

DestinE

Climate Digital Twin

Observation Application

(OBSALL Apps)



University of Helsinki team:

Heikki Järvinen

Jouni Räisänen

Lauri Tuppi

Madeleine Ekblom

Alexander Mahura

Online Demonstration of OBSALL Apps

DestinE

Climate Digital Twin

Observation Application

(OBSALL Apps)

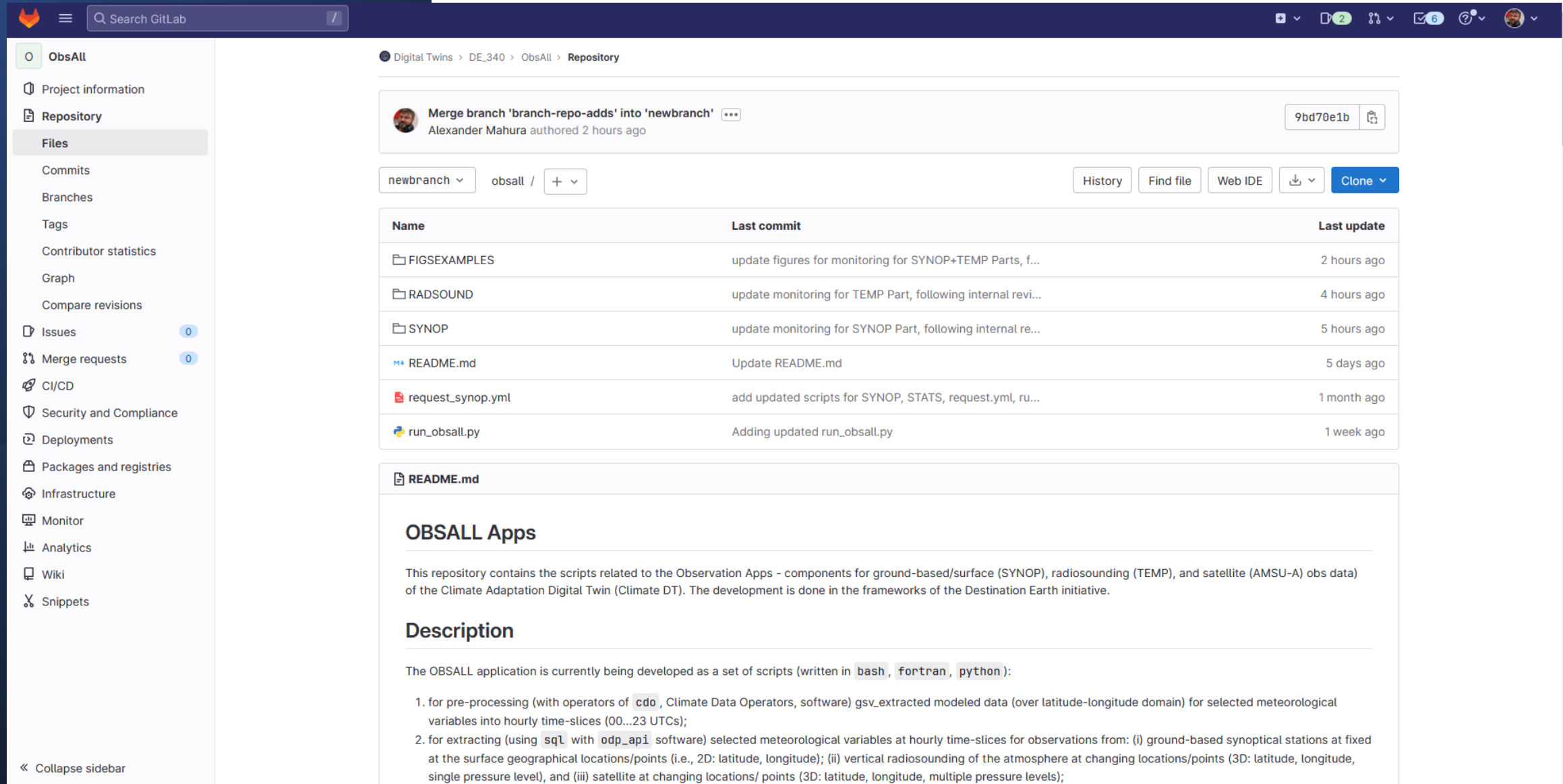
OBSALL Apps includes

3 components for observations:

- Synoptic surface ([SYNOP](#)),
- Upper air sounding ([TEMP](#)),
- Satellite-based ([AMSU-A](#)).

