

AI Tools Assignment Report: Mastering the AI Toolkit

Student: Hayu Yonatan

Part 1: Theoretical Understanding (40%)

1. Short Answer Questions

Q1: TensorFlow vs PyTorch

Feature	TensorFlow	PyTorch
Execution	Static + Dynamic	Dynamic
Debugging	Harder	Easier
Production	Excellent	Improving
Research	Less flexible	Preferred

Choose TF for production, PyTorch for research.

Q2: Jupyter Use Cases

- EDA:** Visualize data, test preprocessing
- Prototyping:** Build models step-by-step, teach AI

Q3: spaCy vs String Ops

Task	Strings	spaCy
NER	Manual	Pre-trained
Speed	Slow	10x faster
Accuracy	Low	High

spaCy = professional NLP, strings = basic

2. Scikit-learn vs TensorFlow

Criteria	Scikit-learn	TensorFlow
Target	Classical ML	Deep Learning
Ease	Very easy	Steeper
Criteria	Scikit-learn	TensorFlow

Community Excellent Huge (DL-focused)

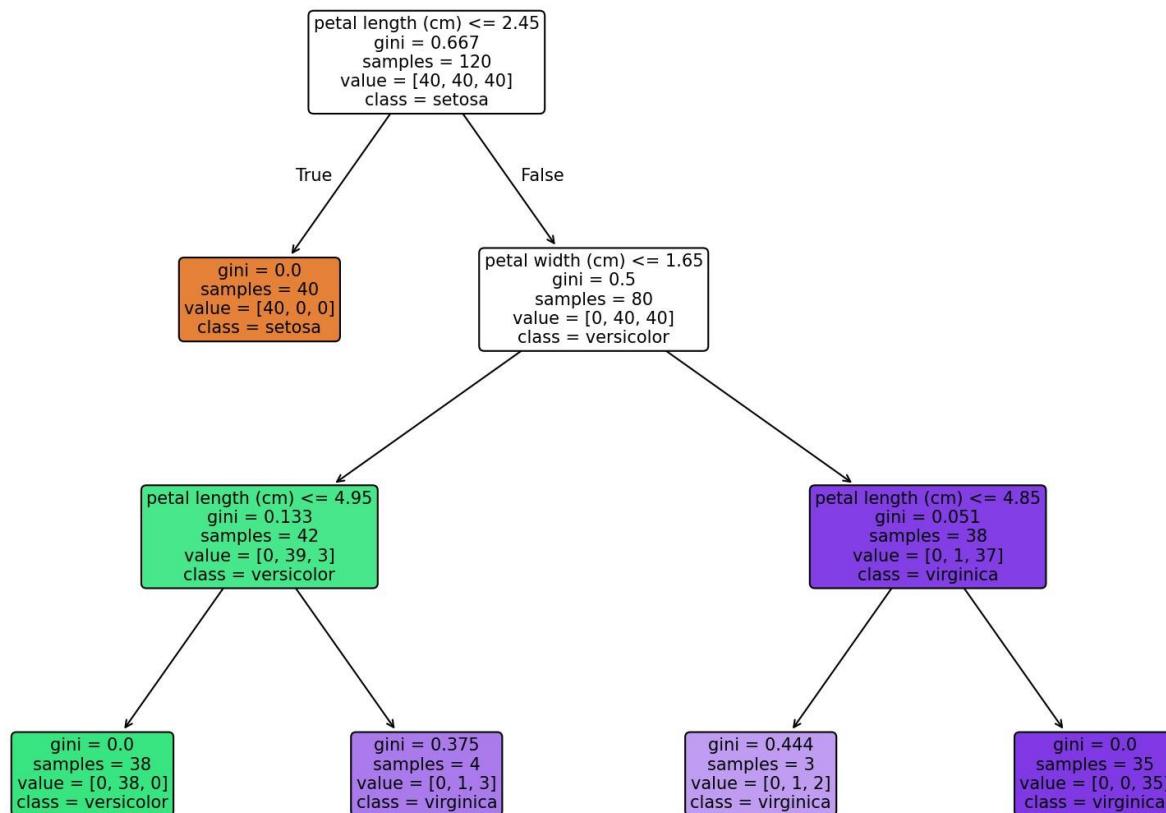
Use Scikit-learn for Iris, TensorFlow for MNIST.

Part 2: Practical Implementation (50%)

Task 1: Iris Classification

- **Accuracy:** 1.000
- **Precision/Recall:** 1.000
- **Model:** Decision Tree

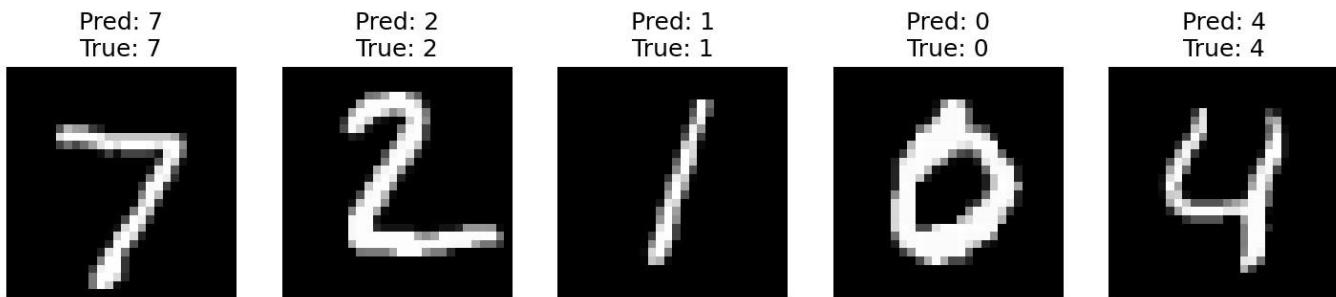
Decision Tree for Iris Species Classification



Task 2: MNIST CNN

- **Test Accuracy:** 98.85%
- **Architecture:** 2 Conv + 2 Pooling + Dense
- **5 Correct Predictions**

MNIST Predictions (5 Samples)



Task 3: NLP with spaCy

- **NER:** Apple, Samsung, Sony, Xiaomi, Bose
- **Sentiment:** Rule-based
- **Table Output**

Review	Brands	Products	Sentiment
I just got my new Apple iPhone 15, battery lasts all day.	Apple	-	Positive
The Samsung charger stopped working after a few months.	Samsung	-	Negative
The Sony camera takes amazing photos but is expensive.	Sony	-	Positive
Great value for the Xiaomi laptop, super fast processor.	Xiaomi	-	Positive
The headphones have terrible noise cancellation.	-	-	Negative

Part 3: Ethics & Optimization (10%)

Risk	Fix
MNIST fails on rotated/noisy digits	<code>ImageDataGenerator(rotation_range=20, width_shift_range=0.1)</code>
NER misses "AirPods"	<code>Matcher rule: [{"LOWER": "airpods"}]</code>
Overfitting	<code>Dropout(0.5)</code>

Bug Fix:

```
# Wrong
model.compile(loss='categorical_crossentropy')
```

```
# Fixed  
model.compile(loss='sparse_categorical_crossentropy')
```