
The EUPP postprocessing benchmark dataset documentation

Release v1.0

GIE EUMETNET - PP module

Dec 26, 2022

CURRENTLY AVAILABLE DATASETS:

1	EUPPBench datasets	3
2	Base datasets over Europe’s domain	31
3	Using climetlab to access the data	41
4	Indices and tables	43

This website document the EUMETNET postprocessing benchmark datasets, an initiative to provide high-quality datasets to derive easily analysis-ready datasets that can be used to perform benchmarking tasks of weather forecast postprocessing methods.

The main tool to download and manage the data is a Python plugin. It can however convert the data to formats that can then be processed by other languages, and a few line of Python codes suffice to obtain the datasets.

Note:

- **Climetlab plugin version:** 0.2.4
 - **EUPPBench dataset version:** v1.0
 - **Base dataset version:** v1.0
 - **Dataset status:** files/datasets_status:Datasets status
-

EUPPBENCH DATASETS

The *EUPPBench* datasets are available on a small portion of Europe stored in [Zarr](#) format for an easy access allowing for slicing. The forecasts and observations datasets are already paired together, providing analysis-ready data for postprocessing benchmarking purposes.

1.1 Datasets description

There are two main datasets:

1.2 1 - Gridded Data

A forecasts and observations dataset on a regular latitude-longitude grid.

- The gridded EUPPBench postprocessing benchmark dataset contains [ECMWF](#) ensemble and deterministic forecasts over a small domain in Europe, from 45.75° to 53.5° in latitude, and from 2.5° to 10.5° in longitude, and covers the years 2017-2018.
- It also contains the corresponding ERA5 reanalysis for the purpose of providing observations for the benchmark.
- For some dates, it contains also [reforecasts](#) that covers 20 years of past forecasts recomputed with the most recent model version at the given date.
- All the forecasts and reforecasts provided are the noon ECMWF runs.
- The ensemble forecasts and reforecasts also contain by default the control run (the 0-th member).
- The gridded data resolution is 0.25° x 0.25° which corresponds roughly to 25 kilometers.
- Forecasts and reforecasts are 6-hourly, and include the analysis at 00Z.

There are 8 gridded sub-datasets:

1.2.1 1.1 - Extreme Forecast Index

All the [Extreme Forecast Index](#) (EFI) variables can be obtained for each forecast date.

It includes:

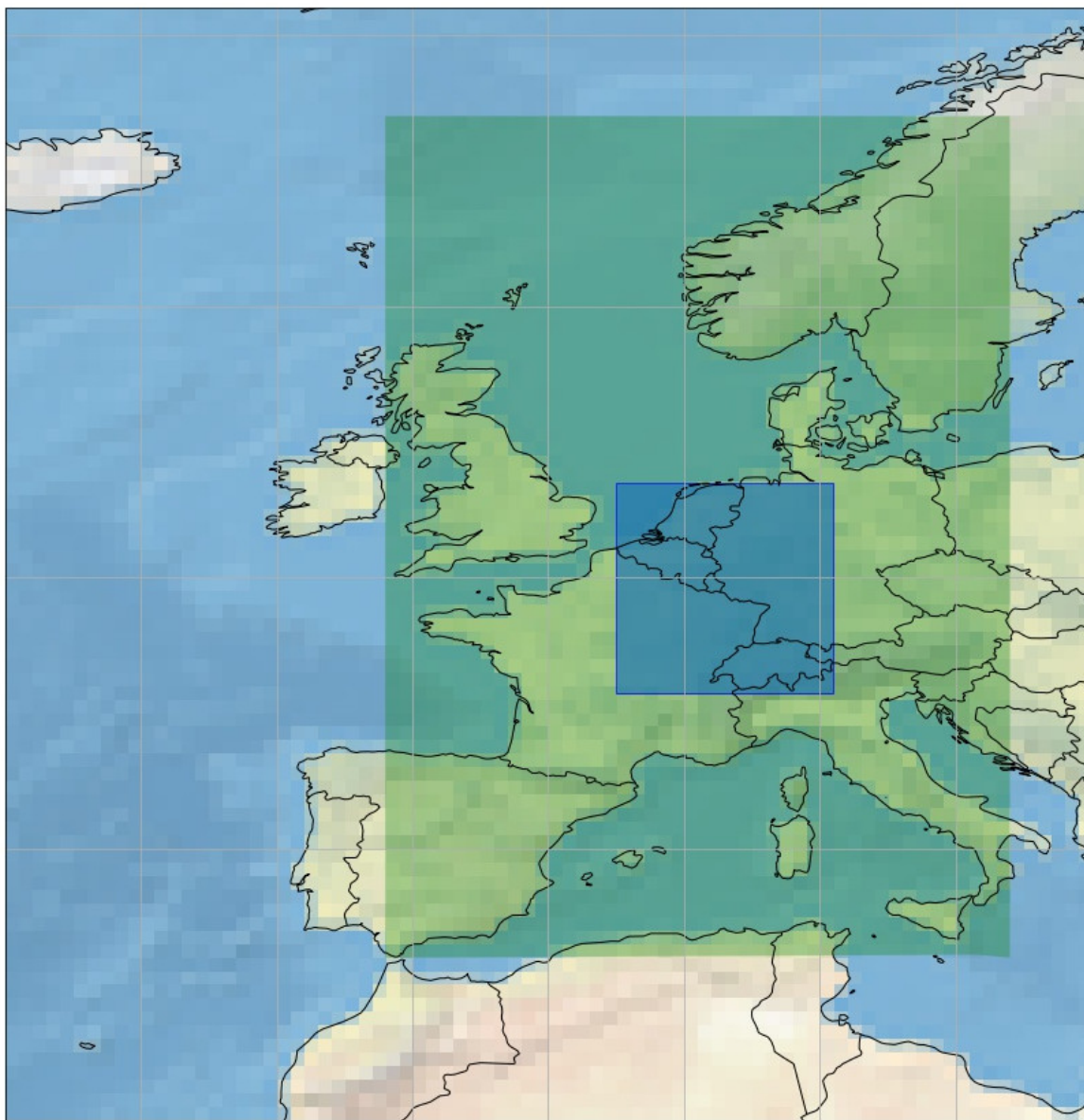


Fig. 1: In blue, the EUPPBench dataset domain inside the *Base datasets over Europe's domain*.

Parameter name	ECMWF key	Remarks
2 metre temperature efi	2ti	
10 metre wind speed efi	10wsi	
10 metre wind gust efi	10fgi	
cape efi	capei	
cape shear efi	capesi	
Maximum temperature at 2m efi	mx2ti	
Minimum temperature at 2m efi	mn2ti	
Snowfall efi	sfi	
Total precipitation efi	tpi	

The EFI are available for the model step ranges (in hours) 0-24, 24-48, 48-72, 72-96, 96-120.

Usage: The EFI variables can be retrieved by calling

```
ds = cml.load_dataset('EUPPBench-training-data-gridded-forecasts-efi')
ds.to_xarray()
```

Example:

```
import climetlab as cml
ds = cml.load_dataset('EUPPBench-training-data-gridded-forecasts-efi')
ds.to_xarray()
```

By downloading data from this dataset, you agree to the terms and conditions defined at

[https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/
DATA_LICENSE](https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/DATA_LICENSE)

If you do not agree with such terms, do not download the data.

```
<xarray.Dataset>
Dimensions:      (number: 1, time: 730, step: 5, surface: 1, latitude: 32, longitude: 33)
Coordinates:
  * latitude      (latitude) float64 53.5 53.25 53.0 52.75 ... 46.25 46.0 45.75
  * longitude      (longitude) float64 2.5 2.75 3.0 3.25 ... 9.75 10.0 10.25 10.5
  * number        (number) int64 0
  * step          (step) timedelta64[ns] 1 days 2 days 3 days 4 days 5 days
  * surface        (surface) float64 0.0
  * time          (time) datetime64[ns] 2017-01-01 2017-01-02 ... 2018-12-31
  valid_time      (time, step) datetime64[ns] ...
Data variables:
  capei           (number, time, step, surface, latitude, longitude) float32 ...
  capesi          (number, time, step, surface, latitude, longitude) float32 ...
  fg10i           (number, time, step, surface, latitude, longitude) float32 ...
  mn2ti           (number, time, step, surface, latitude, longitude) float32 ...
  mx2ti           (number, time, step, surface, latitude, longitude) float32 ...
  sfi             (number, time, step, surface, latitude, longitude) float32 ...
  t2i             (number, time, step, surface, latitude, longitude) float32 ...
  tpi             (number, time, step, surface, latitude, longitude) float32 ...
  ws10i           (number, time, step, surface, latitude, longitude) float32 ...
Attributes:
```

(continues on next page)

(continued from previous page)

Conventions: CF-1.7
 GRIB_centre: ecmf
 GRIB_centreDescription: European Centre for Medium-Range Weather Forecasts
 GRIB_edition: 1
 GRIB_subCentre: 0
 history: 2022-04-26T15:54 GRIB to CDM+CF via cfgrib-0.9.1...
 institution: European Centre for Medium-Range Weather Forecasts

Note: By definition, observations are not available for Extreme Forecast Indices (EFI).

1.2.2 1.2 - Surface variable forecasts

The surface variables can be obtained for each forecast date, both for the ensemble (51 members) and deterministic runs.

It includes:

Parameter name	ECMWF key	Remarks
2 metre temperature	2t/t2m	
10 metre U wind component	10u	
10 metre V wind component	10v	
Total cloud cover	tcc	
100 metre U wind component	100u	Observations not available
100 metre V wind component	100v	Observations not available
Convective available potential energy	cape	
Soil temperature level 1	stl1	
Total column water	tcw	
Total column water vapour	tcwv	
Volumetric soil water layer 1	swvl1	
Snow depth	sd	
Convective inhibition	cin	Observations not available
Visibility	vis	Observations not available

Some missing observations will become available later.

