EMG Classificiation for Hand Gestures

ISyE 6740 - Summer 2020 - Final Report

Authors: Hayden Cornell, Numaer Zaker

1. Problem Statement

Our muscle movements and contractions generate currents that can be measured using a technique known as Electromyography (EMG). EMG has a wide variety of applications ranging from the medical industry, virtual reality, communication, and much more. A few practical examples are as follows:

- Virtual Reality - Simulating various hand gestures within virtual reality would be accomplishable by classifying various sEMG signals of different hand gestures
- Muscle Dysfunction - Identifying whether a patient has dysfunctional muscles on their arms to identify broader issues such as Dementia

Many times detecting certain hand gestures or even abnormalities can be difficult as they are often subtle or hard to measure. We tackle this issue in this report by using an sEMG dataset to try to classify a variety of hand gestures that are used on a day-to-day basis. We believe if we can successfully classify these gestures, our models could be used to detect abnormalities in patients or be used in virtual reality simulations.

2. Data Source

For this analysis, we use the **sEMG for Basic Hand Movements Data Set**. This is an open source dataset that is available here:

- https://archive.ics.uci.edu/ml/datasets/sEMG%20for%20Basic%20Hand%20movements# (https://archive.ics.uci.edu/ml/datasets/sEMG%20for%20Basic%20Hand%20movements#).

It contains two databases containing data of sEMG of various participants doing a variety of hand movements. There are six categories that these hand movements fall under:

- Spherical
- Tip
- Palmar
- Lateral
- Cylindrical
- Hook

Each row in the dataset represents a single trial of recording sEMG data for the participants. sEMG data is recorded over time, so each column will represent a point in time for each trial. For our analysis, we only focus on the first database which has 5 participants, 6 hand signals, and 12 sensors. Our dataset will be a matrix with 900 data points and 6000 features.

3. Methodology

Our methodology for building classification models will follow five high level steps:

1. Data preprocessing to eliminate nosie and smooth data poionts.
2. Dimensionality reduction with principal component analysis
3. Building various model classifiers with support vector machines, k-means, gaussian-mixture, naive-bayes, logistic regression, and neural networks
4. Hypertuning each model to select the best model via gridsearch cross validation
5. Evaluate model performance against a test set and compare results.

Data Processing and Transformation

Preprocessing & Smoothing

We found the EMG data to be very noisy and it resulted in poor model classification performance without processing. In order to improve the quality of the data we performanced two transformations:

- Apply the absolute value function to each datapoint to reduce standard deviation/volatility
- Apply Holt's smoothing to eliminate much of the noise in the data.

Dimensionality Reduction with PCA

Each datapoint has 6000 features where each feature represents a point in time. It's likely the case that many of these features are unlikely to improve the performance of our classification model. More specifically, many of the features will not explain the variance in the classification. Through principal component analysis (PCA), we chose the top components that explained most of the variance in our model

Model Formulation, Creation, & Tuning

Support Vector Machine Classifier

We learned from the class that support vector machines can be used to separate datapoints using a variety of kernels. We believed that SVM would be a good model to try for this problem given that we have 6 different classes with high dimensionality. We formulas the hand gesture classifcation problem for SVM as follows:

$$\min_{w,b} ||w||^2$$
$$\text{s.t.} y^i (w^T x^i + b) \geq 1, \forall i$$

In plain english, the formulation maximizes the soft margins between the 6 different hand gestures to minimize the overall training error. The above is the formulation for the linear kernel; but we try a variety of kernel as we suspect much of the features between the different hand gestures may overlap. A different kernel provides more flexibility to these noisy boundaries.

K-Means Clustering Classifier

Gaussian-Mixture Model

We built a Gaussian-Mixture model because with the processed data, we believe that each class of hand gestures could be presented by a unimodal distribution using the top principal components. Our gaussian mixture model for this problem was formulated as the following:

We initialized $\pi_k = 1/m$, $\mu_k = 0$ and $\sum_k$ to be the identify matrix. Then we run the expectation-maximization algorithm below until we maximize the likelihood (convergence)

Expectation Step

$$t_k^i = p(z_k^i = 1|D, \mu, \sigma) = \frac{\pi_k N(x_i|\mu_k, \sum_k)}{\sum_{k=1}^K \pi_k N(x_i|\mu_k, \sum_k)}$$

Maximization Step

Then once we finish the expectation step to get the new t_k^i , we update $(\pi_k, \mu_k, \sum_k)$ as follows:

$$\pi_k = \frac{\sum_i \tau_k^i}{m}$$
$$\mu_k = \frac{\sum_i \tau_k^i x^i}{\sigma_i \tau_k^i}$$
$$\sum_k = \frac{\sum_i \tau_k^i (x^i - \mu_k)(x^i - \mu_k)}{\sum_i -\mu_k}$$

Naive-Bayes Classifier

We built a Naive-Bayes classifier as well as another model of comparison. One of the assumptions of Naive-Bayes is that the predictors are all independent. Now we know this is clearly not the case with our dataset given that time series data is prone to autocorrelation, and hence each sequential feature is loosely related. We anticipated this model to perform the worst. We formulated the model using the general naive bayes classifer:

- Define the class priors: p(y), which is the likelihood of each hand gesture in the dataset
- Calculate the posterior probability of the training set using Bayes formula; more specifically given the features how often do they result in each of the classes:
  - $P(y = i|x) = \frac{P(x|y)P(y)}{P(x)}$
- Apply bayes decision rule where the class of the point would be the class with the highest posterior probability $P(y = i|x)$
- Maximize the likelihood of all the data points being the correct hand gesture

Multinomial Logistic Regression Classifier

Neural Network Classifier

4. Evaluation and Final Results

After hypertuning all of our models using gridsearch and cross validation, we chose the models that had the highest accuracies. Below are the results:

Model	Correctly Classified	Incorrectly Classified	Accuracy	Notes
SVM			52%	
K-Means			47%	
Gaussian-Mixture			58%	
Naive-Bayes			53%	
Logistic Regression				
Neural Network				

Out[2]: Click here to toggle on/off the raw code.