

Characteristic Functions in the *prob* package

G. Jay Kerns

January 22, 2009

Contents

1	Introduction	2
2	Characteristic functions	2
2.1	Beta distribution: <code>cfbeta(t, shape1, shape2, ncp = 0)</code>	2
2.2	Binomial distribution: <code>cfbinom(t, size, prob)</code>	3
2.3	Cauchy Distribution: <code>cfcauchy(t, location = 0, scale = 1)</code>	4
2.4	Chi-square Distribution: <code>cfchisq(t, df, ncp = 0)</code>	4
2.5	Exponential Distribution: <code>cfexp(t, rate = 1)</code>	4
2.6	<i>F</i> Distribution: <code>cff(t, df1, df2, ncp, kmax = 10)</code>	5
2.7	Gamma Distribution: <code>cfgamma(t, shape, scale = 1/rate)</code>	6
2.8	Geometric Distribution: <code>cfgeom(t, prob)</code>	6
2.9	Hypergeometric Distribution: <code>cfhyper(t, m, n, k)</code>	6
2.10	Logistic Distribution: <code>cflogis(t, location = 0, scale = 1)</code>	7
2.11	Lognormal Distribution: <code>cflnorm(t, meanlog = 0, sdlog = 1)</code>	7
2.12	Negative Binomial Distribution: <code>cfnbinom(t, size, prob, mu)</code>	8
2.13	Normal Distribution: <code>cfnorm(t, mean = 0, sd = 1)</code>	9
2.14	Poisson Distribution: <code>cfpois(t, lambda)</code>	9
2.15	Wilcoxon Signed Rank Distribution: <code>cfsignrank(t, n)</code>	9
2.16	Student's <i>t</i> Distribution: <code>cft(t, df, ncp)</code>	10
2.17	Continuous Uniform Distribution: <code>cfunif(t, min = 0, max = 1)</code>	11
2.18	Weibull Distribution: <code>cfweibull(t, shape, scale = 1)</code>	11
2.19	Wilcoxon Rank Sum Distribution: <code>cfwilcox(t, m, n)</code>	12
3	R Session information	12

1 Introduction

The characteristic function (c.f.) of a random variable X is defined by

$$\phi_X(t) = \mathbb{E}e^{itX}, \quad -\infty < t < \infty.$$

When the distribution of X is discrete with probability mass function (p.m.f.) $p_X(x)$, the c.f. takes the form

$$\phi_X(t) = \sum_{x \in S_X} e^{itx} p_X(x),$$

where S_X is the support of X . When the distribution of X is continuous with probability density function (p.d.f.) $f_X(x)$, the c.f. takes the form

$$\phi_X(t) = \int_{S_X} e^{itx} f_X(x) dx.$$

Characteristic functions have many, many useful properties: for example, every c.f. is uniformly continuous and bounded in modulus (by 1). Furthermore, a random variable has a distribution symmetric about 0 if and only if its associated c.f. is real-valued. For details, see [7].

Most of the below formulas came from [8, 9, 10]. Some of them involve special mathematical functions and a classical reference for them is [2], but many of the definitions have made it to Wikipedia (<http://www.wikipedia.org/>) and selected links to the respective Wikipedia topics have been listed when appropriate.

Note that the returned value of a characteristic function is a *complex* number, and is represented as such in R, even for those c.f.'s which correspond to symmetric distributions. Thus, `cfnorm(0) = 1 + 0i`, and *not* `cfnorm(0) = 1`. Depending on the application, the respective c.f.'s may need to be wrapped in `as.real()`.

All of the below functions were written in straight R code; it would likely be possible to speed up evaluation if for example they were written in C or some other language. I would welcome any contributions for improvement in the *prob* package.

There are three special cases: the noncentral Beta, noncentral Student's t , and Weibull distributions. For these the c.f.'s are integrated numerically and thus are subject to all of numerical integration's limitations and idiosyncracies. I would be especially interested in and appreciative of a reference for these cases to be improved.