Usage: The surface variables forecasts can be retrieved by calling

```
ds = cml.load_dataset('EUPPBench-training-data-gridded-forecasts-surface', kind)
ds.to_xarray()
```

where the kind argument allows to select the deterministic or ensemble forecasts, by setting it to 'highres' or 'ensemble'.

Example:

```
ds = cml.load_dataset('EUPPBench-training-data-gridded-forecasts-surface', "highres")
ds.to_xarray()
```

By downloading data from this dataset, you agree to the terms and conditions defined at

(continues on next page)

(continued from previous page)

<https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/>
↪ DATA_LICENSE

If you do not agree with such terms, do not download the data.

```
<xarray.Dataset>
Dimensions:                (number: 1, time: 730, step: 21, surface: 1, latitude: 32, ↪
↪ longitude: 33, depthBelowLandLayer: 1)
Coordinates:
  * depthBelowLandLayer    (depthBelowLandLayer) float64 0.0
  * latitude               (latitude) float64 53.5 53.25 53.0 ... 46.25 46.0 45.75
  * longitude              (longitude) float64 2.5 2.75 3.0 ... 10.0 10.25 10.5
  * number                 (number) int64 0
  * step                   (step) timedelta64[ns] 0 days 00:00:00 ... 5 days 00...
  * surface                (surface) float64 0.0
  * time                   (time) datetime64[ns] 2017-01-01 ... 2018-12-31
    valid_time             (time, step) datetime64[ns] ...
Data variables: (12/14)
  cape                    (number, time, step, surface, latitude, longitude) float32 ...
  cin                     (number, time, step, surface, latitude, longitude) float32 ...
  sd                      (number, time, step, surface, latitude, longitude) float32 ...
  stl1                    (number, time, step, depthBelowLandLayer, latitude, longitude) ↪
↪ float32 ...
  swvl1                   (number, time, step, depthBelowLandLayer, latitude, longitude) ↪
↪ float32 ...
  t2m                     (number, time, step, surface, latitude, longitude) float32 ...
  ...                     ...
  tcwv                    (number, time, step, surface, latitude, longitude) float32 ...
  u10                     (number, time, step, surface, latitude, longitude) float32 ...
  u100                    (number, time, step, surface, latitude, longitude) float32 ...
  v10                     (number, time, step, surface, latitude, longitude) float32 ...
  v100                    (number, time, step, surface, latitude, longitude) float32 ...
  vis                     (number, time, step, surface, latitude, longitude) float32 ...
Attributes:
  Conventions:            CF-1.7
  GRIB_centre:            ecmf
  GRIB_centreDescription: European Centre for Medium-Range Weather Forecasts
  GRIB_edition:           1
  GRIB_subCentre:         0
  history:                2022-07-08T12:53 GRIB to CDM+CF via cfgrib-0.9.1...
  institution:            European Centre for Medium-Range Weather Forecasts
```

1.2.3 1.3 - Pressure level variable forecasts

The variables on pressure level can be obtained for each forecast date, both for the ensemble (51 members) and deterministic runs.

It includes:

Parameter name	Level	ECMWF key	Remarks
Temperature	850	t	
U component of wind	700	u	
V component of wind	700	v	
Geopotential	500	z	
Specific humidity	700	q	
Relative humidity	850	r	

Usage: The pressure level variables forecasts can be retrieved by calling

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-EUPP-training-data-gridded-
↳forecasts-pressure', level, kind)
ds.to_xarray()
```

where the `level` argument is the pressure level, as a string or an integer. The `kind` argument allows to select the deterministic or ensemble forecasts, by setting it to `'highres'` or `'ensemble'`.

Example:

```
ds = cml.load_dataset('EUPPBench-training-data-gridded-forecasts-pressure', 500, "highres
↳")
ds.to_xarray()
```

By downloading data from this dataset, you agree to the terms and conditions defined at

[https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/
↳DATA_LICENSE](https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/
↳DATA_LICENSE)

If you do not agree with such terms, do not download the data.

```
<xarray.Dataset>
Dimensions:      (isobaricInhPa: 1, latitude: 32, longitude: 33, number: 1, step: 21,
↳time: 730)
Coordinates:
  * isobaricInhPa  (isobaricInhPa) float64 500.0
  * latitude       (latitude) float64 53.5 53.25 53.0 52.75 ... 46.25 46.0 45.75
  * longitude      (longitude) float64 2.5 2.75 3.0 3.25 ... 10.0 10.25 10.5
  * number         (number) int64 0
  * step           (step) timedelta64[ns] 0 days 00:00:00 ... 5 days 00:00:00
  * time           (time) datetime64[ns] 2017-01-01 2017-01-02 ... 2018-12-31
    valid_time     (time, step) datetime64[ns] ...
Data variables:
  z                (number, time, step, isobaricInhPa, latitude, longitude) float32 ...
Attributes:
  Conventions:      CF-1.7
  GRIB_centre:      ecmf
```

(continues on next page)

(continued from previous page)

```

GRIB_centreDescription: European Centre for Medium-Range Weather Forecasts
GRIB_edition:           1
GRIB_subCentre:         0
history:                2022-03-28T22:50 GRIB to CDM+CF via cfgrib-0.9.1...
institution:            European Centre for Medium-Range Weather Forecasts

```

1.2.4 1.4 - Processed surface variable forecasts

Processed surface variables can be obtained for each forecast date, both for the ensemble (51 members) and deterministic runs. A processed variable is either accumulated, averaged or filtered.

It includes:

Parameter name	ECMWF key	Remarks
Total precipitation	tp6	
Surface sensible heat flux	sshf6	
Surface latent heat flux	slhf6	
Surface net solar radiation	ssr6	
Surface net thermal radiation	str6	
Convective precipitation	cp6	
Maximum temperature at 2 metres	mx2t6	
Minimum temperature at 2 metres	mn2t6	
Surface solar radiation downwards	ssrd6	
Surface thermal radiation downwards	strd6	
10 metre wind gust	10fg6	

All these variables are accumulated or filtered over the last 6 hours preceding a given forecast timestamp. As a consequence, a '6' was added to the ECMWF key to denote this.

Usage: The processed surface variables forecasts can be retrieved by calling

```

ds = cml.load_dataset('EUPPBench-training-data-gridded-forecasts-surface-processed',
↪ kind)
ds.to_xarray()

```

where the `kind` argument allows to select the deterministic or ensemble forecasts, by setting it to 'highres' or 'ensemble'.

Example:

```

ds = cml.load_dataset('EUPPBench-training-data-gridded-forecasts-surface-processed',
↪ "highres")
ds.to_xarray()

```

By downloading data from this dataset, you agree to the terms and conditions defined at

https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/DATA_LICENSE

If you do not agree with such terms, do not download the data.

```
<xarray.Dataset>
Dimensions:      (number: 1, time: 730, step: 20, surface: 1, latitude: 32, longitude: 33)
Coordinates:
  * latitude     (latitude) float64 53.5 53.25 53.0 52.75 ... 46.25 46.0 45.75
  * longitude    (longitude) float64 2.5 2.75 3.0 3.25 ... 9.75 10.0 10.25 10.5
  * number       (number) int64 0
  * step         (step) timedelta64[ns] 0 days 06:00:00 ... 5 days 00:00:00
  * surface      (surface) float64 0.0
  * time         (time) datetime64[ns] 2017-01-01 2017-01-02 ... 2018-12-31
    valid_time   (time, step) datetime64[ns] ...
Data variables:
  cp6            (number, time, step, surface, latitude, longitude) float32 ...
  mn2t6          (number, time, step, surface, latitude, longitude) float32 ...
  mx2t6          (number, time, step, surface, latitude, longitude) float32 ...
  p10fg6         (number, time, step, surface, latitude, longitude) float32 ...
  slhf6          (number, time, step, surface, latitude, longitude) float32 ...
  sshf6          (number, time, step, surface, latitude, longitude) float32 ...
  ssr6           (number, time, step, surface, latitude, longitude) float32 ...
  ssrd6          (number, time, step, surface, latitude, longitude) float32 ...
  str6           (number, time, step, surface, latitude, longitude) float32 ...
  strd6          (number, time, step, surface, latitude, longitude) float32 ...
  tp6            (number, time, step, surface, latitude, longitude) float32 ...
Attributes:
  Conventions:    CF-1.7
  GRIB_centre:    ecmf
  GRIB_centreDescription: European Centre for Medium-Range Weather Forecasts
  GRIB_edition:    1
  GRIB_subCentre:  0
  history:         2022-03-25T11:54 GRIB to CDM+CF via cfgrib-0.9.1...
  institution:    European Centre for Medium-Range Weather Forecasts
```

1.2.5 1.5 - Surface variable reforecasts

The surface variables for the ensemble reforecasts (11 members) can be obtained for each reforecast date. All the variables described at in the section [1.2 - Surface variable forecasts](#) above are available.

Note: The ECMWF reforecasts are only available on dates corresponding to Mondays and Thursdays.

Usage: The surface variables reforecasts can be retrieved by calling

```
ds = cml.load_dataset('EUPPBench-training-data-gridded-reforecasts-surface')
ds.to_xarray()
```

Example:

```
ds = cml.load_dataset('EUPPBench-training-data-gridded-reforecasts-surface')
ds.to_xarray()
```

By downloading data from this dataset, you agree to the terms and conditions defined at

(continues on next page)

(continued from previous page)

<https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/>
 ↪ DATA_LICENSE

If you do not agree with such terms, do not download the data.

