

Vector, line and significance plots

cfp.vect - vector plots

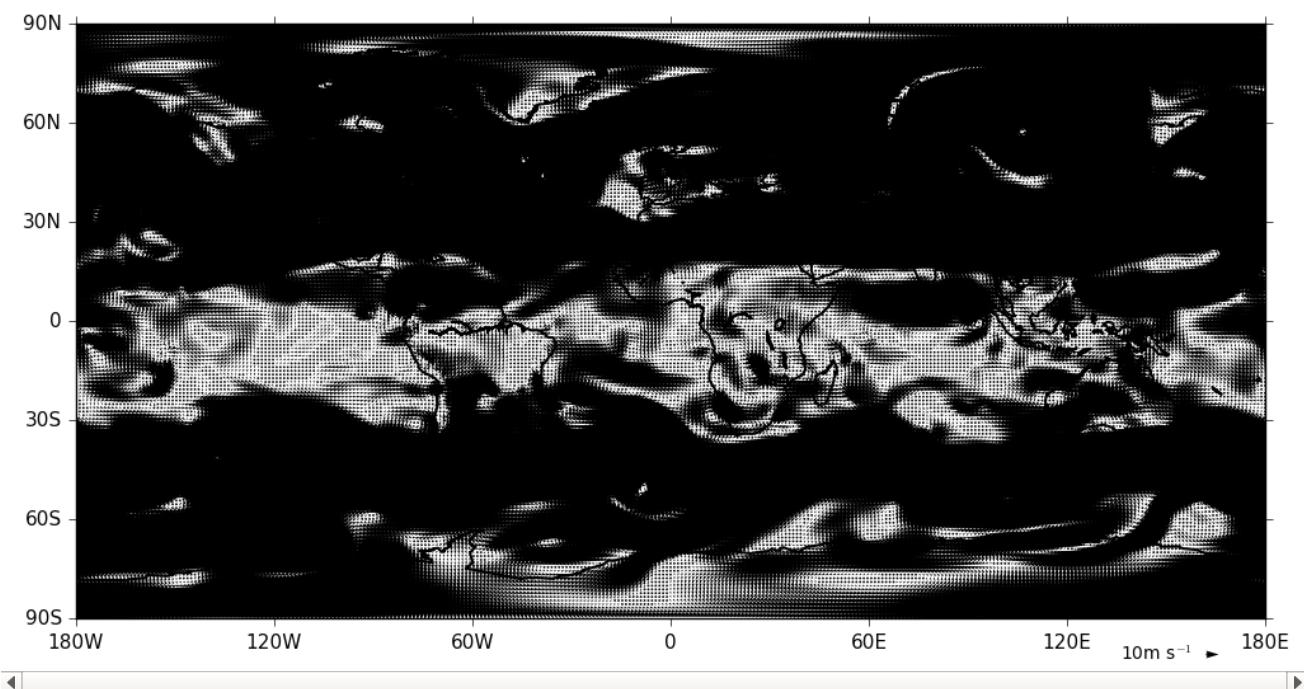
In [1]:

```
# Inline images in Ipython Notebook - not needed in Python
%matplotlib inline

# Import cf-python and cf-plot packages
import cf, cfplot as cfp
```

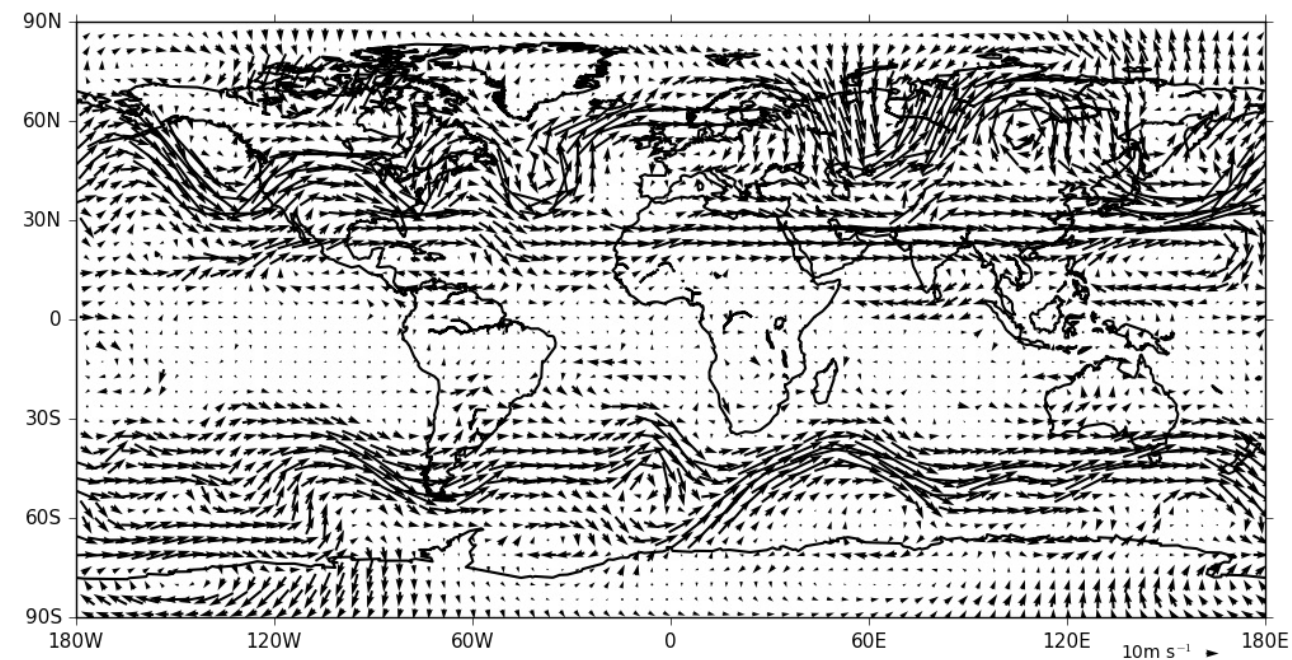
In [2]:

```
# Select u and v wind components at 500mb and make a vector plot
f=cf.read('ncas_data/data1.nc')
u=f[7].subspace(pressure=500)
v=f[9].subspace(pressure=500)
cfp.vect(u=u, v=v, key_length=10, scale=100)
```



