STAC-ML Update

Bishop Brock
Head of Research, STAC
bishop.brock@STACresearch.com

STAC-ML Markets (Training) Benchmark: Underway

- Existing ML training benchmarks are not *specific* to Finance:
  - They focus on qualitative problems
  - Finance requires good quantitative models
- We spoke to many both inside and outside of the Working Group
- Came back to the Working Group with several candidate use cases
  - Value to the end user
  - The ability to fairly evaluate the quality of benchmark solutions
- Consensus – Focus on complex derivative modelling
- Now detailing a proposal - **Join us!**

www.STACresearch.com/ML

STAC®
SECURITIES TECHNOLOGY ANALYSIS CENTER

STAC-ML Markets (Inference)

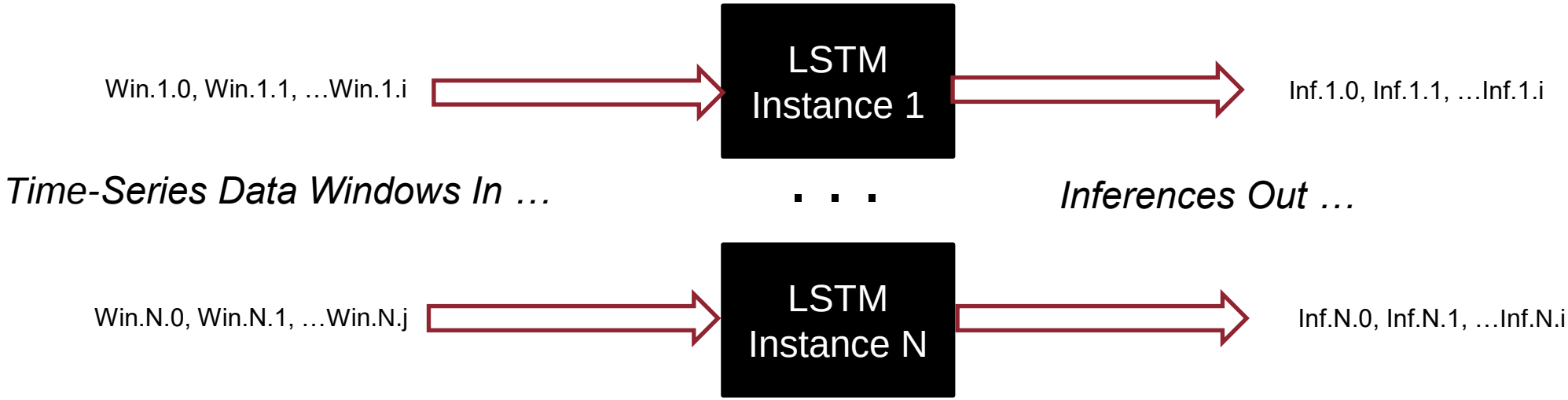
NEWS FLASH

- First audited results from Groq announced today!
- Will get to that shortly…

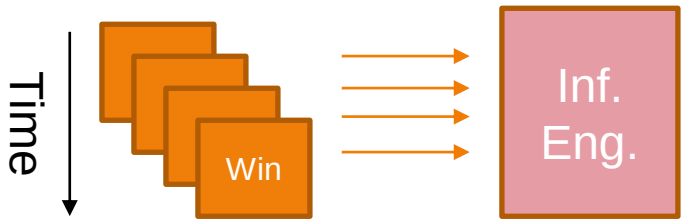
Background - STAC-ML Markets (Inference)

- STAC-ML provides a framework for full-stack evaluation
- Three users of STAC-ML
  - STAC
  - Vendors
  - Financial firms
- I will talk about all three

