The existing radar-based nowcasting software packages are proprietary and being developed in national weather services. The aim of the pysteps initiative is to bring the nowcasting knowledge available to the public in a Python-based open-source software package.

The underlying model is Lagrangian persistence. The precipitation field is extrapolated using the estimated advection velocities (Fig. 1). The model takes into account scale-dependence of predictability. Precipitation fields are decomposed into multiple spatial scales using a set of Fourier bandpass filters (Figs. 2 and 3).

Temporal evolution of precipitation intensities is modeled by using an autoregressive (AR(2)) model in Lagrangian coordinates (Fig. 4).

The precipitation field is extrapolated by using the Lucas-Kanade method with interpolation applied to areas of no precipitation (Fig. 5).

Practically, for the precipitation intensities, a fitted Gaussian noise field is added separately to each cascade level to account for the scale-dependence of predictability (Fig. 5).

The advection field estimated by using the Lucas-Kanade method with interpolation applied to areas of no precipitation.

Figure 3. Level cascade decomposition of a radar composite.


The recommented method for installing pysteps is to use the Anaconda distribution.

Verification


- Several optical flow methods for motion field estimation (Lucas-Kanade, DARTS and Variational Echo Tracking).
- Localization: the statistics of precipitation fields are localized in space (Herny et al., 2017). This gives improved nowcast accuracy in larger domains with different types of precipitation (e.g. stratiform and convective).
- Computational improvements: the cascade decomposition, AR(2) models and noise generation can be applied in the spectral domain to improve performance and reduce memory usage (Pulkkinen et al., 2019).

Enhancements to earlier STEPS implementations

The following code example demonstrates the use of pysteps:

```python
from pysteps import cascade
from pysteps import verification
from pysteps.visualization import *
import numpy as np

# Create a 2D random field
N = 128
x = np.arange(N)
y = np.arange(N)
x, y = np.meshgrid(x, y)
field = cascade.cascade(x, y, 4, 2, 2)

data = cascade.steplike(field, 1, 1)

data = cascade.steplike(field, 2, 1)

data = cascade.steplike(field, 3, 1)

# Verify the nowcast
verification.verify(data, truth, [0.5], 1)
```

```bash
# Install pysteps
pip install pysteps
```

The following list shows the main features of pysteps:

- Multi-scale decomposition of precipitation fields.
- Interpolation and extrapolation of precipitation fields.
- Stochastic perturbations and advection fields to simulate uncertainties.
- For the precipitation intensities, a fitted Gaussian noise field is added separately to each cascade level to account for the scale-dependence of predictability.
- The advection field estimated by using the Lucas-Kanade method with interpolation applied to areas of no precipitation.

Computational requirements

- Stochastic perturbations are added to the precipitation intensity and advection fields to simulate uncertainties.
- The model takes into account scale-dependence of predictability.
- Precipitation fields are decomposed into multiple spatial scales using a set of Fourier bandpass filters (Figs. 2 and 3).

Verification

- Several optical flow methods for motion field estimation (Lucas-Kanade, DARTS and Variational Echo Tracking).
- Localization: the statistics of precipitation fields are localized in space (Herny et al., 2017). This gives improved nowcast accuracy in larger domains with different types of precipitation (e.g. stratiform and convective).
- Computational improvements: the cascade decomposition, AR(2) models and noise generation can be applied in the spectral domain to improve performance and reduce memory usage (Pulkkinen et al., 2019).

Enhancements to earlier STEPS implementations

The following code example demonstrates the use of pysteps:

```python
from pysteps import cascade
from pysteps import verification
from pysteps.visualization import *
import numpy as np

# Create a 2D random field
N = 128
x = np.arange(N)
y = np.arange(N)
x, y = np.meshgrid(x, y)
field = cascade.cascade(x, y, 4, 2, 2)

data = cascade.steplike(field, 1, 1)

data = cascade.steplike(field, 2, 1)

data = cascade.steplike(field, 3, 1)

# Verify the nowcast
verification.verify(data, truth, [0.5], 1)
```

```bash
# Install pysteps
pip install pysteps
```

The following list shows the main features of pysteps:

- Multi-scale decomposition of precipitation fields.
- Interpolation and extrapolation of precipitation fields.
- Stochastic perturbations and advection fields to simulate uncertainties.
- For the precipitation intensities, a fitted Gaussian noise field is added separately to each cascade level to account for the scale-dependence of predictability.
- The advection field estimated by using the Lucas-Kanade method with interpolation applied to areas of no precipitation.