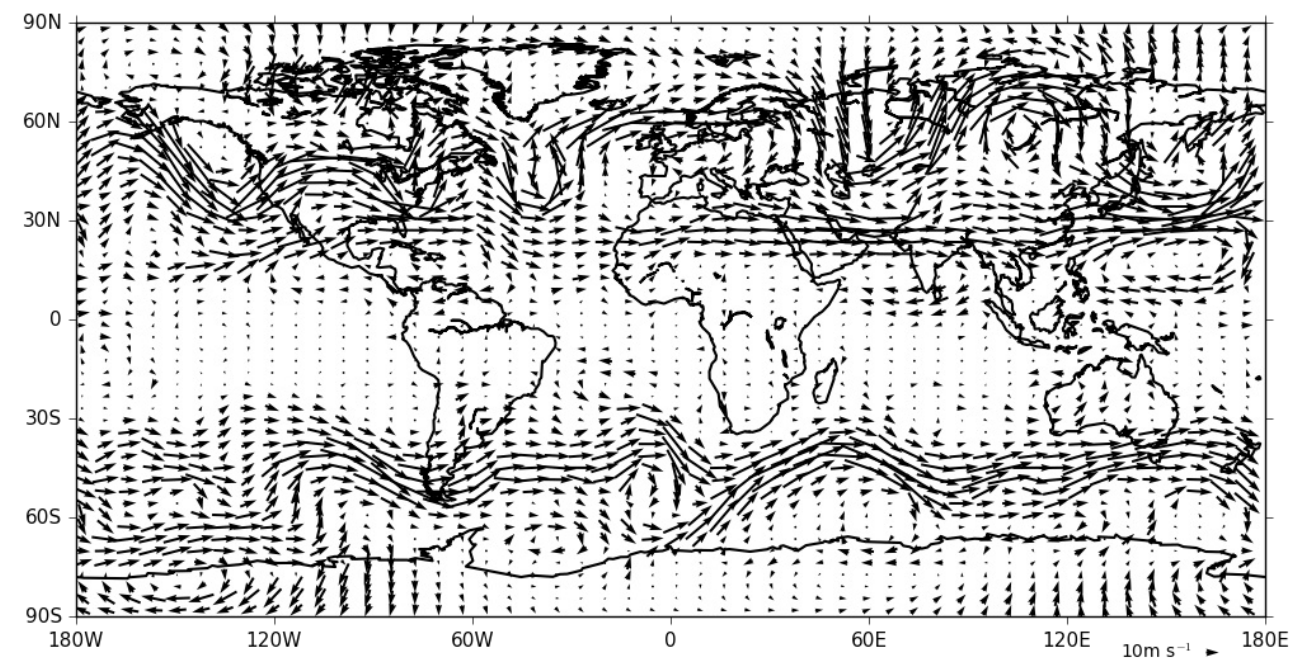
In [3]:

```
# In the example above we have too many points for the vectors to be discernable  
# We can use a stride of 4 in plotting the vectors to thin out the vectors  
cfp.vect(u=u, v=v, key_length=10, scale=100, stride=4)
```



