Thesis Report 10 : 4 May - 11 May

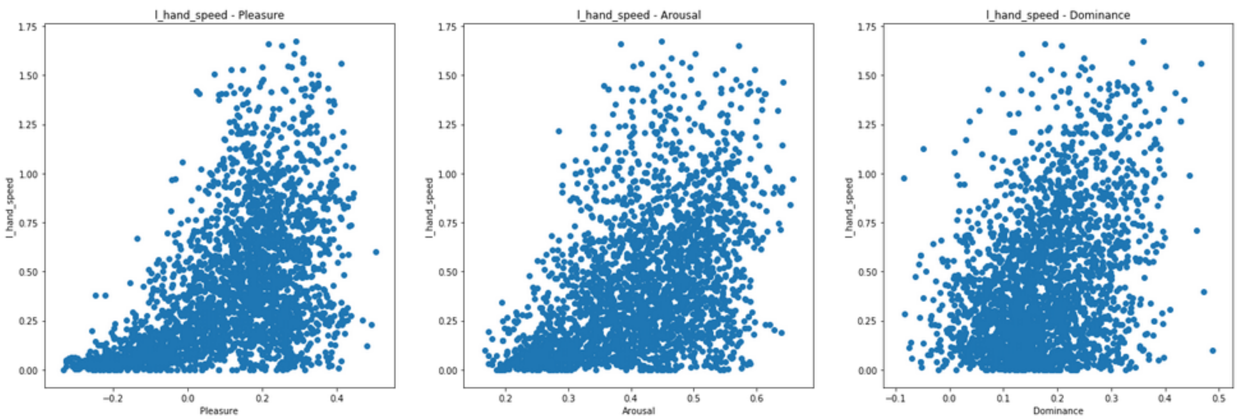
Goals

- Continue working on PAD-LMA ✅

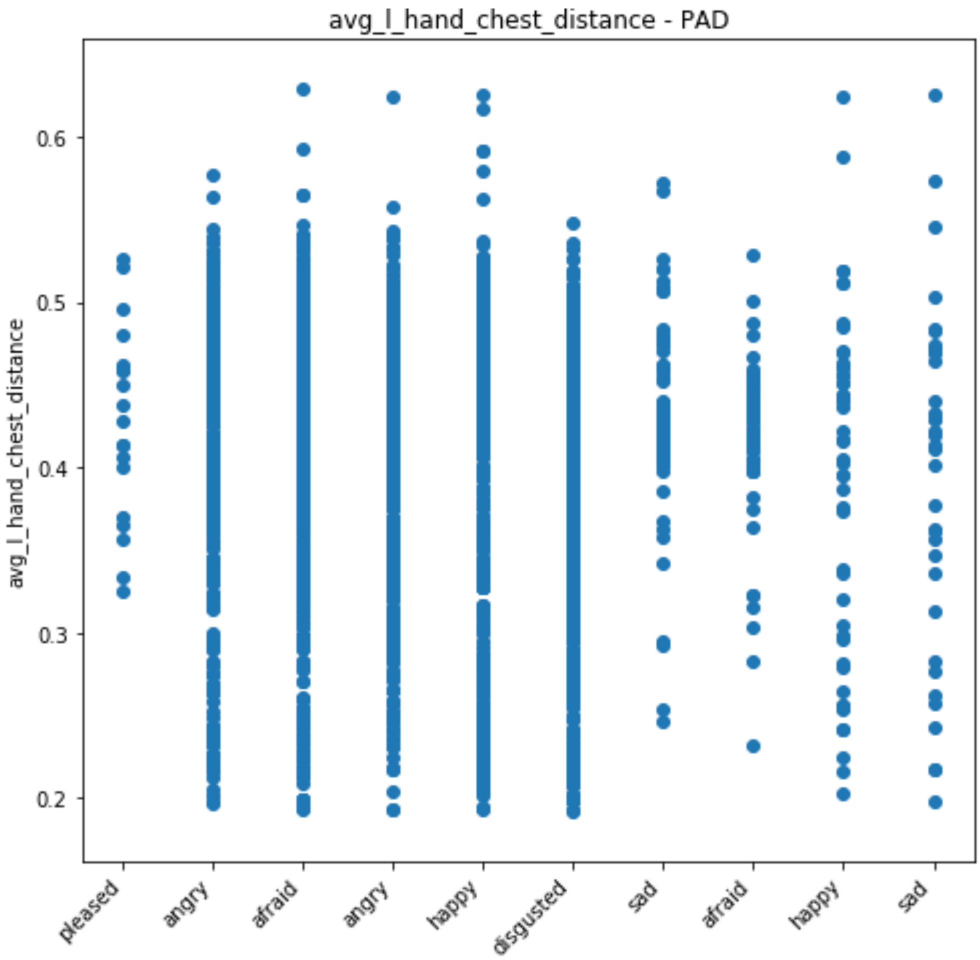
Last Week Leftovers:

None

Done

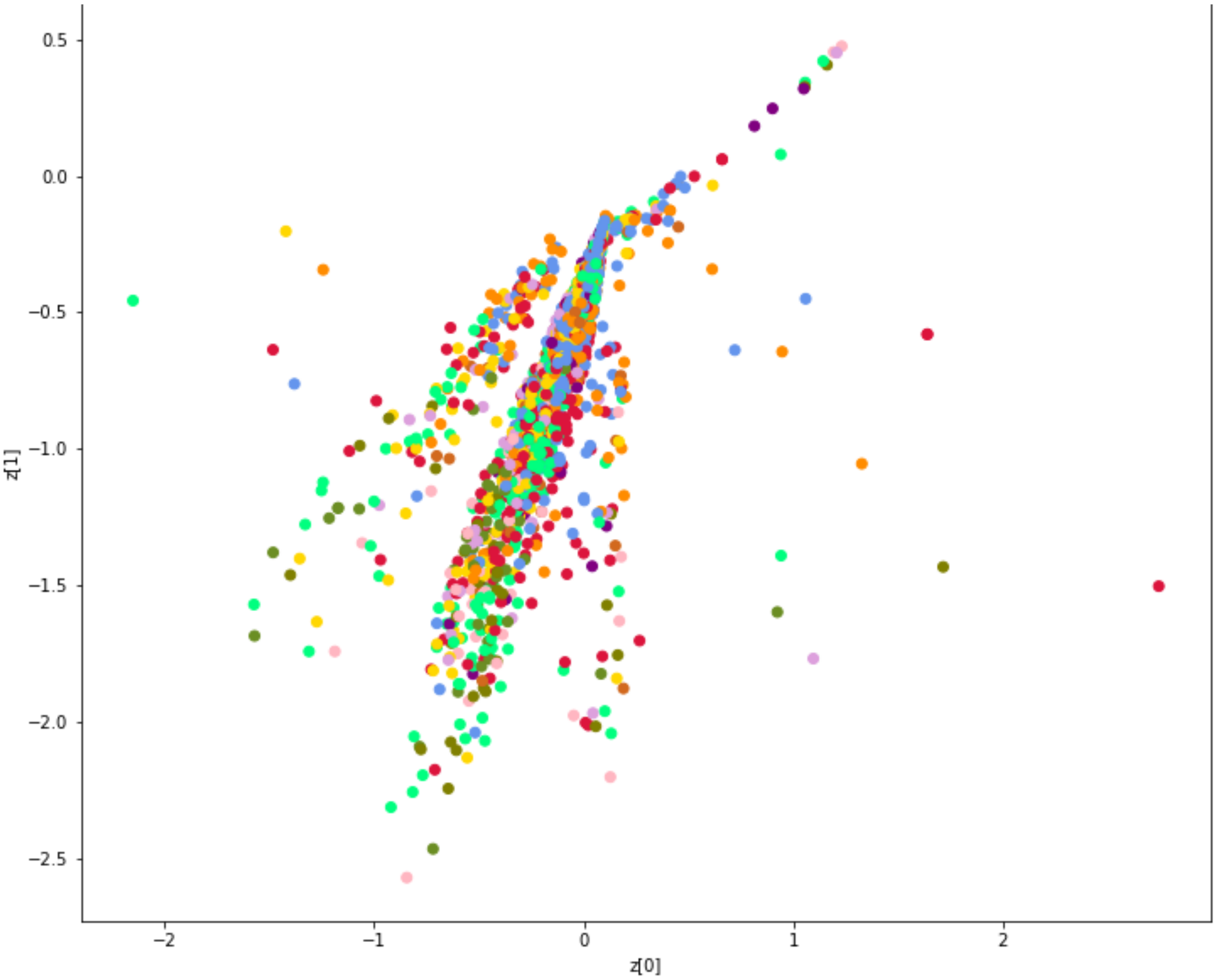
- Performed the suggested analysis of seeing how each LMA feature varies in regards to the emotional coordinates
    - The following graph was created by placing the sampled value of any given LMA feature on the Y axis, and the corresponding (predicted) P,A,D values on the X axis
    - 

    - Another analysis that was performed was checking the range of variance of each LMA feature according to each of our core emotions

- 

    - From these 2 sets of graphs we can see that, regardless of emotion, the values for each LMA feature have quite a wide range in which they vary. Regardless, despite each feature varying so much, I presume our LMA-PAD regression manages to work due to the fact that even if, individually each feature varies a lot, there are still patterns that can be extracted when looking at the entire set of LMA features, making our algorithm capable of discerning between emotions anyways.

- Kept working on the Variational Autoencoder
    - I noticed that there were some bugs in my implementation code, and that I was using a deprecated method for creating the VAE (due to the fact that I was following an out of date set of tutorials from 2016)
    - Re-implemented the VAE from scratch, experimented with the architecture, weight of the reconstruction error VS KL-Convergence error and some other hyper parameters, but never managed to get decent results.
    - Went back to try and use a normal AutoEncoder rather than a Variational one. Whilst the reconstruction was good, the latent space coordinates weren't really distinguishing between emotions and as such, the mapping between PAD coordinates and Latent Space would've been problematic.



- The problems with the Autoencoders could have probably been fixed by adding more latent space coordinates. However, the whole idea behind using autoencoders was to make the mapping between PAD-LMA easier (i.e, it was so that, rather than having to extract a set of 27 LMA features from just 3 PAD coordinates, we could simply perform a mapping between 3 PAD coordinates and 2 Latent Space coordinates, and then have the decoder go from those coordinates to the LMA Features), and as such, I decided to go back a step and retry using both the XGBoostRegression or Neural Networks to go directly from PAD to LMA

- Came to 2 realizations, especially after the aforementioned analysis :
  - First, **instead of using the preset emotional coordinates of each LMA feature set, it would probably be best to use the predicted emotional coordinates**, obtained using our LMA-PAD regressors. This thought came from the fact that our initial labels expect that each LMA set perfectly represents each emotion (e.g, each "angry" LMA set has the exact same Pleasure, Arousal and Dominance values as all others), however this is not true since each different LMA set will have slightly different PAD values (which shouldn't vary too far from the intended emotion one's). As such, using these predicted PAD values may end up giving us a more "true" PAD correspondent to each LMA set
  - Secondly, **rather than evaluating how close the generated LMA set is to the original, a better metric would be how similar the generated LMA set's Emotional Coordinates are to the originals**. After all, we don't really care to generate the same

LMA set, what we really want is to be able to generate an LMA set with the same/similar emotion.

- Started trying to use a neural network to map between PAD and LMA directly
    - Tried a lot of different architectures
    - Used the predicted PAD values rather than the original ones
    - Tried (but failed) to implement a custom loss function that would compute the MSE between true and generated PAD values rather than true and generated LMA features
    - **Managed to achieve a MAE of 0.22 in regards to LMA features, and 0.29 in regards to PAD coordinates**

```
Overall MAE: 0.29988389569649

==PLEASURE==
MSE: 0.14597
MAE: 0.33477

==AROUSAL==
MSE: 0.10633
MAE: 0.27787

==DOMINANCE==
MSE: 0.15841
MAE: 0.28701

Real: [-0.4, 0.25, -0.1]
Predicted: [-0.37550514936447144, 0.24989724159240723, -0.08963838219642639]
Generated: [-0.24819766, 0.41801158, -0.29611397]

Real: [-0.5, 0.6, 0.9]
Predicted: [-0.4983406066894531, 0.5813348293304443, 0.8807997703552246]
Generated: [-0.2042435, 0.2730803, 0.13429697]

Real: [0.9, -0.25, 0.65]
Predicted: [0.8968958258628845, -0.2466982752084732, 0.6371656060218811]
Generated: [0.3796192, 0.3351414, 0.22968563]

Real: [0.1, -0.7, -0.2]
Predicted: [0.10403016954660416, -0.6769911646842957, -0.19417503476142883]
Generated: [-0.021247152, 0.021191992, -0.026079793]

Real: [-0.5, 0.6, 0.9]
Predicted: [-0.490100234746933, 0.6017720103263855, 0.894679605960846]
Generated: [-0.21545453, 0.24272734, 0.16180275]

Real: [-0.6, -0.3, -0.3]
Predicted: [-0.5879993438720703, -0.28604114055633545, -0.281313478946 6858]
Generated: [-0.14052577, 0.085983835, -0.27169922]

Real: [-0.4, 0.25, -0.1]
Predicted: [-0.3952067196369171, 0.2533014714717865, -0.10019537061452866]
Generated: [-0.22174089, 0.47376812, -0.1797663]
```

```
Overall MAE: 0.2252911815184764

==max_hand_distance==
MSE: 0.06063
MAE: 0.19545
Example [Regen-Real]: 0.6066096 - 1.2091379770108246

==avg_l_hand_hip_distance==
MSE: 0.01934
MAE: 0.10790
Example [Regen-Real]: 0.3558449 - 0.3069123971479425

==avg_r_hand_hip_distance==
MSE: 0.02458
MAE: 0.12210
Example [Regen-Real]: 0.349559 - 0.6331036984351217

==max_stride_length==
MSE: 0.02099
MAE: 0.11278
Example [Regen-Real]: 0.34502164 - 0.2674154236347452

==avg_l_hand_chest_distance==
MSE: 0.01147
MAE: 0.08245
Example [Regen-Real]: 0.4421267 - 0.5189878173868507

==avg_r_hand_chest_distance==
MSE: 0.01187
MAE: 0.08358
Example [Regen-Real]: 0.41138095 - 0.5110179179430487

==avg_l_elbow_hip_distance==
MSE: 0.00531
MAE: 0.05451
Example [Regen-Real]: 0.31572235 - 0.3660186672849354

==avg_r_elbow_hip_distance==
MSE: 0.00580
MAE: 0.05573
Example [Regen-Real]: 0.32472882 - 0.3132046876613081

==avg_chest_pelvis_distance==
MSE: 0.00000
MAE: 0.00054
Example [Regen-Real]: 0.2864597 - 0.2861509999974518
```

- Started trying to use the XGBRegression to map between PAD and each LMA feature individually
  - Used the predicted PAD values rather than the original ones
  - **Managed to achieve an MAE of 0.27 in regards to PAD coordinates (under 0.3 for each coordinate)!**

```
Overall MAE: 0.27130481733662537

==PLEASURE==
MSE: 0.11627
MAE: 0.27223

==AROUSAL==
MSE: 0.09735
MAE: 0.24842

==DOMINANCE==
MSE: 0.15357
MAE: 0.29327

Real: [-0.4, 0.25, -0.1]
Predicted: [-0.42116978764533997, 0.28247973322868347, -0.1176881492137909]
Generated: [-0.21761528, 0.3197857, -0.23946458]

Real: [-0.5, 0.6, 0.9]
Predicted: [-0.4979105293750763, 0.5989544987678528, 0.8933809995651245]
Generated: [-0.28674498, 0.39488798, 0.44668525]

Real: [-0.85, -0.1, -0.8]
Predicted: [-0.85365229845047, -0.10117416083812714, -0.7981707453727722]
Generated: [-0.5570986, -0.13602474, -0.6645763]

Real: [0.6, 0.5, 0.2]
Predicted: [0.5783243179321289, 0.49849289655685425, 0.19792142510414124]
Generated: [0.27329704, 0.514154, 0.23011701]
```

- LMA errors were also pretty low for the most part (under 0.1 for all features, except for the acceleration features)

```
max_hand_distance
MSE: 0.02
MAE: 0.06
Example:  0.6362709709532175

avg_l_hand_hip_distance
MSE: 0.01
MAE: 0.06
Example:  0.2915728188725492

avg_r_hand_hip_distance
MSE: 0.01
MAE: 0.07
Example:  0.2938694130511313

max_stride_length
MSE: 0.01
MAE: 0.05
Example:  0.2986755333995243

avg_l_hand_chest_distance
MSE: 0.00
MAE: 0.04
Example:  0.4604544342723543

avg_r_hand_chest_distance
MSE: 0.00
MAE: 0.04
Example:  0.4608447595536516

avg_l_elbow_hip_distance
MSE: 0.00
MAE: 0.03
Example:  0.3389554435714554

avg_r_elbow_hip_distance
MSE: 0.00
MAE: 0.02
Example:  0.3350770081450494

avg_chest_pelvis_distance
MSE: 0.00
MAE: 0.00
Example:  0.2861510000594406
```

```
r_hand_speed
MSE: 0.05
MAE: 0.10
Example:  0.9207111035365434

l_foot_speed
MSE: 0.03
MAE: 0.11
Example:  0.125107086754325

r_foot_speed
MSE: 0.04
MAE: 0.09
Example:  0.1428467602037255

neck_speed
MSE: 0.02
MAE: 0.08
Example:  0.1940831777818028

l_hand_acceleration_magnitude
MSE: 0.37
MAE: 0.36
Example:  3.0529558474600247

r_hand_acceleration_magnitude
MSE: 0.39
MAE: 0.40
Example:  2.913213596919504

l_foot_acceleration_magnitude
MSE: 0.17
MAE: 0.21
Example:  0.1600392314176911

r_foot_acceleration_magnitude
MSE: 0.20
MAE: 0.19
Example:  0.2425906106567401

neck_acceleration_magnitude
MSE: 0.16
MAE: 0.28
Example:  0.5288086905107043
```

- Started working on algorithm for motion synthesis
  - Reread the Emotion Control paper and started trying to think of ways to adapt the way they performed motion synthesis to our own work

Left Undone

Problems

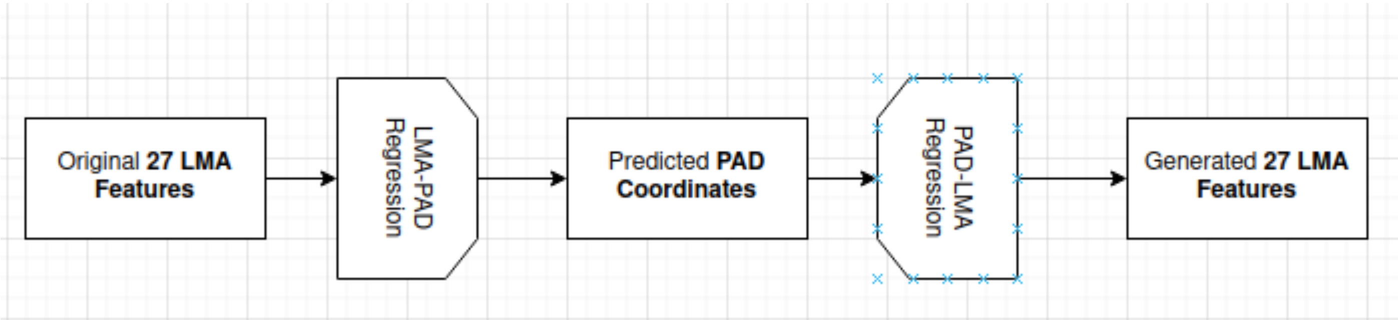
Notes

Thoughts

After a week outside the country I got back to work right where I left off. Progress was made. I think that, at least for now, I will be leaving behind the idea of using the autoencoders. Although I really like the novelty of that solution, and the fact that it sounds good on paper, in practice I wasn't able to achieve the results I wanted. I believe the idea was actually complicating the problem, rather than simplifying it, adding more steps of potential error/failure to the pipeline.

By taking a step back and using either a regular neural network or our tried and tested XGBRegression I managed to create a set of models that can generate sets of LMA features with the requested PAD coordinates (or at least, close-ish PAD coordinates, on the worst case they tend to always fall on the correct octant of the PAD 3D-Diagram). These models still have room for improvement (I wanna try to get the MAE error of the PAD coordinates as low as possible, hopefully to under 0.2 for each coordinate), but right now we have a complete system that can:

- Receive a set of LMA features
- Output the PAD coordinates of said features
- Receive PAD coordinates
- Output a set of LMA features similar to the first one (or at least, one that showcases the same PAD coordinates)



I will be now starting to work on the Motion Synthesis part of the project (the last part missing). I'm still aiming to have the entire implementation work done by May, and as such I will be dedicating the following 2 weeks to using the generated LMA features, and actually synthesizing changes to the motions so that users can edit the emotion the motion is expressing. I'll admit I'm abit afraid about this. I have some ideas of how I'll be proceeding (using the Emotion Control paper as most of my inspiration for the algorithm), but I'm a bit fearful of things not working as intended, or about not being able to actually use the generated LMA features to create new joint positions. As such, I really want to start working on this ASAP.

I'll still be tweaking and trying to improve the PAD-LMA regressions (leaving models training while I create the motion synthesis algorithms), but as they are now, I believe they are, at least, presentable.

Work Hours

- Worked Thursday-Friday from 1pm to 6pm
- Worked Saturday from 7pm to 9pm
- Worked Sunday from 12pm to 10pm
- Worked Monday-Tuesday from 1pm to 8pm