PlotExample_Single_Run

November 10, 2020

1 AerVis Single - Plot example

This notebook shows how to compare results of multiple timesteps..

[1]:
```
# we begin by loading the aervis plotting module
import aervis.plotting as ap
```

```
AerVis 0.0.2
:: AerVis plotting imports complete ::
```

1.1 Selecting a dataset

Next we can load an individual dataset by supplying the full path of our converted netCDF file into the AerData class. If your data is still in the pp file format, please refer to the covert tutorials first.

[2]:
```
dataset = ap.AerData('../../bk417a.nc')
```

1.2 Selecting a variable (and getting help on functions)

Now we have a dataset, we want to chose which variables to plot. To do this we use the `.show_var` function of our class. To see what this we can use pythons `help` function

[3]:
```
help(dataset.show_var)
```

```
Help on method show_var in module aervis.plotting:

show_var(screen: bool = True, returnstr: bool = False) method of
aervis.plotting.AerData instance
    Show variables within the Dataset
    If available this includes the standard/long name and dims

    Arguments:
        screen: bool - print to screen
        returnstr: bool - returns the string for saving to a file..
```

[4]:
```
dataset.show_var()
```

```
   latitude - ('latitude',) - Latitude
  longitude - ('longitude',) - Longitude
pseudo_level - ('pseudo_level',) - pseudo_level
model_level_number - ('model_level_number',) - Model Level Number
       time - ('time',) - Time
forecast_reference_time - () - Forecast Reference Time
forecast_period - ('time',) - Forecast Period
 m01s01i245 - ('time', 'latitude', 'longitude') - product_of_effective_radius_of
_cloud_liquid_water_particle_and_cloud_liquid_water_area_fraction_exposed_to_spa
ce_and_sunlit_binary_mask
 m01s01i246 - ('time', 'latitude', 'longitude') - product_of_cloud_liquid_water_
area_fraction_exposed_to_space_and_sunlit_binary_mask
 m01s01i254 - ('time', 'latitude', 'longitude') - product_of_effective_radius_of
_warm_cloud_liquid_water_particle_and_warm_cloud_liquid_water_area_fraction_expo
sed_to_space_and_sunlit_binary_mask
 m01s01i255 - ('time', 'latitude', 'longitude') - product_of_warm_cloud_liquid_w
ater_area_fraction_exposed_to_space_and_sunlit_binary_mask
 m01s01i260 - ('time', 'latitude', 'longitude') -
 m01s01i270 - ('pseudo_level', 'time', 'latitude', 'longitude') - scaling_factor
_for_surface_diffuse_albedo_of_photosynthetically_active_radiation_assuming_no_s
now
 m01s01i271 - ('pseudo_level', 'time', 'latitude', 'longitude') - scaling_factor
_for_surface_diffuse_albedo_of_near_infra_red_radiation_assuming_no_snow
 m01s01i280 - ('time', 'latitude', 'longitude') - product_of_atmosphere_number_c
ontent_of_cloud_droplets_and_warm_cloud_area_fraction_and_sunlit_binary_mask
 m01s01i281 - ('time', 'latitude', 'longitude') -
product_of_warm_cloud_area_fraction_and_sunlit_binary_mask
 m01s01i298 - ('time', 'latitude', 'longitude') -
 m01s01i299 - ('time', 'latitude', 'longitude') -
 m01s01i501 - ('time', 'latitude', 'longitude') -
 m01s01i503 - ('time', 'latitude', 'longitude') -
 m01s01i505 - ('time', 'latitude', 'longitude') -
 m01s02i240 - ('pseudo_level', 'time', 'latitude', 'longitude') -
 m01s02i241 - ('pseudo_level', 'time', 'latitude', 'longitude') -
 m01s02i242 - ('pseudo_level', 'time', 'latitude', 'longitude') -
 m01s02i243 - ('pseudo_level', 'time', 'latitude', 'longitude') -
 m01s02i251 - ('pseudo_level', 'time', 'latitude', 'longitude') -
 m01s02i252 - ('pseudo_level', 'time', 'latitude', 'longitude') -
 m01s02i253 - ('pseudo_level', 'time', 'latitude', 'longitude') -
 m01s02i254 - ('pseudo_level', 'time', 'latitude', 'longitude') -
 m01s02i300 - ('pseudo_level', 'time', 'latitude', 'longitude') -
atmosphere_optical_thickness_due_to_soluble_aitken_mode_sulphate_aerosol
 m01s02i301 - ('pseudo_level', 'time', 'latitude', 'longitude') -
atmosphere_optical_thickness_due_to_soluble_accumulation_mode_sulphate_aerosol
 m01s02i302 - ('pseudo_level', 'time', 'latitude', 'longitude') -
```

```
atmosphere_optical_thickness_due_to_soluble_coarse_mode_sulphate_aerosol
 m01s02i303 - ('pseudo_level', 'time', 'latitude', 'longitude') -
atmosphere_optical_thickness_due_to_insoluble_aitken_mode_sulphate_aerosol
 m01s02i501 - ('time', 'latitude', 'longitude') -
 m01s03i231 - ('time', 'latitude', 'longitude') -
water_sublimation_flux_in_timestep
 m01s03i232 - ('time', 'latitude', 'longitude') - Evaporation flux from open sea
 m01s03i235 - ('time', 'latitude', 'longitude') - Latent heat flux from sea ice
top melt
 m01s03i304 - ('time', 'latitude', 'longitude') - Turbulent mixing height after
boundary layer
 m01s03i305 - ('time', 'latitude', 'longitude') - Stable boundary layer
indicator
 m01s03i306 - ('time', 'latitude', 'longitude') - Stratocumulus over stable
boundary layer indicator
 m01s03i307 - ('time', 'latitude', 'longitude') - Well-mixed boundary layer
indicator
 m01s03i308 - ('time', 'latitude', 'longitude') - Decoupled stratocumulus not
over cumulus indicator
 m01s03i309 - ('time', 'latitude', 'longitude') - Decoupled stratocumulus over
cumulus indicator