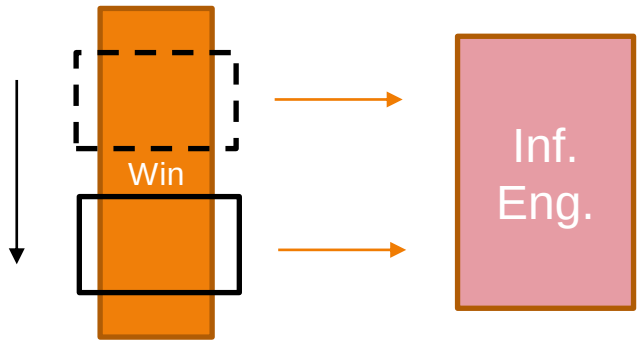
Time-Series Inference using LSTM Models: Perf./Eff./Scalability



- Sumaco – Fixed, Unique Window



- Tacaná - Sliding Window (Streaming)



STAC®
SECURITIES TECHNOLOGY ANALYSIS CENTER

Research Available to ML STAC-Track Subscribers

- GCP Cloud SUT
  - Latency- and Throughput-optimized configurations for ONNX inference
- TensorFlow Performance (on CPU)
  - Looked at different ways to configure TensorFlow for inference
- Azure Cloud-SUT Jamboree (Coming up)
- All research available via free trial for remainder of 2022
  - For those responsible for ML research and infrastructure

council@STACresearch.com

STAC®
SECURITIES TECHNOLOGY ANALYSIS CENTER

STAC-ML Markets (Inference) Azure Cloud-SUT Jamboree!

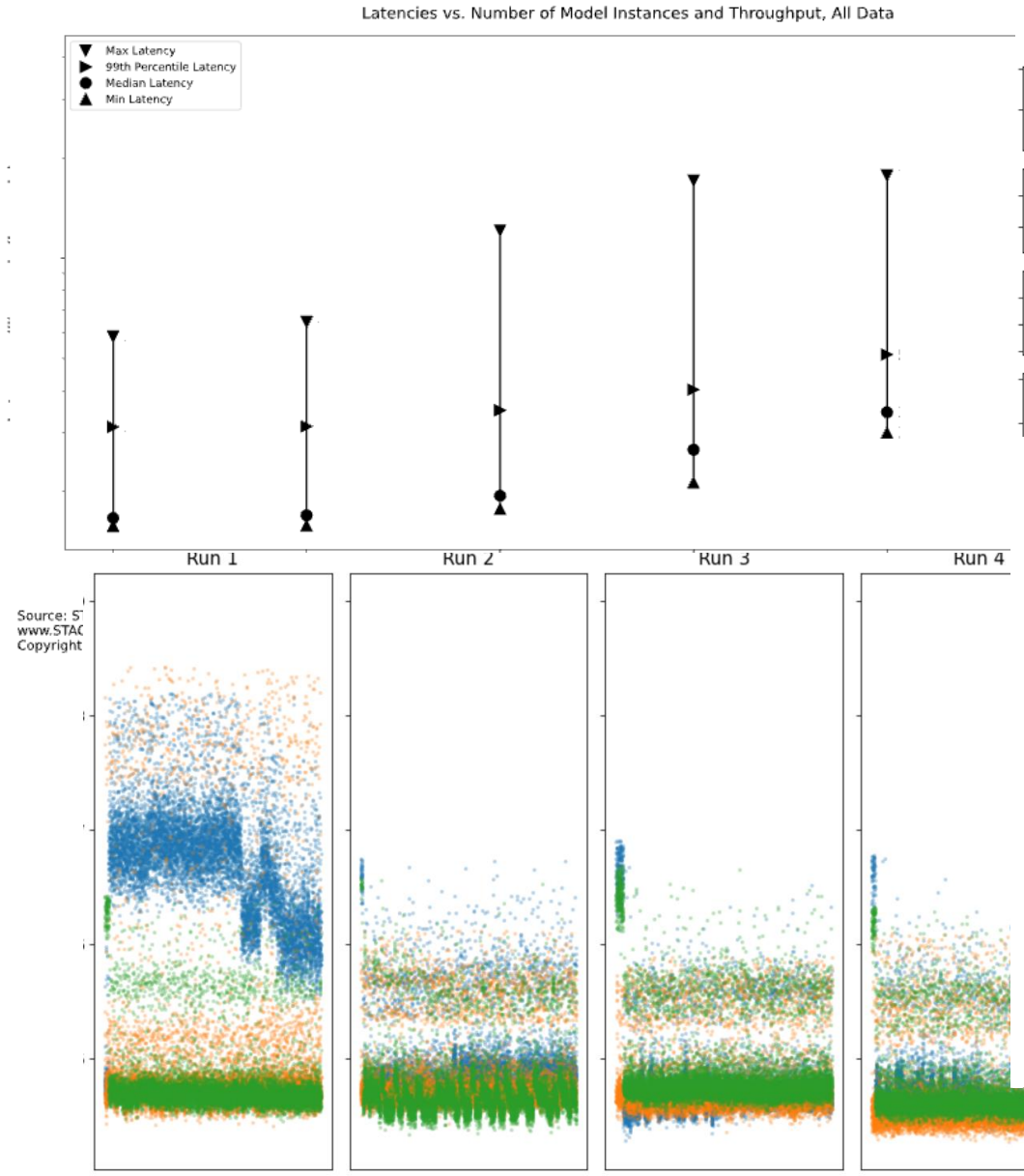
- Goal: compare 3 CPU architectures for inference
  - Intel, AMD, Ampere (ARM)
- Used the STAC "Naïve" Python implementation with ONNX
- Tested on Microsoft Azure

