Project Development Phase
Model Performance Test

Date	8th November 2023
Team ID	Team-591549
Project Name	Audiometric AI:Transforming Hearing TestDiagnosis Through ML
Maximum Marks	10 Marks

Model Performance Testing:

Project team shall fill the following information in model performance testing template.

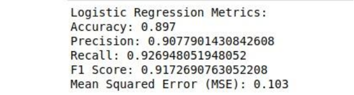
<table>
<tr><th>S.No.</th><th>Parameter</th><th>Values</th><th>Screenshot</th></tr>
<tr><td>1.</td><td>Metrics</td><td>

Regression Model:
MAE - , MSE - , RMSE - , R2 score -

Classification Model:
Confusion Matrix - , Accuray Score- & Classification Report -

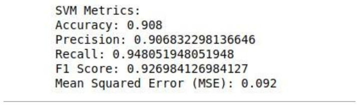
</td><td>

Logistic Regression



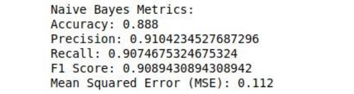
```
Logistic Regression Metrics:
Accuracy: 0.897
Precision: 0.9077901430842608
Recall: 0.926948051948052
F1 Score: 0.9172690763052208
Mean Squared Error (MSE): 0.103
```

SVM



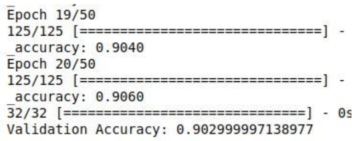
```
SVM Metrics:
Accuracy: 0.908
Precision: 0.906832298136646
Recall: 0.948051948051948
F1 Score: 0.926984126984127
Mean Squared Error (MSE): 0.092
```

Naïve bayes

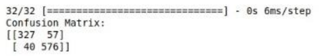


```
Naive Bayes Metrics:
Accuracy: 0.888
Precision: 0.9104234527687296
Recall: 0.9074675324675324
F1 Score: 0.9089430894308942
Mean Squared Error (MSE): 0.112
```

RNN



```
Epoch 19/50
125/125 [==============================] -
_accuracy: 0.9040
Epoch 20/50
125/125 [==============================] -
_accuracy: 0.9060
32/32 [==============================] - 0s
Validation Accuracy: 0.9029999997138977
```



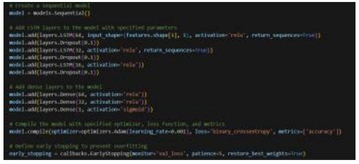
```
32/32 [==============================] - 0s 6ms/step
Confusion Matrix:
[[327  57]
 [ 40 576]]
```

</td></tr>
<tr><td>2.</td><td>Tune the Model</td><td>

Hyperparameter Tuning –
Increasing Number of Layers
Using Adam Optimizer
Validation Method – By checking accuracy of model

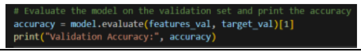
</td><td>

Hyperparameters:



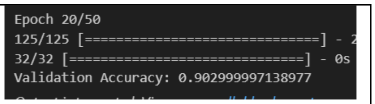
```
# Create a sequential model
model = models.Sequential()

# Add LSTM layers to the model with specified parameters
model.add(layers.LSTM(64, input_shape=(features.shape[1], 1), activation='relu', return_sequences=True))
model.add(layers.Dropout(0.1))
model.add(layers.LSTM(32, activation='relu', return_sequences=True))
model.add(layers.Dropout(0.1))
model.add(layers.LSTM(16, activation='relu'))
model.add(layers.Dropout(0.1))

# Add Dense layers to the model
model.add(layers.Dense(64, activation='relu'))
model.add(layers.Dense(32, activation='relu'))
model.add(layers.Dense(1, activation='sigmoid'))

# Compile the model with specified optimizer, loss function, and metrics
model.compile(optimizer=optimizers.Adam(learning_rate=0.001), loss='binary_crossentropy', metrics=['accuracy'])

# Define early stopping to prevent overfitting
early_stopping = callbacks.EarlyStopping(monitor='val_loss', patience=5, restore_best_weights=True)
```

Validation



```
# Evaluate the model on the validation set and print the accuracy
accuracy = model.evaluate(features_val, target_val)[1]
print("Validation Accuracy:", accuracy)
```

</td></tr>
</table>

			
			`Epoch 20/50`
			`125/125 [==============================] - 2`
			`32/32 [==============================] - 0s`
			`Validation Accuracy: 0.902999997138977`