Linear Regression with Measurement Errors

Marshall Bradley

The problem of fitting a straight line $y = ax + b$ to measured data when there are errors in the measurement of both the x and y is considered. Additionally the effects of intrinsic dispersion on the fit are accounted for as well. Astrophysical data which describes the relationship between the log of a galaxy's's velocity dispersion and the log of the mass of the black hole located at the center of the galaxy is used to illustrate the technique. The unknown parameters are (a, b, σ) where σ is the intrinsic dispersion associated with the black hole mass. The likelihood of the data in the presence of all three types of errors is derived using Bayes theorem and the technique of marginalization. Marginal posterior probability distributions are found for each parameter using brute force computation.

Introduction

The example presented here is adapted from (Andreon and Weaver, 2015). They solve the problem fitting a line $y = ax + b$ to data of the form (x_i, y_i) when there are errors in both the x_i and y_i as well as intrinsic dispersion in the y_i. They use data from (Tremaine, 2002) which describes the relationship between the log of a galaxy's's velocity dispersion and the log of the mass of the black hole located at the center of the galaxy. Their solution is based upon the use of standardized Bayesian analysis packages. We work from first principles here and make all the necessary Bayesian computations using a brute force approach.

Discussion

The super massive, black hole velocity dispersion relationship $(M - \sigma_v)$ relationship is an empirical power law relationship between the mass M of the black hole and the velocity dispersion σ_v of the bulge in the galaxy which the black hole resides. The super massive black hole is located at the center of the bulge and has a mass that is millions to billions of times the mass of the sun. The velocity dispersion is the standard deviation of the velocities of the objects comprising the bulge. The general form of this relationship is

$$\frac{M_{\text{BH}}}{10^8 M_{\text{sun}}} = \beta\left(\frac{\sigma_v}{200\ \text{km/s}}\right)^a.$$

Merritt (1969) proposed

$$\frac{M_{BH}}{10^8\, M_{sun}} = 3.1 \left(\frac{\sigma_v}{200\text{ km/s}}\right)^4.$$

On a log scale these relations are of the form

$$\log_{10}(M_{BH} / 10^8\, M_{sun}) = \log_{10}\beta + \alpha \log_{10}(\sigma_v / 200\text{ km/s}).$$

In order to simplify the discussion we adopt the notation $y = \log_{10}(M_{BH} / 10^8\, M_{sun})$, $x = \log_{10}(\sigma_v / 200\text{ km/s})$ and $b = \log_{10}\beta$. With this notation in hand, the relationship between the mass of the supper-massive black hole and the velocity dispersion of the galaxy bulge is linear:

$$y = a x + b.$$

An example of a spiral galaxy is shown in the following:

In[ ]:= Thumbnail[, Large]