 m01s03i310 - ('time', 'latitude', 'longitude') - Cumulus capped boundary layer
indicator
 m01s03i328 - ('pseudo_level', 'time', 'latitude', 'longitude') -
 m01s03i331 - ('pseudo_level', 'time', 'latitude', 'longitude') - Sublimation
moisture flux on tiles
 m01s03i340 - ('time', 'latitude', 'longitude') - Shear driven boundary layer
indicator
 m01s03i367 - ('time', 'latitude', 'longitude') -
 m01s03i370 - ('time', 'latitude', 'longitude') -
 m01s03i371 - ('time', 'latitude', 'longitude') -
 m01s03i430 - ('pseudo_level', 'time', 'latitude', 'longitude') - Dust friction
velocity
 m01s05i231 - ('time', 'latitude', 'longitude') - Cape time scale (deep)
 m01s05i232 - ('time', 'latitude', 'longitude') - reduced cape time scale
indicator
 m01s05i269 - ('time', 'latitude', 'longitude') - deep convection indicator
 m01s05i270 - ('time', 'latitude', 'longitude') - shallow convection indicator
 m01s05i272 - ('time', 'latitude', 'longitude') - mid level convection indicator
 m01s05i277 - ('time', 'latitude', 'longitude') - deep convective precipitation
rate
 m01s05i278 - ('time', 'latitude', 'longitude') - shallow convective
precipitation rate
 m01s05i279 - ('time', 'latitude', 'longitude') - mid level convective
precipitation rate
 m01s30i403 - ('time', 'latitude', 'longitude') -
 m01s30i419 - ('time', 'latitude', 'longitude') -
 m01s30i428 - ('time', 'latitude', 'longitude') -
```

```
 m01s30i429 - ('time', 'latitude', 'longitude') -
 m01s30i431 - ('time', 'latitude', 'longitude') -
 m01s00i096 - ('time', 'latitude', 'longitude') -
 m01s00i509 - ('time', 'latitude', 'longitude') -
product_of_sea_ice_albedo_and_sunlit_binary_mask
 m01s50i211 - ('time', 'latitude', 'longitude') -
 m01s50i214 - ('time', 'latitude', 'longitude') -
 m01s50i215 - ('time', 'latitude', 'longitude') -
 m01s50i216 - ('time', 'latitude', 'longitude') -
level_height - ('model_level_number',) - level_height
      sigma - ('model_level_number',) - sigma
surface_altitude - ('latitude', 'longitude') - Surface Altitude
   altitude - ('model_level_number', 'latitude', 'longitude') - Altitude
 m01s01i223 - ('time', 'model_level_number', 'latitude', 'longitude') -
product_of_stratiform_cloud_liquid_water_area_fraction_and_sunlit_binary_mask
 m01s01i225 - ('time', 'model_level_number', 'latitude', 'longitude') - product_
of_effective_radius_of_convective_cloud_liquid_water_particle_and_convective_clo
ud_liquid_water_area_fraction_and_sunlit_binary_mask
 m01s01i226 - ('time', 'model_level_number', 'latitude', 'longitude') -
product_of_convective_cloud_liquid_water_area_fraction_and_sunlit_binary_mask
 m01s01i241 - ('time', 'model_level_number', 'latitude', 'longitude') - product_
of_number_concentration_of_stratiform_cloud_liquid_water_particles_and_stratifor
m_cloud_liquid_water_area_fraction_and_sunlit_binary_mask
 m01s01i242 - ('time', 'model_level_number', 'latitude', 'longitude') - product_
of_stratiform_cloud_liquid_water_content_and_stratiform_cloud_liquid_water_area_
fraction_and_sunlit_binary_mask
 m01s03i219 - ('time', 'model_level_number', 'latitude', 'longitude') -
atmosphere_downward_eastward_stress
 m01s03i220 - ('time', 'model_level_number', 'latitude', 'longitude') -
atmosphere_downward_northward_stress
 m01s05i319 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s05i320 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s05i322 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s05i323 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s30i112 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s30i114 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s34i075 - ('time', 'model_level_number', 'latitude', 'longitude') -
mass_fraction_of_dimethyl_sulfoxide
 m01s34i091 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s34i092 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s34i101 - ('time', 'model_level_number', 'latitude', 'longitude') -
number_of_particles_per_air_molecule_of_soluble_nucleation_mode_aerosol_in_air
 m01s34i102 - ('time', 'model_level_number', 'latitude', 'longitude') -
mass_fraction_of_sulfuric_acid_in_soluble_nucleation_mode_dry_aerosol_in_air
 m01s34i103 - ('time', 'model_level_number', 'latitude', 'longitude') -
number_of_particles_per_air_molecule_of_soluble_aitken_mode_aerosol_in_air
 m01s34i104 - ('time', 'model_level_number', 'latitude', 'longitude') -
mass_fraction_of_sulfuric_acid_in_soluble_aitken_mode_dry_aerosol_in_air
```

```
 m01s34i105 - ('time', 'model_level_number', 'latitude', 'longitude') -
mass_fraction_of_black_carbon_in_soluble_aitken_mode_dry_aerosol_in_air
 m01s34i106 - ('time', 'model_level_number', 'latitude', 'longitude') - mass_fra
ction_of_particulate_organic_matter_in_soluble_aitken_mode_dry_aerosol_in_air
 m01s34i107 - ('time', 'model_level_number', 'latitude', 'longitude') -
number_of_particles_per_air_molecule_of_soluble_accumulation_mode_aerosol_in_air
 m01s34i108 - ('time', 'model_level_number', 'latitude', 'longitude') -
mass_fraction_of_sulfuric_acid_in_soluble_accumulation_mode_dry_aerosol_in_air
 m01s34i109 - ('time', 'model_level_number', 'latitude', 'longitude') -