OBSALL Apps: Branch

https://earth.bsc.es/gitlab/digital-twins/de_340/obsall/-/tree/newbranch



ObsAll

Project information

Repository

Files

Commits

Branches

Tags

Contributor statistics

Graph

Compare revisions

Issues 0

Merge requests 0

CI/CD

Security and Compliance

Deployments

Packages and registries

Infrastructure

Monitor

Analytics

Wiki

Snippets

Digital Twins > DE_340 > ObsAll > Repository

Merge branch 'branch-repo-adds' into 'newbranch' Alexander Mahura authored 2 hours ago 9bd70e1b

newbranch obsall +

History Find file Web IDE Clone

Name	Last commit	Last update
FIGSEXAMPLES	update figures for monitoring for SYNOP+TEMP Parts, f...	2 hours ago
RADSOUND	update monitoring for TEMP Part, following internal revi...	4 hours ago
SYNOP	update monitoring for SYNOP Part, following internal re...	5 hours ago
README.md	Update README.md	5 days ago
request_synop.yml	add updated scripts for SYNOP, STATS, request.yml, ru...	1 month ago
run_obsall.py	Adding updated run_obsall.py	1 week ago

README.md

OBSALL Apps

This repository contains the scripts related to the Observation Apps - components for ground-based/surface (SYNOP), radiosounding (TEMP), and satellite (AMSU-A) obs data of the Climate Adaptation Digital Twin (Climate DT). The development is done in the frameworks of the Destination Earth initiative.

Description

The OBSALL application is currently being developed as a set of scripts (written in `bash`, `fortran`, `python`):

- for pre-processing (with operators of `cdo`, Climate Data Operators, software) gsv_extracted modeled data (over latitude-longitude domain) for selected meteorological variables into hourly time-slices (00...23 UTCs);
- for extracting (using `sql` with `odp_api` software) selected meteorological variables at hourly time-slices for observations from: (i) ground-based synoptical stations at fixed at the surface geographical locations/points (i.e., 2D: latitude, longitude); (ii) vertical radiosounding of the atmosphere at changing locations/points (3D: latitude, longitude, single pressure level), and (iii) satellite at changing locations/ points (3D: latitude, longitude, multiple pressure levels);

OBSALL Apps: README.md

https://earth.bsc.es/gitlab/digital-twins/de_340/obsall/-/blob/newbranch/README.md

The screenshot shows the GitLab web interface for the 'OBSALL' repository. The left sidebar contains navigation links: Project information, Repository, Files (selected), Commits, Branches, Tags, Contributor statistics, Graph, Compare revisions, Issues (0), Merge requests (0), CI/CD, Security and Compliance, Deployments, Packages and registries, Infrastructure, Monitor, Analytics, Wiki, and Snippets. The main content area displays the 'README.md' file, which is 17.71 KiB in size. The file content includes a title 'OBSALL Apps', a description of the repository's purpose, a 'Description' section detailing the application's development as a set of scripts, and a 'Part SYNOP (ground-based/surface observations)' section listing specific scripts and their functions.

OBSALL Apps

This repository contains the scripts related to the Observation Apps - components for ground-based/surface (SYNOP), radiosounding (TEMP), and satellite (AMSU-A) obs data) of the Climate Adaptation Digital Twin (Climate DT). The development is done in the frameworks of the Destination Earth initiative.

Description

The OBSALL application is currently being developed as a set of scripts (written in `bash`, `fortran`, `python`):

1. for pre-processing (with operators of `cdo`, Climate Data Operators, software) gsv_extracted modeled data (over latitude-longitude domain) for selected meteorological variables into hourly time-slices (00...23 UTCs);
2. for extracting (using `sql` with `odp_api` software) selected meteorological variables at hourly time-slices for observations from: (i) ground-based synoptical stations at fixed at the surface geographical locations/points (i.e., 2D: latitude, longitude); (ii) vertical radiosounding of the atmosphere at changing locations/points (3D: latitude, longitude, single pressure level), and (iii) satellite at changing locations/ points (3D: latitude, longitude, multiple pressure levels);
3. for extracting and interpolating (using `cdo` operators - for synop; & expecting/using `polytope` - for radiosounding and satellite) modeled data for the same time-slices for same selected meteorological variables into corresponding locations/points of (synop, radiosounding, satellite) observations, and adding (using `import` with `odb_api`) such data to ODB;
4. for calculating/ producing relevant statistics such as quantile rank histogram statistics and plots (t-test and others to be added)

Part SYNOP (ground-based/surface observations)

1. `main_synop.sh` - main bash-script to run Apps for observational data for synoptical stations.
2. `gsv_mod_data.sh` - bash-script to run pre-processing (using `cdo` operators) of modelled data extracted with gsv interface over global domain for selected meteorological variables (measured at synoptical station) into hourly time-slices and saving (temporary) to separate hourly nc-files.
3. `synop_obs.sh` - bash-script to run reading and extraction (using `sql` with `odb_api` software) of selected meteorological variables at hourly time-slices for observations from ground-based synoptical stations at fixed at the surface geographical (latitude, longitude) locations over selected geographical domain (or over the globe) and saving (temporary) to dat-file. Note, that list of available meteorological variables in open multi-year data ODB includes: 91 - total amount of clouds; 108 - sea level pressure; 80 - 1-hour precipitation; 58 - relative humidity; 999 - 10-minute precipitation intensity; 71 - snow depth; 39 - 2m temperature; 40 - dew point temperature; 62 - visibility; 111 - wind direction; 261 - maximum wind gust in last 10 minutes; 221 -surface wind speed.
4. `synop_mod.sh` - bash-script to run reading, extraction and interpolation (using `cdo` operators) of modelled data for selected meteorological variables at the same hourly time-slices to geographical (latitude, longitude) locations of synoptical stations, and saving (temporary) to dat-file, and adding (using `import` with `odb_api`) such data to ODB. Interpolation is preceded by constructing unstructured grid based on locations of synop stations and calculating weights for such grid.
5. `graph_mod_obs.py` - python-script to run an internal self-control in calculating differences between observed and modelled data for locations of synoptical stations
6. `synop_stats.sh` - bash-script to run (additional `bash`, `fortran`, `python` scripts) calculating/producing:
 - (i) rank histograms for all synop stations for 00, 06, 12 and 18 UTCs (with `produce_rank_histograms_all_stations.sh`, using `rank_histograms_one_station.f95` and `sql/import` with `odb_api`),
 - (ii) standard plots for each synop station (with `produce_standard_plots_all_stations.sh` using `plot_quantiles_rankhist.py` and `sql` with `odb_api`; see an example

