The distribution of values of $\mathbf{x}$ among stars of c-th component is described by probability distributions $f_c(\mathbf{x} \mid \boldsymbol{\theta})$ with some (unknown) parameters $\boldsymbol{\theta}$, normalized to unity: $\int f_c(\mathbf{x})\, d\mathbf{x} = 1$.

The mixture DF is a weighted sum of component DFs:
$f(\mathbf{x} \mid \boldsymbol{\theta}) = \sum_{c=1}^{C} \eta_c\, f_c(\mathbf{x} \mid \boldsymbol{\theta})$,
where η_c is the fraction of stars in c-th component, and $\sum_{c=1}^{C} \eta_c = 1$.



Membership determination from mixture modelling

The log-likelihood of the observed dataset, given the model parameters $(\boldsymbol{\theta}, \boldsymbol{\eta})$, is

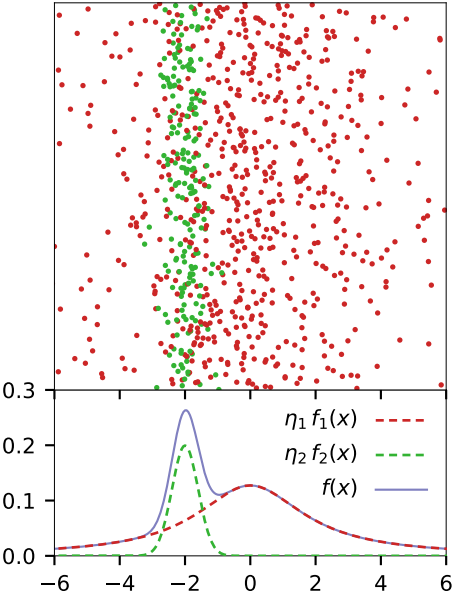
$$\ln \mathscr{L} = \sum_{i=1}^{N} \ln \mathscr{L}_i, \quad \text{where}$$

$$\mathscr{L}_i \equiv f_{a_i}(\mathbf{x}_i \mid \boldsymbol{\theta}) = \sum_{c=1}^{C} \delta_{c\,a_i}\, f_c(\mathbf{x}_i \mid \boldsymbol{\theta}).$$

However, since we do not know the indices a_i, we use the mixture DF:

$$\mathscr{L}_i \equiv f(\mathbf{x}_i \mid \boldsymbol{\theta}) = \sum_{c=1}^{C} \eta_c\, f_c(\mathbf{x}_i \mid \boldsymbol{\theta}).$$

As usual, the best-fit model parameters may be inferred by maximizing $\ln \mathscr{L}$ (optionally with some priors).

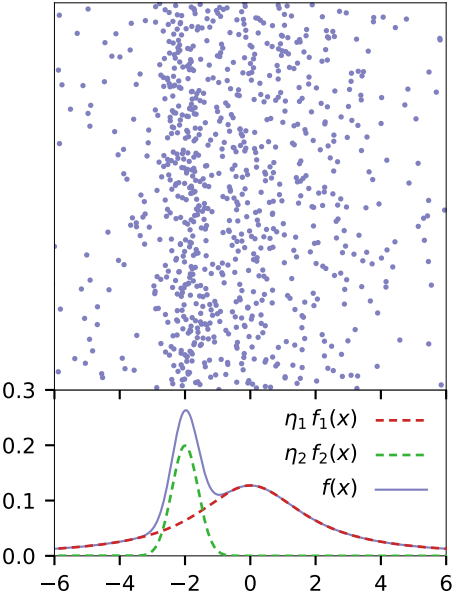


Membership determination from mixture modelling

Assume first that we know the parameters for all DFs $\boldsymbol{\theta}$ and their fractions η_c, but do not know which star belongs to which component.

η_c are prior membership probabilities (identical for all stars), while the posterior probabilities for i-th star with measured properties $\mathbf{x}_i$ are

$$p_i^{(c)} = \frac{\eta_c \, f_c(\mathbf{x}_i \mid \boldsymbol{\theta})}{\sum_{k=1}^{C} \eta_k \, f_k(\mathbf{x}_i \mid \boldsymbol{\theta})} \ .$$

