A heterogeneous clustering approach for Human Activity Recognition

Sabin Kafle

University of Oregon, Eugene, OR
skafle@cs.uoregon.edu

Abstract. Human Activity Recognition (HAR) has a growing research interest due to the widespread presence of motion sensors on user personal devices. The performance of HAR system deployed on large-scale is often significantly lower than reported due to the sensor-, device-, and person-specific heterogeneities. In this work, we develop a new approach for clustering such heterogeneous data, represented as a time series, which incorporates different level of heterogeneities in the data within the model. Our method is based on representing the heterogeneities as a hierarchy where each hierarchy denotes a specific heterogeneity (e.g. a sensor-specific heterogeneity). Experimental evaluation on an EMG sensor dataset with heterogeneities shows that our method performs favourably compared to other time series clustering approaches.

1 Introduction

The widespread availability of sensors in everyday lives enables us to capture contextual information from underlying human behavior in real-time. This has lead to the significant research focus on Human Activity Recognition (HAR) using sensor data [55]. Sensor data is used to determine the specific activity performed by the user at that instant, using either statistical or machine-learning approach. Despite a significant interest on HAR research, real-world performance variations across different sensors have been overlooked [55].

Another significant research problem based on use of sensor networks is development of automatic prosthetic limbs equipped with sensors; e.g. EMG and Accelerometer. The sensor network is used for detecting the intention of the user of the prosthetic limbs to provide a better control mechanism to the prosthetic limbs. The sensor network provides data related to the neural intent of the user, which is then interpreted by the prosthetic limb control mechanism as a signal for providing certain degree of freedom to the limb motion. For example, the control system is able to recognize whether the user is walking along a level ground or climbing up a stair based on the neural impulse of the user (inferred from sensor data using statistical and machine learning models), which then triggers an intent specific freedom on the prosthetic limbs; e.g. automated raising of the prosthetic limb when the user is climbing up a stair. While significant progress has been made in the development of prosthetic limbs with such control mechanisms [27], most of the work focus on having a prosthetic limb trained to a

specific user only. There is a distinct lack of research in unsupervised learning of user intent from such sensor data.

We focus on developing an unsupervised approach to recognize the user intent based on the sensor data. We treat the sensor data as a time series which is the most natural interpretation of such data. While there is a great amount of work done in time series clustering, most of them are inapplicable to our current problem. Time series clustering usually cluster the data obtained from same or similar data source, which is not true for our case. Moreover, most of the approaches require the number of clusters (or activity) in the data to be predetermined which is not always feasible or even possible in sensor data. Another challenge lies in the interpretation of the sensor data itself. The sensor data comprises of additive noises and have been found to be inefficient in representing the user intent as raw data themselves [54]. Time and frequency domain features are extracted from sensor data which are then used in machine-learning models for intent interpretation.

In this work, we address the challenges of performing unsupervised learning approach on sensor datasets. We first introduce the heterogeneities in the dataset as a hierarchy which each level in the hierarchy representing a specific heterogeneity. Next, we perform clustering using Bayesian semiparametric approach to mitigate the problem of pre-specifying the number of clusters in the dataset. Our approach learns the number of clusters (or activities) present in the dataset as a parameter of the model, which is capped by some large number that is considered to be an upper limit of possible number of clusters. Finally, we also develop a feature series clustering approach where we obtain features from the sensor dataset, which is then used to cluster the input data.

In the next section, we cover the background and related work, followed by our approach, experiments and results.

2 Background and Related Work

2.1 Dynamic Linear Model and Sampling Model

Dynamic Linear Models (DLMs) are a special case of general state space models, being linear and Gaussian. State space models [1] consider a time series as the output of a dynamic system perturbed by random noise. This allows a natural interpretation of a time series as the combination of several components such as trend, seasonal and regressive components. State space models are used to model multivariate time series also in presence of non-stationarity, structural changes and irregular patterns [49, Chapter 2]. One important class of state space models is given by Gaussian Linear state space models, also called Dynamic Linear Models [24].

Let $\boldsymbol{y_i} = \{y_{i,t} : t = 1, 2, 3, ...T\}$, $i = 1, 2, 3, ...n$ be a set of n time series, each of them observed during T time periods. A Dynamic Linear Model (DLM) describes each time series in terms of an observation (measurement) equation and an evolution or system equation:

$$y_{it} = F_{it}\theta_{i,t} + \epsilon_{i,t} \tag{1}$$

$$\theta_{it} = \rho\theta_{i,t-1} + \nu_{i,t} \tag{2}$$

where $\epsilon_{i,t}$ is the measure error given by $\epsilon_{i,t} \sim N(0, \sigma^2_{\epsilon_i})$ and $\nu_{i,t}$ is process error given by $\nu_{i,t} \sim N(0, \sigma^2_\theta)$ with independence across i and t. The evolution equation (2) describes a dynamic in the coefficients $\theta_{i,t}$ as an auto-regressive process of order 1 (i.e. an AR(1)). This has proved to be flexible enough representations for most time series [23].

We use the sampling model presented in [45] to separate the clustering and non-clustering parameters, which are interpreted as random variables to enable Bayesian treatment of DLMs.

$$\boldsymbol{y_i} = \boldsymbol{Z\alpha_i} + \boldsymbol{X\beta_i} + \boldsymbol{\theta_i} + \boldsymbol{\epsilon_i}, i = 1, 2, ..., n \tag{3}$$

$\mathbf{Z}$ and $\mathbf{X}$ are two design matrices of dimension $T \times p$ and $T \times d$ respectively. The $p \times 1$ dimensional vector α_i, the $d \times 1$ dimensional vector β_i and $T \times 1$ dimensional vector θ_i are the parameters of the model such that $\eta_i = (\alpha_i, \beta_i, \theta_i)$ but only α_i and β_i are considered for clustering. Finally, $\epsilon_i^{'} = (\epsilon_{i1}, \epsilon_{i2}, ..., \epsilon_{iT}) \sim N_T(0, \sigma^2_{\epsilon_i}\boldsymbol{I})$ is the vector of measurement error such that $\mathbf{I}$ is the identity matrix with dimension $T \times T$.

Here, α_i represents the non-clustering features of the time series (e.g. mean) while β_i represents the clustering features of the time series (e.g. polynomial trend). θ_i represents the dynamic behavior of the DLMs, which are also considered as a clustering parameter similar to [45].

The sampling model can also accommodate multi-dimensional time series data. For multi-dimensional time series, we consider each feature as having a time series of their own. For our particular purpose, this approach also enables us in reducing the additional overhead while an Majority Voting system for posterior inference makes the model more powerful. Then the sampling model is extended to form:

$$\boldsymbol{y_{i,f}} = \boldsymbol{Z\alpha_{i,f}} + \boldsymbol{X\beta_{i,f}} + \boldsymbol{\theta_{i,f}} + \boldsymbol{\epsilon_{i,f}}, i = 1, 2, ..., n, f = 1, 2..F \tag{4}$$

where F is the number of features. The design matrices Z, X can be same or different for each feature based on the data. Since we want the Sampling Model to incorporate trends in our observation equation, we consider the same design matrices for all the features.

2.2 Hierarchical Normal model

Many different kind of data, including observational data collected in human and biological sciences, have a hierarchical structure. For example, Electro Myography(EMG) signals have a natural hierarchy where the measurement of each person is grouped under an individual person and of each type of sensor is grouped under that particular sensor. This natural hierarchical tendency of data requires multi-level analysis, which can be incorporated using Hierarchical Normal Models (HNM). HNMs were first studied in the context of biological and human

sciences where family, race, geographical location introduces a natural hierarchy in the data [38] [11]. Significant work in efficient inference of multivariate HNMs is done in [15].

A hierarchy of normal distribution is considered in hierarchical normal model. The top-most hierarchy include a prior for mean and variance of the model (joint prior or distinct prior). A mean value is sampled from the prior, which is then used to sample different means for Level 1 hierarchies (represented by θ_i in the representation below), with the variance obtained from prior. For each sub-hierarchy in Level 2, the mean is sampled from each θ_i separately (y_i). The variance at each level can be either estimated from the data of that group and kept fixed or obtained from Gibbs sampler step for variance [15]. The HNMs are based on theory of exchangability with different groups at each group being exchangeable and irrelevant to the sampling order and sequence, except for hierarchy [38].

The posterior distribution for HNMs are obtained by either estimation with rejection sampling [15], substitution sampling and Gibbs sampling [2] or by using EM algorithms [6]. In this work, we consider the Gibbs Sampling approach due to the ease of integrating the sampler into the overall clustering approach. We also consider conjugate prior to the multivariate normal distribution with diagonal covariances. The Markov Chain Monte Carlo approach for inference in hierarchical normal models is often slower than compared to estimation with rejection sampling and EM approaches, but provides an easy integration to models where HNMs are used for introducing hierarchies only.

An example representation of Hierarchical Normal Model (HNM) is given in Figure 1. The existence of such hierarchies is the result of differentiation in all kind of activities (e.g. different gait events for different person and differing sensor metrics for different muscle activation). We use hierarchies for different person and sensor level variances with the sensor level variance being the first level and person level variance in the second level. We then use the samples from person level variance for a specific person-sensor combination which is inherent in the data.

