From count to measure

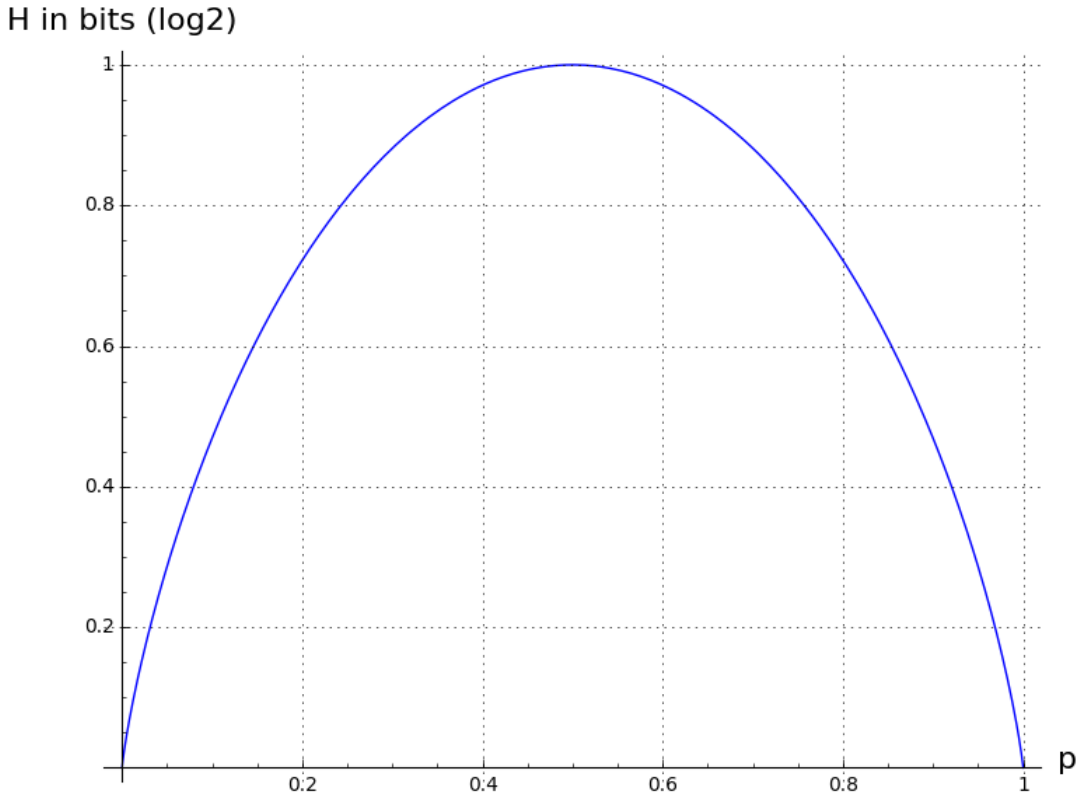
The Bernoulli random variable is binary, let the number 1 stand for *success* and the number 0 for *fail* so the number of successes in a sample of n Bernoulli experiments is the sum of n numbers.

Let p stands for the probability of success, $q = 1 - p$ will be the probability of fail. Then the mean (μ), the variance (σ^2) and the informational or Shanonn entropy (H) will be

$$\begin{pmatrix} \mu = p \\ \sigma^2 = pq \\ H = -\log(p^p q^q) \end{pmatrix}$$

The function $p^p q^q$ is well behaved for probabilities and the entropy has a maximum at p=0.5 $H_{p=0.5} = \log\left(\frac{1}{2}\right)$

```
var('p')
plot(-log(p**p*(1-p)**(1-p))/log(2),(p,0,1),axes_labels=['p','H in bits (log2)'],
gridlines=True, title='')
```



It is kown that the Binomial pdf could be build from n IID Bernoulli random variables. The binomial variable, x, represents the number of successes in a sample of n IID Bernoulli experiments.

