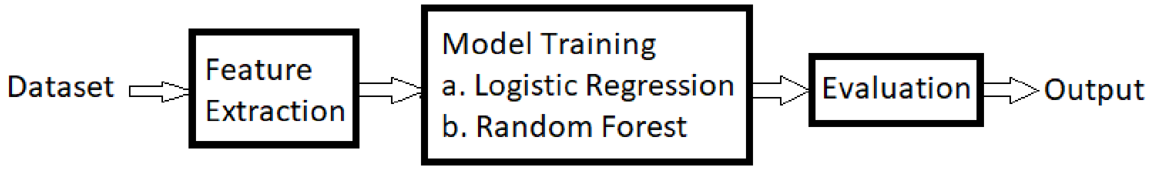
Project: CPU & GPU performance analysis (clustering)
(This project is done as part of the machine learning course in M.Tech at IIT Madras)
(Datasheet: https://www.kaggle.com/datasets/michaelbryantds/cpu-and-gpu-product-data)

Introduction:

This project aims to explore, analyze, and model the evolution and performance trends of CPU and GPU hardware using a machine learning approach. By leveraging real-world chip data, identify patterns in design parameters and apply supervised learning techniques to derive meaningful insights from the datasheet.

System Model:



The system model consists of three key stages:

1. **Feature Extraction:**
   In this stage, raw data from the chip dataset is cleaned and relevant numerical features are extracted. These features include process size, TDP, die size, number of transistors, and frequency. The features are then normalized using standard scaling techniques to prepare them for modelling.
2. **Model Training:**
   Two supervised machine learning models are trained:
   - **Logistic Regression:** used to predict the chip vendor (e.g., AMD, Intel, NVIDIA).
   - **Random Forest Classifier:** used to classify whether the chip is a CPU or a GPU. These models are trained using a portion of the dataset with known labels.
3. **Prediction & Evaluation:**
   After training, the models are tested using unseen samples to predict vendor and chip type. The performance of both models is evaluated using standard metrics like precision, recall, and F1-score. Test cases are also used from the dataset to validate predictions manually.

Problem Statement:

The goal of this project is to analyse the performance evolution of CPUs and GPUs using machine learning techniques. The dataset comprises hundreds of chip records from multiple

vendors, including specifications like process size, die size, power consumption (TDP), number of transistors, and operating frequency.
This project aims to solve the following problems:

1. **Trend Analysis Over Time:**
   Identify how CPUs and GPUs have evolved over the years across key design parameters — particularly focusing on the changes in process size, die size, transistor count, and frequency. This helps in understanding the industry's response to physical and technological constraints, including the breakdown of Dennard scaling.
2. **Vendor Comparison:**
   Compare chip development strategies across major vendors like Intel, AMD, and NVIDIA. The project investigates how these companies have prioritized performance, efficiency, and scaling over the years by plotting and analysing their chip's specifications.
3. **Feature Space Visualization with PCA:**
   Apply **Principal Component Analysis (PCA)** to reduce chip feature dimensions and visualize the dataset in two-dimensional space. This helps identify natural separability between CPUs and GPUs, and observe how densely chips are grouped in feature space based on their characteristics.
4. **Predictive Modelling:**
   Build two supervised models:
   - A model to predict the **vendor** of a chip based on its technical specifications.
   - A model to classify whether the chip is a **CPU or GPU**. These models are trained on a subset of the dataset and evaluated using test data, including known samples from the original datasheet.

Through these goals, the project demonstrates the practical applicability of machine learning in hardware performance analysis, visualization, and classification.

Solution/Methodology:

- **Data Cleaning & Normalization:**
  Filled missing values with column medians and standardized all features using StandardScaler.
- **EDA (Exploratory Data Analysis):**
  Line and scatter plots were generated to show trends in chip design over the years across various metrics and vendors.
- **PCA for Visualization:**
  PCA was applied to project the 5D chip feature space into 2D. This projection was used to visualize separability of CPUs vs GPUs and vendor groupings.
- **Supervised Learning:**
  - Logistic Regression model was trained to predict the chip **vendor**
  - Random Forest model was trained to predict whether the chip is a **CPU or GPU**
- **Model Testing:**
  Real datapoints from the dataset were passed to the trained models and predictions were printed, showing practical use.

Theoretical Analysis:

1. Feature Representation

The dataset contains numerical attributes that define chip characteristics:

- **Process Size (nm):** Fabrication node size; smaller sizes often mean faster and more power-efficient chips.
- **TDP (W):** Thermal Design Power; represents maximum power a chip can draw.
- **Die Size (mm^2):** Physical silicon area; often increases with transistor count.
- **Transistors (million):** A key metric for performance; increases over time as per Moore's Law.
- **Frequency (MHz):** Clock speed; relates to how many operations a chip can perform per second.