```
<xarray.Dataset>
Dimensions:                (time: 209, number: 11, year: 20, step: 21, surface: 1, ↪
↪ latitude: 32, longitude: 33, depthBelowLandLayer: 1)
Coordinates:
  * depthBelowLandLayer    (depthBelowLandLayer) float64 0.0
  * latitude               (latitude) float64 53.5 53.25 53.0 ... 46.25 46.0 45.75
  * longitude              (longitude) float64 2.5 2.75 3.0 ... 10.0 10.25 10.5
  * number                 (number) int64 0 1 2 3 4 5 6 7 8 9 10
  * step                   (step) timedelta64[ns] 0 days 00:00:00 ... 5 days 00...
  * surface                (surface) float64 0.0
  * time                   (time) datetime64[ns] 2017-01-02 ... 2018-12-31
    valid_time              (time, year, step) datetime64[ns] ...
  * year                   (year) int64 1 2 3 4 5 6 7 8 ... 14 15 16 17 18 19 20
Data variables: (12/14)
  cape                    (time, number, year, step, surface, latitude, longitude) ↪
↪ float32 ...
  cin                    (time, number, year, step, surface, latitude, longitude) ↪
↪ float32 ...
  sd                     (time, number, year, step, surface, latitude, longitude) ↪
↪ float32 ...
  stl1                   (time, number, year, step, depthBelowLandLayer, latitude, ↪
↪ longitude) float32 ...
  swvl1                  (time, number, year, step, depthBelowLandLayer, latitude, ↪
↪ longitude) float32 ...
  t2m                    (time, number, year, step, surface, latitude, longitude) ↪
↪ float32 ...
  ...
  tcwv                   (time, number, year, step, surface, latitude, longitude) ↪
↪ float32 ...
  u10                    (time, number, year, step, surface, latitude, longitude) ↪
↪ float32 ...
  u100                   (time, number, year, step, surface, latitude, longitude) ↪
↪ float32 ...
  v10                    (time, number, year, step, surface, latitude, longitude) ↪
↪ float32 ...
  v100                   (time, number, year, step, surface, latitude, longitude) ↪
↪ float32 ...
  vis                    (time, number, year, step, surface, latitude, longitude) ↪
↪ float32 ...
Attributes:
  Conventions:            CF-1.7
  GRIB_centre:            ecmf
  GRIB_centreDescription: European Centre for Medium-Range Weather Forecasts
  GRIB_edition:           1
  GRIB_subCentre:         0
  history:                2022-07-08T08:03 GRIB to CDM+CF via cfgrib-0.9.1...
  institution:            European Centre for Medium-Range Weather Forecasts
```

1.2.6 1.6 - Pressure level variable reforecasts

The variables on pressure level for the ensemble reforecasts (11 members) can be obtained for each reforecast date. All the variables described in the section *1.3 - Pressure level variable forecasts* above are available.

Note: The ECMWF reforecasts are only available on dates corresponding to Mondays and Thursdays.

Usage: The pressure level variables reforecasts can be retrieved by calling

```
ds = cml.load_dataset('EUPPBench-training-data-gridded-reforecasts-pressure', level)
ds.to_xarray()
```

The level argument is the pressure level, as a string or an integer.

Example:

```
ds = cml.load_dataset('EUPPBench-training-data-gridded-reforecasts-pressure', 500)
ds.to_xarray()
```

By downloading data from this dataset, you agree to the terms and conditions defined at

https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/DATA_LICENSE

If you do not agree with such terms, do not download the data.

```
<xarray.Dataset>
Dimensions:      (isobaricInhPa: 1, latitude: 32, longitude: 33, number: 11, step: 21,
↳time: 209, year: 20)
Coordinates:
  * isobaricInhPa  (isobaricInhPa) float64 500.0
  * latitude       (latitude) float64 53.5 53.25 53.0 52.75 ... 46.25 46.0 45.75
  * longitude      (longitude) float64 2.5 2.75 3.0 3.25 ... 10.0 10.25 10.5
  * number         (number) int64 0 1 2 3 4 5 6 7 8 9 10
  * step          (step) timedelta64[ns] 0 days 00:00:00 ... 5 days 00:00:00
  * time          (time) datetime64[ns] 2017-01-02 2017-01-05 ... 2018-12-31
    valid_time    (time, year, step) datetime64[ns] ...
  * year          (year) int64 1 2 3 4 5 6 7 8 9 ... 12 13 14 15 16 17 18 19 20
Data variables:
    z             (time, number, year, step, isobaricInhPa, latitude, longitude)↳
↳float32 ...
Attributes:
  Conventions:      CF-1.7
  GRIB_centre:      ecmf
  GRIB_centreDescription: European Centre for Medium-Range Weather Forecasts
  GRIB_edition:      1
  GRIB_subCentre:    0
  history:           2022-04-15T20:40 GRIB to CDM+CF via cfgrib-0.9.1...
  institution:      European Centre for Medium-Range Weather Forecasts
```


1.2.7 1.7 - Processed surface variable reforecasts

Processed surface variables as described in section *1.4 - Processed surface variable forecasts* can also be obtained as ensemble reforecasts (11 members).

Note: The ECMWF reforecasts are only available on dates corresponding to Mondays and Thursdays.

Usage: The surface variables forecasts can be retrieved by calling

```
ds = cml.load_dataset('EUPPBench-training-data-gridded-reforecasts-surface-processed')
ds.to_xarray()
```

Example:

```
ds = cml.load_dataset('EUPPBench-training-data-gridded-reforecasts-surface-processed')
ds.to_xarray()
```

By downloading data from this dataset, you agree to the terms and conditions defined at

https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/DATA_LICENSE

If you do not agree with such terms, do not download the data.

```
<xarray.Dataset>
Dimensions:      (time: 209, number: 11, year: 20, step: 20, surface: 1, latitude: 32,
↳ longitude: 33)
Coordinates:
  * latitude      (latitude) float64 53.5 53.25 53.0 52.75 ... 46.25 46.0 45.75
  * longitude     (longitude) float64 2.5 2.75 3.0 3.25 ... 9.75 10.0 10.25 10.5
  * number        (number) int64 0 1 2 3 4 5 6 7 8 9 10
  * step          (step) timedelta64[ns] 0 days 06:00:00 ... 5 days 00:00:00
  * surface       (surface) float64 0.0
  * time          (time) datetime64[ns] 2017-01-02 2017-01-05 ... 2018-12-31
  valid_time     (time, year, step) datetime64[ns] ...
  * year          (year) int64 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Data variables:
  cp6            (time, number, year, step, surface, latitude, longitude) float32 ...
  mn2t6          (time, number, year, step, surface, latitude, longitude) float32 ...
  mx2t6          (time, number, year, step, surface, latitude, longitude) float32 ...
  p10fg6         (time, number, year, step, surface, latitude, longitude) float32 ...
  slhf6          (time, number, year, step, surface, latitude, longitude) float32 ...
  sshf6          (time, number, year, step, surface, latitude, longitude) float32 ...
  ssr6           (time, number, year, step, surface, latitude, longitude) float32 ...
  ssrd6          (time, number, year, step, surface, latitude, longitude) float32 ...
  str6           (time, number, year, step, surface, latitude, longitude) float32 ...
  strd6          (time, number, year, step, surface, latitude, longitude) float32 ...
  tp6            (time, number, year, step, surface, latitude, longitude) float32 ...
Attributes:
  Conventions:      CF-1.7
  GRIB_centre:      ecmf
  GRIB_centreDescription: European Centre for Medium-Range Weather Forecasts
```

(continues on next page)

(continued from previous page)

GRIB_edition:	1
GRIB_subCentre:	0
history:	2022-05-04T15:27 GRIB to CDM+CF via cfgrib-0.9.1...
institution:	European Centre for Medium-Range Weather Forecasts

1.2.8 1.8 - Static fields

Various static fields associated to the forecast grid can be obtained, with the purpose of serving as predictors for the postprocessing.

Note: For consistency with the rest of the dataset, we use the ECMWF parameters name, terminology and units here. However, please note that - except for the Surface Geopotential - the fields provided are from other non-ECMWF data sources evaluated at grid points. Currently, the main data source being used is the [Copernicus Land Monitoring Service](#).

It includes:

Parameter name	ECMWF key	Remarks
Land use	landu	Extracted from the CORINE 2018 dataset. Values and associated land type differ from the ECMWF one. Please look at the “legend” entry in the metadata for more details.
Model terrain height	mterh	Extracted from the EU-DEMv1.1 data elevation model dataset.
Surface Geopotential	z	The model orography can be obtained by dividing the surface geopotential by $g=9.80665 \text{ ms}^{-2}$.

Usage: The static fields can be retrieved by calling

```
ds = cml.load_dataset('EUPPBench-training-data-gridded-static-fields', parameter)
ds.to_xarray()
```

where the `parameter` argument is a string with one of the ECMWF keys described above. It is only possible to download one static field per call.

Example:

```
ds = cml.load_dataset('EUPPBench-training-data-gridded-static-fields', 'mterh')
ds.to_xarray()
```

By downloading data from this dataset, you agree to the terms and conditions defined at

https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/DATA_LICENSE

If you do not agree with such terms, do not download the data.

