“You Snooze, You Win”

•••

Medlytics Week 2 Project

Group 8: Arsh Goenka, Angel Li, Kshitij Teotia, Sanjana Pulaparthi

Overview

1. Feature Extraction
2. Heat Maps
3. Choosing Layers
4. Neural Network Architecture
5. Accuracy and ROC
6. LSTM Layer

Feature Extraction

Spectral Flatness

- Increased accuracy slightly (measure to quantify how much noise-like a sound is)

Spectral Centroid

- Changed nans to 0s
- Increased Accuracy
- a measure used in digital signal processing to characterise a spectrum

Spectral Flux

- Increased accuracy
- For-loop deterred program (used .diff)

Max Frequency

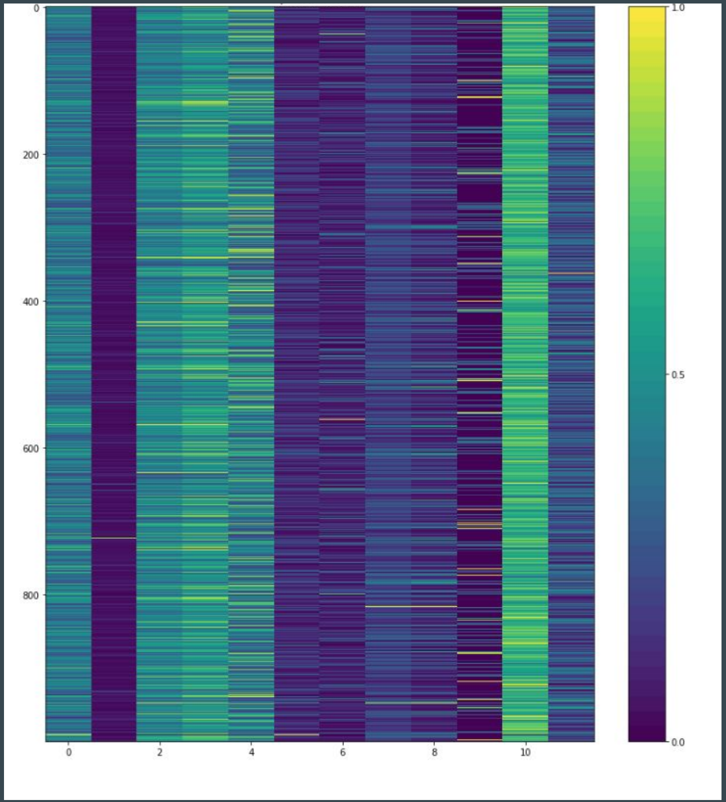
- Showed better performance on heat map

Spectral RollOff

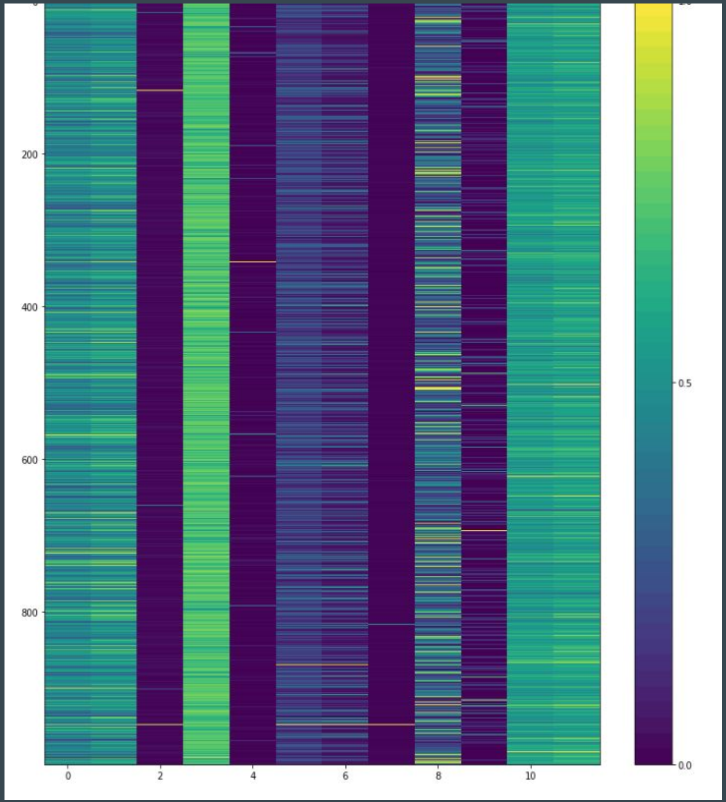
- Increased accuracy, showed better results on heat map
- The frequency below which a specified percentage of the total spectral energy