2. Feature Scaling (Standardization)

Before applying PCA or training ML models, the data was standardized using **StandardScaler**. This transforms each feature to have a mean of 0 and standard deviation of 1. This is required because, features like frequency (in MHz) and process size (in nm) are on very different scales. Without scaling, PCA and ML models would be biased toward features with larger magnitudes.

3. Dimensionality Reduction using PCA

Principal Component Analysis (PCA) is a mathematical technique used to reduce the dimensionality of data while retaining as much variance as possible.

- PCA transforms the original features into new uncorrelated variables called **principal components**.
- The first principal component captures the maximum variance; the second captures the next highest, and so on.
- Reduced the data to **2 dimensions** for visual analysis and interpretation.

PCA was used not for prediction but for visualizing the separability between CPUs and GPUs and analysing density and similarity using distances in 2D PCA space.

4. Distance-Based Grouping with KNN

Computed the local neighborhood distances using **K-Nearest Neighbors (KNN)** in the PCA-reduced space.

- For each chip, find its 3 nearest neighbors and compute the average Euclidean distance to them.
- Chips that were similar in architecture appeared closer in PCA space, and those with unique characteristics had higher distances.

5. Supervised Learning for Classification

Implemented two classification tasks:

1. **Vendor Prediction (Logistic Regression)**
   Logistic Regression is a linear model for multi-class classification. It estimates the probability that a given input belongs to a particular vendor (e.g., Intel, AMD).
   - Works well with standardized features
   - Easy to interpret and analyze
   - Good baseline model

2. **CPU vs GPU Classification (Random Forest)**
   Random Forest is an ensemble method that builds multiple decision trees and averages their outputs.
   - Handles non-linear relationships and feature interactions well
   - Provides high accuracy with minimal tuning
   - Robust to overfitting on medium-sized datasets

Both models were trained on 80% of the data and tested on 20%, achieving strong performance as seen in the classification reports.

6. Model Evaluation and Test Prediction

To verify the models' utility:

- Manually selected 2 clean test entries from the dataset.
- These entries were passed to both models.
- The predicted vendor and chip type matched the expected values, confirming the models' correctness on real examples.

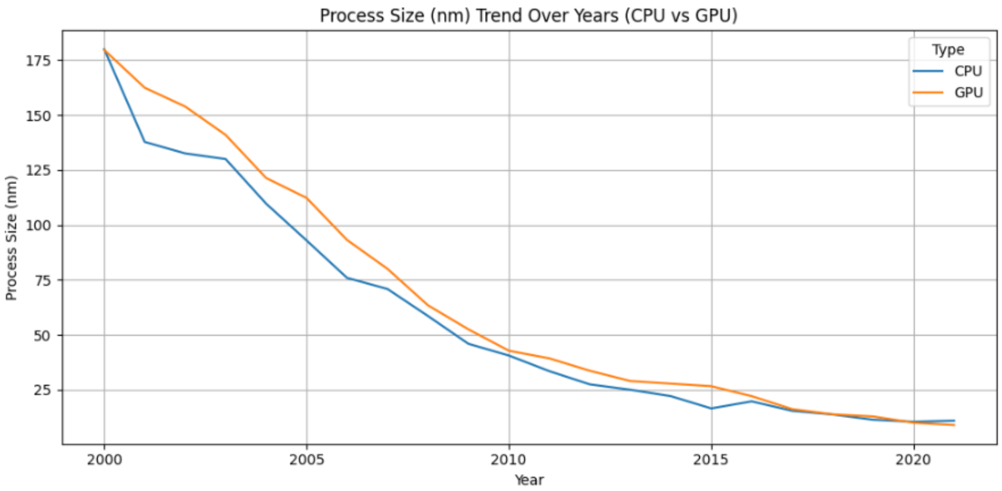
This added practical validation and showed that the models generalize well even on unseen samples from the same distribution.

Machine Learning Tools Used:

- **scikit-learn:**
  Used for PCA, KNN, classification models (Logistic Regression and Random Forest), and preprocessing (scaling, encoding).
- **matplotlib & seaborn:**
  Used for plotting trends, comparisons, PCA visualizations, and distributions.
- **Google Colab:**
  Served as the execution environment for all code, allowing for visualization and testing in real-time.