mass_fraction_of_black_carbon_in_soluble_accumulation_mode_dry_aerosol_in_air
 m01s34i110 - ('time', 'model_level_number', 'latitude', 'longitude') - mass_fra
ction_of_particulate_organic_matter_in_soluble_accumulation_mode_dry_aerosol_in_
air
 m01s34i111 - ('time', 'model_level_number', 'latitude', 'longitude') -
mass_fraction_of_seasalt_in_soluble_accumulation_mode_dry_aerosol_in_air
 m01s34i113 - ('time', 'model_level_number', 'latitude', 'longitude') -
number_of_particles_per_air_molecule_of_soluble_coarse_mode_aerosol_in_air
 m01s34i114 - ('time', 'model_level_number', 'latitude', 'longitude') -
mass_fraction_of_sulfuric_acid_in_soluble_coarse_mode_dry_aerosol_in_air
 m01s34i115 - ('time', 'model_level_number', 'latitude', 'longitude') -
mass_fraction_of_black_carbon_in_soluble_coarse_mode_dry_aerosol_in_air
 m01s34i116 - ('time', 'model_level_number', 'latitude', 'longitude') - mass_fra
ction_of_particulate_organic_matter_in_soluble_coarse_mode_dry_aerosol_in_air
 m01s34i117 - ('time', 'model_level_number', 'latitude', 'longitude') -
mass_fraction_of_seasalt_in_soluble_coarse_mode_dry_aerosol_in_air
 m01s34i119 - ('time', 'model_level_number', 'latitude', 'longitude') -
number_of_particles_per_air_molecule_of_insoluble_aitken_mode_aerosol_in_air
 m01s34i120 - ('time', 'model_level_number', 'latitude', 'longitude') -
mass_fraction_of_black_carbon_in_insoluble_aitken_mode_dry_aerosol_in_air
 m01s34i121 - ('time', 'model_level_number', 'latitude', 'longitude') - mass_fra
ction_of_particulate_organic_matter_in_insoluble_aitken_mode_dry_aerosol_in_air
 m01s34i126 - ('time', 'model_level_number', 'latitude', 'longitude') - mass_fra
ction_of_particulate_organic_matter_in_soluble_nucleation_mode_dry_aerosol_in_ai
r
 m01s34i966 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s34i967 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s34i968 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s38i201 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s38i202 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s38i203 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s38i204 - ('time', 'latitude', 'longitude') -
 m01s38i205 - ('time', 'latitude', 'longitude') -
 m01s38i207 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s38i208 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s38i209 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s50i061 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s50i062 - ('time', 'model_level_number', 'latitude', 'longitude') -
 m01s50i063 - ('time', 'model_level_number', 'latitude', 'longitude') -
```

```
m01s50i217 - ('time', 'model_level_number', 'latitude', 'longitude') -
m01s34i150 - ('time', 'model_level_number', 'latitude', 'longitude') -
m01s00i407 - ('time', 'model_level_number', 'latitude', 'longitude') -
m01s05i209 - ('time', 'latitude', 'longitude') -
m01s16i004 - ('time', 'model_level_number', 'latitude', 'longitude') -
m01s00i025 - ('time', 'latitude', 'longitude') -
m01s30i406 - ('time', 'latitude', 'longitude') -
m01s30i405 - ('time', 'latitude', 'longitude') -
m01s06i201 - ('time', 'model_level_number', 'latitude', 'longitude') -
m01s30i404 - ('time', 'latitude', 'longitude') -
m01s02i285 - ('pseudo_level', 'time', 'latitude', 'longitude') -
m01s02i422 - ('pseudo_level', 'time', 'latitude', 'longitude') -
m01s02i204 - ('time', 'latitude', 'longitude') -
m01s05i205 - ('time', 'latitude', 'longitude') -
m01s05i206 - ('time', 'latitude', 'longitude') -
m01s03i201 - ('time', 'latitude', 'longitude') -
  pressure - ('pressure',) - pressure
m01s30i208 - ('time', 'pressure', 'latitude', 'longitude') -
m01s34i071 - ('time', 'model_level_number', 'latitude', 'longitude') -
m01s34i072 - ('time', 'model_level_number', 'latitude', 'longitude') -
m01s34i073 - ('time', 'model_level_number', 'latitude', 'longitude') -
m01s00i132 - ('time', 'latitude', 'longitude') -
m01s05i216 - ('time', 'latitude', 'longitude') -
m01s05i214 - ('time', 'latitude', 'longitude') -
    height - () - Height
m01s03i245 - ('time', 'latitude', 'longitude') -
m01s00i023 - ('time', 'latitude', 'longitude') -
m01s05i215 - ('time', 'latitude', 'longitude') -
m01s03i237 - ('time', 'latitude', 'longitude') -
m01s04i203 - ('time', 'latitude', 'longitude') -
m01s04i204 - ('time', 'latitude', 'longitude') -
m01s00i409 - ('time', 'latitude', 'longitude') -
m01s00i033 - ('latitude', 'longitude') -
m01s02i207 - ('time', 'latitude', 'longitude') -
m01s02i208 - ('time', 'latitude', 'longitude') -
m01s01i235 - ('time', 'latitude', 'longitude') -
m01s01i210 - ('time', 'latitude', 'longitude') -
m01s00i028 - ('time', 'latitude', 'longitude') -
m01s02i201 - ('time', 'latitude', 'longitude') -
m01s02i203 - ('time', 'latitude', 'longitude') -
m01s01i201 - ('time', 'latitude', 'longitude') -
m01s01i203 - ('time', 'latitude', 'longitude') -
m01s00i029 - ('time', 'latitude', 'longitude') -
m01s00i024 - ('time', 'latitude', 'longitude') -
m01s00i507 - ('time', 'latitude', 'longitude') -
m01s00i508 - ('time', 'latitude', 'longitude') -
m01s03i234 - ('time', 'latitude', 'longitude') -
m01s03i217 - ('time', 'latitude', 'longitude') -
```

