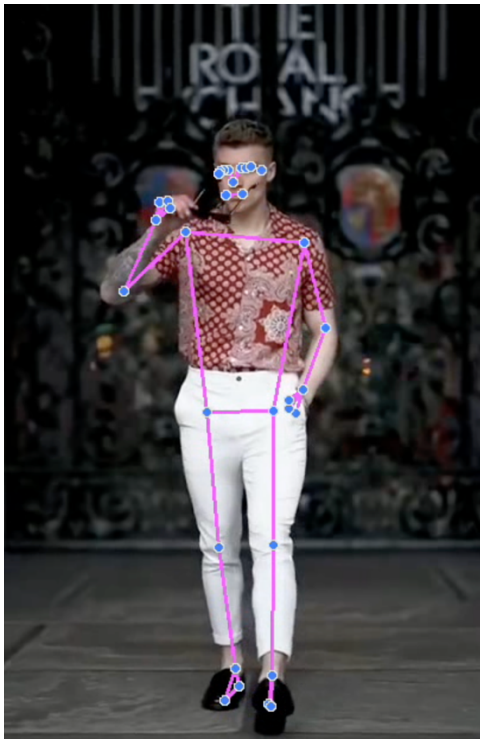
Kiran Kumar NImmakayala
knimmak@emory.edu

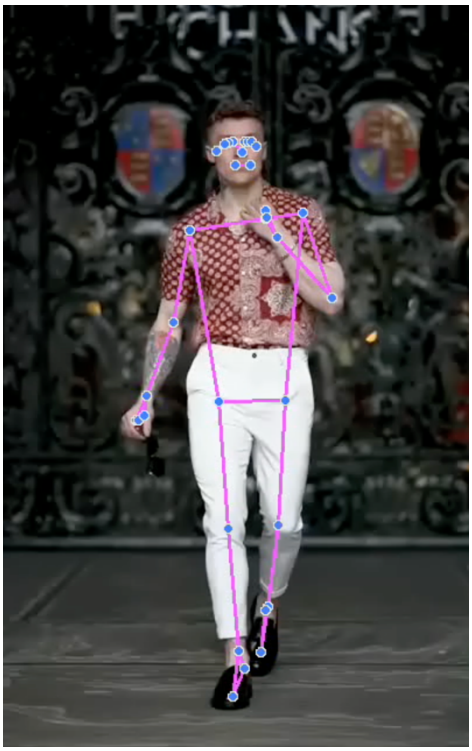
BMI 500: WEEK 13
Edge Computing and Human Activity Analysis

Repository link: https://github.com/kiran-001/BMI-500-Week-13-Homework.git

Mediapipe Using Python

Results:





***The input, output and the code are all in the provided repository ink.*

Human Activity Recognition

Repository link: https://github.com/kiran-001/BMI-500-Week-13-Homework.git

Random Forest Classifier:

Hyperparamters:

```
param_grid = {
   'n_estimators': [20, 30, 50, 100],
   'max_depth': list(np.arange(2, 11)),
   'min_samples_leaf': list(np.arange(2, 11))
}
```

Output:

Best Parameters: {'n_estimators': 30, 'max_depth': 10, 'min_samples_leaf': 5}
Validation Set Accuracy: **0.5487122060470325**
Test Set Accuracy: **0.513840830449827**

Multi-Layer Perceptron Classifier:

Hyperparameters:

```
param_grid = {
   'hidden_layer_sizes': [(100, 50), (50, ), (100, )],
   'alpha': [0.001, 0.01, 0.1],
   'max_iter': [2000],
   'solver': ['adam'],
   'learning_rate': ['adaptive'],
   'activation': ['logistic', 'relu']
}
```

Output:

Best Parameters: {'hidden_layer_sizes': (50,), 'alpha': 0.01, 'max_iter': 2000, 'solver': 'adam', 'learning_rate': 'adaptive', 'activation': 'logistic'}
Validation Set Accuracy: **0.6438969764837627**
Test Set Accuracy: **0.5544982698961938**

Observations:

1. The MLP Classifier outperforms the Random Forest in both validation and test accuracies. With a validation accuracy of around 64.39% and a test accuracy of approximately 55.45%, it seems to generalize better than the Random Forest model. The best parameters for the MLP Classifier indicate that a single hidden layer of 50 neurons, a regularization alpha of 0.01, and the logistic activation function work best for this dataset.
2. The Random Forest Classifier, with its best parameters of 30 trees, a maximum depth of 10, and a minimum samples per leaf of 5, achieved a validation accuracy of about 54.87% and a test accuracy of 51.38%. This suggests that while the model is consistent between validation and testing phases, it doesn't capture the complexity of the data as effectively as the MLP Classifier.
3. The higher performance of the MLP Classifier could indicate that the dataset has complex patterns better captured by a neural network architecture. The Random Forest's slightly lower performance might be due to its inherent simplicity as an ensemble of

decision trees, which might not capture more nuanced patterns in the data as effectively as MLP.

In conclusion, while both models show reasonable performance, the MLP Classifier holds a slight edge in this task, suggesting that the dataset benefits from the more complex, non-linear modeling capability that neural networks provide.

Acknowledgment

I would like to acknowledge the assistance provided by OpenAI's ChatGPT in completing my homework assignment for BMI 500 Week 13. The insights and guidance offered by ChatGPT significantly aided in understanding and solving various aspects of the assignment, particularly in the areas of code development and conceptual explanations. I have thoroughly reviewed and understood all contributions made by ChatGPT and have ensured that they align with the academic integrity standards of the course.

References

1. BMI 500 Lecture Notes: https://drive.google.com/drive/folders/13fXFgqlh6m19XOXiFBn1MwsSPKOHpEps
2. Data Repository: https://www.dropbox.com/s/nzhu004aus5sho8/pose.zip?dl=0
3. Pandas: https://pandas.pydata.org/docs/reference/api/pandas.Series.rolling.html
4. Random Forest Classifier: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
5. Neural Network: https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html#sklearn.neural_network.MLPClassifier
6. Accuracy Score: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html
7. Input Video: https://youtube.com/shorts/_vmErFojNp8?si=HNeJDYeAARAIRQvO
8. Getting Started with MediaPipe in Python: https://google.github.io/mediapipe/getting_started/python.html
9. MediaPipe Pose Solution: https://google.github.io/mediapipe/solutions/pose.html