Simulation results:

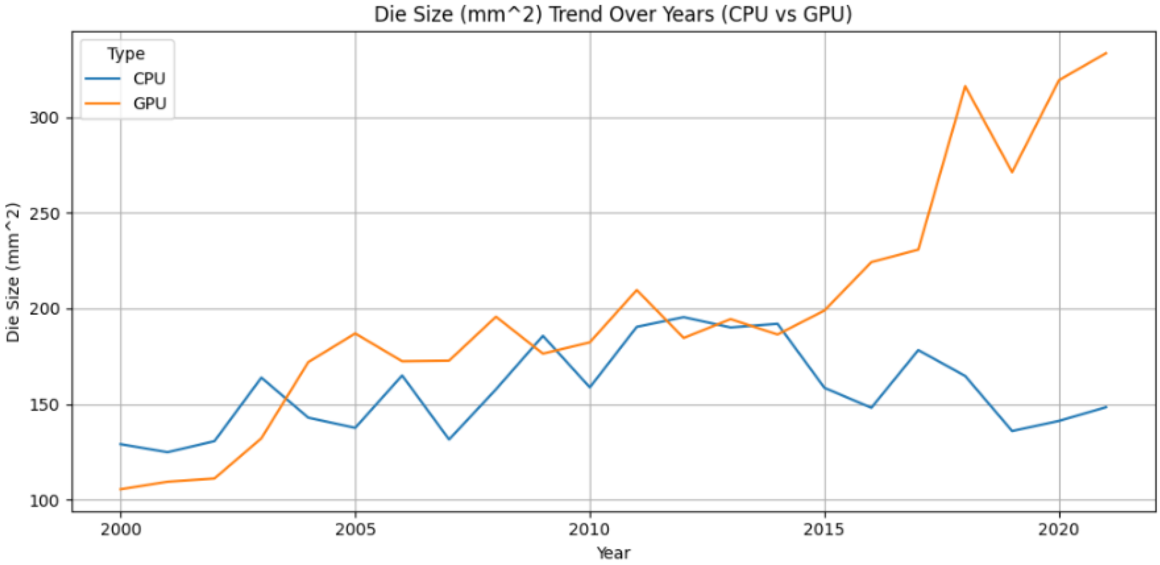
1. **Process Size (nm) Trend Over Years (CPU vs GPU)**



Insights:

- Both CPUs and GPUs have followed a steady downward trend in process size. The process size has shrunk from around 180 nm in 2000 to approximately 7 nm by 2021.
- CPUs appear to have adopted smaller nodes slightly earlier than GPUs in the 2012–2016 range
- The curves for both converge in recent years, suggesting that cutting-edge fabrication technologies (like 7nm and 5nm) are now uniformly adopted by both CPU and GPU manufacturers.

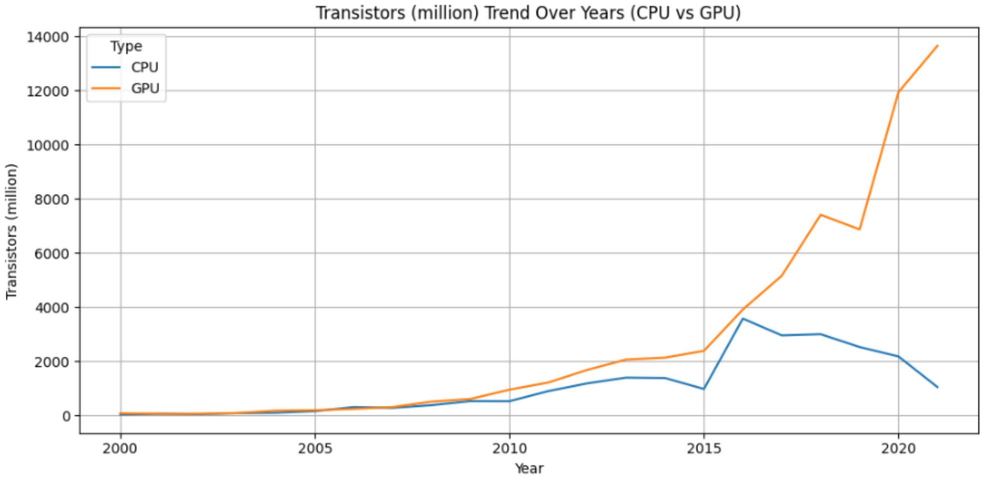
2. Die Size (mm^2) Trend Over Years (CPU vs GPU)



Insights:

- In the early 2000s, both CPU and GPU die sizes were relatively similar (≈100–150 mm^2). From around 2010 onward, GPU die sizes began to increase significantly compared to CPUs.
- This reflects the increasing complexity of GPU architectures, which are designed with thousands of parallel cores for graphics and compute workloads.