The OBSALL Apps is developed as a set of scripts (*written in bash, fortran, python*):

1. Pre-processing modeled data

**2. Extracting observations (from ODB) -
from synoptic and radiosounding stations, and satellites**

**3. Extracting modeled data at observation times and locations,
applying appropriate observation operator, and ODB added with model simulation values**

4. Calculating/ producing relevant statistics for monitoring

Note, the end result is the augmented ODB, which can be used:

- **on-line monitoring of the simulation in near-real time**
- **posterior analysis of the ClimateDT model quality**

The OBSALL Apps is developed as a set of scripts (*written in bash, fortran, python*):

1. **Pre-processing** (with operators of cdo, Climate Data Operators, software) gsv_extracted **modeled data** (over latitude-longitude domain) for selected meteorological variables into hourly time-slices (00...23 UTCs);
2. **Extracting** (using sql with odp_api software) selected meteorological variables at hourly time-slices for **observations (from ODB) from**: (i) ground-based **synoptic** stations at fixed geographical latitude-longitude locations on surface; (ii) **radiosounding stations and satellites** at fixed latitude-longitude locations on surface and on multiple pressure levels;
3. **Extracting and applying appropriate observation operator** (using cdo operators - for synop; & expecting/using polytope - for radiosounding and satellite) **to modeled data** for the same time-slices for same selected meteorological variables into corresponding locations/points of (synop, radiosounding, satellite) observations, **with end result - augmented ODB**;
4. **Calculating/ producing relevant statistics for monitoring** such as quantile rank histogram statistics and plots.

Installation

In order to install, copy "obsall" repository to your local directory on Lumi HPC using:

- ✓ login to your DestinE account on Lumi HPC
- ✓ mkdir your_local_directory
- ✓ cd your_local_directory

Option 1:

- ✓ git clone https://earth.bsc.es/gitlab/digital-twins/de_340/obsall.git
- ✓ cd obsall
- ✓ git checkout newbranch

OR

Option 2:

- ✓ cp /full_path_to_location_of_tar_file/arc_obsall_apps.tar .
- ✓ tar -xvf arc_obsall_apps.tar
- ✓ cd obsall

Required Modules

to set environment in order to run OBSALL Apps on Lumi:

- ✓ # Load modules for OBSALL Apps
- ✓ module use /project/project_465000454/devvaraju/modules/LUMI/23.03/C
- ✓ module load LUMI/23.03
- ✓ module load partition/C
- ✓ module load PrgEnv-gnu
- ✓ module load ecCodes/2.32.0-cpeCray-23.03.lua
- ✓ module load odb_api/0.18.1-cpeCray-23.03.lua
- ✓ module load python-climatedt/3.11.3-cpeCray-23.03.lua

Plus,

later loading of modules for: Polytope & Radiance Simulator

Data Requests (yaml-file)

Data request (sfc):

file **request.yaml** is needed for gsv extracting selected modeled data
*on example for 2t, when modeled data **only at sfc-level** is needed*
(in SYNOP Part)

Data request (sfc+pl):

file **request.yaml** is needed for gsv extracting selected modeled data
*on example for 2t and t850, when modeled data **at both sfc- and pl-levels** are needed*
(in TEMP and AMSU-A Parts) & it is in development by BSC
(!) note all other required modeled data are to be added in such script

How to Run Apps

How to run: runsript **run_obsall.py** is used to execute OBSALL Apps
python run_obsall.py

```
#!/scratch/project_465000454/devaraju/SW/LUMI-23.03/C/python-climatedt/bin/python
# OBSALL Apps (3 parts: SYNOP, TEMP, AMSU-A observations)
# Import required libraries
import sysimport subprocess
```

IN IMPLEMENTATION

```
# --- Processing ground-based observations (SYNOP)
command_synop_run = "cd SYNOP; pwd; ./main_synop.sh; exit 0"
subprocess.run(command_synop_run, shell=True, check=True, executable="/bin/bash")
```

IN IMPLEMENTATION

```
# --- Processing radiosounding observations (TEMP)
command_radsound_run = "cd RAD SOUND; pwd; ./main_radsound.sh; exit 0"
subprocess.run(command_radsound_run, shell=True, check=True, executable="/bin/bash")
```