Out[ ]=

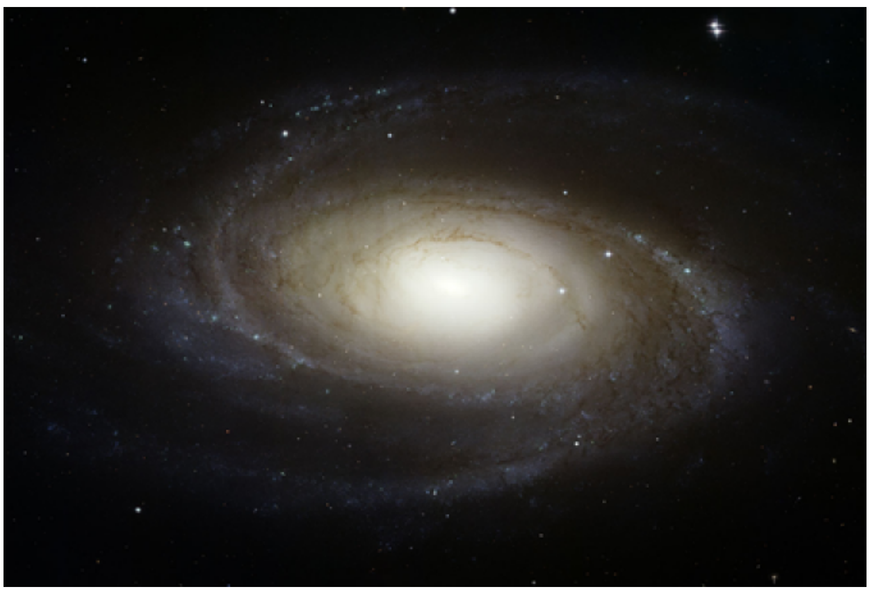


Figure 1. Grand Design spiral galaxy M81 is a spiral-shaped system of stars, dust, and clouds of gas. The arms of the galaxy wind all the way back into the galaxy bulge (also called nucleus). See https://en.wikipedia.org/wiki/Galactic_bulge and https://cdn.spacetelescope.org/archives/images/screen/heic0710a.jpg.

The Magorrian relation data presented here were taken from table 8.3 of Andreon and Weaver (2015). They in turn took the data from Tremaine (2002).

The data are:

In[ ]:=
```
observedX = {2.01284, 1.87506, 2.20412, 2.32015, 2.31175, 2.17898, 2.24304,
   2.14613, 2.36173, 2.31175, 2.16137, 2.31387, 2.15534, 2.26007, 2.11394,
   2.49831, 2.38382, 2.35218, 2.26951, 2.27875, 2.57403, 2.20952, 2.18184,
   2.58546, 2.24797, 1.95424, 2.36922, 2.4624, 2.42488, 1.82607, 2.53148};
observedErrorX = {0.0791813, 0.0211868, 0.0211945, 0.0211945, 0.0211945,
   0.0211945, 0.0211945, 0.0211945, 0.0211945, 0.0211945, 0.0211945, 0.0211945,
   0.0211945, 0.0211945, 0.0211945, 0.0211792, 0.0211945, 0.0211945, 0.0211945,
   0.0211945, 0.0211792, 0.0211945, 0.0211945, 0.0211792, 0.0211945,
   0.0211868, 0.0211945, 0.0211792, 0.0211792, 0.0211868, 0.0211792};
observedY = {6.26717, 6.39794, 7.72016, 7.65321, 7.64345, 7.30103, 7.13033,
   7.60746, 9.07918, 8.32222, 8.14613, 8.02119, 7.19033, 8.32222, 7.59106,
   8.71181, 8.37107, 8.52504, 7.8451, 7.95665, 9.47712, 7.72835, 7.91645,
   9.27875, 8.24304, 7.13033, 8.27875, 8.72016, 8.57978, 6.52504, 9.40654};
observedErrorY =
  {0.0831657, 0.0880456, 0.314194, 0.161466, 0.0495657, 0.238561, 0.321726, 0.048455,
   0.349485, 0.105427, 0.162256, 0.252575, 0.0421605, 0.174227, 0.0111382, 0.0898039,
   0.343987, 0.185534, 0.0816016, 0.342365, 0.150515, 0.0448054, 0.212984, 0.117042,
   0.0373168, 0.150515, 0.223579, 0.150515, 0.223579, 0.170269, 0.077451};
```

A plot of the data with error bars is:

In[∘]:=
```
pnts = Transpose[{observedX, observedY}];
pntsLeftX = Transpose[{observedX - observedErrorX, observedY}];
pntsRightX = Transpose[{observedX + observedErrorX, observedY}];
pntsUpY = Transpose[{observedX, observedY + observedErrorY}];
pntsDownY = Transpose[{observedX, observedY - observedErrorY}];
errorLines =
  Join[Transpose[{pntsLeftX, pntsRightX}], Transpose[{pntsDownY, pntsUpY}] ];
errorLines = {Black, Thin, Map[Line, errorLines]};
gMain0 = ListPlot[Transpose[{observedX, observedY}] , Prolog → {errorLines} , ⋯ ]
```

