



# TerseTS A Framework for Time Series Compression

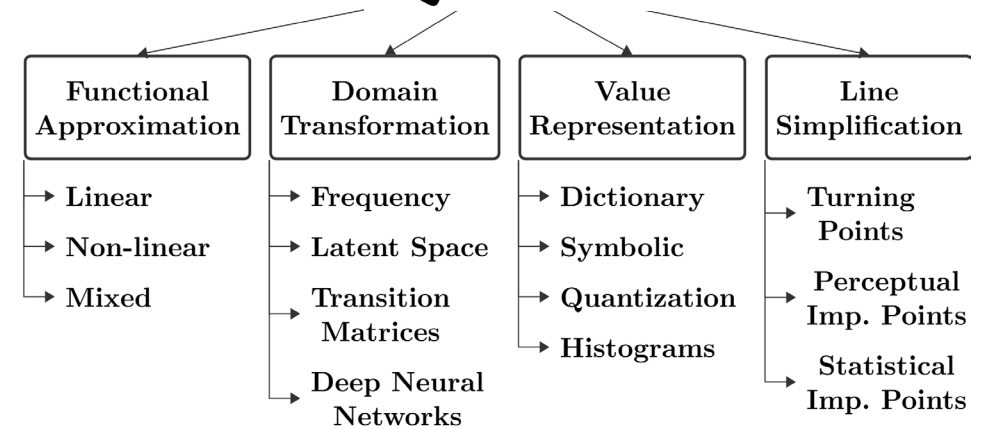
A list of interesting projects to work in.

Carlos Enrique Muñiz-Cuza

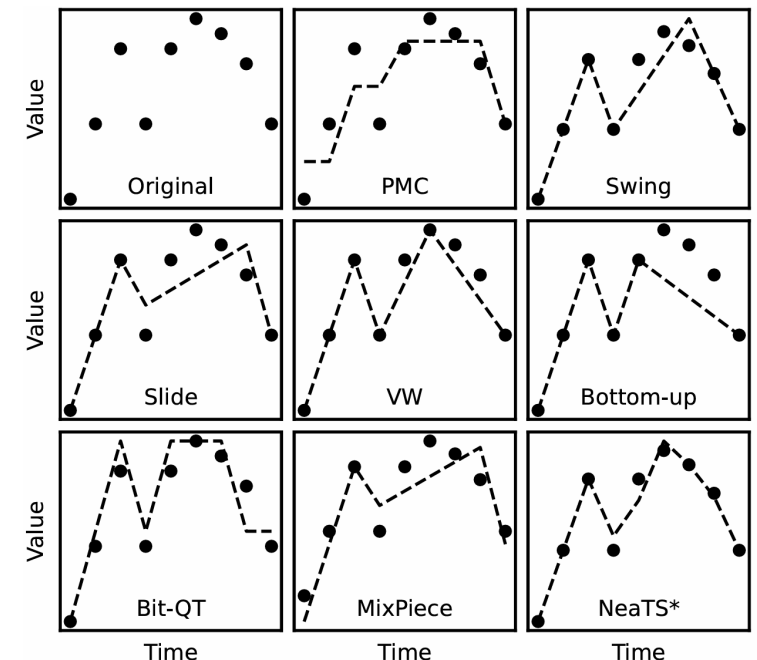
# Introduction to TerseTS



- TerseTS is an open-source native library implemented in **Zig** for efficient time series lossy and lossless compression with bindings for **C, Julia, Python, and Rust**.
- TerseTS lowers the barrier for compression method evaluation, extension, and deployment while improving reproducibility in compression research.
- TerseTS is an open source project licensed under version 2.0 of the Apache License and is hosted on GitHub.
- TerseTS is multi-platform optimized for portability and efficiency on **MacOS, Windows, and Linux**.
- TerseTS received an honorable mention for the Best Demo Paper Award at EDBT Conference.



<https://github.com/cmcuza/TerseTS>



# TerseTS Open Issues

## → AMLS Projects (3 ECTS)

1. Extend TerseTS with the algorithm: Largest Triangle Three Buckets (LTTB). **Issue #110.**
2. Extend TerseTS with lossless compression methods like GORILLA and CHIMP. **Issue #21 and #22.**
3. Extend TerseTS with the CAMEL algorithm. **Issue #59.**
4. Extend TerseTS with the ELF algorithm. **Issue #28.**
5. Extend TerseTS with the SerfXOR algorithm. **Issue #142.**

<https://github.com/cmcuza/TerseTS/issues/>

A screenshot of the GitHub Issues page for the repository 'cmcuza/TerseTS'. The page shows a list of 22 open issues. The issues are sorted by 'Newest' and include details such as issue number, author, and opening date. Some issues have labels like 'bug', 'enhancement', 'question', and 'core'. The issues listed are:

- #141 - Extend TerseTS with lower-bounding-supporting compression methods and distance measures
- #139 - Revise use of usize throughout TerseTS (bug)
- #131 - PMC method failing at zero error bound
- #128 - Simplify memory releasing from C and Python.
- #126 - Revise constness in the C-API (enhancement)
- #125 - Include all methods in Python test.
- #117 - Use Hypothesis for further testing in Python
- #116 - Create special structure for timestamps and coefficients extraction and rebuild.
- #114 - Determine if TerseTS should be binary compatible (question)
- #112 - Add BUFF compression method
- #111 - Add Fast Time Sequence Indexing for Arbitrary Lp Norms
- #110 - Add Largest Triangle Three Buckets (LTTB) algorithm. (core)
- #66 - Implement Bottom-Up, Top-Down, Sliding-Windows, and SWAB
- #65 - use of identity function for hashing in a hashmap
- #62 - Automate Publishing to PyPI Using GitHub Actions
- #59 - Implement Camel (core, enhancement)
- #48 - Implement SHRINK