EP 4130/PH6130

Second Week

Chi-Square Distribution

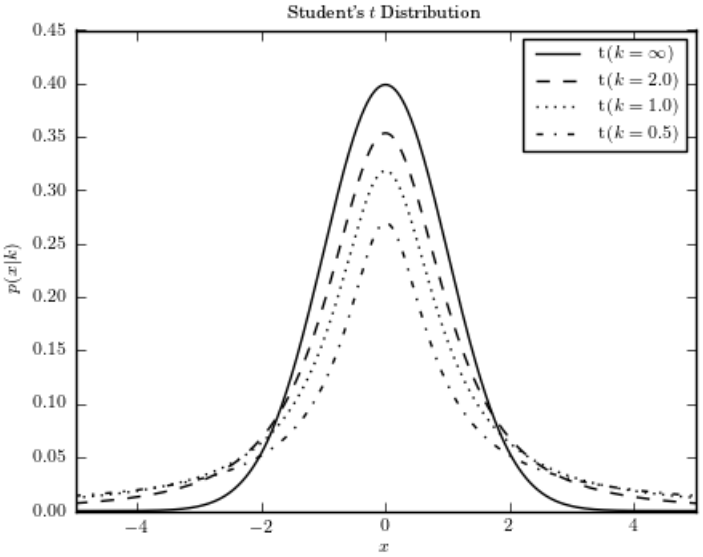
$$p(Q|k) \equiv \chi^2(Q|k) = \frac{1}{2^{k/2}\Gamma(k/2)} Q^{k/2-1} \exp(-Q/2)$$

Sometimes $\chi 2$ distribution per degree of freedom is defined and is given by :

$$\chi^2_{dof} \equiv \chi^2(Q/k|k)$$

Mean value of $\chi 2_{dof}$ is equal to 1. As k increases $\chi 2_{dof}$ tends to $\mathcal{N}(1, \sqrt{2/k})$

Student's t distribution



$$p(x|k) = \frac{\Gamma(k+1/2)}{\sqrt{\pi k}\Gamma(k/2)}\left(1+\frac{x^2}{k}\right)^{-\frac{k+1}{2}}$$

```
from scipy import stats
dist=stats.t(5) #k=5
r= dist.rvs(10) # 10 random draws
P = dist.pdf(4) #pdf evaluated at x=4
```

- For k=1, this distribution is a Cauchy distribution with μ=0 and γ=1
- Mean, Mode and Median=0 for k>1 and undefined for k=1

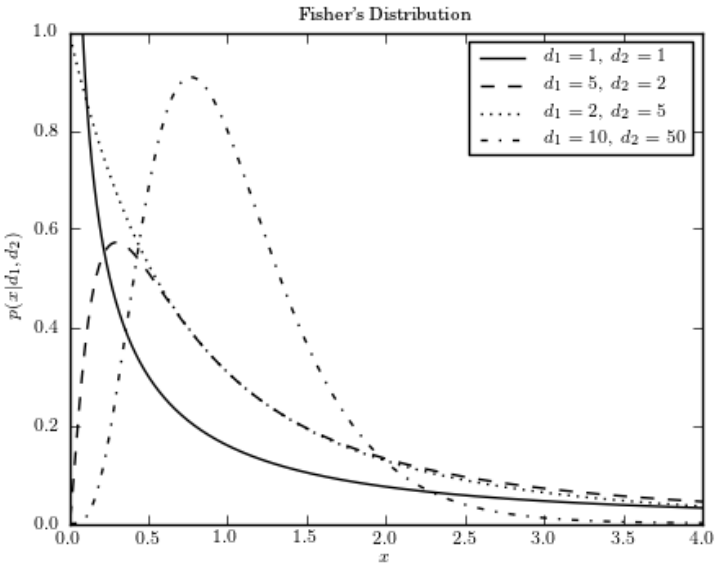
Given a sample of N measurements $\{x_i\}$ drawn from a Gaussian distribution $\mathcal{N}(\mu, \sigma)$

Define $$t = \frac{\bar{x} - \mu}{s/\sqrt{N}}$$

where $\bar{x}$ and s are the sample mean and sample variance, follows Student's t- distribution with k=N-1 degrees of freedom

- Student's t distribution based on **data** based estimates of mean and std. deviation, whereas Cauchy distribution based on true mean and standard deviation.
- t-distribution stays the same for samples drawn from a Gaussian distribution with different values of mean and standard deviation. as long as number of samples stays the same.
- Ratio of a standard normal variable and one drawn from a $\chi 2$ distribution follows Student's t-distribution.
- t-distribution is used when comparing means of two samples

Fisher's F distribution



Also known as Snedecor distribution, variance ratio Distribution, F-distribution

```
from scipy import stats
dist=stats.f(2,3) #d1=2,d2=3
r= dist.rvs(10) # 10 random draws
P = dist.pdf(1) #pdf evaluated at x=1
```

$$p(x|d_1, d_2) = C(1 + \frac{d_1}{d_2}x)^{-\frac{d_1+d_2}{2}} x^{\frac{d_1}{2}-1}$$

where C is given by

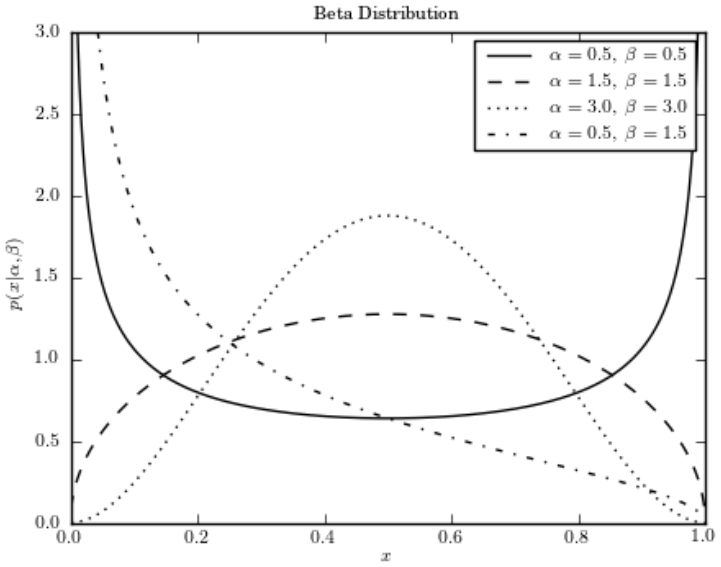
$$C = \frac{1}{B(d_1/2, d_2/2)} \left(\frac{d_1}{d_2}\right)^{d_1/2}$$

B is the Beta function

- Fisher F distribution describes the ratio of two independent $\chi 2$ variables with d_1 and d_2 degrees of freedom.

- If x_1 and x_2 are two independent random variables drawn from the Cauchy distribution with Location parameter μ, then the ratio $|x_1-\mu|/|x_2-\mu|$ follows Fisher's F distribution with $d_1=d_2=2$.

Beta Distribution

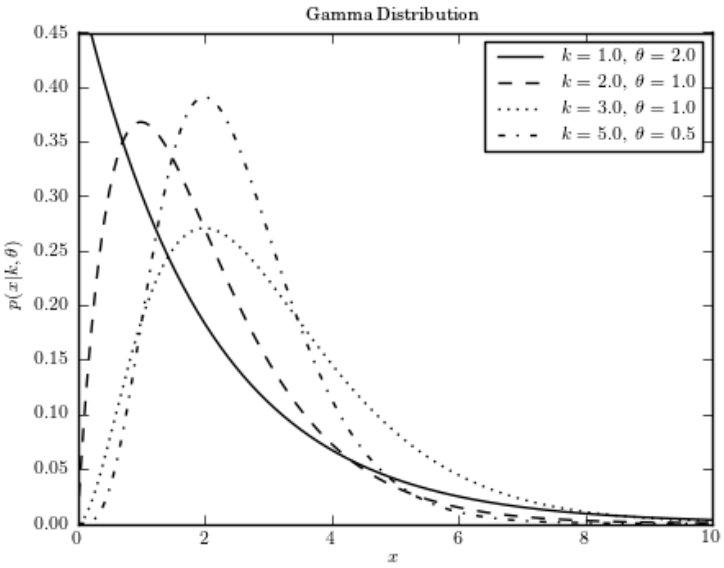



```
from scipy import stats
dist=stats.beta(0.5,1.5) #alpha=0.5, beta=1.5
r= dist.rvs(10) # 10 random draws
P = dist.pdf(0.6) #pdf evaluated at x=0.6
```

$$p(x|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}$$

- Mean value of beta distribution is α/(α+β)
- Beta distribution is conjugate prior of the binomial distribution

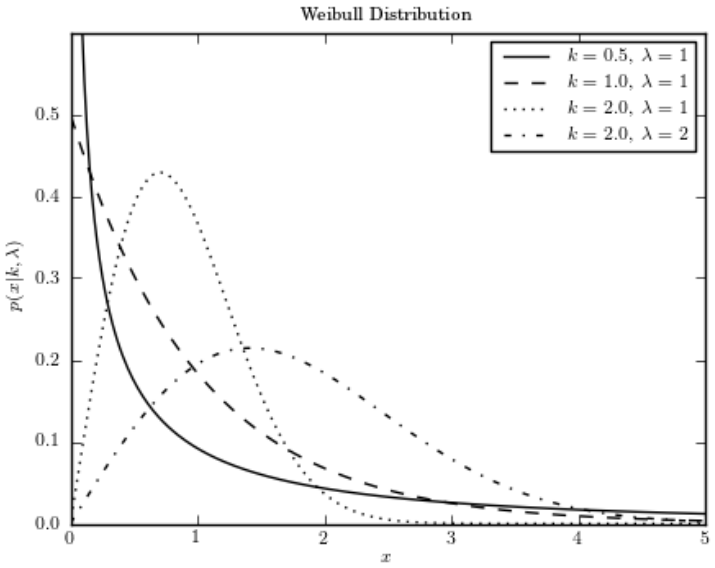
Gamma Distribution




```
from scipy import stats
dist=stats.gamma(1,0,2) #k=1, loc=0, theta=2
r= dist.rvs(10) # 10 random draws
P = dist.pdf(1) #pdf evaluated at x=1
```

