
petrodC
Release 0.1.10

Pro Well Plan AS

Apr 30, 2021

CONTENTS:

1	Installation	3
1.1	Requirements	3
2	Topo-bathymetry data	5
2.1	Example	5
2.2	Web Application	6
3	Athabasca well logs	7
3.1	Example	7
3.2	Web Application	8
4	Norwegian Wellbore Data	9
4.1	Example	9
4.2	Web Application	10
5	About Pro Well Plan	11
	Index	13

Find here the documentation of petrodc. Here you will find all the relevant information about any function included in this package.

INSTALLATION

petrodc is written to be compatible with Python 3+. The best way to install is using pip.

```
$ pip install pwploads
```

This will make sure that all the dependencies are installed. This requirements are listed below.

1.1 Requirements

- `numpy`
- `lasio`
- `pandas`
- `matplotlib`
- `plotly`

TOPO-BATHYMETRY DATA

```
petrodc.usgs_eros.elevation(lat=(40, 41), lon=(-65, -60))
```

Function to request data from SRTM30. Bathymetry / Topography (SRTM30) is a global bathymetry/topography data product distributed by the USGS EROS data center. The data product has a resolution of 30 seconds (roughly 1 km).

Keyword Arguments

- **lat** (*tuple*) – with min and max latitude
- **lon** (*tuple*) – with min and max longitude

Returns

ElevationSurface object.

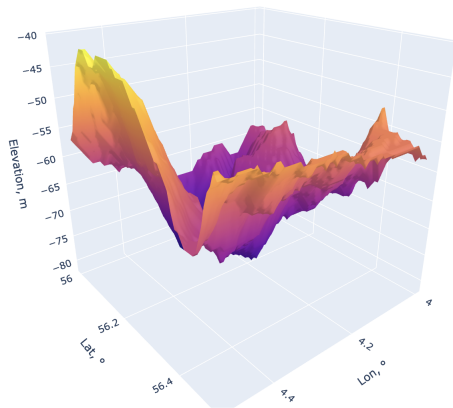
- **Attributes:** df (pandas dataframe): lat, lon and elev columns
- **Methods:** plot: generate a 3D surface plot

2.1 Example

Let's select an squared area within the following coordinates:

- Latitude from 56° to 56.5°
- Longitude from 4° to 4.5°

```
>>> import petrodc.usgs_eros as elev
>>> lat_min = 56
>>> lat_max = 56.5
>>> lon_min = 4
>>> lon_max = 4.5
>>> elev_surface = elev.elevation(lat=(lat_min, lat_max), lon=(lon_min, lon_max)).
↳ plot().show()
```



2.2 Web Application

There is also the web-app based on petrodc:

ATHABASCA WELL LOGS

`petrodc.ags.get_las(number_of_files)`

Function to get well logs data as dataframes from Special Report 006 Athabasca Oil Sands Data McMurray/Wabiskaw Oil Sands Deposit (Alberta, Canada). By Alberta Geological Survey (AGS).

Parameters `number_of_files` (*int*) – number of files to get, max. of 2173

Returns data

Return type dictionary {key -> 'Unique well identifier', value -> dataframe}

`petrodc.ags.plot_log(logs, title=None)`

Function to plot well logs data using matplotlib.

Parameters `logs` (*dataframe*) – each log is a column

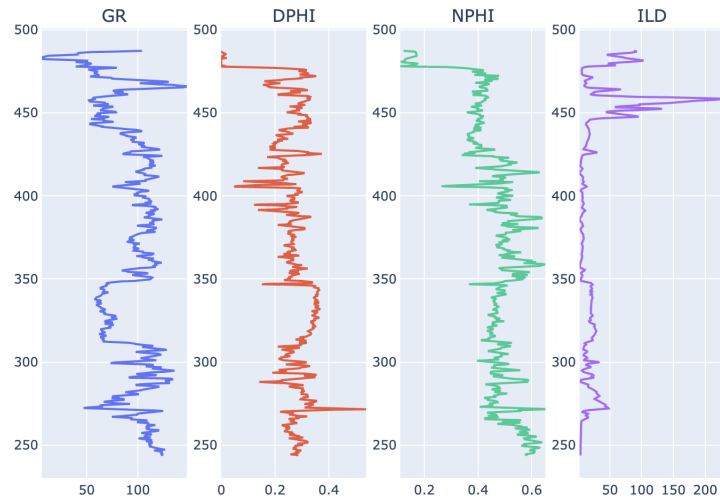
Keyword Arguments `title` (*str or None*) – text to add as title if required

Returns plot

Return type plotly figure

3.1 Example

```
>>> import petrodc.ags as ags
>>> las_files = ags.get_las(2)           # fetch the first 2 well logs from the Special_
↳ Report 006 Athabasca Oil Sands Data
>>> my_las = list(las_files.values())[1] # get its respective df from dict
>>> ags.plot_log(my_las).show()         # plotting logs
```



3.2 Web Application

There is also the web-app based on petrodc:

NORWEGIAN WELLBORE DATA

`petrodc.npd.wellbore(dataset_number)`

Function to request wellbore data from NPD Database.

