



ICG-PIXEL PERSONA

WINTER CAMP

GENDER PREDICTION MODEL

BY -ANJALI KUMARI

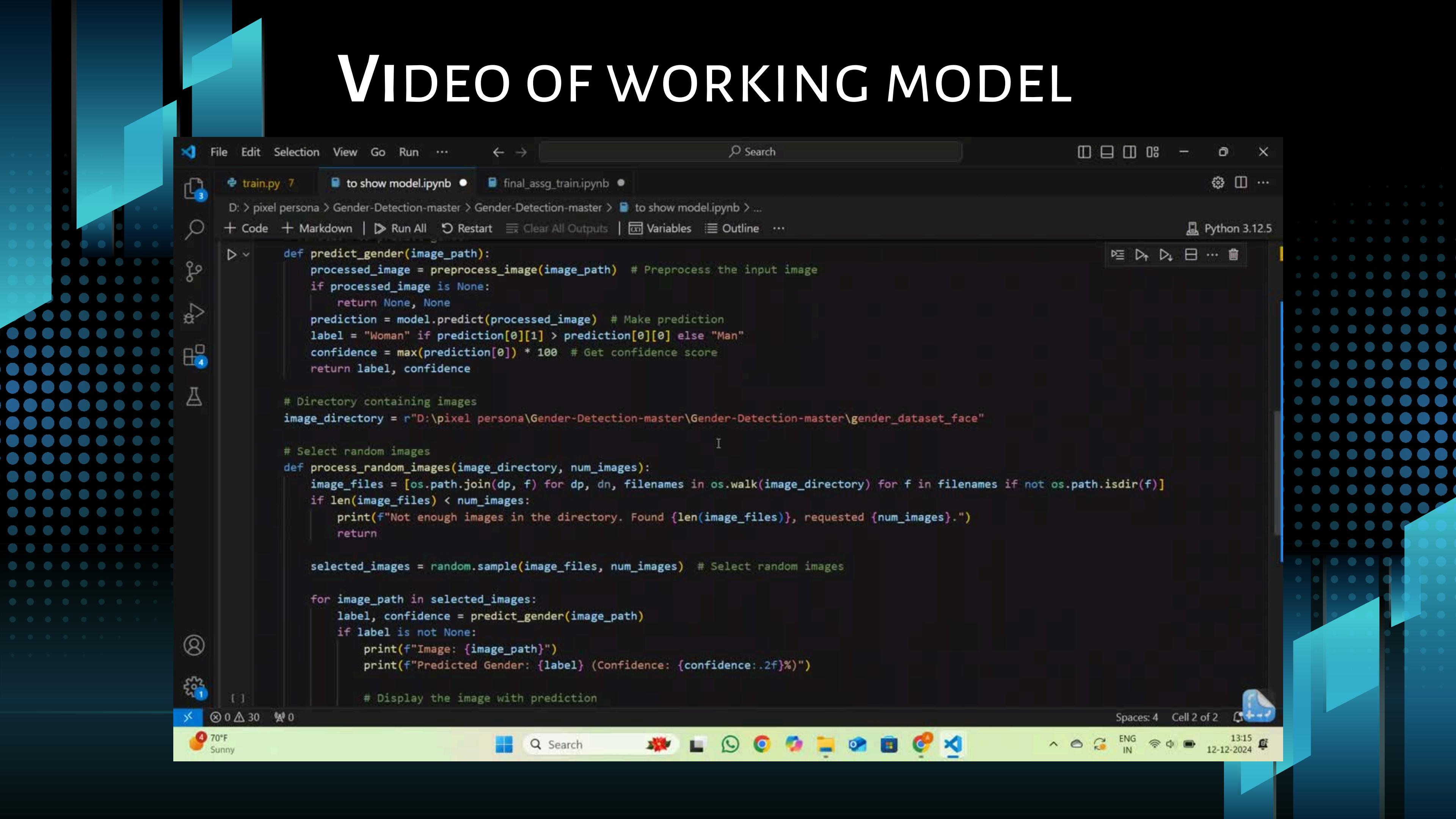
ROLL-241080053

INTRODUCTION

Gender prediction models have emerged as a fascinating application of machine learning and artificial intelligence.

The gender prediction model first trains on labelled dataset(male /female) and then by analyzing patterns and extracting relevant features, it predicts the gender .

