
ATLAS MPL Style

Mar 21, 2023

Contents:

1	atlas_mpl_style Module	1
2	atlas_mpl_style.plot Module	3
3	atlas_mpl_style.stats Module	9
4	atlas_mpl_style.uhi Module	11
5	atlas_mpl_style.utils Module	15
6	ATLAS Style	17
7	Other Styles	19
8	Additional Colors	21
8.1	Default Color Cycle	21
8.2	Paper Colors	22
8.3	Oceanic Next Colors	23
8.4	ATLAS Color Cycle	24
8.5	HDBS Colors	25
8.6	HH Colors	26
8.7	Transparent	27
9	Indices and tables	29
	Python Module Index	31
	Index	33

atlas_mpl_style Module

`atlas_mpl_style.ratio_axes` (*extra_axes=None*)

Splits axes for ratio plots.

Parameters

- **extra_axes** (*int, optional*) – Number of additional axes. If not given, defaults to one.
- **square** (*bool, optional*) – Whether the plot should be square or tall. Defaults to True (square)

Returns

- **fig** (*figure*)
- **main_ax** (*axes*)
- **ratio_ax** (*axes or list of axes*) – Returns list if *extra_axes* is passed

`atlas_mpl_style.set_color_cycle` (*pal=None, n=4*)

Sets a different color cycle.

The ATLAS palette includes the standard green and yellow.

Parameters

- **pal** (*{'ATLAS', 'Paper', 'Oceanic', 'MPL', 'HDBS', 'HH', None}*)
– The palette to use. None resets to default palette. The ATLAS palette is suitable for histograms, not lines.
 'MPL' (alias 'Tab') provides the default matplotlib palette.
- **n** (*int, optional*) – Number of lines or histograms.

`atlas_mpl_style.use_atlas_style` (*atlasLabel='ATLAS', fancyLegend=False, usetex=False*)

Setup ATLAS style.

Parameters

- **atlasLabel** (*str, optional*) – Replace ATLAS with a custom label

- **fancyLegend** (*bool, optional*) – Use matplotlib’s fancy legend frame (defaults to False)
- **usetex** (*bool, optional*) – Use LaTeX installation to set text (defaults to False) If no LaTeX installation is found, this package will fallback to `usetex=False`. This is on a best-effort basis, since the detected LaTeX installation may be incomplete.

atlas_mpl_style.plot Module

class atlas_mpl_style.plot.**Background**(*label*, *hist*, *stat_errs=None*, *syst_errs=None*,
color=None)
Histogram and errors corresponding to a single background

exception atlas_mpl_style.plot.**BinningMismatchError**(*msg*)
Error due to histogram binning mismatch

exception atlas_mpl_style.plot.**DimensionError**(*msg*)
Error due to incorrect / unsupported histogram dimension

exception atlas_mpl_style.plot.**ViolatesPlottableHistogramError**(*msg*)
Error due to histogram object violating the PlottableHistogram protocol

atlas_mpl_style.plot.**draw_atlas_label**(*x*, *y*, *ax=None*, *status='int'*, *simulation=False*, *en-*
ergy=None, *lumi=None*, *desc=None*, *lumi_lt=False*,
args*, *kwargs*)

Draw ATLAS label.

Additional parameters are passed through to `ax.text`.

Parameters

- **x** (*float*) – x position (top left)
- **y** (*float*) – y position (top left)
- **ax** (*mpl.axes.Axes*, *optional*) – Axes to draw label in
- **status** ([*'int'* | *'wip'* | *'prelim'* | *'final'* | *'opendata'*], *optional*) – Approval status
- **simulation** (*bool* (*optional*, default `False`)) – Does the plot show only MC simulation results
- **energy** (*str*, *optional*) – Centre of mass energy, including units
- **lumi** (*float or str*, *optional*) – Integrated luminosity in /fb. If *str*, the units should be included.
- **lumi_lt** (*bool*, *optional*) – True if only a subset of data was processed

- **desc** (*str*, *optional*) – Additional description

`atlas_mpl_style.plot.draw_legend(*args, ax=None, **kwargs)`

Add legend to axes with data first, and uncertainties last.