- Gamma Function is a conjugate prior to several distributions such as the Laplace distribution and the Poisson distribution

Weibull Distribution




```
from scipy import stats
dist=stats.dweibull(1,0,2) #k=1, loc=0,lambda=2
r= dist.rvs(10) # 10 random draws
P = dist.pdf(1) #pdf evaluated at x=1
```

$$p(x|k,\lambda) = \frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1} e^{-(x/\lambda)^k}$$

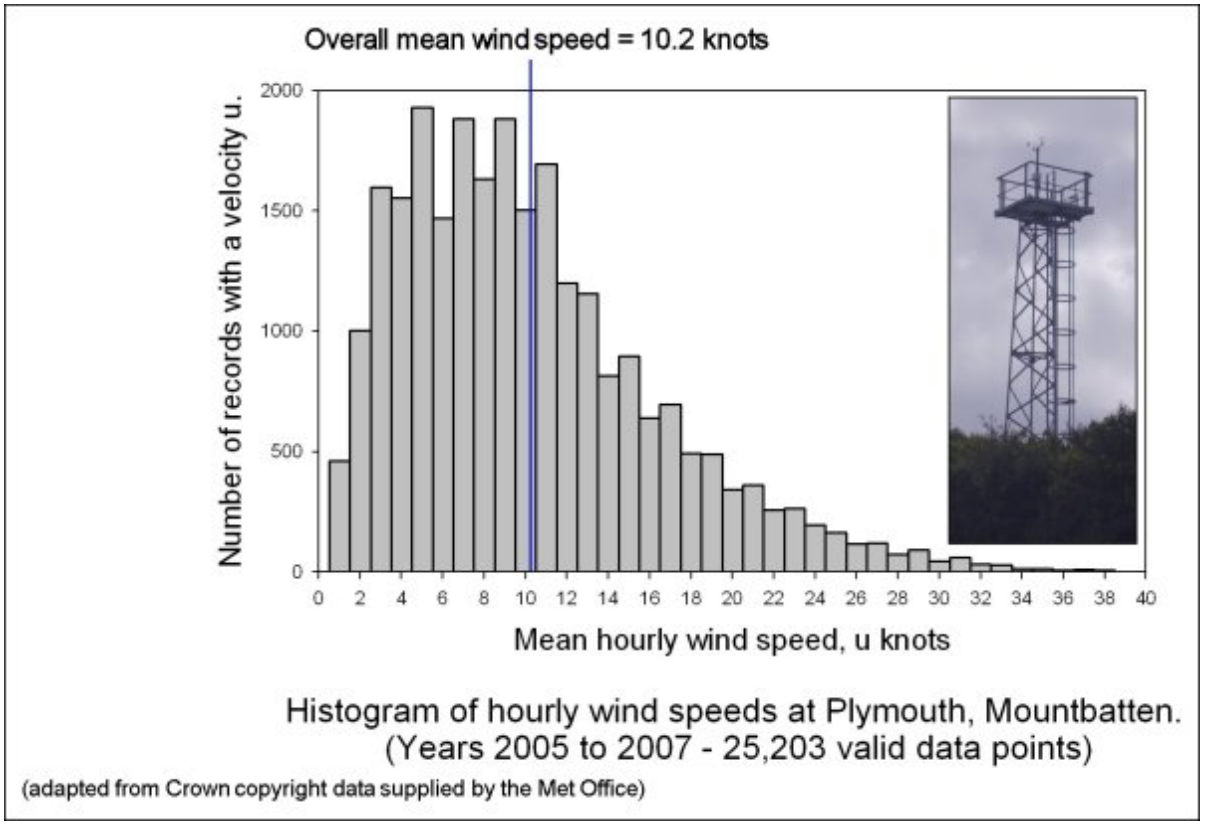
Weibull distribution describes good description of random failure process with variable rate, wind speeds, distribution of extreme values, and size distribution of particles., overvoltage in an Electrical system.

eg:

- If x is the time to failure for a device with a failure rate proportional to time t^m , x follows the Weibull distribution with k=m+1

- Wind speed follows Weibull distribution with k ~2

Example of Weibull Distribution



Histogram of hourly wind speeds at Plymouth, Mountbatten.
(Years 2005 to 2007 - 25,203 valid data points)

(adapted from Crown copyright data supplied by the Met Office)

1 knot = 0.52 m/sec

Typo in astroML book

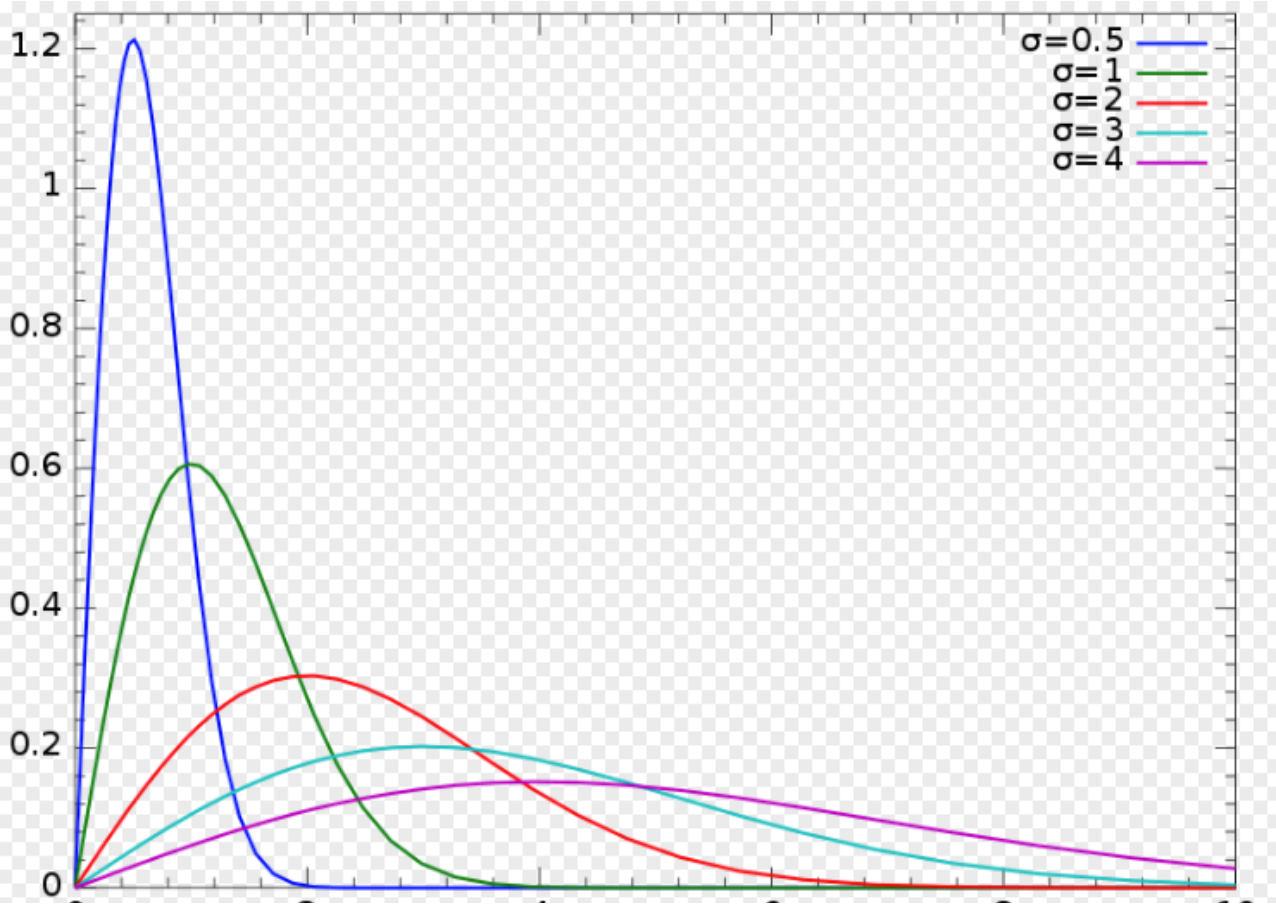
- Typo in Weibull distribution pdf in AstroML book. What is plotted is positive part of double Weibull distribution `(dweibull)`. PDF of Weibull distribution (defined in class) is given by twice this value. Alternatively, use stats.`weibull_min`

Rayleigh Distribution (not discussed in astroML)

$$f(x,\sigma) = \frac{x}{\sigma^2} e^{-x^2/2\sigma^2} \quad x \geq 0$$

Cumulative distribution function is given by $$F(x,\sigma) = 1 - e^{-x^2/2\sigma^2}$$

Equal to Chi-square distribution for two degrees of freedom
Observed when overall magnitude of a vector is related to its directional Components. eg. Wind velocity in two dimensions



Examples

```
>>> from scipy.stats import rayleigh
>>> import matplotlib.pyplot as plt
>>> fig, ax = plt.subplots(1, 1)
```

Example of Rayleigh Distribution in astrophysics

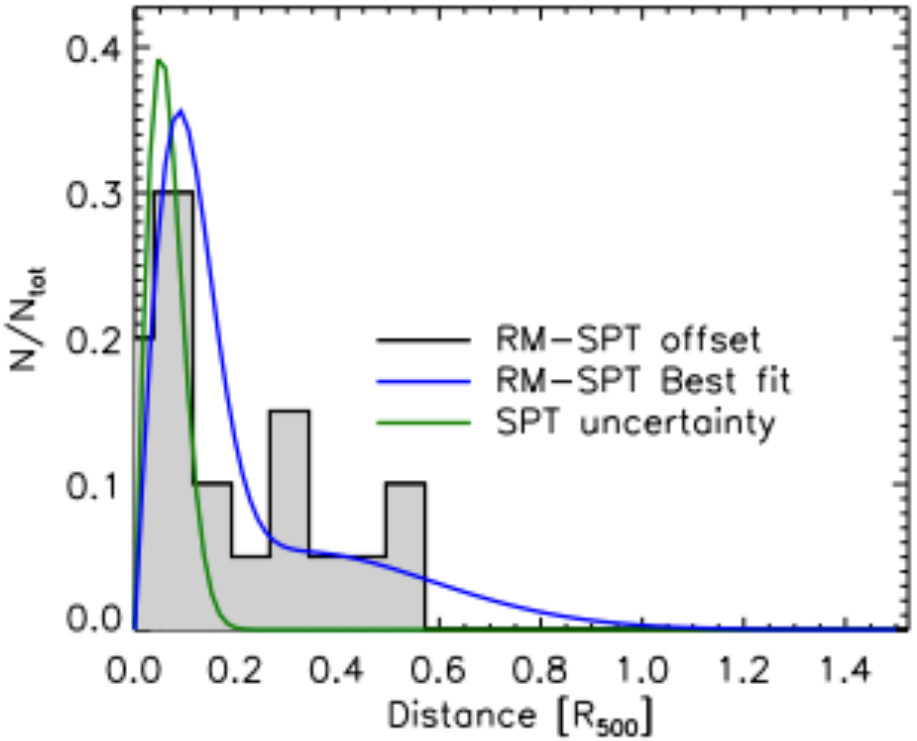


Figure 7. Solid histogram shows the measured fraction of SPT-SZ+RM clusters as a function of the optical-SZE positional offset in units of R_{500}. The green curve shows the SPT-SZ positional uncertainty, and the blue curves shows the best fitting SZE-optical positional offset model.

