LFZip: Lossy compression of multivariate time series data via improved prediction

Shubham Chandak
Stanford University

DCC 2020
Paper ID: 111

Joint work with

- Kedar Tatwawadi, Stanford
- Tsachy Weissman, Stanford
- Chengtao Wen, Siemens
- Max Wang, Siemens
- Juan Aparicio, Siemens

Outline

- Motivation
- Problem formulation and our contribution
- Previous work
- Methods
- Results
- Conclusions and future work

Motivation

- Sensors are omnipresent: generating vast amounts of data
- Data usually in form of real-valued time series

Nanopore genome sequencing

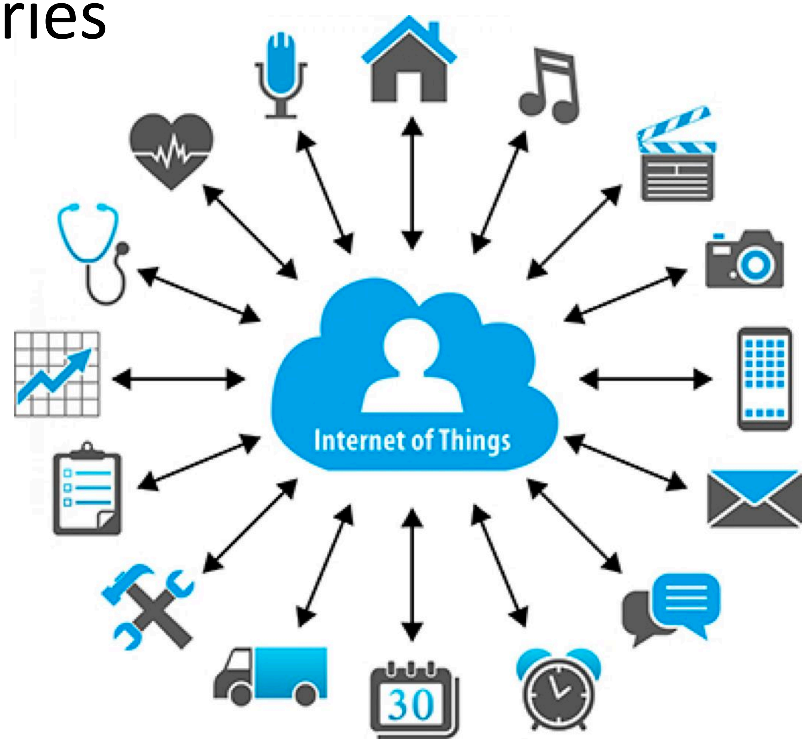


Figure credit:
https://directorsblog.nih.gov/2018/02/06/sequencing-human-genome-with-pocket-sized-nanopore-device/
https://semielectronics.com/sensors-lifeblood-internet-things/

Motivation

- Floating-point time series data typically noisy
  - Lossy compression can lead to vast gains without affecting performance of downstream applications
- Multivariate time series
  - Different variables can have correlations
- Compression performed on computationally constrained devices
  - Low CPU and memory usage (streaming compression)

Problem formulation

$$x_1, x_2, \dots, x_n \xrightarrow{\text{Compress}} \textit{compressed bitstream} \xrightarrow{\text{Decompress}} \hat{x}_1, \hat{x}_2, \dots, \hat{x}_n$$

32-bit floats

$$\text{Compression ratio} = \frac{4 \times n}{Size\ of\ compressed\ bitsream\ in\ bytes}$$

$$\text{Error constraint:}\quad \max_{i=1,\dots,n} |x_i - \hat{x}_i| \le \epsilon$$