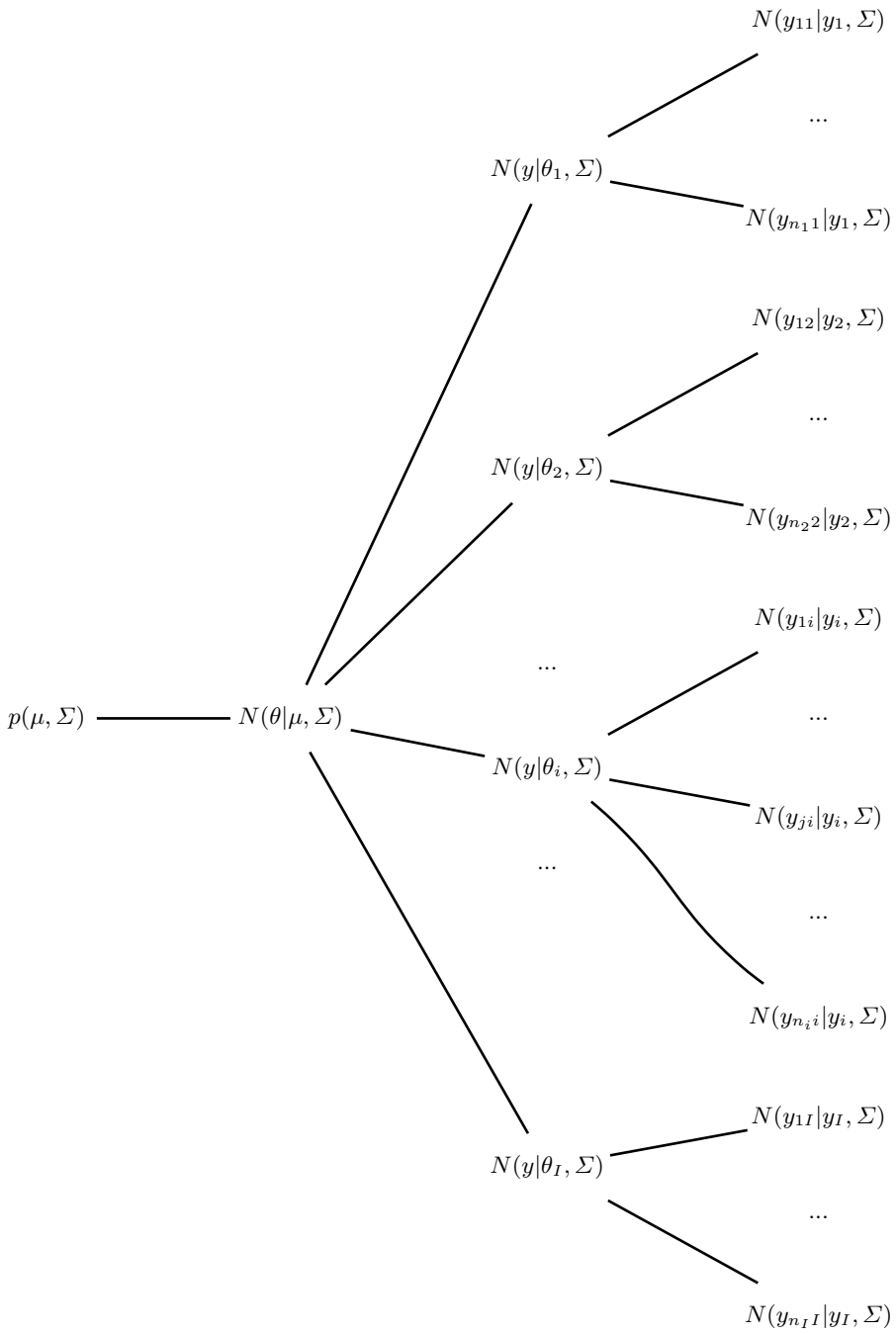


Fig. 1: A hierarchical Normal model.

2.3 Non parametric hierarchical models

Truncated non parametric hierarchical models are useful for introducing clustering parameters in model where the probability distribution for each cluster (multinomial parameters) is required. It is a generalization of Dirichlet process with truncation where Polya Urn characterization ([13] and [14]) is unknown. We fist describe the Polya Urn sampler for clustering, followed by a blocked Gibbs Sampler where Polya Urn Characterization is limited or unknown [29].

Stick Breaking Priors We represent a random probability measure P as

$$P(.) = \sum_{k=1}^{N} p_k \delta_{Z_k}(.) \tag{5}$$

where $\delta_{Z_k}(.)$ denote discrete measures concentrated at Z_k,
p_k are random variables (called random weights) independent of Z_k such that $0 \leq p_k \leq 1$ and $\sum_{k=1}^{N} p_k = 1$ almost surely.

It is assumed that Z_k are independent and identically distributed (i.i.d.) random elements with a distribution H. Stick breaking priors can be constructed can be constructed using either a finite or infinite number of terms.

The random weights are constructed by means of independent Beta random variables, with $p_1 = V_1$ and $p_k = (1 - V_1)(1 - V_2)....(1 - V_{k-1})V_k, k \geq 2$ where V_k are independent Beta variables $(Be(a, b))$ with $P = P_N(a, b)$ being random probability measure or stick breaking random measure. For finite dimensional measure, we set $V_N = 1$, where N is the number of measure.

If V_k are independent $Be(1, \alpha)$ variables with P_∞ being the random probability measure, then P_∞ is a Dirichlet process with concentration parameter $\alpha > 0$ and reference distribution H. It is also represented as $DP(\alpha H)$ [56].

The approach introduced by [29] is based on truncating P_∞ to P_N which provides a good approximation for P_∞ and is also possible to perform simple multivariate update to $p_1, p_2, ..., p_N$ in the Gibbs Sampler. We proceed to explain the approximation and consider $p(.)$ to be a Beta two-parameter process given as $Be(a, b, H)$, which for Dirichlet process is $P_\infty = B(1, \alpha, H) = DP(\alpha H)$.

Random Variables Description We recast the variables in Equation (5) completely in terms of random variables. Let $p = (p_1, p_2, ..., p_N)$ and $Z = (Z_1, Z_2, ..., Z_N)$. Then we can rewrite the above model as:

$$\begin{aligned} (X_i|Z, K) &\sim \pi(X_i|Z_{K_i}) \\ (K_i|p) &\sim \sum_{k=1}^{N} p_k \delta_k(.) \\ (p, Z) &\sim \pi(p)\pi(Z) \end{aligned} \tag{6}$$

where $K = (K_1, K_2, ..., K_N)$ and K_i are conditionally independent classification variables that identify the Z_k associated with Y_i, so that $Y_i = Z_{K_i}$

The Gibbs sampler implementation iteratively draw values from the conditional distributions of

$$\pi(p, Z|K, X) \tag{7}$$
$$\pi(K|p, Z, X) \tag{8}$$

We represent the random variable p as a Generalized Dirichlet Distribution (GDD). Z is calculated using multivariate normal distribution updates. We follow the adequate truncation value selection described in [29] to determine the upper limit of number of clusters.

2.4 EMG Signal

The neuromuscular system can be studied by measuring the electrical potential (signal) generated from a muscle contraction. This signal is a function of time and can be described in terms of amplitude, phase and frequency. The EMG signal (Electromyographic signal) measures electrical current generated in muscles during its contraction representing neuromuscular activity. The EMG signal is controlled by nervous system and is dependent on the anatomical and physiological property of muscles along with the noise it gathers while travelling through different tissues. Surface EMG detectors, placed on the surface of skin to gather muscle contraction signals, collects signals from different motor units at a time, which may generate interaction of different signals. The individual motor neurons and its muscle fibres are referred to as Motor Unit (MU) and the waveform generated by such motor units is called motor unit action potential (MUAP).

The detection and recordings of EMG signal are influenced by two main issues. The first one is the signal-to-noise ratio i.e. the ratio of energy in EMG signal to ratio of energy in the noise signal. Noise is defined as electrical signal that is not part of the desired EMG signal. The other is the distortion of signal, meaning the relative contribution of any frequency component in the EMG signal should not be altered.

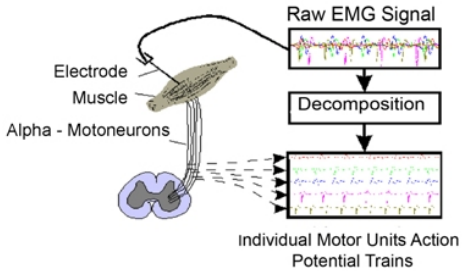


Fig. 2: EMG signal and decomposition of MUAPs [9].

EMG sensor is applied to the study of skeletal muscle, which is attached to the bone and is responsible for supporting and moving the skeleton. The contraction of skeletal muscle is initiated by impulses in the neuron to the muscle and is usually under voluntary control. Skeletal muscle fibres are well supported with neurons for contraction. These type of neurons are called motor neuron. In response to the stimulus from the neuron, a muscle fibre depolarizes as the signal propagates through the surface. This depolarization generates an electric field near each muscle fibre. An EMG signal is the train of MUAPs characterizing the muscle response to neural stimulation. Figure (2) shows the process of acquiring EMG signal.