$$P(x) = 2\pi x\left(\frac{\rho_0}{2\pi\sigma_0^2}e^{-\frac{x^2}{2\sigma_0^2}} + \frac{1-\rho_0}{2\pi\sigma_1^2}e^{-\frac{x^2}{2\sigma_1^2}}\right) \qquad (12)$$

Mixture of two Rayleigh distributions

arXiv:1506.07814

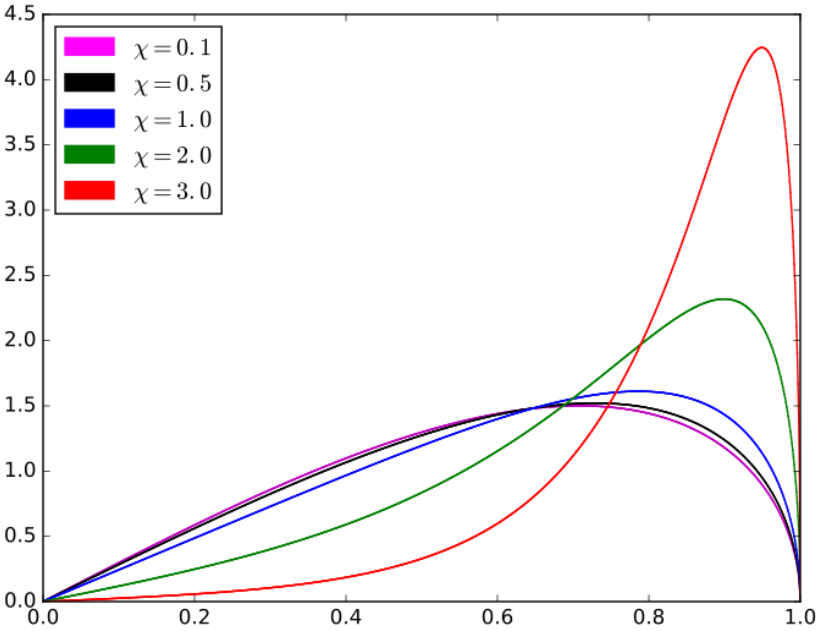
Other distributions in scipy can be found at

https://docs.scipy.org/doc/scipy-0.18.1/reference/stats.html

Other potential distributions among these : Rice, Pareto, Power-Law,

Argus Distribution

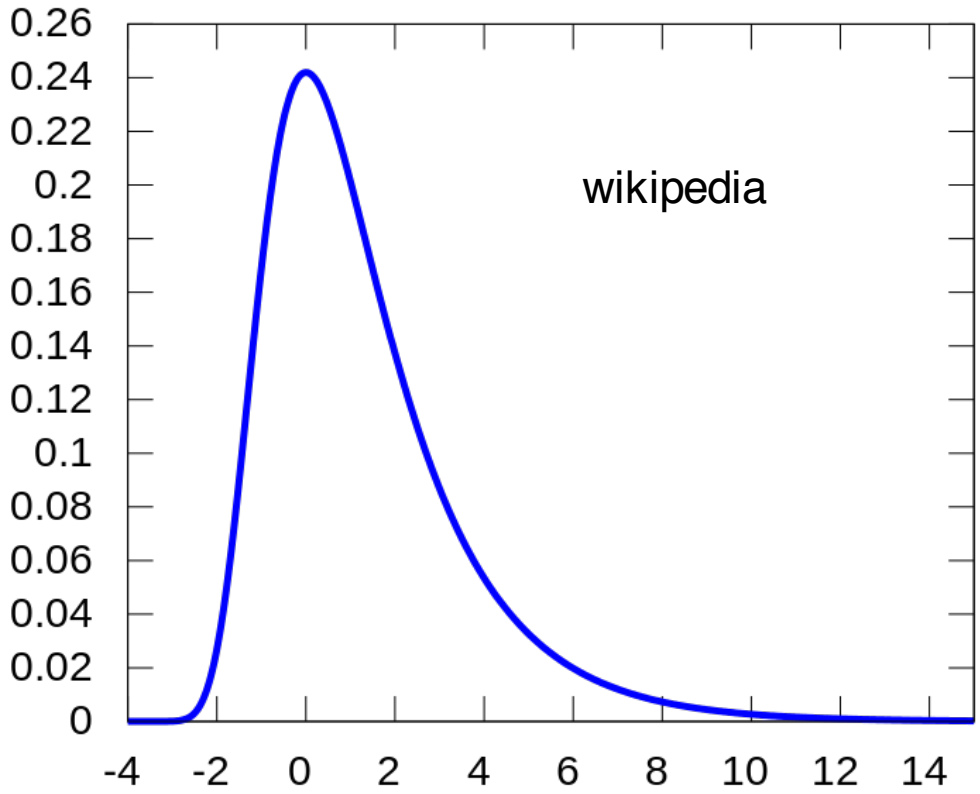
Probability distribution named after the particle physics experiment ARGUS is the reconstructed invariant mass of a decayed particle candidate in a continuum background.



Look up scipy.stats.argus

wikipedia

Landau Distribution



Not in stats.scipy

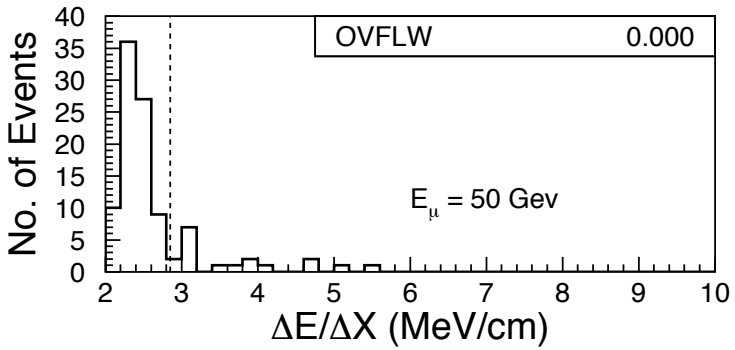
No analytic formulae for pdf

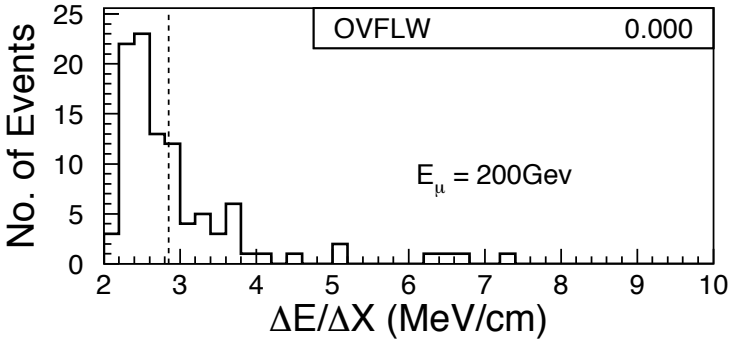
Mean and variance are undefined because of very long tail

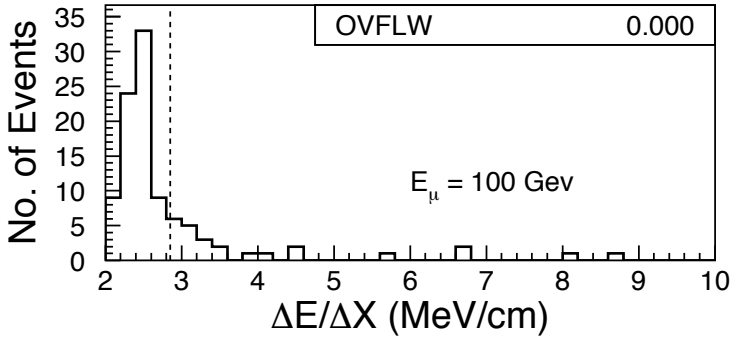
Python function (pyLandau) can be downloaded from the web

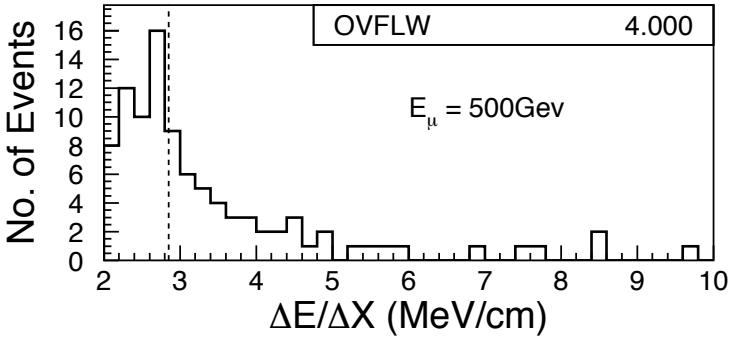
Scipy.stats.moyal provides a good approximation to Landau distribution.