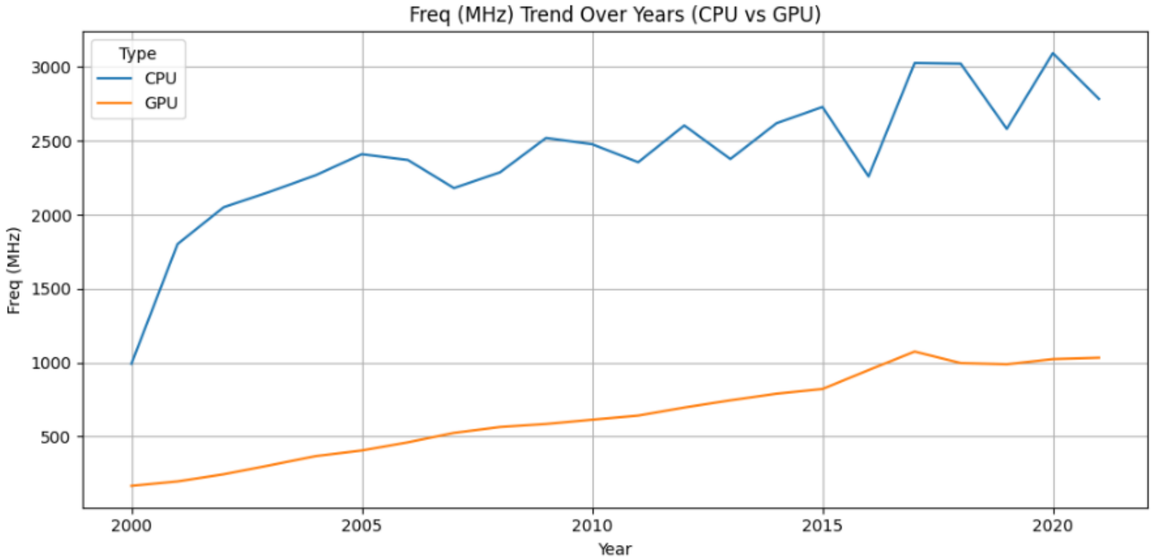
3. Transistor Count (Millions) Trend Over Years (CPU vs GPU)



Insights:

- In the early 2000s, both CPUs and GPUs had transistor counts under 100 million.
- Post-2010, both chip types began increasing in transistor count, but GPUs began outpacing CPUs significantly after 2015.
- This rapid growth in GPUs correlates with their rising role in high-performance computing, AI acceleration.

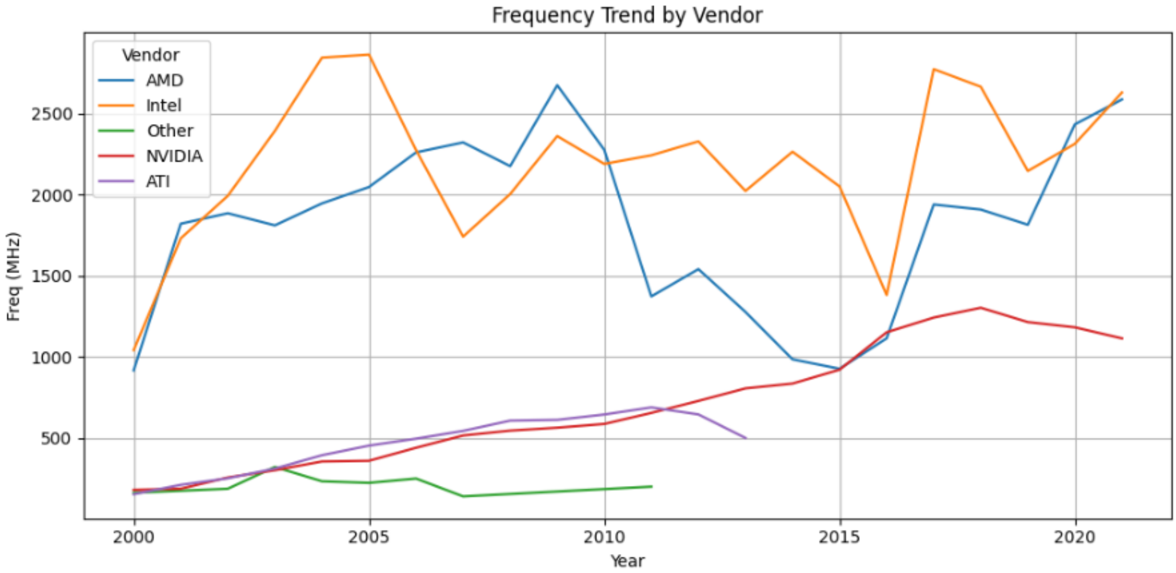
4. Frequency (MHz) Trend Over Years (CPU vs GPU)



