**Image Classification on CIFAR-10 Dataset using Convolutional Neural Networks**

**by**

**Shruti Andraskar (MST03-0067)**

**Submitted to Scifor Technologies**



**UNDER GUIDANCE OF**

**Urooj Khan**

## TABLE OF CONTENTS

**Image Classification on CIFAR-10 Dataset using Convolutional Neural**

**Networks.......................................................................................................... 1**

TABLE OF CONTENTS...........................................................................2

Abstract........................................................................................................ 3

Introduction...................................................................................................4

Technology Used..........................................................................................5

1. Python................................................................................................. 5
2. TensorFlow and Keras.........................................................................5
3. NumPy.................................................................................................5
4. Matplotlib............................................................................................. 5

Dataset Information...................................................................................... 6

Methodology.................................................................................................7

1. Data Preparation................................................................................. 7
2. Model Development............................................................................ 7
3. Model Compilation...............................................................................7
4. Model Training.....................................................................................7
5. Model Evaluation.................................................................................7

Code Snippet................................................................................................8

Results and Discussion.............................................................................. 12

Conclusion..................................................................................................13

References………………………………………………………………………14

## Abstract

This report outlines the development and evaluation of a Convolutional Neural Network (CNN) model for image classification using the CIFAR-10 dataset. The model is trained to classify images into one of 10 classes, such as airplanes, automobiles, birds, cats, and others. The report includes a comprehensive explanation of the dataset, the architecture of the CNN, the training process, and the results obtained from the evaluation. The performance of the model is analyzed through various metrics, and a specific example of image classification is demonstrated.

## Introduction

Image classification is a fundamental task in computer vision that involves assigning a label to an image based on its content. Convolutional Neural Networks (CNNs) have proven to be highly effective for this task due to their ability to learn hierarchical representations of visual data. The CIFAR-10 dataset, a widely-used benchmark in image classification, consists of 60,000 32x32 color images in 10 different classes. This project aims to build and evaluate a CNN model to classify images from the CIFAR-10 dataset accurately

## Technology Used

1. Python

Python is a versatile programming language widely used in machine learning and deep learning. Its rich ecosystem of libraries and frameworks makes it ideal for developing image classification models.

1. TensorFlow and Keras

TensorFlow is an open-source deep learning framework developed by Google. Keras, a high-level API for TensorFlow, allows for easy and fast prototyping of neural networks. Together, they provide the necessary tools to build, train, and evaluate CNN models.

1. NumPy

NumPy is a fundamental library for numerical computations in Python. It provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays.

1. Matplotlib

Matplotlib is a plotting library for Python that enables the visualisation of data through charts and graphs. It is essential for visualising training progress and model performance.

## Dataset Information

The CIFAR-10 dataset consists of 60,000 32x32 colour images in 10 classes, with 6,000 images per class. The dataset is divided into 50,000 training images and 10,000 test images. Each image is labelled with one of the following classes:

* Airplane
* Automobile
* Bird
* Cat
* Deer
* Dog
* Frog
* Horse
* Ship
* Truck

The images in the CIFAR-10 dataset are low resolution, making it a challenging task for image classification models.

## Methodology

### **1. Loading and Preprocessing the Data**

The CIFAR-10 dataset is loaded from the datasets module in TensorFlow. The pixel values of the images are normalized to be between 0 and 1 to facilitate the training process.

### **2. Building the CNN Model**

A sequential CNN model is constructed with the following layers:

* **Convolutional Layers**: Extract features from the input images.
* **Batch Normalization Layers**: Normalize the activations to speed up training.
* **Max Pooling Layers**: Downsample the feature maps to reduce dimensionality.
* **Dropout Layers**: Prevent overfitting by randomly setting a fraction of input units to 0.
* **Fully Connected Layers**: Perform the final classification based on the extracted features.

### **3. Training the Model**

The model is compiled using the Adam optimizer and the Sparse Categorical Crossentropy loss function. It is trained on the training dataset for 20 epochs, with the validation dataset used to monitor the model's performance.