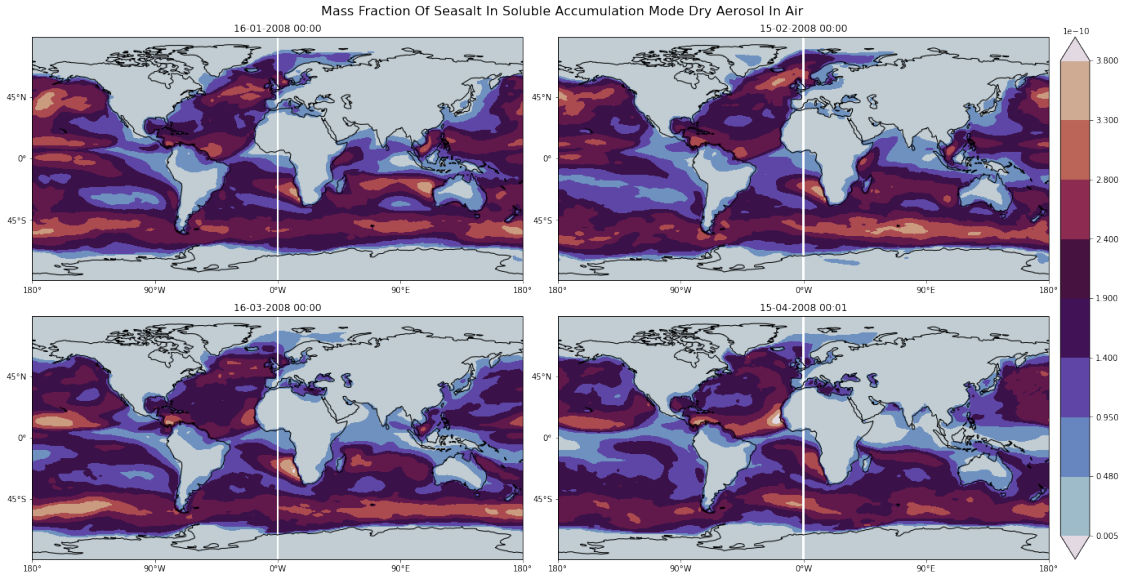
```
m01s03i228 - ('time', 'latitude', 'longitude') -
m01s03i223 - ('time', 'latitude', 'longitude') -
m01s01i211 - ('time', 'latitude', 'longitude') -
m01s01i207 - ('time', 'latitude', 'longitude') -
m01s02i205 - ('time', 'latitude', 'longitude') -
m01s03i332 - ('time', 'latitude', 'longitude') -
m01s02i206 - ('time', 'latitude', 'longitude') -
m01s01i208 - ('time', 'latitude', 'longitude') -
m01s01i209 - ('time', 'latitude', 'longitude') -
m01s30i451 - ('time', 'latitude', 'longitude') -
m01s30i453 - ('time', 'latitude', 'longitude') -
m01s30i203 - ('time', 'pressure', 'latitude', 'longitude') -
m01s03i224 - ('time', 'latitude', 'longitude') -
m01s03i227 - ('time', 'latitude', 'longitude') -
m01s03i225 - ('time', 'latitude', 'longitude') -
m01s03i365 - ('time', 'latitude', 'longitude') -
m01s03i226 - ('time', 'latitude', 'longitude') -
m01s03i366 - ('time', 'latitude', 'longitude') -
```