2 Characteristic functions

The formulas for all characteristic functions supported in the *prob* package are listed below, in alphabetical order of the function name.

2.1 Beta distribution: `cfbeta(t, shape1, shape2, ncp = 0)`

Let α and β denote the `shape1` and `shape2` parameters, respectively. The p.d.f. is then

$$f_X(x) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1} (1-x)^{\beta-1}, \quad 0 < x < 1,$$

where Γ is the *gamma function* defined by

$$\Gamma(\alpha) = \int_0^\infty u^{\alpha-1} e^{-u} du, \quad \alpha \neq 0, -1, -2, \dots$$

The characteristic function is given by

$$\phi_X(t) = {}_1F_1(\alpha; \alpha + \beta; it),$$

where ${}_1F_1$ is *Kummer's confluent hypergeometric function of the first kind*, also known as Kummer's M , defined by

$${}_1F_1(a; b; z) = \sum_{n=0}^{\infty} \frac{(a)_n z^n}{(b)_n n!},$$

with $(a)_n = a(a+1)(a+2)\cdots(a+n-1)$ the *rising factorial*. We calculate ${}_1F_1$ using `kummerM` in the *fAsianOptions* package.

As of the time of this writing, it seems that we must resort to calculating the characteristic function for the noncentral Beta by numerical integration according to the definition; see the source below. If you are aware of a way to more quickly/reliably calculate this c.f. with R, I would appreciate it if you would let me know.

Source Code:

```
function (t, shape1, shape2, ncp = 0)
{
  if (shape1 <= 0 || shape2 <= 0)
    stop("shape1, shape2 must be positive")
  if (identical(all.equal(ncp, 0), TRUE)) {
    require(fAsianOptions)
    kummerM((0+1i) * t, shape1, shape1 + shape2)
  }
  else {
    fr <- function(x) cos(t * x) * dbeta(x, shape1, shape2,
      ncp)
    fi <- function(x) sin(t * x) * dbeta(x, shape1, shape2,
      ncp)
    Rp <- integrate(fr, lower = 0, upper = 1)$value
    Ip <- integrate(fi, lower = 0, upper = 1)$value
    return(Rp + (0+1i) * Ip)
  }
}
<environment: namespace:prob>
```

2.2 Binomial distribution: `cfbinom(t, size, prob)`

Let n and p denote the size and prob arguments, respectively. Then the p.m.f. is

$$p_X(x) = \binom{n}{x} p^x (1-p)^{n-x}, \quad x = 0, 1, 2, \dots, n.$$

The characteristic function is given by

$$\phi_X(t) = [pe^{it} + (1-p)]^n.$$

Source Code:

```
function (t, size, prob)
{
  if (size <= 0)
    stop("size must be positive")
  if (prob < 0 || prob > 1)
    stop("prob must be in [0,1]")
  (prob * exp((0+1i) * t) + (1 - prob))^size
}
<environment: namespace:prob>
```

2.3 Cauchy Distribution: `cfcauchy(t, location = 0, scale = 1)`

Let θ and σ denote the `location` and `scale` parameters, respectively. The p.d.f. is then

$$f_X(x) = \frac{1}{\pi\sigma} \frac{1}{\left[1 + \left(\frac{x-\theta}{\sigma}\right)^2\right]}, \quad -\infty < x < \infty.$$

The characteristic function is given by

$$\phi_X(t) = e^{it\theta - \sigma|t|}.$$

Source Code:

```
function (t, location = 0, scale = 1)
{
  if (scale <= 0)
    stop("scale must be positive")
  exp((0+1i) * location * t - scale * abs(t))
}
<environment: namespace:prob>
```

2.4 Chi-square Distribution: `cfchisq(t, df, ncp = 0)`

Let p and δ denote the `df` and `ncp` parameters, respectively. The p.d.f. of the central chi-square distribution ($\delta = 0$) is then

$$f_X(x) = \frac{1}{\Gamma(p/2) \cdot 2^{p/2}} x^{p/2-1} e^{-x/2}, \quad x > 0.$$