Insights:

- CPUs have significantly higher frequencies than GPUs throughout the timeline.
- GPU frequencies, on the other hand, have remained much lower (typically below 1100 MHz), but have gradually increased over time.
- This contrast reflects a design trade-off: GPUs prioritize massive parallelism (many slower cores), while CPUs rely on fewer but faster cores.

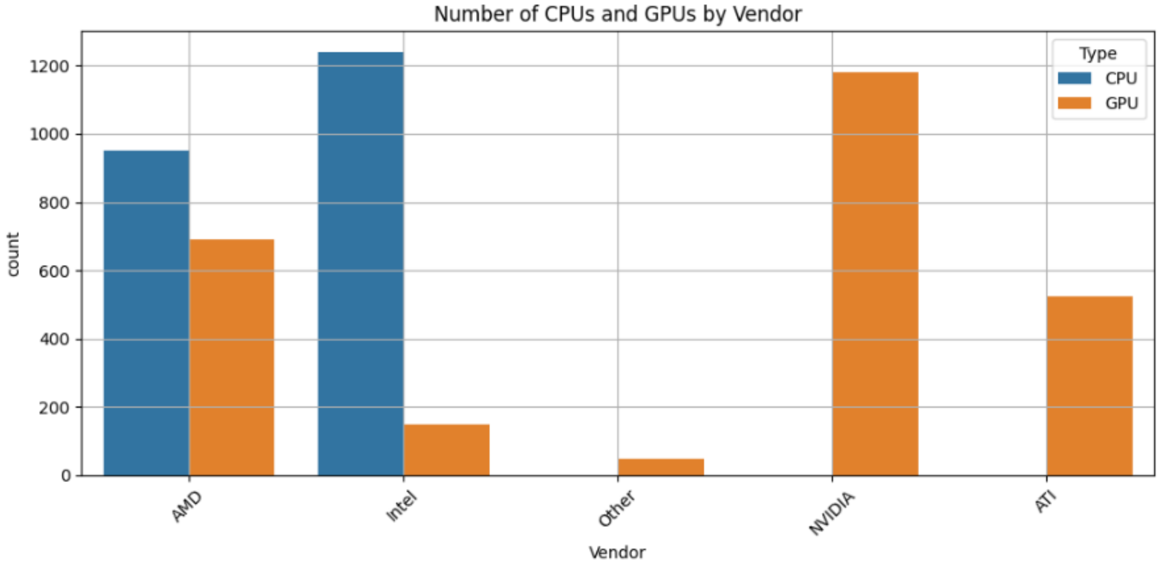
5. Frequency Trend by Vendor



Insights:

- Intel leads in average clock frequency. AMD shows a similar trend but with more fluctuation and generally lower average frequencies than Intel.
- NVIDIA (primarily GPU vendor) shows steady growth in frequency, especially post-2010, stabilizing around 1100–1300 MHz.
- Intel and AMD (CPU-dominant) aim for higher clock rates
- NVIDIA and ATI (GPU-dominant) prioritize parallel architecture with modest but consistent frequency gains