1.3 Default plotting of a variable

By default we plot in a two column format using the 'twilight' colourmap

[15]:
```
dataset.singleplot('m01s34i111')
```

```
model_level_number ('time', 'model_level_number', 'latitude', 'longitude')

    --- ONLY DISPLAYING A SINGLE model_level_number ---

        The level currently selected is 0, To change this use the pseudolevel
argument.
```



1.4 Tweaking Plot parameters

There are several plot parameter we can tweak at once. These can again be viewed using the help function

[6]:
```
help(dataset.singleplot)
```

```
Help on method singleplot in module aervis.plotting:

singleplot(what, **kwargs) method of aervis.plotting.AerData instance
    A wrapper function for aervis.plotting.single.singleplot

    Arguments:
        what:str - The stash key variable name

    optional arguments:
        level:int - the required level when dealing with pseudo or model_level
        t_steps:list - a list of selected timesteps in the same format as
self.data.variables['time']
        figsize:tuple - figure size in inches
        col:int - number of columns in plot grid
        save:str - file name with path of where to save the figure. Enabling
this does not show the figure on screen.

        projection; - cartopy projection
        cmap:str - colourmap name
        vmin:float - manual min colourmap value
        vmax:float - manual max colourmap value
```

```
        discrete_cbar:bool - continuous or discrete colourbar values
```

1.4.1 Selecting specific timesteps, changing the number of columns

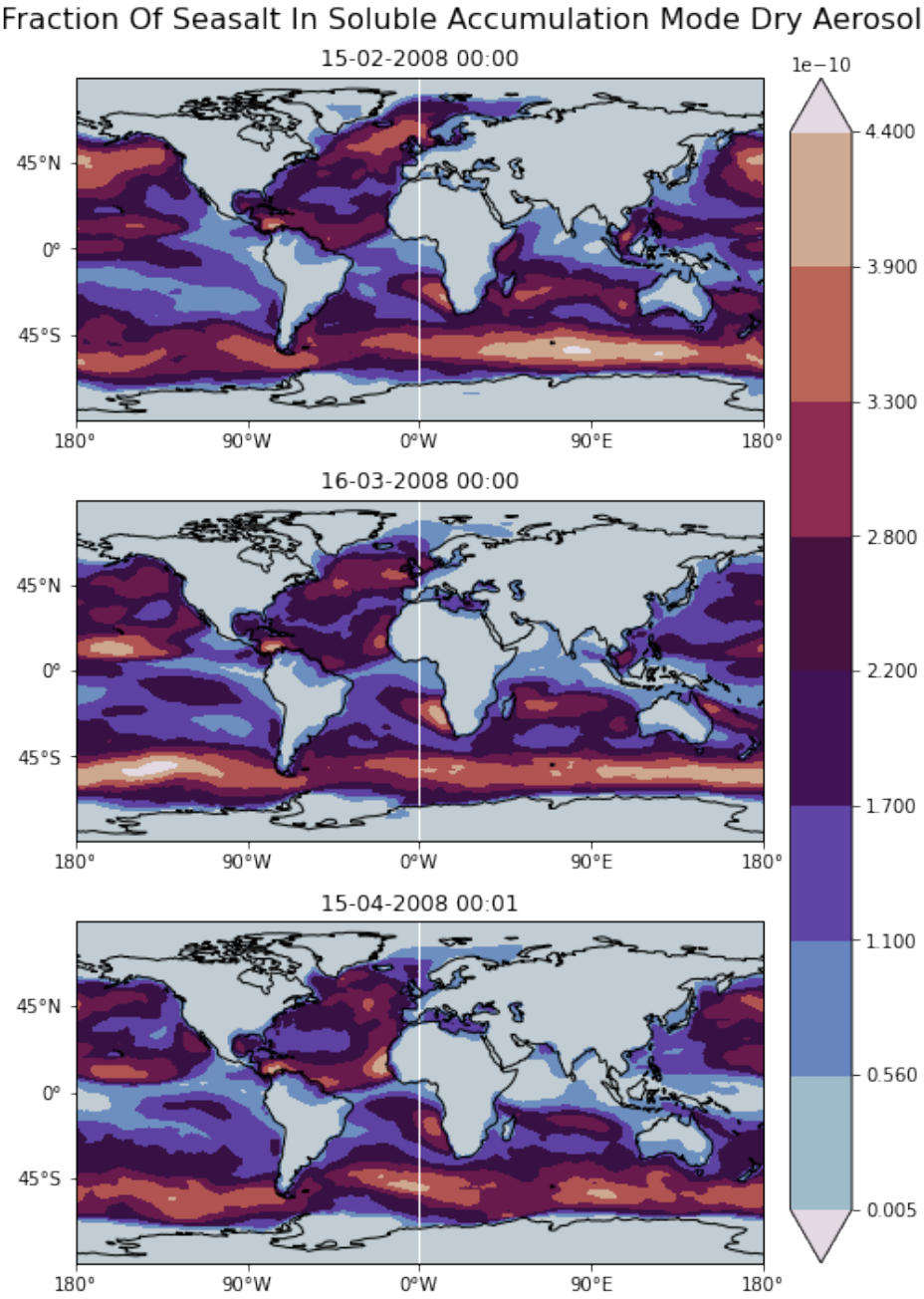
Here we select the last three timesteps, and plot them with a column format.