Maximum absolute error

Our contribution

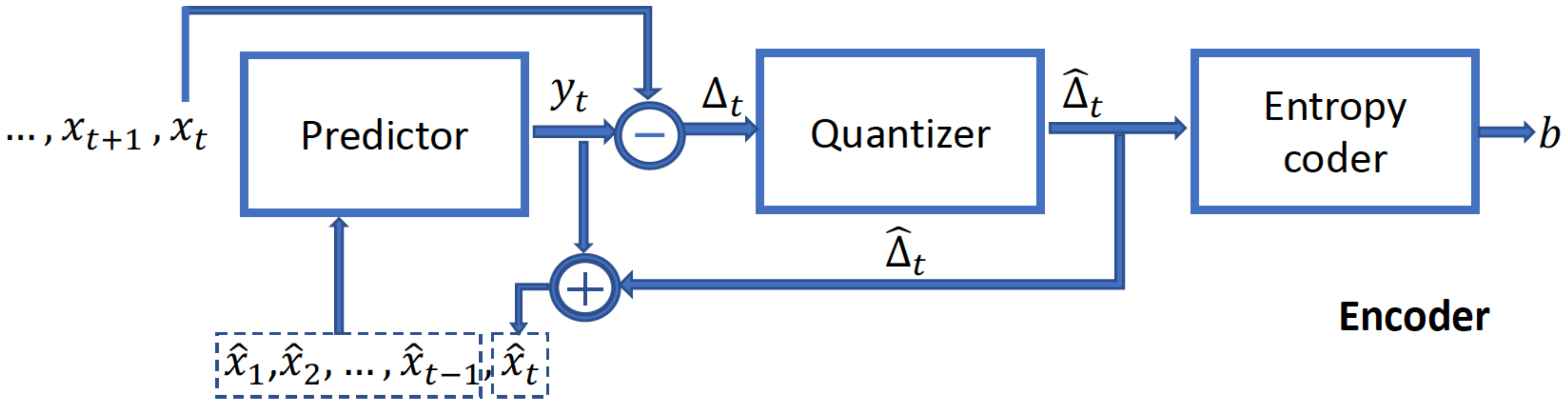
- LFZip: Lossy compressor for time series data
- Works with user-specified maximum absolute error
- Multivariate time series compression
- Based on prediction-quantization-entropy coder framework
  - Normalized Least Mean Squares (NLMS) prediction
  - Neural Network prediction
- Significant improvement for a variety of datasets
- Open source: https://github.com/shubhamchandak94/LFZip

Previous work

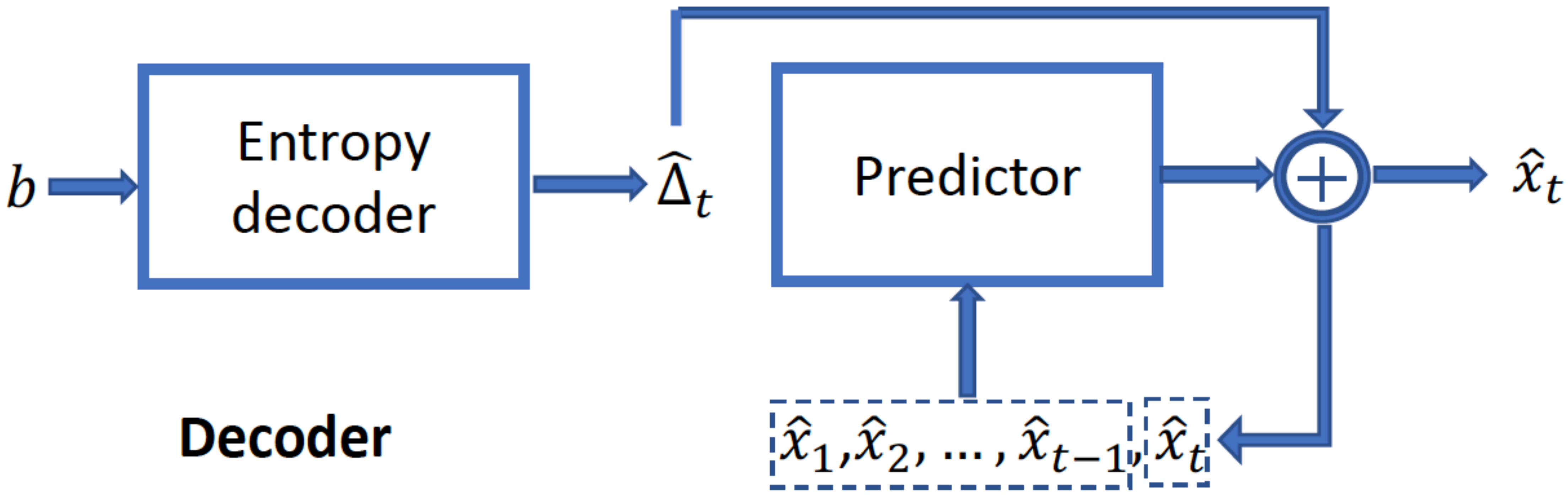
- Swinging door and critical aperture
  - retain a subset of the points in the time series based on the maximum error constraint and use linear interpolation during decompression
- SZ, ISABELA, NUMARCK
  - polynomial/linear regression model followed by quantization
  - SZ current state-of-the-art