Out[∘]=

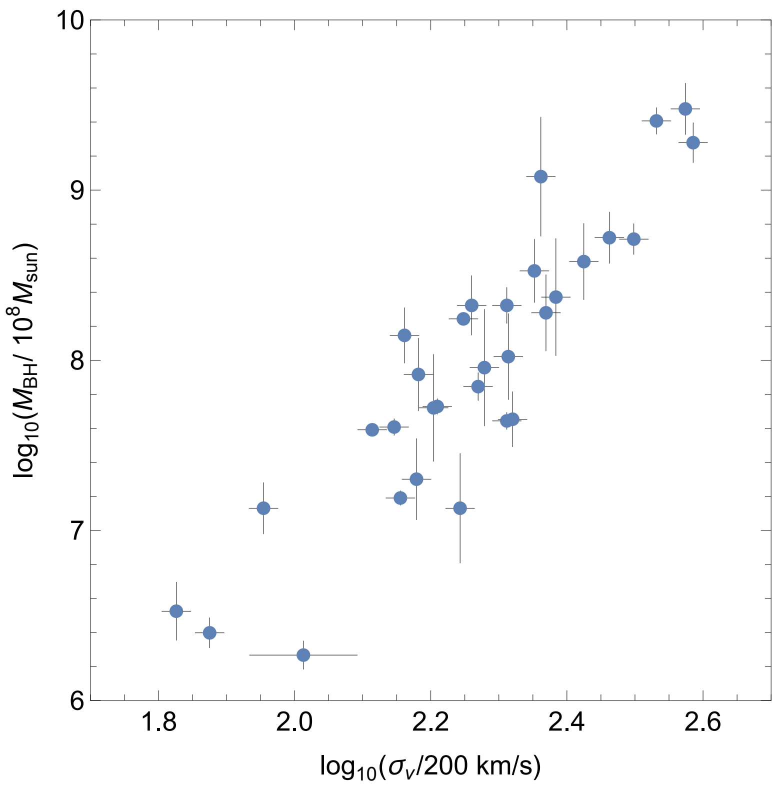


Figure 2. $M - \sigma_v$ data plotted with one standard deviation error bars.

In tabular form the data look like this:

x	σ_x	y	σ_y	x	σ_x	y	σ_y
2.01284	0.07918⋱ 13	6.26717	0.08316⋱ 57	2.38382	0.02119⋱ 45	8.37107	0.343987
1.87506	0.02118⋱ 68	6.39794	0.08804⋱ 56	2.35218	0.02119⋱ 45	8.52504	0.185534
2.20412	0.02119⋱ 45	7.72016	0.314194	2.26951	0.02119⋱ 45	7.8451	0.08160⋱ 16
2.32015	0.02119⋱ 45	7.65321	0.161466	2.27875	0.02119⋱ 45	7.95665	0.342365
2.31175	0.02119⋱ 45	7.64345	0.04956⋱ 57	2.57403	0.02117⋱ 92	9.47712	0.150515
2.17898	0.02119⋱ 45	7.30103	0.238561	2.20952	0.02119⋱ 45	7.72835	0.04480⋱ 54
2.24304	0.02119⋱ 45	7.13033	0.321726	2.18184	0.02119⋱ 45	7.91645	0.212984
2.14613	0.02119⋱ 45	7.60746	0.048455	2.58546	0.02117⋱ 92	9.27875	0.117042
2.36173	0.02119⋱ 45	9.07918	0.349485	2.24797	0.02119⋱ 45	8.24304	0.03731⋱ 68
2.31175	0.02119⋱ 45	8.32222	0.105427	1.95424	0.02118⋱ 68	7.13033	0.150515
2.16137	0.02119⋱ 45	8.14613	0.162256	2.36922	0.02119⋱ 45	8.27875	0.223579
2.31387	0.02119⋱ 45	8.02119	0.252575	2.4624	0.02117⋱ 92	8.72016	0.150515
2.15534	0.02119⋱ 45	7.19033	0.04216⋱ 05	2.42488	0.02117⋱ 92	8.57978	0.223579
2.26007	0.02119⋱ 45	8.32222	0.174227	1.82607	0.02118⋱ 68	6.52504	0.170269
2.11394	0.02119⋱ 45	7.59106	0.01113⋱ 82	2.53148	0.02117⋱ 92	9.40654	0.077451
2.53148	0.02117⋱ 92	9.40654	0.077451				

Table 1. $M-\sigma_v$ data. In the table x denotes $\log_{10}(\sigma_v/200\text{ km/s})$ and y denotes $\log_{10}(M_{\text{BH}}/10^8 M_{\text{sun}})$. $x=0$ corresponds to 200 km/s and $y=0$ corresponds to 10^8 solar masses.