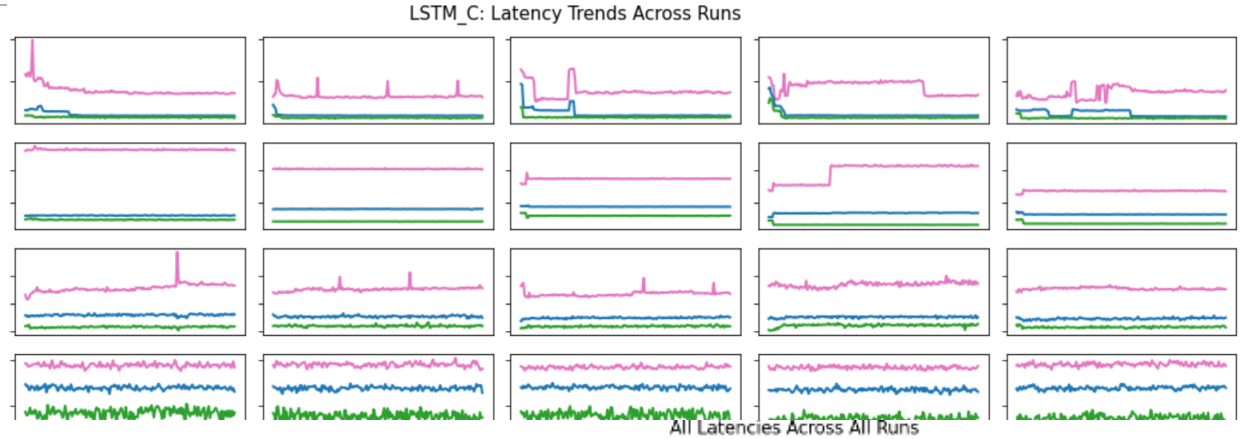
Thanks to Microsoft for supporting the STAC community by providing credits for this research!