- Bristol, E. H. "Swinging door trending: Adaptive trend recording?." *ISA National Conf. Proc., 1990*. 1990.
- Williams, George Edward. "Critical aperture convergence filtering and systems and methods thereof." U.S. Patent No. 7,076,402. 11 Jul. 2006.
- Liang, Xin, et al. "An efficient transformation scheme for lossy data compression with point-wise relative error bound." *2018 IEEE International Conference on Cluster Computing (CLUSTER)*. IEEE, 2018.
- Lakshminarasimhan, Sriram, et al. "ISABELA for effective in situ compression of scientific data." *Concurrency and Computation: Practice and Experience* 25.4 (2013): 524-540.
- Chen, Zhengzhang, et al. "NUMARCK: machine learning algorithm for resiliency and checkpointing." *SC'14: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. IEEE, 2014.

Encoder architecture



Decoder architecture



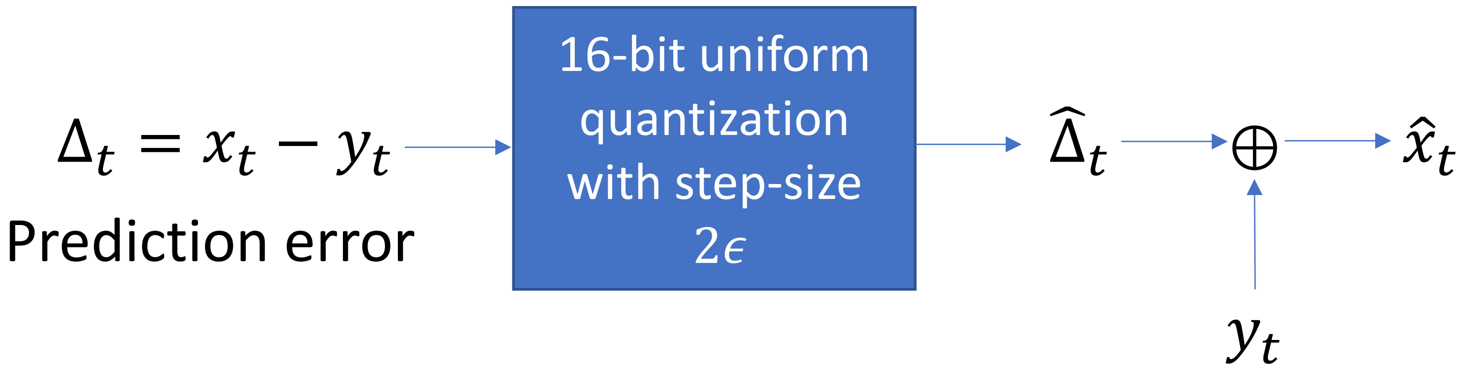
Predictor

- Predict based on past window (default 32 steps)
- NLMS (normalized least mean square)
  - Adaptively trained (gradient descent) after every step
  - Multivariate: predict based on past values for all variables