Heat Maps

Spectral Flatness

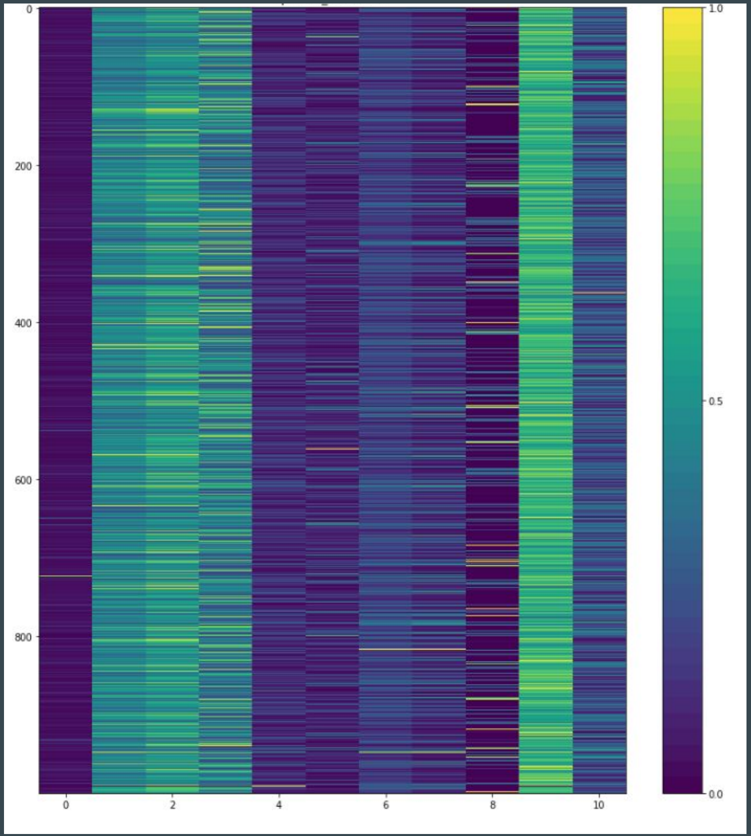


Spectral Centroid

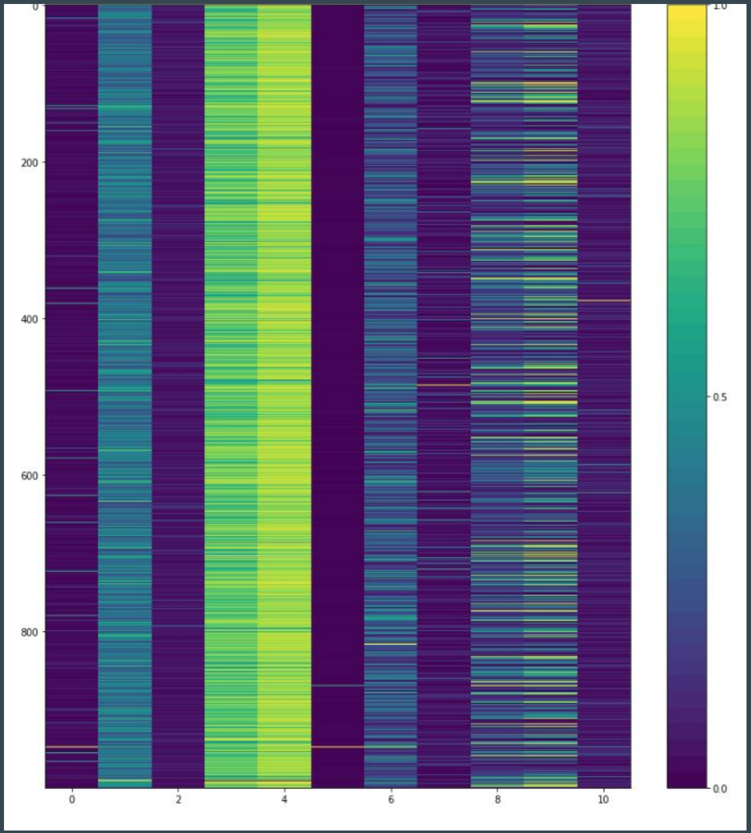


Heat Maps

Spectral Flux

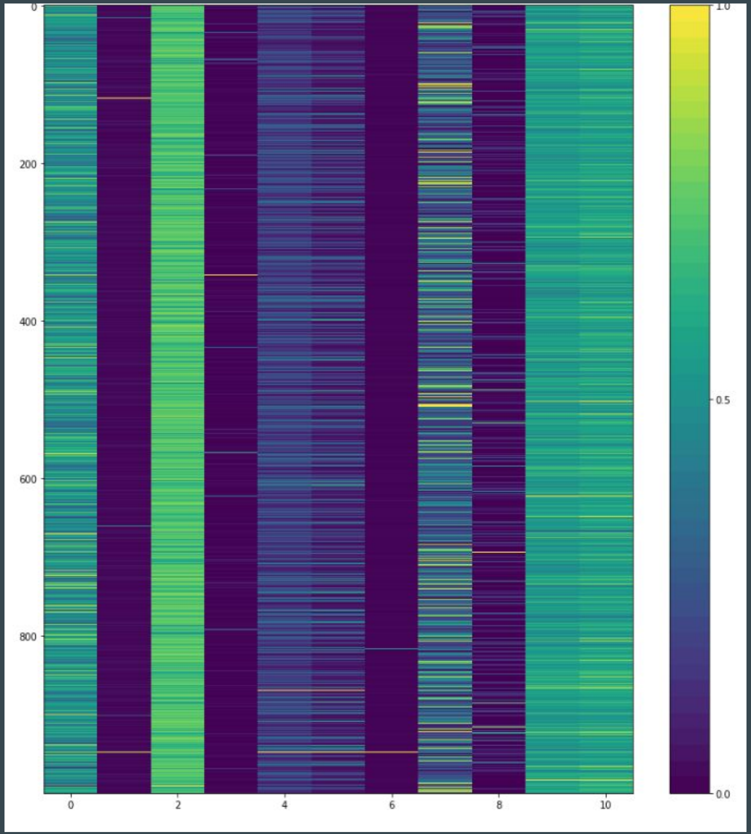


Max Frequency



Heat Maps

Spectral Rolloff



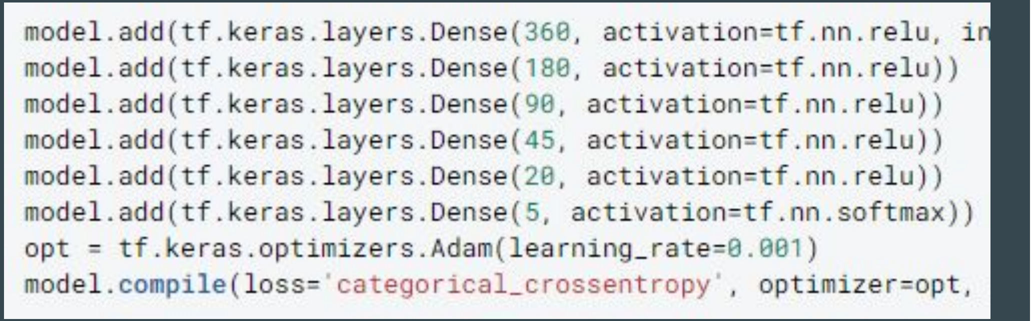
Choosing Layers

Tried many different combos of layers, e.g. dropout

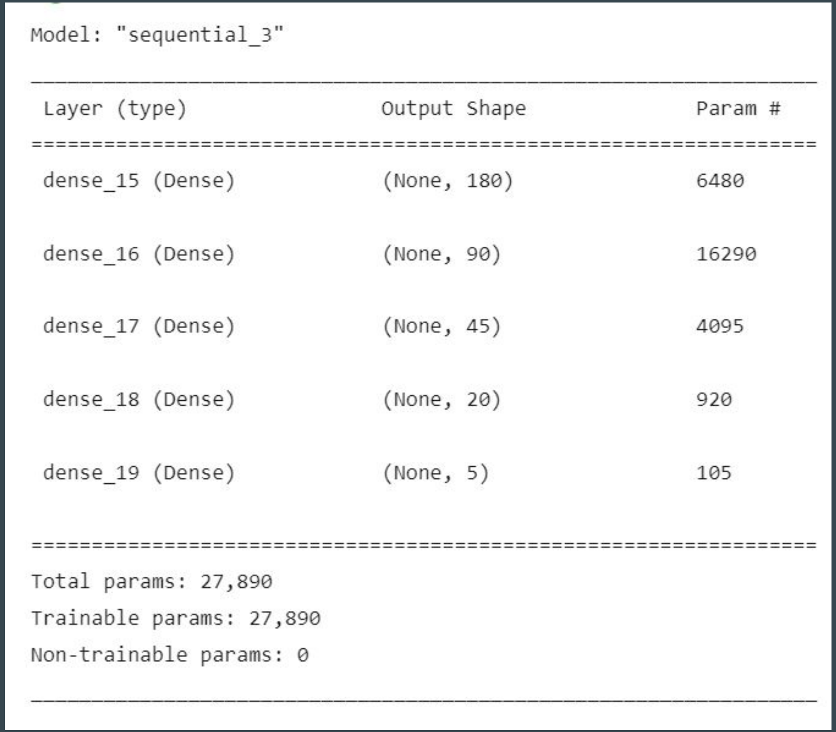
Layers Used:

- Dense Layers in decreasing order
- Softmax for predictions
- RelU Activation
- Adam optimizer
- 100 Epochs
- lr = 0.001

Our Neural Network!



```python