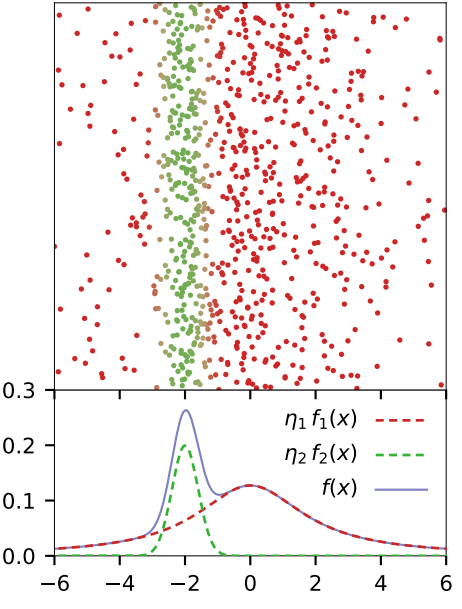


Membership determination from mixture modelling

Assume first that we know the parameters for all DFs $\boldsymbol{\theta}$ and their fractions η_c, but do not know which star belongs to which component.

η_c are prior membership probabilities (identical for all stars), while the posterior probabilities for i-th star with measured properties $\mathbf{x}_i$ are

$$p_i^{(c)} = \frac{\eta_c \, f_c(\mathbf{x}_i \mid \boldsymbol{\theta})}{\sum_{k=1}^{C} \eta_k \, f_k(\mathbf{x}_i \mid \boldsymbol{\theta})} \ .$$



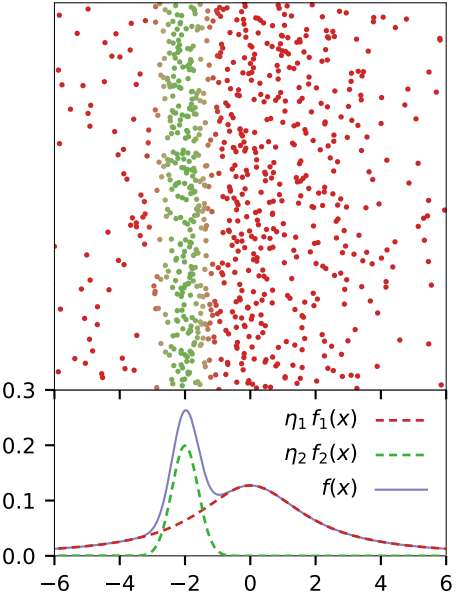
Membership determination from mixture modelling

Assume first that we know the parameters for all DFs $\boldsymbol{\theta}$ ~~and their fractions η_c~~, but do not know which star belongs to which component.

η_c are prior membership probabilities (identical for all stars), while the posterior probabilities for i-th star with measured properties $\mathbf{x}_i$ are

$$p_i^{(c)} = \frac{\eta_c \, f_c(\mathbf{x}_i \mid \boldsymbol{\theta})}{\sum_{k=1}^{C} \eta_k \, f_k(\mathbf{x}_i \mid \boldsymbol{\theta})} \; .$$

At the same time, $\eta_c = \dfrac{1}{N} \sum_{i=1}^{N} p_i^{(c)}$, so the fractions can be computed alongside membership probabilities.



Membership determination from mixture modelling

Now that we know (probabilistically) the membership of each point $p_i^{(c)}$, we may update the parameters of the DFs $\boldsymbol{\theta}$:

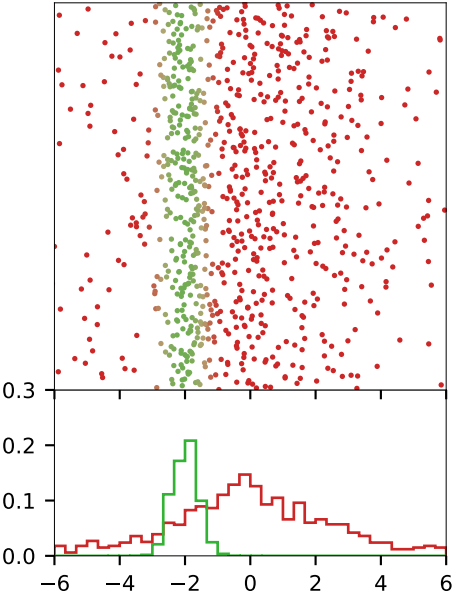
$$\boldsymbol{\theta}^{(new)} = \arg\max_{\boldsymbol{\theta}} \left(\ln \mathscr{L} \right)$$

$$= \arg\max_{\boldsymbol{\theta}} \left(\sum_{i=1}^{N} \sum_{c=1}^{C} p_i^{(c)} \ln f_c(\mathbf{x}_i \mid \boldsymbol{\theta}) \right).$$

Fit the parameters $\boldsymbol{\theta}$ of each DF f_c to the measured values $\mathbf{x}_i$, weighted by probabilities $p_i^{(c)}$.