Predictor

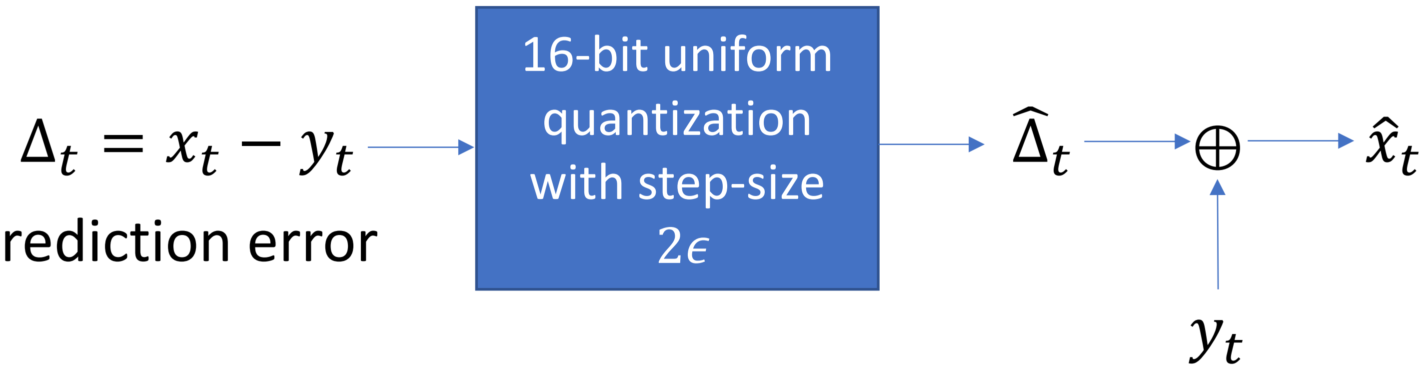
- Predict based on past window (default 32 steps)
- NLMS (normalized least mean square)
  - Adaptively trained (gradient descent) after every step
  - Multivariate: predict based on past values for all variables
- NN (neural network)
  - Offline training performed on separate dataset
  - We tested fully connected (FC) and RNN models (results shown for FC)
  - To simulate quantization error during training, we add random noise

Quantizer and entropy coder



- If prediction error above $2^{16}\epsilon$, set $\hat{x}_t = x_t$

Quantizer and entropy coder

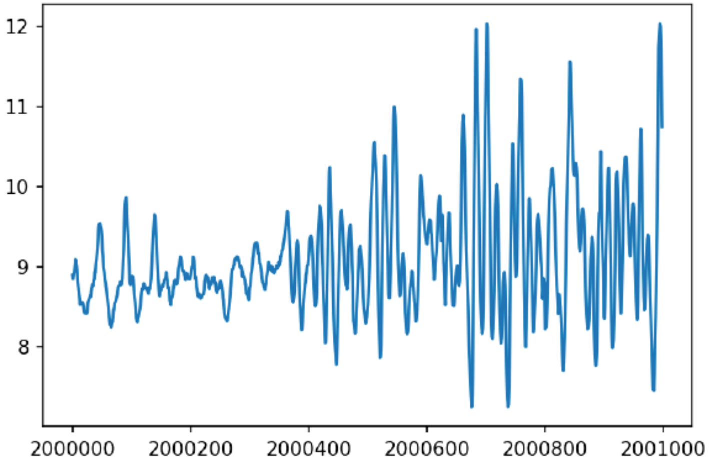


- If prediction error above $2^{16}\epsilon$, set $\hat{x}_t$ = x_t
- Entropy coding: BSC (https://github.com/IlyaGrebnov/libbsc)
  - High performance compressor based on BWT