```
<xarray.Dataset>
Dimensions:    (latitude: 32, longitude: 33)
```

(continues on next page)

(continued from previous page)

```
Coordinates:
* latitude  (latitude) float64 53.5 53.25 53.0 52.75 ... 46.25 46.0 45.75
* longitude (longitude) float64 2.5 2.75 3.0 3.25 ... 9.75 10.0 10.25 10.5
Data variables:
  mterh      (latitude, longitude) float64 ...
Attributes:
  full_dataset_metadata:
  history:      Retrieved from https://land.copernicus.eu/imagery...
  source:      European Union, Copernicus Land Monitoring Servic...
```

1.3 2 - Stations Data

A dataset similar to the gridded one, but with station observations.

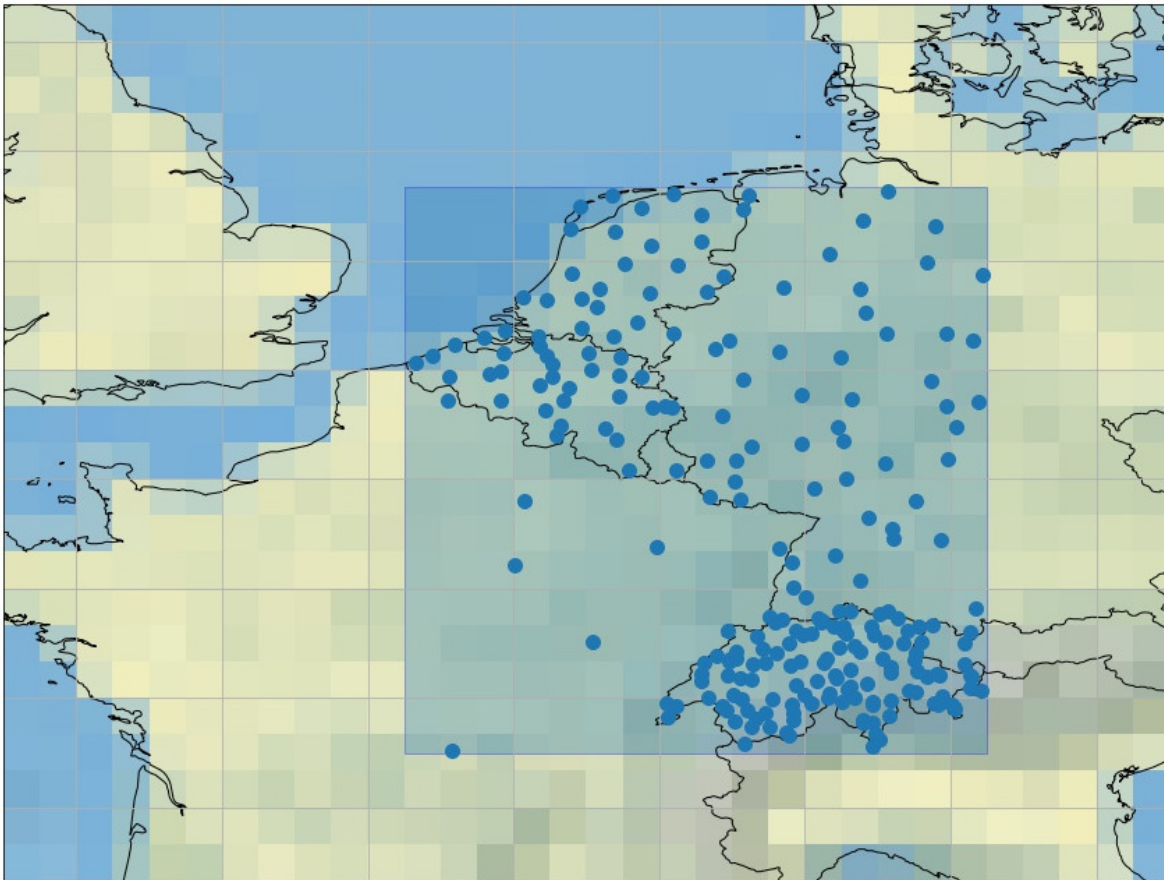


Fig. 2: The stations included in the EUPPBench dataset.

- The stations EUPPBench postprocessing benchmark dataset contains [ECMWF](#) ensemble and deterministic forecasts at the grid point closest to the station locations, and covers the years 2017-2018.

- It also contains the corresponding stations observations.
- For some dates, it contains also [reforecasts](#) that covers 20 years of past forecasts recomputed with the most recent model version at the given date.
- All the forecasts and reforecasts provided are the noon ECMWF runs.
- The ensemble forecasts and reforecasts also contain by default the control run (the 0-th member).
- 5 countries are presently available: Belgium, Austria, France, Germany, The Netherlands.

There are 7 stations sub-datasets:

1.3.1 2.1 - Extreme Forecast Index

All the [Extreme Forecast Index](#) (EFI) variables can be obtained for each forecast date.

The same variables as in section [1.1 - Extreme Forecast Index](#) are available.

The EFI are available for the model step ranges (in hours) 0-24, 24-48, 48-72, 72-96, 96-120.

Usage: The EFI variables can be retrieved by calling

```
ds = cml.load_dataset('EUPPBench-training-data-stations-forecasts-efi', country)
ds.to_xarray()
```

where the country argument must be chosen amongst the list [belgium, austria, france, germany, netherlands].

Example:

```
import climetlab as cml
ds = cml.load_dataset('EUPPBench-training-data-stations-forecasts-efi', 'austria')
ds.to_xarray()
```

By downloading data from this dataset, you agree to the terms and conditions defined at

https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/→DATA_LICENSE

If you do not agree with such terms, do not download the data.

```
<xarray.Dataset>
Dimensions:                (station_id: 4, number: 1, time: 730, step: 5, surface: 1)
Coordinates: (12/15)
  model_altitude           (station_id) float32 ...
  model_land_usage         (station_id) int8 ...
  model_latitude           (station_id) float64 ...
  model_longitude          (station_id) float64 ...
  model_orography          (station_id) float64 ...
  * number                 (number) int64 0
  ...
  station_latitude         (station_id) float64 ...
  station_longitude        (station_id) float64 ...
  station_name             (station_id) <U20 ...
  * step                   (step) timedelta64[ns] 1 days 2 days ... 4 days 5 days
  * surface                (surface) float64 0.0
  * time                   (time) datetime64[ns] 2017-01-01 ... 2018-12-31
```

(continues on next page)

(continued from previous page)

Data variables:

capei	(station_id, number, time, step, surface) float32 ...
capesi	(station_id, number, time, step, surface) float32 ...
fg10i	(station_id, number, time, step, surface) float32 ...
mn2ti	(station_id, number, time, step, surface) float32 ...
mx2ti	(station_id, number, time, step, surface) float32 ...
sfi	(station_id, number, time, step, surface) float32 ...
t2i	(station_id, number, time, step, surface) float32 ...
tpi	(station_id, number, time, step, surface) float32 ...
valid_time	(time, step) datetime64[ns] ...
ws10i	(station_id, number, time, step, surface) float32 ...

Attributes:

Conventions:	CF-1.7
GRIB_centre:	ecmf
GRIB_centreDescription:	European Centre for Medium-Range Weather Forecasts
GRIB_edition:	1
GRIB_subCentre:	0
history:	2022-04-26T15:54 GRIB to CDM+CF via cfgrib-0.9.1...
institution:	European Centre for Medium-Range Weather Forecasts
land usage history:	Retrieved from https://land.copernicus.eu/pan-eu...
land usage legend:	{1: {'label': '111 - Continuous urban fabric', '...
land usage source:	European Union, Copernicus Land Monitoring Servi...
model altitude history:	Retrieved from https://land.copernicus.eu/imager...
model altitude source:	European Union, Copernicus Land Monitoring Servi...

Note: By definition, observations are not available for Extreme Forecast Indices (EFI).

1.3.2 2.2 - Surface variable forecasts

The surface variables can be obtained for each forecast date, both for the ensemble (51 members) and deterministic runs.

The same variables as in section [1.2 - Surface variable forecasts](#) are available.

Note: Only the variables `t2m`, `vis` and `tcc` have presently station observations.

Usage: The surface variables forecasts can be retrieved by calling

```
ds = cml.load_dataset('EUPPBench-training-data-stations-forecasts-surface', kind,
    ↪country)
ds.to_xarray()
```

where the `kind` argument allows to select the deterministic or ensemble forecasts, by setting it to `'highres'` or `'ensemble'`. The `country` argument must be chosen amongst the list `[belgium, austria, france, germany, netherlands]`.

Example:

```
ds = cml.load_dataset('EUPPBench-training-data-stations-forecasts-surface', "highres",
↳ "austria")
ds.to_xarray()
```

By downloading data from this dataset, you agree to the terms and conditions defined at

https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/↳DATA_LICENSE

If you do not agree with such terms, do not download the data.