One way to then write the p.d.f. of the noncentral chi-square distribution ($\delta > 0$) is with an infinite series:

$$f_X(x) = \sum_{k=0}^{\infty} \frac{e^{-\delta/2} (\delta/2)^k}{k!} f_{p+2k}(x), \quad x > 0,$$

where f_{p+2k} is the p.d.f. of a central chi-square distribution with $p+2k$ degrees of freedom. The characteristic function in both cases is given by

$$\phi_X(t) = \frac{\exp\left\{\frac{i\delta t}{1-2it}\right\}}{(1-2it)^{p/2}}.$$

Source Code:

```
function (t, df, ncp = 0)
{
  if (df < 0 || ncp < 0)
    stop("df and ncp must be nonnegative")
  exp((0+1i) * ncp * t / (1 - (0+2i) * t)) / (1 - (0+2i) * t)^(df/2)
}
<environment: namespace:prob>
```

2.5 Exponential Distribution: `cfexp(t, rate = 1)`

This is the special case of the Gamma distribution when $\alpha = 1$. See Section [2.7](#).

Source Code:

```
function (t, rate = 1)
{
  cfgamma(t, shape = 1, scale = 1/rate)
}
<environment: namespace:prob>
```

2.6 F Distribution: `cff(t, df1, df2, ncp, kmax = 10)`

Let p and q denote the `df1` and `df2` parameters, respectively, and let λ denote the noncentrality parameter `ncp`. We may write the p.d.f. for the central F distribution ($\lambda = 0$) with

$$f_X(x) = \frac{\Gamma[(p+q)/2]}{\Gamma(p/2)\Gamma(q/2)} \left(\frac{p}{q}\right)^{p/2} x^{p/2-1} \left(1 + \frac{p}{q}x\right)^{-(p+q)/2}, \quad x > 0.$$

The characteristic function for central F is given by

$$\phi_X(t) = \frac{\Gamma[(p+q)/2]}{\Gamma(q/2)} \Psi\left(\frac{p}{2}, 1 - \frac{q}{2}; -\frac{q}{p}it\right),$$

where Ψ is *Kummer's confluent hypergeometric function of the second kind*, also known as Kummer's U , defined by

$$\Psi(a, b; z) = \frac{\pi}{\sin \pi b} \left(\frac{{}_1F_1(a; b; z)}{\Gamma(1+a-b)\Gamma(b)} - z^{1-b} \frac{{}_1F_1(1+a-b; 2-b; z)}{\Gamma(a)\Gamma(2-b)} \right).$$

See [1] in the references. Kummer's U is calculated with `kummerU`, again from the *fAsianOptions* package.

The p.d.f. of the noncentral F distribution ($\lambda \neq 0$) as

$$f_X(x) = f_{p,q}(x) e^{-\lambda/2} \sum_{k=0}^{\infty} \left\{ \left(\frac{\frac{1}{2}\lambda p x}{q + p x} \right)^k \cdot \frac{(p+q)(p+q+2) \cdots (p+q+2 \cdot k-1)}{k! p(p+2) \cdots (p+2 \cdot k-1)} \right\}, \quad x > 0,$$

where $f_{p,q}$ is the p.d.f. of the central F distribution. The characteristic function for the noncentral F distribution is given by

$$\phi_X(t) = e^{-\lambda/2} \sum_{k=0}^{\infty} \frac{(\lambda/2)^k}{k!} {}_1F_1\left(\frac{p}{2} + k; -\frac{q}{2}; -\frac{q}{p}it\right),$$

where ${}_1F_1$ is Kummer's confluent hypergeometric function of the first kind defined above; see Section 2.1. For the purposes of calculation, we may only use a finite sum to approximate the infinite series, thus the user should specify an upper value of k to be used, denoted `kmax`, which has the default value of `kmax = 10`.

