

# Accelerate tSNE with GPU

Over 30x faster tSNE than Sklearn.



AVI CHAWLA

MAY 24, 2024



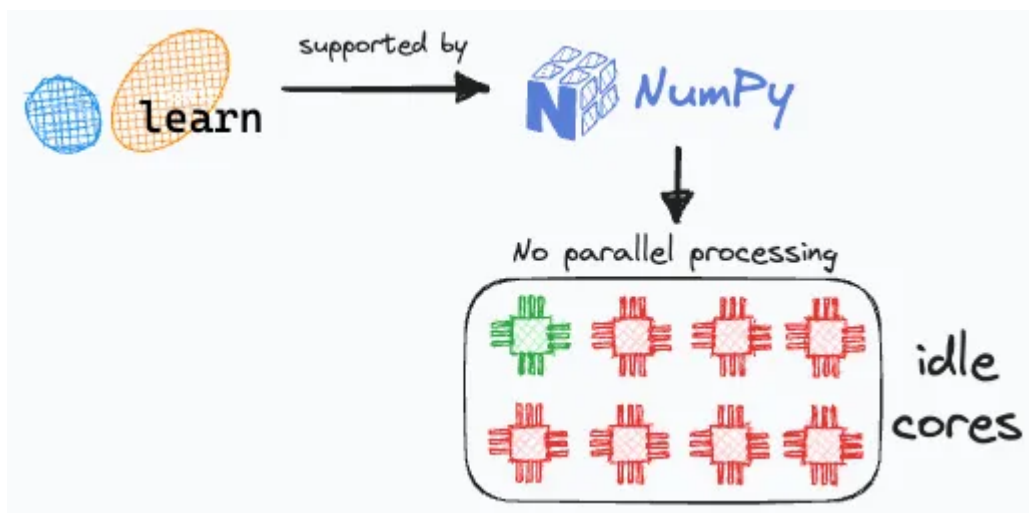
13



Share

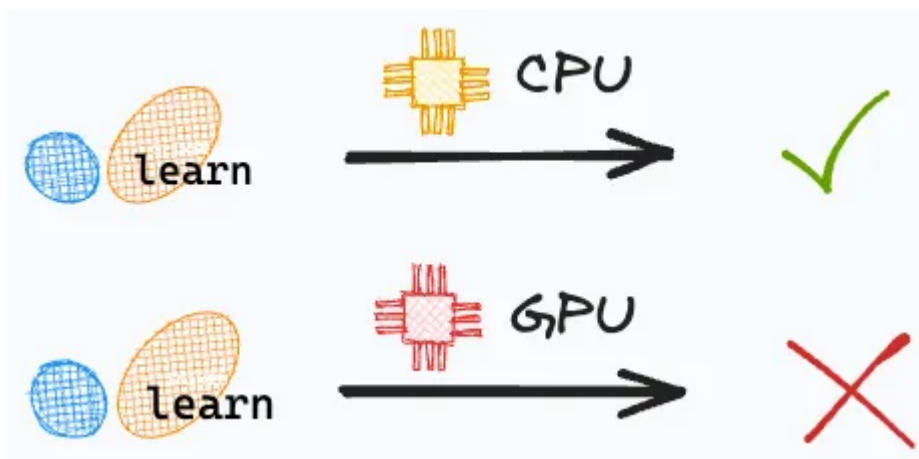


Sklearn implementations are driven by NumPy, which runs on a single core of a CPU. Thus, the ability to run parallelized operations is quite limited at times.