model.add(tf.keras.layers.Dense(360, activation=tf.nn.relu, in
model.add(tf.keras.layers.Dense(180, activation=tf.nn.relu))
model.add(tf.keras.layers.Dense(90, activation=tf.nn.relu))
model.add(tf.keras.layers.Dense(45, activation=tf.nn.relu))
model.add(tf.keras.layers.Dense(20, activation=tf.nn.relu))
model.add(tf.keras.layers.Dense(5, activation=tf.nn.softmax))
opt = tf.keras.optimizers.Adam(learning_rate=0.001)
model.compile(loss='categorical_crossentropy', optimizer=opt,
```



```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_15 (Dense)            (None, 180)               6480

 dense_16 (Dense)            (None, 90)                16290

 dense_17 (Dense)            (None, 45)                4095

 dense_18 (Dense)            (None, 20)                920

 dense_19 (Dense)            (None, 5)                 105

=================================================================
Total params: 27,890
Trainable params: 27,890
Non-trainable params: 0
_________________________________________________________________
```

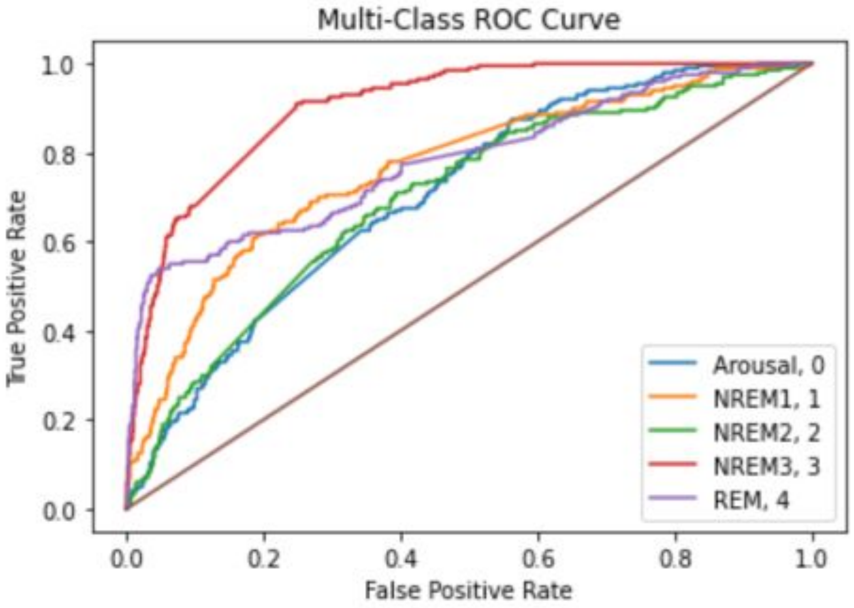
Regular NN Model

Test Accuracy: 0.5220

AUC: 0.8054615

MCC: 0.232700262714036

How can we improve our accuracy?