$f_c(\mathbf{x})$ may have any suitable functional form:
– a Gaussian ($\boldsymbol{\theta}$ are the mean and dispersion);
– a histogram ($\boldsymbol{\theta}$ are the bin heights);
– ...

Repeat these steps until convergence:
this is the expectation/maximization algorithm.



Membership determination from mixture modelling

Expectation/maximization (EM) is one possible way of finding the maximum-likelihood solution for the mixture DF – it gives only the best-fit solution, but no associated uncertainty on η_c and $\boldsymbol{\theta}_c$. One can compute them from the Hessian of the likelihood function $\mathscr{L}(\{\mathbf{x}_i\} \mid \boldsymbol{\eta}, \boldsymbol{\theta})$ at its maximum, or by running a MCMC simulation instead of the EM algorithm. In the latter case, it is also straightforward to marginalize over the nuisance parameters.

Outlier rejection and model fitting with imprecise data

1. Assume a two-component model specified by two intrinsic distributions, convolved with individual measurement errors for each datapoint:
   - points belonging to the object of interest: $x_i \sim \mathcal{F}_{\rm int}(\boldsymbol{\theta}) * \mathcal{N}(0, \delta x_i)$
   - ouliers: $x_i \sim \mathcal{F}_{\rm out}(\boldsymbol{\zeta}) * \mathcal{N}(0, \delta x_i)$

   could be a Gaussian, but not necessarily

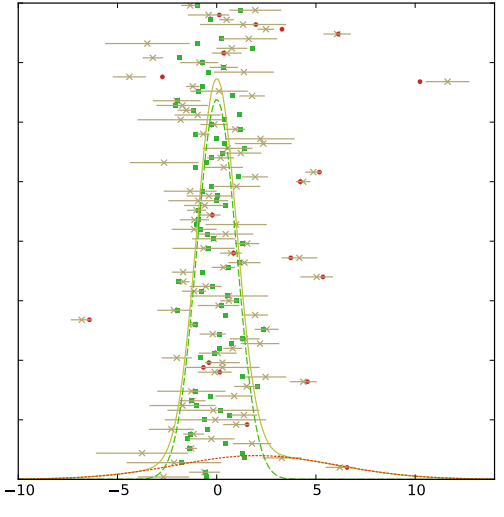
   $\boldsymbol{\theta}, \boldsymbol{\zeta}$ are the parameters of the intrinsic distributions (e.g., mean and width of Gaussians)

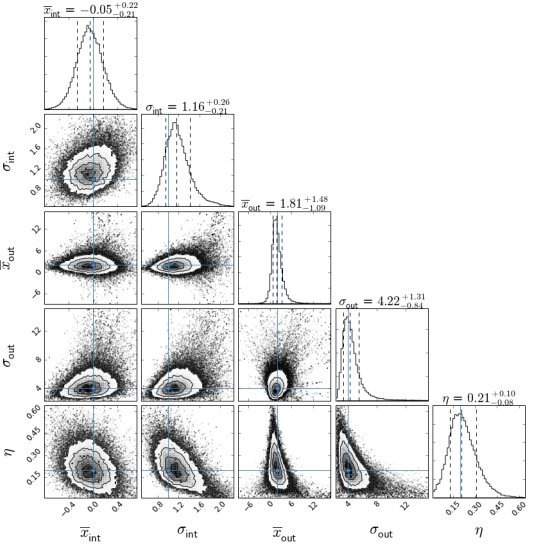
2. Assume that an (unknown) fraction η of all datapoints are outliers: the probability distribution of the mixture model for datapoint x_i is $x_i \sim \mathcal{F}_{{\rm mix},i} \equiv \big[(1-\eta)\,\mathcal{F}_{\rm int}(\boldsymbol{\theta}) + \eta\,\mathcal{F}_{\rm out}(\boldsymbol{\zeta})\big] * \mathcal{N}(0, \delta x_i)$, and the likelihood of the entire model is $\mathscr{L}\big(\boldsymbol{\theta}, \boldsymbol{\zeta}, \eta \mid \{x_i, \delta x_i\}\big) = \prod_{i=1}^{N} \mathcal{F}_{{\rm mix},i}$