Parameters

- **ax** (*mpl.axes.Axes*, *optional*) – Axes to draw legend on (defaults to current axes)
- ***args** – Passed to `ax.legend`
- ****kwargs** – Passed to `ax.legend`

`atlas_mpl_style.plot.draw_tag(text, ax=None)`

Draw tag just outside plot region

Parameters

- **text** (*str*) –
- **ax** (*mpl.axes.Axes*, *optional*) – Axes to draw on (defaults to current axes)

`atlas_mpl_style.plot.plot_1d(label, bins, hist, stat_errs=None, color=None, attach_bands=False, ax=None, **kwargs)`

Plot single 1D histogram

NB: `atlas_mpl_style.uhi.plot_1d()` provides a version of this function that accepts a `PlottableHistogram`.

Parameters

- **label** (*str*) – Label for legend
- **bins** (*array_like*) – Bin edges
- **hist** (*array_like*) – Bin contents
- **stat_errs** (*array_like*, *optional*) – Statistical errors
- **color** (*color*, *optional*) – Line color
- **attach_bands** (*boolean*, *optional*) – Attach bands to line in legend. Defaults to `False`.
- **ax** (*mpl.axes.Axes*, *optional*) – Axes to draw on (defaults to current axes)
- ****kwargs** – Extra parameters passed to `plt.hist`

`atlas_mpl_style.plot.plot_2d(xbins, ybins, hist, ax=None, pad=0.05, **kwargs)`

Plot 2D histogram

NB: `atlas_mpl_style.uhi.plot_2d()` provides a version of this function that accepts a `PlottableHistogram`.

Parameters

- **xbins** (*array_like*) – x bin edges
- **ybins** (*array_like*) – y bin edges
- **hist** (*array_like*) – Bin contents
- **ax** (*mpl.axes.Axes*, *optional*) – Axes to draw on (defaults to current axes)
- **pad** (*float*, *optional*) – Padding for colorbar in inches (defaults to 0.05)
- ****kwargs** – Extra parameters passed to `pcolormesh`

Returns

- **mesh** (*QuadMesh*)
- **cbar** (*mpl.colorbar.Colorbar*)

`atlas_mpl_style.plot.plot_backgrounds` (*backgrounds, bins=None, *, total_err=None, empty_stat_legend=False, ax=None*)

Plot stacked backgrounds

Parameters

- **backgrounds** (*[Background]*) – List of backgrounds to be plotted, in order (bottom to top)
- **bins** (*array_like, optional*) – Bin edges. To preserve backward compatibility, backgrounds and bins may be exchanged.
- **total_err** (*array_like, optional*) – Total uncertainty. If given, overrides per-background systematics. This is useful for showing post-fit uncertainties.
- **empty_stat_legend** (*boolean, optional*) – Add stat error band to legend even if empty. Defaults to False.
- **ax** (*mpl.axes.Axes, optional*) – Axes to draw on (defaults to current axes)

Returns

- **total_hist** (*array_like*) – Total background histogram
- **total_err** (*array_like*) – Total error on background histogram

`atlas_mpl_style.plot.plot_band` (*bins, low, high, label=None, ax=None, **kwargs*)

Draw a shaded band between high and low

Use this for drawing error bands

Parameters

- **bins** (*array_like*) – Bin edges
- **low** (*array_like*) – Bin contents defining lower bound
- **high** (*array_like*) – Bin contents defining upper bound
- **label** (*str, optional*) – Label for legend. If label matches a line, the band will be attached to that line if `draw_legend` is used.
- **ax** (*mpl.axes.Axes, optional*) – Axes to draw band on (defaults to current axes)
- ****kwargs** – Keyword arguments passed to `fill_between`