2.5 Gait Event Detection

Gait Cycle A *gait* is defined as someone's manner of ambulation or locomotion, involving the total body [19]. The gait cycle is a repetitive pattern involving steps and strides [40]. A step time is the time from one foot hitting the floor to the other foot hitting the floor. A stride is a whole gait cycle. The sequences for walking could be summarized as:

1. Registration and activation of the gait command within the central nervous system
2. Transmission of the gait systems to peripheral nervous system
3. Contraction of muscles
4. Generation of several forces
5. Regulation of joint forces and movements across synovial joints and skeletal segments
6. Generation of ground reaction forces

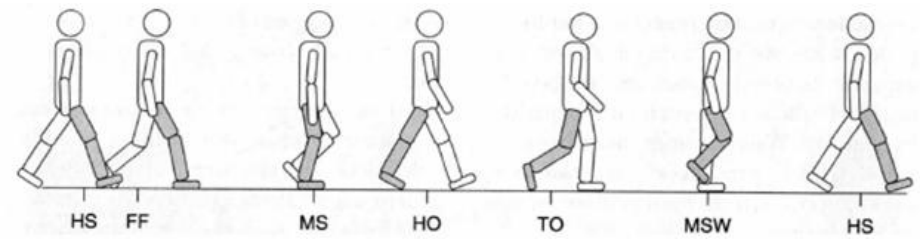


Fig. 3: Phases of gait cycle [59].

The two main phases of gait cycle are the stance phase and the swing phase. The stance phase occupies 60% of the gait cycle while the swing phase occupies only 40% of it. A more detailed classification of gait cycle recognizes six phases which are listed below and shown graphically in Figure (3).

1. Heel Strike (HS)

2. Flat Foot (FF)
3. Mid-stance (MS)
4. Heel Off (HO)
5. Toe Off (TO)
6. Mid Swing (MS)

Two phases of gait cycle have been found to be most efficient in recognizing locomotion mode. The first is Heel Strike (also called initial contact), a short period which begins the moment the foot touches the ground. The second phase is toe-off (also called pre-swing phase), a period when the toe begins to take stance.

Surface EMG for Gait event detection Surface EMG has been widely used for gait event detections. The application of gait event detection is in assisting amputees have automated control of prosthetic limbs as opposed to manually controlled limbs, which is both inconvenient and tiresome. The design of lower limb prostheses have offered the amputees patient improved stability and decrease in energy consumption in level ground walking ([51] and [32]). The advances in computerized control and powered prosthetics limb design have improved the function of artificial limbs; with these legs able to assist users with versatile action beyond the level walking. However, in order to properly select the correct control mode to adjust the joint impedance, the limbs should have a mechanism for knowing the user movement intent ([48] and [25]).

Surface EMG signals are one of the primary neural control sources for powered upper limb prostheses. While the use of EMG signals for upper limb prostheses has been prominent for decades ([61], [47], [28] and [12]), there has also been recent influx of work on using surface EMG signals for lowered limb prosthetics ([27], [26], [58], [21], [22] and [46]). It was demonstrated by [48] that there is a difference in EMG signal envelope among level-ground walking and descending and ascending a ramp, with conclusion that EMG signals from hip-muscles could be used to classify the locomotion modes. [31] presented an algorithm for terrain identification during walking. The features are extracted from EMG signals for a complete stride cycle, which are then used to make one decision per stride cycle. This led to further development of applications which are able to give real time decisions by integrating them into the prosthetic limb.

Most of the work in EMG signal based terrain identification (locomotion mode identification) is based on using classification algorithms, which depends on having training data with correct labels. The earlier work for EMG signal analysis is based upon wavelet analysis [35] and auto-regressive models [18]. The more recent approaches are based on using the features extracted from EMG signals which are then fed to a machine learning model in order to train a classifier. Support Vector Machine (SVM), Linear Discriminant Analysis and Markov Models are more prominent in recent works [41].

The muscles on which surface EMGs are placed for locomotion mode identification are usually based on the application and the extent of amputation.

In particular, the following muscle's EMG signal have been found to be most significant in better gait event detection

- Tibialis anterior muscle (near shin)
- Gastrocnemius muscle (back of calf)
- Rectus Femoris muscle (middle of the front of thigh)
- Vastus Lateralis muscle (thigh muscle)
- Biceps femoris muscle (posterior thigh muscle)
- Gluteus maximus muscle (hip muscle)
- Gluteus medius muscle (outer surface of pelvis)

The skeletal muscles used for detecting terrains are illustrated in Figure (4). The EMG signals by themselves are just random signal with zero mean, but have significance during stages where the muscle contraction is maximum (i.e. during the phase where the electrical impulse generated from MUs could be measured). In terrain detection using EMG signal data, two phases of gait cycle are most significant, namely Heel Strike and Toe Off. During these two phases, the MUAP potential is maximum and thus provides a vital information for classifying terrains based on the input signals. Moreover, due to the random nature of EMG signal themselves, even the EMG signal collected during the significant phases of gait cycle (heel strike and toe off) are unable to predict terrain with any reliable accuracy [26].

Features extracted from EMG signals The features extracted from EMG signals are crucial for getting proper classification accuracy during prediction. The features extracted during the $150ms$ phase before and after the *Heel Strike* and *Toe Off* is found to be most accurate for terrain event detection [27]. In this section, we describe the features that are most relevant to the terrain classification task using EMG signals given by [20] and [3]. Only the time-domain features are described since they are the easiest to compute and most relevant to most type of time series (e.g. stock market data and rainfall measurement).

- *Mean*
- *Variance*
- *Mean Trend*
- *Variance Trend*
- *Windowed Mean Difference*
- *Windowed Variance Difference*
- *Auto-regressive coefficient*

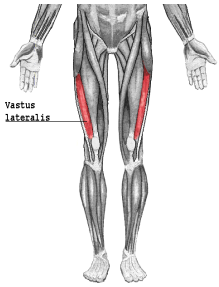
2.6 Time series clustering

Time series clustering is one of the most fundamental and complex task in data mining research. The summary of approaches for clustering of time series is shown in Figure (5). Time series clustering algorithms are usually applied by either converting the popular static clustering approaches to handle time series

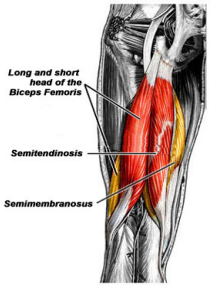
(a) Tibialis anterior muscle

(b) Gastrocnemius muscle

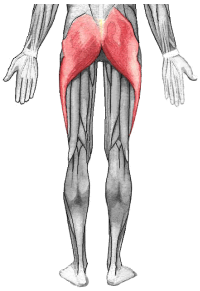
(c) Rectus Femoris muscle



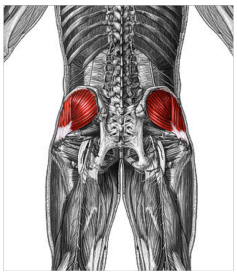
(d) Vastus Lateralis muscle



(e) Biceps femoris muscle



(f) Gluteus maximus muscle



(g) Gluteus medius muscle

Fig. 4: **Skeletal muscles used for terrain detection during locomotion.**

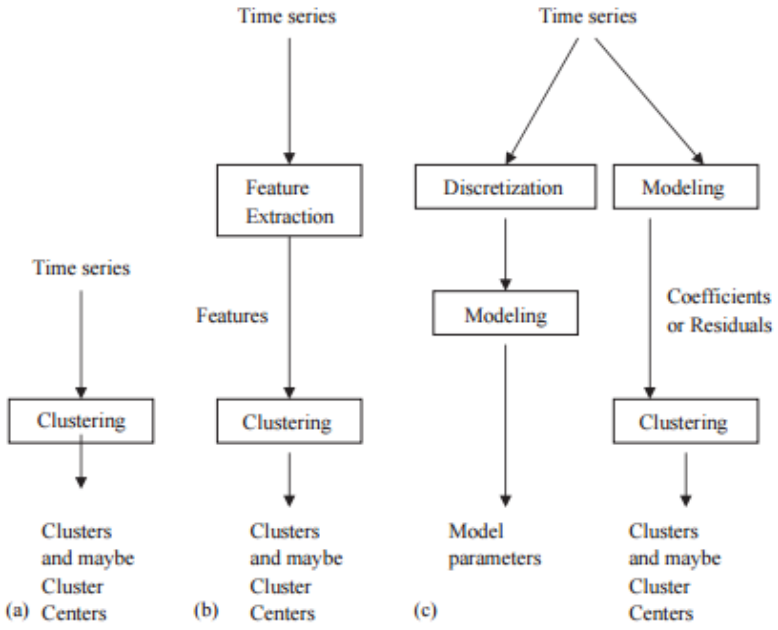


Fig. 5: **Time series clustering approaches [37].**

or by modifying the time series such that static clustering algorithms could be applied [37].

One of the most popular approach for static data clustering is k-means or k-mediods, which generate spherical-shaped cluster with a distance metric being considered for deciding cluster membership. Another popular approach for clustering is hierarchical clustering which generate clusters in agglomerative manner (assign each data as individual cluster and proceed with merging to generate ideal cluster) or divisive manner (partition the data based on some metric). This approach requires cluster quality check metric to determine the best cluster partitions. Density-based clustering approaches grow a cluster as long as the density of the "neighbourhood" exceeds some threshold. Model-based clustering approaches assume a model for each cluster and attempt to best fit the data to the assumed model.