Example of Landau Distribution







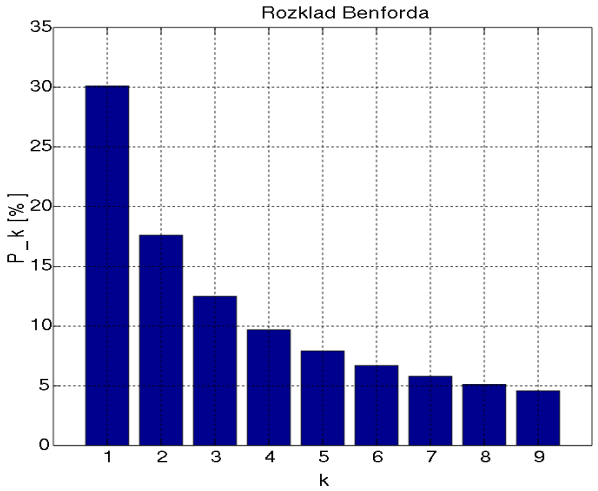


Simulations of Muon Energy loss in water for various mono-energetic muon samples (SD thesis)

Benford Distribution

- Benford's law is a law about frequency of leading digits
- A set of numbers is said to satisfy Benford's law if the leading digit occurs with probability given by

$$P(d) = \log_{10}(d+1) - \log_{10}(d) = \log_{10}\left(\frac{d+1}{d}\right) = \log_{10}\left(1+\frac{1}{d}\right)$$

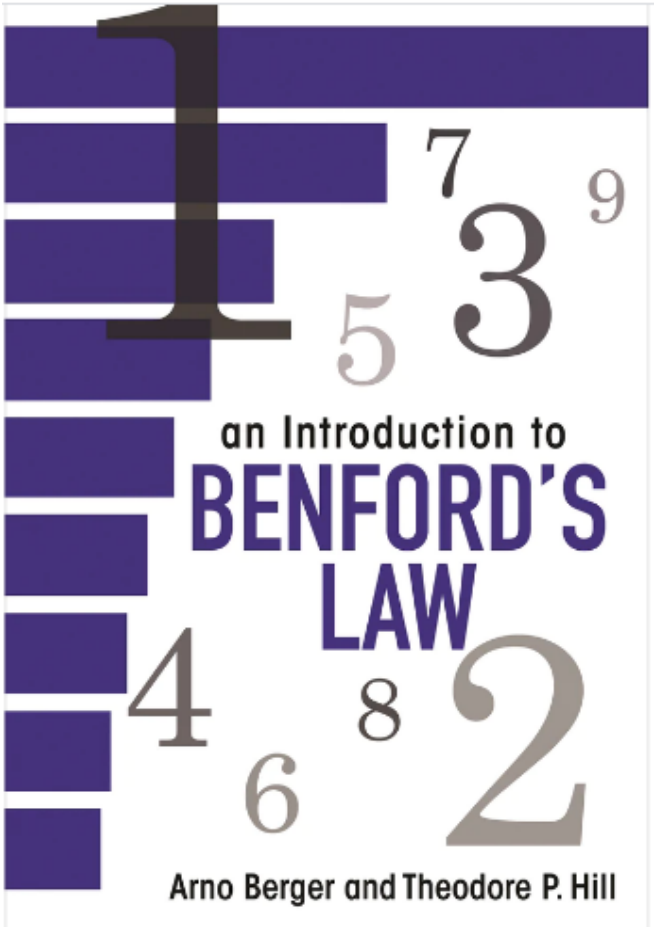


Examples of Benford Law

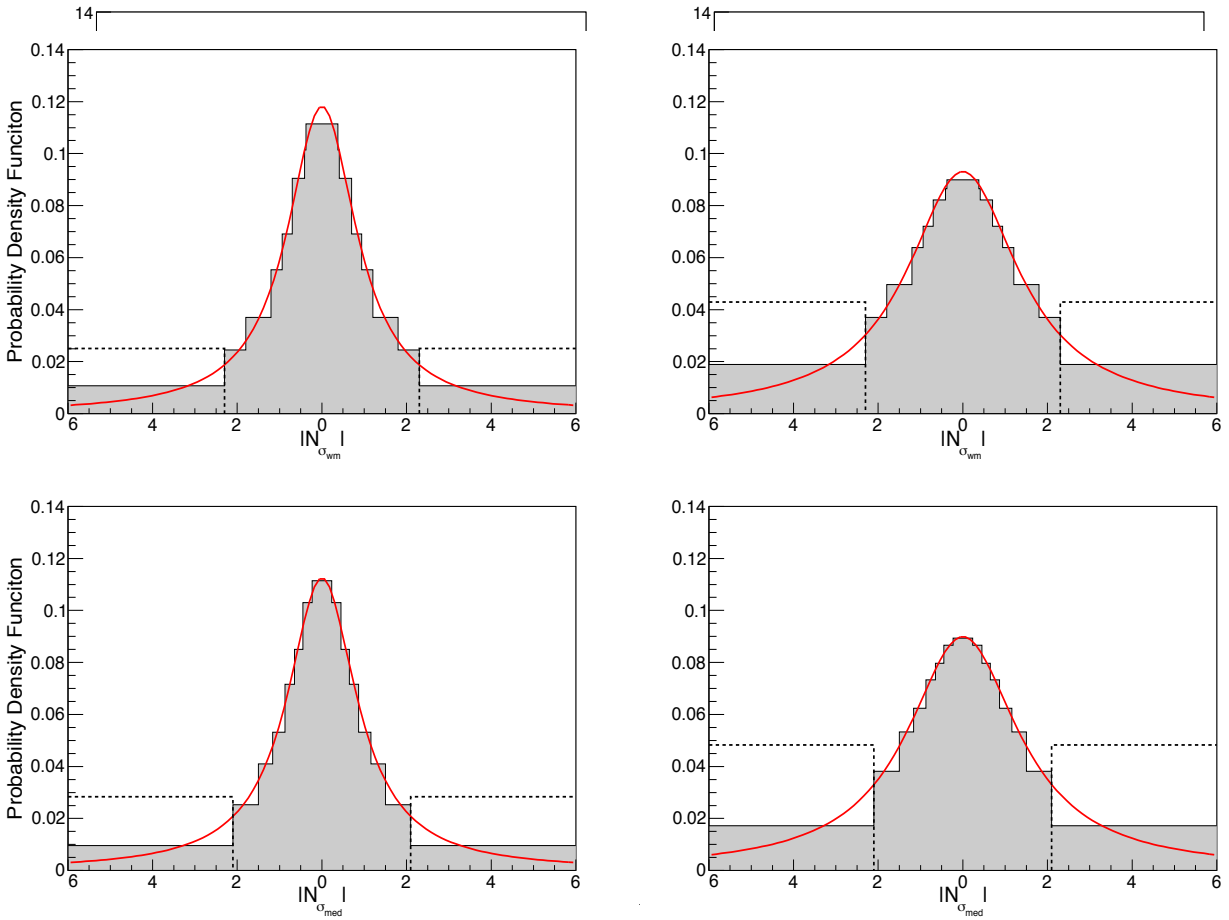
- Surface Areas of Rivers
- Physical Constants
- Molecular Weights
- Street Addresses of people listed in American Men of Science
- Heights of Tallest structures in world
- Telephone Bills etc (see wikipedia)

Has been Used to detect bank & election fraud and election rigging
For more details, please read
arxiv:2008.12271 and
arXiv:1709.09823 by A. Dantuluri & SD
arXiv:2207.09696 by P. Mamidipaka & SD
Also a whole book on Benford's law

HW : do google search on COVID-19 & Benford's law



Primordial Lithium-7 Measurements



tom) row uses the weighted mean (median) of the 66 measurements as the central estimate.

Fig. 3. The left (right) column shows the probability density (absolution) function. The top left (right) plot represents positive (negative) N_σ represent a value that is greater (less) than the central estimate.

Crandall , Houston and Ratra arXiv:1409.7332

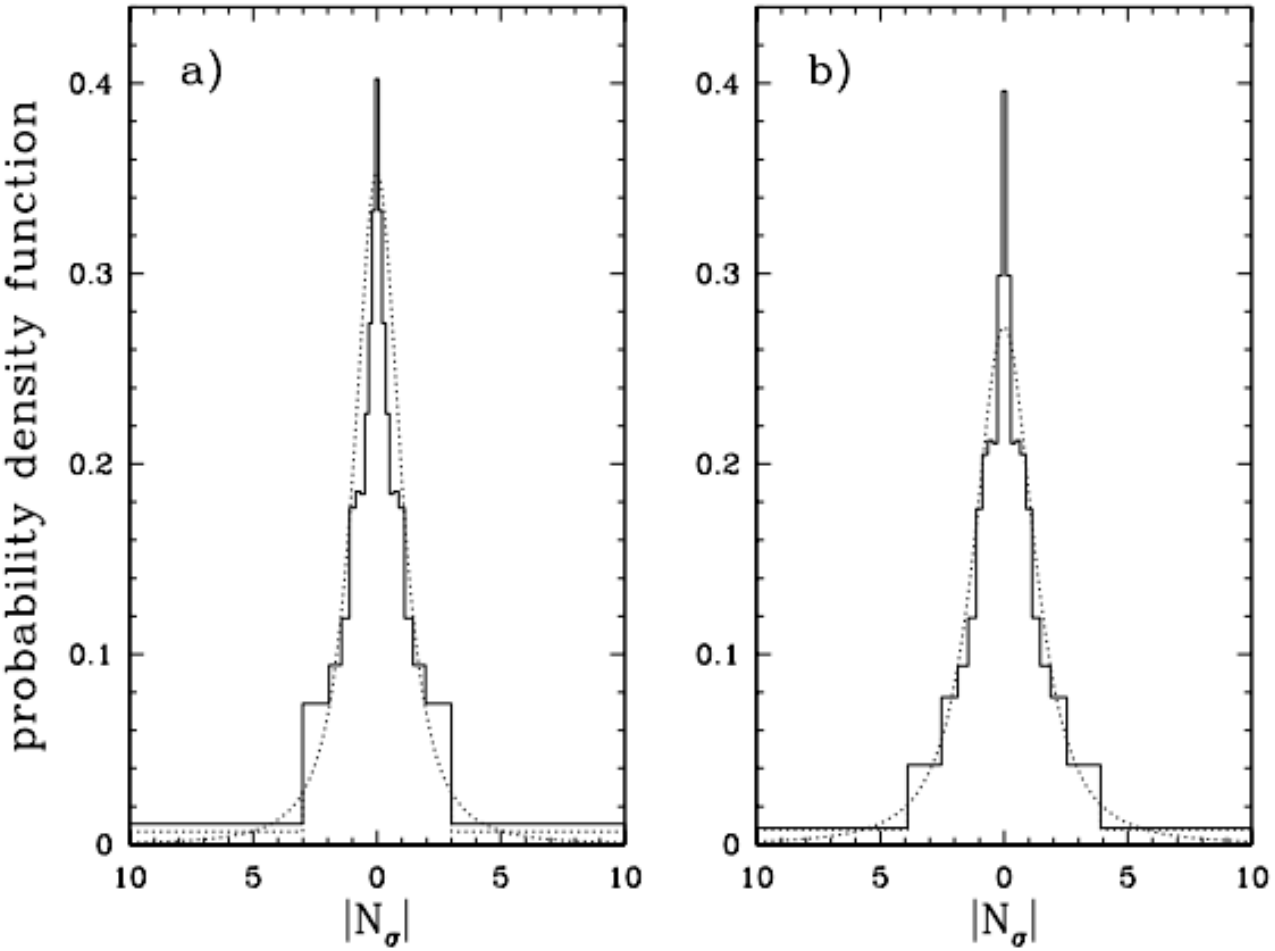


Fig. 4.— Binned data (solid lines) and best-fit $n = 2$ Student's t probability distribution functions (dotted lines) for $H_0 = 71$ km s^{-1} Mpc^{-1} estimated by the WMAP collaboration, all normalized to unit area. See Fig. 2 caption for more details. Left panel a) shows a distribution with scale

