Calculating confidence limits for Gain and Phase spectra

for which we need the cumulative Fisher distribution

$$F(n,m,x) \equiv F_x(n,m) = 1 - I_{\frac{m}{m+n\,x}}\left(\frac{m}{2},\frac{n}{2}\right)$$

where

$$I_y(\mu,\nu) = \frac{\mathrm{B}(\mu,\nu,y)}{\mathrm{B}(\mu,\nu)}$$

is the normalized incomplete Beta function (IMSL: BETAI, Numerical Recipes also BETAI, but arguments are ordered differently). The inverse of function F will be named f

$$F(n, m, x) = c \quad <=> \; f(n, m, c) \equiv f_{n,m}(c) = x$$

where $0 < c < 1$ is a confidence level.

Jenkins and Watts (1968, Chap. 10) show that in smoothed co-spectra the degree of freedom of a line in the power spectrum is n=2 and in the cross-spectrum m>>2, the latter owing to smoothing by windowing (ahum, isn't the power spectrum smoothed the same way? Well, yes, but .... see J&W chap. 6.4.2). The degrees of freedom m of C_{xy} is calculated from J&W (6.4.14) or

$$\nu = 2\,\frac{T}{M\,I}$$

where

$$I \approx \frac{1}{2\,M+1}\sum_{i=-M}^{M} w_i^2$$

(J&W use Fourier integrals) is the smoothing effect of a window of $2M < T$ and T the number of samples. Let's call I the window's shape factor. An example for the Kaiser-Bessel window with parameter 2.2 and $2M = 2^{12}$, $T = 10\times 2^{12}$ => $I = 0.34092$ => $\nu = 58.66$

In http://holt.oso.chalmers.se/hgs/4me/pdgs/KB-wsp.png the black curve is for the Kaiser-Bessel window, the blue one for a rectangular window.

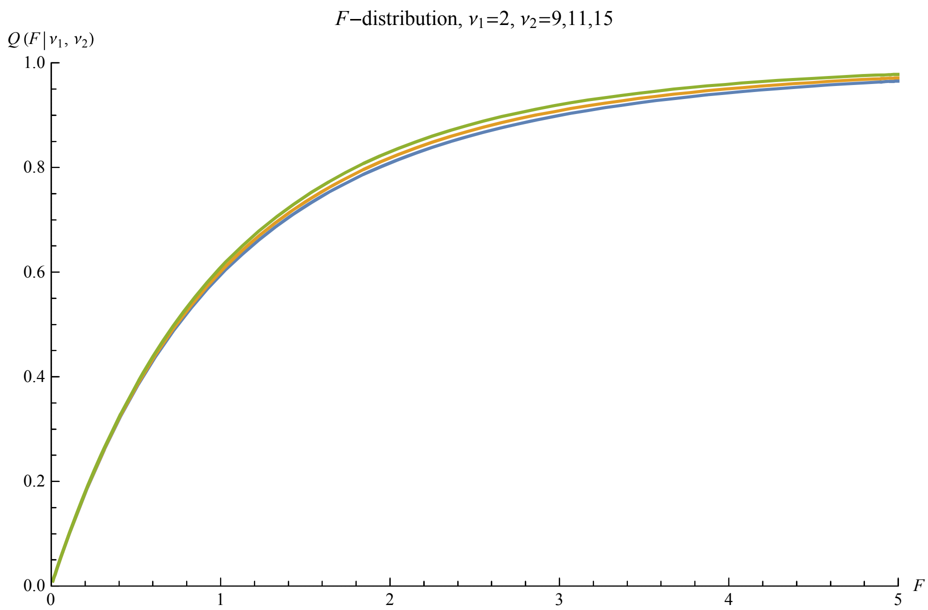
At the end of this document the confidence interval for coherence is given.

```
In[2]:=  betai[na_, nb_, x_] := Beta[x, na, nb] / Beta[na, nb]

In[55]:= fish[na_, nb_, x_] := 1 - betai[nb / 2, na / 2, nb / (nb + na x)]
```

```
Plot[{fish[2, 9, f], fish[2, 11, f], fish[2, 15, f]}, {f, 0.01, 5},
 PlotRange → {{0, 5}, {0, 1}},
 AxesLabel →
  {HoldForm[F],
   RawBoxes[
    RowBox[
     {"Q", RowBox[{"(", RowBox[{RowBox[{"F", "|", "ν1"}], ",", "ν2"}], ")"}]}]]},
 PlotLabel → "F-distribution, ν1=2, ν2=9,11,15", LabelStyle → {GrayLevel[0]}]
```

Out[95]=



For 15 degrees of freedom:

In[59]:=
```
y = 2 x / n  /. Solve[fish[2, n, x] == 0.95, x]  /.  n → 13
```

Solve::ifun : Inverse functions are being used by Solve, so
some solutions may not be found; use Reduce for complete solution information. ≫

Out[59]= {0.585472}

This shows the special case ν_1=2 of Jenkins and Watts Figure 3.12:

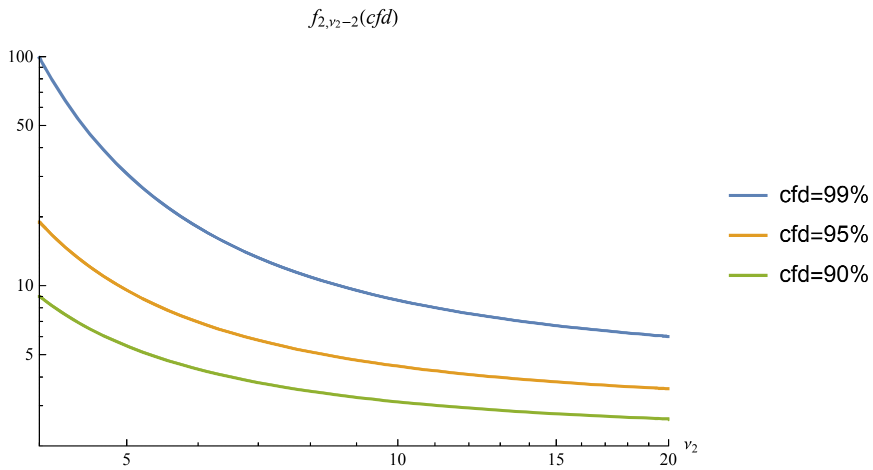
In[137]:=
```
LogLogPlot[{x /. Solve[fish[2, n - 2, x] == 0.99], x /. Solve[fish[2, n - 2, x] == 0.95],
  x /. Solve[fish[2, n - 2, x] == 0.90]}, {n, 4, 20}, PlotRange → {{4, 20}, {2, 100}},
 PlotLegends → {"cfd=99%", "cfd=95%", "cfd=90%"}, PlotLabel → "f2,ν2-2 (cfd)",
 AxesLabel → {"ν2", ""}]
```

Solve::ifun : Inverse functions are being used by Solve, so
some solutions may not be found; use Reduce for complete solution information. ≫

Solve::ifun : Inverse functions are being used by Solve, so
some solutions may not be found; use Reduce for complete solution information. ≫

Solve::ifun : Inverse functions are being used by Solve, so
some solutions may not be found; use Reduce for complete solution information. ≫

General::stop : Further output of Solve::ifun will be suppressed during this calculation. ≫

Out[137]=


In[61]:=
```
fishi[f_, na_, nb_] := x /. Solve[fish[na, nb, x] == f]
```

In[53]:=
```
2 / 13 fishi[0.9, 2, 13]
```

Out[53]= {0.425103}

Lower limit for significant coherence k:

In[62]:=
```
Solve[2 / 13 fishi[0.9, 2, 13] (1 - k) / k == 1, k]
```

Solve::ratnz : Solve was unable to solve the system with inexact coefficients. The
answer was obtained by solving a corresponding exact system and numericizing the result. ≫

Out[62]= {{k → 0.298296}}

In[110]:=
```
k0[c_, n_] := (2 / n) x / ((2 / n) x + 1) /. x → fishi[c, 2, n]
```