Derivation of the likelihood

The likelihood of observing the data pair (x_i, y_i) in the presences of errors in *x* and *y* as well as intrinsic scatter in *y* is:

In[∘]:=
```
Clear[a, b, σ, ξi, ηi, xi, yi];
likelihoodData =
```
$$\frac{1}{\sqrt{2\pi}\,\sigma}\,\frac{1}{\sqrt{2\pi}\,\sigma yi}\,\frac{1}{\sqrt{2\pi}\,\sigma xi}\,\mathrm{Exp}\left[-\frac{(\eta i - yi)^2}{2\,\sigma^2}\right]\,\mathrm{Exp}\left[-\frac{(\eta i - a * \xi i - b)^2}{2\,\sigma yi^2}\right]\,\mathrm{Exp}\left[-\frac{(\xi i - xi)^2}{2\,\sigma xi^2}\right];$$

The above equation contains two hidden values: ξ_i and η_i. These are the true *x* and *y* values that we do not observe. We are not interested in these parameters so we marginalize them out. First we marginalize out *x* in its factor:

In[∘]:=
$$\mathrm{Integrate}\left[\mathrm{Exp}\left[-\frac{(\eta i - a * \xi i - b)^2}{2\,\sigma yi^2}\right]\,\mathrm{Exp}\left[-\frac{(\xi i - xi)^2}{2\,\sigma xi^2}\right],\ \{\xi i, -\infty, \infty\},\ \mathrm{Assumptions} \to \{\sigma > 0, \sigma xi > 0, \sigma yi > 0, a > 0\}\right]$$

Out[∘]=
$$\frac{e^{-\frac{(b + a\,xi - \eta i)^2}{2\,(a^2\,\sigma xi^2 + \sigma yi^2)}}\,\sqrt{2\pi}\,\sigma xi\,\sigma yi}{\sqrt{a^2\,\sigma xi^2 + \sigma yi^2}}$$

Next we marginalize out *y* in its factor:

In[∘]:=
$$\mathrm{Integrate}\left[e^{-\frac{(b + a\,xi - \eta i)^2}{2\,(a^2\,\sigma xi^2 + \sigma yi^2)}}\,\mathrm{Exp}\left[-\frac{(\eta i - yi)^2}{2\,\sigma^2}\right],\ \{\eta i, -\infty, \infty\},\ \mathrm{Assumptions} \to \{\sigma > 0, \sigma xi > 0, \sigma yi > 0, a > 0\}\right]$$

Out[∘]=
$$e^{-\frac{(b + a\,xi - yi)^2}{2\,(\sigma^2 + a^2\,\sigma xi^2 + \sigma yi^2)}}\,\sqrt{\pi}\,\sigma\,\sqrt{2 - \frac{2\,\sigma^2}{\sigma^2 + a^2\,\sigma xi^2 + \sigma yi^2}}$$

The result that we want is obtained multiplying marginalized factors together and simplifying:

In[∘]:=
$$\mathrm{FullSimplify}\left[\frac{1}{\sqrt{2\pi}\,\sigma}\,\frac{1}{\sqrt{2\pi}\,\sigma yi}\,\frac{1}{\sqrt{2\pi}\,\sigma xi}\,\frac{\sqrt{2\pi}\,\sigma xi\,\sigma yi}{\sqrt{a^2\,\sigma xi^2 + \sigma yi^2}}\,e^{-\frac{(b + a\,xi - yi)^2}{2\,(\sigma^2 + a^2\,\sigma xi^2 + \sigma yi^2)}}\,\sqrt{\pi}\,\sigma\,\sqrt{2 - \frac{2\,\sigma^2}{\sigma^2 + a^2\,\sigma xi^2 + \sigma yi^2}}\right]$$

Out[∘]=
$$\frac{e^{-\frac{(b + a\,xi - yi)^2}{2\,(\sigma^2 + a^2\,\sigma xi^2 + \sigma yi^2)}}\,\sqrt{1 - \frac{\sigma^2}{\sigma^2 + a^2\,\sigma xi^2 + \sigma yi^2}}}{\sqrt{2\pi}\,\sqrt{a^2\,\sigma xi^2 + \sigma yi^2}}$$