Error Distributions for Hubble Constant measurements
Fit to students-t distribution

astro-ph/0308099

Central Limit Theorem

- Given an arbitrary distribution h(x) characterized by its mean μ and standard deviation **σ**, mean of N values x_i drawn from that distribution will approximately follow a Gaussian distribution $\mathcal{N}(\mu, \sigma/\sqrt{N})$ with the accuracy increasing with increasing N.

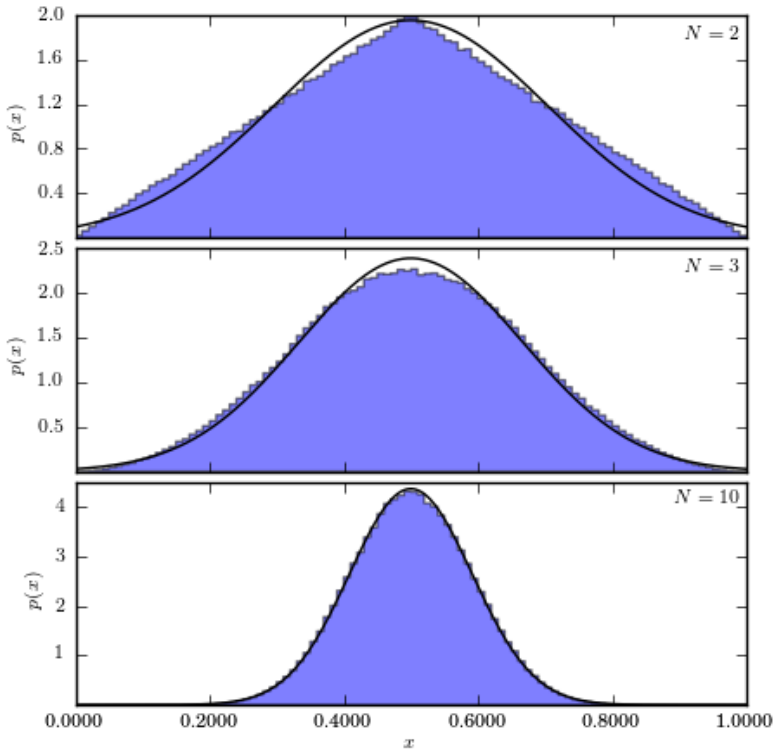
- h(x) must have a standard deviation and therefore its tails must fall off faster than $1/x^2$ for large x

- Central limit theorem can can be derived using method of characteristic functions (arXiv:0712.3028) or repeated convolutions (Gregory 2005).

- Central limit theorem does not apply to Cauchy distributions, as it does not have a well-defined mean and standard deviation.

- Sample Mean converges to distribution mean as sample size increases. This is called weak law of Large Numbers.

Illustration of Central Limit Theorem



Histogram of N random variables (from uniform distribution) drawn from 0 to 1

Bivariate Gaussian Distributions

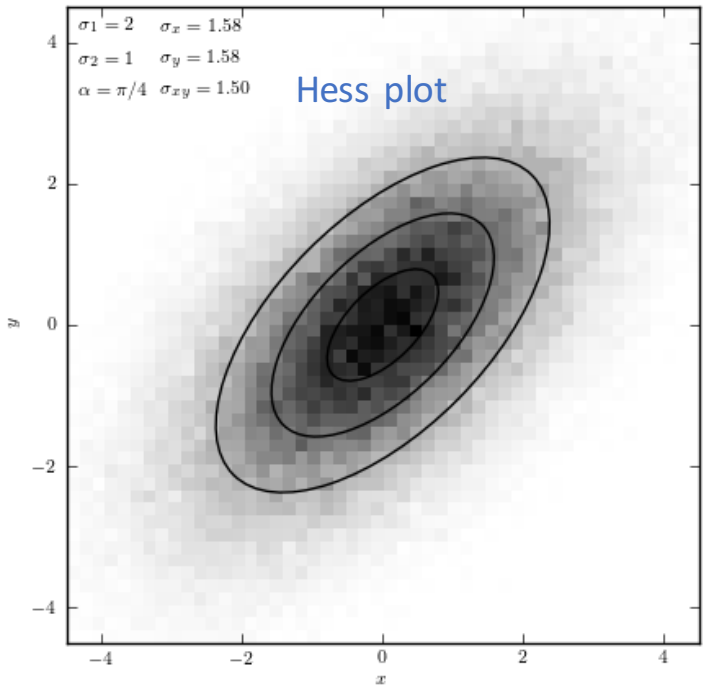
$$p(x, y|\mu_x, \mu_y, \sigma_x, \sigma_y, \sigma_{xy}) = \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} \exp\left(-\frac{z^2}{2(1-\rho^2)}\right)$$

where $$z^2 = \frac{(x-\mu_x)^2}{\sigma_x^2} + \frac{(y-\mu_y)^2}{\sigma_y^2} - 2\rho\frac{(x-\mu_x)(y-\mu_y)}{\sigma_x\sigma_y}$$

$$\rho = \frac{\sigma_{xy}}{\sigma_x\sigma_y}$$

ρ is called population correlation coefficient. For correlated variables **ρ** ~ 1. For uncorrelated variables **ρ** = 0

Data from Bivariate Gaussian Distributions



Ellipses centered on (μ_x , μ_y)
Angle between X-axis and ellipse major axis is given by :

$$\tan(2\alpha) = 2\rho\frac{\sigma_x\sigma_y}{\sigma_x^2 - \sigma_y^2} = 2\frac{\sigma_{xy}}{\sigma_x^2 - \sigma_y^2}$$

Correlation between x and y can be eliminated by Principal axis transformation. Correlation between two variables disappears once we rotate the vectors to P1 and P2

$$P_1 = (x - \mu_x)\cos\alpha + (y - \mu_y)\sin\alpha$$

$$P_2 = -(x - \mu_x)\sin\alpha + (y - \mu_y)\cos\alpha$$

Multivariate Gaussian Distributions

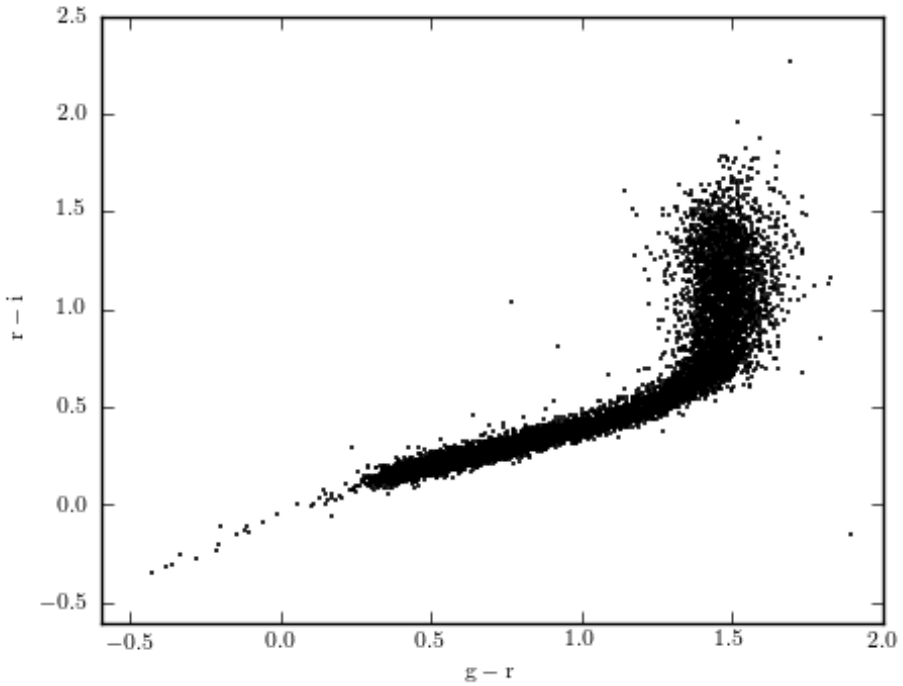
```
import numpy as np
mu=[1,2]
Cov = [[1,0.2],[0.2,0.3]]
np.random.multivariate_normal(mu,cov)
```

$$p(x|I) = \frac{1}{(2\pi)^{M/2}\sqrt{\det(C)}} \exp(-\frac{1}{2}x^T H x)$$

where x is vector. C is covariance matrix. H is inverse of covariance matrix also called Hessian

$$C_{kj} = \int_{-\infty}^{+\infty} x^k x^j p(x|I) d^M x \qquad x^T H x = \sum_{k=1}^{M}\sum_{j=1}^{M} H_{kj} x^k x^j$$