`atlas_mpl_style.plot.plot_cutflow` (*labels, hist, ax=None, text=True, textcolor='w', horizontal=True, **kwargs*)

Plot cutflow from `PlottableHistogram`

Parameters

- **labels** (*[str]*) – Cutflow labels
- **hist** (*PlottableHistogram*) – Cutflow histogram
- **ax** (*mpl.axes.Axes, optional*) – Axes to draw on (defaults to current axes)
- **text** (*bool, optional*) – Whether to label bars (default: True)
- **textcolor** (*str, optional*) – Text color
- **horizontal** (*bool, optional*) – Whether to draw horizontal bars (default: True)
- ****kwargs** – Extra parameters passed to `bar` or `barh`

`atlas_mpl_style.plot.plot_data` (*bins*, *hist*, *stat_errs=None*, *color='k'*, *label='Data'*, *ax=None*)

Plot data

NB: `atlas_mpl_style.uhi.plot_data()` provides a version of this function that accepts a `PlottableHistogram`.

Parameters

- **label** (*str*, *optional*) – Label for legend (default: “Data”)
- **bins** (*array_like*) – Bin edges
- **hist** (*array_like*) – Bin contents
- **stat_errs** (*array_like*, *optional*) – Statistical errors
- **color** (*color*, *optional*) – Point color, defaults to black
- **ax** (*mpl.axes.Axes*, *optional*) – Axes to draw on (defaults to current axes)

Returns

- **hist** (*array_like*) – Data histogram
- **stat_errs** (*array_like*) – Statistical errors

`atlas_mpl_style.plot.plot_limit` (*expected_label*, *x*, *expected*, *minus_one_sigma=None*,
plus_one_sigma=None, *minus_two_sigma=None*,
plus_two_sigma=None, *observed_label=None*, *observed=None*, *color=None*, *ax=None*)

Plot a limit

Parameters

- **expected_label** (*str*) – Label for expected limit (for legend)
- **x** (*array_like*) – x values
- **expected** (*array_like*) – Expected limit
- **minus_one_sigma** (*array_like*, *optional*) – Lower edge of one sigma band
- **plus_one_sigma** (*array_like*, *optional*) – Upper edge of one sigma band
- **minus_two_sigma** (*array_like*, *optional*) – Lower edge of two sigma band
- **plus_two_sigma** (*array_like*, *optional*) – Upper edge of two sigma band
- **observed_label** (*str*, *optional*) – Label for observed limit
- **observed** (*array_like*, *optional*) – Observed limit
- **color** (*color*, *optional*) – Line color (if multiple limits are being drawn)
- **ax** (*mpl.axes.Axes*, *optional*) – Axes to draw on (defaults to current axes)

`atlas_mpl_style.plot.plot_ratio` (*bins*, *data*, *data_errs*, *bkg*, *bkg_errs*, *ratio_ax*,
max_ratio=None, *plottype='diff'*, *offscale_errs=False*)

Plot ratio plot

NB: `atlas_mpl_style.uhi.plot_ratio()` provides a version of this function that accepts “`Plottable-Histogram`”s.

Parameters

- **bins** (*array_like*) – Bin edges
- **data** (*array_like*) – Data histogram bin contents

- **data_errs** (*array_like*) – Statistical errors on data
- **bkg** (*array_like*) – Total background histogram bin contents
- **bkg_errs** (*array_like*) – Total errors on total background
- **ratio_ax** (*mpl.axes.Axes*) – Ratio axes (produced using `atlas_mpl_style.ratio_axes()`)
- **max_ratio** (*float, optional*) – Maximum ratio (defaults to 0.25 for “diff”, 1.25 for “raw”, 3.5 for “significances”).
- **plottype** (`{"diff", "raw", "significances"}`) – Type of ratio to plot.
 ”diff” : $(\text{data} - \text{bkg}) / \text{bkg}$
 ”raw” : data / bkg
 ”significances” : Significances (using `atlas_mpl_style.utils.significance()`)
- **offscale_err** (*boolean*) – Draw error bars on off-scale points

`atlas_mpl_style.plot.plot_signal` (*label, bins, hist, stat_errs=None, syst_errs=None, color=None, attach_bands=False, ax=None*)

Plot signal histogram

NB: `atlas_mpl_style.uhi.plot_signal()` provides a version of this function that accepts a `PlottableHistogram`.