The most used approach for clustering time series data is based on computing the similarity measure between different time series and then using the similarity measure to obtain either a spherical cluster partitions using *k-means* algorithm or a non-spherical cluster partitions using *fuzzy k-means*. Another popular approach is to extract features from time series and then use those features to perform clustering, either by using a multinomial distribution (when the number of clusters is known *apriori*) or a Dirichlet Process (when the number of clusters is not known apriori). The recent focus is on model-based clustering [53] and Markov Chain Monte Carlo approaches to generate clusters [10]. Us-

ing latent variables to determine the cluster parameters is also another popular approach ([50] and [45]). However, there is a distinct lack of approaches which consider clustering among heterogeneous data sources, unless we consider Hierarchical Dirichlet Process [57] and its extensions, even these are rarely used for actual time series data.

We now describe some of the similarity measures which are used for computing the similarity/dissimilarity between different time series, the measure which are then applied to clustering algorithms. *Please refer to [43] for more details*

- *Autocorrelation based distances (ACF):* The autocorrelation distance is obtained by calculating the difference between two time series represented by their estimated autocorrelation vectors [16].
- *Periodogram-based distances (PER):* This approach is based on calculation of periodogram of time series and then calculating the distance between two time series using the measure.
- *Normalized Compression Distance (NCD):* [36] It is one of the compression based dissimilarity measure where the dissimilarity is calculated using compression distance.
- *Euclidean Distance (EUCL):* It is the simplest and most primitive dissimilarity measure, but is surprisingly robust and performs well on generic time series data [33].
- *Compression-based dissimilarity measure (CDM):* It is another of compression based dissimilarity measure [36].
- *Dynamic Time Warping (DTW) measure:* It is one of the most popular similarity measure in recent time series literature [52]. It calculates the optimal match between two given sequences with certain restrictions. Due to the exponential time complexity of DTW approaches, several alternatives have been proposed, including [34] which is able to reduce the complexity significantly without compromising the performance.
- *Discrete Wavelet Transform (DWT) measure:* This similarity measure is calculated by performing an unsupervised feature extraction using orthogonal wavelets on the series. The distance is then calculated as Euclidean Distance between wavelet approximations [39].
- *Correlation based dissimilarity(COR):* This approach calculates the dissimilarity between two time series using estimated Pearson's correlation between them [17].
- *Autocorrelation based dissimilarity (PACF):* This approach computes the similarity between two time series as the distance between their estimated **partial autocorrelation coefficients**, in a very similar way to ACF measure [7].
- *Complexity Invariant Distance (CID) measure:* This measure computes the distance based on Euclidean distance corrected by the complexity estimation of the series [4].
- *Permutation Distribution Clustering (PDC):* PDC represents an alternative complexity-based approach to clustering time series with dissimilarity between series described in terms of divergence between permu-

tation of distributions of order patterns in m-embedding of the original series [5].

3 Our approach

In this section, we describe our method for performing heterogeneous time series clustering.

3.1 Model Specification

We represent each time series as a sampling model [45].

$$\boldsymbol{y_i} = \boldsymbol{Z\alpha_i} + \boldsymbol{X\beta_i} + \boldsymbol{\theta_i} + \boldsymbol{\epsilon_i}, i = 1, 2, ..., n \tag{9}$$

Here, y_i is a $T \times 1$ dimensional time series and ϵ_i is the random noise of same dimension. A multi-variate time series of dimension $T \times f$ can be represented by considering each feature (column, if the time factor is represented row-wise) to be independent from one another and each column being represented as a single one-dimensional time series. We will cover more on that later.

The three parameters given in Equation (9) are used to model three different components of time series, namely

- α_i is $p \times 1$ dimensional vector representing the non-clustering parameter of the time series. It is used to enhance the fit of the model with respect to time series magnitude.
- β_i is the $d \times 1$ dimensional vector representing the clustering but non-autoregressive aspect of time series. It can be interpreted as the representation of latent features of time series that are not related to the auto-regressive aspect of time series
- θ_i is the $T \times 1$ dimensional vector representing the $AR(1)$ aspect of the time series assuming stationarity in the time series.

The matrices Z and X are design matrices used to represent temporal components of time series. Z is $T \times p$ dimensional matrix, while X is $T \times d$ dimensional. The clustering parameters of the model are θ_i and β_i which is represented jointly as γ_i. Also $\epsilon_i^{'} \sim N_T(0, \sigma_{\epsilon_i}^2 I)$, which is the noise in the data with I being an identity matrix of dimension $T \times T$.

The hyper parameters for $\alpha_i \sim N_P(0, \Sigma_\alpha)$, with $\Sigma_\alpha = diag(\sigma_{\alpha_1}^2, ..., \sigma_{\alpha_p}^2)$ and $\sigma_{\alpha_i}^2 \sim IGa(c_0^\alpha, c_1^\alpha)$ where IGa represents the Inverse-Gamma distribution. Also $\sigma_{\epsilon_i}^2 \sim IGa(c_0^\epsilon, c_1^\epsilon)$ for ϵ_i.

Hierarchical heterogeneity For clustering parameters β_i, θ_i (referred to as γ_i jointly in this section, where $\gamma_i = \beta_i \times \theta_i$ to obtain a $(p + d) \times 1$ dimensional vector), a hierarchy is used in our approach to address the heterogeneity in the data. We consider a hierarchical normal model for simplicity purpose.

At the top-most level (root) of hierarchy,

$$\gamma_{root} \sim N(0, \Sigma_\beta) \times N(0, \Sigma_\theta) \tag{10}$$

Here, $\Sigma_\beta = diag(\sigma^2_{\beta_1}, ..., \sigma^2_{\beta_d})$ and $\sigma^2_{\beta_j} \sim IGa(c^\beta_0, c^\beta_1)$ where IGa represents the Inverse-Gamma distribution as hyper parameter. And $\Sigma_\theta = (R_{jk})$ with each $R_{jk} = \sigma^2_\theta \rho^{|j-k|}$.

We consider the hyper prior for ρ and σ_θ jointly with distribution which maximizes the power of data to represent the best value [42].

$$f(\sigma^2_\theta, \rho) \propto (\sigma^2_\theta)^{-1} \frac{\sqrt{1+\rho^2}}{1-\rho^2} \tag{11}$$

where $\sigma^2_\theta > 0$ and $\rho \in (-1, 1)$

We have a separate Hierarchical Normal tree for each cluster. The number of clusters is determined semi-parametrically using the **Adequate truncation values** approach for Beta two-parameter variable. We use two-level hierarchy, with each level having mean as a sample from the level immediately above and covariances specific to that level.

The top-most level of the hierarchy for cluster k is

$$\gamma_k \sim N(0, \Sigma_{\beta,k}) \times N(0, \Sigma_{\theta,k}) \tag{12}$$

where γ_k represents the clustering random variable for cluster k and level 0 (or root). $\Sigma_{\beta,k}$ represents the covariances for β parameter for cluster k and level 0 and $\Sigma_{\theta,k}$ represents the covariances for θ parameter for cluster k and level 0.

For the first level of hierarchy, let us assume there are R branches, then for a branch r of cluster k at level 1:

$$\begin{aligned} \gamma_{r,k} \sim N(\overline{\beta_{r,k}}, \Sigma_{\beta,r,k}) \times N(\overline{\theta_{r,k}}, \Sigma_{\theta,r,k}) \\ \overline{\beta_{r,k}} \sim N(0, \Sigma_{\beta,k}) \\ \overline{\theta_{r,k}} \sim N(0, \Sigma_{\theta,k}) \end{aligned} \tag{13}$$

where $\Sigma_{\beta,r,k}$ and $\Sigma_{\theta,r,k}$ are covariances for β and θ parameters of cluster k, level 1 and branch r.

For level 2, we proceed in similar manner. For each branch r in level 1, there are S branches at level 2, which are obtained in similar manner to previous level for a branch s

$$\begin{aligned} \gamma_{s,r,k} \sim N(\overline{\beta_{s,r,k}}, \Sigma_{\beta,s,r,k}) \times N(\overline{\theta_{s,r,k}}, \Sigma_{\theta,s,r,k}) \\ \overline{\beta_{s,r,k}} \sim N(\overline{\beta_{r,k}}, \Sigma_{\beta,r,k}) \\ \overline{\theta_{s,r,k}} \sim N(\overline{\theta_{r,k}}, \Sigma_{\theta,r,k}) \end{aligned} \tag{14}$$

Here, k represents a specific hierarchical model among K such models, r represents the first level selection of the model among R such groups and s represents the second level selection of the model among S groups. The number

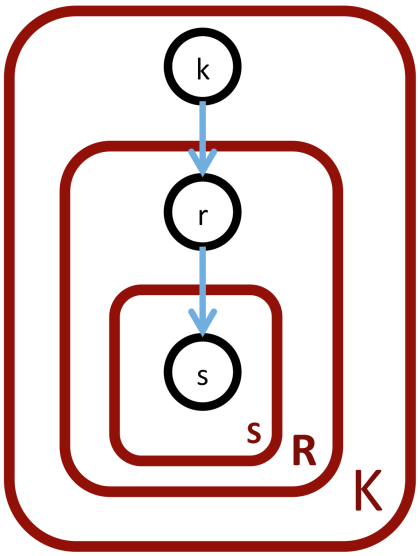


Fig. 6: **Block HNMs for heterogeneity.**

of variables at the subscript represents the level of the specific parameter. The block diagram of HNM given in Figure (6) for more clarity.