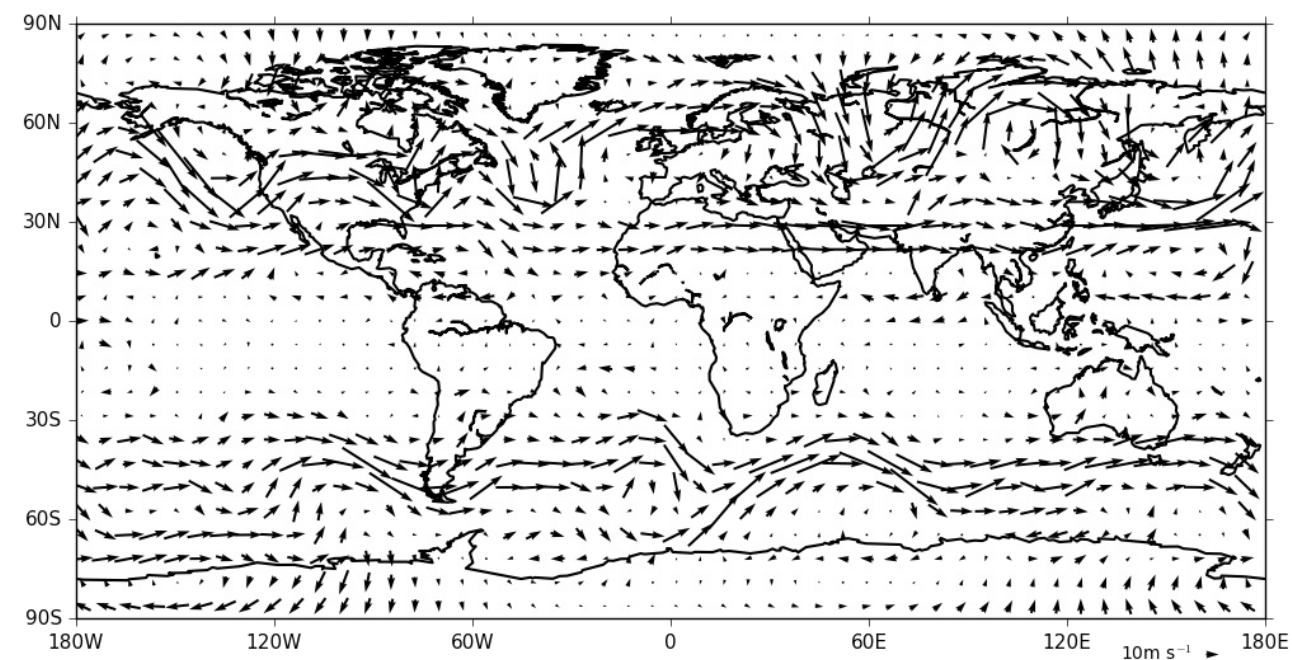
In [4]:

```
# The pts parameter controls the interpolation of the vectors to a new grid  
# One value will give the same number of points in both directions  
cfp.vect(u=u, v=v, key_length=10, scale=100, pts=50)
```



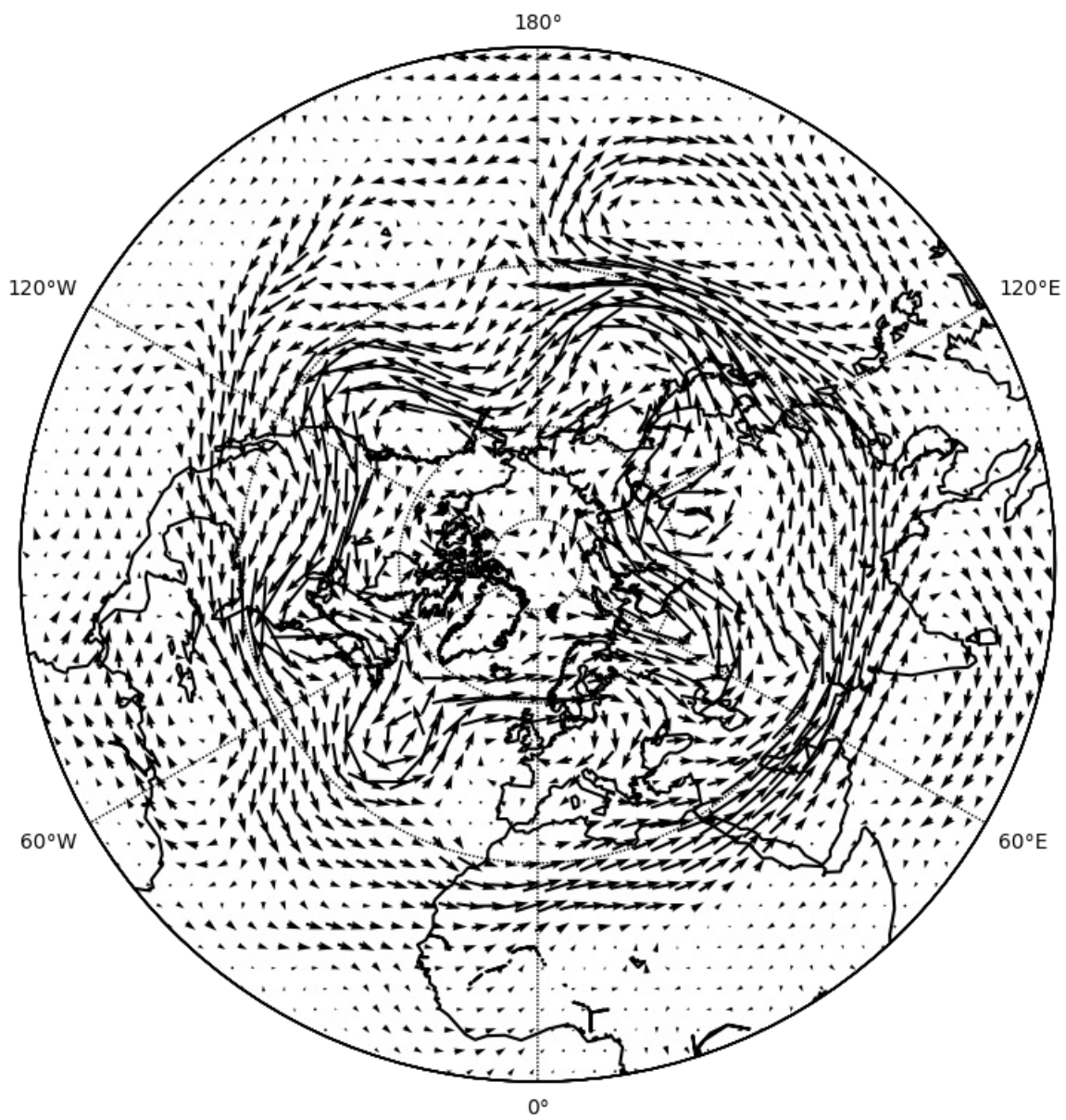
In [5]:

```
# Two values gives different nuymbers of points in each direction
cfp.vect(u=u, v=v, key_length=10, scale=100, pts=[50, 25])
```