Parameters

- **label** (*str*) – Label for legend
- **bins** (*array_like*) – Bin edges
- **hist** (*array_like*) – Bin contents
- **stat_errs** (*array_like*) – Statistical errors
- **syst_errs** (*array_like*) – Systematic errors
- **color** (*color*) – Line color
- **attach_bands** (*boolean, optional*) – Attach bands to line in legend. Defaults to False.
- **ax** (*mpl.axes.Axes, optional*) – Axes to draw on (defaults to current axes)

`atlas_mpl_style.plot.register_band` (*label, artist, ax=None*)

Register a manually draw (e.g. with `fill_between`) error band.

Parameters

- **label** (*str*) – Label of line to attach band to.
- **artist** (*mpl.artist.Artist*) – Band artist, e.g. `PolyCollection` returned by `fill_between`.
- **ax** (*mpl.axes.Axes, optional*) – Axes to register band in (defaults to current axes).

`atlas_mpl_style.plot.set_xlabel` (*label, ax=None, *args, **kwargs*)

Set x label in ATLAS style (right aligned). If `ratio_axes` was used, the label will be set on the lowest ratio axes.

Additional parameters are passed through to `ax.set_xlabel`.

Parameters

- **label** (*str*) – Label (LaTeX permitted)
- **ax** (*mpl.axes.Axes, optional*) – Axes to set x label on

`atlas_mpl_style.plot.set_ylabel` (*label, ax=None, *args, **kwargs*)
Set y label in ATLAS style (top aligned).

Additional parameters are passed through to `ax.set_ylabel`.

Parameters

- **label** (*str*) – Label (LaTeX permitted)
- **ax** (*mpl.axes.Axes, optional*) – Axes to set y label on

`atlas_mpl_style.plot.set_zlabel` (*label, cbar=None, ax=None, **kwargs*)
Set z label in ATLAS style (top aligned)

The colorbar to add the label to is *required* unless `plot_2d` was used.

Parameters

- **label** (*str*) – Label (LaTeX permitted)
- **cbar** (*mpl.colorbar.Colorbar, optional*) – Colorbar to set label on. Not required if `plot_2d` was used.
- **ax** (*mpl.axes.Axes, optional*) – If `plot_2d` was used, the axes can optionally be provided here.

atlas_mpl_style.stats Module

exception atlas_mpl_style.stats.**IncorrectAxesError** (*msg*)

Error due to passing incorrect axes to draw_pull_impact_legend

atlas_mpl_style.stats.**draw_pull_impact_legend** (**args, ax=None, **kwargs*)

Add legend to a pull / impact plot.

Parameters

- **ax** (*mpl.axes.Axes, optional*) – Pull axes
- ****kwargs** – Passed to ax.legend

atlas_mpl_style.stats.**make_impact_figure** (*num_parameters*)

Create a new figure for a pull / impact plot.

Parameters **num_parameters** (*Integer*) – Number of parameters on this plot

Returns **fig** – Created figure

Return type Figure

atlas_mpl_style.stats.**plot_impacts** (*data, draw_prefit=False, up_color='paper:blue',
down_color='paper:red', ax=None*)

Plot impacts from a Pandas dataframe (*data*).