The hyper parameters for all Σ_x's in the hierarchical model is represented as $diag(\sigma^2_{x1}, ..., \sigma^2_{xy})$, where $y = D$ if $x \in \beta$ or $y = T$ if $x \in \theta$ and $\sigma^2_{xi} \sim IGa(c^x_0, c^x_1)$.

The sampling model explained previously then can be explained more clearly as

$$\begin{aligned} \boldsymbol{f(y_i) \propto N_T(Z\alpha_i + X\beta_i + \theta_i, \sigma^2_{\epsilon_i} I)} \\ \alpha_i \sim N(0, \Sigma_\alpha) \\ \beta_i \sim N(\overline{\beta_{s,r,k}}, \Sigma_{\beta,s,r,k}) \\ \theta_i \sim N(\overline{\theta_{s,r,k}}, \Sigma_{\theta,s,r,k}) \end{aligned} \tag{15}$$

where s, r and k are the level 2, level 1 and cluster value for series $y_i, i = 1, .., T$ respectively.

Clustering Parameter The hierarchies for the data are obtained from the data heterogeneity, while the selection of a particular k for y_i is based upon Generalised Dirichlet Distribution (GDD).

The probability are obtained by means of truncated stick-breaking process

$$P_N(.) = V_1 \delta_{Z_1}(.) + \sum_{k=2}^{N} \{(1 - V_1)(1 - V_2)...(1 - V_{k-1})V_k\} \delta_{Z_k}(.) \tag{16}$$

where

$$p_1 = V_1, p_k = (1 - V_1)(1 - V_2)...(1 - V_{k-1})V_k (k = 2, ..., N) \tag{17}$$

and V_k are independent $Be(a_k, b_k)$ random variables with $a_k = a$ and $b_k = b$, for $k \leq N-1$. V_N is set to 1 to ensure $p_1 + ... + p_N = 1$.

GDD has the distribution of

$$p \sim G(a_1, b_1, ..., a_{N-1}, b_{N-1}) \tag{18}$$

Its density is equal to ([8]):

$$\begin{aligned} &\{\prod_{k=1}^{N-1} \frac{\Gamma(a_k + b_k)}{\Gamma(a_k)\Gamma(b_k)}\} p_1^{a-1} ... p_{N-1}^{a_{N-1}-1} p_n^{b_{N-1}-1} \\ &\times (1 - P_1)^{b_1 - (a_2 + b_2)} .... (1 - P_{N-2})^{b_{N-2} - (a_{N-1} + b_{N-1})}) \end{aligned} \tag{19}$$

where $P_k = p_1 + ... + p_k$. It can be seen that this distribution is conjugate with multinomial sampling with $a_k = a$ and $b_k = b$ is $G(a_1^*, b_1^*, ..., a_{N-1}^*, b_{N-1}^*)$, where

$$\begin{aligned} a_k^* &= a + m_k \\ b_k^* = b + \sum_{j=k+1}^{N} m_j &= b + M_k (k = 1, 2, ..., N-1) \end{aligned} \tag{20}$$

and m_k is the number of K_i's which equal k.

Here a_i and b_i are the hyper parameters of GDD and p_i is the probability distribution of the model which can be updated easily using conjugacy in case of multivariate normal distribution which is true for our case. This completes the description of the model.

3.2 Posterior Characterization

The likelihood function is

$$\boldsymbol{f(y) = \prod_{i=1}^{N} N_T(Z\alpha_i + x\beta_i + \theta_i, \Sigma_y)} \tag{21}$$

where $\Sigma_y = \sigma_\epsilon^2 I$.

The conditional distribution for all the parameters used in the model can be obtained analytically. The posteriors are further updated in same order as given below.

- α_i: The posterior for α_i is

$$\begin{aligned} f(\alpha_i | rest) &\propto N_P(\mu_a, \Sigma_a) \\ \Sigma_a &= (\Sigma_\alpha^{-1} + Z^T \Sigma_y^{-1} Z)^{-1} \\ \mu_a &= \Sigma_a Z^T \Sigma_y^{-1} (y_i - X\beta_i - \theta_i) \\ f(\sigma_{\alpha_j}^2 | rest) &= IGa(c_0^\alpha + \frac{n}{2}, c_1^\alpha + \frac{1}{2} \sum_{i=1}^{n} \alpha_{ij}^2), j = 1, .., p \end{aligned} \tag{22}$$

- β_i: The posterior for β_i (or $\beta_{s,r,k}$) is

$$\begin{aligned} f(\beta_i|rest) &\propto N_D(\mu_b, \Sigma_b) \\ \Sigma_b &= (\Sigma^{-1}_{\beta,s,r,k} + X^T\Sigma_y^{-1}X)^{-1} \\ \mu_b &= \Sigma_b[X^T\Sigma_y^{-1}(y_i - Z\alpha_i - \theta_i) + \Sigma_{\beta,s,r,k}\overline{\beta_{s,r,k}}] \\ f(\sigma^2_{\beta_{s,r,k,i}}|rest) &= IGa(c_0^{\beta_{s,r,k,i}} + \frac{m}{2}, c_1^{\beta_{s,r,k,i}} + \frac{1}{2}\sum_{j=1}^{m}\beta^2_{s,r,k,i}) \\ & i = 1, .., p \end{aligned} \tag{23}$$

  where m is the number of data points belonging to that cluster.
- θ_i: The posterior for θ_i (or $\theta_{s,r,k}$) is

$$\begin{aligned} f(\theta_i|rest) &\propto N_T(\mu_c, \Sigma_c) \\ \Sigma_c &= (\Sigma^{-1}_{\theta,s,r,k} + \Sigma_y^{-1})^{-1} \\ \mu_c &= \Sigma_c[\Sigma_y^{-1}(y_i - Z\alpha_i - X\beta_i) + \Sigma_{\theta,s,r,k}\overline{\theta_{s,r,k}}] \\ f(\sigma^2_{\theta_{s,r,k,i}}|rest) &= IGa(c_0^{\theta_{s,r,k,i}} + \frac{m}{2}, c_1^{\theta_{s,r,k,i}} + \frac{1}{2}\sum_{j=1}^{m}\theta^2_{s,r,k,i}) \\ & i = 1, .., T \end{aligned} \tag{24}$$

  where m is the number of data points belonging to that cluster.
- $\sigma^2_{\epsilon_i}$: The posterior for $\sigma^2_{\epsilon_i}$ is

$$\begin{aligned} f(\sigma^2_{\epsilon_i}|rest) &\propto IGa(c_0^{\epsilon} + \frac{T}{2}, c_1^{\epsilon} + \frac{1}{2}M_i^{'}M_i) \\ M_i &= (y_i - Z\alpha_i - X\beta_i - \theta_i) \end{aligned} \tag{25}$$

- *Level 1 posterior*: The posterior for level 1 of hierarchy is

$$\begin{aligned} f(\beta_{r,k}|rest) &\propto N_D(\mu_x, \Sigma_x) \\ \Sigma_x &= (\Sigma^{-1}_{\beta,s,r,k} + \Sigma^{-1}_{\beta,r,k})^{-1} \\ \mu_x &= \Sigma_x(\Sigma_{\beta,r,k}\overline{\beta_{r,k}} + \Sigma^{-1}_{\beta,s,r,k}\beta_{s,r,k}) \\ f(\sigma^2_{\beta_{r,k,i}}|rest) &= IGa(c_0^{\beta_{r,k,i}}\frac{S}{2}, c_1^{\beta_{r,k,i}}\sum_{j=1}^{S}\beta^2_{r,k,i}) \\ f(\theta_{r,k}|rest) &\propto N_T(\mu_y, \Sigma_y) \\ \Sigma_y &= (\Sigma^{-1}_{\theta,s,r,k} + \Sigma^{-1}_{\theta,r,k})^{-1} \\ \mu_y &= \Sigma_y(\Sigma_{\theta,r,k}\overline{\theta_{r,k}} + \Sigma^{-1}_{\theta,s,r,k}\beta_{s,r,k}) \\ f(\sigma^2_{\theta_{r,k,i}}|rest) &= IGa(c_0^{\theta_{r,k,i}}\frac{S}{2}, c_1^{\theta_{r,k,i}}\sum_{j=1}^{S}\theta^2_{r,k,i}) \end{aligned} \tag{26}$$

- *k level posterior*: The posterior for k level of hierarchy is

$$\begin{aligned}
f(\beta_k|rest) &\propto N_D(\mu_g, \Sigma_g) \\
\Sigma_g &= (\Sigma_{\beta,r,k}^{-1} + \Sigma_{\beta,k}^{-1})^{-1} \\
\mu_g &= \Sigma_g(\Sigma_{\beta,k}\overline{\beta_k} + \Sigma_{\beta,r,k}^{-1}\beta_{r,k}) \\
f(\sigma^2_{\beta_{k,i}}|rest) &= IGa(c_0^{\beta_{k,i}}\frac{R}{2}, c_1^{\beta_{k,i}}\sum_{j=1}^{S}\beta_{k,i}^2) \\
f(\theta_k|rest) &\propto N_D(\mu_h, \Sigma_h) \\
\Sigma_h &= (\Sigma_{\theta,r,k}^{-1} + \Sigma_{\theta,k}^{-1})^{-1} \\
\mu_h &= \Sigma_h(\Sigma_{\theta,k}\overline{\theta_k} + \Sigma_{\theta,r,k}^{-1}\theta_{r,k}) \\
f(\sigma^2_{\theta_{k,i}}|rest) &= IGa(c_0^{\theta_{k,i}}\frac{R}{2}, c_1^{\theta_{k,i}}\sum_{j=1}^{R}\beta_{k,i}^2)
\end{aligned} \tag{27}$$