In [6]:

```
# When making polar stereographic plots use the pts keyword to cfp.vect
# to specify the number of interpolated points in x and y
cfp.mapset(proj='npstere')
cfp.vect(u=u, v=v, key_length=10, scale=100, pts=50)
```

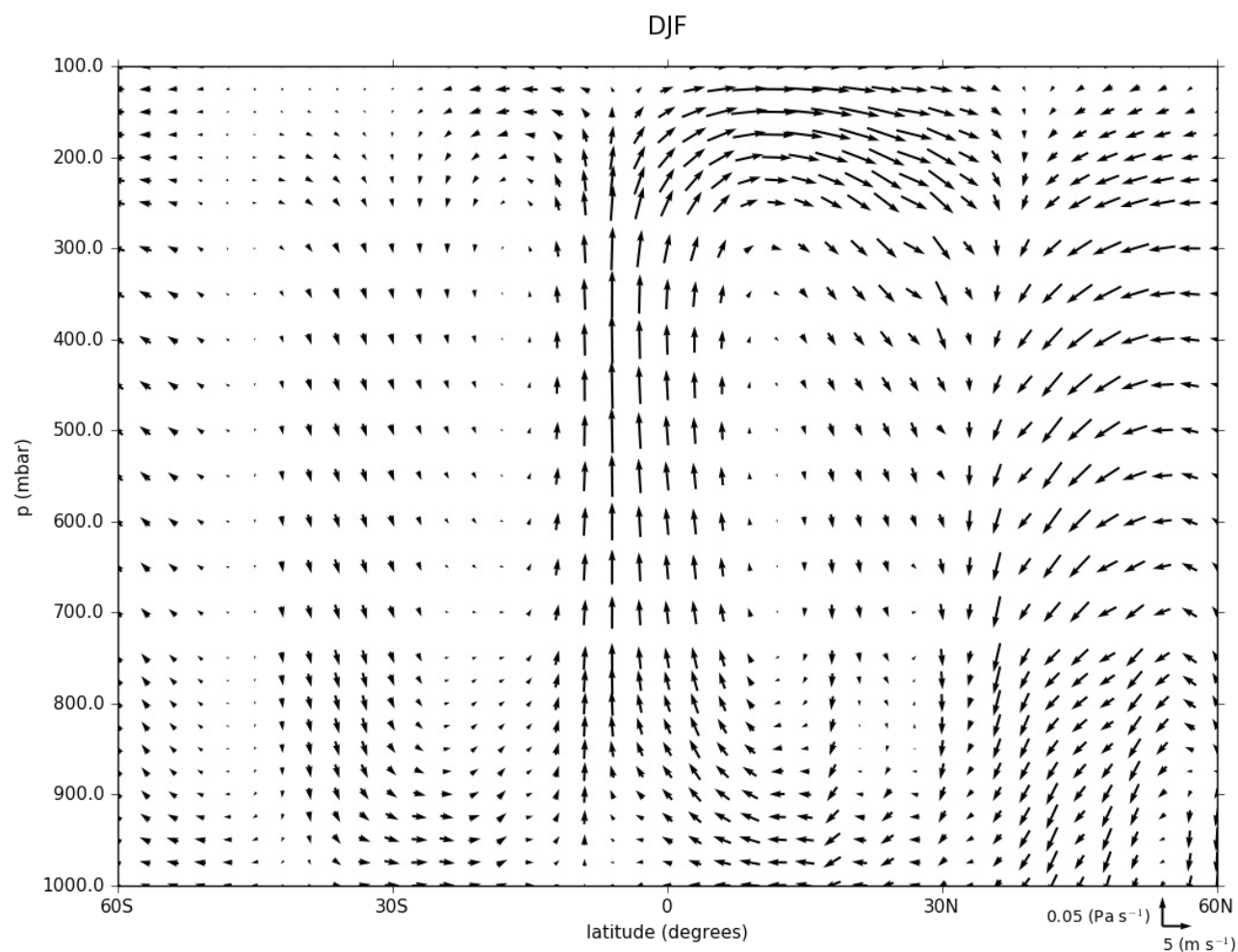


In [7]:

```
# Vectors can have different lengths and scales as in the example below
c=cf.read('ncas_data/vaAMIPlcd_DJF.nc')[0]
c=c.subspace(Y=cf.wi(-60,60))
c=c.subspace(X=cf.wi(80,160))
c=c.collapse('T: mean X: mean')

g=cf.read('ncas_data/wapAMIPlcd_DJF.nc')[0]
g=g.subspace(Y=cf.wi(-60,60))
g=g.subspace(X=cf.wi(80,160))
g=g.collapse('T: mean X: mean')

cfp.vect(u=c, v=-g, key_length=[5, 0.05], scale=[20,0.2], title='DJF', key_location=[0.95, -0.05])
```



cfp-lineplot - making line plots

In [8]:

```
# Read in some temerature data and convert to Celsius
f = cf.read('ncas_data/data1.nc')[2]
f = f.collapse('mean','longitude')
f.Units -= 273.15
```