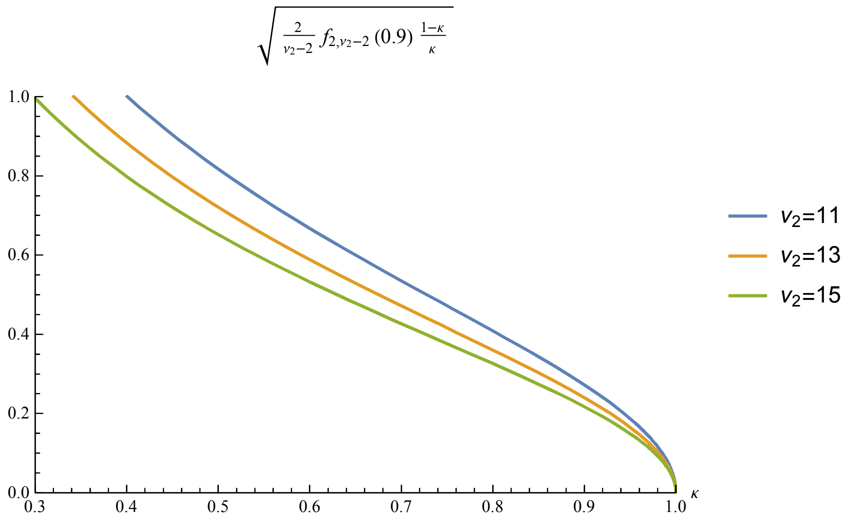
In[111]:=
```
k0[0.95, 13]
```

Out[111]= {0.369273}

Show 90%-confidence limit for Gain as a function of coherence κ

In[126]:=
```
Plot[{Sqrt[(2 (1 - k))/(9 k) fishi[0.9, 2, 9]], Sqrt[(2 (1 - k))/(11 k) fishi[0.9, 2, 11]],
  Sqrt[(2 (1 - k))/(13 k) fishi[0.9, 2, 13]]}, {k, 0.3, 1},
 PlotLegends → {"ν2=11", "ν2=13", "ν2=15"}, PlotRange → {{0.3, 1}, {0, 1}},
 AxesLabel → {"κ", ""}, PlotLabel → "Sqrt[2/(ν2 - 2) f2,ν2-2 (0.9) (1 - κ)/κ]"]
```

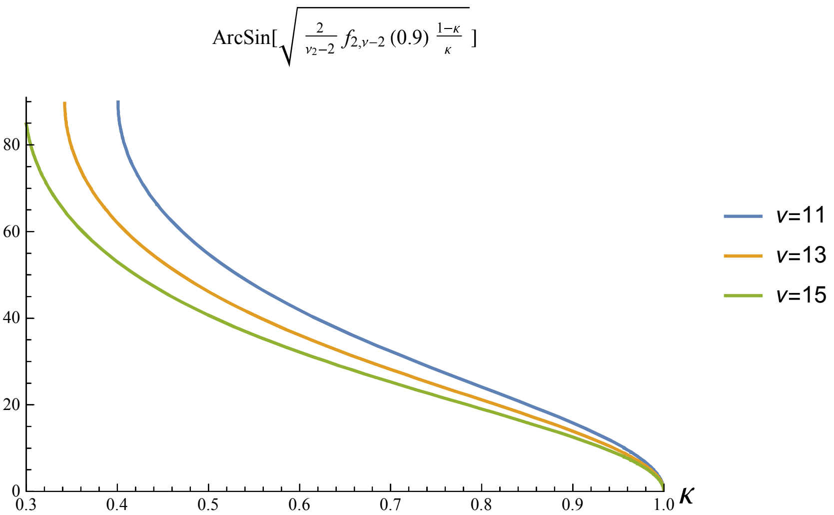
Out[126]=



Same for phase:

In[135]:=
```
Plot[{180/π ArcSin[Sqrt[(2 (1 - k))/(9 k) fishi[0.9, 2, 9]]],
  180/π ArcSin[Sqrt[(2 (1 - k))/(11 k) fishi[0.9, 2, 11]]],
  180/π ArcSin[Sqrt[(2 (1 - k))/(13 k) fishi[0.9, 2, 13]]]}, {k, 0.3, 1},
 PlotLegends → {"ν=11", "ν=13", "ν=15"}, PlotRange → {{0.3, 1}, {0, 91}},
 AxesLabel → {Style[κ, FontSize → 18], ""},
 PlotLabel → "ArcSin[Sqrt[2/(ν₂ - 2) f₂,ν-2 (0.9) (1 - κ)/κ] ]"]
```

Out[135]=



Finally the confidence interval for coherence (we always use the coherence-square albeit denoted by a bare and straight κ) :

According to J&W (Chap. 9.2, p. 380) ArcTanh($\sqrt{\kappa}$) is approximately Normal with variance $I/(2T)$. Denote the thus transformed variable with y. Then with the confidence level c and η the inverse of the cumulative Normal distribution

$$\delta y = \pm\, \eta\left(\tfrac{c+1}{2}\right) \sqrt{\frac{I}{2\,T}}$$

where I is again the window's shape factor. Then

$$\delta\kappa = \mathrm{Tanh}^2\left(\delta y \pm \mathrm{ArcTanh}\left[\sqrt{\kappa}\,\right]\right) - \kappa$$

Proof:

In[142]:=
```
yk[k_] := ArcTanh[Sqrt[k]]
```

In[148]:=
```
{Solve[yk[kk + u] == yk[kk] + v, u], Solve[yk[kk + u] == yk[kk] - v, u]}
```

Out[148]=
```
{{{u → -kk + Tanh[v + ArcTanh[Sqrt[kk]]]^2}}, {{u → -kk + Tanh[v - ArcTanh[Sqrt[kk]]]^2}}}
```