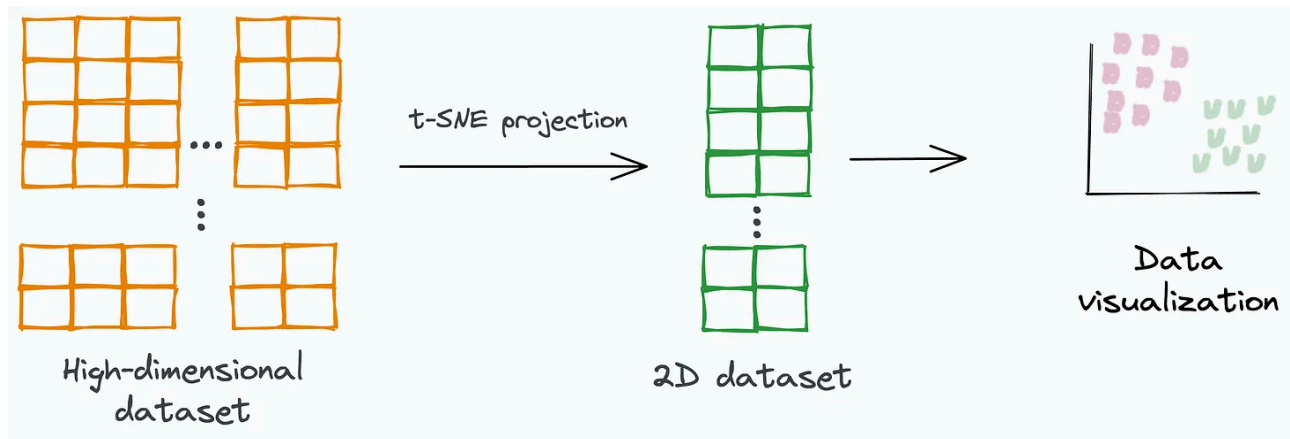
Of course, this is not applicable to all algorithms.

Another major limitation is that scikit-learn models cannot run on GPUs.

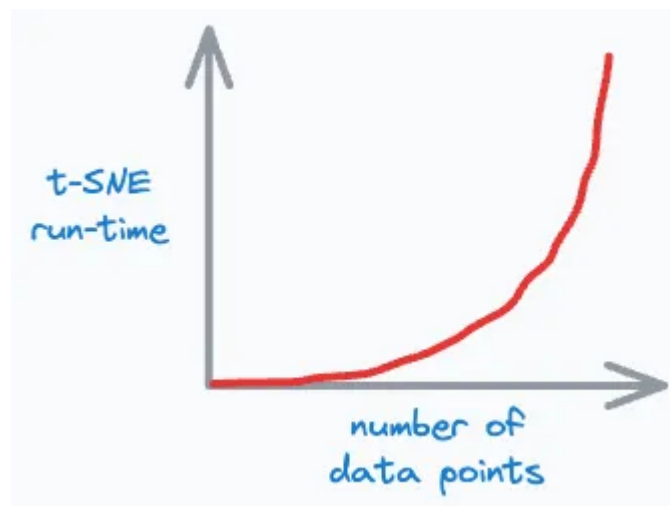


This bottleneck provides massive room for run-time improvement.

The same applies to the tSNE algorithm, which is among the most powerful dimensionality reduction techniques to visualize high-dimensional datasets.



Because the biggest issue with tSNE ([which we also discussed in the tSNE article here](#)) is that its run-time is quadratically related to the number of data points.



Thus, beyond, say, 20k-25k data points, it becomes pretty difficult to use tSNE from Sklearn implementations.

There are two ways to handle this:

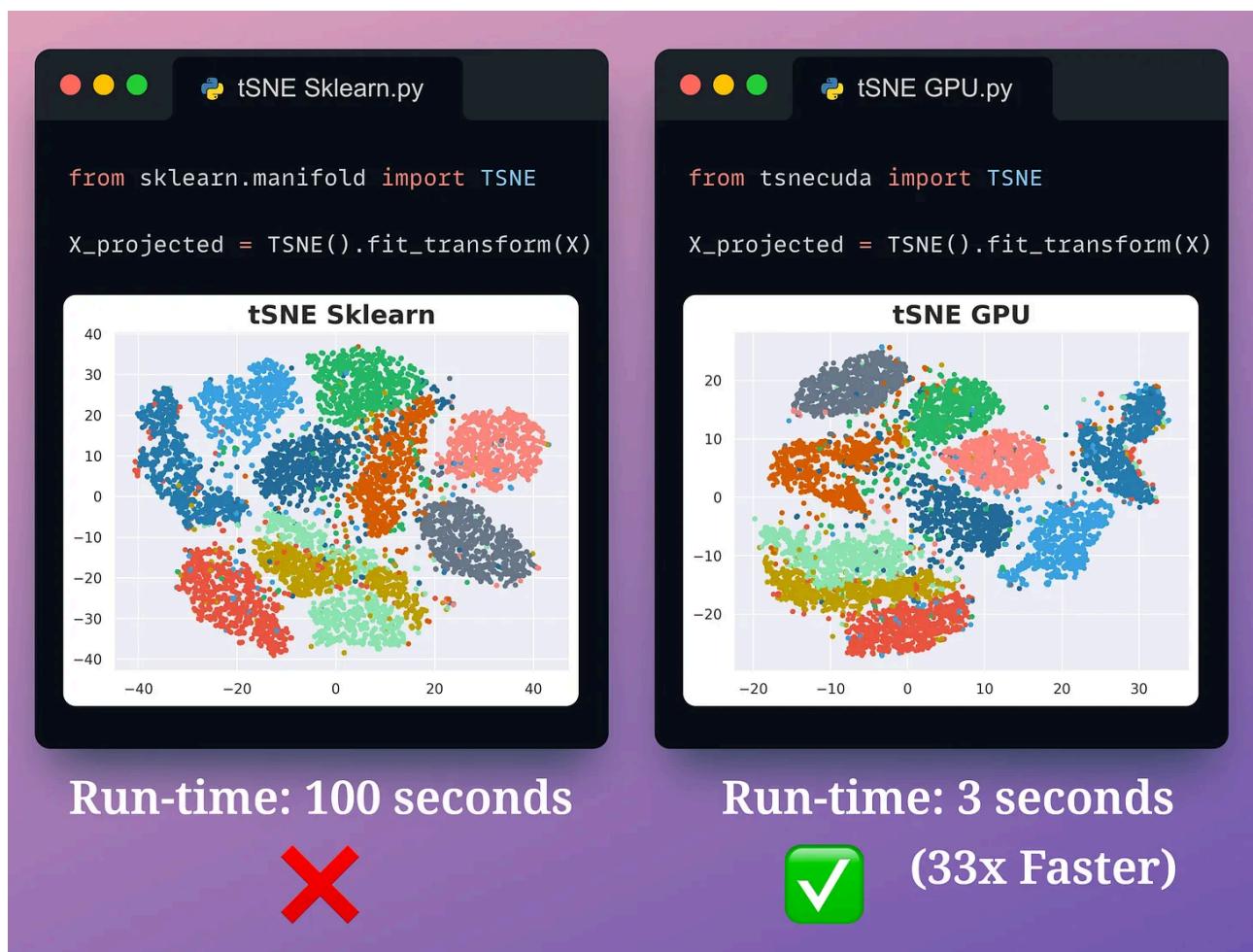
- Either keep waiting.

- Or use optimized implementations that could be possibly accelerated with GPU.

Recently, I was stuck due to the same issue in one of my projects, and I found a pretty handy solution that I want to share with you today.

**tSNE-CUDA** is an optimized CUDA version of the tSNE algorithm, which, as the name suggests, can leverage hardware accelerators.

As a result, it provides immense speedups over the standard Sklearn implementation, which is evident from the image below:



As depicted above, the GPU-accelerated implementation:

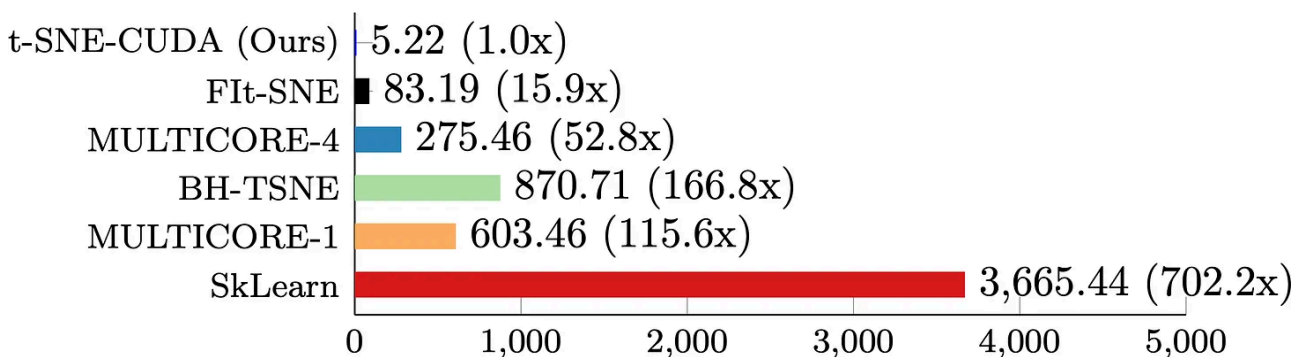
- Is 33 times faster than the Sklearn implementation.
- Produces similar quality clustering as the Sklearn implementation.