The dataframe must have the following columns. The prefit columns are not required if draw_prefix == False.

name	Parameter name
impact_prefit_up	Impact on POI from fit with parameter fixed to $+1\sigma$ (using prefit σ)
impact_prefit_down	Impact on POI from fit with parameter fixed to -1σ (using prefit σ)
impact_postfit_up	Impact on POI from fit with parameter fixed to $+1\sigma$ (using postfit σ)
impact_postfit_down	Impact on POI from fit with parameter fixed to -1σ (using postfit σ)

Parameters

- **data** (*pd.DataFrame*) – Pandas dataframe containing impacts

- **draw_prefit** (*Boolean, optional*) – Whether to draw the prefit bands
- **up_color** (*Color specification*) – Color to use for upper band (postfit band will be at 50% opacity)
- **up_color** – Color to use for lower band (postfit band will be at 50% opacity)
- **ax** (*mpl.axes.Axes, optional*) – Axes to pulls were drawn on (defaults to current axes). Impact axes will be a twin of these.

Returns **ax** – The axes impacts were drawn on

Return type `mpl.axes.Axes`

`atlas_mpl_style.stats.plot_pulls` (*data, ax=None, **kwargs*)

Plot pulls from a Pandas dataframe (*data*).

The dataframe must have at least the following columns:

name	Parameter name
value	Post nominal fit central value of parameter
err_high	Post nominal fit error (high side) on parameter
err_low	Post nominal fit error (low side) on parameter.

Parameters

- **data** (*pd.DataFrame*) – Pandas dataframe containing pulls
- **ax** (*mpl.axes.Axes, optional*) – Axes to draw pulls on (defaults to current axes)
- ****kwargs** – Keyword arguments passed to `errorbar`

Returns

- **ax** (*mpl.axes.Axes*) – The axes pulls were drawn on
- **pull_plot** (*mpl.container.ErrorbarContainer*) – The return value of `errorbar`

`atlas_mpl_style.stats.sort_impacts` (*data*)

Sort an impact (and pull) dataframe in-place by descending postfit impacts.

The frame must have at least the following columns:

name	Parameter name
impact_postfit_up	Impact on POI from fit with parameter fixed to $+1\sigma$ (using postfit σ)
impact_postfit_down	Impact on POI from fit with parameter fixed to -1σ (using postfit σ)

Parameters **data** (*pd.DataFrame*) – Pandas dataframe containing impacts

atlas_mpl_style.uhi Module

This module contains versions of the histogram plotting functions that take `PlottableHistograms`

These are in a separate module to preserve backward compatibility since the array versions of these functions take the array of bins before the histogram.

`atlas_mpl_style.plot.Background` can be constructed using a `PlottableHistogram` and therefore there is no `atlas_mpl_style.uhi.plot_backgrounds` function.

exception `atlas_mpl_style.uhi.LabeledBinsError(msg)`
Labeled bins when edges expected (or vice versa)

`atlas_mpl_style.uhi.plot_1d(hist, label, ignore_variances=False, stat_err=True, color=None, attach_bands=False, ax=None, **kwargs)`
Plot single 1D histogram from `PlottableHistogram`

Parameters

- **hist** (`PlottableHistogram`) – Histogram
- **label** (`str`) – Label for legend
- **ignore_variances** (`bool`) – Ignore variances and substitute `hist`. Defaults to `False`.
- **stat_err** (`bool`) – Draw statistical errors. Defaults to `True`.
- **color** (`color, optional`) – Line color
- **attach_bands** (`boolean, optional`) – Attach bands to line in legend. Defaults to `False`.
- **ax** (`mpl.axes.Axes, optional`) – Axes to draw on (defaults to current axes)
- ****kwargs** – Extra parameters passed to `plt.hist`

`atlas_mpl_style.uhi.plot_2d(hist, ax=None, pad=0.05, **kwargs)`
Plot 2D histogram from `PlottableHistogram`