3. Define suitable priors $\mathcal{P}$ for the nuisance parameters $\boldsymbol{\zeta}, \eta$

4. Obtain the posterior probability distribution for the parameters of interest $\boldsymbol{\theta}$ by marginalizing over the nuisance parameters $\boldsymbol{\zeta}, \eta$: $\mathscr{P}\big(\boldsymbol{\theta} \mid \{x_i\}\big) = \int \mathscr{L}\big(\boldsymbol{\theta}, \boldsymbol{\zeta}, \eta \mid \{x_i\}\big)\, \mathcal{P}(\boldsymbol{\zeta}, \eta)\, d\boldsymbol{\zeta}\, d\eta$; determine the confidence intervals for $\boldsymbol{\theta}$ from this posterior distribution

Example: measuring the dispersion and pruning outliers





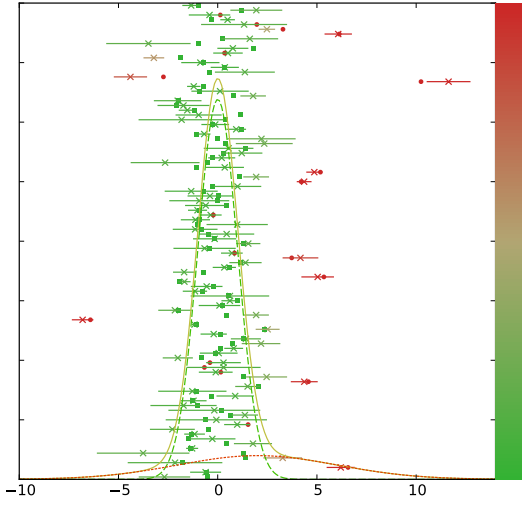
Mixture models and classification

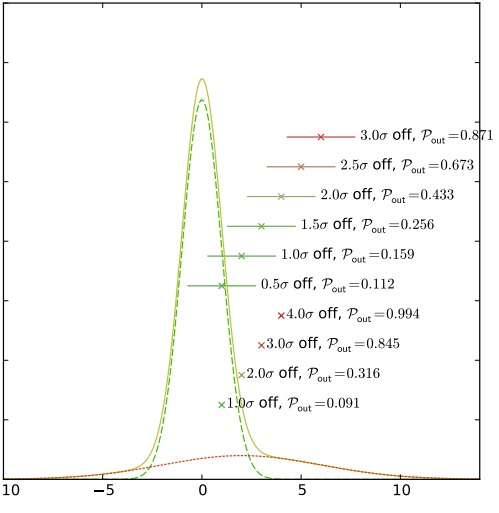
- Each datapoint has the same prior probability η of being an outlier; however, the posterior probability does depend on the measured value x_i and its uncertainty δx_i, as well as the model parameters $\boldsymbol{\theta}$, $\boldsymbol{\zeta}$:

$$\mathcal{P}_{\mathrm{out}}(x_i, \delta x_i \mid \boldsymbol{\theta}, \boldsymbol{\zeta}, \eta) = \frac{\left[\eta\, \mathcal{F}_{\mathrm{out}}(\boldsymbol{\zeta}) * \mathcal{N}(0, \delta x_i)\right](x_i)}{\left[\left\{(1-\eta)\, \mathcal{F}_{\mathrm{int}}(\boldsymbol{\theta}) + \eta\, \mathcal{F}_{\mathrm{out}}(\boldsymbol{\zeta})\right\} * \mathcal{N}(0, \delta x_i)\right](x_i)}$$

- For the best-fit values of parameters, $\sum_{i=1}^{N} \mathcal{P}_{\mathrm{out}}(x_i, \delta x_i) = N\,\eta$
- There is no single "N-σ" criterion: if there were no model for outliers, one couldn't reject a point even when it is 10σ off!
- The dataset should contain enough contaminants to reliably infer their fraction η and the parameters $\boldsymbol{\zeta}$ for $\mathcal{F}_{\mathrm{out}}$
- Probabilistic membership classification should be carried onward to subsequent modelling procedures, if possible.