LSTM Layer

😭

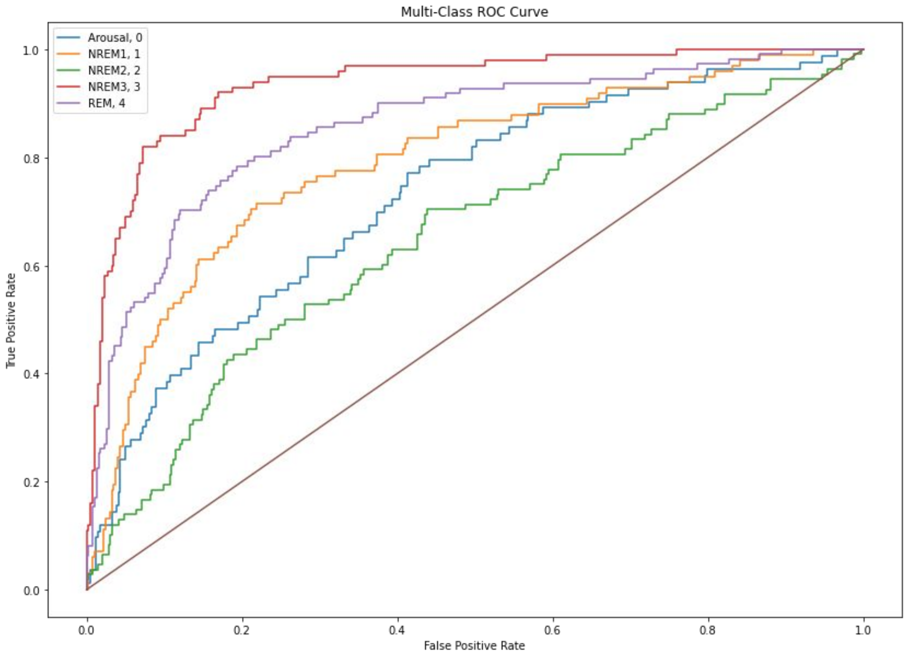
- *Long short-term memory* (LSTM) layer is the most common used type of RNN layer
- Online researches of the use of RNN on sleep stages
- Problem of implementation
  - The layer takes 3d input → reshaping 2d creates issue
- But...

RNN Results

Test Accuracy: 0.5415

AUC: 0.805851

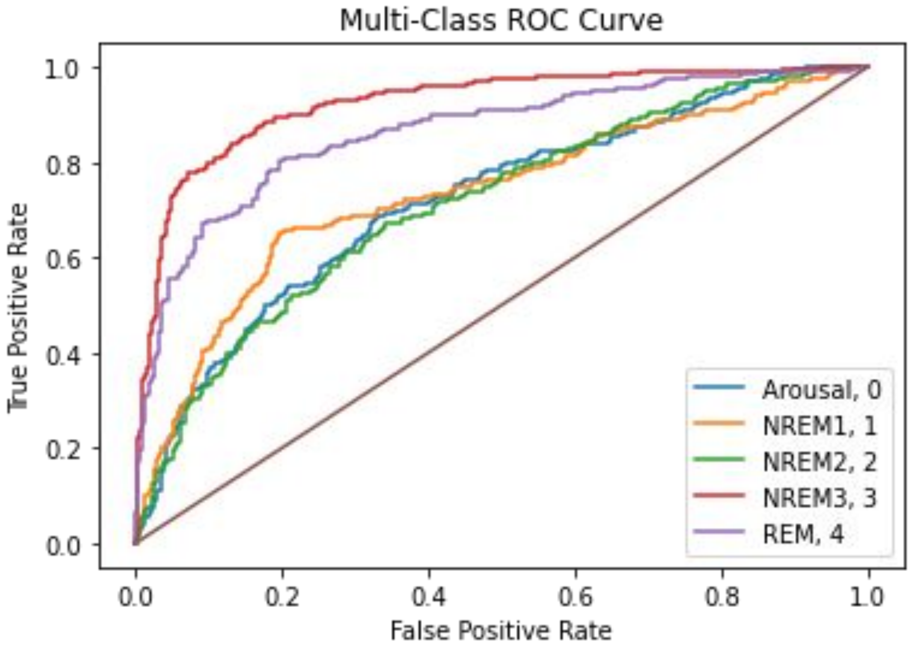
MCC: 0.278322



RNN Pkl Results

AUC: 0.8029

MCC: 0.2781


Multi-Class ROC Curve
True Positive Rate
False Positive Rate
Arousal, 0
NREM1, 1
NREM2, 2
NREM3, 3
REM, 4