In [9]:

```
# Reset the plotting limits
cfp.gset()
```

In [10]:

```
f.item('latitude').array
```

Out[10]:

```
array([ 89.14151764,  88.02942657,  86.91077423,  85.79062653,
        84.66992188,  83.5489502 ,  82.4278183 ,  81.30659485,
        80.18531036,  79.0639801 ,  77.94262695,  76.82124329,
        75.69984436,  74.57843018,  73.45700836,  72.33557892,
        71.21413422,  70.09268951,  68.97123718,  67.84978485,
        66.72832489,  65.60686493,  64.48539734,  63.36393356,
        62.24246216,  61.12099075,  59.99951935,  58.87804413,
        57.75656891,  56.63509369,  55.51361465,  54.39213562,
        53.27065659,  52.14917374,  51.0276947 ,  49.90621185,
        48.784729 ,  47.66324615,  46.54176331,  45.42028046,
        44.29879379,  43.17731094,  42.05582428,  40.93433762,
        39.81285095,  38.69136429,  37.56988144,  36.44839096,
        35.3269043 ,  34.20541763,  33.08393097,  31.96244431,
        30.84095573,  29.71946716,  28.59797859,  27.47649002,
        26.35500336,  25.23351479,  24.11202431,  22.99053574,
        21.86904716,  20.74755859,  19.62607002,  18.50457954,
        17.38309097,  16.26160049,  15.14011192,  14.0186224 ,
        12.89713287,  11.77564335,  10.65415287,   9.53266335,
         8.41117382,   7.2896843 ,   6.16819429,   5.04670429,
         3.92521453,   2.80372477,   1.68223488,   0.56074494,
        -0.56074494,  -1.68223488,  -2.80372477,  -3.92521453,
        -5.04670429,  -6.16819429,  -7.2896843 ,  -8.41117382,
        -9.53266335, -10.65415287, -11.77564335, -12.89713287,
       -14.0186224 , -15.14011192, -16.26160049, -17.38309097,
       -18.50457954, -19.62607002, -20.74755859, -21.86904716,
       -22.99053574, -24.11202431, -25.23351479, -26.35500336,
       -27.47649002, -28.59797859, -29.71946716, -30.84095573,
       -31.96244431, -33.08393097, -34.20541763, -35.3269043 ,
       -36.44839096, -37.56988144, -38.69136429, -39.81285095,
       -40.93433762, -42.05582428, -43.17731094, -44.29879379,
       -45.42028046, -46.54176331, -47.66324615, -48.784729 ,
       -49.90621185, -51.0276947 , -52.14917374, -53.27065659,
       -54.39213562, -55.51361465, -56.63509369, -57.75656891,
       -58.87804413, -59.99951935, -61.12099075, -62.24246216,
       -63.36393356, -64.48539734, -65.60686493, -66.72832489,
       -67.84978485, -68.97123718, -70.09268951, -71.21413422,
       -72.33557892, -73.45700836, -74.57843018, -75.69984436,
       -76.82124329, -77.94262695, -79.0639801 , -80.18531036,
       -81.30659485, -82.4278183 , -83.5489502 , -84.66992188,
       -85.79062653, -86.91077423, -88.02942657, -89.14151764], dtype=float32)
```