This equation can be further simplified to yield:

$$\frac{e^{-\frac{(b + a\,xi - yi)^2}{2\,(\sigma^2 + a^2\,\sigma xi^2 + \sigma yi^2)}}}{\sqrt{2\pi}\,\sqrt{\sigma^2 + a^2\,\sigma xi^2 + \sigma yi^2}}$$

In mathematical terms the likelihood of all n data pairs (x_i, y_i) and their respective standard deviations $(\sigma_{x,i}, \sigma_{y,i})$ after marginalization out of all of the hidden parameters (ξ_i, η_i) is

$$L(D \mid a, b, \sigma) = \prod_{i=1}^{n} \frac{e^{-\frac{(b+a x_i-y_i)^2}{2\left(\sigma^2+a^2 \sigma_{x,i}{}^2+\sigma_{y,i}{}^2\right)}}}{\sqrt{2\pi}\,\sqrt{\sigma^2+a^2\,\sigma_{x,i}{}^2+\sigma_{y,i}{}^2}}\,.$$

Here we have adopted the notation that D denotes all of the data. If we assume flat priors on the unknown parameters (a, b, σ) then the posterior distribution of the parameters is

$$p(a, b, \sigma \mid D) = \frac{1}{E}\;\frac{1}{a_{\max}-a_{\min}}\;\frac{1}{b_{\max}-b_{\min}}\;\frac{1}{\sigma_{\max}-\sigma_{\min}}\;L(D \mid a, b, \sigma)$$

where E is the evidence

$$E = \frac{1}{a_{\max}-a_{\min}}\;\frac{1}{b_{\max}-b_{\min}}\;\frac{1}{\sigma_{\max}-\sigma_{\min}}\int_{a_{\min}}^{a_{\max}}\int_{b_{\min}}^{b_{\max}}\int_{\sigma_{\min}}^{\sigma_{\max}} L(D \mid a, b, \sigma)\,da\,db\,d\sigma.$$

The evidence quantifies how well the model (linear regression in log space with intrinsic dispersion) fits the data. A somewhat more compact representation of the posterior that follows from the assumption of flat priors:

$$p(a, b, \sigma \mid D) = \frac{1}{\int_{a_{\min}}^{a_{\max}}\int_{b_{\min}}^{b_{\max}}\int_{\sigma_{\min}}^{\sigma_{\max}} L(D \mid a,b,\sigma)\,da\,db\,d\sigma}\;L(D \mid a, b, \sigma).$$

In addition to the evidence E, the quantities that we are interested in are the marginal distributions of the parameters (a, b, σ) given knowledge of the data D. They are

$$p(a \mid D) = \int_{b_{\min}}^{b_{\max}}\int_{\sigma_{\min}}^{\sigma_{\max}} p(a, b, \sigma \mid D)\,db\,d\sigma\,, \quad p(b \mid D) = \int_{a_{\min}}^{a_{\max}}\int_{\sigma_{\min}}^{\sigma_{\max}} p(a, b, \sigma \mid D)\,da\,d\sigma$$

and

$$p(\sigma \mid D) = \int_{a_{\min}}^{a_{\max}}\int_{b_{\min}}^{b_{\max}} p(a, b, \sigma \mid D)\,da\,db\,.$$

Computations

We begin by defining the log likelihood of the data. Computations are performed on a log scale to avoid underflow and overflow. The code is complied to increase execution speed in the computations that follow:

In[ ]:=
```
Clear[logLikelihood];
logLikelihood = Compile[{{θ, _Real, 1}},
   Module[{x, σx, y, σy, A, B, σ, logL},
    x = observedX; σx = observedErrorX;
    y = observedY; σy = observedErrorY;
    {A, B, σ} = θ;
    logL = -Length[x] Log[Sqrt[2.0 π]] + Sum[-0.5 (y[[i]] - A * x[[i]] - B)^2 / (σ^2 + A^2 σx[[i]]^2 + σy[[i]]^2)
         - Log[Sqrt[σ^2 + A^2 σx[[i]]^2 + σy[[i]]^2]], {i, 1, Length[x]}];
    logL
   ], CompilationOptions -> {"InlineExternalDefinitions" -> True},
   Parallelization -> True, CompilationTarget -> "WVM"];
```

