

# SECTION 1

## ANS TO QUESTION 1

### TENSORFLOW

TensorFlow is an open-source machine learning (ML) and deep learning (DL) frameworks. Developed by Google, it is used AI research and industry applications. Tendorflow was released in 2015.

### PYTORCH

PyTorch is an open source machine learning library, released by Facebook's AI Research lab in 2016. It can be used across a range of tasks, but is particularly focused on training and inference of deep learning tasks, like computer vision and natural language processing.

### COMPARING AND CONSTRASTING

#### **TENSORFLOW**

Better for production (scalability, deployment)  
Stronger mobile support (**TF lite**)  
Better TPU intergration (Google Cloud)  
Harder to debug(TF 1.x)  
Less intuitive API (before TF 2.x)

#### **PYTORCH**

Preferred for research (easier prototyping)  
More Pythonic & intuitive  
Dynamic graphs (better for RNNs, GANs)  
Historically weaker in production (improving with TorchScript)  
Less optimized for mobile (compared to TF Lite)

Use Case	Recommended Framework
Research & Academia	PyTorch (dominant in papers)
Production & Large-Scale Deployment	TensorFlow (better tooling)
Beginner Learning DL	PyTorch (easier to understand)
Google Cloud/TPUs	TensorFlow
Cutting-Edge Models (NLP, GANs)	PyTorch (Hugging Face, OpenAI)

- PyTorch is for research and is growing in industry.
- TensorFlow is better for production but has lost some research dominance.
- PyTorch is becoming more production-ready, while TensorFlow is simplifying its API.

Both are good and are used for different purpose but i prefer TensorFlow because it is best for production, scalability and deployment

## ANS TO QUESTION 2

Jupyter Notebook is an interactive, web-based computing environment that is used in AI/ML development due to its flexibility, visualization capabilities, and ease of use.

Two uses in AI development is

- Prototyping and Experimentation:  
Execute AI/ML models step by step and see results immediately  
No need to re-run entire script for small changes
- Data exploration and visualization:  
For plots, tables and image render inline

With the use of pandas and matplotlib data can be cleaned, analysed and visualized before training.

## ANS TO QUESTION 3

SpaCy is a modern, industrial-strength NLP library that significantly outperforms basic Python string operations (e.g., `split()`, `replace()`, `regex`) in efficiency, accuracy, and functionality.

The following why SpaCy is favourable to Python string operation

1. Tokenization: spaCy uses Linguistic rules and machine learning while python used splitting eg `str.split()`
2. Named Entity Recognition(NER): spaCy uses pre-trained ML models while python uses regex patterns(manual rules)
3. Sentence Segmentation: spaCy uses statistical model and rules while python uses splits on period.
4. Efficiency and Scalability: SpaCy is built for large scale text processing
5. For tasks like data extraction and chatbots, spaCy saves lots of hours when compared to python string operation.

## SECTION 2

### TARGET APPLICATION

Scikit-learn is good for ML task like linear regression and k-means clustering. It uses small files like CSV

TensorFlow is good for deep learning and neural networks. For chatbots and image recognition. It uses large and unstructured data

Scikit-learn is great for math problems

TensorFlow is needed for advanced AI tasks

## EASE OF USE FOR BEGINNERS

Scikit-learn requires less code thus making it easier to learn and does not require GPU

TensorFlow requires a GPU as CPU is slow when used for deep learning. It requires more code, data processing and training loop

Scikit-learn is easier for beginners

Start with Scikit-learn if you're new to ML.

Learn TensorFlow later if you want to work with AI (e.g., image/text processing).

## COMMUNITY SUPPORT

Scikit-learn has a lot of tutorials eg Kaggle and youtube, fewer updates that makes it more stable thus making it more beginner friendly.

TensorFlow has a large community backed by Google and its fast changing in terms of update.

## Summary Table

Feature	Scikit-learn	TensorFlow
Best for	Classical ML	Deep Learning
Ease of Use	Very easy	Moderate to hard
GPU Needed?	No	Yes (for big models)
Code Length	Short (3-5 lines)	Longer (10+ lines)
Community	Stable & simple	Large & fast-moving