In [13]:

```
# We cannot select on one of these values as they are Gaussian latitudes
# and numeric representation rounding comes into effect
g=f.subspace(latitude=0.56074494)
```

```
-----
IndexError                                Traceback (most recent call last)
<ipython-input-13-15b8e5633c3b> in <module>()
      1 # We cannot select on one of these values as they are Gaussian latitudes
      2 # and numeric representation rounding comes into effect
----> 3 g=f.subspace(latitude=0.56074494)
      4

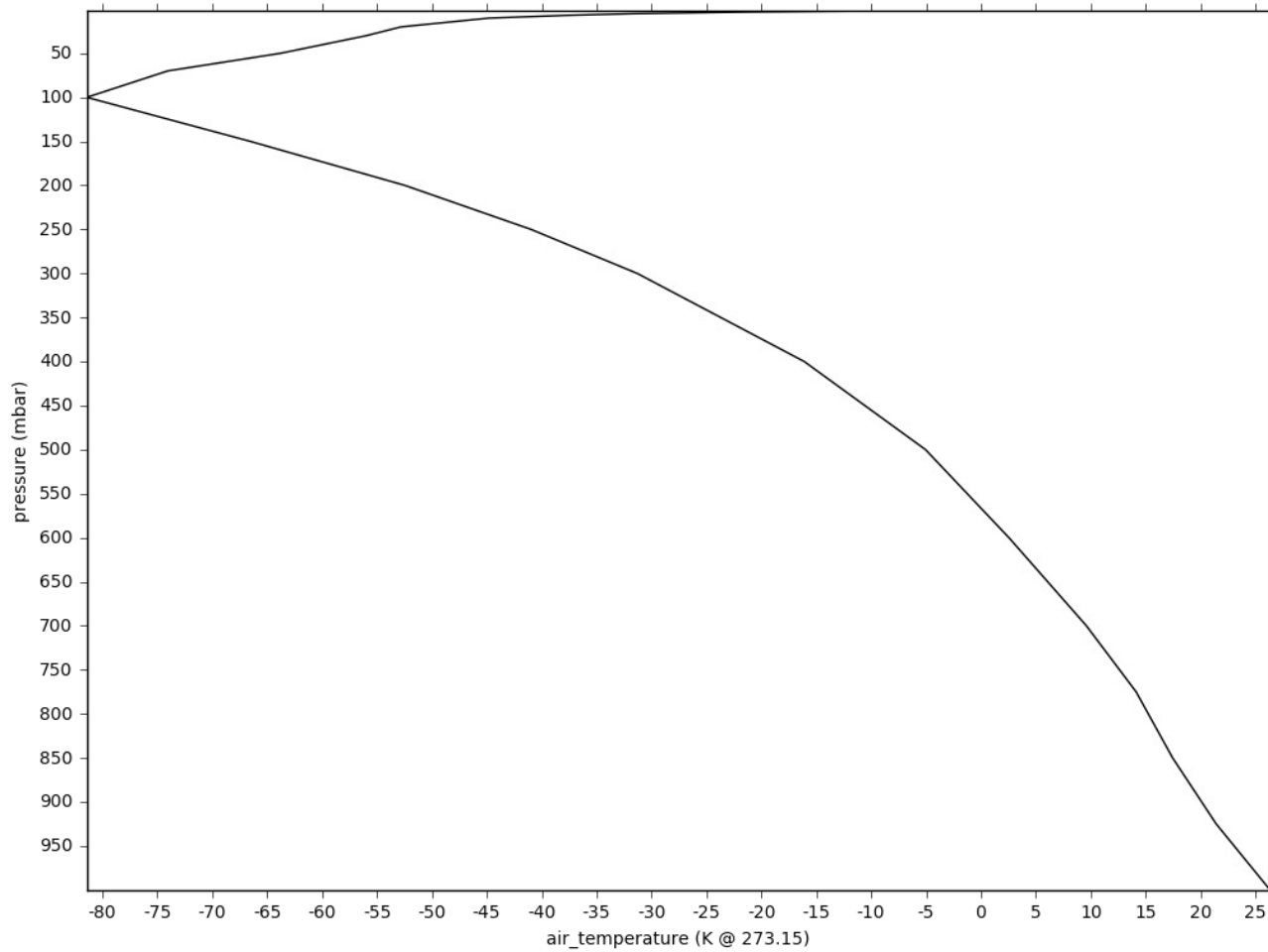
/opt/tools/ncas/anaconda2/lib/python2.7/site-packages/cf/field.pyc in __call__(self, *args, **k
wargs)
    15664         return field.copy()
    15665
> 15666         return field[field.indices(*args, **kwargs)]
    15667     #--- End: def
    15668

/opt/tools/ncas/anaconda2/lib/python2.7/site-packages/cf/field.pyc in indices(self, *args, **kw
args)
    8090         if not item_match.any():
    8091             raise IndexError(
-> 8092 "No {!r} axis indices found from: {}".format(identity, value))
    8093
    8094         index = numpy_asanyarray(item_match)
```