Parameters

- **hist** (`PlottableHistogram`) – Histogram

- **ax** (*mpl.axes.Axes, optional*) – Axes to draw on (defaults to current axes)
- **pad** (*float, optional*) – Padding for colorbar in inches (defaults to 0.05)
- ****kwargs** – Extra parameters passed to `pcolormesh`

Returns

- **mesh** (*QuadMesh*)
- **cbar** (*mpl.colorbar.Colorbar*)

`atlas_mpl_style.uhi.plot_cutflow` (*hist, ax=None, text=True, textcolor='w', horizontal=True, **kwargs*)

Plot cutflow from PlottableHistogram

Parameters

- **hist** (*PlottableHistogram*) – Cutflow histogram
- **ax** (*mpl.axes.Axes, optional*) – Axes to draw on (defaults to current axes)
- **text** (*bool, optional*) – Whether to label bars (default: True)
- **textcolor** (*str, optional*) – Text color
- **horizontal** (*bool, optional*) – Whether to draw horizontal bars (default: True)
- ****kwargs** – Extra parameters passed to `bar` or `barh`

`atlas_mpl_style.uhi.plot_data` (*hist, ignore_variances=False, color='k', label='Data', ax=None*)

Plot data from PlottableHistogram

Parameters

- **label** (*str, optional*) – Label for legend (default: “Data”)
- **hist** (*PlottableHistogram*) – Histogram
- **ignore_variances** (*bool*) – Ignore variances and substitute `hist`. Defaults to False.
- **color** (*color, optional*) – Point color, defaults to black
- **ax** (*mpl.axes.Axes, optional*) – Axes to draw on (defaults to current axes)

Returns

- **hist** (*array_like*) – Data histogram
- **stat_errs** (*array_like*) – Statistical errors

`atlas_mpl_style.uhi.plot_ratio` (*data, total_bkg, ratio_ax, max_ratio=None, plottype='diff'*)

Plot ratio plot from PlottableHistogram

Parameters

- **data** (*PlottableHistogram*) – Data histogram
- **total_bkg** (*(array_like, array_like)*) – Tuple returned from `atlas_mpl_style.plot.plot_backgrounds()`
- **ratio_ax** (*mpl.axes.Axes*) – Ratio axes (produced using `atlas_mpl_style.ratio_axes()`)
- **max_ratio** (*float, optional*) – Maximum ratio (defaults to 0.2 for “diff”, 1.2 for “raw”, 3 for “significances”)
- **plottype** (*{ "diff", "raw", "significances" }*) –

Type of ratio to plot.

"diff" : $(\text{data} - \text{bkg}) / \text{bkg}$

"raw" : data / bkg

"significances" : Significances (from *ampl.utils.significance()*)

`atlas_mpl_style.uhi.plot_signal` (*hist*, *label*, *ignore_variances=False*, *syst_errs=None*,
color=None, *attach_bands=False*, *ax=None*)

Plot signal histogram from PlottableHistogram

Parameters

- **hist** (*PlottableHistogram*) – Histogram
- **label** (*str*) – Label for legend
- **ignore_variances** (*bool*) – Ignore variances and substitute `hist`. Defaults to False.
- **syst_errs** (*array_like* or *PlottableHistogram*, *optional*) – Systematic errors
- **color** (*color*) – Line color
- **attach_bands** (*boolean*, *optional*) – Attach bands to line in legend. Defaults to False.
- **ax** (*mpl.axes.Axes*, *optional*) – Axes to draw on (defaults to current axes)

atlas_mpl_style.utils Module

`atlas_mpl_style.utils.significance` (*data*, *data_errs*, *bkg*, *bkg_errs*)

Calculates significance in each bin

Uses the significance definition in <https://cds.cern.ch/record/2643488>

Parameters

- **data** (*array_like*) –
- **data_errs** (*array_like*) – Errors / uncertainties on *data*
- **bkg** (*array_like*) – Total background prediction
- **bkg_errs** (*array_like*) – Errors / uncertainties on *bkg*

ATLAS Style

The main purpose of this package is to provide a Matplotlib style closely resembling that used by the [ATLAS](#) experiment for its plots.

