WEEK 3 ASSIGNMENT

1. Short Answer Questions

- **Q1: Explain the primary differences between TensorFlow and PyTorch. When would you choose one over the other?**

  TensorFlow is a framework for large-scale deep learning with strong deployment support, while PyTorch is more flexible and Pythonic, great for research and experimentation.

  I would choose TensorFlow for production models and PyTorch for fast prototyping and academic work.

- **Q2: Describe two use cases for Jupyter Notebooks in AI development.**

  (i) Used for interactive coding and visualization of data and models
  (ii) Used for experiment tracking, where codes output and notes are kept together.

- **Q3: How does spaCy enhance NLP tasks compared to basic Python string operations?**

spaCy provides pre-trained NLP models and efficient tokenization, tagging and parsing, unlike basic string operations that handle text only at the character or word level.

2. Comparative Analysis

- **Compare Scikit-learn and TensorFlow in terms of:**
  - **Target applications (e.g., classical ML vs. deep learning).**
  - **Ease of use for beginners.**
  - **Community support.**

Feature	**Scikit-learn**	**TensorFlow**
Target applications	Classical ML (regression, Classification, Clustering)	Deep learning (neural networks, large-scale AI models)
Ease of use	Easier for beginners; simple API	Steeper learning curve; more complex setup
Community support	Large for ML tasks	Huge for AI/ deep learning, supported by google

ETHICS & OPTIMIZATION

Potential biases:

- If trained on only certain handwriting styles, the model may fail on unfamiliar styles.
- Digit misclassification can affect applications like digit-based forms, banking, or accessibility tools.

Mitigation:

- Use **TensorFlow Fairness Indicators** to analyse prediction fairness across subgroups (e.g, handwriting styles)
- Use **data augmentation** to diversify training data (rotations, scaling, slant variations).

Amazon Reviews (NLP) example:

Potential biases:

- If reviews contain biased language, sentiment analysis may unfairly favor/penalize certain brands or products.
- Rule-based sentiment might misinterpret sarcasm or context.

Mitigation:

- Use spaCy rule-based systems to detect biased words or entities.
- Combine with lexicons or pre-trained models for sentiment correction.

Troubleshooting Challenge

- **Buggy Code**: A provided TensorFlow script has errors (e.g., dimension mismatches, incorrect loss functions). Debug and fix the code.

Dimension mismatch (The input data wasn't reshaped correctly for the CNN)
to f*ix*, reshaped the images to include the channel dimension using:

```
x_train = x_train.reshape(-1, 28, 28, 1)
x_test = x_test.reshape(-1, 28, 28, 1)
```

Incorrect loss function (The model used a regression loss instead of a classification one)
to f*ix, c*hanged the loss to sparse_categorical_crossentropy for multi-class classification:

```
model.compile(optimizer='adam',
loss='sparse_categorical_crossentropy', metrics=['accuracy'])
```