IndexError: No 'latitude' axis indices found from: 0.56074494

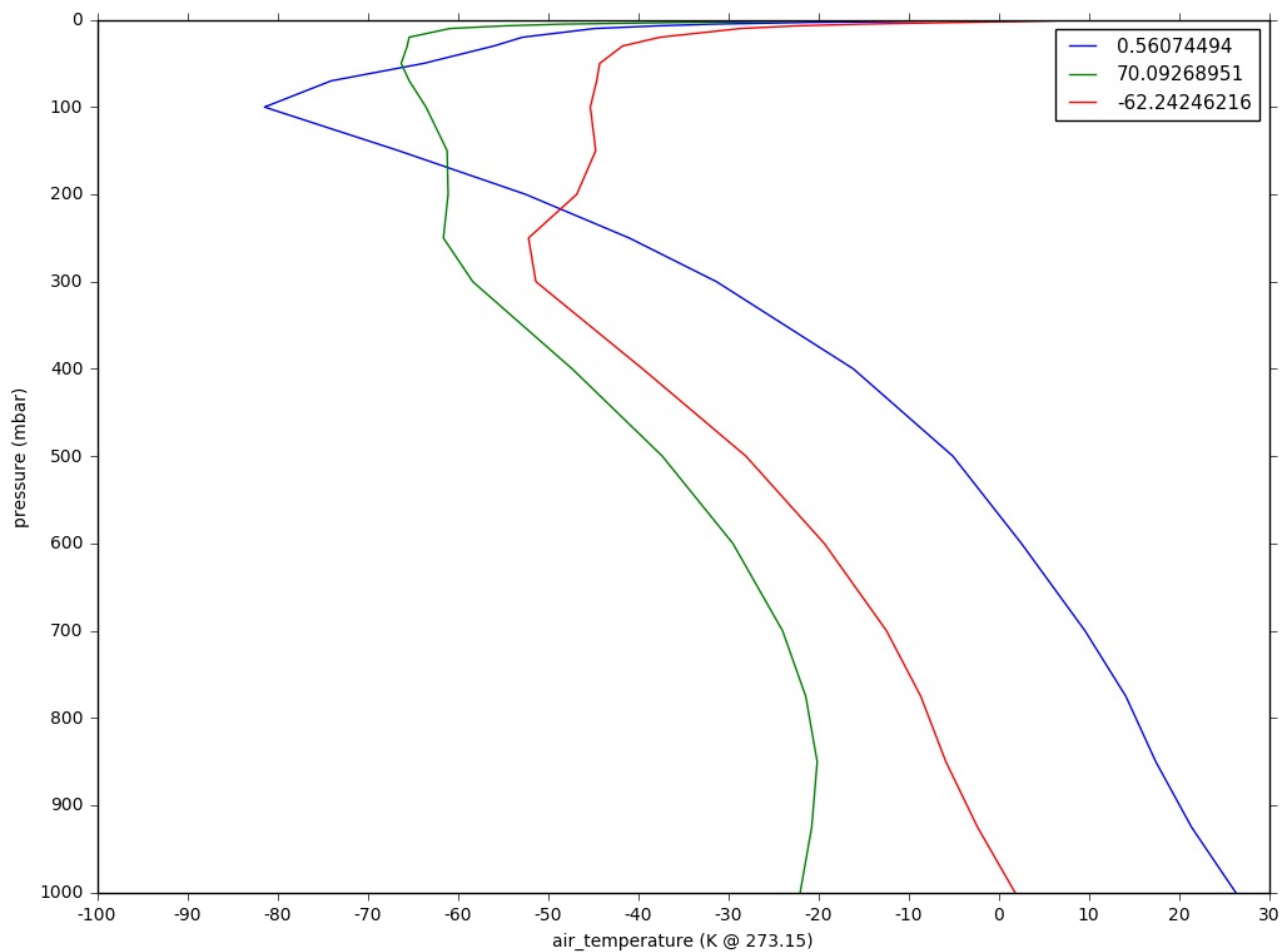
In [15]:

```
# Set a lower matching tolerance value  
tol=cf.RTOL(1e-5)  
g=f.subspace(latitude=0.56074494)  
cfp.lineplot(g)
```



In [16]:

```
cfp.gopen()
cfp.gset(xmin=-100,xmax=30, ymin=1000, ymax=0)
yticks=[1000, 900,800,700, 600,500,400,300,200,100,0]
cfp.lineplot(f.subspace(latitude=0.56074494), label='0.56074494', color='b')
cfp.lineplot(f.subspace(latitude=70.09268951), label='70.09268951', color='g')
cfp.lineplot(f.subspace(latitude=-62.24246216), label='-62.24246216', color='r', yticks=yticks)
cfp.gclose()
```