VIDEO OF WORKING MODEL



The image shows a screenshot of a Windows desktop environment. In the foreground, a code editor window is open, displaying Python code for a gender detection model. The code includes functions for predicting gender from an image path, processing random images from a directory, and displaying the results. The code editor interface includes a toolbar, a file list, and a status bar at the bottom. In the background, the Windows taskbar is visible, showing various pinned icons and the system tray. The system tray includes a weather widget showing '70°F Sunny', a search bar, and several system icons.

```
File Edit Selection View Go Run ... ⏪ ⏪ Search
train.py 7 to show model.ipynb final_assg_train.ipynb
D: > pixel persona > Gender-Detection-master > Gender-Detection-master > to show model.ipynb > ...
+ Code + Markdown | Run All ⏪ Restart ⏪ Clear All Outputs | Variables Outline ...
Python 3.12.5

def predict_gender(image_path):
    processed_image = preprocess_image(image_path) # Preprocess the input image
    if processed_image is None:
        return None, None
    prediction = model.predict(processed_image) # Make prediction
    label = "Woman" if prediction[0][1] > prediction[0][0] else "Man"
    confidence = max(prediction[0]) * 100 # Get confidence score
    return label, confidence

# Directory containing images
image_directory = r"D:\pixel persona\Gender-Detection-master\Gender-Detection-master\gender_dataset_face"

# Select random images
def process_random_images(image_directory, num_images):
    image_files = [os.path.join(dp, f) for dp, dn, filenames in os.walk(image_directory) for f in filenames if not os.path.isdir(f)]
    if len(image_files) < num_images:
        print(f"Not enough images in the directory. Found {len(image_files)}, requested {num_images}.")
        return

    selected_images = random.sample(image_files, num_images) # Select random images

    for image_path in selected_images:
        label, confidence = predict_gender(image_path)
        if label is not None:
            print(f"Image: {image_path}")
            print(f"Predicted Gender: {label} (Confidence: {confidence:.2f}%)")

    # Display the image with prediction
```