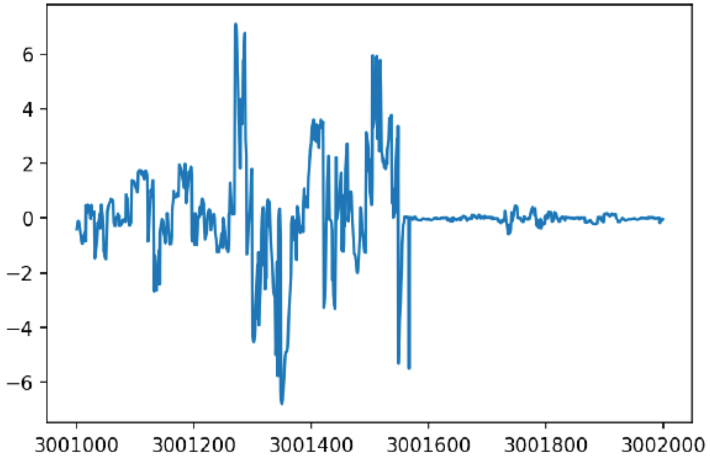
Results: datasets

Name	Length	Description	BSC lossless compression ratio
acc	3.54M	Heterogeneity Activity Recognition - smartwatch accelerometer [24]	2.84
gyr	3.21M	Heterogeneity Activity Recognition - smartwatch gyroscope [24]	2.79
pow	2.05M	Household electric power consumption - active power [25]	5.21
ppg	0.50M	Blood volume pulse/photoplethysmography (PPG) [26]	2.48
gas	0.93M	Home activity monitoring - MOX gas sensors resistance [27]	4.97
dna	1.17M	Nanopore DNA sequencing raw current data	4.55
vib	1.55M	Siemens healthy tool vibration data	1.79
sen	0.75M	Siemens sensor data	4.27