IN DEVELOPMENT

```
# --- Processing satellite observations (AMSU-A)
#command_satellite_run = "cd SATELLITE; pwd; ./main_amsua.sh; exit 0"
#subprocess.run(command_satellite_run, shell=True, check=True, executable="/bin/bash")
```

```
sys.exit(0)
```

Online Demonstration

*on Lumi's DestinE project account
for a limited size dataset of synop and radiosounding observations
and based on modelled data from a fake-FDB*

LUMI

About LUMI

Get Started

News and Articles

Events and Training

Newsletters

User Support



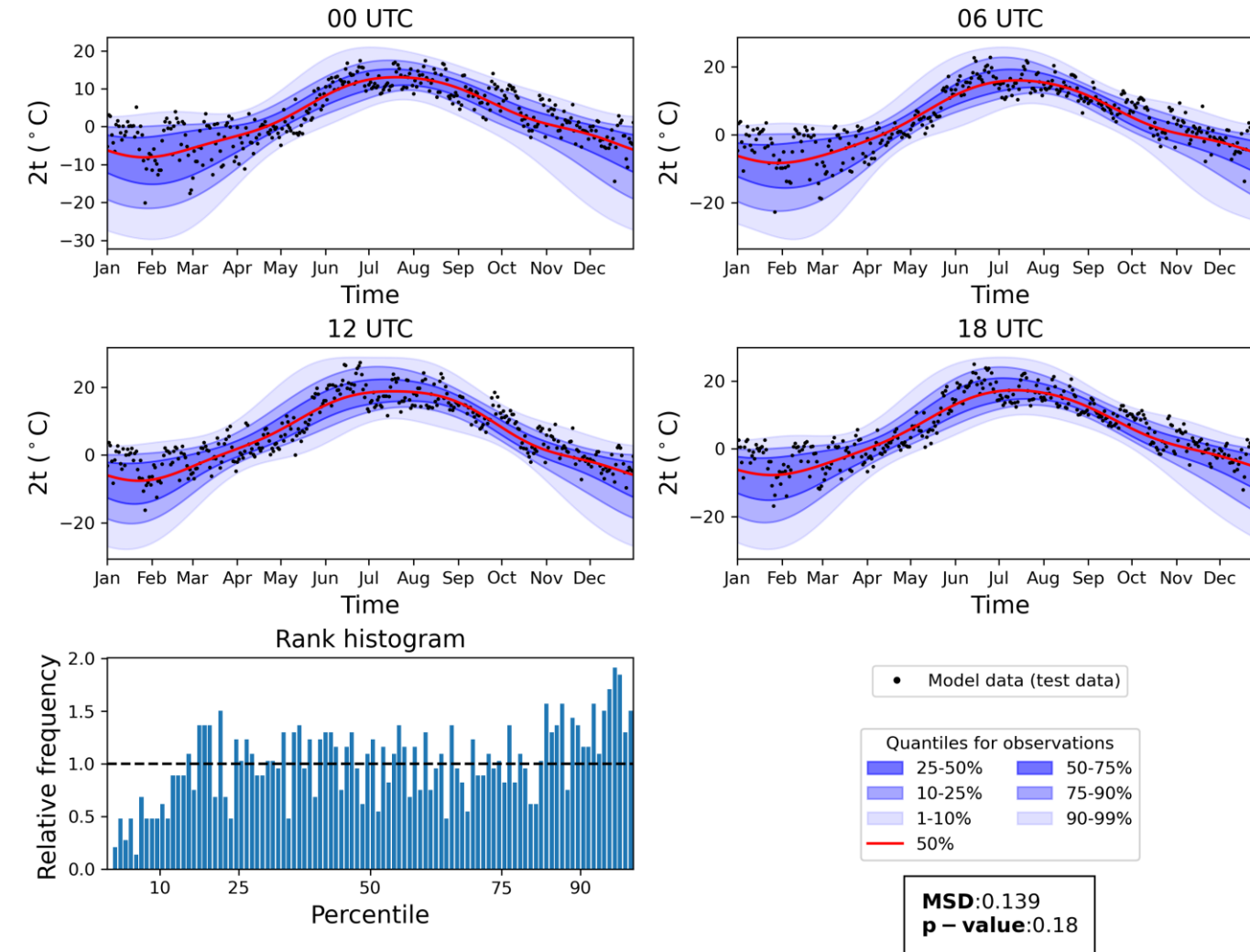
source - <https://www.lumi-supercomputer.eu>



Monitoring for SYNOP (1)

SYNOP station: 126736 (lat: 64.217° N, lon: 27.752° E)