Data visualization techniques



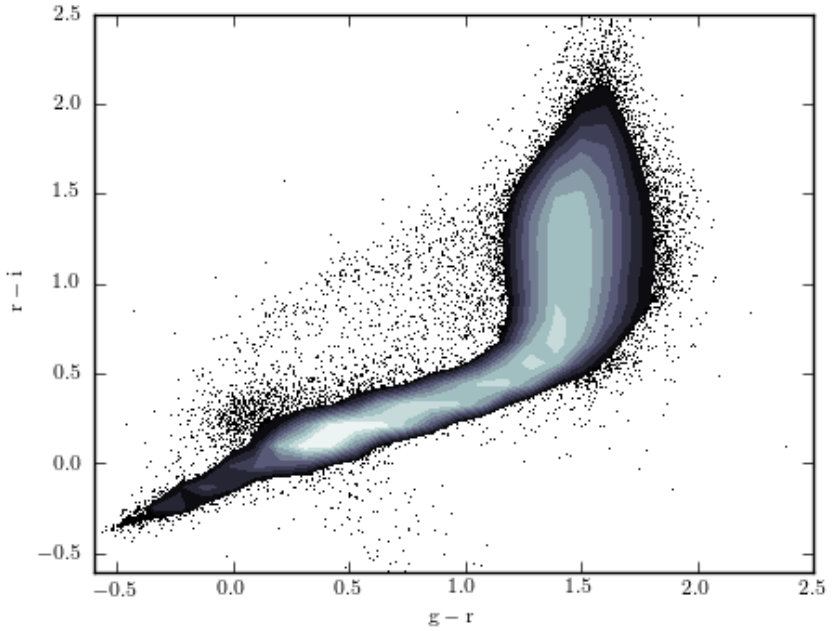
Simple Scatter plots

For more details, see *The Visual Display of Quantitative Information* by Tufte

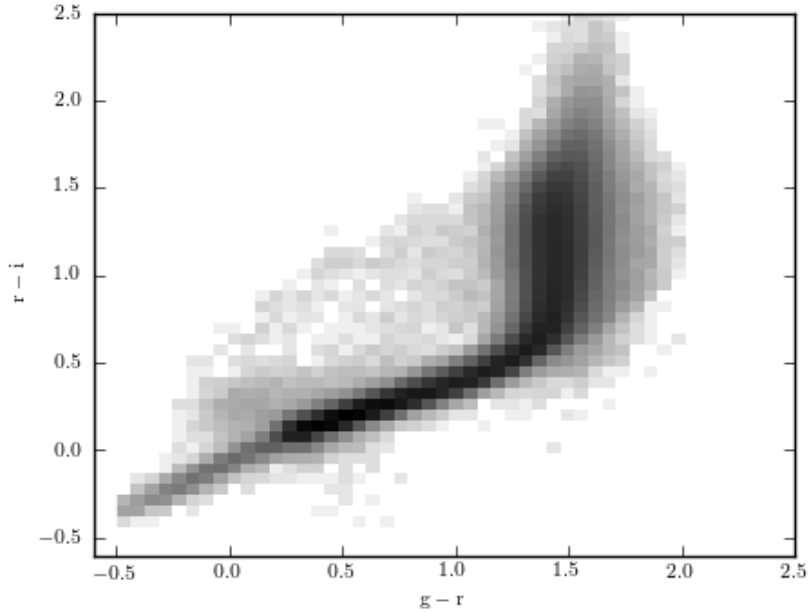
Data visualization techniques

If density of two points too high, no longer practical to use ordinary ‘scatter plots’. Need to switch to Contour plots. However this loses information in case of only a few points.

⟹ Use contours for high density region and individual points in low density region (used extensively in SDSS)

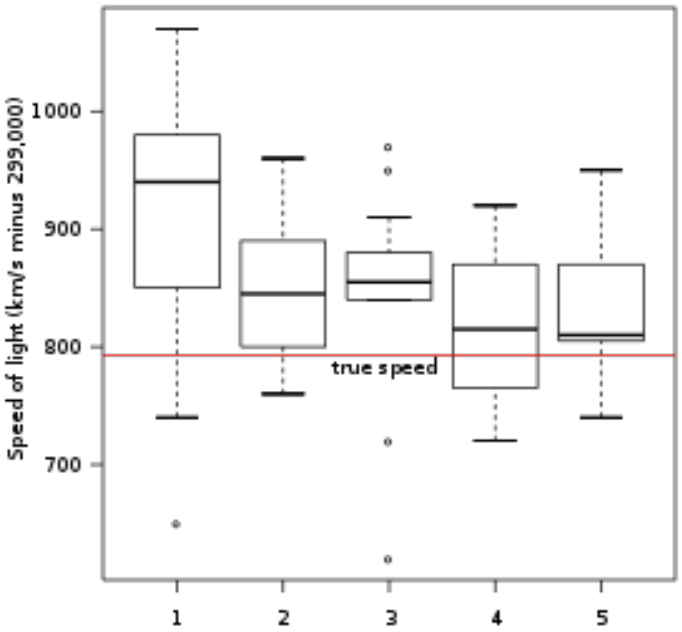


Alternatively pixelize the plotted diagram and display the counts of points in each pixel.
Known as Hess-diagram

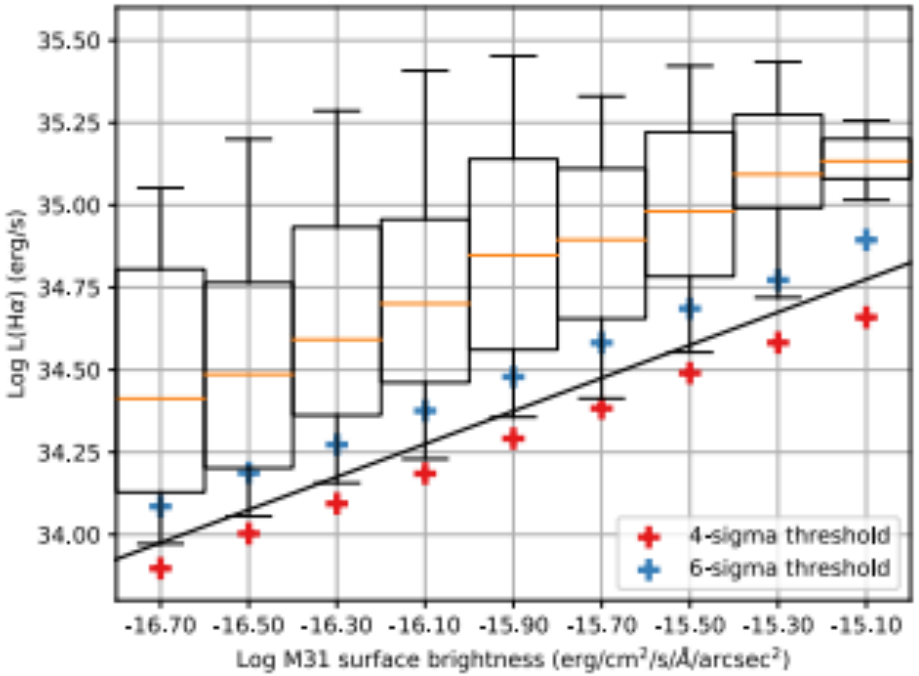


For higher dimensions, see section 1.6.2 of astroML book or source code of Figure 1.11 and 1.12

Box and Whisker Plots



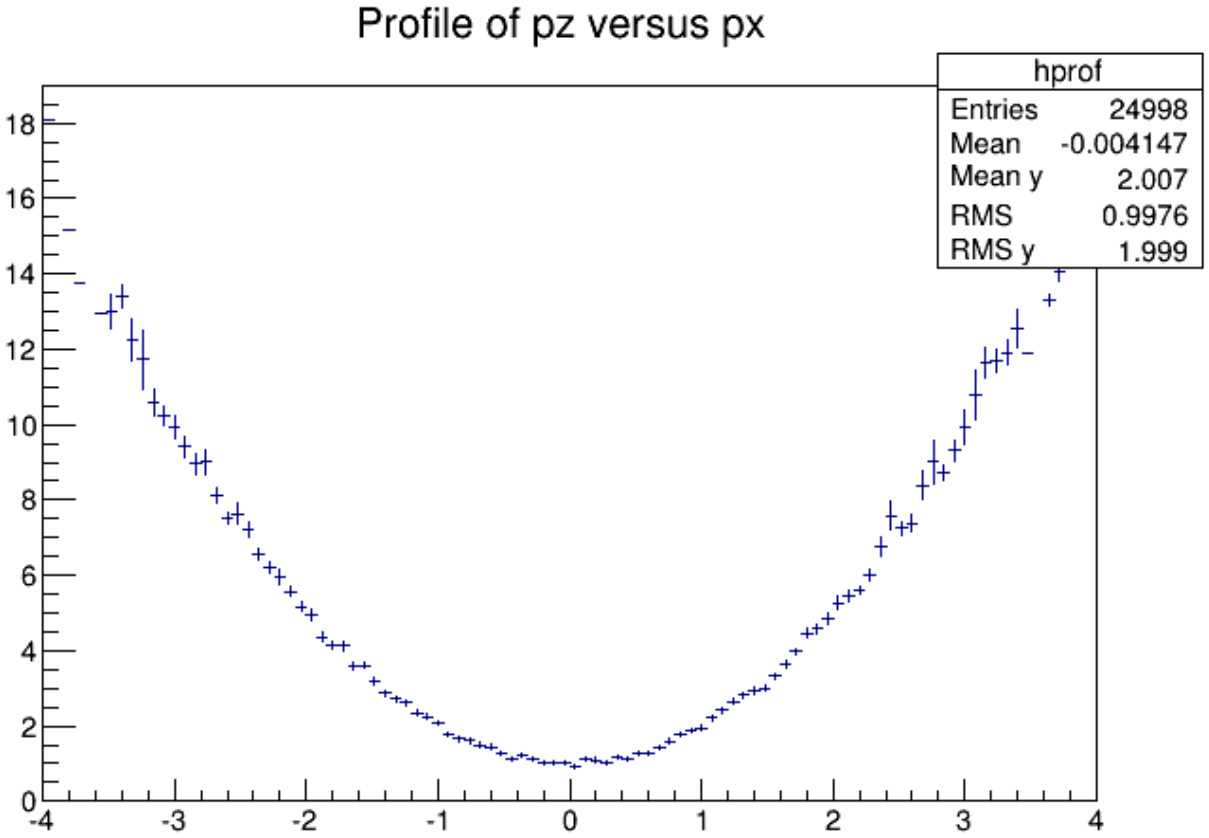
wikipedia



arxiv:1707.01366

Uses quartiles to represent data. In high energy physics a variant of this called profile histogram is used.