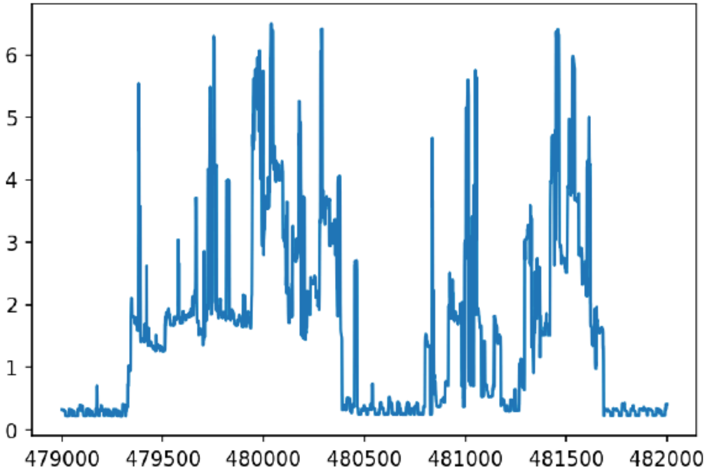
Results: datasets



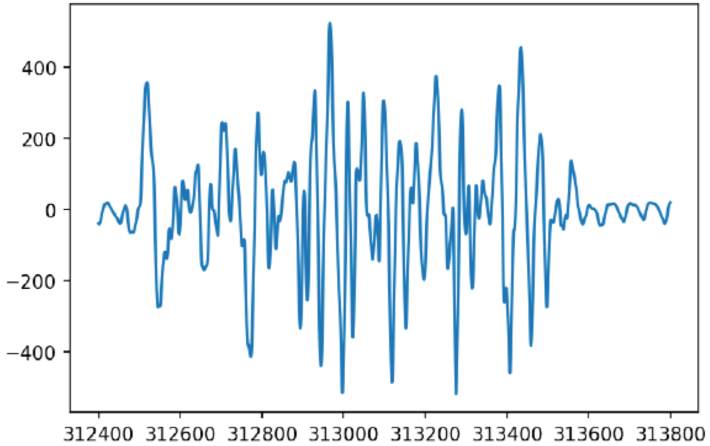
(a) *acc*



(b) *gyr*

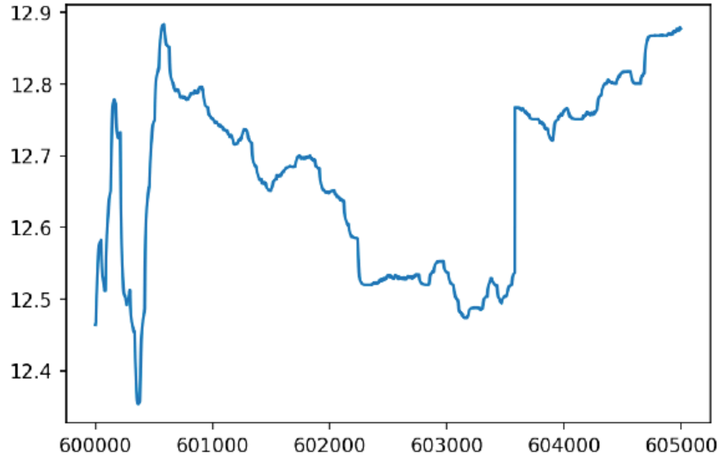


(c) *pow*

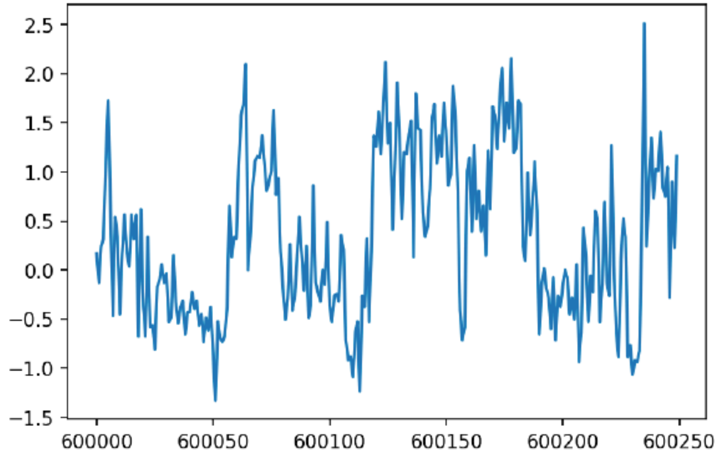


(d) *ppg*

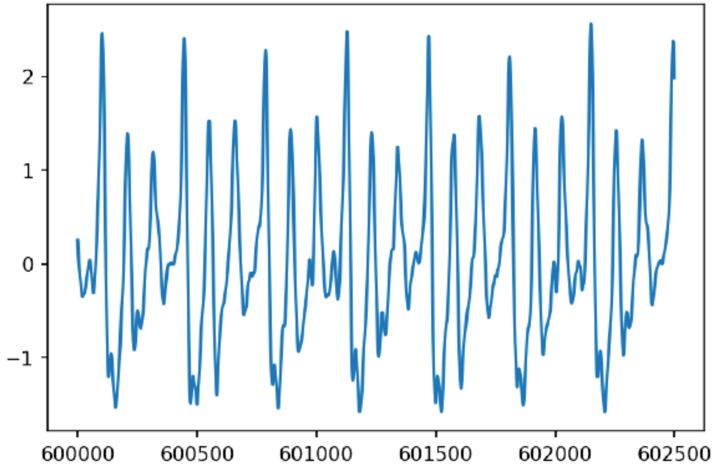
Results: datasets



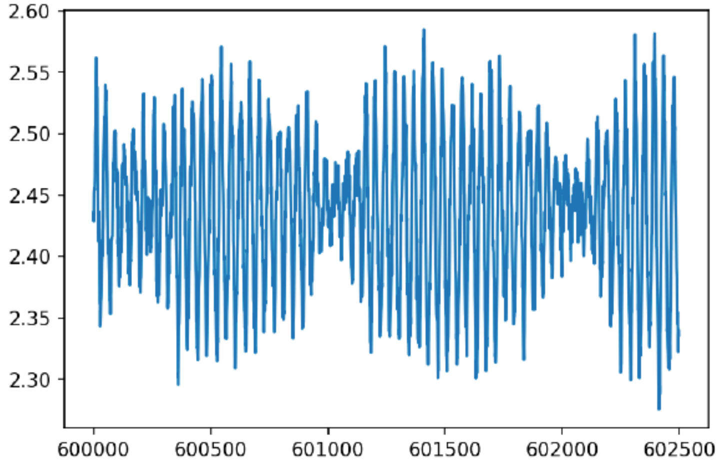
(e) *gas*



(f) *dna*



(g) *vib*



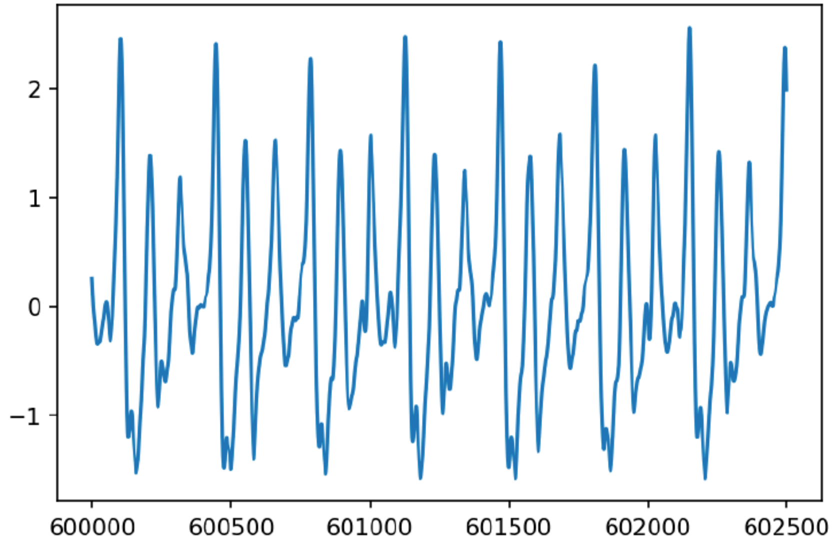
(h) *sen*

Results: univariate (NLMS prediction)

Dataset	Compressor	Maximum error ϵ		
		10^{-3}	10^{-2}	10^{-1}
acc	CA	2.84	3.01	5.19
	SZ	3.25	5.05	11.00
	LFZip (NLMS)	**3.55**	**5.86**	**12.71**
gyr	CA	2.88	4.27	10.75
	SZ	4.26	8.08	24.79
	LFZip (NLMS)	**6.05**	**12.26**	**28.77**
pow	CA	5.05	6.23	12.47
	SZ	**5.09**	**9.65**	**23.99**
	LFZip (NLMS)	4.17	7.37	17.98
ppg	CA	2.48	2.49	2.74
	SZ	2.43	2.80	4.39
	LFZip (NLMS)	**3.18**	**5.28**	**9.13**