In[ ]:=
```
logLikelihood[{1.0, 1.0, 1.0}]
```

Out[ ]=
```
-367.058
```

There are three unknown parameters namely (a, b, σ). This is a small enough total to allow brute force Bayesian computation. The code is:

In[ ]:=
```
Bayes3DFlatPriors[priorPDFList_, nXnYnZ_ : {121, 111, 61}] :=
 Module[{nx, ny, nz, priorSupport, logpriorfactor, xmin, xmax, ymin, ymax, zmin, zmax,
   dx, dy, dz, xgrid, ygrid, zgrid, loggrid3d, bayesOut, maxlog, logZ, posteriorPDF,
   marginalx, marginaly, marginalz, xmean, ymean, zmean, H, posteriorFlat},
  {nx, ny, nz} = nXnYnZ;
  priorSupport = Map[First, priorPDFList];
  {{xmin, xmax}, {ymin, ymax}, {zmin, zmax}} = priorSupport;
  logpriorfactor = Log[1/(xmax - xmin) 1/(ymax - ymin) 1/(zmax - zmin)];
  dx = (xmax - xmin)/(nx - 1); dy = (ymax - ymin)/(ny - 1); dz = (zmax - zmin)/(nz - 1);
  xgrid = Table[x, {x, xmin, xmax, dx}];
  ygrid = Table[y, {y, ymin, ymax, dy}];
  zgrid = Table[z, {z, zmin, zmax, dz}];
  loggrid3d = Table[logLikelihood[{x, y, z}],
    {x, xmin, xmax, dx}, {y, ymin, ymax, dy}, {z, zmin, zmax, dz}];
  bayesOut["unNormalizedLogLikelihood"] = loggrid3d;
  (* Compute log evidence. *)
  loggrid3d =
   ArrayReshape[Flatten[loggrid3d] + logpriorfactor, Dimensions[loggrid3d]];
  maxlog = Max[Flatten[loggrid3d]];
  loggrid3d = ArrayReshape[Flatten[loggrid3d] - maxlog, Dimensions[loggrid3d]];
```

```
 logZ = maxlog + Log[dx * dy * dz * Total[Exp[Flatten[loggrid3d]]]];
 (* Compute normalized posterior. *)
 posteriorPDF = Exp[loggrid3d];
 posteriorPDF = posteriorPDF / Total[Flatten[posteriorPDF]];
 (* Compute marginal probability distributions. *)
 marginalx = Chop@Map[Total, Map[Flatten, posteriorPDF]];
 marginaly = Chop@Map[Total, Map[Flatten, Transpose[posteriorPDF]]];
 marginalz = Chop@Map[Total, Map[Flatten, Transpose[posteriorPDF, {3, 2, 1}]]];
 xmean = xgrid.marginalx;
 ymean = ygrid.marginaly;
 zmean = zgrid.marginalz;
 marginalx = Transpose[{xgrid, dx^-1 marginalx}];
 marginaly = Transpose[{ygrid, dy^-1 marginaly}];
 marginalz = Transpose[{zgrid, dz^-1 marginalz}];
 (* Compute information H. *)
 posteriorFlat = Flatten[posteriorPDF];
 posteriorFlat = Select[posteriorFlat, # > 0.0 &];
 H = Total@(dx * dy * dz (1/(dx * dy * dz) posteriorFlat) Log[(1/(dx*dy*dz) posteriorFlat)/(1/(xmax-xmin) 1/(ymax-ymin) 1/(zmax-zmin))]);
 bayesOut["Properties"] = {"posteriorPDF", "logZ", "InformationH", "marginalx",
   "marginaly", "marginalz", "xgrid", "ygrid", "zgrid", "xmean", "ymean", "zmean"};
 bayesOut["posteriorPDF"] = posteriorPDF;
 bayesOut["logZ"] = logZ; bayesOut["InformationH"] = H;
 bayesOut["marginalx"] = marginalx;
 bayesOut["xmean"] = xmean;
 bayesOut["xgrid"] = xgrid;
 bayesOut["marginaly"] = marginaly;
 bayesOut["ymean"] = ymean;
 bayesOut["ygrid"] = ygrid;
 bayesOut["marginalz"] = marginalz;
 bayesOut["zmean"] = zmean;
 bayesOut["zgrid"] = zgrid;
 bayesOut]
```