Verification

- Several optical flow methods for motion field estimation (Lucas-Kanade, DARTS and Variational Echo Tracking).
- Localization: the statistics of precipitation fields are localized in space (Herny et al., 2017). This gives improved nowcast accuracy in larger domains with different types of precipitation (e.g. stratiform and convective).
- Computational improvements: the cascade decomposition, AR(2) models and noise generation can be applied in the spectral domain to improve performance and reduce memory usage (Pulkkinen et al., 2019).

Enhancements to earlier STEPS implementations

The following code example demonstrates the use of pysteps:

```python
from pysteps import cascade
from pysteps import verification
from pysteps.visualization import *
import numpy as np

# Create a 2D random field
N = 128
x = np.arange(N)
y = np.arange(N)
x, y = np.meshgrid(x, y)
field = cascade.cascade(x, y, 4, 2, 2)

data = cascade.steplike(field, 1, 1)

data = cascade.steplike(field, 2, 1)

data = cascade.steplike(field, 3, 1)

# Verify the nowcast
verification.verify(data, truth, [0.5], 1)
```

```bash
# Install pysteps
pip install pysteps
```

The following list shows the main features of pysteps:

- Multi-scale decomposition of precipitation fields.
- Interpolation and extrapolation of precipitation fields.
- Stochastic perturbations and advection fields to simulate uncertainties.
- For the precipitation intensities, a fitted Gaussian noise field is added separately to each cascade level to account for the scale-dependence of predictability.
- The advection field estimated by using the Lucas-Kanade method with interpolation applied to areas of no precipitation.

Verification

- Several optical flow methods for motion field estimation (Lucas-Kanade, DARTS and Variational Echo Tracking).
- Localization: the statistics of precipitation fields are localized in space (Herny et al., 2017). This gives improved nowcast accuracy in larger domains with different types of precipitation (e.g. stratiform and convective).
- Computational improvements: the cascade decomposition, AR(2) models and noise generation can be applied in the spectral domain to improve performance and reduce memory usage (Pulkkinen et al., 2019).

Enhancements to earlier STEPS implementations

The following code example demonstrates the use of pysteps:

```python
from pysteps import cascade
from pysteps import verification
from pysteps.visualization import *
import numpy as np

# Create a 2D random field
N = 128
x = np.arange(N)
y = np.arange(N)
x, y = np.meshgrid(x, y)
field = cascade.cascade(x, y, 4, 2, 2)

data = cascade.steplike(field, 1, 1)

data = cascade.steplike(field, 2, 1)

data = cascade.steplike(field, 3, 1)

# Verify the nowcast
verification.verify(data, truth, [0.5], 1)
```

```bash
# Install pysteps
pip install pysteps
```

The following list shows the main features of pysteps:

- Multi-scale decomposition of precipitation fields.
- Interpolation and extrapolation of precipitation fields.
- Stochastic perturbations and advection fields to simulate uncertainties.
- For the precipitation intensities, a fitted Gaussian noise field is added separately to each cascade level to account for the scale-dependence of predictability.
- The advection field estimated by using the Lucas-Kanade method with interpolation applied to areas of no precipitation.

Verification

- Several optical flow methods for motion field estimation (Lucas-Kanade, DARTS and Variational Echo Tracking).
- Localization: the statistics of precipitation fields are localized in space (Herny et al., 2017). This gives improved nowcast accuracy in larger domains with different types of precipitation (e.g. stratiform and convective).
- Computational improvements: the cascade decomposition, AR(2) models and noise generation can be applied in the spectral domain to improve performance and reduce memory usage (Pulkkinen et al., 2019).