Source Code:

```
function (t, df1, df2, ncp, kmax = 10)
{
  if (df1 <= 0 || df2 <= 0)
    stop("df1 and df2 must be positive")
  require(fAsianOptions)
  if (identical(all.equal(ncp, 0), TRUE)) {
    gamma((df1 + df2)/2)/gamma(df2/2) * kummerU(-(0+1i) *
      df2 * t/df1, df1/2, 1 - df2/2)
  }
  else {
```

```

      exp(-ncp/2) * sum((ncp/2)^(0:kmax)/factorial(0:kmax) *
        kummerM(-(0+1i) * df2 * t/df1, df1/2 + 0:kmax, -df2/2))
    }
  }
<environment: namespace:prob>

```

2.7 Gamma Distribution: `cfgamma(t, shape, scale = 1/rate)`

Let α and β denote the **shape** and **scale** parameters, respectively. The p.d.f. is then

$$f_X(x) = \frac{1}{\Gamma(\alpha)\beta^\alpha} x^{\alpha-1} e^{-x/\beta}, \quad x > 0.$$

The characteristic function is given by

$$\phi_X(t) = (1 - \beta it)^{-\alpha}.$$

Source Code:

```

function (t, shape, rate = 1, scale = 1/rate)
{
  if (rate <= 0 || scale <= 0)
    stop("rate must be positive")
  (1 - scale * (0+1i) * t)^(-shape)
}
<environment: namespace:prob>

```

2.8 Geometric Distribution: `cfgeom(t, prob)`

This is the special case of the Negative Binomial distribution when $r = 1$; see Section [2.12](#).

Source Code:

```

function (t, prob)
{
  cfnbinom(t, size = 1, prob = prob)
}
<environment: namespace:prob>

```

2.9 Hypergeometric Distribution: `cfhyper(t, m, n, k)`

The p.m.f. takes the form

$$p_X(x) = \frac{\binom{m}{x} \binom{n}{k-x}}{\binom{m+n}{k}}, \quad x = 0, \dots, k; \quad x \leq m; \quad k - x \leq n.$$

The characteristic function is given by

$$\phi_X(t) = \frac{{}_2F_1(-k, -m; n - k + 1; e^{it})}{{}_2F_1(-k, -m; n - k + 1; 1)},$$

where ${}_2F_1$ is the *Gaussian hypergeometric series* defined by

$${}_2F_1(a, b; c; z) = \sum_{n=0}^{\infty} \frac{(a)_n (b)_n}{(c)_n} \frac{z^n}{n!},$$

with $(a)_n$ the rising factorial defined as above in Section 2.1. See [3] in the References for details concerning ${}_2F_1$. We calculate it by means of the `hypergeo` function in the *hypergeo* package.

Source Code:

```
function (t, m, n, k)
{
  if (m < 0 || n < 0 || k < 0)
    stop("m, n, k must be positive")
  hypergeo:::hypergeo(-k, -m, n - k + 1, exp((0+1i) * t))/hypergeo:::hypergeo(-k,
    -m, n - k + 1, 1)
}
<environment: namespace:prob>
```

2.10 Logistic Distribution: `cflogis(t, location = 0, scale = 1)`

Let μ and σ denote the `location` and `scale` parameters, respectively. The p.d.f. is then

$$f_X(x) = \frac{e^{-(x-\mu)/\sigma}}{\sigma (1 + e^{-(x-\mu)/\sigma})^2}, \quad -\infty < x < \infty.$$

The characteristic function is given by

$$\phi_X(t) = e^{i\mu t} \frac{\pi \sigma t}{\sinh(\pi \sigma t)},$$

where

$$\sinh(x) = \frac{e^x - e^{-x}}{2} = -i \sin ix,$$

see [4] in the References.