```
<xarray.Dataset>
Dimensions:                (station_id: 4, number: 1, time: 730, step: 21, surface: 1,↳
↳ depthBelowLandLayer: 1)
Coordinates: (12/16)
  * depthBelowLandLayer    (depthBelowLandLayer) float64 0.0
  model_altitude           (station_id) float32 ...
  model_land_usage         (station_id) int8 ...
  model_latitude           (station_id) float64 ...
  model_longitude          (station_id) float64 ...
  model_orography          (station_id) float64 ...
  ...
  station_latitude         (station_id) float64 ...
  station_longitude        (station_id) float64 ...
  station_name             (station_id) <U20 ...
  * step                   (step) timedelta64[ns] 0 days 00:00:00 ... 5 days 00...
  * surface                (surface) float64 0.0
  * time                   (time) datetime64[ns] 2017-01-01 ... 2018-12-31
Data variables: (12/15)
  cape                    (station_id, number, time, step, surface) float32 ...
  cin                     (station_id, number, time, step, surface) float32 ...
  sd                      (station_id, number, time, step, surface) float32 ...
  stl1                    (station_id, number, time, step, depthBelowLandLayer) float32 ..
↳ .
  swvl1                   (station_id, number, time, step, depthBelowLandLayer) float32 ..
↳ .
  t2m                     (station_id, number, time, step, surface) float32 ...
  ...
  u10                     (station_id, number, time, step, surface) float32 ...
  u100                    (station_id, number, time, step, surface) float32 ...
  v10                     (station_id, number, time, step, surface) float32 ...
  v100                    (station_id, number, time, step, surface) float32 ...
  valid_time              (time, step) datetime64[ns] ...
  vis                     (station_id, number, time, step, surface) float32 ...
Attributes:
  Conventions:            CF-1.7
  GRIB_centre:            ecmf
  GRIB_centreDescription: European Centre for Medium-Range Weather Forecasts
  GRIB_edition:           1
  GRIB_subCentre:         0
  history:                2022-07-08T12:53 GRIB to CDM+CF via cfgrib-0.9.1...
  institution:            European Centre for Medium-Range Weather Forecasts
```

(continues on next page)

(continued from previous page)

land usage history:	Retrieved from https://land.copernicus.eu/pan-eu...
land usage legend:	{1: {'label': '111 - Continuous urban fabric', '...
land usage source:	European Union, Copernicus Land Monitoring Servi...
model altitude history:	Retrieved from https://land.copernicus.eu/imager...
model altitude source:	European Union, Copernicus Land Monitoring Servi...

1.3.3 2.3 - Pressure level variable forecasts

The variables on pressure level can be obtained for each forecast date, both for the ensemble (51 members) and deterministic runs.

The same variables as in section 2.3 - *Pressure level variable forecasts* are available.

Note: For obvious reasons, station observations are not available on pressure levels.

Usage: The pressure level variables forecasts can be retrieved by calling

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-EUPP-training-data-stations-
↳forecasts-pressure', level, kind, country)
ds.to_xarray()
```

where the `level` argument is the pressure level, as a string or an integer. The `kind` argument allows to select the deterministic or ensemble forecasts, by setting it to 'highres' or 'ensemble'. The `country` argument must be chosen amongst the list [belgium, austria, france, germany, netherlands].

Example:

```
ds = cml.load_dataset('EUPPBench-training-data-stations-forecasts-pressure', 500,
↳"highres", "austria")
ds.to_xarray()
```

By downloading data from this dataset, you agree to the terms and conditions defined at

[https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/
↳DATA_LICENSE](https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/
↳DATA_LICENSE)

If you do not agree with such terms, do not download the data.

```
<xarray.Dataset>
Dimensions:                (isobaricInhPa: 1, station_id: 4, number: 1, step: 21, time: 730)
Coordinates: (12/15)
  * isobaricInhPa          (isobaricInhPa) float64 500.0
    model_altitude         (station_id) float32 ...
    model_land_usage       (station_id) int8 ...
    model_latitude         (station_id) float64 ...
    model_longitude        (station_id) float64 ...
    model_orography        (station_id) float64 ...
    ...                    ...
    station_land_usage     (station_id) int8 ...
    station_latitude       (station_id) float64 ...
```

(continues on next page)

(continued from previous page)

```

station_longitude (station_id) float64 ...
station_name      (station_id) <U20 ...
* step            (step) timedelta64[ns] 0 days 00:00:00 ... 5 days 00:...
* time            (time) datetime64[ns] 2017-01-01 ... 2018-12-31
Data variables:
  valid_time      (time, step) datetime64[ns] ...
  z               (station_id, number, time, step, isobaricInhPa) float32 ...
Attributes:
  Conventions:    CF-1.7
  GRIB_centre:    ecmf
  GRIB_centreDescription: European Centre for Medium-Range Weather Forecasts
  GRIB_edition:    1
  GRIB_subCentre:  0
  history:        2022-03-28T22:50 GRIB to CDM+CF via cfgrib-0.9.1...
  institution:    European Centre for Medium-Range Weather Forecasts
  land usage history: Retrieved from https://land.copernicus.eu/pan-eu...
  land usage legend: {1: {'label': '111 - Continuous urban fabric', '...
  land usage source: European Union, Copernicus Land Monitoring Servi...
  model altitude history: Retrieved from https://land.copernicus.eu/imager...
  model altitude source: European Union, Copernicus Land Monitoring Servi...
```

1.3.4 2.4 - Processed surface variable forecasts

Processed surface variables can be obtained for each forecast date, both for the ensemble (51 members) and deterministic runs. A processed variable is either accumulated, averaged or filtered.

The same variables as in section 2.4 - *Processed surface variable forecasts* are available.

Note: Only the variables tp6 and 10fg6 have presently station observations.

Usage: The processed surface variables forecasts can be retrieved by calling

```
ds = cml.load_dataset('EUPPBench-training-data-stations-forecasts-surface-processed',
↪ kind, country)
ds.to_xarray()
```

where the kind argument allows to select the deterministic or ensemble forecasts, by setting it to 'highres' or 'ensemble'. The country argument must be chosen amongst the list [belgium, austria, france, germany, netherlands].

Example:

```
ds = cml.load_dataset('EUPPBench-training-data-stations-forecasts-surface-processed',
↪ "highres", "austria")
ds.to_xarray()
```

By downloading data from this dataset, you agree to the terms and conditions defined at

https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/DATA_LICENSE

(continues on next page)

(continued from previous page)

If you do not agree with such terms, do not download the data.

```

<xarray.Dataset>
Dimensions:                (station_id: 4, number: 1, time: 730, step: 20, surface: 1)
Coordinates: (12/16)
  model_altitude           (station_id) float32 ...
  model_land_usage         (station_id) int8 ...
  model_latitude           (station_id) float64 ...
  model_longitude          (station_id) float64 ...
  model_orography          (station_id) float64 ...
  * number                 (number) int64 0
  ...
  station_longitude        (station_id) float64 ...
  station_name             (station_id) <U20 ...
  * step                   (step) timedelta64[ns] 0 days 06:00:00 ... 5 days 00:...
  * surface                (surface) float64 0.0
  * time                   (time) datetime64[ns] 2017-01-01 ... 2018-12-31
  valid_time               (time, step) datetime64[ns] ...
Data variables:
  cp6                     (station_id, number, time, step, surface) float32 ...
  mn2t6                   (station_id, number, time, step, surface) float32 ...
  mx2t6                   (station_id, number, time, step, surface) float32 ...
  p10fg6                  (station_id, number, time, step, surface) float32 ...
  slhf6                   (station_id, number, time, step, surface) float32 ...
  sshf6                   (station_id, number, time, step, surface) float32 ...
  ssr6                    (station_id, number, time, step, surface) float32 ...
  ssrd6                   (station_id, number, time, step, surface) float32 ...
  str6                    (station_id, number, time, step, surface) float32 ...
  strd6                   (station_id, number, time, step, surface) float32 ...
  tp6                     (station_id, number, time, step, surface) float32 ...
Attributes:
  Conventions:            CF-1.7
  GRIB_centre:            ecmf
  GRIB_centreDescription: European Centre for Medium-Range Weather Forecasts
  GRIB_edition:           1
  GRIB_subCentre:         0
  history:                2022-03-25T11:54 GRIB to CDM+CF via cfgrib-0.9.1...
  institution:            European Centre for Medium-Range Weather Forecasts
  land usage history:      Retrieved from https://land.copernicus.eu/pan-eu...
  land usage legend:       {1: {'label': '111 - Continuous urban fabric', '...
  land usage source:       European Union, Copernicus Land Monitoring Servi...
  model altitude history:  Retrieved from https://land.copernicus.eu/imager...
  model altitude source:   European Union, Copernicus Land Monitoring Servi...

```

1.3.5 2.5 - Surface variable reforecasts

The surface variables for the ensemble reforecasts (11 members) can be obtained for each reforecast date. All the variables described at in the section [1.2 - Surface variable forecasts](#) above are available.

Note: The ECMWF reforecasts are only available on dates corresponding to Mondays and Thursdays.

Note: Only the variables `t2m`, `vis` and `tcc` have presently station observations.

Usage: The surface variables reforecasts can be retrieved by calling

```
ds = cml.load_dataset('EUPPBench-training-data-stations-reforecasts-surface', country)
ds.to_xarray()
```

where the `country` argument must be chosen amongst the list `[belgium, austria, france, germany, netherlands]`.