If a random variable sticks to Binomial distribution with parameters n and p then the probability of an outcome x is $P_{B(n,p)}(x) = \binom{n}{x} p^x (1-p)^{(n-x)}$

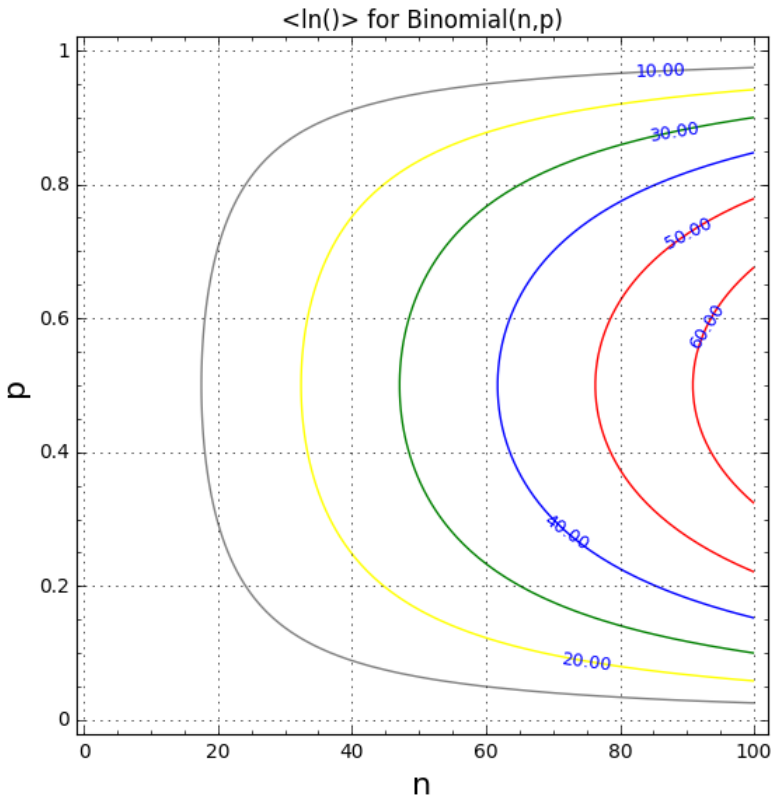
For the binomial pdf the mean (μ), the variance (σ^2) and the informational or Shanonn entropy (H) will be

$$\begin{pmatrix} \mu = np \\ \sigma^2 = npq \\ H = -n\log(p^p q^q) - \left\langle \log\binom{n}{x} \right\rangle \end{pmatrix}$$

Where $\left\langle \log\binom{n}{x} \right\rangle$ stands for the expected value of the log of the combinatorial number. I suppose that the <> notation for the expected value is more convenient than E()