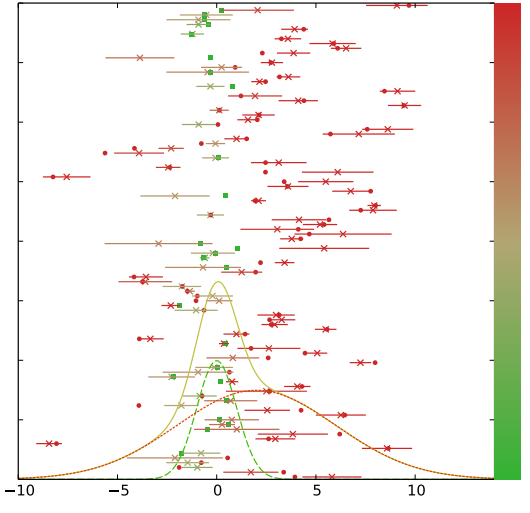
Mixture models, classification and rejection of outliers

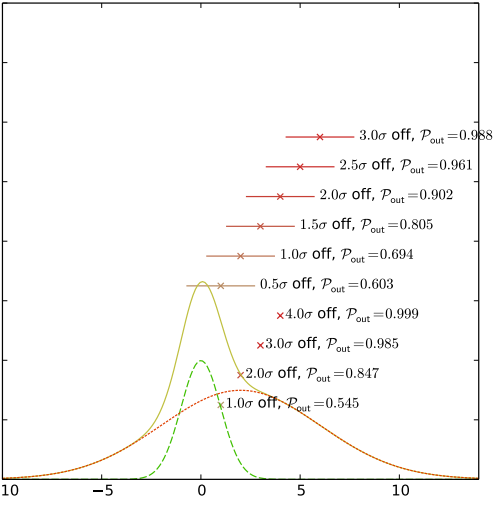




20% outliers

Mixture models, classification and rejection of outliers





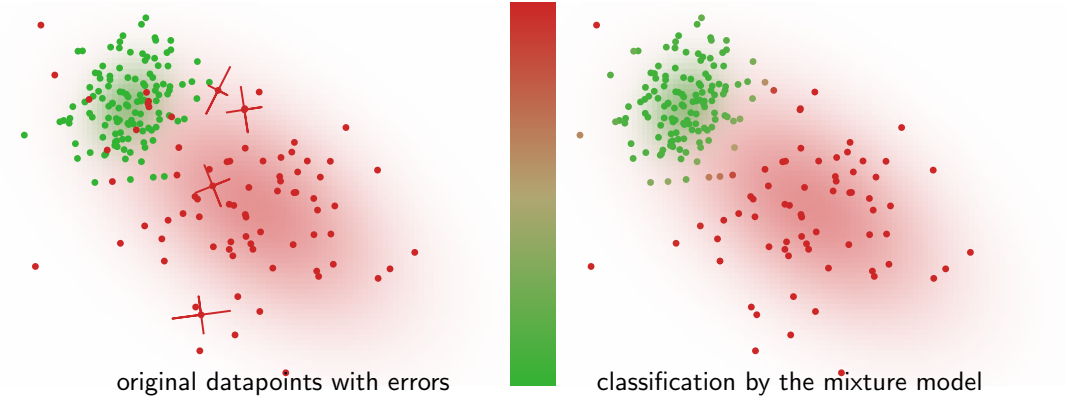
75% outliers

Multidimensional case

- D-dimensional Gaussian with mean $\boldsymbol{\mu}$ and covariance matrix Σ:

$$\mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}, \Sigma) = \frac{1}{\sqrt{(2\pi)^D \det \Sigma}} \exp\left[-\tfrac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x}-\boldsymbol{\mu}) \right]$$

- Measurement errors for i-th datapoint described by error covariance matrix δx_i
- Convolution of two Gaussians is also a Gaussian with covariance $\Sigma + \delta x_i$