Model data from 202001-202012



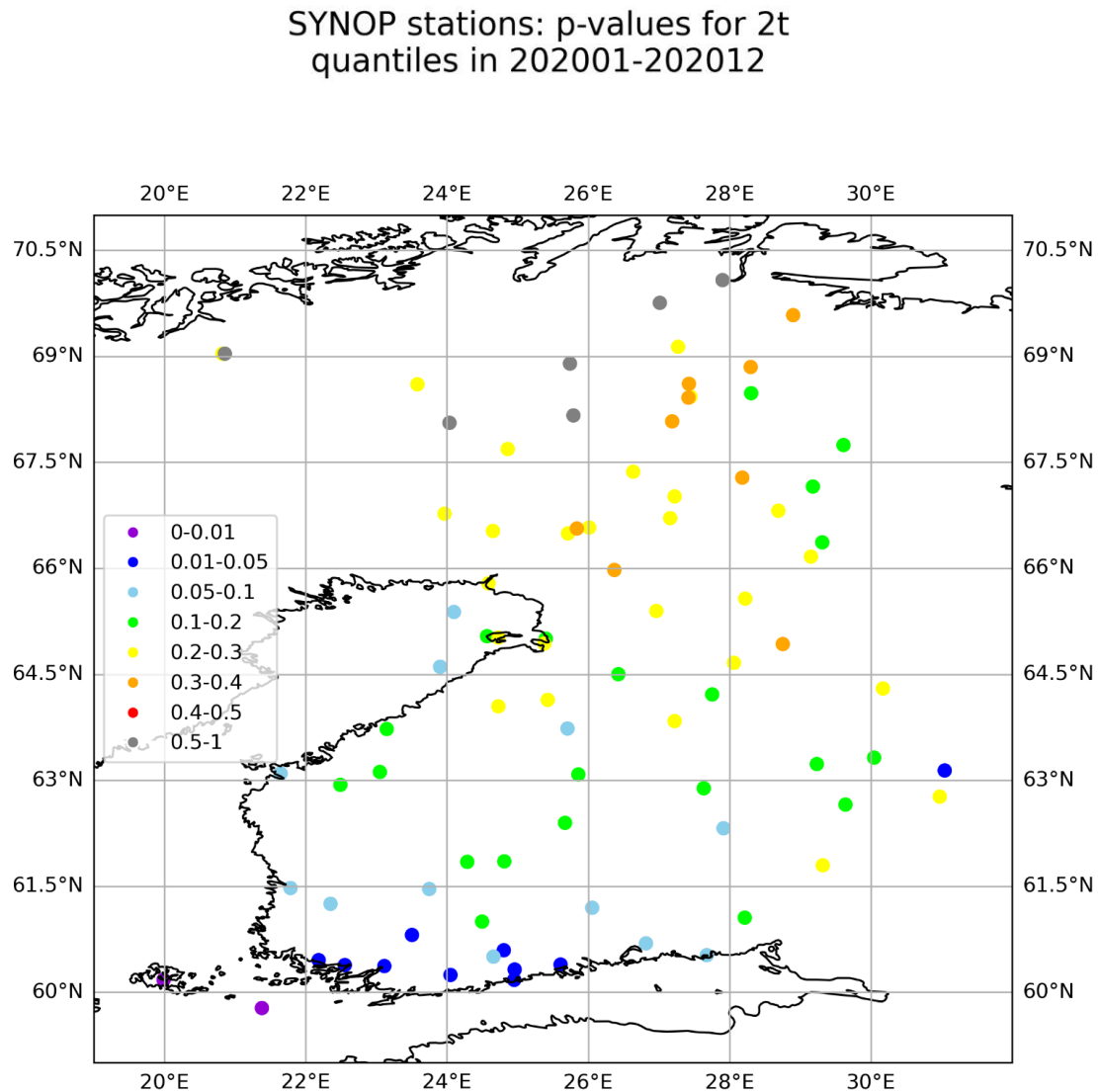
Mean Square Deviation (**MSD** relative to flat histogram)
p-value of MSD relative to sampling distribution
Quantiles are estimated based on observations at the station 126736 (period: 2010-2021)
Model data from experiment (test data) interpolated to location of this station

Graphical/ figures output files:
SYNOP/STATS/figures/

**Plots for 2 metre air temperature (2t)
for period of 1 Jan - 31 Dec 2020:**

Figure 1a - Quantiles/ time series data (00, 06, 12, and 18 UTCs) and rank histogram (combination of 00, 06, 12, and 18 UTCs) for synop station 126736.

Monitoring for SYNOP (2)