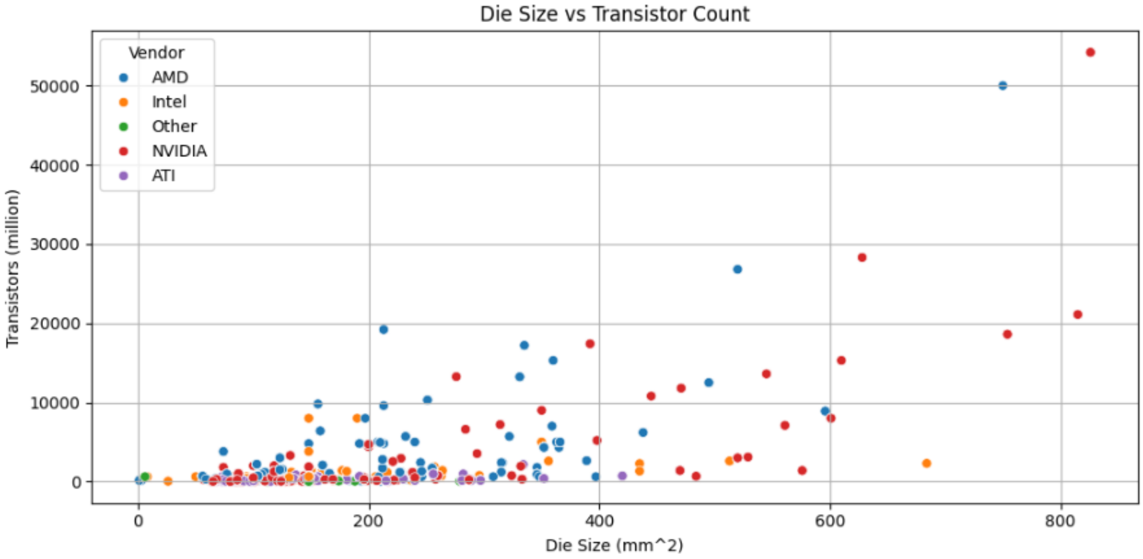
6. Number of CPUs and GPUs by Vendor



Insights:

- Intel → CPU dominant
- NVIDIA → GPU exclusive
- AMD → Balanced CPU + GPU portfolio

7. Die Size vs Transistor count

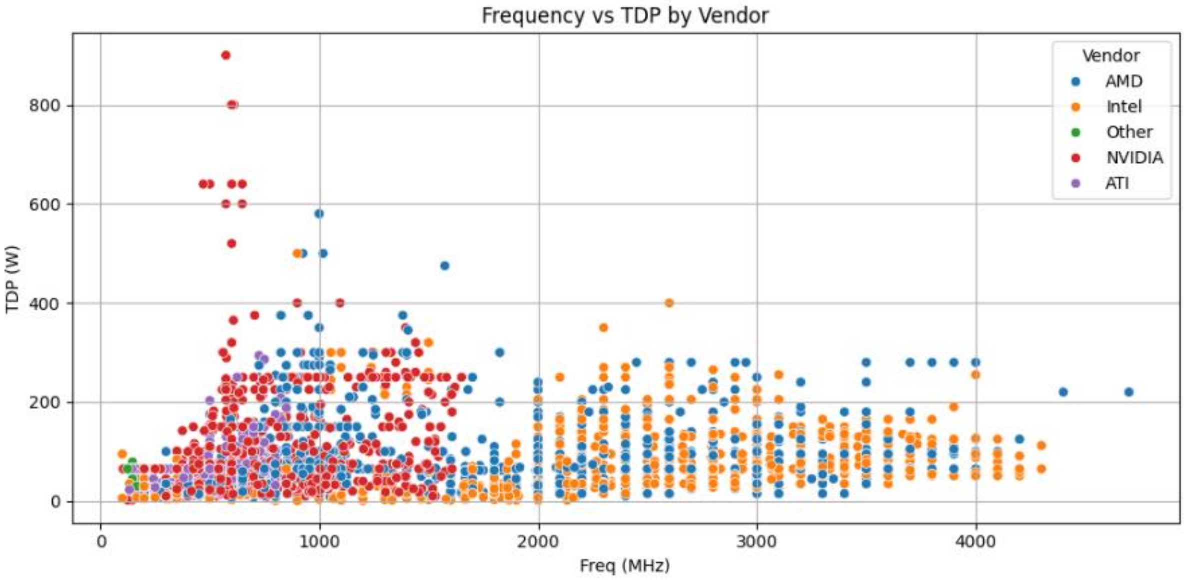


Insights:

NVIDIA prioritizes maximizing transistor count via large dies.

Intel optimizes for compactness and power efficiency.