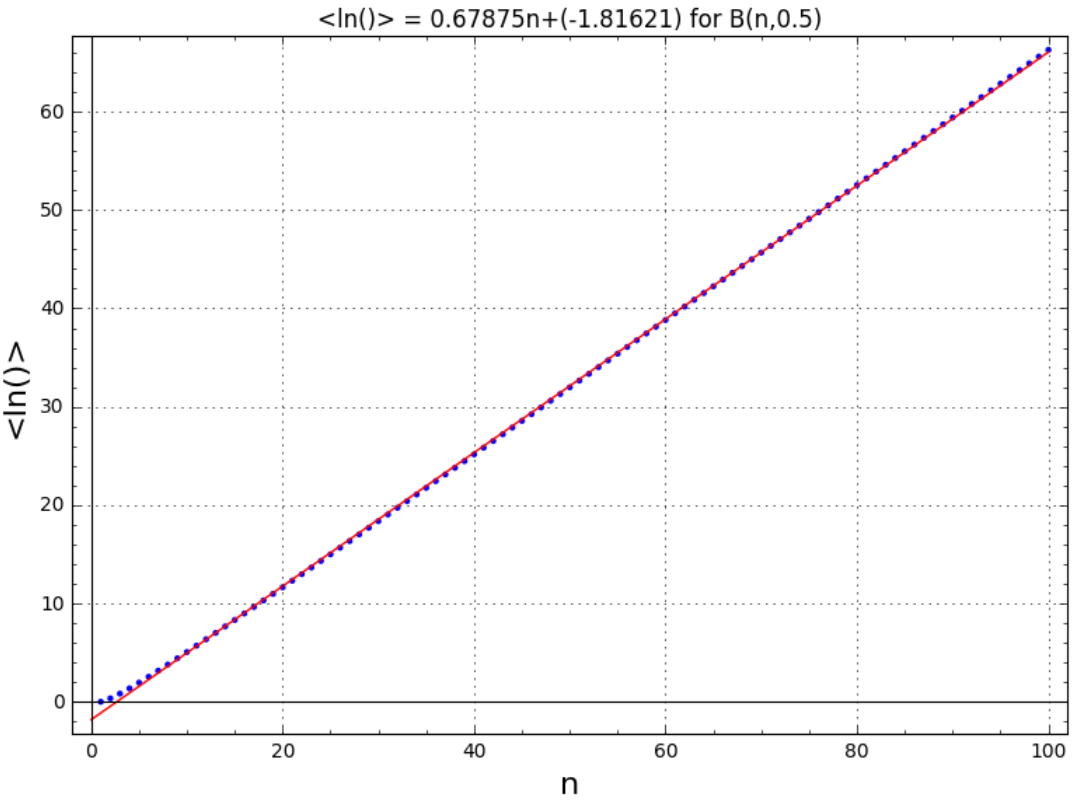
$\left\langle \log\binom{n}{x} \right\rangle$ is a well behaved function

```
n,p=var('n','p')
contour_plot(expectec, (n,1,100),(p,0,1),fill=False, plot_points=100, cmap
=['grey','yellow','green','blue','red'], labels=True,aspect_ratio=100,frame=True,
axes_labels=['n','p'], gridlines=True, title='<ln()> for Binomial(n,p)')
```



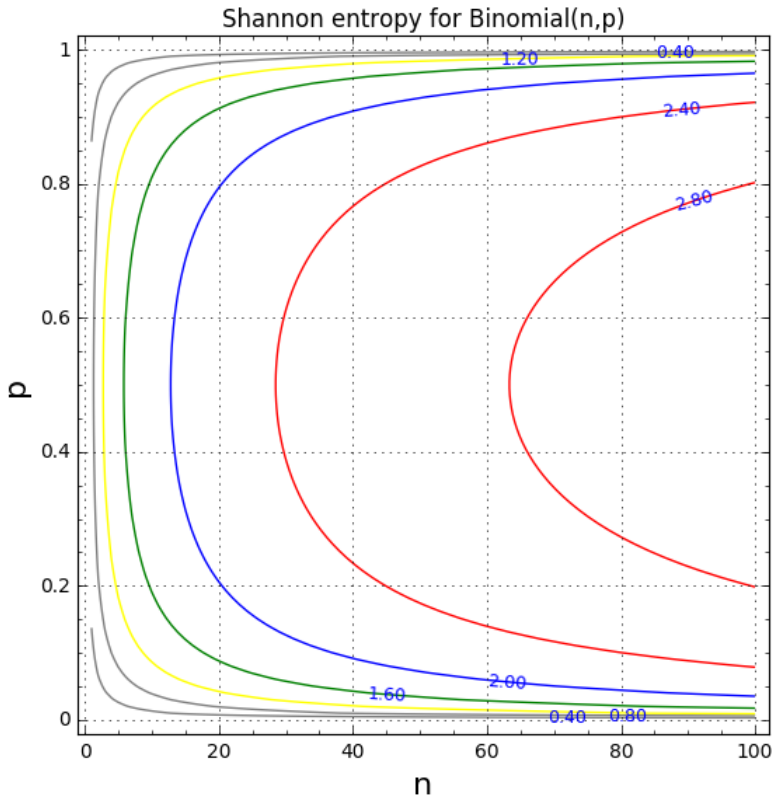
For $p = 0.5$ it shows a clear dependency on n

```
data=[(i,expectec(i)) for i in range(1,101)]
var('a,b,x')
fit=find_fit(data,a*x+b,variables=[x])

xdata=[i[0] for i in data]
ydata=[i[1] for i in data]

cad=('<ln()> = %gn+(%g) for B(n,0.5)' % (float(fit[0].rhs()),float(fit[1].rhs())))


list_plot(data)+plot(fit[0].rhs()*x+fit[1].rhs(),(x,0,100),color='red',frame=True,
axes_labels=['n','<ln()>'], gridlines=True,title=cad)
```