- *Top level posterior*: The posterior at the top-most level is

$$\begin{aligned}
f(\beta|rest) &\propto N_D(\mu_e, \Sigma_e) \\
\Sigma_e &= (\Sigma_{\beta,k}^{-1} + \Sigma_{\beta}^{-1})^{-1} \\
\mu_e &= \Sigma_e(\Sigma_{\beta,k}^{-1}\beta_k) \\
f(\sigma^2_{\beta_i}|rest) &= IGa(c_0^{\beta_i}\frac{K}{2}, c_1^{\beta_i}\sum_{j=1}^{K}\beta_i^2) \\
f(\theta|rest) &\propto N_D(\mu_f, \Sigma_f) \\
\Sigma_f &= (\Sigma_{\theta,k}^{-1} + \Sigma_{\theta}^{-1})^{-1} \\
\mu_f &= \Sigma_f(\Sigma_{\theta,k}^{-1}\theta_k) \\
f(\sigma^2_{\theta}|rest) &= IGa(\frac{KT}{2}, \frac{1}{2}\sum_{j=1}^{K}\theta_j^{'}Q^{-1}\theta_j) \\
f(\rho|rest) &\propto |Q|^{-K/2}\exp\frac{-1}{2\sigma_\theta^2}\sum_{j=1}^{K}\theta_j^{'}Q^{-1}\theta_j\frac{\sqrt{1+\rho^2}}{1-\rho^2}
\end{aligned} \tag{28}$$

  where $Q_{ij} = \rho^{|i-j|}$ for $i, j = 1, .., T$.
- *Posterior for GDD and p*: The posterior for GDD is conjugate with multinomial sampling and is obtained as explained in Section (3.1). The probability p is updated based on the fit of the data with respect to the individual clusters lowest level mean using the likelihood function [29].

3.3 Model Learning and Majority Voting

Gibbs Sampling algorithm is used for posterior inference, with Metropolis within Gibbs sampler being used for sampling ρ and σ_θ^2. The Gibbs Sampler algorithm

used is same as given in [30] except for sampling from the hierarchical model and Metropolis steps for θ parameters. The hierarchical model posterior sampling is done in bottom-top approach. The hierarchy is sampled beginning from the Sampling model until the top level is reached.

In order to handle time series where the signal themselves are not significant but the set of features extracted from them are significant in both clustering and classification task, we extend our approach to handle such cases. We assume that each feature series is independent of one another in order to reduce the complexity of the model. Then each feature is considered as an independent time series and the above model is applied to all the features.

The most significant aspect during handling such multiple features is to consider how each feature series affect the overall clustering aspect of the model. We propose two approach for such cases:

- The Generalized Dirichlet Distribution is used for combining different feature series. A single cluster label is selected for all the feature series. The probability distribution p is updated with each feature series likelihood w.r.t. the data. Rest of the model is kept same as explained above.
- Different Generalized Dirichlet Distribution is used for each feature series. The final cluster membership is based on majority voting used for determining cluster membership.

The problem of non exchangeability in second approach is handled by considering the mean of each cluster for different feature series and assigning one cluster as reciprocal to another cluster based on the similarity (Euclidean distance) and the updating the index label before performing Majority Voting.

3.4 Cluster Selection

Each iteration of Gibbs Sampling produces a cluster assignment among the data, which is then filtered using selection criteria to select one cluster assignment as the best fit. One way of selecting a cluster membership used by [45] is Heterogeneity Measure (HM).

$$HM(G_1, .., G_m) = \sum_{k=1}^{m} \frac{2}{n_k - 1} \sum_{i<j \in G_k} \sum_{t=1}^{T} (y_{it} - y_{jt})^2 \tag{29}$$

The larger the value of HM, the more heterogeneous a clustering is. It is preferable to have a cluster with small HM and small m.

4 Experiments and Results

4.1 Data

As an application for our method, we apply the proposed method to Electromyography (EMG) signals collected from 7 sensors placed in different muscles and

12 human subject. The EMG signals are collected while the subject walks on different terrains (level ground, stair ascent/descent, ramp ascent/descent). The data is naturally heterogeneous with each sensor being considered the first level of hierarchy and person-wise differentiation being considered the second level of hierarchy. This is done due to sensor wise differences being much more prominent than person-wise differences (in our data).

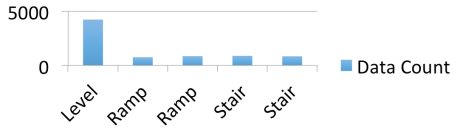


Fig. 7: **Each class count of data.**

There are altogether 9450 records of gait cycles as time series data. The data count for each type of terrain where gait events are happening is given in Figure (7).

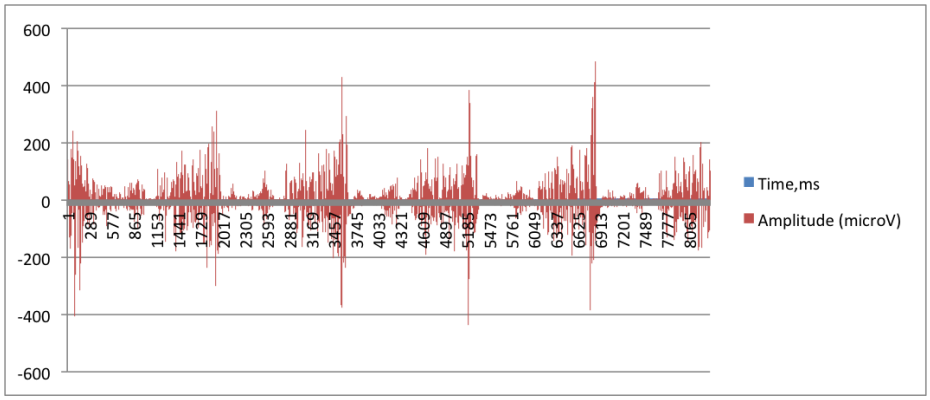


Fig. 8: **EMG sample with 4 gait cycles**

An EMG sample with 4 gait cycles is shown in Figure (8). In order to apply our method to the data, we split the EMG signals into each gait cycles. Each such gait cycle consists of a label for terrain on which the person is moving when the data is collected. This enables us to compare and contrast our method with other time series clustering algorithms.

For efficient sampling, We compress each gait cycle into a time series of length $T = 23$. The compression was performed using peak amplitude values for non-feature series, while the individual features were extracted for feature series. Another reason for compression is to bring consistency in the length of

each gait cycle. This is achieved by analysing the best split points for each time series. The signal is then reduced to zero mean, unit standard deviation in order to maintain consistency among amplitude values. For experiment, we consider two different data set extracted from the original data set, (1). Peak amplitude based non-feature series (2). Multi-dimensional feature series with all features given in Section (2.5) being considered.

4.2 Experiments

We conduct several experiments with different configuration for the number of hyper parameters in order to determine the best configuration for the data. The number of clusters is determined initially using the **Adequate Truncation value** measure during the initial Gibbs Sampling phase. We found out 15 is sufficient number of clusters for this data.

The design matrices $Z_{T\times p}$ and $X_{T\times d}$ play a significant role for incorporating additional information into the model. We experimented with different settings for value of p with 1 and 2 being considered, while all the cells in the matrix has value 1. For X, $d = 7$ with first three columns representing the polynomial trend of degree 3, with remaining four columns being used as a latent trait indicator for four gait phases (Before Heel Strike, After Heel Strike, Before Toe Off, After Toe Off). The specific results selected below is based on the heterogeneity score of sampled cluster membership. The Heterogeneity Score for every results obtained is between 0.5 and 1.95, with random impact on the accuracy of the clustering.

It is important to investigate whether having a hierarchy in the method actually helps in getting better cluster or not. We conduct experiments where we consider our model without any hierarchies (the posterior are modified accordingly when needed). For all the experiments, we run Gibbs sampler upto 5000 iterations, with 3000 as burn-in phase and collect a sample every 200 iterations after the burn-in phase.

Evaluation on peak amplitude series We run a Gibbs Sampler for peak amplitude series, with number of clusters being set to 5. This is being done to determine whether EMG signal itself is informative to give any sort of clustering information. The rest of cluster parameters is given in table below.

Parameter	**Configuration**
Hierarchy	Yes
Dimension of Z matrix	$T \times 1$
Number of clusters	*5*
Features Used	No
Majority Voting	Not applicable
Inverse-Gamma Prior	$[2, 1]$
Heterogeneity Score	1.95

The confusion matrix is given in Table (1). As is evident from the confusion matrix, the model prefers a single cluster. This is due to two factors:

- The EMG signal is mostly a random noise and feature extraction plays a important role in making use of EMG signal.
- The data is unbalanced with one class (Level Ground) having more data than other class data combined, which skews the model, greatly.

In order to combat the above mentioned issues, we consider the multi-dimensional feature series for clustering along with sub-sampling approach to reduce the data imbalance.