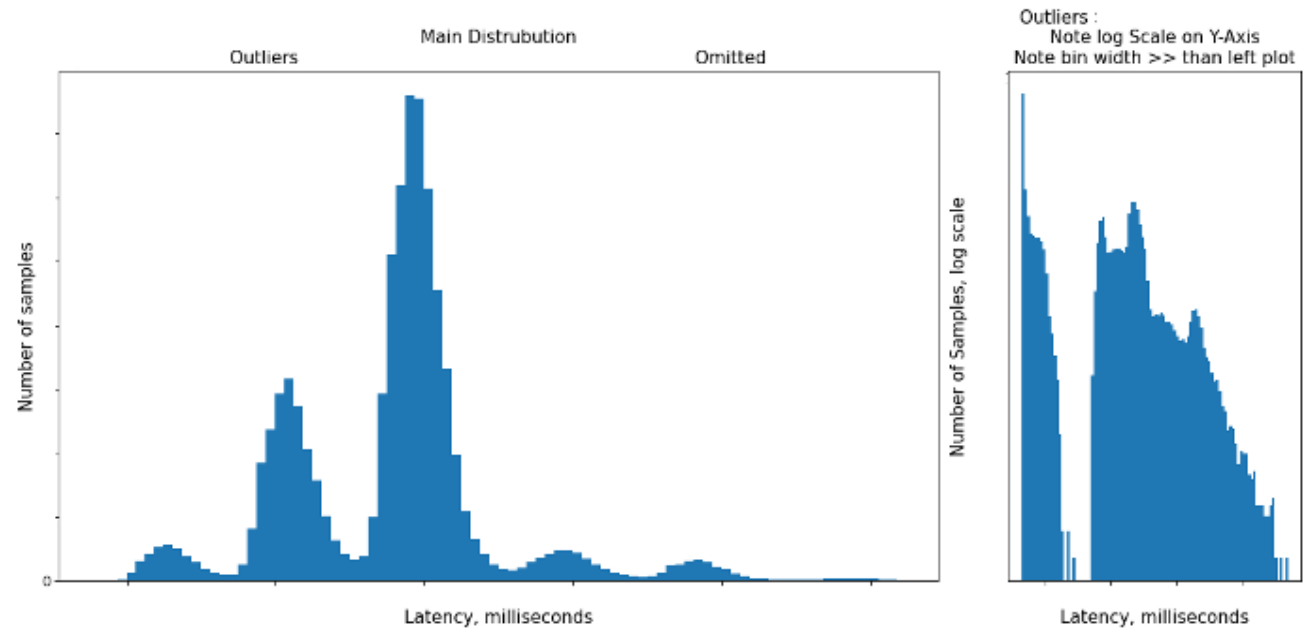
- Tested two configs for each VM (latency optimized, tput optimized)
- All 6 reports & comparison report will soon be in the STAC Vault
- No vendors participated in the setup and optimization of the SUTs