[7]:
```
# Out of interest, lets see what datetimes each time corresponds to
dataset.get_times(datetimedict=True)
```

[7]:
```
{'01/16/08': masked_array(data=333456.,
                  mask=False,
            fill_value=1e+20), '02/15/08': masked_array(data=334176.,
                  mask=False,
            fill_value=1e+20), '03/16/08': masked_array(data=334896.,
                  mask=False,
            fill_value=1e+20), '04/15/08': masked_array(data=335616.,
                  mask=False,
            fill_value=1e+20)}
```

[8]:
```
## get last three times
selected_times = dataset.get_times()[-3:]

## select the second model_level
nlevel = 1

## number of columns
ncol = 1


dataset.singleplot('m01s34i111',t_steps = selected_times, level=nlevel,␣
 ↪col=ncol)
```

```
model_level_number ('time', 'model_level_number', 'latitude', 'longitude')

    --- ONLY DISPLAYING A SINGLE model_level_number ---

        The level currently selected is 1, To change this use the pseudolevel
argument.
```



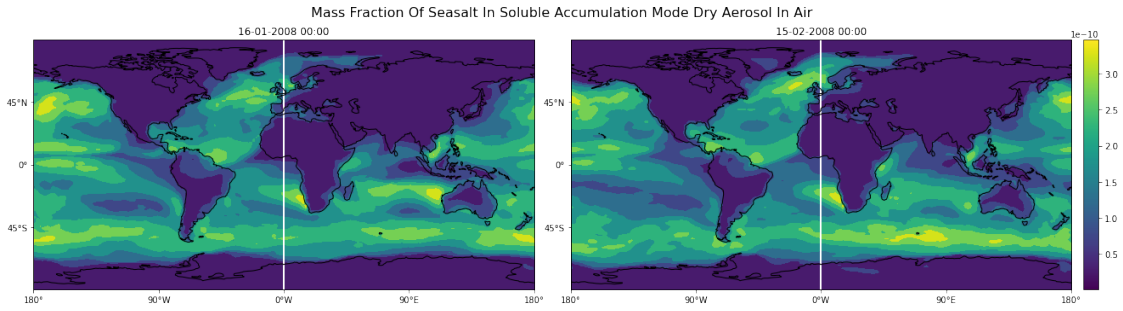
1.5 Changing plot aesthetics

This time we can change the colourmap, and convert the colourbar into a continuous one instead of a discrete on. We shall also be viewing a different variable.

Here you might notice, that since the afforementioned variable has a long name associated with it, the plot title is updated accordingly.

[9]:
```
## new hash
selection = 'm01s34i111'

# colourmap
newcmap = 'viridis'

# cbar
discrete = False

#timesteps
times = dataset.get_times()[0:2]

# fig size (x,y)
fs = (20,5)

dataset.singleplot(selection, cmap = newcmap, discrete_cbar = discrete, t_steps
↪= times , figsize = fs)
```

```
model_level_number ('time', 'model_level_number', 'latitude', 'longitude')

    --- ONLY DISPLAYING A SINGLE model_level_number ---

        The level currently selected is 0, To change this use the pseudolevel
argument.
```



1.6 Single Plot

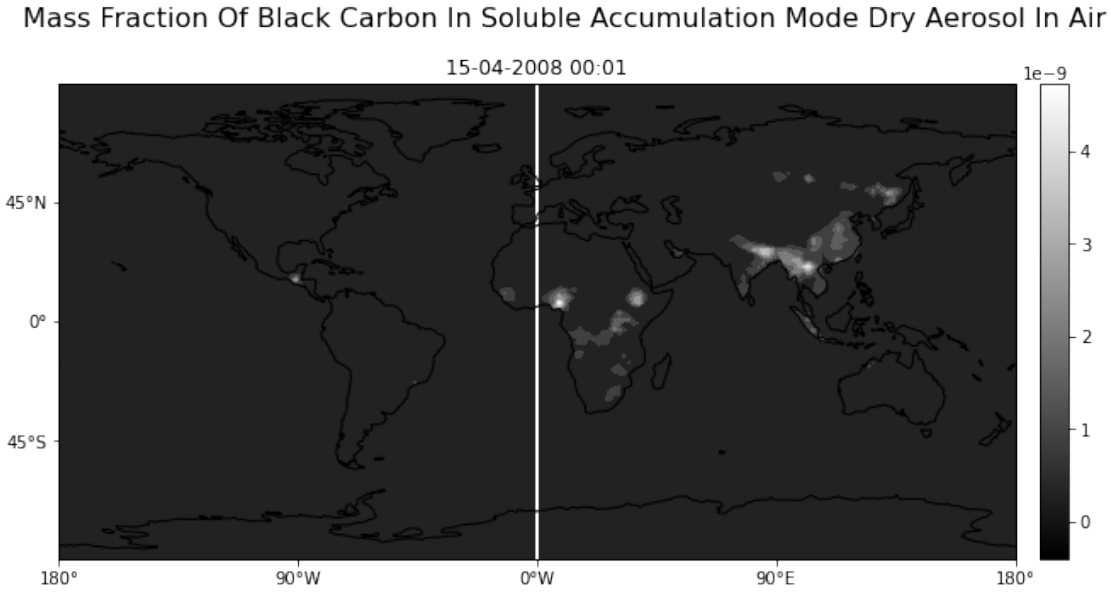
Plotting a single timestep is also possible by selecting a single timestep. Here the number of columns automatically revert to 1 without the need for explicit specificaiton.