Example:

```
ds = cml.load_dataset('EUPPBench-training-data-stations-reforecasts-surface', "austria")
ds.to_xarray()
```

By downloading data from this dataset, you agree to the terms and conditions defined at

https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/DATA_LICENSE

If you do not agree with such terms, do not download the data.

```
<xarray.Dataset>
Dimensions:                (station_id: 4, time: 209, number: 11, year: 20, step: 21,
↳ surface: 1, depthBelowLandLayer: 1)
Coordinates: (12/17)
  * depthBelowLandLayer    (depthBelowLandLayer) float64 0.0
  model_altitude           (station_id) float32 ...
  model_land_usage         (station_id) int8 ...
  model_latitude           (station_id) float64 ...
  model_longitude          (station_id) float64 ...
  model_orography          (station_id) float64 ...
  ...
  station_longitude        (station_id) float64 ...
  station_name             (station_id) <U20 ...
  * step                   (step) timedelta64[ns] 0 days 00:00:00 ... 5 days 00...
  * surface                (surface) float64 0.0
  * time                   (time) datetime64[ns] 2017-01-02 ... 2018-12-31
  * year                   (year) int64 1 2 3 4 5 6 7 8 ... 14 15 16 17 18 19 20
Data variables: (12/15)
  cape                    (station_id, time, number, year, step, surface) float32 ...
  cin                    (station_id, time, number, year, step, surface) float32 ...
  sd                     (station_id, time, number, year, step, surface) float32 ...
  stl1                   (station_id, time, number, year, step, depthBelowLandLayer)
↳ float32 ...
```

(continues on next page)

(continued from previous page)

```

    swvl1                (station_id, time, number, year, step, depthBelowLandLayer)
↪ float32 ...
    t2m                  (station_id, time, number, year, step, surface) float32 ...
    ...
    u10                  (station_id, time, number, year, step, surface) float32 ...
    u100                 (station_id, time, number, year, step, surface) float32 ...
    v10                  (station_id, time, number, year, step, surface) float32 ...
    v100                 (station_id, time, number, year, step, surface) float32 ...
    valid_time           (time, year, step) datetime64[ns] ...
    vis                  (station_id, time, number, year, step, surface) float32 ...
Attributes:
  Conventions:          CF-1.7
  GRIB_centre:          ecmf
  GRIB_centreDescription: European Centre for Medium-Range Weather Forecasts
  GRIB_edition:         1
  GRIB_subCentre:       0
  history:              2022-07-08T08:03 GRIB to CDM+CF via cfgrib-0.9.1...
  institution:          European Centre for Medium-Range Weather Forecasts
  land usage history:    Retrieved from https://land.copernicus.eu/pan-eu...
  land usage legend:     {1: {'label': '111 - Continuous urban fabric', '...
  land usage source:     European Union, Copernicus Land Monitoring Servi...
  model altitude history: Retrieved from https://land.copernicus.eu/imager...
  model altitude source: European Union, Copernicus Land Monitoring Servi...

```

1.3.6 2.6 - Pressure level variable reforecasts

The variables on pressure level for the ensemble reforecasts (11 members) can be obtained for each reforecast date. All the variables described in the section [1.3 - Pressure level variable reforecasts](#) above are available.

Note: The ECMWF reforecasts are only available on dates corresponding to Mondays and Thursdays.

Note: For obvious reasons, station observations are not available on pressure levels.

Usage: The pressure level variables reforecasts can be retrieved by calling

```

ds = cml.load_dataset('EUPPBench-training-data-stations-reforecasts-pressure', level,
↪ country)
ds.to_xarray()

```

The `level` argument is the pressure level, as a string or an integer. The `country` argument must be chosen amongst the list [belgium, austria, france, germany, netherlands].

Example:

```

ds = cml.load_dataset('EUPPBench-training-data-stations-reforecasts-pressure', 500,
↪ "austria")
ds.to_xarray()

```

By downloading data from this dataset, you agree to the terms and conditions defined at

https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/→DATA_LICENSE

If you do not agree with such terms, do not download the data.

```
<xarray.Dataset>
Dimensions:                (isobaricInhPa: 1, station_id: 4, number: 11, step: 21, time: 209, year: 20)
Coordinates: (12/16)
  * isobaricInhPa          (isobaricInhPa) float64 500.0
    model_altitude         (station_id) float32 ...
    model_land_usage       (station_id) int8 ...
    model_latitude         (station_id) float64 ...
    model_longitude        (station_id) float64 ...
    model_orography        (station_id) float64 ...
    ...
    station_latitude       (station_id) float64 ...
    station_longitude      (station_id) float64 ...
    station_name           (station_id) <U20 ...
  * step                   (step) timedelta64[ns] 0 days 00:00:00 ... 5 days 00:...
  * time                   (time) datetime64[ns] 2017-01-02 ... 2018-12-31
  * year                   (year) int64 1 2 3 4 5 6 7 8 ... 13 14 15 16 17 18 19 20
Data variables:
  valid_time               (time, year, step) datetime64[ns] ...
  z                        (station_id, time, number, year, step, isobaricInhPa) float32 ...
Attributes:
  Conventions:             CF-1.7
  GRIB_centre:             ecmf
  GRIB_centreDescription:  European Centre for Medium-Range Weather Forecasts
  GRIB_edition:            1
  GRIB_subCentre:         0
  history:                 2022-04-15T20:40 GRIB to CDM+CF via cfgrib-0.9.1...
  institution:            European Centre for Medium-Range Weather Forecasts
  land usage history:      Retrieved from https://land.copernicus.eu/pan-eu...
  land usage legend:      {1: {'label': '111 - Continuous urban fabric', '...
  land usage source:      European Union, Copernicus Land Monitoring Servi...
  model altitude history:  Retrieved from https://land.copernicus.eu/imager...
  model altitude source:  European Union, Copernicus Land Monitoring Servi...
```

1.3.7 2.7 - Processed surface variable reforecasts

Processed surface variables as described in section *1.4 - Processed surface variable forecasts* can also be obtained as ensemble reforecasts (11 members).

Note: The ECMWF reforecasts are only available on dates corresponding to Mondays and Thursdays.

Note: Only the variables `tp6` and `10fg6` have presently station observations.

Usage: The surface variables forecasts can be retrieved by calling

```
ds = cml.load_dataset('EUPPBench-training-data-stations-reforecasts-surface-processed',
↳country)
ds.to_xarray()
```

The country argument must be chosen amongst the list [belgium, austria, france, germany, netherlands].

Example:

```
ds = cml.load_dataset('EUPPBench-training-data-stations-reforecasts-surface-processed',
↳"austria")
ds.to_xarray()
```

By downloading data from this dataset, you agree to the terms and conditions defined at

https://github.com/Climdyn/climetlab-eumetnet-postprocessing-benchmark/blob/main/↳DATA_LICENSE

If you do not agree with such terms, do not download the data.

```
<xarray.Dataset>
Dimensions:                (station_id: 4, time: 209, number: 11, year: 20, step: 20,
↳surface: 1)
Coordinates: (12/17)
  model_altitude            (station_id) float32 ...
  model_land_usage          (station_id) int8 ...
  model_latitude            (station_id) float64 ...
  model_longitude           (station_id) float64 ...
  model_orography           (station_id) float64 ...
  * number                  (number) int64 0 1 2 3 4 5 6 7 8 9 10
  ...
  station_name              (station_id) <U20 ...
  * step                    (step) timedelta64[ns] 0 days 06:00:00 ... 5 days 00:...
  * surface                 (surface) float64 0.0
  * time                    (time) datetime64[ns] 2017-01-02 ... 2018-12-31
  valid_time                (time, year, step) datetime64[ns] ...
  * year                    (year) int64 1 2 3 4 5 6 7 8 ... 13 14 15 16 17 18 19 20
Data variables:
  cp6                      (station_id, time, number, year, step, surface) float32 ...
  mn2t6                    (station_id, time, number, year, step, surface) float32 ...
  mx2t6                    (station_id, time, number, year, step, surface) float32 ...
  p10fg6                   (station_id, time, number, year, step, surface) float32 ...
  slhf6                    (station_id, time, number, year, step, surface) float32 ...
  sshf6                    (station_id, time, number, year, step, surface) float32 ...
  ssr6                     (station_id, time, number, year, step, surface) float32 ...
  ssrd6                    (station_id, time, number, year, step, surface) float32 ...
  str6                     (station_id, time, number, year, step, surface) float32 ...
  strd6                    (station_id, time, number, year, step, surface) float32 ...
  tp6                      (station_id, time, number, year, step, surface) float32 ...
Attributes:
  Conventions:              CF-1.7
  GRIB_centre:              ecmf
  GRIB_centreDescription:   European Centre for Medium-Range Weather Forecasts
```

(continues on next page)

(continued from previous page)

```

GRIB_edition:      1
GRIB_subCentre:    0
history:           2022-05-04T15:27 GRIB to CDM+CF via cfgrib-0.9.1...
institution:       European Centre for Medium-Range Weather Forecasts
land usage history: Retrieved from https://land.copernicus.eu/pan-eu...
land usage legend: {1: {'label': '111 - Continuous urban fabric', '...
land usage source: European Union, Copernicus Land Monitoring Servi...
model altitude history: Retrieved from https://land.copernicus.eu/imager...
model altitude source: European Union, Copernicus Land Monitoring Servi...

```

1.4 3 - Getting the observations corresponding to the (re)forecasts

Once obtained, the observations (if available) corresponding to the downloaded forecasts or reforecasts can be retrieved in the `xarray` format by using the `get_observations_as_xarray` method:

```

ds = cml.load_dataset('EUPPBench-training-data-stations-reforecasts-surface-processed',
    ↪ "austria")
obs = ds.get_observations_as_xarray()
obs

```

```

<xarray.Dataset>
Dimensions:      (station_id: 4, time: 209, year: 20, step: 20)
Coordinates:
  altitude      (station_id) float64 ...
  land_usage    (station_id) int8 ...
  latitude      (station_id) float64 ...
  longitude     (station_id) float64 ...
  * station_id  (station_id) int64 11101 11105 11308 11312
  station_name  (station_id) <U20 ...
  * step        (step) timedelta64[ns] 0 days 06:00:00 ... 5 days 00:00:00
  * time        (time) datetime64[ns] 2017-01-02 2017-01-05 ... 2018-12-31
  * year        (year) int64 1 2 3 4 5 6 7 8 9 ... 12 13 14 15 16 17 18 19 20
Data variables:
  p10fg6        (time, year, step, station_id) float64 ...
  tp6           (time, year, step, station_id) float64 ...
Attributes:
  full_dataset_metadata:
  history:        Gathered and compiled by Markus Dabernig (ZAMG).
  land usage history: Retrieved from https://land.copernicus.eu/pan-eur...
  land usage legend: {1: {'label': '111 - Continuous urban fabric', 'n...
  land usage source: European Union, Copernicus Land Monitoring Servic...
  source:         ZAMG, Zentralanstalt für Meteorologie und Geodyna...