STAC®
SECURITIES TECHNOLOGY ANALYSIS CENTER

Detailed analysis available for each SUT

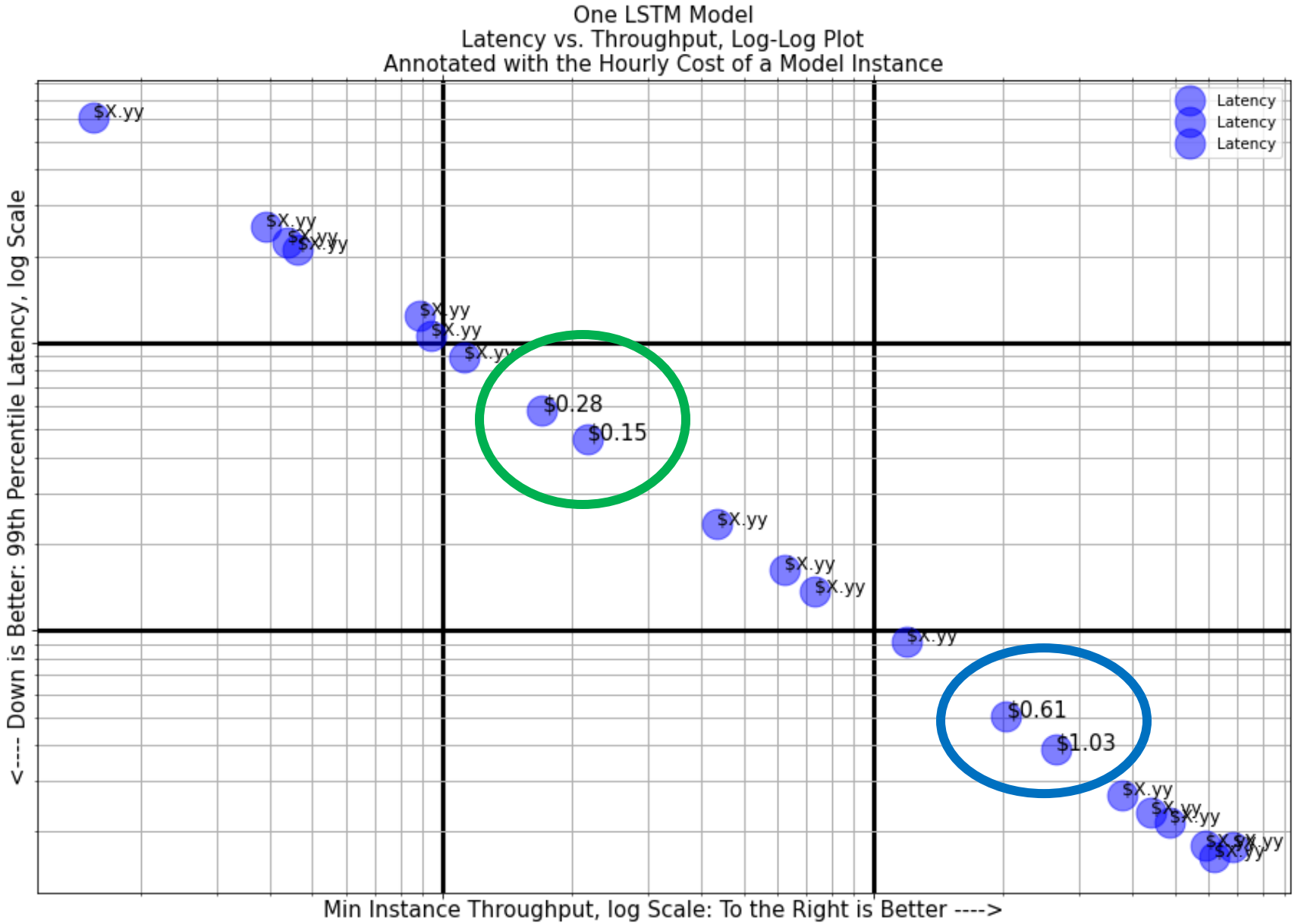






STAC®
SECURITIES TECHNOLOGY ANALYSIS CENTER

Research Summary Note: Business-oriented comparisons



STAC®
SECURITIES TECHNOLOGY ANALYSIS CENTER

First public tested SUT!

- STAC-ML Pack for GroqWare (Rev A)
  - Version of STAC “Naïve” implementation adapted for GroqWare APIs
- GroqWare™ SDK 0.9.0.5 devtools and runtime
- Python 3.8.15; NumPy 1.23.4
- Ubuntu Linux 22.04.1 LTS
- GroqNode™ GN1-B8C-ES:
  - 8 x TSP-100 A1.4b 10b GroqCards™
  - 2 x AMD EPYC™ 7413 24-core Processors @ 2650 MHz
  - 16 slots x 64GiB DDR4 - 1024GiB Total

STAC®
SECURITIES TECHNOLOGY ANALYSIS CENTER

Results highlights

For the small model (LSTM_A) at 1, 2, & 4 model instances

- Min latency to 99P only varied by 6%
- 99P latency only varied by 1% across these numbers of model instances
- Worst case 99P latency was 56.4 microseconds

Results highlights

For large model (LSTM_C) for 1 to 8 model instances:

- Minimum to 99p latency only varied by 3%
- 99P latency only varied by 2% across these numbers of model instances
- Worst 99P latency was 2.77 milliseconds

STAC-ML tools are ready for you, too

- Vendor implementations – See how it works
- Test harness software and analysis tools – Test your own stacks
  - In fact, test your own models!

www.STACresearch.com/ML

STAC®
SECURITIES TECHNOLOGY ANALYSIS CENTER