To prevent an entirely dark map I will also adjust the colourmap thresholds with `vmin`

[10]:
```
selection = 'm01s34i109'
newcmap = 'gist_gray'
```

```
## NOTE as times has to be a list, selecting one variable makes it a constant,
# and therefore requires us to put square brackets around it again
times = [dataset.get_times()[3]]


dataset.singleplot(selection, cmap = newcmap, discrete_cbar =␣
↪False,figsize=(20,5),t_steps=times,vmin=-40e-11)
```

```
model_level_number ('time', 'model_level_number', 'latitude', 'longitude')

    --- ONLY DISPLAYING A SINGLE model_level_number ---

        The level currently selected is 0, To change this use the pseudolevel
argument.
```



1.7 Advanced - replacing time varaibles for tiles where it does not exist

Some variables - e.g. altitude do not have a time variable. However we can still plot them by replacing how the plotting function handles time.

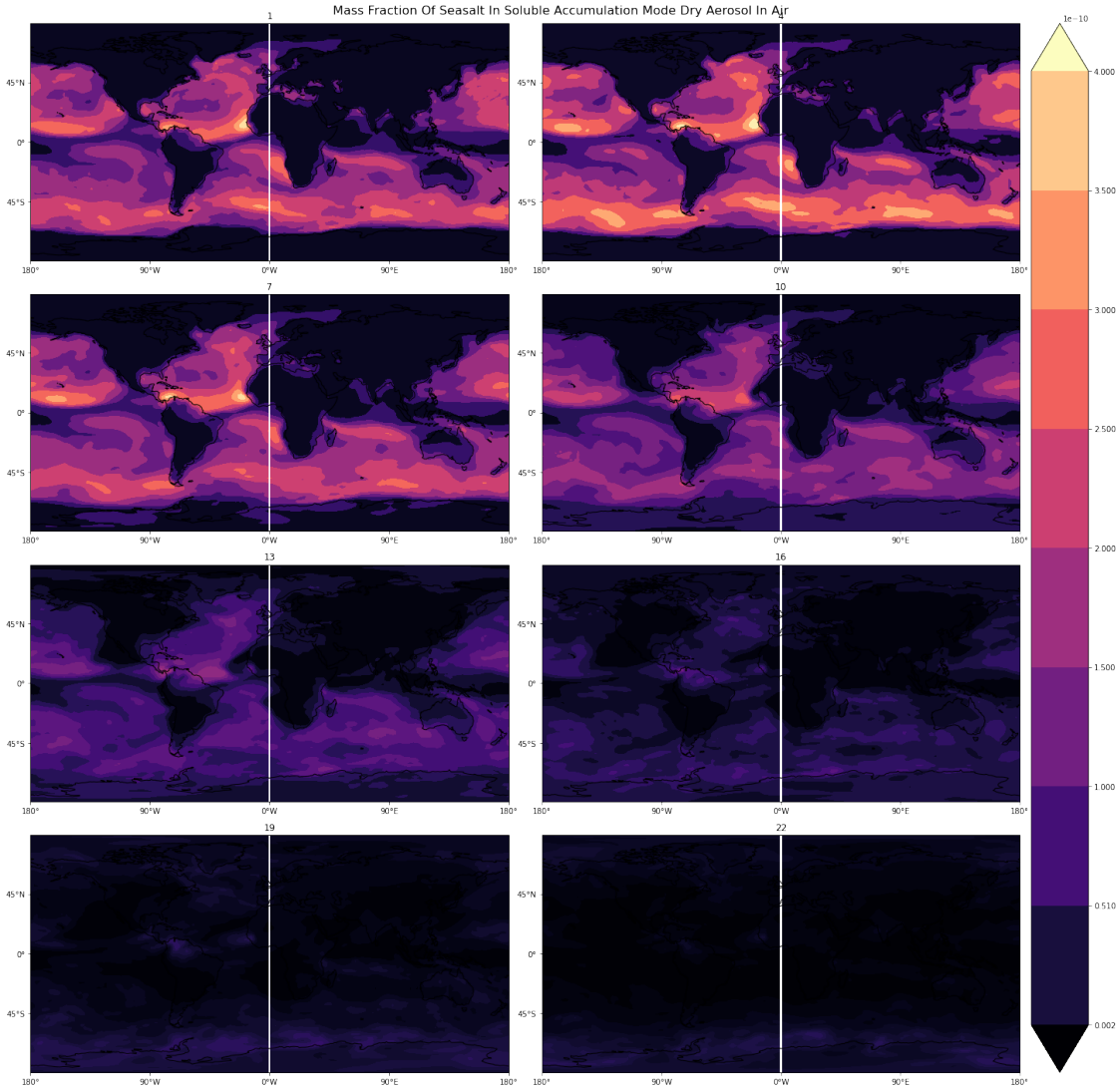
[11]:
```
selection = 'm01s34i111'

# lets view the dim names
dataset.data.variables[selection]
```

