ins_py

A Python package for computing the **Index of Non-Stationarity (INS)** of discrete-time signals using the surrogate-based time-frequency method proposed by Borgnat et al. (2010).

Overview

`ins_py` implements the multi-scale stationarity test from [1]. The core idea is to compare the **local** (time-varying) spectrum of a signal against its **global** (time-averaged) spectrum in the time-frequency plane. To decide whether the observed spectral variation is statistically significant, the method builds a reference distribution under the null hypothesis of stationarity by generating **phase-randomized surrogates** — signals that share the same power spectral density as the original but whose temporal structure has been destroyed.

The test produces:

- **INS(scale)** — an index that equals approximately 1 for stationary signals and grows above 1 as non-stationarity increases.
- **Threshold gamma(scale)** — a decision boundary derived from a Gamma-fitted surrogate distribution at a 5% false-alarm rate. The signal is declared non-stationary at a given scale when INS > gamma.

Installation

Clone or download this repository and install in development mode:

```
cd ins_py
pip install -e .
```

Dependencies

- Python >= 3.8
- NumPy
- SciPy

Quick Start

```
import scipy.io.wavfile as wav
import matplotlib.pyplot as plt
from ins_py import ins

# Load a WAV file
samplerate, data = wav.read("bin/speech.wav")

# Compute the INS across the default set of observation scales
INS, threshold, scales = ins(data, numberOfSurrogates=50)
```

```python
# Plot the results
plt.plot(scales, INS, "s-k", label="INS")
plt.plot(scales, threshold, "d--r", label=r"$\gamma$")
plt.xlabel(r"$T_h / T$")
plt.ylabel("INS")
plt.title("INS stationarity test")
plt.legend()
plt.grid(axis="y", linestyle="--")
plt.show()
```

API Reference

`ins(x, scales, numberOfSurrogates, calcThreshold)`

Main entry point. Computes the INS profile over a list of observation scales.

Parameter	Type	Default	Description
`x`	`ndarray`	*(required)*	Real-valued input signal.
`scales`	`list[float]`	`[0.015, 0.03, ..., 0.5]`	Relative observation scales (Th/T).
`numberOfSurrogates`	`int`	`50`	Number of surrogates per scale.
`calcThreshold`	`bool`	`True`	Whether to compute the Gamma-fitted decision threshold.

Returns: `(ins_values, thresholds, scales)` — all as NumPy arrays / list.

Project Structure

```
ins_py/
  bin/
     speech.wav          # Example speech signal
     noisy.wav           # Example noisy signal
  ins_py/
     __init__.py
     ins.py              # Core implementation
  tests/
     00_speech.py        # Example script
```

```
setup.py
README.md
```

References

[1] P. Borgnat, P. Flandrin, P. Honeine, C. Richard and J. Xiao, "Testing Stationarity With Surrogates: A Time-Frequency Approach," *IEEE Transactions on Signal Processing*, vol. 58, no. 7, pp. 3459-3470, July 2010. doi: 10.1109/TSP.2010.2043971