Profile Histograms (used in High energy Physics)



A profile histogram example

Not inbuilt in python. Only inbuilt in ROOT programming language. However see https://stackoverflow.com/questions/23709403/plotting-profile-hitstograms-in-python

How to test if two datasets are correlated?

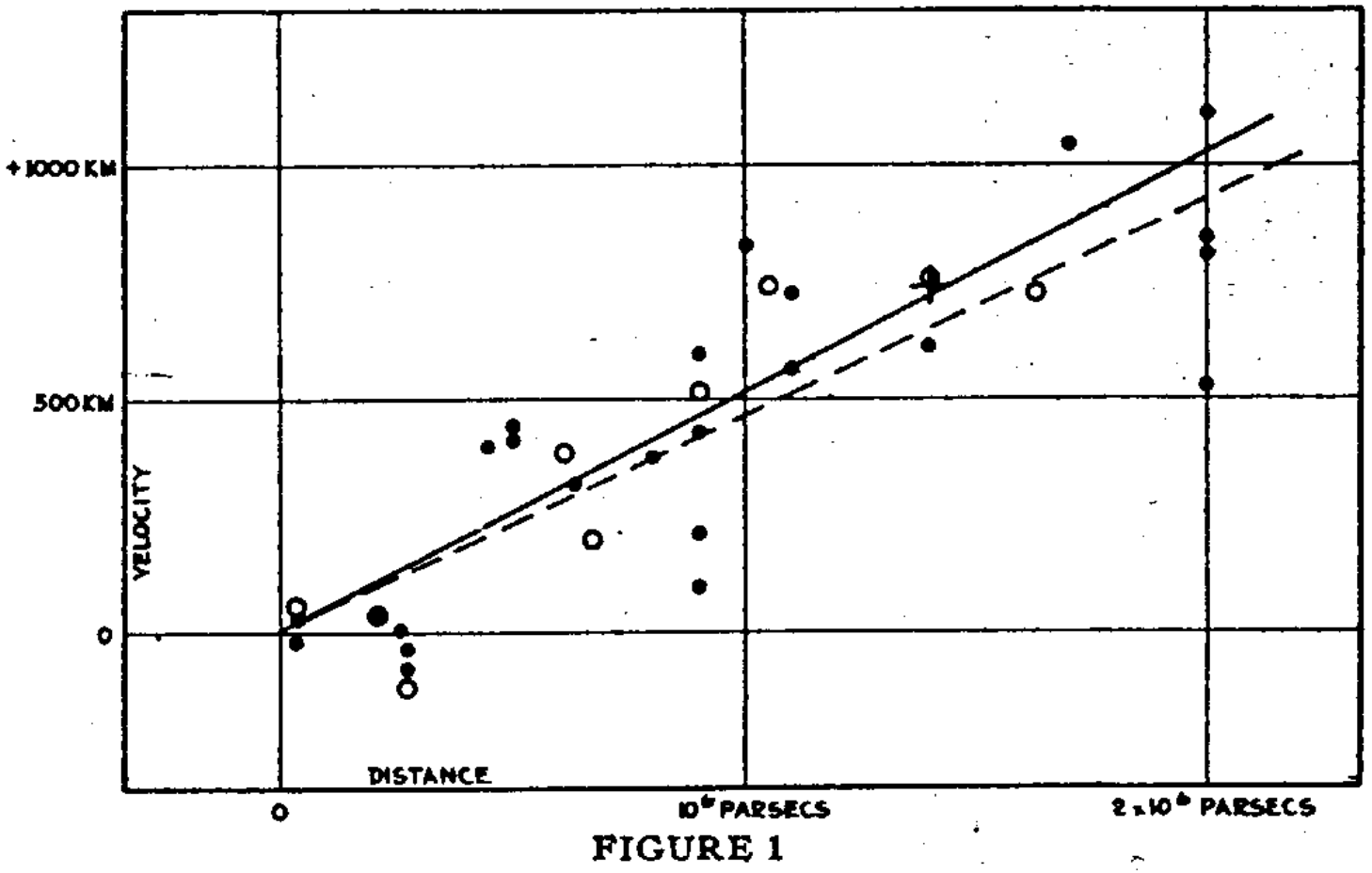
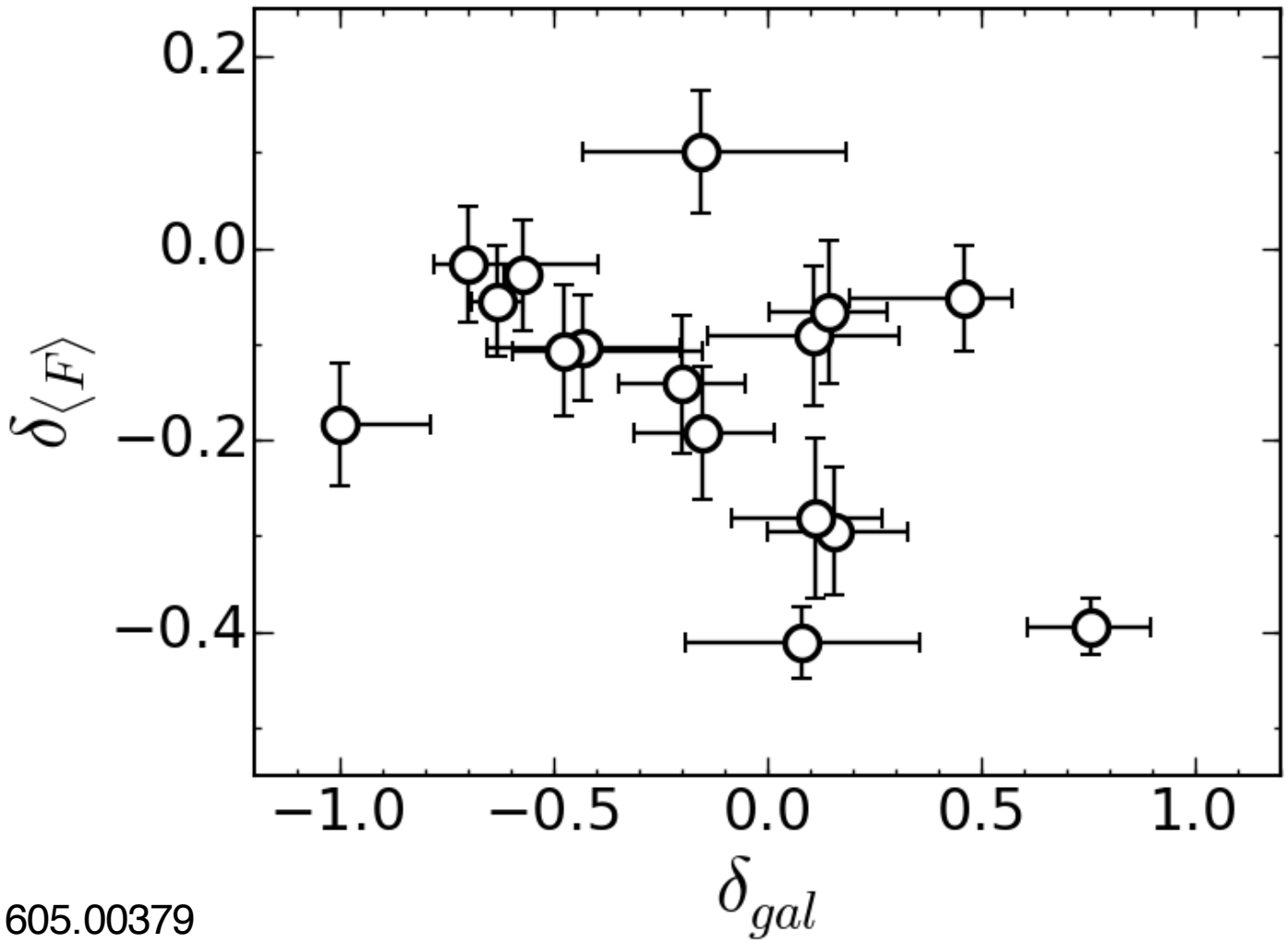


FIGURE 1

Hubble 1929 data showing Expansion of universe



arXiv:1605.00379

Correlation Coefficients

Qt : How to quantitatively assess if two datasets $\{x_i\}$ and $\{y_i\}$ of size N are correlated

Introduce three such terms : Pearson's sample correlation coeff. , Spearman rank correlation coeff., and Kendall tau

```
from scipy import stats
x,y = np.random.random((2,100)) # two random arrays
corr_coeff,p_value = stats.pearsonr(x,y)
rho,p_value = stats.spearmanr(x,y)
tau,p_value = stats.kendalltau(x,y)
```

Pearson Correlation Coefficient

$$r = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}}$$

-1 <= r <=1 For uncorrelated datasets r = 0

If the pairs (x_i, y_i) are drawn from two uncorrelated univariate Gaussian distributions (i.e. the population correlation coefficient ρ=0), distribution of r follows Students t-distribution with k=N-2 degrees of freedom and t given by

$$t = r\sqrt{\frac{N-2}{1-r^2}}$$

Example:

N=10 r=0.72

→ Probability of getting a value as large as the observed value of r which is 0.72 (by chance from a random Fluctuation) is 1%

Or one-sided 99% confidence level for Student's t-distribution with k=8 DOF is t=2.93

```
>>> import numpy as np
>>> x=0.72*np.sqrt(8/(1-0.72*0.72))
>>> x
2.9345009253812004
>>> from scipy.stats import t
>>> t.cdf(2.934,8)
0.99253
```

→ Cumulative Distribution function of t-distribution

- For bivariate Gaussian distribution with a non-vanishing population correlation coefficient **ρ** , the Fisher transformation (or Fisher z-transformation) can be used to estimate the confidence interval for **ρ** from the measured value of r. The distribution of F is given by:

$$F(r) = \frac{1}{2} \ln \left(\frac{1+r}{1-r} \right)$$

approximately follows a Gaussian distribution with mean $\mu = F(\boldsymbol{\rho})$ and a standard deviation given by $\boldsymbol{\sigma}_F = (N-3)^{-1/2}$

Advantages/Disadvantages of Pearson Correlation Coefficient

- r does not take into account errors in x_i and y_i
- r is very sensitive to Gaussian outliers

Spearman Rank Correlation Coefficient

Spearman Correlation Coefficient (r_s) based on the concept of **ranks** (sort the data in ascending order and the index i of a value x_i in the sorted data is its rank R_i^x)

$$r_s = \frac{\sum_{i=1}^{N}(R_i^X - \overline{R^X})(R_i^Y - \overline{R^Y})}{\sqrt{\sum_{i=1}^{N}(R_i^X - \overline{R^X})^2}\sqrt{\sum_{i=1}^{N}(R_i^Y - \overline{R^Y})^2}}$$

Same as Pearson correlation coefficients for the ranks. Distribution of Spearman correlation coeff. (for null hypothesis) is same as Pearson correlation coefficient

Alternate definition:
(Lupton 93)

$$r_s = 1 - \frac{6}{N(N^2-1)}\sum_{i=1}^{N}(R_i^X - R_i^Y)^2$$

Kendall Tau Correlation Coefficient

- Rank the two datasets. Count the number of *concordant* pairs, defined by $(x_j - x_k)(y_j - y_k) > 0$ and *discordant* pairs defined by $(x_j - x_k)(y_j - y_k) < 0$

For a perfect correlation (anti-correlation) all possible N(N-1)/2 pairs will be concordant (discordant) .