Source Code:

```
function (t, location = 0, scale = 1)
{
  if (scale <= 0)
    stop("scale must be positive")
  ifelse(identical(all.equal(t, 0), TRUE), return(1), return(exp((0+1i) *
    location) * pi * scale * t/sinh(pi * scale * t)))
}
<environment: namespace:prob>
```

2.11 Lognormal Distribution: `cflnorm(t, meanlog = 0, sdlog = 1)`

Let μ and σ denote the `meanlog` and `sdlog` parameters, respectively. The p.d.f. is then

$$f_X(x) = \frac{1}{\sigma \sqrt{2\pi} x} e^{-(\ln x - \mu)^2 / 2\sigma^2}, \quad -\infty < x < \infty.$$

The characteristic function is uniquely complicated and delicate. See [5] in the References. For fast numerical computation an algorithm due to Beaulieu is used, see [11].

Source Code:

```
function (t, meanlog = 0, sdlog = 1)
{
  if (sdlog <= 0)
    stop("sdlog must be positive")
  if (identical(all.equal(t, 0), TRUE)) {
    return(1 + (0+0i))
  }
  else {
    t <- t * exp(meanlog)
    Rp1 <- integrate(function(y) exp(-log(y/t)^2/2/sdlog^2) *
      cos(y)/y, lower = 0, upper = t)$value
    Rp2 <- integrate(function(y) exp(-log(y * t)^2/2/sdlog^2) *
      cos(1/y)/y, lower = 0, upper = 1/t)$value
    Ip1 <- integrate(function(y) exp(-log(y/t)^2/2/sdlog^2) *
      sin(y)/y, lower = 0, upper = t)$value
    Ip2 <- integrate(function(y) exp(-log(y * t)^2/2/sdlog^2) *
      sin(1/y)/y, lower = 0, upper = 1/t)$value
    return((Rp1 + Rp2 + (0+1i) * (Ip1 + Ip2))/(sqrt(2 * pi) *
      sdlog))
  }
}
<environment: namespace:prob>
```

2.12 Negative Binomial Distribution: cfnbinom(t, size, prob, mu)

Let r and p denote the **size** and **prob** parameters, respectively. We may write the p.m.f. as

$$p_X(x) = \binom{r+x-1}{r-1} p^r (1-p)^x, \quad x = 0, 1, 2, \dots$$

The characteristic function is given by

$$\phi_X(t) = \left(\frac{p}{1 - (1-p)e^{it}} \right)^r.$$

Source Code:

```
function (t, size, prob, mu)
{
  if (size <= 0)
    stop("size must be positive")
  if (prob <= 0 || prob > 1)
    stop("prob must be in (0,1]")
  if (!missing(mu)) {
    if (!missing(prob))
      stop("'prob' and 'mu' both specified")
    prob <- size/(size + mu)
  }
  (prob/(1 - (1 - prob) * exp((0+1i) * t)))^size
}
<environment: namespace:prob>
```


2.13 Normal Distribution: `cfnorm(t, mean = 0, sd = 1)`

Let μ and σ denote the `mean` and `sd` parameters, respectively. The p.d.f. is

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}, \quad -\infty < x < \infty.$$

The characteristic function is given by

$$\phi_X(t) = e^{i\mu t + \sigma^2 t^2/2}.$$

Source Code:

```
function (t, mean = 0, sd = 1)
{
  if (sd <= 0)
    stop("sd must be positive")
  exp((0+1i) * mean - (sd * t)^2/2)
}
<environment: namespace:prob>
```

2.14 Poisson Distribution: `cfpois(t, lambda)`

Let λ denote the `lambda` parameter. The p.m.f. is

$$p_X(x) = e^{-\lambda} \frac{\lambda^x}{x!}, \quad x = 0, 1, 2, \dots$$

The characteristic function is given by

$$\phi_X(t) = \exp \{ \lambda (e^{it} - 1) \}.$$

Source Code:

```
function (t, lambda)
{
  if (lambda <= 0)
    stop("lambda must be positive")
  exp(lambda * (exp((0+1i) * t) - 1))
}
<environment: namespace:prob>
```

2.15 Wilcoxon Signed Rank Distribution: `cfsignrank(t, n)`

See `?dsignrank` for a discussion of the p.m.f. for this distribution; it is sufficient for our purposes to know that f_X is supported on the integers $x = 0, 1, \dots, n(n+1)/2$. Since the support is finite, we may calculate the characteristic function according to the definition:

$$\phi_X(t) = \sum_{x=0}^{n(n+1)/2} e^{itx} f_X(x),$$

where f_X is given by `dsignrank()`.