### **4. Evaluating the Model**

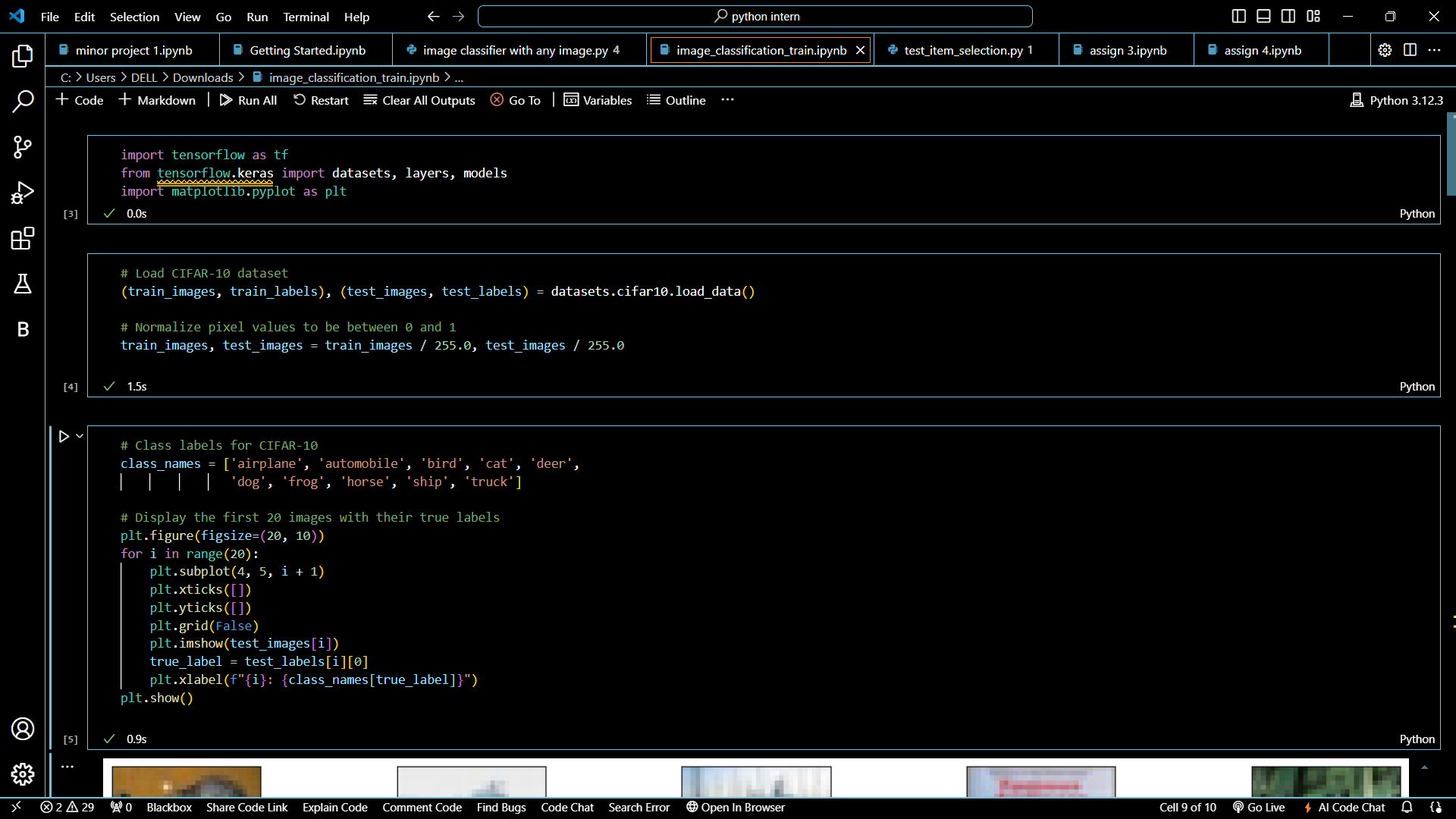
The model's performance is evaluated on the test dataset using accuracy and loss metrics. The training and validation accuracy and loss are plotted to visualize the model's learning progress.