Kendall's tau is defined as :

$$\tau = 2\frac{N_c - N_d}{N(N-1)} \qquad -1 \le \tau \le 1$$

- When N> 10 distribution of Kendall tau in case of no-correlation (null hypothesis) can be approximated as a Gaussian distribution with μ=0 and width given by

$$\sigma_\tau = \left[\frac{2(2N+5)}{9N(N-1)}\right]^{1/2}$$

- True distributions of Spearman and Pearson correlation coefficient harder to represent analytically in case of a true correlation for a general case

- For a bi-variate Gaussian distribution with true correlation coefficient **ρ** expectation value for Kendall's tau is given by (arXiv:1011.2009)

$$\bar{\tau} = \frac{2}{\rho} sin^{-1}(\rho)$$

.

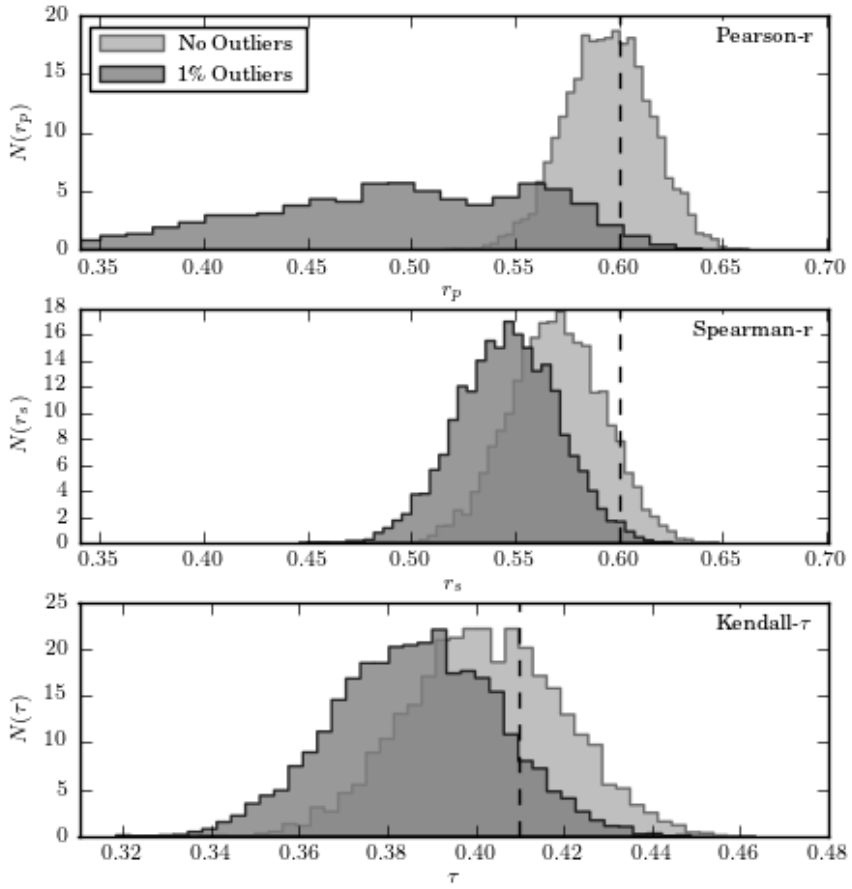
- Kendall's co-relation coefficient approaches normality faster than Spearman's correlation coefficient

- Usually, a bootstrap method is used to place confidence estimates on the measured values

Advantages of Kendall tau over Spearman r correlation coeff.

- Kendall tau correlation coefficient approaches normality faster than Spearman correlation coefficient
- Kendall tau offers a more unbiased estimate of population value while Spearman correlation coeff. does not. (Lupton 1993)
- Efficiency of Kendall tau relative to Pearson correlation coefficient for a bi-variate Gaussian greater than 90% and much greater than non-Gaussian distributions.

Distribution of correlation coefficient



2000 bootstrap resamples of 1000 datapoints drawn from a bivariate Gaussian with **ρ**=0.6 without and with (1%) outliers

Pearson correlation coefficient not robust against outliers.

Spearman and Kendall correlation coefficient have variance which is robust to outliers

Random Number Generation

Distributions in scipy.stats.distributions have a method called rvs which generates pseudo—random sample from the distribution.

Also numpy.random implements samplers for a number of distributions (look up numpy.random documentation)
(behind the scenes, scipy calls numpy)

Select 5 random integers between 5 and 10:

```
>>> import numpy as np
>>> np.random.random_integers(0,10,5)
    array([10, 4, 0, 8, 2])
```

Numerical simulations of measurement processes are called Monte Carlo simulations and the resulting samples are called Monte Carlo or mock samples.

See discussion in Numerical Recipes on how to generate random number distributions

Parameter Estimation to bivariate Gaussian distribution

- For 1-D Gaussian mean and standard deviation can be estimated from sample mean and sample variance . This can be extended to 2-D

- Correlation coefficient (**ρ**) can be estimated using Pearson's sample correlation coefficient. Principal axes can be found with α estimated using

$$\tan(2\alpha) = 2\frac{s_x s_y}{s_x^2 - s_y^2} r$$

However this doesn't work in case of outliers.
$\boldsymbol{\sigma}_x$ and $\boldsymbol{\sigma}_y$ can estimated from interquartile range (or MAD).

Robust estimate of ρ

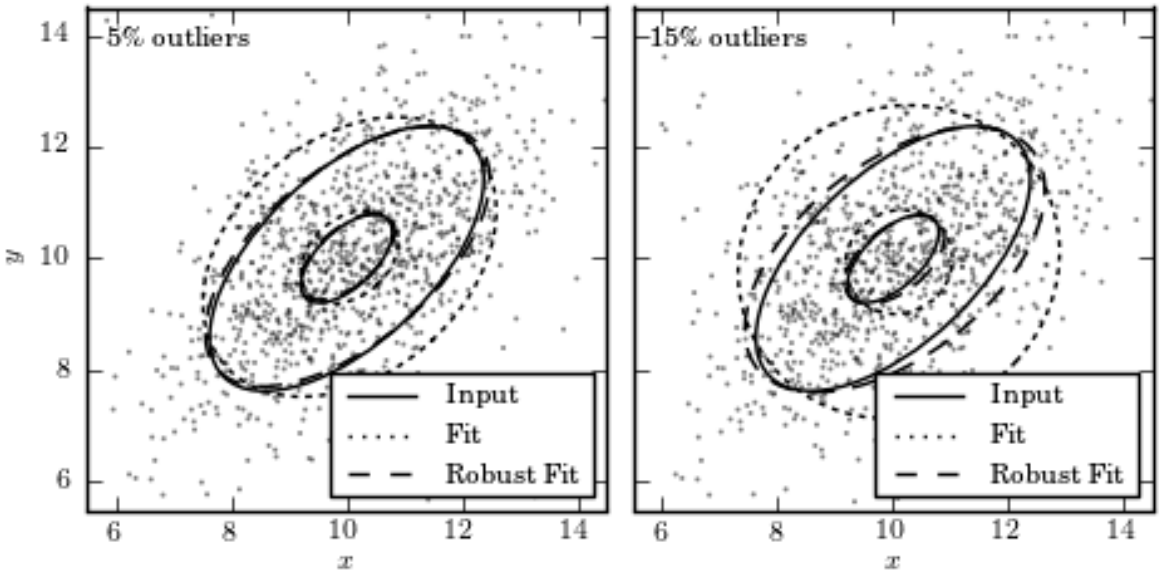
$$\rho = \frac{V_u - V_w}{V_u + V_w}$$

$$u = \sqrt{2}\left(\frac{x}{\sigma_x} + \frac{y}{\sigma_y}\right) \qquad v = \sqrt{2}\left(\frac{x}{\sigma_x} - \frac{y}{\sigma_y}\right)$$

Ref : Shevlyakov and Smirnov, Austrian Journal of Statistics, 40, 147-156 (2011)

Python Code for robust estimate of bivariate dist params

```
From astroML.stats import fit_bivariate_normal
(mu_r,sigma1_r,sigma2_r,alpha_r) = fit_bivariate_normal(x,y,robust=True)
```



An example of computing the components of a bivariate Gaussian using a sample with 1000 data values (points), with two levels of contamination. The core of the distribution is a bivariate Gaussian with $(\mu_x, \mu_y, \sigma_1, \sigma_2, \alpha) = (10, 10, 2, 1, 45^\odot)$ The "contaminating" subsample contributes 5% (left) and 15% (right) of points centered on the same (μ_x, μ_y), and with $\sigma_1 = \sigma_2 = 5$. Ellipses show the 1- and 3-sigma contours. The solid lines correspond to the input distribution. The thin dotted lines show the nonrobust estimate, and the dashed lines show the robust estimate of the best-fit distribution parameters (see Section 3.5.3 for details).