Thesis Report 9 : 20 April - 27 April

Goals

- Keep tweaking the Variational AutoEncoder ✅
- Create model(s) to map PAD-Latent Space ✅

Last Week Leftovers:

None

Done

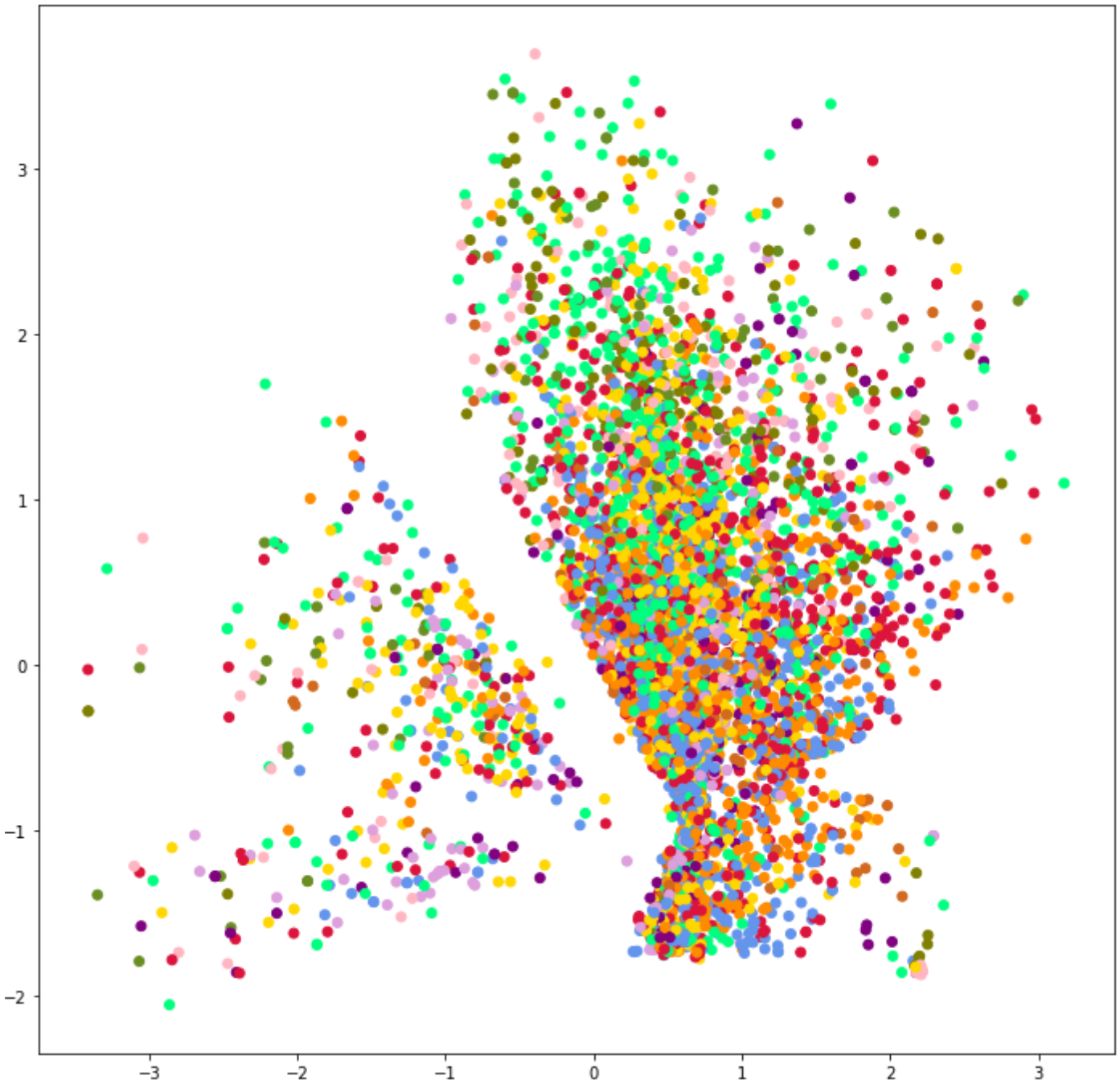
- Kept trying new architectures for the Variational Auto Encoder networks (both the encoder and decoder)
- Increased latent space from 1 variable to 2 (and also tried with 3)
- Managed to reduce average absolute reconstruction error from 0.15 to about 0.11
- Used XGBoostRegression to map PAD coordinates to Latent Space
    - Results were better than last week (when we only used a latent space of 1), but were still lacking

```
index = 2

# Real -> PAD coordinates
#   - get latent space
#   - convert from latent to LMA features
#   - standardize
#   - predict coordinates of generated features

sample = np.asarray([test_X.iloc[index]])
latent = np.asarray([[model_1.predict(sample)[0], model_2.predict(sample)[0]]])#, model_3.predict(sample)[0]]])


generated_lma = decoder.predict(latent)


scaled_gen = scaler.transform(generated_lma)

generated_coord = (
    model_p.predict(scaled_gen)[0],
    model_a.predict(scaled_gen)[0],
    model_d.predict(scaled_gen)[0]
)

print('Real: %s' % sample[0])
print('Predicted: %s' % np.asarray(generated_coord))
```

```
Real: [-0.5813058  -0.29254228 -0.28772822]
Predicted: [-0.29099813  0.26557478 -0.10210333]
```