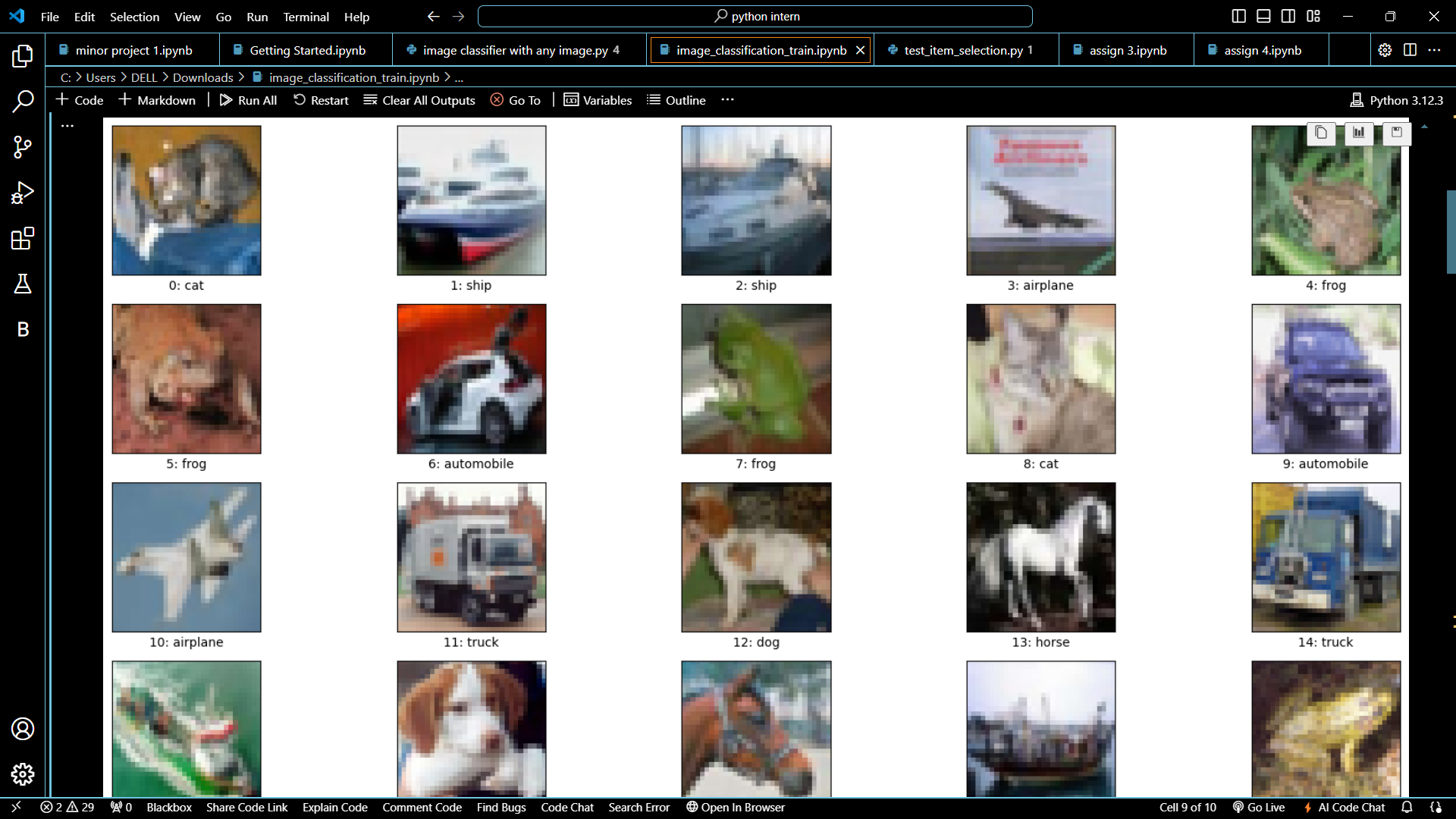
### **5. Making Predictions**

The trained model is used to predict the class of a specific image from the test dataset. The prediction is compared to the true label to demonstrate the model's classification capability.