Table 1: Confusion matrix using Peak Amplitude data

	Clusters generated by Gibbs Sampler (Membership Percentage)				
	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Level Ground	5.6	**76.3**	9.17	5.5	3.43
Ramp Ascent	4.5	**80.1**	6.7	5.2	3.3
Ramp Descent	6.0	**76.8**	9.1	5.2	2.9
Stair Ascent	5.4	**77.7**	8.39	5.1	3.2
Stair Descent	5.8	**74.5**	10.8	5.6	3.2

Evaluation on Feature Series We use the feature series generated using the raw EMG data for clustering the series. In this section, we conduct experiment to determine whether having hierarchy helps in clustering or not. We also compare the performance of majority voting based feature clustering with single GDD parameter based feature clustering.

Hierarchy vs Without Hierarchy

The configuration for two experiments are given below

Parameter	**Experiment 1**	**Experiment 2**
Hierarchy	Yes	No
Dimension of Z matrix	$T \times 1$	$T \times 1$
Number of clusters	*15*	*15*
Features Used	Yes	Yes
Majority Voting	Yes	Yes
Inverse-Gamma Prior	$[2, 1]$	$[2, 1]$
Heterogeneity Score	0.735	0.645

The only change in the above two experiments configuration is the presence and absence of hierarchy. We take a look at the confusion matrix for both experiments.

The confusion matrix for Experiment 1 is given in Table (2) while the confusion matrix for Experiment 2 is given in Table (3). It is evident that introducing hierarchy improves the classification accuracy.

Table 2: Confusion matrix for Experiment 1

	Clusters generated By Gibbs Sampler (Membership Percentage)														
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15
Level Ground	0.0	**33.69**	22.8	15.54	10.77	6.9	3.93	2.38	2.2	1.13	0.6	0.06	0.0	0.0	0.0
Ramp Ascent	0.0	**35.05**	23.41	15.61	9.26	6.88	3.31	2.91	1.85	0.93	0.79	0.0	0.0	0.0	0.0
Ramp Descent	0.0	**35.01**	24.36	14.4	11.12	6.67	4.22	0.94	1.64	1.05	0.47	0.12	0.0	0.0	0.0
Stair Ascent	0.0	23.02	16.67	10.2	**35.49**	6.58	3.51	1.81	1.7	0.79	0.11	0.11	0.0	0.0	0.0
Stair Descent	0.0	23.57	16.9	11.19	**32.86**	6.67	3.33	2.14	1.67	0.71	0.48	0.36	0.0	0.12	0.0

Table 3: Confusion matrix for Experiment 2

	Clusters generated By Gibbs Sampler (Membership Percentage)														
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15
Level Ground	0.0	**50.0**	31.58	7.89	5.26	5.26	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ramp Ascent	0.0	**70.0**	10.0	20.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ramp Descent	0.0	**50.0**	33.33	0.0	16.67	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Stair Ascent	0.0	**70.83**	16.67	8.33	4.17	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Stair Descent	0.0	**73.33**	23.33	3.33	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

We repeat the same set of experiments with exact same difference once again except without the majority voting for combining features. The configuration is as follow:

Parameter	**Experiment 3**	**Experiment 4**
Hierarchy	Yes	No
Dimension of Z matrix	$T \times 1$	$T \times 1$
Number of clusters	*15*	*15*
Features Used	Yes	Yes
Majority Voting	No	No
Inverse-Gamma Prior	$[2, 1]$	$[2, 1]$
Heterogeneity Score	0.843	0.975

The confusion matrices for each experiment are given in Table (4) and Table (5). Here, we can see again that the model without hierarchy tend to favour a single large cluster. Also, the accuracy of clustering without using Majority Voting for combining different features gives a slightly better performance in comparison to the Majority Voting usage approach.

Table 4: Confusion Matrix for Experiment 3

	Clusters generated By Gibbs Sampler (Membership Percentage)														
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15
Level Ground	0.0	**36.07**	22.02	14.94	9.05	5.48	4.17	3.1	2.14	1.61	0.83	0.36	0.18	0.06	0.0
Ramp Ascent	0.0	**37.7**	21.43	13.49	10.05	5.95	3.84	3.44	1.46	1.98	0.4	0.26	0.0	0.0	0.0
Ramp Descent	0.0	**36.53**	23.19	14.4	9.37	6.79	4.22	2.69	1.76	0.7	0.0	0.23	0.0	0.0	0.12
Stair Ascent	0.0	24.38	**36.51**	13.61	8.84	6.01	4.08	3.63	1.02	0.79	0.34	0.45	0.34	0.0	0.0
Stair Descent	0.0	19.4	**37.98**	16.31	9.76	6.79	4.17	2.62	0.95	1.07	0.71	0.0	0.24	0.0	0.0

Table 5: Confusion Matrix for Experiment 4

	Clusters generated By Gibbs Sampler (Membership Percentage)														
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15
Level Ground	0.0	**65.79**	26.32	5.26	0.0	2.63	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ramp Ascent	0.0	**60.0**	30.0	10.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ramp Descent	0.0	**100.0**	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Stair Ascent	0.0	**66.67**	20.83	12.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Stair Descent	0.0	**63.33**	20.0	3.33	10.0	3.33	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

As the result suggests, it is difficult to cluster EMG signals with enough accuracy. The best effort from our approach (including hierarchy without Majority Voting) was able to obtain accuracy of only around 35%. In the next section, we compare our approach with existing time series clustering algorithms using the same data set.

Comparison with existing approaches We compare our method with existing approaches including k-means clustering with various similarity/dissimilarity measure for time series. For each method, we obtained clusters from the original time series, peak amplitude series and feature series. Here, we report the best result we obtained for each approach reported in Table (6).

Table 6: Performance measure of different time series clustering approaches

Method	**Accuracy (%)**	**Remark**
Bayesian Nonparametrics Time Series Clustering (BNPTSclust)	26.0	This approach is based on Bayesian non parametric where the number of clusters from the data is detected from the model itself. This approach favors a single cluster most of the time
Rest of the algorithms are k-means clustering algorithm		
Autocorrelation based Dissimilarity (ACF)	26.0	
Periodogram-based distances (PER)	25.3	
Normalized Compression Distance (NCD)	23.3	
Euclidean Distance (EUCL)	36.0	
Compression-based dissimilarity measure (CDM)	24.7	
Dynamic Time Warping (DTW) measure	31.3	
Discrete Wavelet Transform (DWT)	30.7	
Correlation Based Dissimilarity (COR)	29.3	
Partial Autocorrelation based Dissimilarity (PACF)	28.0	
Complexity Invariant Distance (CID)	30.7	
Permutation Distribution Clustering (PDC)	18.7	Used default configuration for clustering
Heterogeneous time series clustering	39.1	The best accuracy is obtained when not considering Majority Voting, while specifying the number of clusters to be only 5.

5 Conclusion and Future Work

We study the feasibility of clustering approach for Human Activity Recognition using sensor dataset. Our approach introduces hierarchy-based heterogeneity for clustering time series where the number of clusters is not known in advance. Experimental results show that introducing hierarchy helps in clustering such sensor data time series better. Though the accuracy of our approach for EMG data is lower than expected, comparison with other time series clustering approaches shows that our method is superior in terms of accuracy. The current method expresses the time series as a linear model only, future work will involve extension to non-linear model to handle more complex time series, along with using more datasets for experiments.

References

1. Akaike, H. A new look at the statistical model identification. *Automatic Control, IEEE Transactions on 19*, 6 (1974), 716–723.
2. Alan E. Gelfand, A. F. M. S. Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association 85*, 410 (1990), 398–409.
3. Bao, L., and Intille, S. S. Activity recognition from user-annotated acceleration data. In *Pervasive computing*. Springer, 2004, pp. 1–17.
4. Batista, G. E., Wang, X., and Keogh, E. J. A complexity-invariant distance measure for time series. In *SDM* (2011), vol. 11, SIAM, pp. 699–710.
5. Brandmaier, A. M. pdc: An r package for complexity-based clustering of time series. *Journal of Statistical Software 67*, 5 (2015).
6. Bryk, A. S., and Raudenbush, S. W. *Hierarchical linear models: applications and data analysis methods*. Sage Publications, Inc, 1992.
7. Caiado, J., Crato, N., and Peña, D. A periodogram-based metric for time series classification. *Computational Statistics & Data Analysis 50*, 10 (2006), 2668–2684.
8. Connor, R. J., and Mosimann, J. E. Concepts of independence for proportions with a generalization of the dirichlet distribution. *Journal of the American Statistical Association 64*, 325 (1969), 194–206.
9. De Luca, C. J., Adam, A., Wotiz, R., Gilmore, L. D., and Nawab, S. H. Decomposition of surface emg signals. *Journal of neurophysiology 96*, 3 (2006), 1646–1657.
10. de Wiljes, J., Majda, A., and Horenko, I. An adaptive markov chain monte carlo approach to time series clustering of processes with regime transition behavior. *Multiscale Modeling & Simulation 11*, 2 (2013), 415–441.
11. Efron, B., and Morris, C. Empirical bayes on vector observations: An extension of stein's method. *Biometrika 59*, 2 (1972), 335–347.
12. Englehart, K., and Hudgins, B. A robust, real-time control scheme for multifunction myoelectric control. *Biomedical Engineering, IEEE Transactions on 50*, 7 (2003), 848–854.
13. Escobar, M. D. *Estimating the means of several normal populations by nonparametric estimation of the distribution of the means*. PhD thesis, Department of Statistics, Yale University, New Haven, 1988.
14. Escobar, M. D., and West, M. Bayesian density estimation and inference using mixtures. *Journal of the american statistical association 90*, 430 (1995), 577–588.