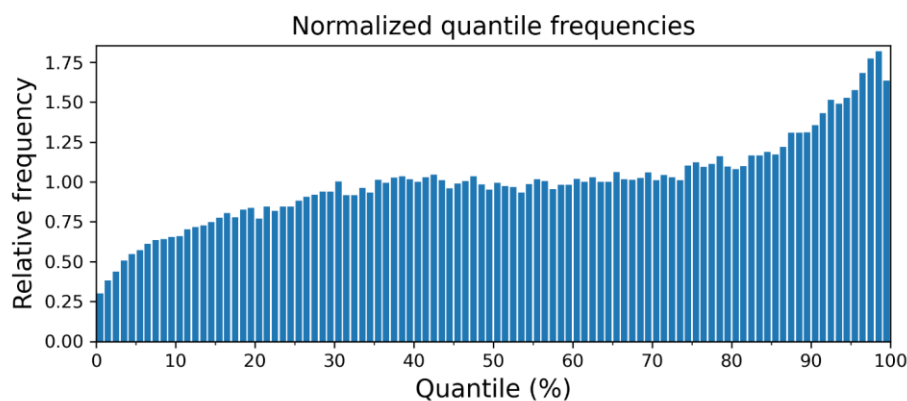
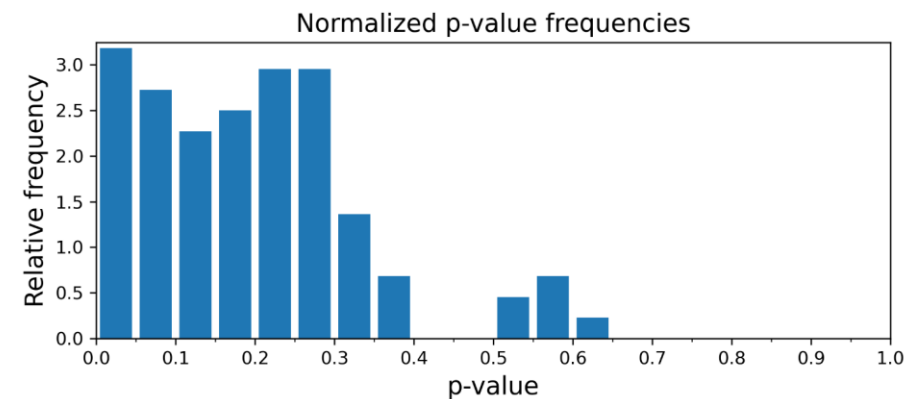


p-value: the probability that the MSD statistics for the quantile rank histogram would be exceeded by pure chance when comparing a perfect model simulation with observations.

**Plots for 2 metre air temperature (2t)
for period of 1 Jan - 31 Dec 2020:**

Figure 1b - p-values for 2t quantiles.

Monitoring for SYNOP (3)



Number of stations: 88
Frequency, $p < 0.001$: 0.023
Frequency, $p < 0.01$: 0.045
Frequency, $p < 0.05$: 0.159

Normalized p-values: Frequency distribution of p-values at all individual stations, normalized to give an expected value of 1.
Normalized quantiles: Frequency distribution of model data relative to quantiles of observations, averaged over all stations and divided by the expected value of 0.01.

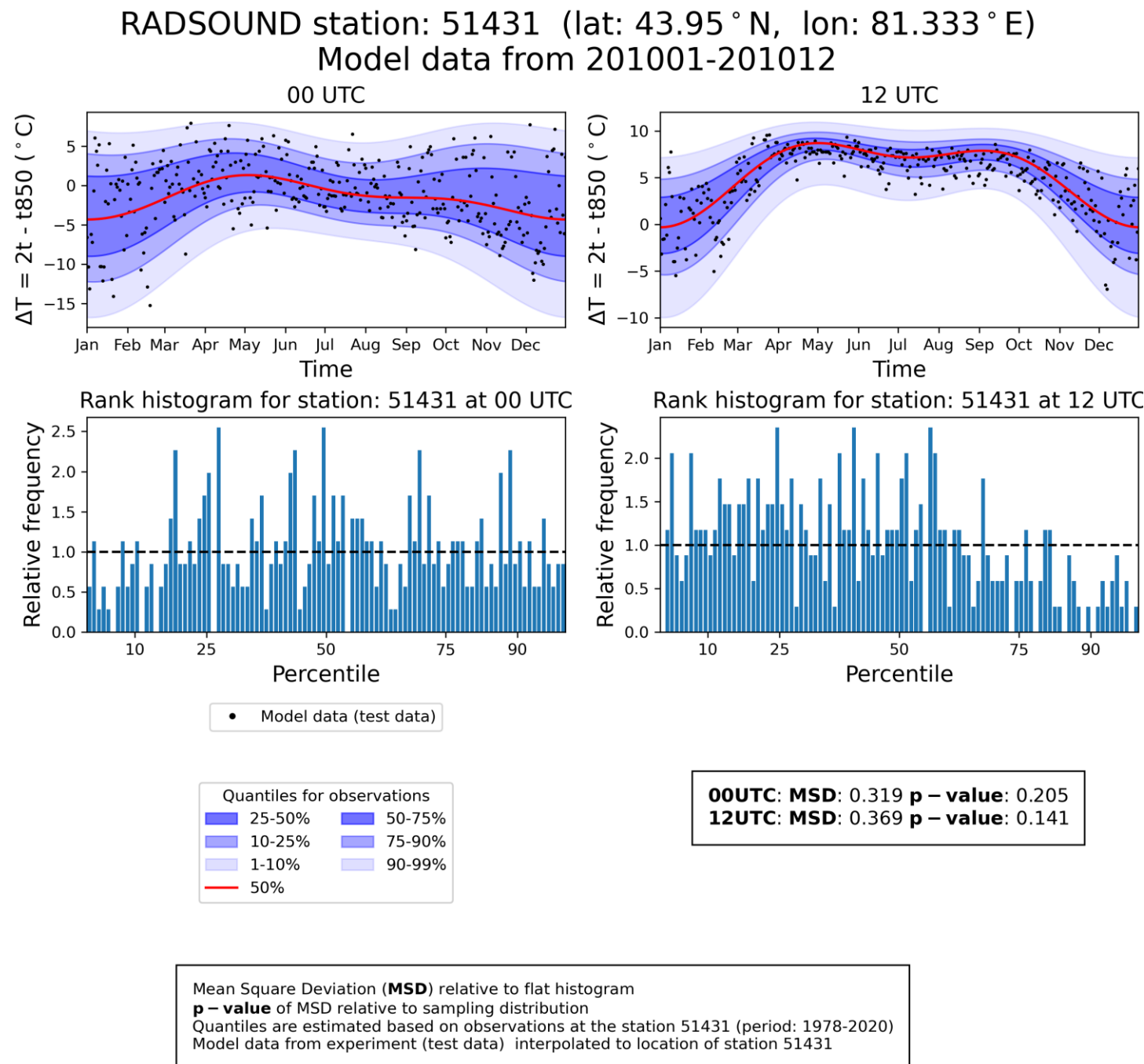
Plots for 2 metre air temperature (2t)
for period of 1 Jan - 31 Dec 2020:

Figure 1c - Rank histogram summary statistics:
normalized p-value frequencies and normalized
quantile frequencies.

Monitoring for TEMP (1)

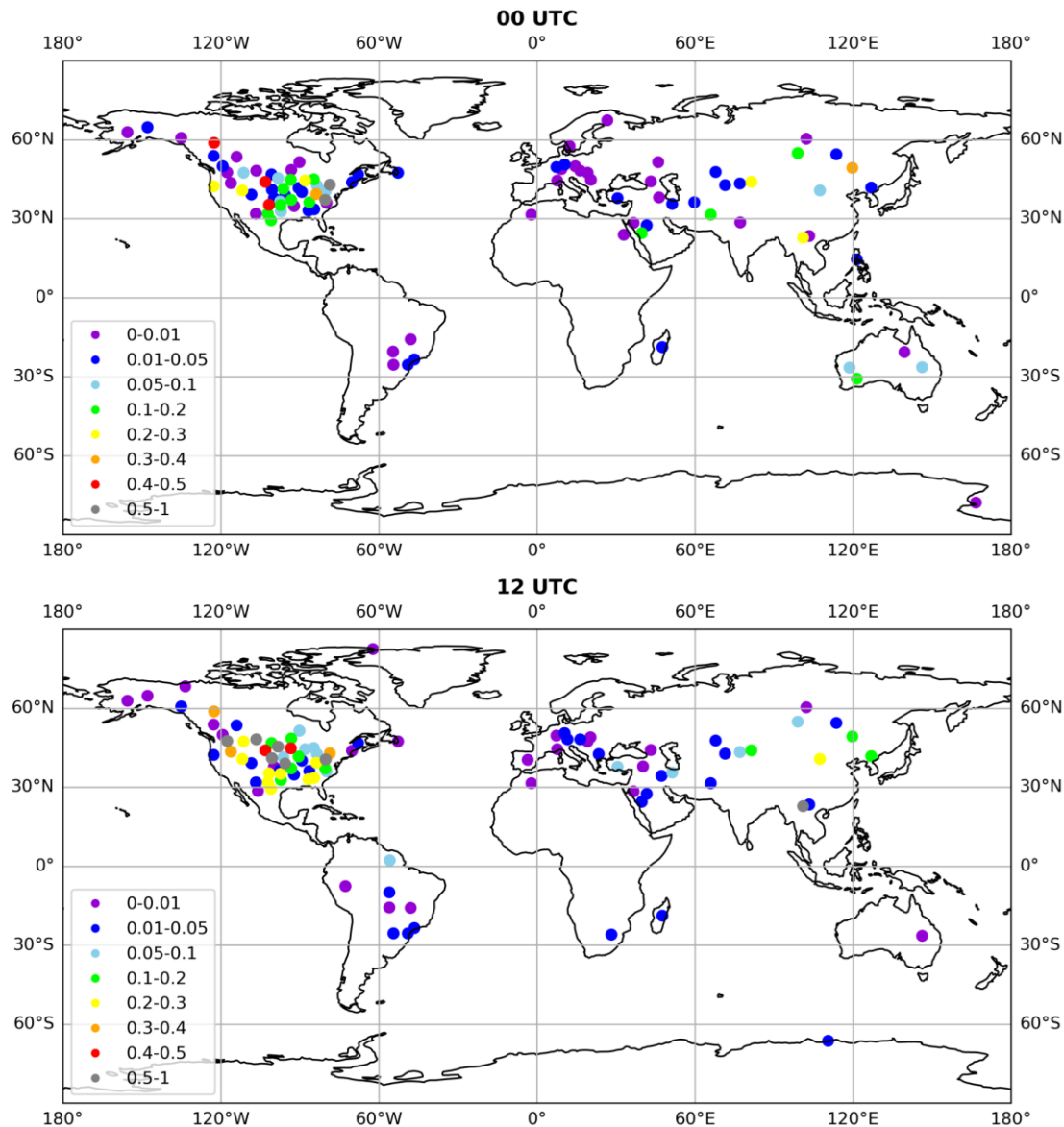
Graphical/ figures output files:
RADSOUND/STATRS/figures/

**Plots for difference in air
temperatures at 2 metre and at
850 hPa pressure level (2t - t850)
for period of 1 Jan - 31 Dec 2010:**



**Figure 2a - Quantiles/ time series data and
rank histogram (00 and 12 UTCs)
for radiosounding station 51431**

RADSOUND stations: p-values for $\Delta T = 2t - t_{850}$
quantiles in 201001-201012



p-value: the probability that the MSD statistics for the quantile rank histogram would be exceeded by pure chance when comparing a perfect model simulation with observations.