- Tried using a Neural Network (MLP) to perform the mapping between PAD and Latent Space
    - Results were, once again, not good and the model seemed to not be learning much

- After multiple attempts at trying to perform the PAD-Latent space mapping, I was starting to think there was something wrong. So I created a graph that showcased the Latent Space coordinates and the class to which each point belonged to.
    - As can be seen in the image below, each color corresponds to a different emotions
    - **All points are clustered together, irregardless of their emotion, which is why neither regression nor neural network were able to create a decent mapping between PAD coordinates and Latent Space coordinates**
    - As such, I don't think the idea of having a mapping between PAD and Latent space is possible...



- Tried using a normal Deep Autoencoder rather than a Variational

- Average reconstruction error was +- the same

```
sample = np.asarray(x_test[1])
sample = sample.reshape(1,-1)
print(sample)
```

```
[[ 0.48787081  0.25350573  0.26362314  0.28867775  0.46451875  0.46793826
   0.34736974  0.3513925   0.286151    0.27886443  0.03283048 -0.18807481
  -0.01528917  0.98136552  0.11219     0.15637932  0.12656324  0.34512916
   0.21659343  0.21783337  0.18811772  0.28809448  1.68458422  0.12708601
   0.25069947  0.49993658  0.44727669]]
```

```
generated = encoder.predict(sample)
print(generated)
```

```
[[-0.5224355  -0.8290596   0.05003154]]
```

```
regen = decoder.predict(generated)
print(regen)
```

```
[[ 0.48628756  0.31383598  0.27678993  0.3235078   0.44406185  0.44796878
   0.3380221   0.33729854  0.28648007  0.27869436  0.00637171 -0.12865256
  -0.04899131  0.98284596  0.23394868  0.12174594  0.13326922  0.45180503
   0.13425937  0.14767751  0.16074066  0.15782107  1.6859633   0.12582749
   0.42136034  0.43196467  0.3978675 ]]
```

- PAD to Latent space mapping worked a bit better here (probably because the VAE has the random Epsilon variable meaning each time the Encoder runs, a different latent space set is generated). They were still not great, but at least they were decent

```
Latent 1
MSE: 0.17
MAE: 0.22

Latent 2
MSE: 0.24
MAE: 0.26

Latent 3
MSE: 0.71
MAE: 0.44
```

- **Tried reducing the feature set from 27 LMA features down to 19 (only for the PAD-LMA mapping).**
  - Didn't really help the results on the autoencoders/pad-latent space mappings
  - Does make using a direct regression between PAD-LMA feasible, since the removed features were the ones that had the highest error

Left Undone

Problems

- **[NOT SOLVED]** VAE autoencoder generates a latent space coordinates that don't form clusters according to the emotion. All sample points are clustered together, which makes creating a regression between PAD and the Latent Space impossible

Notes

Thoughts

Continued working on trying to improve the Autoencoders. Whilst I managed to reduce the overall reconstruction error by a bit, it wasn't that great of an improvement. On the other hand, I really struggled creating the PAD-Latent Space regression. Tried using regression, and creating a Neural Network for the mapping (whilst playing around with the architecture and all that), but even then results were pretty poor, and a discovery made me think that this route is not feasible (see above).

Using a regular autoencoder rather than a variational made the PAD-Latent Space mapping more feasible but regardless results weren't as great as I wanted. Tried reducing the feature set down but even that didn't work.

Despite the lacking results on the PAD-LMA portion (which did stress me quite a bit this week), I think I'm going to start working on the Motion Synthesis algorithm. I'll still leave models training and will keep trying to improve this portion of the work, but I wanna start doing the motion synthesis to get that underway (and to maybe try to come up with better solutions to perform the PAD-LMA mapping).

Work Hours

- Worked Thursday-Friday from 1pm to 6-8pm
- Worked Sunday-Tuesday from 1pm to 6pm