SciPy Solution

Contents

Choosing a probability distribution to fit daily precipitation depths is important for precipitation frequency analysis, stochastic precipitation modeling, and climate assessments. Read the file ../../assets/data/Changi_daily_rainfall.csv and complete the following tasks.

4.3.1. Task 1

Extract the wet-day series for 2020 from the raw daily rainfall dataset and calculate a few descriptive statistics: mean, variance, skewness, kurtosis, L-CV, and L-skewness. Is the distribution left-skewed or right-skewed? Is the distribution more or less peaked than the normal distribution?

Note: The wet-day series should be constructed by excluding events whose magnitude is less than 0.25 mm/day (0.25 mm/day is the minimum precipitation that can be recorded by the in situ rain gauge).

The L-CV is L-coefficient of variation and can be calculated as:

$$L\text{-}CV = \lambda_2/\lambda_1$$

[Skip to main content](#)

and the L-skewness is L-coefficient of skewness, which is calculated as:

$$\text{L-skewness} = \tau_3 = \lambda_3 / \lambda_2,$$

where λ_1, λ_2 and λ_3 are the first three L-moments (details can be found in `SciPy tutorial`)

```python
# Your solution goes here.
import pandas as pd
import numpy as np
from scipy import stats

df = pd.read_csv('../../assets/data/Changi_daily_rainfall.csv', index_col=0, parse_dates=True)
df_sb = df.loc['2020']
wet_series = df_sb[df_sb>0.25].dropna().values.flatten()


def samlmom3(sample):
    """
    samlmom3 returns the first three L-moments of samples
    sample is the 1-d array
    n is the total number of the samples, j is the j_th sample
    """
    n = len(sample)
    sample = np.sort(sample.reshape(n))[::-1]
    b0 = np.mean(sample)
    b1 = np.array([(n - j - 1) * sample[j] / n / (n - 1)
                   for j in range(n)]).sum()
    b2 = np.array([(n - j - 1) * (n - j - 2) * sample[j] / n / (n - 1) / (n - 2)
                   for j in range(n - 1)]).sum()
    lmom1 = b0
    lmom2 = 2 * b1 - b0
    lmom3 = 6 * (b2 - b1) + b0

    return lmom1, lmom2, lmom3


lmom1, lmom2, lmom3 = samlmom3(wet_series)
```

Skip to main content

```python
L_skew = lmon3/lmon2

print('mean: %.2f' % wet_series.mean())
print('variance %.2f' % wet_series.var())
print('Skewness %.2f' % stats.skew(wet_series, axis=0))
print('Kurtosis: %.2f' % stats.kurtosis(wet_series, axis=0, bias=False))
print('L-CV: %.2f' % L_CV)
print('L-skewness %.2f' % L_skew)
```

```
mean: 11.42
variance 165.47
Skewness 1.72
Kurtosis: 3.04
L-CV: 0.56
L-skewness 0.39
```

```python
# the distribution is right-skewed and more peaked than the normal distribution.
```

4.3.2. Task 2

Early studies identified the gamma (G2) distribution as a suitable distribution for wet-day precipitation based on the traditional goodness-of-fit tests. Does wet-day series of Changi follow a gamma distribution? Check the `scipy.stats` documentation and do the following:

- fit a gamma distribution to the Changi wet-day series;

- print out the estimated parameters;

- print out the goodness-of-fit test results using KS test.

Skip to main content

```python
a, loc, scale = gamma.fit(wet_series)

print(a, loc, scale)
gamma_dist = gamma(a, loc=loc, scale=scale)
print(stats.kstest(wet_series, gamma_dist.cdf))
```

```
0.6597146798552613 0.39999999999999997 15.511976229013442
KstestResult(statistic=0.05634367405742169, pvalue=0.6503157533710182)
```

```python
# The p-value of ks-stastistic is larger than 0.05.
# The wet-day series for 2020 of Changi follows a gamma distribution.
```