Actual computations occur here :

```
In[∘]:= nXnYnZ = {121, 111, 61};
priorPDFList = {UniformDistribution[{1.0, 7.0}],
   UniformDistribution[{-4.0, 3.0}], UniformDistribution[{0.1, 0.7}]};
bayesOut = Bayes3DFlatPriors[priorPDFList];
```

The available outputs are:

In[∘]:=
```
bayesOut["Properties"]
```
Out[∘]=
```
{posteriorPDF, logZ, InformationH, marginalx,
 marginaly, marginalz, xgrid, ygrid, zgrid, xmean, ymean, zmean}
```

The log evidence is :

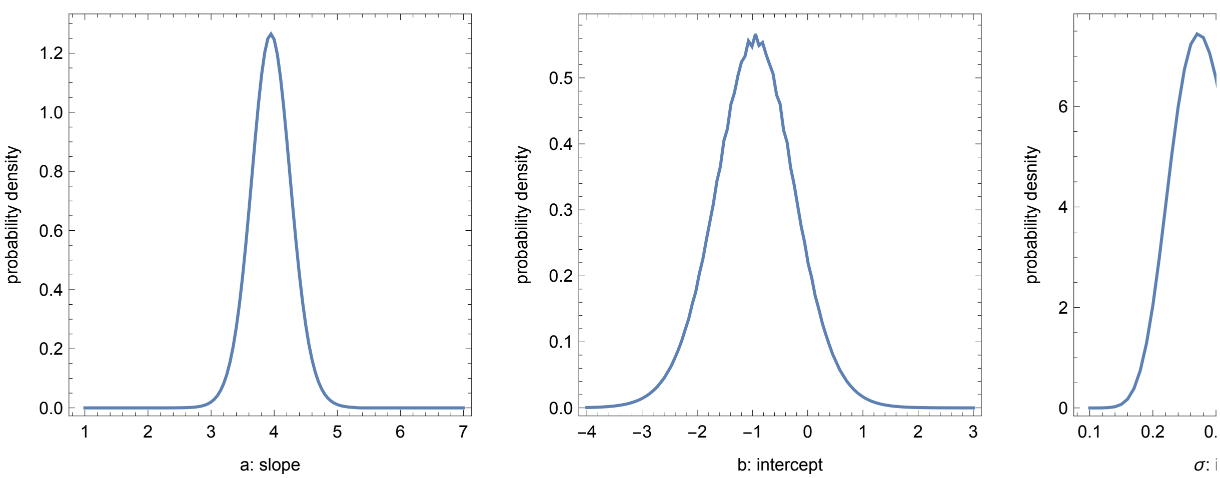
In[∘]:=
```
bayesOut["logZ"]
```
Out[∘]=
```
-17.1509
```

The information is:

In[∘]:=
```
bayesOut["InformationH"]
```
Out[∘]=
```
5.79967
```

The marginal pdfs are:

In[∘]:=
```
marginalx = bayesOut["marginalx"];
marginaly = bayesOut["marginaly"];
marginalz = bayesOut["marginalz"];
priorSupport = Map[First, priorPDFList];
gx = ListLinePlot[marginalx, ⋯];
gy = ListLinePlot[marginaly, ⋯];
gz = ListLinePlot[marginalz, ⋯];
GraphicsRow[{gx, gy, gz}, ImageSize → 800]
```
Out[∘]=



Find mean slope, intercept and intrinsic dispersion:

In[ ]:=
```
xgrid = bayesOut["xgrid"];
ygrid = bayesOut["ygrid"];
zgrid = bayesOut["zgrid"];
aMean = bayesOut["xmean"];
bMean = bayesOut["ymean"];
σMean = bayesOut["zmean"];
{aMean, bMean, σMean}
```

Out[ ]=
```
{3.95241, -0.956994, 0.288501}
```

Plot regression and error bounds on data:

In[ ]:=
```
fitline = Transpose[{xgrid, Map[(aMean * # + bMean) &, xgrid]}];
fitlineUp = Transpose[{xgrid, Map[(aMean * # + bMean + σMean) &, xgrid]}];
fitlineDown = Transpose[{xgrid, Map[(aMean * # + bMean - σMean) &, xgrid]}];
pnts = Transpose[{observedX, observedY}];
pntsLeftX = Transpose[{observedX - observedErrorX, observedY}];
pntsRightX = Transpose[{observedX + observedErrorX, observedY}];
pntsUpY = Transpose[{observedX, observedY + observedErrorY}];
pntsDownY = Transpose[{observedX, observedY - observedErrorY}];
errorLines =
  Join[Transpose[{pntsLeftX, pntsRightX}], Transpose[{pntsDownY, pntsUpY}] ];
errorLines = {Black, Thin, Map[Line, errorLines]};
gMain0 = ListPlot[Transpose[{observedX, observedY}] , Prolog → {errorLines} , ... ]
```

Out[ ]=

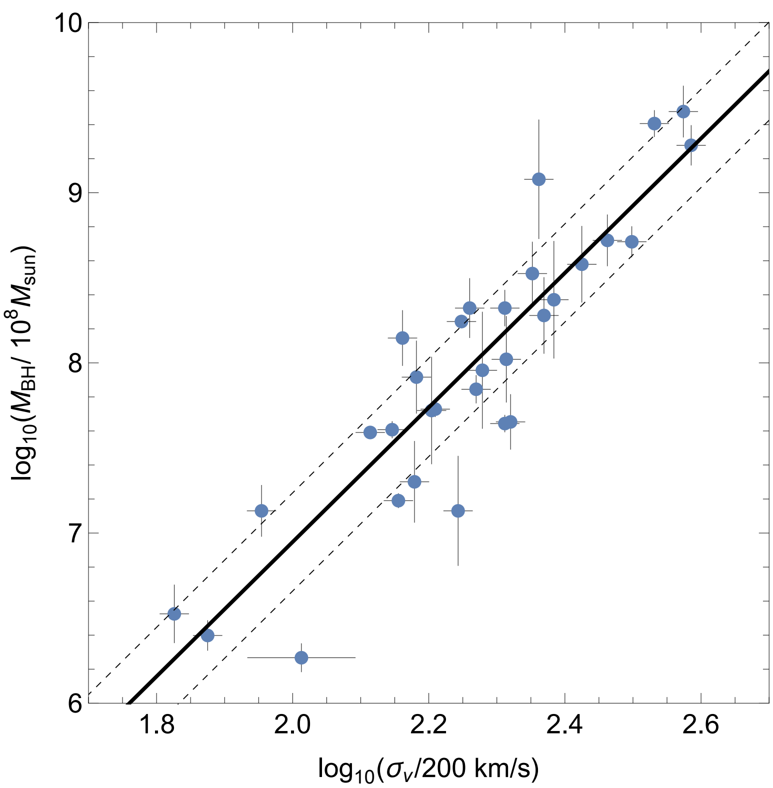


Figure 3. Magorrian relation. The solid line marks the mean relation. The dashed lines show the mean plus or minus the intrinsic scatter. Error bars on the data points represent errors in both variables. The above figure agrees with the result in *Bayesian Methods for the Physical Sciences: Learning from Examples in Astronomy and Physics* by Stefano Andreon and Brain Weaver, page 135.

Notes and References

The Magorrian relation data presented here were taken from table 8.3 of Andreon and Weaver (2015). They in turn took the data from Tremaine (2002).

Andreon, Stefano and Weaver, Brian (2015), *Bayesian Methods for the Physical Sciences : Learning from Examples in Astronomy,* Springer.

Merritt, David (1999). “Black holes and galaxy evolution”. In Combes, F.; Mamon, G. A.; Charmandaris, V. (eds.). Dynamics of Galaxies: from the Early Universe to the Present. Vol. 197. See also the Wikipedia article: " https://en.wikipedia.org/wiki/M–sigma_relation"

Tremaine, S., Gebhardt, K., Bender, R., Bower, G., Dressler, A., Faber, S. and et al (2002), "The slope of the black hole mass velocity dispersion correlation", The Astrophysical Journal, 574: 740-753.