And the Shannon entropy for the ***Binomial(n,p)*** pdf is a well defined and well behaved function

```
n,p=var('n','p')
contour_plot(H_bin, (n,1,100),(p,0,1),fill=False, plot_points=100, cmap
=['grey','yellow','green','blue','red'], labels=True,aspect_ratio=100.0,frame=True,
axes_labels=['n','p'], gridlines=True,title='Shannon entropy for Binomial(n,p)')
```



But the construction of the Binomial(n,p) fulfills the conditions for apply the Central Limit Theorem, and pdf for *x* [B(n,p)] must tend to a Normal with parameters $\mu = np$ and $\sigma^2 = npq$ as *n* grows

I suppose that this mean two things:

- The convergence process is smooth
- In the limit of the convergence the two pdfs are indistinguishable.

In general there is no problem and it is accepted the approximation (for practical purposes $np \geqslant 5$) of the B(n,p) for the N(np,npq), except for informational entropy.

The problem here is that the Normal pdf is a continuous function. This implies extend the sums of the Shannon entropy original definition to Riemann integrals:

$H = - < \log(f(x, \mu, \sigma^2, \dots)) >= \int_R f(x, \mu, \sigma^2, \dots) \log(f(x, \mu, \sigma^2, \dots)) dx$

Where *x* is the random variable, and $f(x, \mu, \sigma^2, \dots)$ is the associated pdf and *R* is the set of possible values for *x*.

The big problem is that the integral could result in negative values. This have not satisfactory interpretation if you believe that $H = 0$ signifies absolute certainty and not only absence of uncertainty.

The solution established in the literature is that the H for continuous pdf is other thing than the H for discrete pdf.

So the differential entropy concept arrives and the Kullback-Lieber framework of relative entropies, and another bunch of measures looking for an invariant to scale transformations measure of uncertainty.

The (differential) entropy associated to a $N(\mu, \sigma^2)$ is $H(N(\mu, \sigma^2)) = \frac{1}{2}\log(2\pi e \sigma^2)$ and this is hard to compare with the $H(B(n, p))$

Let try another approximation. The discrete uniform distribution has *k* posible outcomes with equal probability $\frac{1}{k}$ If the minumum outcome is a then the maximum outcome is $a + k$

For the discrete uniform pdf, *U(k)* the mean (μ), the variance (σ^2) and the informational or Shanonn entropy (H) will be

$$\begin{pmatrix} \mu = a + \frac{k}{2} \\ \sigma^2 = \frac{(k+1)^2 - 1}{12} \\ H = ln(k) \end{pmatrix}$$

A sample, a vector of n IID *U(k)*, will have $H = n \log(k) = \log(k^n)$ and this could be interpreted as the log of the uniform volume of posible outcomes, that is the log of the volume of uniform uncertainty.

Suppose your k is big and is useful take the outcome as $\frac{k}{w}$ referred to new units which 1 new unit equates w old units.

This is no innocent: you go from integers to rationals.

Then the new entropy, the entropy expressed in the new units will be $H = \log\left(\frac{k}{w}\right)^n = \log(k^n) - \log(w^n)$

And the catastrophic negative entropy arrives if $w > k$. Would this implies that the H is a function only defined on discrete outcomes, wich could be expressed as natural numbers or counted?