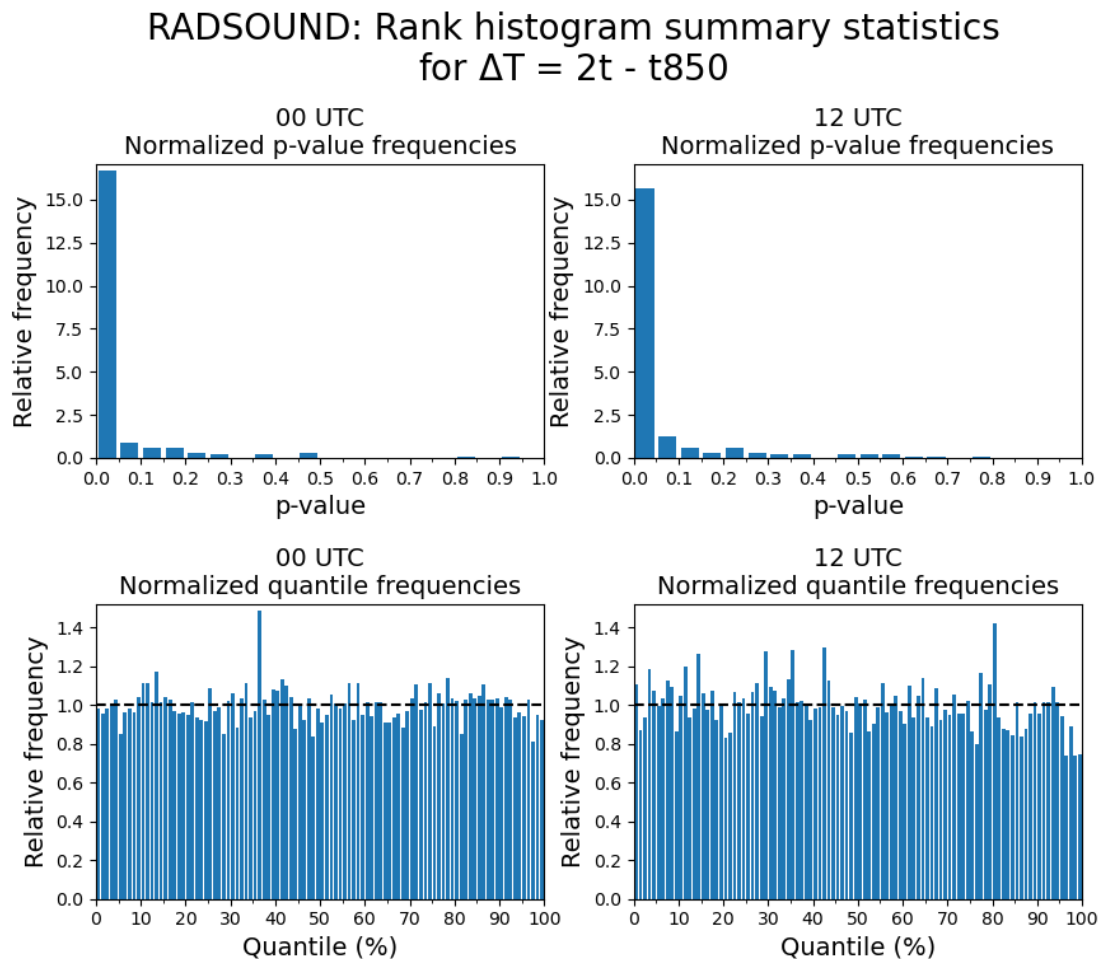
Monitoring for TEMP (2)

Plots for difference in air
temperatures at 2 metre and at
850 hPa pressure level ($2t - t_{850}$)
for period of 1 Jan - 31 Dec 2010:

Figure 2b - p-values for $2t - t_{850}$
quantiles.

Monitoring for TEMP (3)

**Plots for difference in air
temperatures at 2 metre and at
850 hPa pressure level (2t - t850)
for period of 1 Jan - 31 Dec 2010:**



Number of stations: 212
Frequency, $p < 0.001$, 00 UTC: 0.562 Frequency, $p < 0.001$, 12 UTC: 0.556
Frequency, $p < 0.01$, 00 UTC: 0.687 Frequency, $p < 0.01$, 12 UTC: 0.638
Frequency, $p < 0.05$, 00 UTC: 0.836 Frequency, $p < 0.05$, 12 UTC: 0.786

Normalized p-values: Frequency distribution of p-values at all individual stations, normalized to give an expected value of 1.
Normalized quantiles: Frequency distribution of model data relative to quantiles of observations, averaged over all stations and divided by the expected value of 0.01.

Figure 2c - Rank histogram summary statistics for 00 and 12 UTCs: normalized p-value frequencies and normalized quantile frequencies.