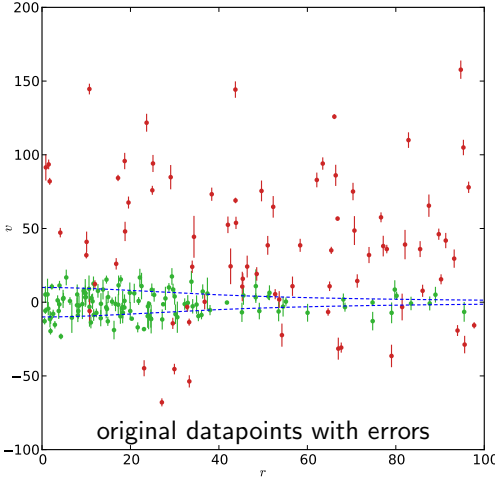


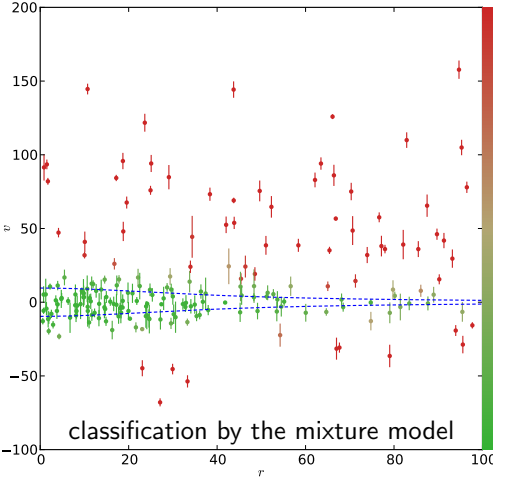
original datapoints with errors

classification by the mixture model

Additional parameters in the model

The probability distribution $\mathcal{F}_{\text{int}}$ and the fraction of outliers η
may depend on some additional parameters ς and measured properties $\{\xi_i\}$
(e.g., scale radius a, the distance R_i of a star from the cluster center, etc.)





Fitting dynamical models to discrete-kinematic data

Example: Plummer-like model for the cluster, uniform contamination:

$$\mathcal{F}_{\rm int}\big(\{R_i, v_i, \delta v_i\} \mid a, \overline{v}, \sigma_0\big) = \mathcal{N}\Big(v_i \mid \overline{v}, \sqrt{\sigma^2(R_i) + \delta v_i^2}\Big)\, \frac{1 + (R_{\rm max}/a)^2}{\big[1 + (R_i/a)^2\big]^2}$$

scale radius — cluster mean velocity — central velocity dispersion

$$\sigma(R_i \mid a, \sigma_0) \equiv \frac{\sigma_0}{\big[1 + (R_i/a)^2\big]^{1/4}}$$

normalized surface density

mean and dispersion of contaminants

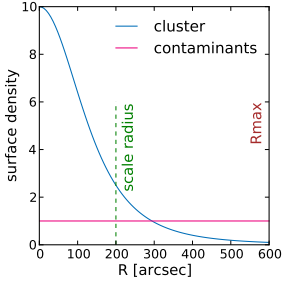
$$\mathcal{F}_{\rm out}\big(\{v_i, \delta v_i\} \mid \overline{v}_{\rm out}, \sigma_{\rm out}\big) = \mathcal{N}\Big(v_i \mid \overline{v}_{\rm out}, \sqrt{\sigma_{\rm out}^2 + \delta v_i^2}\Big)$$

Distribution function of the mixture model:

$$\mathcal{F}_{\rm mix} = (1 - \eta)\, \mathcal{F}_{\rm int} + \eta\, \mathcal{F}_{\rm out}$$

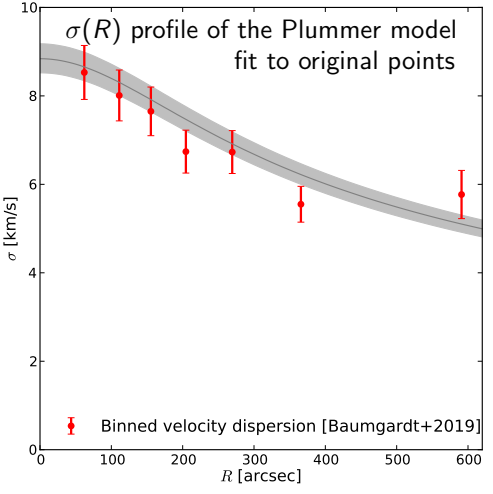
parameters: $a, \overline{v}, \sigma_0, \overline{v}_{\rm out}, \sigma_{\rm out}, \eta$