Do note that this implementation only supports `n_components=2`, i.e., you can only project to two dimensions.

As per the docs, the authors have no plans to support more dimensions, as this will require significant changes to the code.

But in my opinion, that doesn't matter because, for more than 99% of cases, tSNE is used to obtain 2D projections. So we are good here.

Before I conclude, I also found the following benchmarking results by the authors:



It depicts that on the CIFAR-10 training set (50k images), **tSNE-CUDA is 700x Faster than Sklearn, which is an insane speedup.**

Isn't that cool?

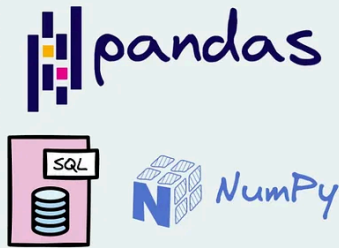


I prepared this Colab notebook for you to get started: [tSNE-CUDA Colab Notebook](#).

### Further reading:

- While this was just about tSNE, do you know we can accelerate other ML algorithms with GPUs? Read this article to learn more: [Sklearn Models are Not Deployment Friendly! Supercharge Them With Tensor Computations](#).
- Also, do you know how tSNE works end-to-end? Read this article to learn more: [Formulating and Implementing the t-SNE Algorithm From Scratch](#)

👉 Over to you: What are some other ways to boost the tSNE algorithm?

Thanks for reading Daily Dose of Data Science!  
Subscribe for free to learn something new and insightful about Python and Data Science every day. Also, get a Free Data Science PDF (550+ pages) with 320+ tips.

1 Referral	2 Referrals	3 Referrals
450+ practice questions on...	Beginner to advanced Python OOP article (code included)	Develop big-data mastery with...
		 *Beginner-friendly and code included

- **1 Referral:** Unlock 450+ practice questions on NumPy, Pandas, and SQL.
- **2 Referrals:** Get access to advanced Python OOP deep dive.
- **3 Referrals:** Get access to the PySpark deep dive for big-data mastery.

Get your unique referral link:

## Let me help you more...

Every week, I publish in-depth ML deep dives. The topics align with the practical skills that typical ML/DS roles demand.

Join below to unlock all full articles:

Here are some of the top articles:

- [\[FREE\] A Beginner-friendly Introduction to Kolmogorov Arnold Networks \(KANs\).](#)
- [Implementing Parallelized CUDA Programs From Scratch Using CUDA Programming](#)
- [Understanding LoRA-derived Techniques for Optimal LLM Fine-tuning](#)
- [8 Fatal \(Yet Non-obvious\) Pitfalls and Cautionary Measures in Data Science.](#)
- [5 Must-Know Ways to Test ML Models in Production \(Implementation Included\).](#)
- [11 Powerful Techniques To Supercharge Your ML Models.](#)
- [A Beginner-Friendly Guide to Multi-GPU Model Training.](#)
- [Don't Stop at Pandas and Sklearn! Get Started with Spark DataFrames and Big Data ML using PySpark.](#)

Join below to unlock all full articles:

## SPONSOR US

Get your product in front of more than 76,000 data scientists and other tech professionals.

Our newsletter puts your products and services directly in front of an audience that matters — thousands of leaders, senior data scientists, machine learning engineers, data analysts, etc., who have influence over significant tech decisions and big purchases.

To ensure your product reaches this influential audience, reserve your space [here](#) or reply to this email.



13 Likes

← Previous

## Comments



Write a comment...

---

© 2024 Avi Chawla · [Privacy](#) · [Terms](#) · [Collection notice](#)  
[Substack](#) is the home for great culture