```

1.5 4 - Explanation of the metadata

For all data, attributes specifying the sources and the license are always present. Depending on the kind of dataset, dimensions and information are embedded in the data as follow:

1.5.1 Gridded data

The following metadata are available in the gridded forecast, reforecast and observation data:

Metadata	Description
latitude	Latitude of the grid points.
longitude	Longitude of the grid points.
depthBelowLand-Layer	Layer below the surface (valid for some variables only, here there is only the upper surface level).
number	Number of the ensemble member. The 0-th member is the control run. Also present in observation for compatibility reasons, but set to 0.
time	Forecast or reforecast date (reforecasts are only issued on Mondays and Thursdays).
year	Dimension to identify the year in the past, year=1 means a forecast valid 20 years ago at the reforecast day and month, year=20 means a forecast valid one year before the reforecast date. Only valid for reforecasts.
step	Step of the forecast (the lead time).
surface	Layer of the variable considered (here there is just one, at the surface).
isobaricInhPa	Pressure level in hectopascal (or millibar).
valid_time	Actual time and date of the corresponding forecast data.

Note: **Bold** metadata denotes dimensions indexing the datasets.

1.5.2 Stations data

For station forecast and reforecast data, the following metadata are available:

Metadata	Description
station_latitude	Latitude of the station.
station_longitude	Longitude of the station.
station_altitude	Altitude of the station (in meter).
station_id	Unique identifier of the station.
depthBelowLand-Layer	Layer below the surface (valid for some variables only, here there is only the upper surface level).
number	Number of the ensemble member. The 0-th member is the control run. Also present in observation for compatibility reasons, but set to 0.
time	Forecast or reforecast date (reforecasts are only issued on Mondays and Thursdays).
year	Dimension to identify the year in the past, year=1 means a forecast valid 20 years ago at the reforecast day and month, year=20 means a forecast valid one year before the reforecast date. Only valid for reforecasts.
step	Step of the forecast (the lead time).
surface	Layer of the variable considered (here there is just one, at the surface).
isobaricInhPa	Pressure level in hectopascal (or millibar).
station_land_usage	Land usage at the station location, extracted from the CORINE 2018 dataset.
station_name	Name of the station.
model_latitude	Latitude of the model grid point.
model_longitude	Longitude of the model grid point.
model_altitude	True altitude (in meter) of the model grid point, extracted from the EU-DEMv1.1 data elevation model dataset.
model_orography	Surface height (in meter) in the model at the model grid point.
model_land_usage	Land usage at the model grid point, extracted from the CORINE 2018 dataset.
valid_time	Actual time and date of the corresponding forecast data.

Note: The metadata with 'model' in their name indicate properties of the model grid point the closest to the station location, and at which the forecasts corresponding to the station observations was extracted from the gridded dataset.

For the station observations, the following metadata are available:

Metadata	Description
altitude	Altitude of the station (in meter).
land_usage	Land usage at the station location, extracted from the CORINE 2018 dataset.
latitude	Latitude of the station.
longitude	Longitude of the station.
station_id	Unique identifier of the station.
station_name	Name of the station.
step	Step of the forecast (the lead time).
time	Forecast or reforecast date (reforecasts are only issued on Mondays and Thursdays).

1.6 5 - Major ECMWF model changes

In 2017 and 2018, there were 2 model changes of the ECMWF model on total:

Implementation date	Summary of changes	Resolution	Full IFS documentation
05-Jun-2018	Cycle 45r1	Unchanged	Cycle 45r1 full documentation
11-Jul-17	Cycle 43r3	Unchanged	Cycle 43r3 full documentation

Source: <https://www.ecmwf.int/en/forecasts/documentation-and-support/changes-ecmwf-model>

1.7 Tips & Tricks

1.7.1 Saving the data to a NetCDF file

This is particularly useful if one wants to reuse the data with another programming language. For example, if one has downloaded the observations shown in section 3 - *Getting the observations corresponding to the (re)forecasts*, one can save them to disk by using the `xarray.Dataset.to_netcdf()` functionality of the `xarray Dataset`:

```
ds = cml.load_dataset('EUPPBench-training-data-stations-reforecasts-surface-processed',
    ↪ "austria")
obs = ds.get_observations_as_xarray()
obs.to_netcdf('austria_reforecasts.nc')
```

1.7.2 Finding the units of a given data

In general, we align with the units of the `ECMWF` data. You can find the particular units of a given data by clicking on the parameter's name in the table above. For many variables, the units are also available in the metadata of the forecasts. For example, the following code snippet show how to retrieve the units of surface variable in the station dataset:

```
ds = cml.load_dataset('EUPPBench-training-data-stations-reforecasts-surface', "austria")
fcs = ds.to_xarray()
fcs.v100.units
```

```
'm s**-1'
```

1.8 Data License

See the `DATA_LICENSE` file.

Station observations were provided by European National Meteorological Services within the framework of their open data policy, and are sourced in the metadata of the corresponding datasets.

Swiss station data are part of this dataset but are presently restricted. These station data may be obtained from `IDAWEB` at MeteoSwiss and we are not entitled to provide it online. Registration with IDAWEB can be initiated [here](#). Please also read [these information](#).

BASE DATASETS OVER EUROPE'S DOMAIN

These are the global datasets available on a large portion of Europe which constitute the base to develop more specific benchmark datasets.

Warning: Access to the forecasts and observations is currently time-granular: these datasets cannot be sliced over the issuance time dimension.

2.1 Datasets description

There are two main datasets:

2.2 1 - Gridded Data

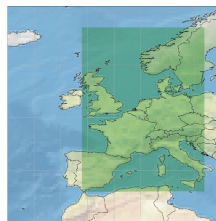


Fig. 1: The global EUMETNET postprocessing benchmark domain.

- The gridded main Eumetnet postprocessing benchmark dataset contains **ECMWF** ensemble and deterministic forecasts over a large portion of Europe, from 36 to 67° in latitude and from -6 to 17° of longitude, and covers the years 2017-2018.
- It also contains the corresponding ERA5 reanalysis for the purpose of providing observations for the benchmark.
- For some dates, it contains also **reforecasts** that covers 20 years of past forecasts recomputed with the most recent model version.
- All the forecasts and reforecasts provided are the noon ECMWF runs.
- The ensemble forecasts and reforecasts also contain by default the control run (the 0-th member).
- The gridded data resolution is 0.25° x 0.25° which corresponds roughly to 25 kilometers.
- **Please note that you can presently only retrieve one forecast date** for each `climetlab.load_dataset` call.

There are 8 gridded sub-datasets:

2.2.1 1.1 - Extreme Forecast Index

All the [Extreme Forecast Index](#) (EFI) variables can be obtained for each forecast date.

It includes:

Parameter name	ECMWF key	Remarks
2 metre temperature efi	2ti	
10 metre wind speed efi	10wsi	
10 metre wind gust efi	10fgi	
cape efi	capei	
cape shear efi	capei	
Maximum temperature at 2m efi	mx2ti	
Minimum temperature at 2m efi	mn2ti	
Snowfall efi	sfi	
Total precipitation efi	tpi	

The EFI are available for the model step range (in hours) 0-24, 24-48, 48-72, 72-96, 96-120, 120-144 and 144-168.

Usage: The EFI variables can be retrieved by calling

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-forecasts-  
→efi', date, parameter)  
ds.to_xarray()
```

where the `date` argument is a string with a single date, and the `parameter` argument is a string or a list of string with the ECMWF keys described above. Setting 'all' as `parameter` download all the EFI parameters.

Example:

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-forecasts-  
→efi', "2017-12-02", "2ti")  
ds.to_xarray()
```

Note: By definition, observations are not available for Extreme Forecast Indices (EFI).

2.2.2 1.2 - Surface variable forecasts

The surface variables can be obtained for each forecast date, both for the ensemble (51 members) and deterministic runs.

It includes:

Parameter name	ECMWF key	Remarks
2 metre temperature	2t	
10 metre U wind component	10u	
10 metre V wind component	10v	
Total cloud cover	tcc	
100 metre U wind component anomaly	100ua	Observations not available
100 metre V wind component anomaly	100va	Observations not available
Convective available potential energy	cape	
Soil temperature level 1	stl1	
Total column water	tcw	
Total column water vapour	tcwv	
Volumetric soil water layer 1	swvl1	
Snow depth	sd	
Convective inhibition	cin	Observations not available
Visibility	vis	Observations not available

Some missing observations will become available later.

The forecasts are available for the model steps (in hours) 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 93, 96, 99, 102, 105, 108, 111, 114, 117, 120, 123, 126, 129, 132, 135, 138, 141, 144, 150, 156, 162, 168, 174, 180, 186, 192, 198, 204, 210, 216, 222, 228, 234 and 240. All the steps are automatically retrieved.

Usage: The surface variables forecasts can be retrieved by calling

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-forecasts-
→surface', date, parameter, kind)
ds.to_xarray()
```

where the `date` argument is a string with a single date, and the `parameter` argument is a string or a list of string with the ECMWF keys described above. Setting 'all' as `parameter` download all the surface parameters. The `kind` argument allows to select the deterministic or ensemble forecasts, by setting it to 'highres' or 'ensemble'.

Example:

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-forecasts-
→surface', "2017-12-02", "sd", "highres")
ds.to_xarray()
```

2.2.3 1.3 - Pressure level variable forecasts

The variables on pressure level can be obtained for each forecast date, both for the ensemble (51 members) and deterministic runs.