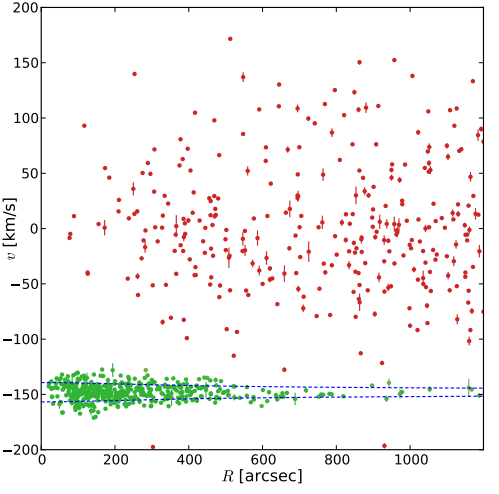
fraction of contaminants



Fitting dynamical models to discrete-kinematic data

Example: globular cluster NGC 6656

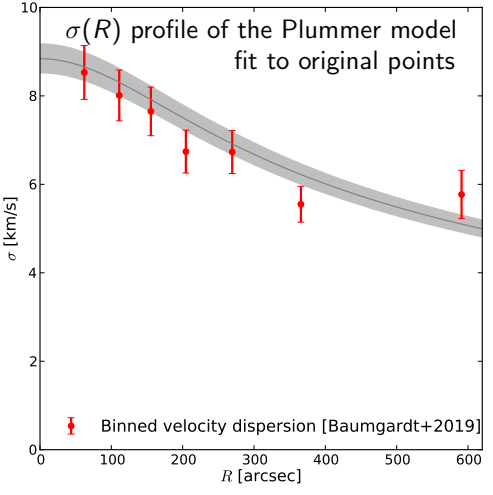


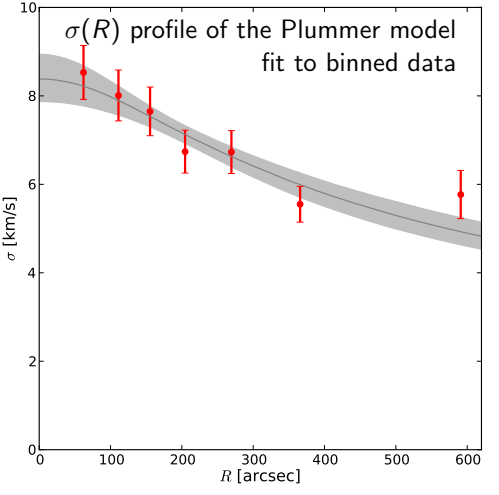


Fitting dynamical models to discrete-kinematic data

Example: globular cluster NGC 6656

General problem with parametric models:
inferred uncertainty intervals are too small

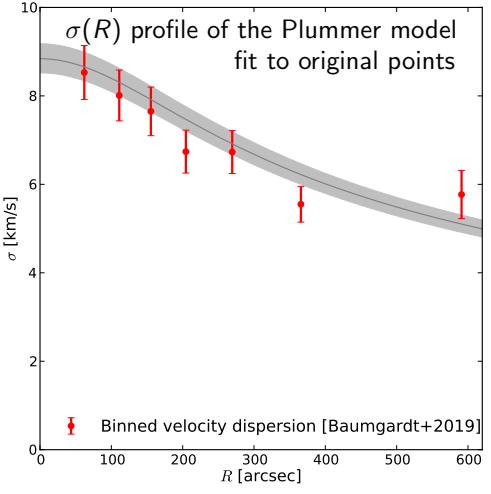


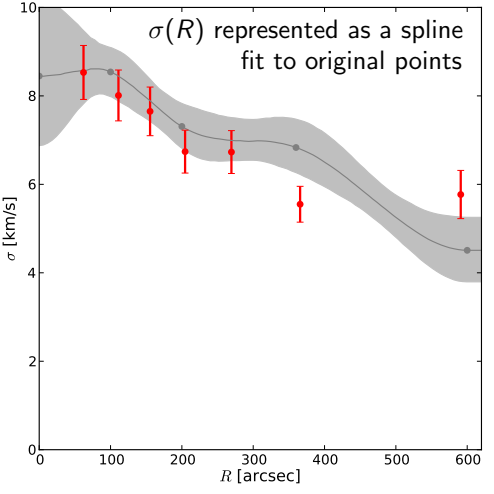


Fitting "nonparametric" models to discrete-kinematic data

Consider a more flexible (but not dynamically motivated) dispersion profile: $\sigma(R)$ represented as a cubic spline with fixed nodes and free coefficients.

Similar uncertainties as binned data, but in a mathematically consistent model.