Is there no entropy for continuous outcomes that need be expressed as real or even rational numbers or measured?

Sincerely, it seems a nonsense.

What about a genuine differential entropy like the continuous uniform?

For the continuous uniform pdf, *U(X)* the mean (μ), the variance (σ^2) and the informational or Shanonn entropy (H) will be

$$\begin{pmatrix} \mu = a + \frac{R}{2} \\ \sigma^2 = \frac{R^2}{12} \\ H = ln(R) \end{pmatrix}$$

Where $a = \min(X)$ and $\max(X) = a + R$

In this case continuous and discrete versions of entropy match exactly and the interpretation must be the same for both

Another question is if R is measured it has units, and the argument of the log function must be dimensionless.

Well, physics has a bunch of examples of dimensionless numbers made ad hoc with the trick of divide by 1 unit.

The only consequence is that the result of the computation is referred to this unit.

Then, if $H = 0$ the length of $R = 1$ unit, the magnitude of the entropy is 0 referred to units, and the uniform uncertainty occupies a volume of 1 unit

The practical consecuence is: All the outcomes of this random variable have the same number at the units (and frozen digits all over to the left).

But nothing is say about the digits to the right of units. There is a volume of uniform uncertainty of magnitude 1 unit.

If $R = \frac{1}{10}$ then $H = -\log(10)$ and the practical consecuence is that the first digit at the right of the decimal separator contributes its entropy, wich is supposed 0 in the example, that is a fixed digit.

You can repeat this procedure an infinite number of times, making the volume of uniform uncertainty $R = 0$ and $H = -\infty$, then you reach absolute certainty.

So the value $H = 0$ is "equidistant" from the absolute certainty ($H = -\infty$) and the absolute uncertainty ($H = \infty$).

From a practical point of view $H = 0$ means "I am sure about the units, but not about tenths", and a negative value of entropy could be interpreted as "I am sure about this number up to the n-th decimal place, but not about the others".

This interpretation of entropy has consequences

- Entropy is a continous measure of uncertainty, referred to the unitary scale (which is chosen by the observer for continuous variables, and has a "natural" scale, counts, for discrete variables)
- The underpinning of a not parametric method for the estimation (mesure) of H from real samples.This is necessary because the parametric estimation of H heavily depends on the pdf attributed to data.
- It may be that the problem of measuring uncertainty and the problem of choosing the right scale for the measure are inseparable and that they are, in some way, solved at once by H

Useful functions.

1. **espectec(n,p=0.5) :** Returns the mean of the log of the combinatorial number for the Binomial(n,p)
2. **H_bin(n,p=0.5,method=1):** Returns the Shannon entropy for Binomial(n,p)
    1. method=1 theoretical <>
    2. method=2 computes actual <>
    3. method!=[1,2] returns a tuple (m1,m2)

```
def expectec(n,p=0.5):
    '''
    returns the mean of the log of the combinatorial number
    for the Binomial(n,p)
    '''
    import scipy.stats
    binom_dist = scipy.stats.binom(n,p)
    n=int(n)
    total=0

    for i in range(n+1):
        total+= log(binomial(n,i))*binom_dist.pmf(i)
    return total

def H_bin(n,p=0.5,method=1):
    '''
```

```
    Returns the Shannon entropy for
    Binomial(n,p)
    method=1 theoretical <>
    method=2 computes actual <>
    method=0 returns a tuple (m1,m2)
    '''


    H1=-1.0*(expectec(n,p)+n*log(p**p*(1-p)**(1-p)))

    if method==1:
        return N(H1)

    import scipy.stats
    binom_dist = scipy.stats.binom(n,p)
    H2=sum([-(i*log(p)+(n-i)*log(1-p))*binom_dist.pmf(i) for i in range(n +1)])-
expectec(n,p)
    if method==2:
        return N(H2)

    return (H1,H2)
```