15. Everson, P. J., and Morris, C. N. Inference for multivariate normal hierarchical models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 62*, 2 (2000), 399–412.
16. Galeano, P., and Peña, D. Multivariate analysis in vector time series. *Resenhas* (2000), 383–404.
17. Golay, X., Kollias, S., Stoll, G., Meier, D., Valavanis, A., and Boesiger, P. A new correlation-based fuzzy logic clustering algorithm for fmri. *Magnetic Resonance in Medicine 40*, 2 (1998), 249–260.
18. Graupe, D., and Cline, W. K. Functional separation of emg signals via arma identification methods for prosthesis control purposes. *Systems, Man and Cybernetics, IEEE Transactions on*, 2 (1975), 252–259.
19. Griffin, L. Y., Albohm, M. J., Arendt, E. A., Bahr, R., Beynnon, B. D., DeMaio, M., Dick, R. W., Engebretsen, L., Garrett, W. E., Hannafin, J. A., et al. Understanding and preventing noncontact anterior cruciate ligament injuries a review of the hunt valley ii meeting, january 2005. *The American journal of sports medicine 34*, 9 (2006), 1512–1532.
20. Gupta, P., and Dallas, T. Feature selection and activity recognition system using a single triaxial accelerometer. *Biomedical Engineering, IEEE Transactions on 61*, 6 (2014), 1780–1786.
21. Ha, K. H., Varol, H. A., and Goldfarb, M. Volitional control of a prosthetic knee using surface electromyography. *Biomedical Engineering, IEEE Transactions on 58*, 1 (2011), 144–151.
22. Hargrove, L. J., Simon, A. M., Young, A. J., Lipschutz, R. D., Finucane, S. B., Smith, D. G., and Kuiken, T. A. Robotic leg control with emg decoding in an amputee with nerve transfers. *New England Journal of Medicine 369*, 13 (2013), 1237–1242.
23. Harrison, J., and West, M. *Bayesian Forecasting & Dynamic Models*. Springer, 1999.
24. Harrison, P. J., and Stevens, C. F. Bayesian forecasting. *Journal of the Royal Statistical Society. Series B (Methodological)* (1976), 205–247.
25. Herr, H., and Wilkenfeld, A. User-adaptive control of a magnetorheological prosthetic knee. *Industrial Robot: An International Journal 30*, 1 (2003), 42–55.
26. Huang, H., Kuiken, T., Lipschutz, R. D., et al. A strategy for identifying locomotion modes using surface electromyography. *Biomedical Engineering, IEEE Transactions on 56*, 1 (2009), 65–73.
27. Huang, H., Zhang, F., Hargrove, L. J., Dou, Z., Rogers, D. R., and Englehart, K. B. Continuous locomotion-mode identification for prosthetic legs based on neuromuscular–mechanical fusion. *Biomedical Engineering, IEEE Transactions on 58*, 10 (2011), 2867–2875.
28. Hudgins, B., Parker, P., and Scott, R. N. A new strategy for multifunction myoelectric control. *Biomedical Engineering, IEEE Transactions on 40*, 1 (1993), 82–94.
29. Ishwaran, H., and James, L. F. Gibbs sampling methods for stick-breaking priors. *Journal of the American Statistical Association 96*, 453 (2001).
30. Ishwaran, H., and Zarepour, M. Markov chain monte carlo in approximate dirichlet and beta two-parameter process hierarchical models. *Biometrika 87*, 2 (2000), 371–390.
31. Jin, D., Yang, J., Zhang, R., Wang, R., and Zhang, J. Terrain identification for prosthetic knees based on electromyographic signal features. *Tsinghua Science & Technology 11*, 1 (2006), 74–79.

32. Johansson, J. L., Sherrill, D. M., Riley, P. O., Bonato, P., and Herr, H. A clinical comparison of variable-damping and mechanically passive prosthetic knee devices. *American journal of physical medicine & rehabilitation 84*, 8 (2005), 563–575.
33. Keogh, E., and Kasetty, S. On the need for time series data mining benchmarks: a survey and empirical demonstration. *Data Mining and knowledge discovery 7*, 4 (2003), 349–371.
34. Keogh, E., Wei, L., Xi, X., Lee, S.-H., and Vlachos, M. Lb_keogh supports exact indexing of shapes under rotation invariance with arbitrary representations and distance measures. In *Proceedings of the 32nd international conference on Very large data bases* (2006), VLDB Endowment, pp. 882–893.
35. Kumar, D. K., Pah, N. D., and Bradley, A. Wavelet analysis of surface electromyography. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on 11*, 4 (2003), 400–406.
36. Li, M., Chen, X., Li, X., Ma, B., and Vitányi, P. The similarity metric. *Information Theory, IEEE Transactions on 50*, 12 (2004), 3250–3264.
37. Liao, T. W. Clustering of time series data—a survey. *Pattern recognition 38*, 11 (2005), 1857–1874.
38. Lindley, D. V., and Smith, A. F. Bayes estimates for the linear model. *Journal of the Royal Statistical Society. Series B (Methodological)* (1972), 1–41.
39. Lindsay, R. W., Percival, D. B., and Rothrock, A. D. The discrete wavelet transform and the scale analysis of the surface properties of sea ice. *Geoscience and Remote Sensing, IEEE Transactions on 34*, 3 (1996), 771–787.
40. Loudon, J. K., Swift, M., and Bell, S. *The clinical orthopedic assessment guide*. Human Kinetics, 2008.
41. Mannini, A., Genovese, V., and Sabatin, A. M. Online decoding of hidden markov models for gait event detection using foot-mounted gyroscopes. *Biomedical and Health Informatics, IEEE Journal of 18*, 4 (2014), 1122–1130.
42. Mendoza, M., and E Nieto-Barajas, L. Bayesian solvency analysis with autocorrelated observations. *Applied Stochastic Models in Business and Industry 22*, 2 (2006), 169–180.
43. Montero, P., and Vilar, J. A. Tsclust: An r package for time series clustering. *Journal of* (2014).
44. Muliere, P., and Walker, S. Extending the family of bayesian bootstraps and exchangeable urn schemes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 60*, 1 (1998), 175–182.
45. Nieto-Barajas, L. E., and Contreras-Cristan, A. A bayesian nonparametric approach for time series clustering. *Bayesian Anal. 9*, 1 (2014), 147–170.
46. Ortiz-Catalan, M., Brånemark, R., and Håkansson, B. Biopatrec: A modular research platform for the control of artificial limbs based on pattern recognition algorithms. *Source code for biology and medicine 8*, 11 (2013).
47. Parker, P. A., and Scott, R. Myoelectric control of prostheses. *Critical reviews in biomedical engineering 13*, 4 (1985), 283–310.
48. Peeraer, L., Aeyels, B., and Van der Perre, G. Development of emg-based mode and intent recognition algorithms for a computer-controlled above-knee prosthesis. *Journal of biomedical engineering 12*, 3 (1990), 178–182.
49. Petris, G., Petrone, S., and Campagnoli, P. *Dynamic linear models with R*. Springer Science & Business Media, 2009.
50. Prado, R., and West, M. Exploratory modelling of multiple non-stationary time series: Latent process structure and decompositions. In *Modelling Longitudinal and Spatially Correlated Data*. Springer, 1997, pp. 349–361.

51. Psonak, R. Transfemoral prosthetics. *Orthotics and Prosthetics in Rehabilitation* (2000), 491–520.
52. Rakthanmanon, Q. Z. G. B. T., and Keogh, E. A novel approximation to dynamic time warping allows anytime clustering of massive time series datasets.
53. Rakthanmanon, T., Keogh, E. J., Lonardi, S., and Evans, S. Mdl-based time series clustering. *Knowledge and information systems 33*, 2 (2012), 371–399.
54. Reaz, M., Hussain, M., and Mohd-Yasin, F. Techniques of emg signal analysis: detection, processing, classification and applications. *Biological procedures online 8*, 1 (2006), 11–35.
55. Stisen, A., Blunck, H., Bhattacharya, S., Prentow, T. S., Kjærgaard, M. B., Dey, A., Sonne, T., and Jensen, M. M. Smart devices are different: Assessing and mitigatingmobile sensing heterogeneities for activity recognition. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems* (2015), ACM, pp. 127–140.
56. Teh, Y. W. Dirichlet process. In *Encyclopedia of machine learning*. Springer, 2010, pp. 280–287.
57. Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. Hierarchical Dirichlet processes. *Journal of the American Statistical Association 101*, 476 (2006), 1566–1581.
58. Varol, H. A., Sup, F., and Goldfarb, M. Multiclass real-time intent recognition of a powered lower limb prosthesis. *Biomedical Engineering, IEEE Transactions on 57*, 3 (2010), 542–551.
59. Vleeming, A., Pool-Goudzwaard, A. L., Stoeckart, R., van Wingerden, J.-P., and Snijders, C. J. The posterior layer of the thoracolumbar fascia— its function in load transfer from spine to legs. *Spine 20*, 7 (1995), 753–758.
60. Walker, S., and Muliere, P. Beta-stacy processes and a generalization of the pólya-urn scheme. *The Annals of Statistics* (1997), 1762–1780.
61. Williams III, T. W. Practical methods for controlling powered upper-extremity prostheses. *Assistive Technology 2*, 1 (1990), 3–18.