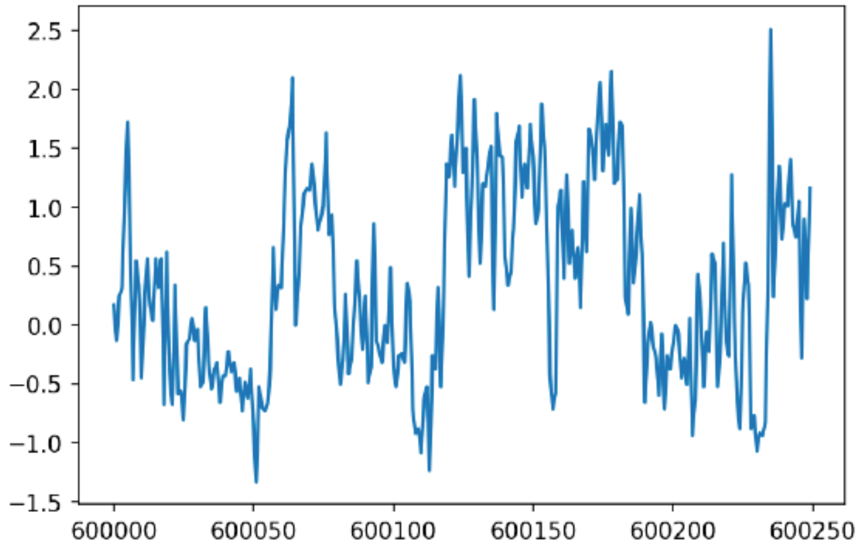
Dataset	Compressor	Maximum error ϵ		
		10^{-3}	10^{-2}	10^{-1}
gas	CA	16.97	64.36	245.51
	SZ	22.69	75.84	**299.65**
	LFZip (NLMS)	**31.56**	**101.48**	252.55
dna	CA	**4.54**	4.54	4.86
	SZ	4.03	**4.55**	**8.62**
	LFZip (NLMS)	3.04	4.48	8.40
vib	CA	2.07	4.85	18.51
	SZ	4.77	11.77	40.61
	LFZip (NLMS)	**10.64**	**22.36**	**53.15**
sen	CA	4.34	7.60	125.04
	SZ	6.55	20.58	179.87
	LFZip (NLMS)	**6.88**	**21.70**	**180.98**

Results: univariate (NLMS prediction)



(g) *vib*

LFZip performs better



(f) *dna*

LFZip performs worse

Results: univariate (NN prediction)

Dataset	Compressor	Maximum error ϵ 10^{-2}	10^{-1}
acc	SZ	4.64	9.38
	LFZip (NLMS)	5.10	10.19
	LFZip (NN)	**5.26**	**10.78**
gyr	SZ	6.99	20.96
	LFZip (NLMS)	10.22	23.33
	LFZip (NN)	**10.35**	**25.00**
pow	SZ	**9.44**	23.57
	LFZip (NLMS)	7.21	17.74
	LFZip (NN)	9.29	**25.38**
dna	SZ	4.45	8.67
	LFZip (NLMS)	4.46	8.40
	LFZip (NN)	**4.60**	**8.99**

Results: multivariate (NLMS prediction)

Dataset	Mode	Maximum error ϵ		
		10^{-3}	10^{-2}	10^{-1}
acc (X, Y, Z)	univariate	3.588	5.931	13.220
	multivariate	3.592	5.934	13.250
gyr (X, Y, Z)	univariate	6.295	13.605	34.181
	multivariate	6.409	13.763	34.597
gas (8 sensors)	univariate	26.239	63.304	152.378
	multivariate	27.614	75.179	204.006
sen (3 sensors)	univariate	6.627	19.669	166.568
	multivariate	6.669	20.334	304.878

Results: computation

- LFZip (NLMS): ~2M timesteps/s for univariate
  - Slower than SZ but practical for most applications
- LFZip (NN): ~1K timesteps/s for the fully connected model used
  - Run single-threaded on a CPU to allow reproducibility
  - Requires further optimizations for practical usage

Conclusions and future work

- LFZip: error-bounded lossy compressor for multivariate floating-point time series
- Based on prediction-quantization-entropy coder framework
- Achieve improved compression using NLMS and NN models

Conclusions and future work

- LFZip: error-bounded lossy compressor for multivariate floating-point time series
- Based on prediction-quantization-entropy coder framework
- Achieve improved compression using NLMS and NN models
- Future work includes
    - optimized implementation for the neural network based framework
    - extension of the framework to multidimensional datasets
    - exploration of other predictive models to further boost compression

Thank You!

Check out

https://github.com/shubhamchandak94/LFZip