Source Code:

```
function (t, n)
{
  sum(exp((0+1i) * t * 0:((n + 1) * n/2)) * dsignrank(0:((n +
    1) * n/2), n))
}
<environment: namespace:prob>
```

2.16 Student's t Distribution: `cft(t, df, ncp)`

Let p denote the `df` parameter. The p.d.f. is

$$f_X(x) = \frac{\Gamma[(p+1)/2]}{\sqrt{p\pi}\Gamma(p/2)} \left(1 + \frac{x^2}{p}\right)^{-(p+1)/2}, \quad -\infty < x < \infty.$$

The formula used for the characteristic function was published by Hurst, see [12]. The characteristic function is given by

$$\phi_X(t) = \frac{K_{p/2}(\sqrt{p}|t|) \cdot (\sqrt{p}|t|)^{p/2}}{\Gamma(p/2)2^{p/2-1}},$$

where K_ν is the *modified Bessel Function of the second kind*, defined by

$$K_\nu(x) = \frac{\pi}{2} \frac{I_{-\nu}(x) - I_\nu(x)}{\sin(\nu\pi)},$$

and I_α is the *modified Bessel Function of the first kind*, defined by

$$I_\alpha(x) = i^{-\alpha} J_\alpha(ix),$$

with $J_\alpha(x)$ being a *Bessel function of the first kind*, defined by

$$J_\alpha(x) = \sum_{m=0}^{\infty} \frac{(-1)^m}{m!\Gamma(m+\alpha+1)} \left(\frac{x}{2}\right)^{2m+\alpha}.$$

Whew! See [6] in the References.

As of the time of this writing, it seems that we must resort to calculating the characteristic function for the noncentral Student's t by numerical integration according to the definition; see the source below. If you are aware of a way to more quickly/reliably calculate this c.f. with R, I would appreciate it if you would let me know.

Source Code:

```
function (t, df, ncp)
{
  if (missing(ncp))
    ncp <- 0
  if (df <= 0)
    stop("df must be positive")
  if (identical(all.equal(ncp, 0), TRUE)) {
    ifelse(identical(all.equal(t, 0), TRUE), 1 + (0+0i),
      as.complex(besselK(sqrt(df) * abs(t), df/2) * (sqrt(df) *
```

```

        abs(t))^(df/2)/(gamma(df/2) * 2^(df/2 - 1))))
    }
    else {
      fr <- function(x) cos(t * x) * dt(x, df, ncp)
      fi <- function(x) sin(t * x) * dt(x, df, ncp)
      Rp <- integrate(fr, lower = -Inf, upper = Inf)$value
      Ip <- integrate(fi, lower = -Inf, upper = Inf)$value
      return(Rp + (0+1i) * Ip)
    }
  }
}
<environment: namespace:prob>

```

2.17 Continuous Uniform Distribution: cfunif(t, min = 0, max = 1)

Let a and b denote the `min` and `max` parameters, respectively. The p.d.f. is

$$f_X(x) = \frac{1}{b-a}, \quad a < x < b.$$

The characteristic function is given by

$$\phi_X(t) = \frac{e^{itb} - e^{ita}}{(b-a)it}.$$

Source Code:

```

function (t, min = 0, max = 1)
{
  if (max < min)
    stop("min cannot be greater than max")
  ifelse(identical(all.equal(t, 0), TRUE), 1 + (0+0i), (exp((0+1i) *
    t * max) - exp((0+1i) * t * min))/((0+1i) * t * (max -
    min)))
}
<environment: namespace:prob>

```

2.18 Weibull Distribution: cfweibull(t, shape, scale = 1)

Let a and b denote the `shape` and `scale` parameters, respectively. The p.d.f. is

$$f_X(x) = \frac{a}{b} \left(\frac{x}{b}\right)^{a-1} e^{-(x/b)^a}, \quad 0 < x < \infty.$$

At the time of this writing, we must resort to calculating the characteristic function according to the definition; see the source below. If you know of a way to more quickly/reliably calculate this c.f. with R, I would appreciate it if you would let me know.