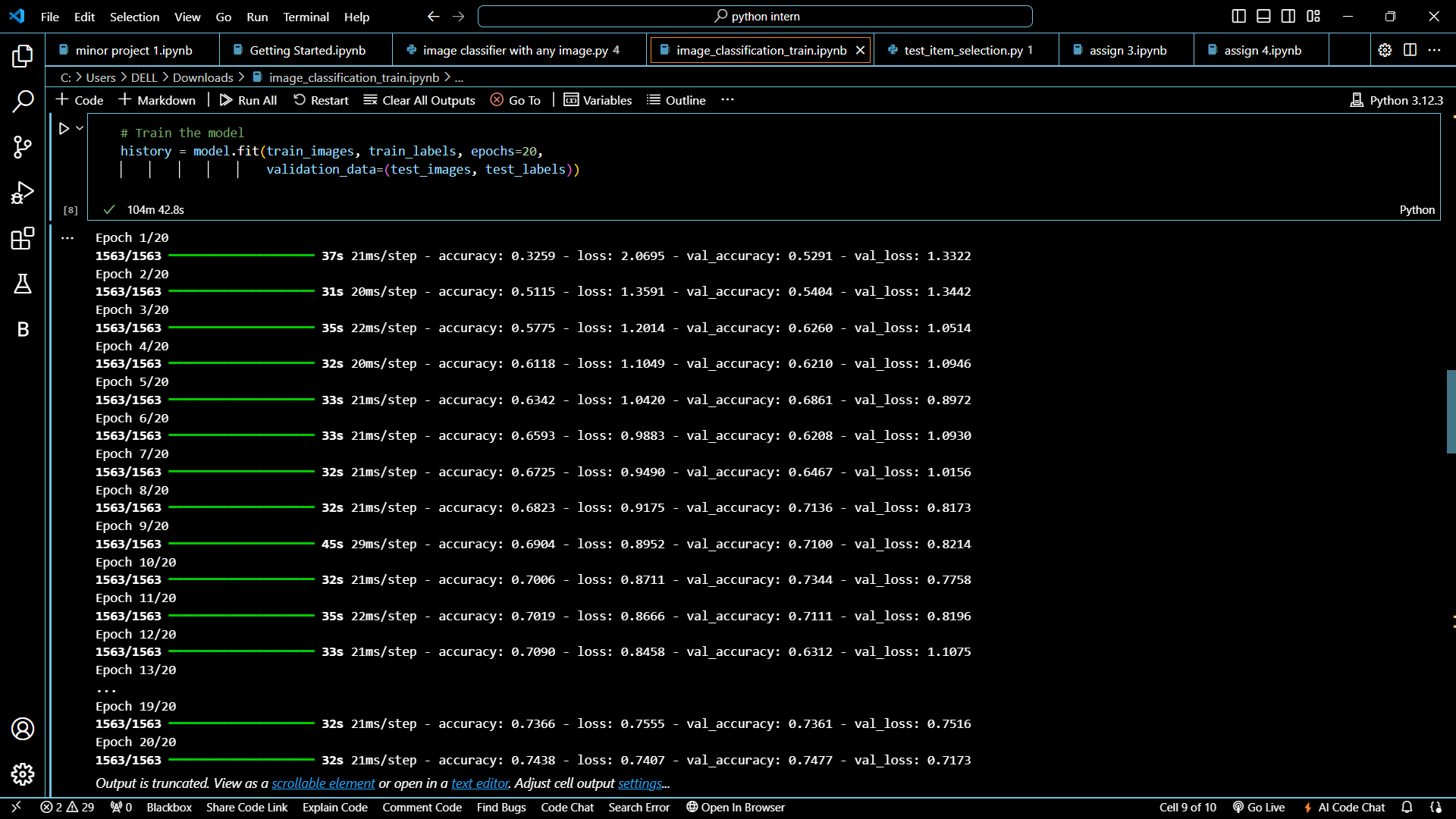
# Code Snippet

Below is the complete code used for this project:



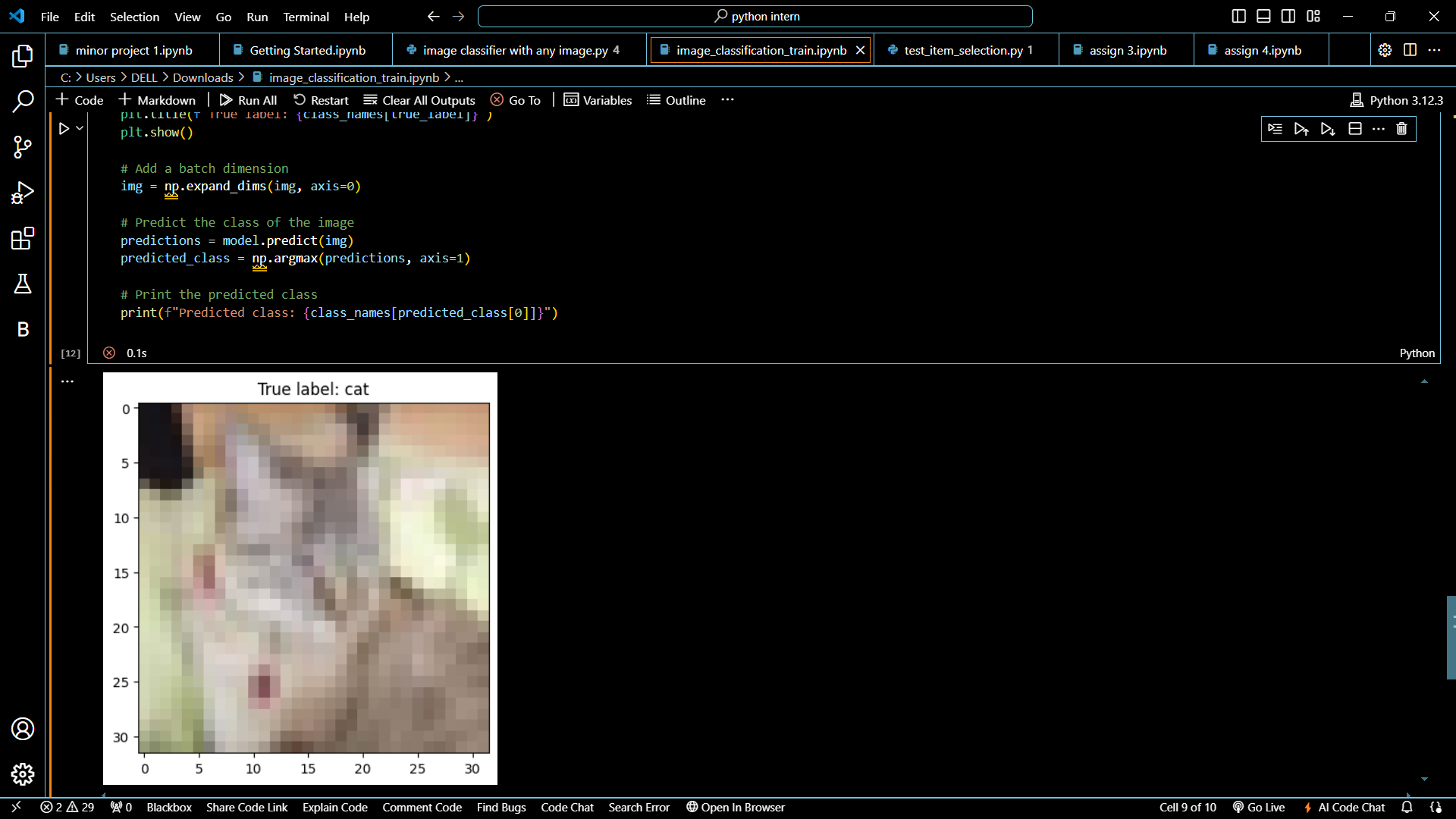












## Results and Discussion

### **Model Training and Evaluation**

The CNN model was trained for 20 epochs, with both training and validation datasets. The following are key observations from the training process:

* **Training Accuracy**: The training accuracy increased steadily, reaching approximately 78% by the 20th epoch.
* **Validation Accuracy**: The validation accuracy fluctuated but generally trended upward, reaching approximately 74%.
* **Training Loss**: The training loss decreased consistently, indicating that the model was learning effectively.
* **Validation Loss**: The validation loss also decreased, although there were occasional increases, suggesting some overfitting.

The final test accuracy of the model was 74.34%, demonstrating its ability to generalize to unseen data. The accuracy and loss plots provide a visual representation of the model's performance over the epochs.

### **Image Prediction**

A specific image from the test dataset was selected to demonstrate the model's prediction capability. The image was correctly classified by the model, aligning with the true label. This example highlights the model's effectiveness in image classification tasks

## Conclusion

This project successfully demonstrates the implementation of a Convolutional Neural Network for image classification using the CIFAR-10 dataset. The model achieved a reasonable accuracy, indicating its effectiveness in classifying images into one of 10 categories. However, there is potential for improvement through further hyperparameter tuning, data augmentation, and exploring different architectures. Overall, this project highlights the power of CNNs in solving complex image classification tasks and provides a foundation for future enhancements

## REFERENCES

1. [How](https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/) [to](https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/) [Develop](https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/) [a](https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/) [CNN](https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/) [From](https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/) [Scratch](https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/) [for](https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/) [CIFAR-10](https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/) [Photo](https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/) [Classification](https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/) [-](https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/)

[MachineLearningMastery.com](https://machinelearningmastery.com/how-to-develop-a-cnn-from-scratch-for-cifar-10-photo-classification/)

1. [CIFAR-10](https://www.geeksforgeeks.org/cifar-10-image-classification-in-tensorflow/) [Image](https://www.geeksforgeeks.org/cifar-10-image-classification-in-tensorflow/) [Classification](https://www.geeksforgeeks.org/cifar-10-image-classification-in-tensorflow/) [in](https://www.geeksforgeeks.org/cifar-10-image-classification-in-tensorflow/) [TensorFlow](https://www.geeksforgeeks.org/cifar-10-image-classification-in-tensorflow/) [-](https://www.geeksforgeeks.org/cifar-10-image-classification-in-tensorflow/) [GeeksforGeeks](https://www.geeksforgeeks.org/cifar-10-image-classification-in-tensorflow/)
2. [GitHub](https://github.com/hnm0284/Image-Classification-Using-CNN) [-](https://github.com/hnm0284/Image-Classification-Using-CNN) [hnm0284/Image-Classification-Using-CNN:](https://github.com/hnm0284/Image-Classification-Using-CNN) [A](https://github.com/hnm0284/Image-Classification-Using-CNN) [comprehensive deep](https://github.com/hnm0284/Image-Classification-Using-CNN) [dive](https://github.com/hnm0284/Image-Classification-Using-CNN) [into](https://github.com/hnm0284/Image-Classification-Using-CNN) [Convolutional](https://github.com/hnm0284/Image-Classification-Using-CNN) [Neural](https://github.com/hnm0284/Image-Classification-Using-CNN) [Networks](https://github.com/hnm0284/Image-Classification-Using-CNN) [(CNNs)](https://github.com/hnm0284/Image-Classification-Using-CNN) [for](https://github.com/hnm0284/Image-Classification-Using-CNN) [image](https://github.com/hnm0284/Image-Classification-Using-CNN) [classification, incorporating](https://github.com/hnm0284/Image-Classification-Using-CNN) [both](https://github.com/hnm0284/Image-Classification-Using-CNN) [custom](https://github.com/hnm0284/Image-Classification-Using-CNN) [and](https://github.com/hnm0284/Image-Classification-Using-CNN) [pretrained](https://github.com/hnm0284/Image-Classification-Using-CNN) [VGG16](https://github.com/hnm0284/Image-Classification-Using-CNN) [models](https://github.com/hnm0284/Image-Classification-Using-CNN) [on](https://github.com/hnm0284/Image-Classification-Using-CNN) [the](https://github.com/hnm0284/Image-Classification-Using-CNN) [CIFAR-10 dataset.](https://github.com/hnm0284/Image-Classification-Using-CNN)
3. [Image](https://www.analyticsvidhya.com/blog/2021/01/image-classification-using-convolutional-neural-networks-a-step-by-step-guide/) [Classification](https://www.analyticsvidhya.com/blog/2021/01/image-classification-using-convolutional-neural-networks-a-step-by-step-guide/) [Using](https://www.analyticsvidhya.com/blog/2021/01/image-classification-using-convolutional-neural-networks-a-step-by-step-guide/) [CNN](https://www.analyticsvidhya.com/blog/2021/01/image-classification-using-convolutional-neural-networks-a-step-by-step-guide/) [with](https://www.analyticsvidhya.com/blog/2021/01/image-classification-using-convolutional-neural-networks-a-step-by-step-guide/) [Keras](https://www.analyticsvidhya.com/blog/2021/01/image-classification-using-convolutional-neural-networks-a-step-by-step-guide/) [&](https://www.analyticsvidhya.com/blog/2021/01/image-classification-using-convolutional-neural-networks-a-step-by-step-guide/) [CIFAR-10](https://www.analyticsvidhya.com/blog/2021/01/image-classification-using-convolutional-neural-networks-a-step-by-step-guide/)