This style can be activated by calling

```
import atlas_mpl_style as ampl
ampl.use_atlas_style()
```

When the ATLAS style is active, text is typeset using LaTeX, and the standard ATLAS label can be drawn using the `ampl.draw_atlas_label` function. The use of LaTeX can be disabled by instead calling `ampl.use_atlas_style(usetex=False)`.

The axis labels should be set using the `ampl.set_xlabel` and `ampl.set_ylabel` functions, to ensure they are correctly right / top aligned.

Other Styles

Additionally, two other styles based on the Paper VIM color scheme are provided. **Slides** has an off-white central background, and works well on slides. **Print** has a white background, for use in print. These styles do not use LaTeX for text typesetting, and can be activated using

```
# import matplotlib in the usual way
import matplotlib.pyplot as plt
import atlas_mpl_style as ampl
plt.style.use('slides')
# or
plt.style.use('print')
```


ATLAS MPL Style adds a number of additional color definitions to Matplotlib.

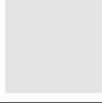
8.1 Default Color Cycle

These are the colors of the default color cycle.

<code>series:cyan</code>	<code>#54c9d1</code>	
<code>series:orange</code>	<code>#eca89a</code>	
<code>series:blue</code>	<code>#95bced</code>	
<code>series:olive</code>	<code>#ceb776</code>	
<code>series:purple</code>	<code>#d3a9ea</code>	
<code>series:green</code>	<code>#9bc57f</code>	
<code>series:pink</code>	<code>#f0a1ca</code>	
<code>series:turquoise</code>	<code>#5fcbaa</code>	

8.2 Paper Colors

These colors are from the [Paper](#) color scheme for VIM.

paper:bg	#eeeeee	
paper:fg	#444444	
paper:bgAlt	#e4e4e4	
paper:red	#af0000	
paper:green	#008700	
paper:blue	#005f87	
paper:yellow	#afaf00	
paper:orange	#d75f00	
paper:pink	#d70087	
paper:purple	#8700af	
paper:lightBlue	#0087af	
paper:olive	#5f7800	

8.3 Oceanic Next Colors

These colors are from the [Oceanic Next](#) color scheme.

on:bg	#1b2b34	
on:fg	#cdd3de	
on:bgAlt	#343d46	
on:fgAlt	#d8dee9	
on:red	#ec5f67	
on:orange	#f99157	
on:yellow	#fac863	
on:green	#99c794	
on:cyan	#5fb3b3	
on:blue	#6699cc	
on:pink	#c594c5	
on:brown	#ab7967	

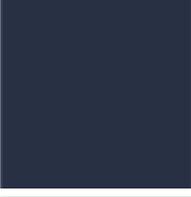
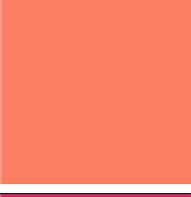
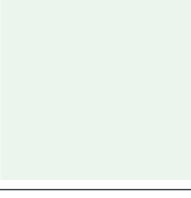
8.4 ATLAS Color Cycle

These are the colors from the ATLAS color cycle. Green and yellow also have the additional aliases `atlas:onesigma` and `atlas:twosigma` respectively for use in limit plots.

atlas:onesigma	#00ff26	
atlas:twosigma	#fbff1f	
series2:green	#00ff26	
series2:yellow	#fbff1f	
series2:blue	#00a1e0	
series2:red	#a30013	
series2:purple	#5100c2	

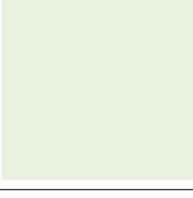
8.5 HDBS Colors

These are the ATLAS HDBS physics groups colors. Mint cream is not included in the color cycle.