Source Code:

```

function (t, shape, scale = 1)
{
  if (shape <= 0 || scale <= 0)
    stop("shape and scale must be positive")
  fr <- function(x) cos(t * x) * dweibull(x, shape, scale)
}

```

```

    fi <- function(x) sin(t * x) * dweibull(x, shape, scale)
    Rp <- integrate(fr, lower = 0, upper = Inf)$value
    Ip <- integrate(fi, lower = 0, upper = Inf)$value
    return(Rp + (0+1i) * Ip)
}
<environment: namespace:prob>

```

2.19 Wilcoxon Rank Sum Distribution: cfwilcox(t, m, n)

See ?dwilcox for a discussion of the p.m.f. for this distribution; it is sufficient for our purposes to know that f_X is supported on the integers $x = 0, 1, \dots, mn$. Since the support is finite, we may calculate the characteristic function according to the definition:

$$\phi_X(t) = \sum_{x=0}^{mn} e^{itx} f_X(x),$$

where f_X is given by dwilcox().

Source Code:

```

function (t, m, n)
{
    sum(exp((0+1i) * t * 0:(m * n)) * dwilcox(0:(m * n), m, n))
}
<environment: namespace:prob>

```

3 R Session information

```
> toLatex(sessionInfo())
```

- R version 2.8.0 (2008-10-20), i486-pc-linux-gnu
- Locale: LC_CTYPE=en_US.UTF-8;LC_NUMERIC=C;LC_TIME=en_US.UTF-8;LC_COLLATE=en_US.UTF-8;LC_MONETARY=C;LC_PAPER=en_US.UTF-8;LC_NAME=C;LC_ADDRESS=C;LC_TELEPHONE=C;LC_MEASUREMENT=en_US.UTF-8;LC_IDENTIFICATION=C
- Base packages: base, datasets, graphics, grDevices, methods, stats, tcltk, utils
- Other packages: prob 0.9-2, svGUI 0.9-43, svMisc 0.9-45, svSocket 0.9-42

References

- [1] http://en.wikipedia.org/wiki/Confluent_hypergeometric_function
- [2] Abramowitz, Milton; Stegun, Irene A., eds. (1965) *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, New York: Dover
- [3] http://en.wikipedia.org/wiki/Hypergeometric_series
- [4] http://en.wikipedia.org/wiki/Hyperbolic_function
- [5] <http://anziamj.austms.org.au/V32/part3/Leipnik.html>

- [6] http://en.wikipedia.org/wiki/Bessel_function
- [7] Lukacs, E. (1970). *Characteristic Functions*, Second Edition. London: Griffin.
- [8] Johnson, N. L., Kotz, S., and Kemp, A. W. (1992) *Univariate Discrete Distributions*, Second Edition. New York: Wiley.
- [9] Johnson, N. L., Kotz, S. and Balakrishnan, N. (1995) *Continuous Univariate Distributions*, volume 1. Wiley, New York.
- [10] Johnson, N. L., Kotz, S. and Balakrishnan, N. (1995) *Continuous Univariate Distributions*, volume 2. Wiley, New York.
- [11] Beaulieu, N.C. (2008) Fast convenient numerical computation of lognormal characteristic functions, *IEEE Transactions on Communications*, **56** (3): 331–333.
- [12] Hurst, S. (1995) The Characteristic Function of the Student-t Distribution, Financial Mathematics Research Report No. FMRR006-95, Statistics Research Report No. SRR044-95, available online: <http://www.maths.anu.edu.au/research.reports/srr/95/044/>