Parameters `dataset_number` (*int*) – to select the dataset (1: oil samples, 2: NPD ID, 3: lithostratigraphy, 4: history, 5: drilling mud, 6: drill stem tests, 7: documents, 8: cores, 9: core photos, 10: coordinates, 11: casing and leak off, 12: exploration, 13: development, 14: shallow)

Returns requested data

Return type dataframe

4.1 Example

```
>>> import petrodc.npd as npd
>>> data = npd.wellbore(11) # load dataset 11 - casing and leak off
```

	wlbHoleDepth	wlbName	wlbCasingDiameter	wlbNPDID_wellbore	wlbLotMudDensity	wlbCasingDateUpdated	wlbFormationTestType	datesyncNPD	wlbHoleDiameter
0	0.0	6507/6-4 S	30	6725	0.00	11.04.2017	LOT	10.11.2020	36
1	0.0	6406/9-3	36	7141	0.00	10.10.2016	nan	10.11.2020	42
2	46.0	2/5-4	30	259	0.00	11.04.2017	LOT	10.11.2020	36
3	110.0	7/9-1	30	191	0.00	11.04.2017	LOT	10.11.2020	36
4	114.0	8/12-1	30	193	0.00	11.04.2017	LOT	10.11.2020	36
5	117.0	2/3-2	30	169	0.00	11.04.2017	LOT	10.11.2020	36
6	118.0	8/10-2	30	226	0.00	11.04.2017	LOT	10.11.2020	36
7	126.0	2/7-14	30	116	0.00	11.04.2017	LOT	10.11.2020	36
8	123.0	2/3-3	30	198	0.00	11.04.2017	LOT	10.11.2020	36
9	124.0	2/3-1	30	162	0.00	11.04.2017	LOT	10.11.2020	36
10	123.0	9/10-1	30	183	0.00	11.04.2017	LOT	10.11.2020	36
11	125.0	8/4-1	30	302	0.00	11.04.2017	LOT	10.11.2020	36
12	125.0	7/8-1	30	158	0.00	11.04.2017	LOT	10.11.2020	36
13	127.0	2/6-1	30	160	0.00	11.04.2017	LOT	10.11.2020	36
14	130.0	8/11-1	30	304	0.00	11.04.2017	LOT	10.11.2020	36
15	127.0	9/11-1	30	194	0.00	11.04.2017	LOT	10.11.2020	36
16	131.0	16/9-1	30	151	0.00	11.04.2017	LOT	10.11.2020	36
17	128.0	2/5-1	30	178	0.00	11.04.2017	LOT	10.11.2020	36
18	128.0	2/7-24	30	1613	0.00	11.04.2017	LOT	10.11.2020	36
19	127.0	3/7-1	30	292	0.00	11.04.2017	LOT	10.11.2020	36
20	128.0	2/7-15	30	225	0.00	11.04.2017	LOT	10.11.2020	36
21	136.0	2/8-2	30	155	0.00	11.04.2017	LOT	10.11.2020	36
22	129.0	2/7-2	30	187	0.00	11.04.2017	LOT	10.11.2020	36
23	129.0	7/12-1 S	30	157	0.00	10.10.2016	nan	10.11.2020	36
24	130.0	2/10-1 S	30	284	0.00	11.04.2017	LOT	10.11.2020	36
25	131.0	11/10-1	20	170	0.00	11.04.2017	LOT	10.11.2020	26

4.2 Web Application

There is also the web-app based on petrodc:

ABOUT PRO WELL PLAN

Pro Well Plan offers an easy and effective well planning platform for the entire team. Check out [our website](#) to know more about us.

INDEX

E

`elevation()` (*in module `petrodc.usgs_eros`*), 5

G

`get_las()` (*in module `petrodc.ags`*), 7

P

`plot_log()` (*in module `petrodc.ags`*), 7

W

`wellbore()` (*in module `petrodc.npd`*), 9