<code>hdbs:starcommandblue</code>	<code>#047cbc</code>	
<code>hdbs:spacecadet</code>	<code>#283044</code>	
<code>hdbs:maroonX11</code>	<code>#b8336a</code>	
<code>hdbs:outrageousorange</code>	<code>#fa7e61</code>	
<code>hdbs:pictorialcarmine</code>	<code>#ca1551</code>	
<code>hdbs:mintcream</code>	<code>#ebf5ee</code>	

8.6 HH Colors

These are the ATLAS di-Higgs colors. Light turquoise and off-white are not included in the automatic color cycle.

hh:darkblue	#343844	
hh:darkpink	#f2385a	
hh:darkyellow	#fdc536	
hh:medturquoise	#36b1bf	
hh:lightturquoise	#4ad9d9	
hh:offwhite	#e9f1df	

8.7 Transparent

A fully transparent color is also provided for convenience.

transparent	#ffffff00	
-------------	-----------	--

CHAPTER 9

Indices and tables

- `genindex`
- `modindex`
- `search`

a

atlas_mpl_style, 1
atlas_mpl_style.plot, 3
atlas_mpl_style.stats, 9
atlas_mpl_style.uhi, 11
atlas_mpl_style.utils, 15

A

atlas_mpl_style (*module*), 1
atlas_mpl_style.plot (*module*), 3
atlas_mpl_style.stats (*module*), 9
atlas_mpl_style.uhi (*module*), 11
atlas_mpl_style.utils (*module*), 15

B

Background (*class in atlas_mpl_style.plot*), 3
BinningMismatchError, 3

D

DimensionError, 3
draw_atlas_label() (*in module atlas_mpl_style.plot*), 3
draw_legend() (*in module atlas_mpl_style.plot*), 4
draw_pull_impact_legend() (*in module atlas_mpl_style.stats*), 9
draw_tag() (*in module atlas_mpl_style.plot*), 4

I

IncorrectAxesError, 9

L

LabeledBinsError, 11

M

make_impact_figure() (*in module atlas_mpl_style.stats*), 9

P

plot_1d() (*in module atlas_mpl_style.plot*), 4
plot_1d() (*in module atlas_mpl_style.uhi*), 11
plot_2d() (*in module atlas_mpl_style.plot*), 4
plot_2d() (*in module atlas_mpl_style.uhi*), 11
plot_backgrounds() (*in module atlas_mpl_style.plot*), 5
plot_band() (*in module atlas_mpl_style.plot*), 5
plot_cutflow() (*in module atlas_mpl_style.plot*), 5

plot_cutflow() (*in module atlas_mpl_style.uhi*), 12
plot_data() (*in module atlas_mpl_style.plot*), 5
plot_data() (*in module atlas_mpl_style.uhi*), 12
plot_impacts() (*in module atlas_mpl_style.stats*), 9
plot_limit() (*in module atlas_mpl_style.plot*), 6
plot_pulls() (*in module atlas_mpl_style.stats*), 10
plot_ratio() (*in module atlas_mpl_style.plot*), 6
plot_ratio() (*in module atlas_mpl_style.uhi*), 12
plot_signal() (*in module atlas_mpl_style.plot*), 7
plot_signal() (*in module atlas_mpl_style.uhi*), 13

R

ratio_axes() (*in module atlas_mpl_style*), 1
register_band() (*in module atlas_mpl_style.plot*), 7

S

set_color_cycle() (*in module atlas_mpl_style*), 1
set_xlabel() (*in module atlas_mpl_style.plot*), 7
set_ylabel() (*in module atlas_mpl_style.plot*), 8
set_zlabel() (*in module atlas_mpl_style.plot*), 8
significance() (*in module atlas_mpl_style.utils*), 15
sort_impacts() (*in module atlas_mpl_style.stats*), 10

U

use_atlas_style() (*in module atlas_mpl_style*), 1

V

ViolatesPlottableHistogramError, 3