It includes:

Parameter name	Level	ECMWF key	Remarks
Temperature	850	t	
U component of wind	700	u	
V component of wind	700	v	
Geopotential	500	z	
Specific humidity	700	q	
Relative humidity	850	r	

The forecasts are available for the same model steps as the surface variables above.

Usage: The pressure level variables forecasts can be retrieved by calling

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-forecasts-
↳pressure', date, parameter, level, kind)
ds.to_xarray()
```

where the `date` argument is a string with a single date, and the `parameter` argument is a string or a list of string with the ECMWF keys described above. Setting 'all' as `parameter` download all the parameters at the given pressure level. The `level` argument is the pressure level, as a string or an integer. The `kind` argument allows to select the deterministic or ensemble forecasts, by setting it to 'highres' or 'ensemble'.

Example:

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-forecasts-
↳pressure', "2017-12-02", "z", 500, "highres")
ds.to_xarray()
```

2.2.4 1.4 - Processed surface variable forecasts

Processed surface variables can be obtained for each forecast date, both for the ensemble (51 members) and deterministic runs. A processed variable is either accumulated, averaged or filtered.

It includes:

Parameter name	ECMWF key	Remarks
Total precipitation	tp	
Surface sensible heat flux	sshf	
Surface latent heat flux	slhf	
Surface net solar radiation	ssr	
Surface net thermal radiation	str	
Convective precipitation	cp	
Maximum temperature at 2 metres	mx2t6	
Minimum temperature at 2 metres	mn2t6	
Surface solar radiation downwards	ssrd	
Surface thermal radiation downwards	strd	
10 metre wind gust	10fg6	

All these variables are accumulated or filtered over the last 6 hours preceding a given forecast timestamp. Therefore, the forecasts are available for the model steps (in hours) 6, 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192, 198, 204, 210, 216, 222, 228, 234 and 240. All the steps are automatically retrieved.

Usage: The processed surface variables forecasts can be retrieved by calling

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-forecasts-  
→surface-processed', date, parameter, kind)  
ds.to_xarray()
```

where the `date` argument is a string with a single date, and the `parameter` argument is a string or a list of string with the ECMWF keys described above. The `kind` argument allows to select the deterministic or ensemble forecasts, by setting it to 'highres' or 'ensemble'.

Note: For technical reason, most fields cannot be retrieved along the others and must be downloaded alone. E.g. a request with `parameter=['tp', 'mx2t6']` will fail while one with `parameter='tp'` will succeed.

Example:

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-forecasts-  
→surface-processed', "2017-12-02", "mx2t6", "highres")  
ds.to_xarray()
```

2.2.5 1.5 - Surface variable reforecasts

The surface variables for the ensemble reforecasts (11 members) can be obtained for each reforecast date. All the variables described at the point 1.2 above are available.

The reforecasts are available for the model steps (in hours) 0, 6, 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192, 198, 204, 210, 216, 222, 228, 234 and 240. All the steps are automatically retrieved.

Note: The ECMWF reforecasts are only available Mondays and Thursdays. Providing any other date will fail.

Usage: The surface variables reforecasts can be retrieved by calling

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-  
→reforecasts-surface', date, parameter)  
ds.to_xarray()
```

where the `date` argument is a string with a single date, and the `parameter` argument is a string or a list of string with the ECMWF keys. Setting 'all' as `parameter` download all the surface parameters.

Example:

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-  
→reforecasts-surface', "2017-12-28", "sd")  
ds.to_xarray()
```

2.2.6 1.6 - Pressure level variable reforecasts

The variables on pressure level for the ensemble reforecasts (11 members) can be obtained for each reforecast date. All the variables described at the point 1.3 above are available.

The reforecast are available for the same model steps as the surface variables above.

Note: The ECMWF reforecasts are only available Mondays and Thursdays. Providing any other date will fail.

Usage: The pressure level variables reforecasts can be retrieved by calling

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-  
↪reforecasts-pressure', date, parameter, level)  
ds.to_xarray()
```

where the `date` argument is a string with a single date, and the `parameter` argument is a string or a list of string with the ECMWF keys. Setting 'all' as `parameter` download all the parameters at the given pressure level. The `level` argument is the pressure level, as a string or an integer.

Example:

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-  
↪reforecasts-pressure', "2017-12-28", "z", 500)  
ds.to_xarray()
```

2.2.7 1.7 - Processed surface variable reforecasts

Processed surface variables as described in section 1.4 can also be obtained as ensemble reforecasts (11 members).

The reforecast are available for the same model steps as the surface variables described in section 1.5.

Note: The ECMWF reforecasts are only available Mondays and Thursdays. Providing any other date will fail.

Usage: The surface variables forecasts can be retrieved by calling

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-  
↪reforecasts-surface-processed', date, parameter)  
ds.to_xarray()
```

where the `date` argument is a string with a single date, and the `parameter` argument is a string or a list of string with the ECMWF keys.

Note: For technical reason, most fields cannot be retrieved along the others and must be downloaded alone. E.g. a request with `parameter=['tp', 'mx2t6']` will fail while one with `parameter='tp'` will succeed.

Example:

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-  
↪reforecasts-surface-processed', "2017-12-28", "mx2t6")  
ds.to_xarray()
```


2.2.8 1.8 - Static fields

Various static fields associated to the forecast grid can be obtained, with the purpose of serving as predictors for the postprocessing.

Note: For consistency with the rest of the dataset, we use the ECMWF parameters name, terminology and units here. However, please note that - except for the Surface Geopotential - the fields provided are from other non-ECMWF data sources evaluated at grid points. Currently, the main data source being used is the [Copernicus Land Monitoring Service](#).

It includes:

Parameter name	ECMWF key	Remarks
Land use	landu	Extracted from the CORINE 2018 dataset. Values and associated land type differ from the ECMWF one. Please look at the “legend” entry in the metadata for more details.
Model terrain height	mterh	Extracted from the EU-DEMv1.1 data elevation model dataset.
Surface Geopotential	z	The model orography can be obtained by dividing the surface geopotential by $g=9.80665 \text{ ms}^{-2}$.

Usage: The static fields can be retrieved by calling

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-static-
↳fields', parameter)
ds.to_xarray()
```

where the `parameter` argument is a string with one of the ECMWF keys described above. It is only possible to download one static field per call.

Example:

```
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-static-
↳fields', 'mterh')
ds.to_xarray()
```

2.3 2 - Stations Data

Not yet provided.

2.4 3 - Getting the observations corresponding to the (re)forecasts

Once obtained, the observations corresponding to the forecasts or reforecasts (if available) can be retrieved in the `xarray` format by using the `get_observations_as_xarray` method:

```
obs = ds.get_observations_as_xarray()
```

2.5 4 - Explanation of the metadata

The following metadata are available in the gridded forecast, reforecast and observation data:

1. **latitude**: The latitude of the grid points.
2. **longitude**: The longitude of the grid points.
3. **depthBelowLandLayer**: the layer below the surface (valid for some variables only, here there is only the upper surface level).
4. **number**: the number of the ensemble member. The 0-th member is the control run. Also present in observation, but set to 0.
5. **time**: the forecast or reforecast date (reforecasts are only issued on Mondays and Thursdays).
6. **year**: a dimension to identify the year in the past, year=1 means a forecast valid 20 years ago at the reforecast day and month, year=20 means a forecast valid one year before the reforecast date. Only valid for reforecasts.
7. **step**: the step of the forecast (the lead time).
8. **surface**: the layer of the variable considered (here there is just one, at the surface).
9. **valid_time**: the actual time and date of the corresponding forecast data.

2.6 5 - Major ECMWF model changes

In 2017 and 2018, there were 2 model changes of the ECMWF model on total:

Implementation date	Summary of changes	Resolution	Full IFS documentation
05-Jun-2018	Cycle 45r1	Unchanged	Cycle 45r1 full documentation
11-Jul-17	Cycle 43r3	Unchanged	Cycle 43r3 full documentation

Source: <https://www.ecmwf.int/en/forecasts/documentation-and-support/changes-ecmwf-model>

2.7 Data License

See the [DATA_LICENSE](#) file.

USING CLIMETLAB TO ACCESS THE DATA

A plugin for `climetlab` to retrieve the Eumetnet postprocessing benchmark datasets is available.

`pypi package` **0.2.4** `build` **passing**

It facilitates the download of the dataset time-aligned forecasts, reforecasts (hindcasts) and observations (ERA5 reanalysis).

See the [demo notebooks](#)

- [demo_training_data_forecasts.ipynb](#) `render` `nbviewer` `CO` `Open in Colab`
- [demo_ensemble_forecasts.ipynb](#) `render` `nbviewer` `CO` `Open in Colab` `Launch in Deepnote`
- [demo_EUPPBench_germany_station_data.ipynb](#) `render` `nbviewer` `CO` `Open in Colab` `Launch in Deepnote`
- [demo_EUPPBench_gridded_data.ipynb](#) `render` `nbviewer` `CO` `Open in Colab` `Launch in Deepnote`

The `climetlab` python plugin allows users to easily access the data with a few lines of code such as:

```
# Uncomment the line below if climetlab and the plugin are not yet installed
#!pip install climetlab climetlab-eumetnet-postprocessing-benchmark
import climetlab as cml
ds = cml.load_dataset('eumetnet-postprocessing-benchmark-training-data-gridded-forecasts-
→ surface', "2017-12-02", "2t", "highres")
fcs = ds.to_xarray()
```

which for instance download the deterministic (high-resolution) forecasts for the 2 metres temperature.

INDICES AND TABLES

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