HAF Friction Model — Reproducibility Package

This package fits the three-parameter **HAF** (Hysteresis–Attrition–Friction) model to gear tooth-friction traction curves, quantifies the uncertainty of every parameter by the **BIG bootstrap**, and reproduces all tables and figures of the accompanying paper.

Two equivalent implementations are provided:

File	Language	Requirements
`HAF_model.m`	MATLAB (R2016b+) / GNU Octave	base install, **no toolbox**
`haf_model.py`	Python 3	`numpy scipy matplotlib pandas openpyxl pillow`

Both read the same input file and write the same outputs (same file names and numbering).

Model

Each experimental traction curve is described by

```
F(SRR) = 2·ν · ( 1 / (1 + exp(−h·SRR)) + a·SRR − 0.5 )
```

- **h** — hysteresis (steepness of the sigmoidal transition around SRR = 0)
- **a** — attrition (linear trend on the plateau)
- **ν** — friction level (overall scaling)

fitted by nonlinear least squares. The dependence of h, a and ν on the operating conditions is then described by linear laws (per surface state, then a generalized model including roughness), and finally by a single complete non-linear model obtained by substituting the linear laws back into the HAF form. Every parameter, regression coefficient and R^2 inherits a full bootstrap distribution — hence a standard deviation and a significance test.

Input

A single semicolon-separated file **`SRR_frot_data.csv`**, one row per measurement, with a header line and the columns:

```
rugo;pression;temperature;vitesse;SRR;F
```

Column	Meaning	Unit

Column	Meaning	Unit
`rugo`	surface state (rough / smooth; `1/0` or `r/s`)	—
`pression`	contact pressure	GPa
`temperature`	injection temperature	°C
`vitesse`	mean entrainment speed	m/s
`SRR`	slide-to-roll ratio	%
`F`	measured friction coefficient	—

The dataset provided comprises the fourteen operating conditions of the star design (reference 1.6 GPa / 80 °C / 20 m/s; pressures 1.2/1.6/1.9 GPa, temperatures 40/80/100 °C, speeds 10/20/30 m/s; SRR from −20 % to +20 %; smooth and rough discs).

Usage

Place the program and `SRR_frot_data.csv` in the same folder.

MATLAB / Octave

```
run HAF_model.m            % or press Run
```

Python (install once: `pip install numpy scipy matplotlib pandas openpyxl pillow`)

```
python haf_model.py
```

Output

```
HAF_tables/
    Table1.csv  Table2.csv  Table3.csv  Table4.csv
    HAF_tables.xlsx            (formatted workbook; MATLAB R2019a+ and
Python)
HAF_figures/
    PNG/  Fig01…Fig21 + FigA1_…  (body + appendix, one file per figure)
    BMP/  same set in BMP
```

- **Body figures:** Fig 1–3 and Fig 6–21.
- **Appendix figures (complete sets):** A1–A4 and A10 — one file per operating condition, named `FigAn_<surface>_<load>GPa_<temp>C_<speed>ms`; A5–A6 — one file per swept factor (pressure / temperature / speed); A7–A9 — one file per coefficient type (intercept, load, temperature, speed, roughness) plus R^2.

Options

Grouped at the top of each program (`opt` in MATLAB, the `OPT` dict in Python):

Option	Default	Meaning
`nBootCurve`	`1000`	per-curve bootstrap replicates (1e3 quick … 1e5 for publication-level σ)
`nBootComplete`	`200`	replicates of the 15-coefficient complete model (≈ `nBootCurve`/20 suffices)
`refitMode`	`'local'`	`'local'` (fast) or `'grid'` (exhaustive search, faithful to the SAS BIG scheme)
`progress`	`true`	live progress line during the bootstraps
`bodyFigs` / `appendixFigs`	`true`	toggle each figure group
`makeTables` / `makeXlsx`	`true`	CSV tables / formatted Excel workbook

Reproducibility

The random-number generator is initialised with a **fixed seed**, so a given configuration yields identical tables and figures on every run. Results are deterministic within each language; because MATLAB and NumPy use different generators, bootstrap standard deviations may differ in the 3rd–4th significant digit at low `nBootCurve` and converge as it increases.

Notes

- **Figures 4–5** of the paper compare several bootstrap schemes; this program implements the BIG scheme only and does not regenerate them.
- On digitized curve data the complete-model R^2 is ≈ 0.995; the value ≈ 0.99998 reported in the paper is obtained on the original (less noisy) measurements.
- Runtime scales roughly as `nBootCurve`; a quick run (default) takes on the order of a minute, a full 1e5 run is intended for a final, publication-quality pass.

License

Released under the MIT License (edit to your preference).