[(analyticsvidhya.com)](https://www.analyticsvidhya.com/blog/2021/01/image-classification-using-convolutional-neural-networks-a-step-by-step-guide/)

1. [Convolutional](https://www.tensorflow.org/tutorials/images/cnn) [Neural](https://www.tensorflow.org/tutorials/images/cnn) [Network](https://www.tensorflow.org/tutorials/images/cnn) [(CNN)](https://www.tensorflow.org/tutorials/images/cnn) [|](https://www.tensorflow.org/tutorials/images/cnn) [TensorFlow](https://www.tensorflow.org/tutorials/images/cnn) [Core](https://www.tensorflow.org/tutorials/images/cnn)
2. [Cifar-10](https://ieeexplore.ieee.org/document/8745428) [Classification](https://ieeexplore.ieee.org/document/8745428) [using](https://ieeexplore.ieee.org/document/8745428) [Deep](https://ieeexplore.ieee.org/document/8745428) [Convolutional](https://ieeexplore.ieee.org/document/8745428) [Neural](https://ieeexplore.ieee.org/document/8745428) [Network](https://ieeexplore.ieee.org/document/8745428) [|](https://ieeexplore.ieee.org/document/8745428) [IEEE](https://ieeexplore.ieee.org/document/8745428)

[Conference](https://ieeexplore.ieee.org/document/8745428) [Publication](https://ieeexplore.ieee.org/document/8745428) [|](https://ieeexplore.ieee.org/document/8745428) [IEEE](https://ieeexplore.ieee.org/document/8745428) [Xplore](https://ieeexplore.ieee.org/document/8745428)

1. [CIFAR-10](https://www.kaggle.com/code/farzadnekouei/cifar-10-image-classification-with-cnn) [Image](https://www.kaggle.com/code/farzadnekouei/cifar-10-image-classification-with-cnn) [Classification](https://www.kaggle.com/code/farzadnekouei/cifar-10-image-classification-with-cnn) [with](https://www.kaggle.com/code/farzadnekouei/cifar-10-image-classification-with-cnn) [CNN](https://www.kaggle.com/code/farzadnekouei/cifar-10-image-classification-with-cnn) [(kaggle.com)](https://www.kaggle.com/code/farzadnekouei/cifar-10-image-classification-with-cnn)