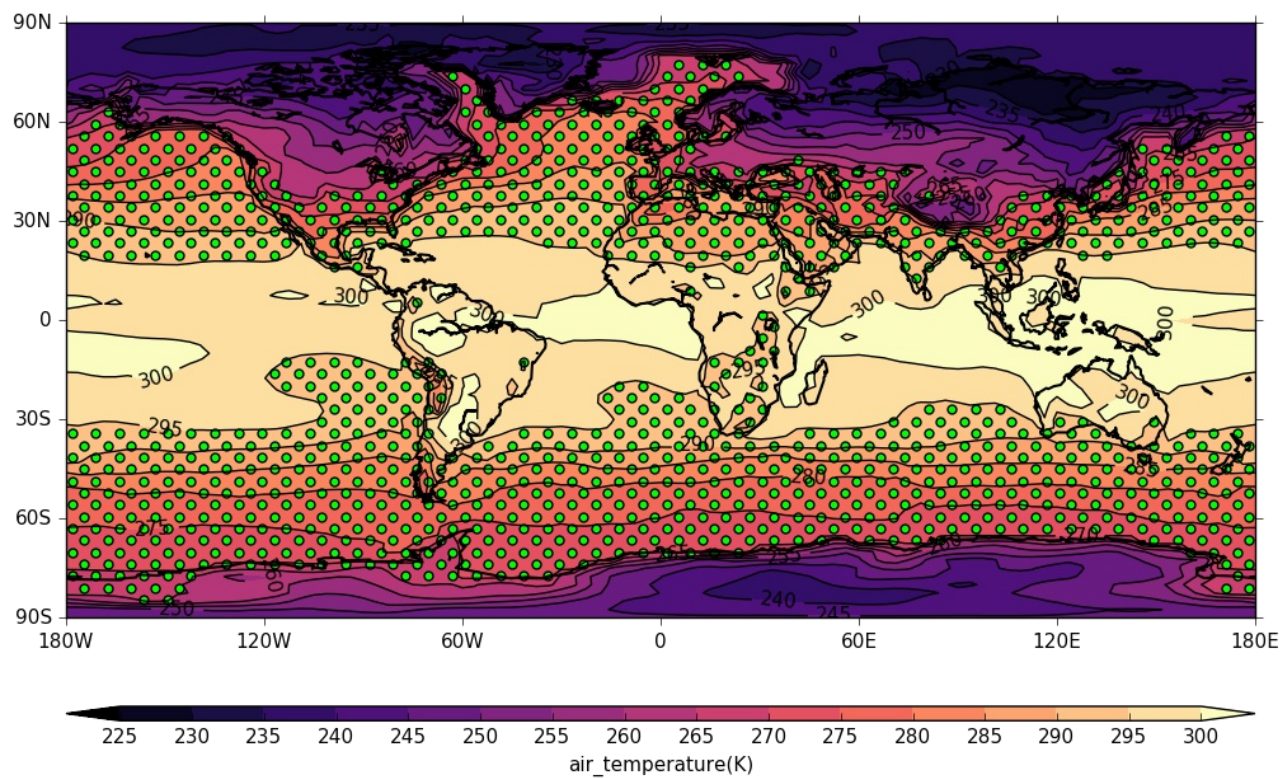


cfp.stipple - Significance plots

plotting areas of significance with coloured symbols

In [17]:

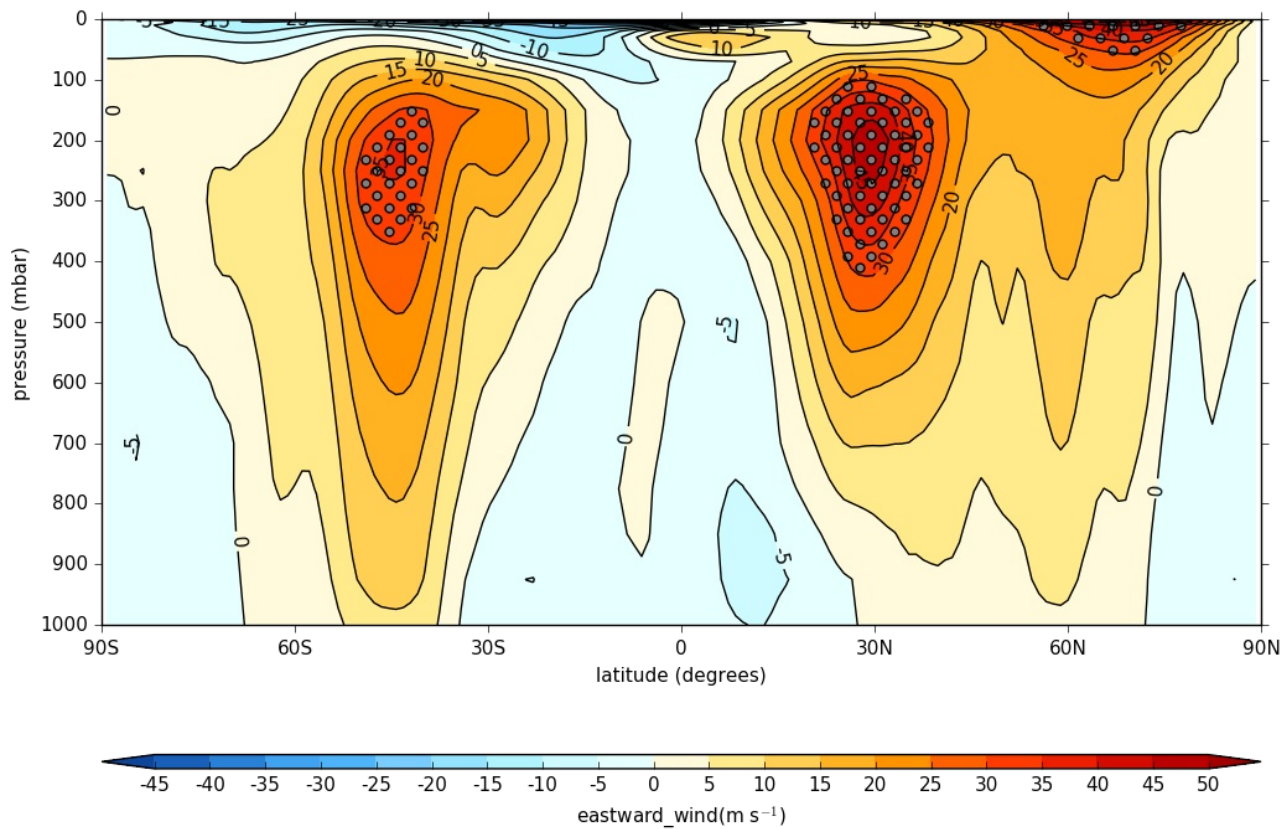
```
cfp.mapset()
f=cf.read('ncas_data/data4.nc')[0]
g=f.subspace(time=15)
cfp.gopen()
cfp.cscale('magma')
cfp.con(g)
cfp.stipple(f=g, min=265, max=295, size=100, color='#00ff00')
cfp.gclose()
```



In [19]:

```
cfp.gopen()
cfp.cscale()
f=cf.read('/opt/graphics/cfplot_data/ggap.nc')[7]
g=f.collapse('mean','longitude')
cfp.con(g)
cfp.stipple(f=g, min=30, max=100, size=100, color='grey')
cfp.gclose()
```

plotvars.plot_type is 2
flipping Y



In []: