## Neural Networks

The model predicts quality scores transformed from GMM loglikelihood for the between-subject case. The inputs to the networks are the skeletal joints data with 117 dimensions.

Spatio-Temporal NN for Vicon folder contains the proposed deep learning model in the corresponding article.

SpatioTemporalNN-Vicon: predicts quality scores and "Data_Load_GMM_Bet" is called to load data. The fixed permutation "M1_Shuffled_Indices" in Data_shuffle is used to shuffle the data.

Spatio-Temporal NN for Kinect v2 folder contains codes for the proposed deep learning model using skeletal data collected with Kinect v2 sensor from the open dataset KIMORE. The implementation details are provided in the paper.

CNN_GMM_Between_M1: predict quality scores and "Data_Load_GMM_Bet" is called to load data. The fixed permutation "M1_Shuffled_Indices" in Data_shuffle to shuffle the data.

The same naming rule is applied for RNN.

CNN_GMM_Between_M1_Aug: predict quality scores for the augmented data and "DataA_Load" is called to load data. The split function is used to shuffle data randomly.

The same naming rule is applied to RNN.

## Distance Functions

Euc_between_subjects: calculates Euclidean distance on the raw data for the between-subject case

Euc_within_subjects: calculates Euclidean distance on the raw data for the within-subject case

Maha_between_subjects: calculates Mahalanobis distance on the raw data for the between-subject case

Maha_within_subjects: calculates Mahalanobis distance on the raw data for the within-subject case

DTW_between_subjects: calculates DTW distance on the raw data for the between-subject case

DTW_within_subjects: calculates DTW distance on the raw data for the within-subject case

Var_Euc_between_subjects: uses maximum variance to reduce the dimensionality of raw data, and afterward calculates Euclidean distance on the low-dimensional data for the between-subject case

Var_Euc_within_subjects: uses maximum variance to reduce the dimensionality of raw data, and afterward calculates Euclidean distance on the low-dimensional data for the within-subject case

Var_Maha_between_subjects: uses maximum variance to reduce the dimensionality of raw data, and afterward calculates Mahalanobis distance on the low-dimensional data for the between-subject case

Var_Maha_within_subjects: uses maximum variance to reduce the dimensionality of raw data, and afterward calculates Mahalanobis distance on the low-dimensional data for the within-subject case

Var_DTW_between_subjects: uses maximum variance to reduce the dimensionality of raw data, and afterward calculates DTW distance on the low-dimensional data for the between-subject case

Var_DTW_within_subjects: uses maximum variance to reduce the dimensionality of raw data, and afterward calculates DTW distance on the low-dimensional data for the within-subject case

Var_loglikelihood_between_subjects: uses maximum variance to reduce the dimensionality of raw data, and afterward calculates loglikelihood on the low-dimensional data for the between-subject case

Var_loglikelihood_within_subjects: uses maximum variance to reduce the dimensionality of raw data, and afterward calculates loglikelihood on the low-dimensional data for the within-subject case

PCA_Euc_between_subjects: uses PCA to reduce the dimensionality of raw data, and afterward calculates Euclidean distance on the low-dimensional data for the between-subject case

PCA_Euc_within_subjects: uses PCA to reduce the dimensionality of raw data, and afterward calculates Euclidean distance on the low-dimensional data for the within-subject case

PCA_Maha_between_subjects: uses PCA to reduce the dimensionality of raw data, and afterward calculates Mahalanobis distance on the low-dimensional data for the between-subject case

PCA_Maha_within_subjects: uses PCA to reduce the dimensionality of raw data, and afterward calculates Mahalanobis distance on the low-dimensional data for the within-subject case

PCA_DTW_between_subjects: uses PCA to reduce the dimensionality of raw data, and afterward calculates DTW distance on the low-dimensional data for the between-subject case

PCA_DTW_within_subjects: uses PCA to reduce the dimensionality of raw data, and afterward calculates DTW distance on the low-dimensional data for the within-subject case

PCA_loglikelihood_between_subjects: uses PCA to reduce the dimensionality of raw data, and afterward calculates loglikelihood on the low-dimensional data for the between-subject case

PCA_loglikelihood_within_subjects: uses PCA to reduce the dimensionality of raw data, and afterward calculates loglikelihood on the low-dimensional data for the within-subject case

En_Euc_between_subjects: uses autoencoder neural network to reduce the dimensionality of raw data, and afterward calculates Euclidean distance on the low-dimension data for the between-subject case

En_Euc_within_subjects: uses autoencoder neural network to reduce the dimensionality of raw data, and afterward calculates Euclidean distance on the low-dimension data for the within-subject case

En_Maha_between_subjects: uses autoencoder neural network to reduce the dimensionality of raw data, and afterward calculates Mahalanobis distance on the low-dimension data for the between-subject case

En_Maha_within_subjects: uses autoencoder neural network to reduce the dimensionality of raw data, and afterward calculates Mahalanobis distance on the low-dimension data for the within-subject case

En_ DTW_between_subjects: uses autoencoder neural network to reduce the dimensionality of raw data, and afterward calculates DTW distance on the low-dimensional data for the between-subject case

En_ DTW_within_subjects: uses autoencoder neural network to reduce the dimensionality of raw data, and afterward calculates DTW distance on the low-dimensional data for the within-subject case

En_Loglikelihood_between_subjects: uses autoencoder neural network to reduce the dimensionality of raw data, and afterward calculates loglikelihood on the low-dimension data for the between-subject case

En_Loglikelihood_within_subjects: uses autoencoder neural network to reduce the dimensionality of raw data, and afterward calculates loglikelihood on the low-dimension data for the within-subject case

## Utility Functions

EM_boundingCov: learns the parameters of a Gaussian Mixture Model (GMM) using a recursive Expectation-Maximization (EM) algorithm. After each EM step, the covariance matrices are bounded to avoid numerical instability

EM_init_regularTiming: initializes the parameters of a Gaussian Mixture Model (GMM) by using k-means clustering algorithm

gausPDF: computes the Probability Density Function (PDF) of a multivariate Gaussian represented by means and covariance matrix

loglik: computes the loglikelihood of a GMM model

## Data Augmentation (used in the v1 version of the article)

M1_Augmentation: generates new instances by adding random noise to the correct instances. The input data is “M1-DeepSquat” in the folder “Data for Distance Functions”.

## Data for Distance Functions

M1-DeepSquat-Correct: the original data performed correctly in the first exercise - Deep Squat

M1-DeepSquat-Incorrect: the original data performed incorrectly in the first exercise - Deep Squat

M1-Reduced-DeepSquat: obtained by performing dimensionality reduction with autoencoder neural networks to compress M1-DeepSquat

## Data for Neural Networks

M1_DeepSquat folder

Train_X1: the raw measurements of correct movements

Train_Y1: the corresponding quality scores for the correct movements

Test_X1: the raw measurements of incorrect movements

Test_Y1: the corresponding quality scores for the incorrect movements

M1_Aug_DeepSquat folder

X1_movement1: the raw measurements of correct movements

Y1_movement1: the corresponding quality scores for the correct movements

Xk_movement1 (k=2, 3, 4, 5): synthetically generated sequences for the correct movements

Yk_movement1 (k=2, 3, 4, 5): the corresponding quality scores for the synthetically generated sequences

X6_movement1: the raw measurements of incorrect movements

Y6_movement1: the corresponding quality scores for the incorrect movements