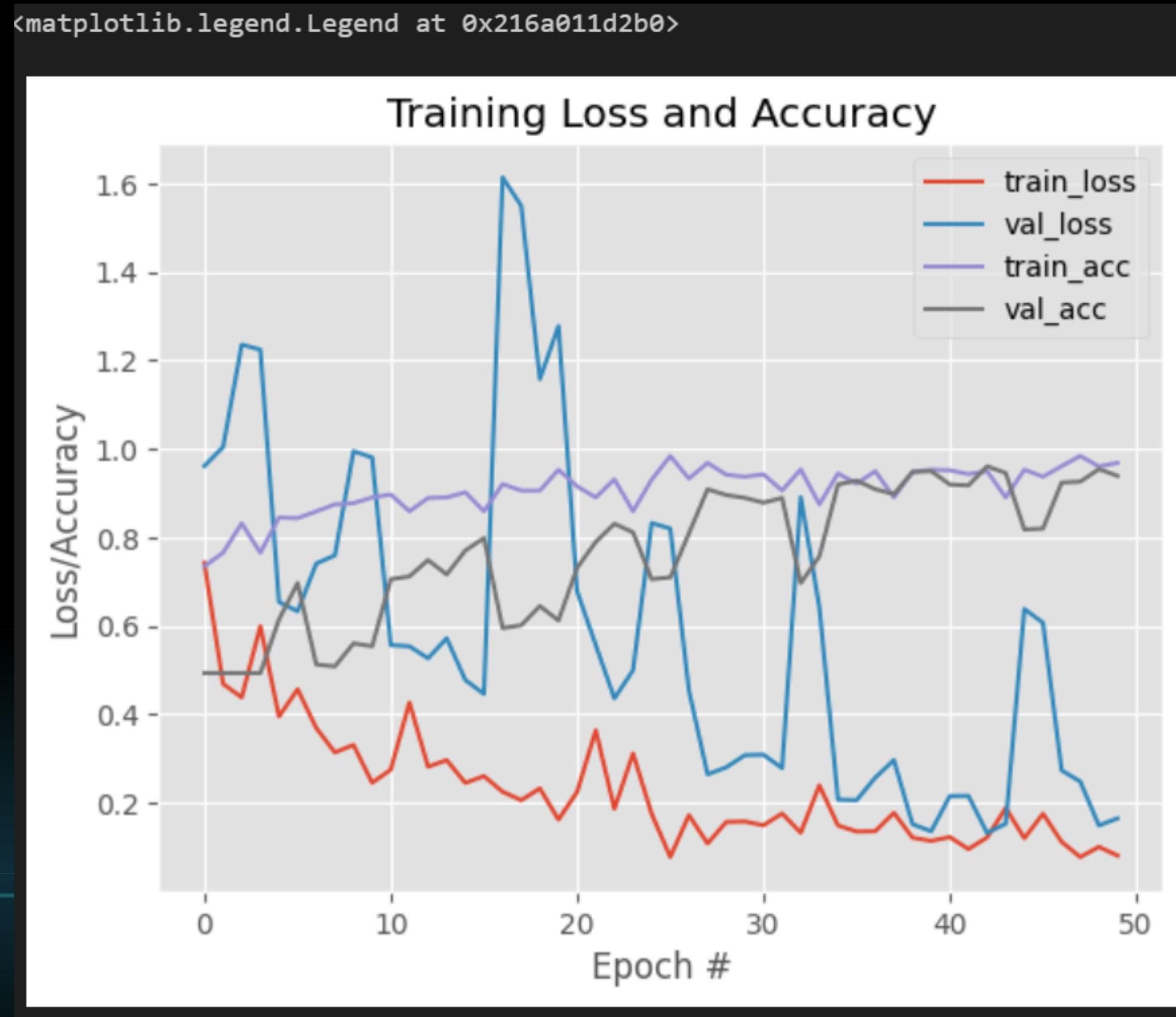
Spaces: 4 Cell 2 of 2

4 70°F Sunny

Search

13:15 12-12-2024

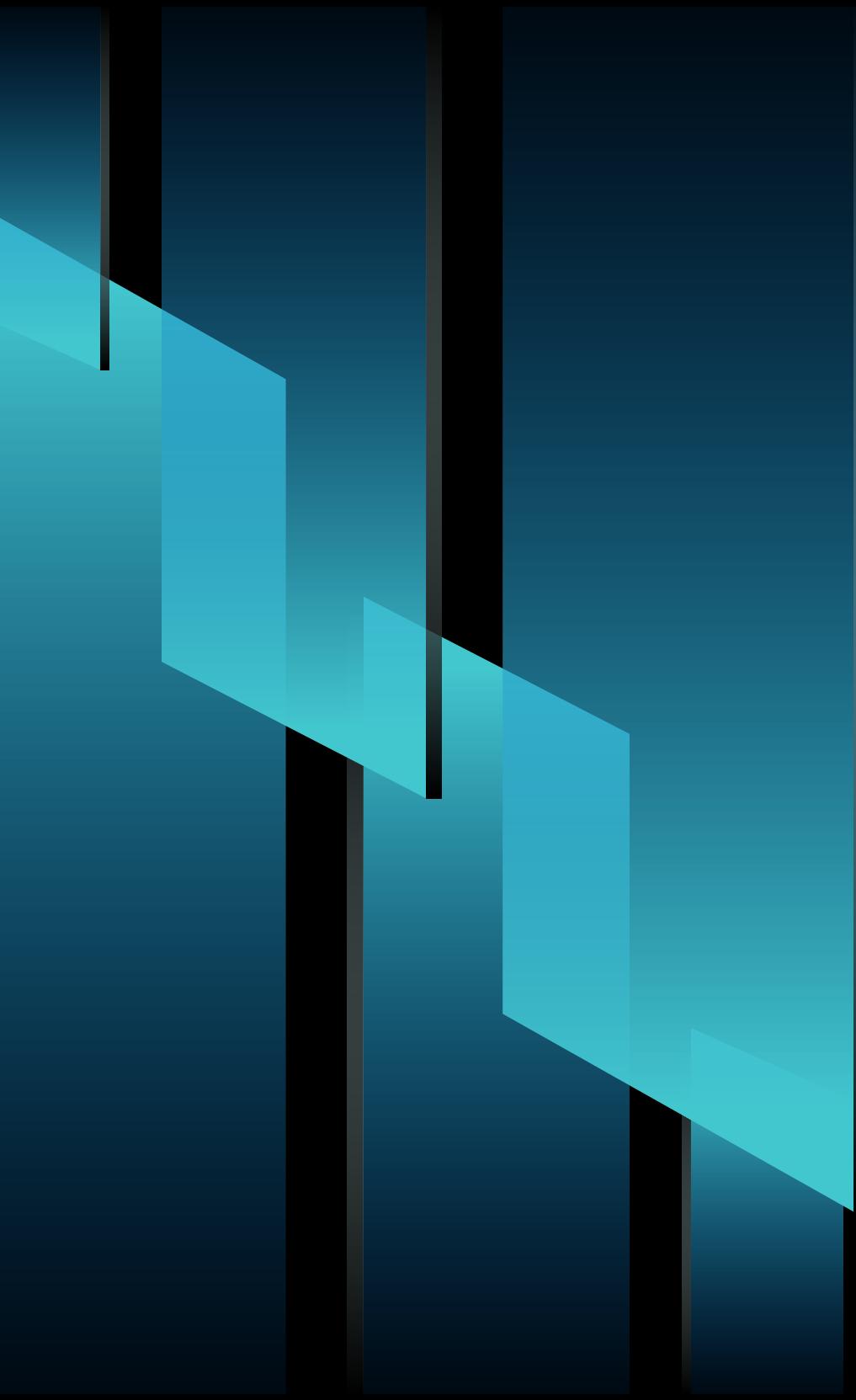
PREDICTION MODEL ACCURACY AND LOSS



CONCLUSION AND FUTURE IMPROVEMENTS

The gender prediction model has an accuracy of 99.01 percentage which is very good for a prediction model.

Along with gender we could have predicted age .



THANK YOU