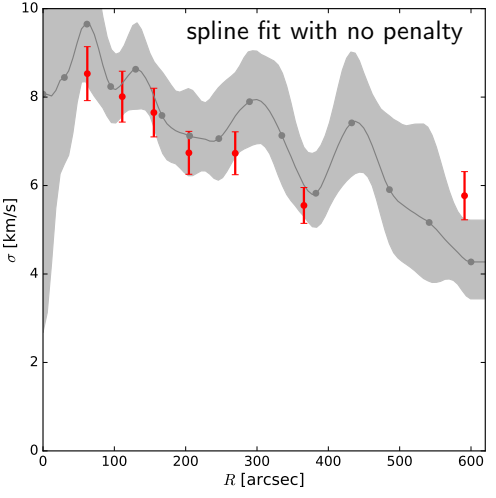


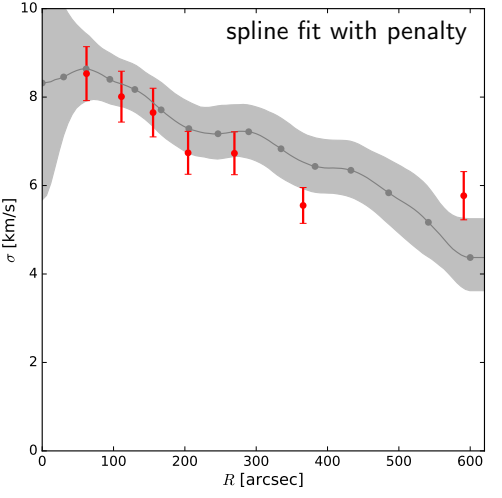


Penalization of overly flexible models

If one has too many free parameters, the models would overfit the data.

To prevent this, one needs to impose stronger priors on the parameters, e.g., penalizing large variations between adjacent spline values.





Penalization of overly flexible models

If one has too many free parameters, the models would overfit the data.

To prevent this, one needs to impose stronger priors on the parameters, e.g., penalizing large variations between adjacent spline values.

There are objective methods for determining the optimal value of penalty (smoothing parameter), based on cross-validation:

- adopt a particular value of the smoothing parameter λ;
- split the data sample into two parts – training set and validation set;
- find the best-fit model for the training set;
- evaluate the goodness-of-fit of this model for the validation set;
- average this over different choices of training and validation subsets;
- adjust the parameter λ to maximize this validation score;

The same approach is used in machine learning to prevent overfitting

Pros and cons of fitting probabilistic mixture models

- \+ outlier rejection or fitting multiple populations are straightforward
- \+ easy to account for selection function and measurement uncertainties
- \+ use all available information (individual datapoints, no binning)
- \+ can propagate uncertainty in membership into subsequent analysis
- − computationally demanding in case of large datasets
- − marginalization and convolution often non-analytic $\Rightarrow$ expensive

The most common variant of this technique is the Gaussian mixture modelling, with several implementations available in Python:

- scikit-learn (no measurement errors)
- Extreme Deconvolution [Bovy, Hogg & Roweis 2011]
- XDGMM [Holoien, Marshall & Wechsler 2016] (also in AstroML)
- PyGMMis [Melchior & Goulding 2018]

All these variants use the expectation/maximization approach and give only the best-fit parameters, without uncertainties.

Exercise: simple 1d Gaussian mixture models

Scenario: a dataset of N points drawn from a mixture of two populations: broad and narrow.

- Write a routine for generating the mock dataset (assuming precise measurements).
- Write another routine for computing the likelihood of the dataset given a model with some free parameters – a sum of two Gaussians with unknown mean, dispersion, and relative weights.
- Use your favourite method to find the best-fit parameters (e.g., direct maximization of the likelihood function, or the expectation/maximization algorithm, or MCMC).
- Extend the procedure to the case of measurement errors (preferrably, varying between datapoints) – both for the mock data generation and for the fit.
- Fiducial values: $N \simeq 100$, fraction of broad component (contaminants) between 10 and 90%, measurement uncertainties between 0 and $1 - 2\times$ the dispersion of the narrow component.

Exercise 2: application to globular clusters

Many globular clusters in the Galaxy have plenty of stars with measured values of line-of-sight velocity; a good starting point is the catalogue of Baumgardt+ 2019: https://people.smp.uq.edu.au/HolgerBaumgardt/globular/appendix/appendix.html

The tables for individual clusters contain both members and non-members. For each star, the table contains the value and uncertainty of $v_{\rm los}$, distance from the cluster centre, and the membership probability – we will not use the latter quantity in the fit, but will compare it with the results of the mixture modelling classifier. Ideally, the distance information can be used to estimate the density profile of the cluster, but the coverage of the spectroscopic dataset is very non-uniform, making this task difficult. We may instead restrict the range of distances to a few arcmin (perhaps depending on the cluster under consideration).

The goal is to measure the mean velocity and its dispersion, and to estimate the membership probability of all stars in the dataset. The results can be compared with the original paper (which used a different approach), shown in the table of structural parameters (for σ) and orbits (for $\overline{v_{\rm los}}$).