8. Frequency vs TDP by Vendor

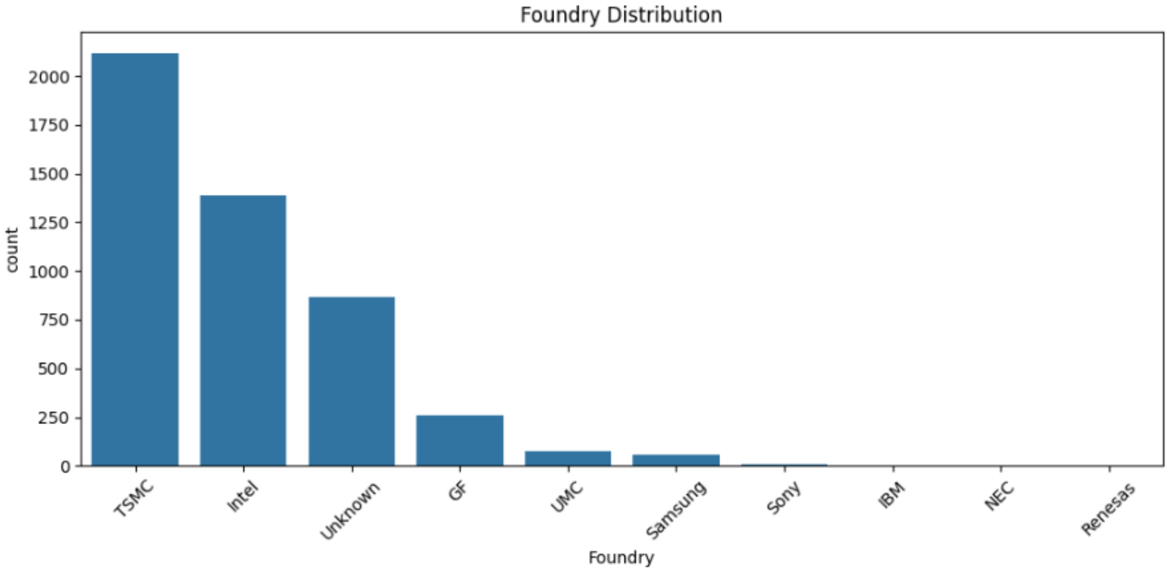


Insights:

CPUs (especially Intel/AMD) focus on high frequency with efficient power usage.

GPUs (especially NVIDIA) emphasize raw processing power, accepting higher power budgets to support massive parallelism.

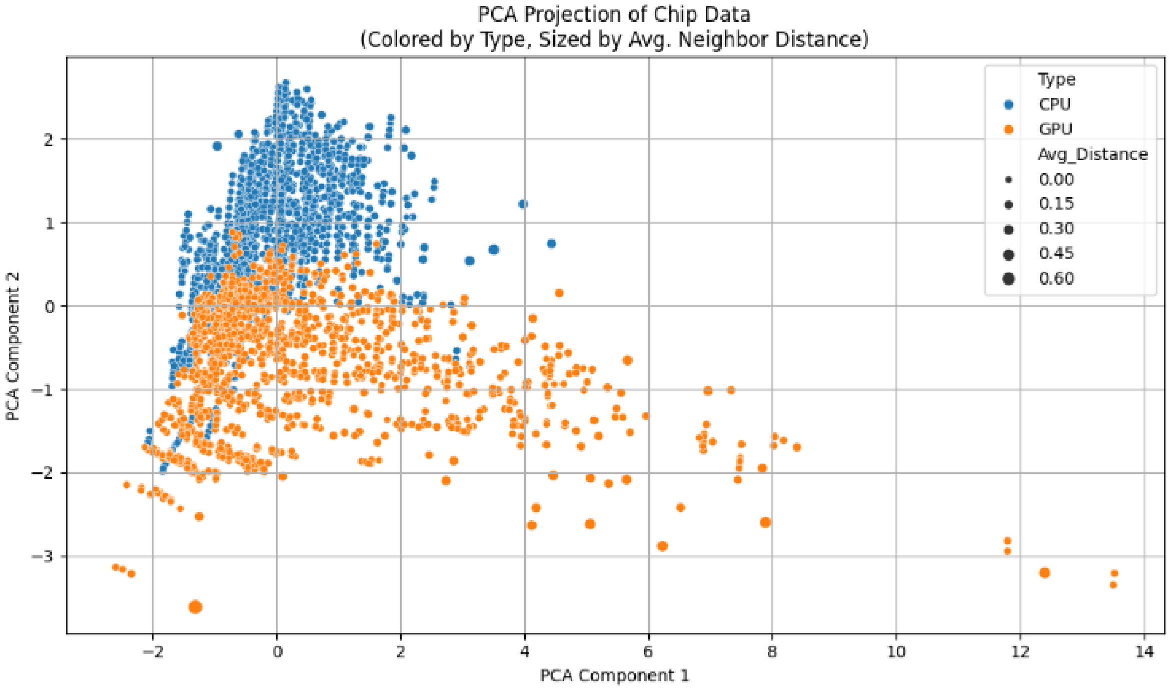
9. Foundry distribution



Insights:

TSMC is the clear winner to manufacture the chips

10. PCA Projection of Chip Data

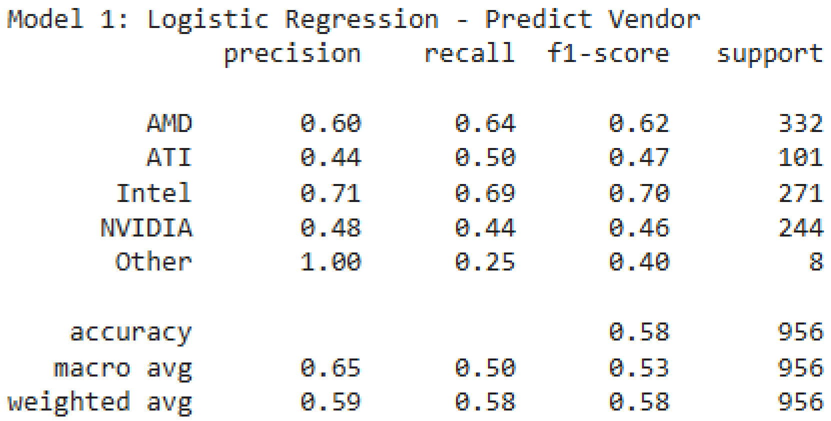


Insights:

- CPUs and GPUs form visibly separate clusters in PCA space, demonstrating that their design parameters differ enough to be distinguished even without labels.
- CPUs are tightly packed around a dense region in the top-left quadrant, indicating similar designs across vendors. GPUs are more spread out and heterogeneous.
- Larger points (bigger sizes) suggest unique or edge-case chips that differ from their nearest neighbors — possibly flagship or legacy architectures.

Models developed in this project to predict the vendor of a chip and if a given chip is CPU/GPU.

Model 1: Predict the vendor.

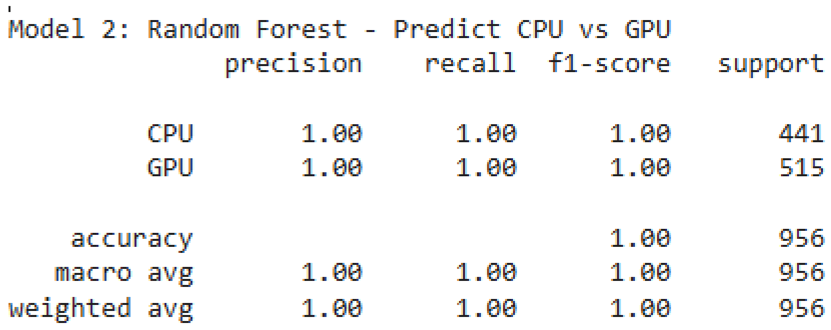


```
Model 1: Logistic Regression - Predict Vendor
              precision    recall  f1-score   support

         AMD       0.60      0.64      0.62       332
         ATI       0.44      0.50      0.47       101
       Intel       0.71      0.69      0.70       271
      NVIDIA       0.48      0.44      0.46       244
       Other       1.00      0.25      0.40         8

    accuracy                           0.58       956
   macro avg       0.65      0.50      0.53       956
weighted avg       0.59      0.58      0.58       956
```

Model 2: Predict whether given chip is CPU or GPU



```
Model 2: Random Forest - Predict CPU vs GPU
              precision    recall  f1-score   support

         CPU       1.00      1.00      1.00       441
         GPU       1.00      1.00      1.00       515

    accuracy                           1.00       956
   macro avg       1.00      1.00      1.00       956
weighted avg       1.00      1.00      1.00       956
```

Test samples:

Test Sample #1

Input: {'Process Size (nm)': 65.0, 'TDP (W)': 45.0, 'Die Size (mm^2)': 77.0, 'Transistors (million)': 122.0, 'Freq (MHz)': 2200.0}

Predicted Vendor: NVIDIA

Predicted Type : CPU

Test Sample #2

Input: {'Process Size (nm)': 14.0, 'TDP (W)': 35.0, 'Die Size (mm^2)': 192.0, 'Transistors (million)': 4800.0, 'Freq (MHz)': 3200.0}

Predicted Vendor: NVIDIA

Predicted Type : CPU

Inferences/Insights Obtained from this Study

- **Transistor counts have continued to grow**, consistent with Moore's Law, but frequency has plateaued — indicating the breakdown of Dennard scaling.
- **TDP (power consumption)** has increased significantly for GPUs, showing their shift toward high-performance parallel processing.
- **Vendors follow different design philosophies:**
  Intel tends to optimize frequency and power, while AMD and NVIDIA often push transistor counts and die sizes.
- **PCA visualizations clearly showed that CPUs and GPUs are well-separated in feature space**, confirming that chip characteristics naturally distinguish them.
- **Both supervised models performed well**, demonstrating that hardware characteristics can be used to reliably to predict both vendor and chip type.