[11]:
```
<class 'netCDF4._netCDF4.Variable'>
float32 m01s34i111(time, model_level_number, latitude, longitude)
    _FillValue: nan
    long_name:
mass_fraction_of_seasalt_in_soluble_accumulation_mode_dry_aerosol_in_air
    units: kg kg-1
    source: Data from Met Office Unified Model
    um_version: 11.1
    STASH: [  1  34 111]
    cell_methods: time: mean (interval: 1 hour)
    coordinates: surface_altitude height forecast_reference_time altitude
level_height sigma forecast_period
unlimited dimensions:
current shape = (4, 85, 144, 192)
filling on
```

[18]:
```
level_name = 'model_level_number'

#### Here we see that we have 85 dimensions on model_level_number, so we select␣
 ↪the first 20 in steps of 3
level_numbers = list(dataset.data.variables[level_name])[:22:3]
print(len(level_numbers))

# also select a single time to pass as level
ts = 3 # this is the value at dataset.get_times()[3]
```

```
8
```

[17]:
```
newcmap = 'magma'
## make the figure size larger
fs = (20,20)

dataset.singleplot(selection, cmap = newcmap, figsize=fs,t_steps=level_numbers,␣
 ↪tname=level_name, level=ts )
```



1.8 List all available colourmaps

[14]:
```
import aervis.plotting.cmaps as c
print(c.colours)
```

```
['Accent', 'Accent_r', 'Blues', 'Blues_r', 'BrBG', 'BrBG_r', 'BuGn', 'BuGn_r',
'BuPu', 'BuPu_r', 'CMRmap', 'CMRmap_r', 'Dark2', 'Dark2_r', 'GnBu', 'GnBu_r',
'Greens', 'Greens_r', 'Greys', 'Greys_r', 'OrRd', 'OrRd_r', 'Oranges',
'Oranges_r', 'PRGn', 'PRGn_r', 'Paired', 'Paired_r', 'Pastel1', 'Pastel1_r',
'Pastel2', 'Pastel2_r', 'PiYG', 'PiYG_r', 'PuBu', 'PuBuGn', 'PuBuGn_r',
'PuBu_r', 'PuOr', 'PuOr_r', 'PuRd', 'PuRd_r', 'Purples', 'Purples_r', 'RdBu',
'RdBu_r', 'RdGy', 'RdGy_r', 'RdPu', 'RdPu_r', 'RdYlBu', 'RdYlBu_r', 'RdYlGn',
'RdYlGn_r', 'Reds', 'Reds_r', 'Set1', 'Set1_r', 'Set2', 'Set2_r', 'Set3',
```

```
'Set3_r', 'Spectral', 'Spectral_r', 'Wistia', 'Wistia_r', 'YlGn', 'YlGnBu',
'YlGnBu_r', 'YlGn_r', 'YlOrBr', 'YlOrBr_r', 'YlOrRd', 'YlOrRd_r', 'afmhot',
'afmhot_r', 'autumn', 'autumn_r', 'binary', 'binary_r', 'bone', 'bone_r', 'brg',
'brg_r', 'bwr', 'bwr_r', 'cividis', 'cividis_r', 'cool', 'cool_r', 'coolwarm',
'coolwarm_r', 'copper', 'copper_r', 'cubehelix', 'cubehelix_r', 'flag',
'flag_r', 'gist_earth', 'gist_earth_r', 'gist_gray', 'gist_gray_r', 'gist_heat',
'gist_heat_r', 'gist_ncar', 'gist_ncar_r', 'gist_rainbow', 'gist_rainbow_r',
'gist_stern', 'gist_stern_r', 'gist_yarg', 'gist_yarg_r', 'gnuplot', 'gnuplot2',
'gnuplot2_r', 'gnuplot_r', 'gray', 'gray_r', 'hot', 'hot_r', 'hsv', 'hsv_r',
'inferno', 'inferno_r', 'jet', 'jet_r', 'magma', 'magma_r', 'nipy_spectral',
'nipy_spectral_r', 'ocean', 'ocean_r', 'pink', 'pink_r', 'plasma', 'plasma_r',
'prism', 'prism_r', 'rainbow', 'rainbow_r', 'seismic', 'seismic_r', 'spring',
'spring_r', 'summer', 'summer_r', 'tab10', 'tab10_r', 'tab20', 'tab20_r',
'tab20b', 'tab20b_r', 'tab20c', 'tab20c_r', 'terrain', 'terrain_r', 'twilight',
'twilight_r', 'twilight_shifted', 'twilight_shifted_r', 'viridis', 'viridis_r',
'winter', 'winter_r']
```

[ ]: