PeerJ Computer Science

Solving musculoskeletal biomechanics with machine learning

Yaroslav Smirnov[1], Denys Smirnov[2], Anton Popov[1,3] and Sergiy Yakovenko[4,5,6,7,8]

[1] Department of Electronic Engineering, Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, Ukraine
[2] Department of Computer-aided Management and Data Processing Systems, Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, Ukraine
[3] Data & Analytics, Ciklum, Kyiv, Ukraine
[4] Department of Human Performance—Exercise Physiology, School of Medicine, West Virginia University, Morgantown, West Virginia, United States
[5] Department of Biomedical Engineering, Benjamin M. Statler College of Engineering and Mineral Resources, West Virginia University, Morgantown, West Virginia, United States
[6] Rockefeller Neuroscience Institute, School of Medicine, West Virginia University, Morgantown, West Virginia, United States
[7] Mechanical and Aerospace Engineering, Benjamin M. Statler College of Engineering and Mineral Resources, West Virginia University, Morgantown, West Virginia, United States
[8] Department of Neuroscience, School of Medicine, West Virginia University, Morgantown, West Virginia, United States

ABSTRACT

Deep learning is a relatively new computational technique for the description of the musculoskeletal dynamics. The experimental relationships of muscle geometry in different postures are the high-dimensional spatial transformations that can be approximated by relatively simple functions, which opens the opportunity for machine learning (ML) applications. In this study, we challenged general ML algorithms with the problem of approximating the posture-dependent moment arm and muscle length relationships of the human arm and hand muscles. We used two types of algorithms, light gradient boosting machine (LGB) and fully connected artificial neural network (ANN) solving the wrapping kinematics of 33 muscles spanning up to six degrees of freedom (DOF) each for the arm and hand model with 18 DOFs. The input-output training and testing datasets, where joint angles were the input and the muscle length and moment arms were the output, were generated by our previous phenomenological model based on the autogenerated polynomial structures. Both models achieved a similar level of errors: ANN model errors were 0.08 ± 0.05% for muscle lengths and 0.53 ± 0.29% for moment arms, and LGB model made similar errors—0.18 ± 0.06% and 0.13 ± 0.07%, respectively. LGB model reached the training goal with only 10^3 samples, while ANN required 10^6 samples; however, LGB models were about 39 times slower than ANN models in the evaluation. The sufficient performance of developed models demonstrates the future applicability of ML for musculoskeletal transformations in a variety of applications, such as in advanced powered prosthetics.

Subjects Bioinformatics, Computational Biology, Human-Computer Interaction, Algorithms and Analysis of Algorithms, Artificial Intelligence
Keywords Machine learning, Deep neural networks, Muscle, Hand, Real-time, Biomechanics

Submitted 12 March 2021
Accepted 16 July 2021
Published 26 August 2021

Corresponding author
Sergiy Yakovenko,
seyakovenko@hsc.wvu.edu

Academic editor
Li Zhang

Additional Information and Declarations can be found on page 15

DOI 10.7717/peerj-cs.663

How to cite this article Smirnov Y, Smirnov D, Popov A, Yakovenko S. 2021. Solving musculoskeletal biomechanics with machine learning. PeerJ Comput. Sci. 7:e663 DOI 10.7717/peerj-cs.663

INTRODUCTION

Machine learning (ML) with artificial neural networks (ANN) is revolutionizing applications where recognition, cognition, and categorization abilities used to require direct human involvement (*LeCun, Bengio & Hinton, 2015*). While the initial focus was on the substitution of human operators, for example, in driving a vehicle (*Shi et al., 2020*), classifying media (*Gupta & Katarya, 2020*), or for radiological diagnostics of benign or malignant mass in mammography (*Shen et al., 2019*; *McKinney et al., 2020*), the reach of ANN extended to the exploration of sensorimotor mechanisms (*Richards et al., 2019*) that could potentially work in the closed-loop systems with human operator. The structural and functional complexity of this system, which is distributed across multiple neural and mechanical pathways (*Valero-Cuevas, 2015*) and has high-dimensional computations for our segmented body control (*Bernstein, 1967*), offers a unique challenge and opportunities for this approach. One of the opportunities lies within the innate ability of ANNs to absorb and classify a high volume of multidimensional input-output relationships. This is not dissimilar to biological processing responsible for coordinated spatiotemporal action of multiple muscles generating movement. Yet another extreme expression of the neural processing complexity and efficiency is the brain's ability to solve the Bernsteinian degrees of freedom problem where the same motor goal of body control can be accomplished with different kinematic solutions (*Bernstein, 1967*). ML methods based on ANNs can potentially resolve or, at least, identify targets for the long-standing theoretical challenge that can provide insight to the current theories of neural processing (*McNamee & Wolpert, 2019*) and has multiple practical human-machine applications, *e.g*., in advanced prosthetics (*Dantas et al., 2019*).

Developing a fast and intuitive interface with a high-dimensional artificial limb is a challenging task solved increasingly with the help of pattern recognition algorithms (*Muceli & Farina, 2012*; *Geethanjali, 2016*). Typically, myoelectric direct and proportional control is used to decode motor intent from the recorded surface electromyography (EMG) and convert it into joint torques or furthermore positions of the powered prosthetic devices (*Geethanjali, 2016*). This was previously accomplished using ANN algorithms decoding kinematics from EMG signals (*Muceli & Farina, 2012*). The intuitive control requires an additional transformation based on the representation of the controlled device and its neural control (*Castellini & Van der Smagt, 2009*). The failure to account for the dynamics of prosthesis would lead to direct kinematic errors. The failure to recognize the biological strategies in solving limb dynamics would reduce robustness and intuitiveness of control even when the control of mechanical devices is perfectly tuned. The latter would occur because the interlimb inertial dynamics is encoded within neural commands even when limb dynamics changes. For example, mechanical shoulder immobilization does not abolish the stabilizing shoulder muscle activity during elbow movement (*Gribble & Ostry, 1998*, *1999*; *Debicki & Gribble, 2005*). The expected musculoskeletal dynamics persists within neural commands months and years after the acute stage of limb trauma and amputations. The successful use of these commands for prosthetic control would theoretically require the representation of pre-trauma musculoskeletal and segmental limb dynamics to account for the dynamics encoded within neural control signals.

Relatively few studies applied ML techniques to musculoskeletal dynamics. The early applications of simple feed-forward ANNs allowed mapping of an average locomotor pattern of 16 EMGs to hip, knee, and ankle joint angles and moments (*Sepulveda, Wells & Vaughan, 1993*). Even though the muscle paths and muscle force generation were not simulated in this study, the decoding of posture was demonstrated with low errors. The changes in the locomotor pattern at slow and fast speeds can also be generalized by a simple ANN with supervised learning *via* backpropagation (*Heller et al., 1993*). Furthermore, the mapping can be done not only between the locomotor activation patterns, but also with the muscle forces; albeit, the accurate predictions within trials have not been demonstrated (*Liu, Herzog & Savelberg, 1999*). Yet, similar type of statistical mapping, admittedly, can be expressed with standard dimensionality reduction techniques with high precision and low computational cost, *e.g*., principal component analysis (PCA) (*Patla, 1985*). Multiple methodological variations have been since developed and applied to solve musculoskeletal problems. Notably recurrent ANNs were used to predict elbow torques from EMGs and showed the benefit of taking into account kinematic inputs, joint angle and velocity (*Song & Tong, 2005*). A combination of convolutional and recurrent ANNs can accurately and robustly map from the time-frequency frames of multi-channel EMG to limb movement (*Xia, Hu & Peng, 2018*). Purely statistical learning of the musculoskeletal transformation from posture to the control inputs has been also demonstrated with multiple hybrid ANN methods for a 7 DOF robotic arm with artificial muscles (*Hartmann et al., 2013*). Similar to our approach the musculoskeletal transformation was learnt from an input-output dataset. While the tracking of a two DOF arm was achieved, the control of both robot and its simulation resulted in large tracking errors. The high-dimensional control remains a challenge.

While data-driven mapping with ANNs or using PCA and other statistical classification alternatives are robust, these methods do not generally capture the mechanistic relationships. Intrinsic muscle properties and musculoskeletal organization contribute to the muscle force generation, and these mechanistic details may assist in the reconstruction of relationships between neural commands and generated movement. In this study, the problem of learning the musculoskeletal dynamics (MSD) will be addressed with several ML techniques that may generate a computationally efficient solution. MSD requires high-dimensional transformations of posture into muscle moment arms and length, which are the essential variables in the calculation of generated muscle forces. The Hill-type muscle model can then allow us to define the posture-dependent force-length-velocity dependency (Fig. 1, Muscle Model) (*Zajac, 1989*) and next to compute muscle and joint torques (*Buchanan et al., 2004*). The remaining step for the generation of movement is the simulation of equations of motion by using a physics engine or its approximation. The critical constraint for the accuracy of these computations is the loop latency that limits the computational stability of integration (*Todorov, Erez & Tassa, 2012*). The trade-off between accuracy and latency can be achieved using methods similar to least-squared

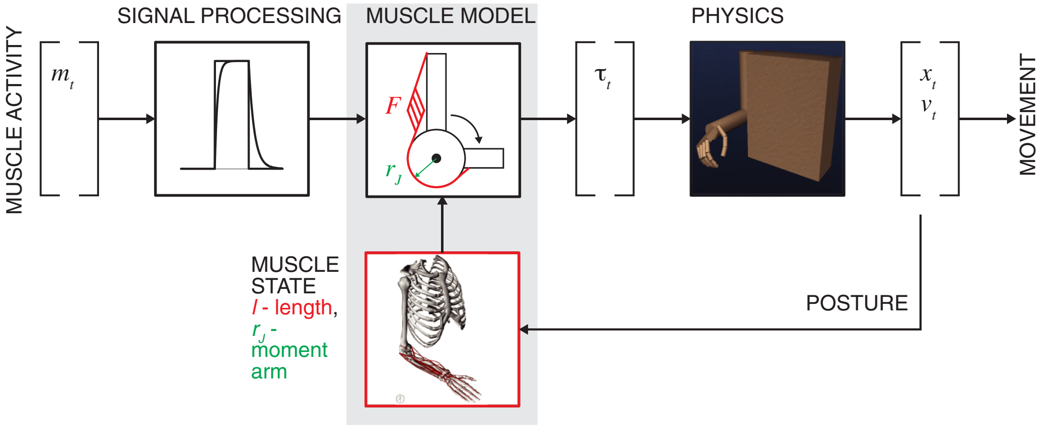

Figure 1 **General concept of motor intent decoding from muscle activity.** The schematic illustrates the transformation of EMG inputs through signal processing and musculoskeletal relationships (muscle model) into estimated torques that actuate limbs to generate movement by solving the equations of motion (physics). Limb posture modifies nonlinear muscle force-length-velocity relationship and torques. Full-size DOI: 10.7717/peerj-cs.663/fig-1

approximation, for example, used for the inverse dynamics computations (*Kuo, 1998*). Thus, the goal for real-time biomechanics is to implement a method with high accuracy and low computational cost (low latency) of musculoskeletal transformations.

In this study, we solve the problem of estimating muscle moment arms and their muscle lengths from joint angles with two ML approaches. We used an arm and hand model to generate input-output kinematic datasets, where joint angles were the input and the muscle length and moment arms were the output, and presented the comparative validation and performance metrics for the two solutions. The results of this study develop the potential for mechanistic ML approaches that utilize the musculoskeletal transformation for online control problems.

MATERIALS & METHODS

Musculoskeletal polynomial model for generation of training and testing datasets

We have previously developed the method of autogenerated polynomial models (*Sobinov et al., 2020*). In these polynomials, the composition of terms was expanded using objective information measurements, *i.e.*, the corrected Akaike Information Criterion. In brief, the posture-dependent musculotendon actuator length and joint moment arms for each muscle in the upper-limb model (*Gritsenko et al., 2016*) were accurately approximated using the selection of up to 5th power polynomial terms, where muscle length and moment arms were connected through a partial derivative of the muscle length in local coordinates corresponding to limb posture. Overall, the 18 DOF model of the human arm and hand is actuated by 33 muscles, each spanning about 3 DOFs and up to 6 DOFs for thumb muscles. Thus, each actuator is represented by a set of one length- and about 3 moment arm-posture polynomials. The goal of this development was to bypass costly calculations of geometrical transformations with high-quality approximations. Previously, we have demonstrated high-fidelity of these approximations with kinematic errors below 1% (Sobinov et al., 2020). In biomechanics, the errors of 1–5° in joint angles are expected from flaws in the observations in motion capture, and errors of 2° and less are not meaningful in the clinical context (McGinley et al., 2009). Thus, the errors below 1% of joint range of motion are negligible. These polynomial models of muscle posture-dependent state were used to develop an ANN-based approximation method for the musculoskeletal dynamics in this study.

Training, validation and testing datasets

Training, validation and testing datasets for the assessment of model performance were generated by the musculoskeletal polynomial model, which was used as a reference in this study. Input-output relationships were extracted randomly with uniform distribution where inputs were 18 DOF vectors of joint angles and the outputs were 33 length vectors and 99 moment arm vectors. An average muscle crosses 3 DOFs and has, consequently, 3 moment arm relationships on average. We used the supervised learning approach for training ML models. The training dataset was used for two tasks, tuning the model hyper-parameters and model training, to maximize the model performance in replicating the desired outputs with given inputs. The testing dataset contained ~5% of all data (5×10^4 samples). The remaining ~95% were divided into the training dataset (80%, 8×10^5 samples) and the validation dataset (20%, 2×10^5 samples). The validation dataset was used to prevent overfitting, *i.e.*, higher performance on the training data as compared to that on the validation data. These datasets were similarly used for training ANN and LGB models, described below. Overall, the training time was about 15 times longer for ANN then for LGB models. The training of all ANN and LGB models on the standard hardware took about 3.5 days.

Metrics

The performance of the trained models was further evaluated with the testing dataset, which was not used during the training procedure. We expected to reach the same error tolerances as in our previous polynomial fitting method study (Sobinov et al., 2020). Consequently, we used the same normalization of lengths and moment arms as in our previous work. The RMSE values were calculated as the absolute difference between reference and predicted muscle length values. To normalize the results, we divided each reference and predicted length value by the muscle length range respectively:

$$RMSE_L = \frac{1}{m}\sum_{l=1}^{m}\frac{1}{n}\sum_{i=1}^{n}\sqrt{\left(\frac{x_{r,l,i} - x_{p,l,i}}{Lmax_l - Lmin_l}\right)^2}$$

where m is the number of muscles ($m = 33$), n is the number of test samples, $x_{r,l,i}$ and $x_{p,l,i}$ are reference and predicted length values, respectively, $Lmax_l$ and $Lmin_l$ are the maximum and minimum values over the full range of lth muscle length.

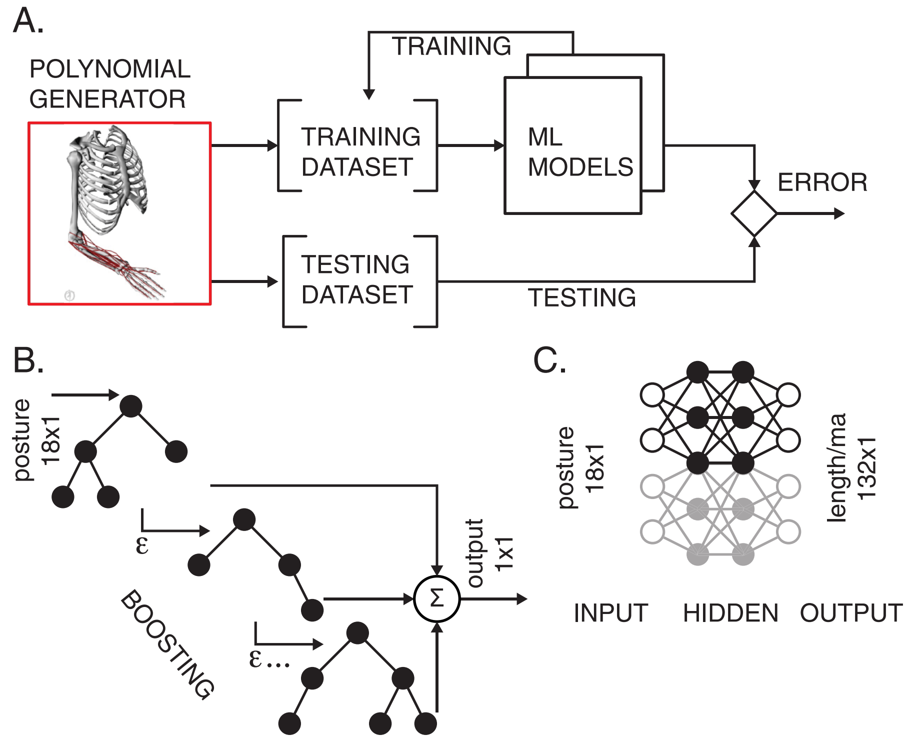

Figure 2 Training and testing of ML models. (A) The polynomial generator created reference datasets, which were then iteratively used for training and testing. (B) LGB model with decision binary trees using gradient boosting. The transformation from input postures to output scalar values corresponding to either muscle length or moment arm values was performed in boosting stages to improve accuracy. (C) ANN with two hidden layers performed transformation for all lengths and moment arms in the model. Full-size DOI: 10.7717/peerj-cs.663/fig-2

Similarly, the RMSE of moment arms was calculated as the absolute difference between reference and predicted values, which were normalized to the moment arm maximum ($Mmax_l$):

$$RMSE_{MA} = \frac{1}{m}\sum_{l=1}^{m}\frac{1}{n}\sum_{i=1}^{n}\sqrt{\left(\frac{x_{r,l,i} - x_{p,l,i}}{Mmax_l}\right)^2}$$

where $x_{r,l,i}$ and $x_{p,l,i}$ are reference and predicted values, m is the number of moment arms (m = 99), and n is the number of test samples.

Machine learning models

Two types of ML models were used to map the musculoskeletal input-output relationships. We used the light gradient boosting machine (LGB) models and artificial neural network (ANN) with two hidden layers. The models were trained and tested according to the workflow in Fig. 2A. Validation accompanied the training process to prevent overfitting.

Light gradient boosting machine (LGB)

LGB algorithms belong to the group of gradient boosting methods based on choosing iteratively simple learner functions that point to the global minimum in the cost function.

Gradient boosting is a technique to assemble weak prediction models, in our case regression trees, as processing stages that reduce performance errors. Here, the regression trees use binary recursive decisions to follow a path along hierarchically organized nodes that terminate with the final branches, called leaves. The training process was the search for the optimal routing of inputs so that similar outputs were grouped together (*Dantas et al., 2019*; *Microsoft Corporation, 2020*). The boosting method assembles the sequences of multiple regression trees to process errors in stages and gradually improve output accuracy (Fig. 2B). We used gradient-based one-side sampling in LGB to select a set of inputs where previous weak learner models have the largest output errors. The structure of the decision trees adapted to the required error tolerance by expanding the number of nodes (leaves) up to the maximal preset value determined empirically. We used Microsoft open source implementation of LGB (lightgbm v.2.2.3, Microsoft Corporation). The implementation of LGB requires multiple parameters for training the model that improves the transformation by adding nodes to trees (leaf-wise tree growth). We kept a set of parameters constant across all models:

```
import lightgbm
params = {‘seed’: 2523252, ‘tree_learner’: ‘serial’, ‘pre_partition’: True, ‘is_unbalance’:
False, ‘early_stopping_rounds’: 200, ‘metric’: ‘mse’}
lightgbm.LGBMRegressor(**params)
```

The additional training parameters were added as inputs to *params* statement and varied across models. The following is the example implementation for one of the models:

```
params = {‘seed’: 2523252, ‘tree_learner’: ‘serial’, ‘pre_partition’: True, ‘is_unbalance’:
False, ‘early_stopping_rounds’: 200, ‘metric’: ‘mse’, “num_iterations”: 358,
“num_leaves”: 41, “learning_rate”: 0.040600000000000004, “max_depth”: 18,
“min_data_in_leaf”: 67, “max_drop”: 23, “bagging_fraction”: 0.8, “feature_fraction”: 0.8 }
```

The full list of all training parameters is provided in the Supplemental Materials.

Each muscle length and moment arm relationship with posture was fitted with one LGB model. The full arm and hand model was simulated by 33 length and 99 moment arm transformations of 18 dimensional posture input. Three types of hyper-parameters were iteratively optimized prior to training: (1) the number of leaves in a single decision tree (range: 20–100); (2) the minimal number of samples in one leaf (range: 10–100); (3) the maximum tree depth as the number of split levels (range: 1–100). Values for each LGB model were determined iteratively using the Bayesian optimization (*Snoek, Larochelle & Adams, 2012*) on training and validation datasets selected as 70% and 30% of all data, respectively. Other hyper-parameters within LGB models, *e.g*. the number of weak estimators in boosting (100), were chosen as defaults of Microsoft implementation v.2.2.3 (*Ke et al., 2017*).

Artificial neural network (ANN)

We developed two ANN models to evaluate posture-dependent muscle lengths and moment arms. We selected fully connected feed-forward layers with one input, one output, and two hidden layers (Fig. 2C) (*Sazli, 2006*) with rectifying linear units as the outputs of every layer. This standard model provides robust gradient propagation with efficient computation (*Glorot, Bordes & Bengio, 2011*). Using TensorFlow (*Abadi et al., 2016*), we composed our networks consisting of the following number of nodes in input, two hidden, and output layers: (18, 1,024, 512, 33) for the approximation of 33 muscle lengths, and (18, 2,048, 1,024, 99) for the approximation of 99 moment arms.

Xavier initialization method was used to select the initial weights for each layer from the normal distribution with zero mean and its variance as $2/(n_{in}+n_{out})$, where n_{in} and n_{out} were the number of inputs and outputs in this layer (*Glorot & Bengio, 2010*). The network was trained with the batches of sample data (256 samples) using a gradient based stochastic optimization method minimizing a custom cost function (*Kingma & Ba, 2017*). We developed the cost function that focused on the performance of the worst approximations evaluated as RMSE of the worst 5% of input-output pairs from each muscle. The scalar cost was evaluated as the mean of all errors within the upper 30% range.

The variable learning rate was used to improve the learning dynamics. The initial rate of 0.001 was reduced by 20% if the measured metric stopped improving after two full training dataset evaluations, or epochs. We have tested additional two manipulations to improve learning. We tested the variation of processing structure to improve the generalization of solutions distributed across multiple nodes in the ANN. The model was trained with 50% of the nodes skipped in each evaluation and temporarily and randomly assigned to the dropout layer. In addition, we have also tested the normalization of input samples. However, the improvements due to the additional structure variation and the normalization were marginal, and we chose to exclude these manipulations from the processing pipeline.

The presence of overfitting in training was assessed by tracking the divergence in the error rates for training (observed) and testing (unobserved) samples. The difference in errors was less than 0.4% for all muscles without the divergence. For example, for as little as 1,000 samples, the RMSE of the trained model for *Pronator Teres* length was 78.69% for the training set and 79.02% for the testing set, which indicated the absence of overfitting in the early stage of fitting. The difference between training and testing evaluations remained below 0.1% until the terminal level was achieved.

RESULTS

Two types of ML models were trained to approximate the musculoskeletal relationships. Our findings detail the training outcome and the training dynamics for learning the transformation from joint posture to muscle lengths and moment arms.

Estimation of the training dataset size

The selection of the training dataset size for ANN and LGB models is a non-trivial step in the model development. Our source of data was expressed functionally allowing unlimited source of training data. However, the selection of an optimal dataset that captures the relationships without the tendency for overfitting was the initial goal of our development. We used RMSE metric for both length and moment arm models trained with several

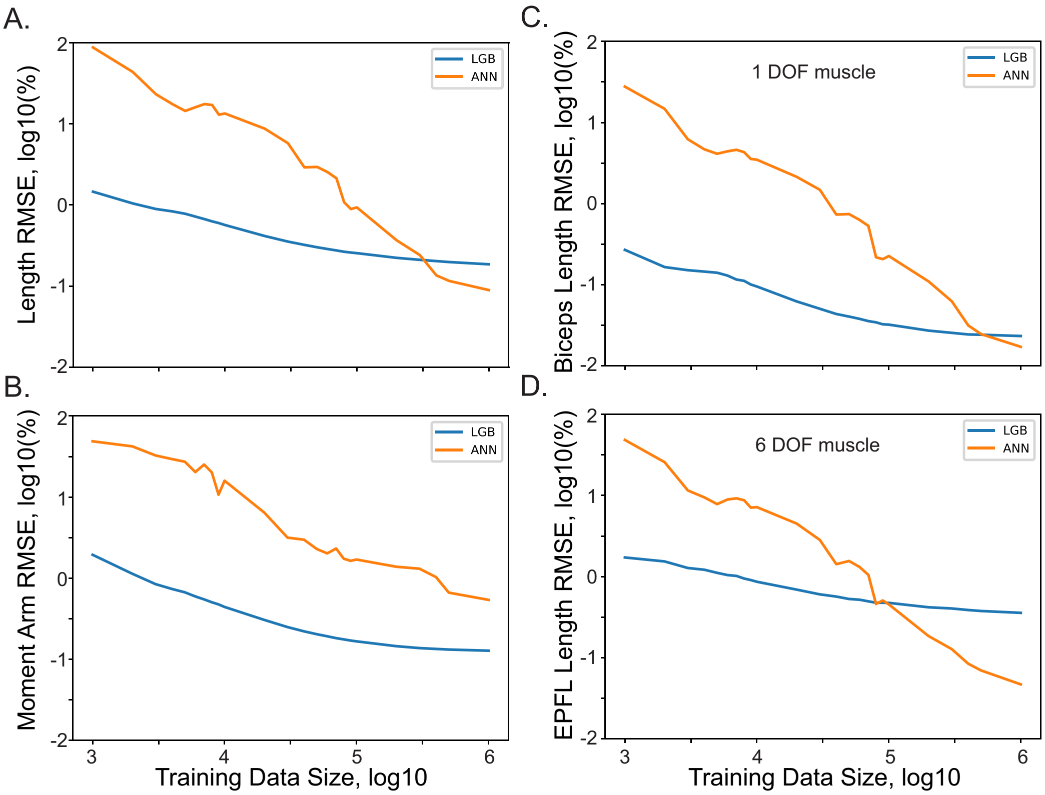

Figure 3 The relationship between model performance (RMSE) and the training dataset size for ANN and LGB models. The length (A) and moment arm (B) relationships with the size are shown for all muscles. Selected examples of muscle length and the dataset size are shown for *Biceps Brachii Long Head* in (C) and *Extensor Pollicis Longus* in (D). Full-size DOI: 10.7717/peerj-cs.663/fig-3

datasets of incremental size. The relationship between the metric and the dataset size are shown in Fig. 3. As the size of the dataset increased logarithmically (from 10^3 to 10^6 samples), the training accuracy also increased, with minor improvement in the range above 10^5 samples. The improvements with the dataset size were not as pronounced showing 1.45% and 1.94% errors with the smallest datasets (10^3 samples). The improvement curve of LGB is flat, showing no further improvement, after 10^5 sample size. The performance of the relatively simple (*Biceps Brachii Long Head*) and complex (*Extensor Pollicis Longus*) muscles is illustrated in Figs. 3C and 3D. These two muscles are on the opposite extremes of complexity expressed as the number of terms required for accurate polynomial fit (*Sobinov et al., 2020*). The improvements are qualitatively similar for both of these muscles indicating no strong dependency on muscle path complexity with both types of models. We used the largest dataset size for all further model development described below.

Model accuracy

High accuracy was achieved with both LGB and ANN model types. The distribution of errors is shown in Fig. 4 with the histograms of RMSE values for the testing dataset (5×10^4 samples). The highest achieved performance of ANN models was with 10^7 with

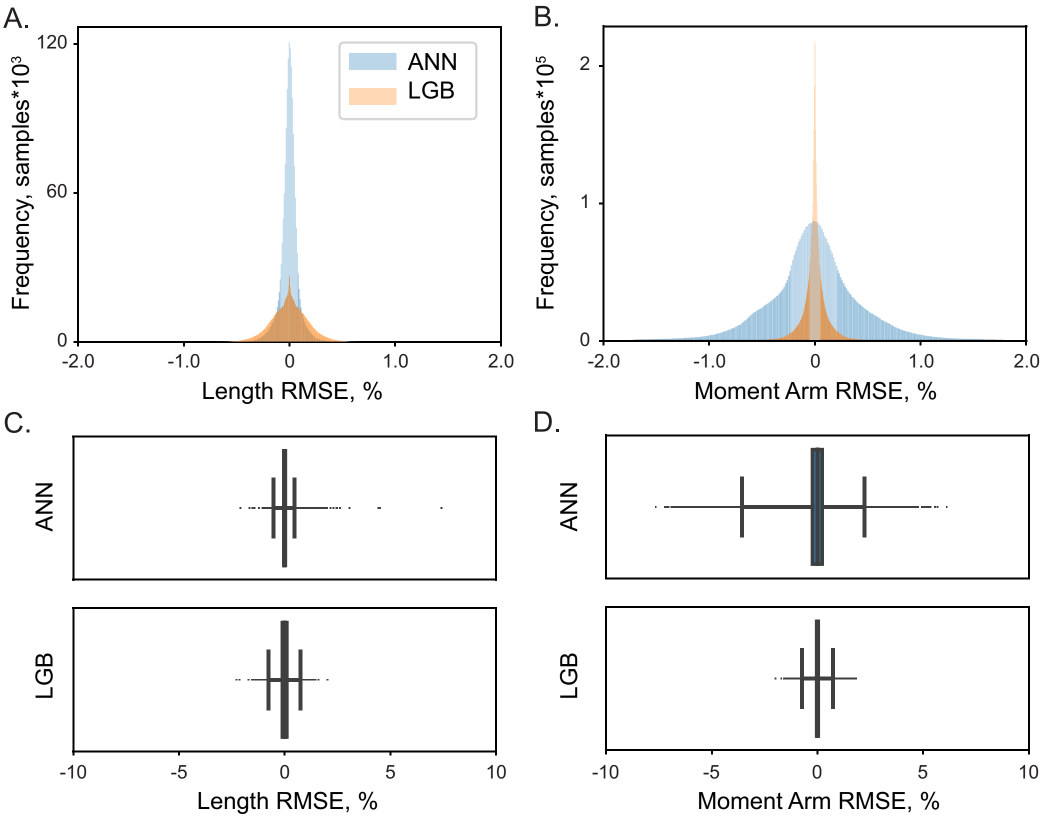

Figure 4 The distribution of normalized errors. Errors in the prediction of muscle length (A) and moment arms (B) are shown for two ML models, LGB (red) and ANN (blue). The distribution of all errors including all the outliers are shown for muscle length (C) and moment arms (D). The box plots indicate 25% to 75% IQR with whiskers set to cut ± 0.1% of the distribution.

Full-size DOI: 10.7717/peerj-cs.663/fig-4

absolute errors at 0.08 ± 0.05% for muscle lengths and 0.53 ± 0.29% for moment arms. Similarly, LGB models generated accurate predictions with large training datasets, 0.18 ± 0.06% and 0.13 ± 0.07% errors, respectively. This is the expected error rate based on our previous analysis (*Sobinov et al., 2020*).

Overall, the error span did not exceed 0.6% for muscle lengths and 2% for muscle moment arms, shown in Figs. 4 and 5. We found no relationship between length and moment arm errors (p = 0.746, R^2 = 0.003). The level of errors was comparable for both simple and complex muscles (spanning more than 3 DOF), *e.g.*, the errors of ECR_BR (a two DOF muscle) were comparable to those of EDM. The accuracy of LGB and ANN models was comparable. The interquartile ranges (IQR), corresponding to the distance between 25% and 75% level for the distribution of all length error values were 0.075% (ANN) and 0.216% (LGB). The 25-75% IQRs for moment arm errors in Fig. 4B were 0.464% (ANN) and 0.0782% (LGB). Median RMSE values for all models were less than 0.01%. To check if accuracy declines in the extreme postures (1st and 4th quartiles), we repeated testing and found a similar rate of errors, 0.144% and 0.587% for lengths and moment arms.

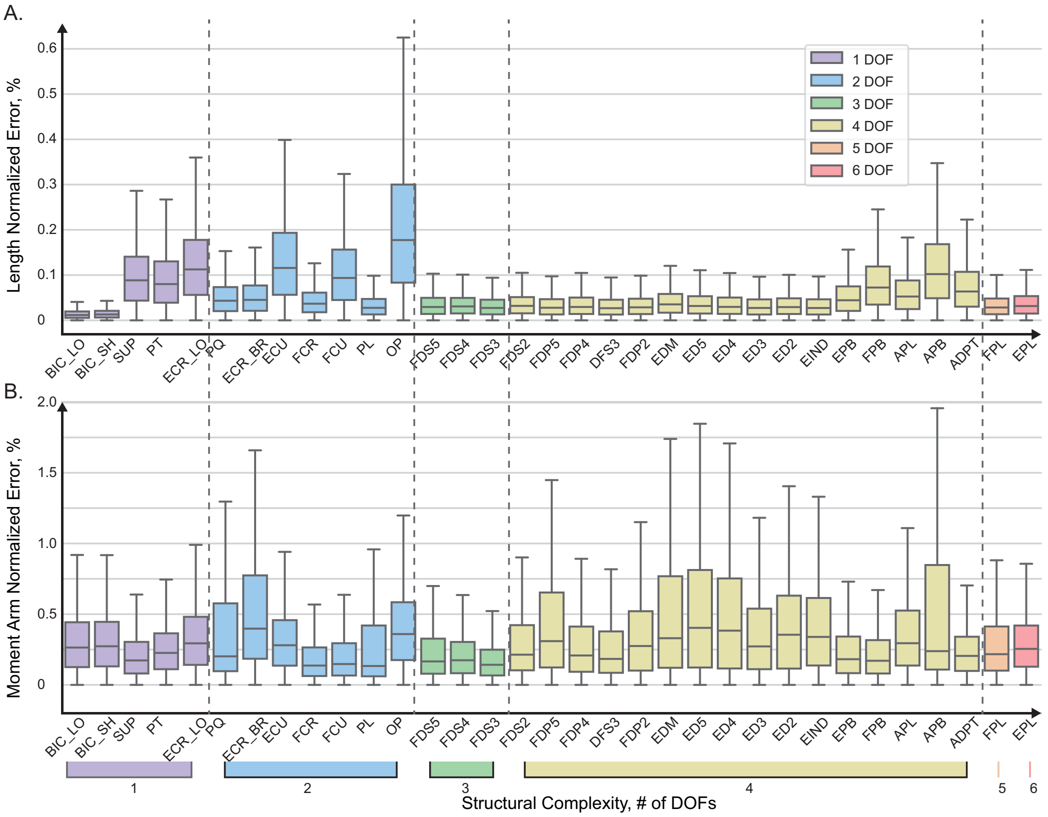

Figure 5 **Normalized absolute errors.** Errors for muscle length (A) and moment arms (B) evaluated with ANN. The box plots indicate 25% to 75% IQR with whiskers set to cut 1.5% of the distribution.

Full-size DOI: 10.7717/peerj-cs.663/fig-5

The distribution of absolute errors is shown for each muscle in Fig. 5. The muscles are sorted according to the number of DOFs they actuate with relatively simple muscles on the left and complex (thumb) muscles on the right. Overall, the majority of distributions were below 0.2% for 75% of all vales, as indicated by the top value of the interquartile range in the box plots. The largest length errors were observed in OP, which was also one of the most difficult muscles to design structurally (*Boots et al., 2020*). In this muscle the top value (Q3) of the interquartile range was about 0.3%, which corresponds to the error of 0.018 mm. In Fig. 5A, the normalized errors of muscle length are presented for each muscle. For all muscles, the most errors (up to 75% of the distributions) are below 0.2%. The prominent exception is OP with the highest normalized errors, which is explained by the minimal full physiological range of only 6 mm. The error of 0.3% in OP length corresponds to the absolute error of 0.018 mm. The errors did not increase with muscle structural complexity. The evaluation errors in muscle length where generally larger in the group of muscles spanning 2 DOF (blue, see *Extensor Carpi Radialis*, *ERC_LO*) and were comparable to the errors in complex muscles (*e.g.*, *Abductor Pollicis Brevis*, *APB*).

Training and evaluation time

The execution times were compared for ANN and LGB models (1.4 GHz Quad-Core 8th-generation Intel Core i5) by measuring the duration of 1000 evaluations (using method *time* from the standard *time* library in Python 3.7). For a given posture, ANN models evaluated both muscle length and moment arms with the combined latency of 1.1 ± 0.6 ms, as compared to 43.1 ± 8.3 ms for LGB models, which were about 39 times slower.

DISCUSSION

In this study, we solved the musculoskeletal kinematics problem over the full physiological range of limb postures using ML approaches. We tested two standard types of models—LGB and ANN—that both accomplished the mapping from limb posture to muscle kinematic state described by multidimensional muscle length and moment arms. LGB and ANN approaches (*Natekin & Knoll, 2013*; *McGovern et al., 2019*) were chosen as the ML equivalents to the phenomenological model previously developed to approximate posture dependent muscle parameters with polynomial structures (*Sobinov et al., 2020*). Both ML methods produced close approximations with the best results achieved by the ANN approach (RMSE = 0.08%) as compared to the LGB approach (RMSE = 0.12%) for moment arms. LGB and ANN methods have not been previously demonstrated for the solution of the musculoskeletal kinematics.

Motor intent decoding

Estimating limb posture from EMG in real-time applications remains to be a challenge in human-machine interfaces due to: (1) the difficulty in the theoretical description and (2) the lack of experimental data to validate these models (*Alber et al., 2019*). In general, a statistical mapping between posture and recorded activity from descending pathways, nerves, and muscles has been used as the transformation to predict motor intent (*Wang & Buchanan, 2002*; *Ting et al., 2019*; *Dantas et al., 2019*), to investigate interplay of mechanical and neural components in pathologies (*Sartori et al., 2017*) or to control powered prosthetic limbs or exoskeletal devices (*Kiguchi & Hayashi, 2012*; *Collinger et al., 2013*; *Zhang et al., 2017*; *Furui et al., 2019*; *George et al., 2020*). However, the accuracy of decoding realistic movements remains to be a challenge especially for movements that require dexterous object manipulation (*Downey et al., 2017*). Many current decoding methods in brain-computer interfaces assume that neural activity is related to limb end-point position and/or velocity and lack the description of movement kinetics generated by muscle forces. The resulting movements of prosthetics are typically slower and less robust than natural movements. Up to five (*Wendelken et al., 2017*) to six (*George et al., 2020*) arm and hand DOFs can be simultaneously controlled using nerve signals recorded with penetrating electrodes. Accurate but slow movements can be generated for high-dimensional artificial hands with wrist and digits; however, the accuracy is challenged by changing limb posture and orientation. One potential solution is the use of closed-loop control systems that provide not only the forward control of prosthetic, but also incorporate the sensory feedback within neuroprosthetics (*Hughes et al., 2020*; *Ganzer et al., 2020*; *Charkhkar, Christie & Triolo, 2020*). The closed-loop control system that takes into account muscle forces would critically depend on the accurate representation of musculoskeletal actions that we describe with our novel approach.

The generalizable control solutions based on the biomechanical transformations have potential advantages over the nontransparent statistical approaches. Mechanistic musculoskeletal models of legs (*Sartori et al., 2017*) and arms (*Crouch & Huang, 2016*; *Boots et al., 2017*; *Mansouri et al., 2017*; *Sartori et al., 2018*) have been previously used for motor intent decoding. The major advantage of biomechanical modeling over statistical methods is in the explicit representation of kinematic and kinetic dependencies in the generated motor command signals during multisegmented limb movements. Thus, the transformation from the recorded biological signals to the proportional control of limbs should be intuitive, given that the underlying computations are *sufficiently accurate* and *without extensive processing delays*.

Together, these possible solutions motivated our exploration and rationale for developing accurate ML methods of approximating the musculoskeletal transformations with physics-informed neural networks. We used model driven training and testing of ML algorithms to approximate posture-dependent changes in muscle lengths and moment arms of distal arm and hand muscles. The previously developed polynomial model provided a functional representation of data across all possible limb postures. Since any volume of data could be generated, we tested the extent of data required to train ANN and LGB models. Figure 3 shows the expected inverse relationship between the approximation errors and the training dataset size. Using only 10^6 samples for training ANN and LGB models resulted in kinematic errors that were less than 0.5%. This number of samples is about one order of magnitude lower than the number of samples required for our previous implementation using an information theory-based algorithm for approximating the same kinematic variables. This may indicate that ANN and LGB methods may further generalize the polynomial relationships that required a larger volume of samples to generate functional muscle-posture representations limited to polynomial power terms.

Computational delays

The transformation of EMG signals into movement rarely accounts for the musculoskeletal anatomy and physiology. This is partly due to the extreme complexity of muscle organization (*Gritsenko et al., 2016*; *Murphy et al., 2018*) and nonlinear intrinsic muscle properties that include independent force-length and force-velocity relationships (*Zajac, 1989*) and less popular in modeling, short-range stiffness, which is a hysteretic force-length property (*Cui et al., 2008*). The task of simulating muscle force generation requires adequate structural information about muscle paths and posture-dependent changes in moment arms. The development of complex musculoskeletal models has been recently simplified by the dedicated simulation tools for editing and simulating segmental dynamics—OpenSim (*Seth et al., 2018*), MuJoCo (*Todorov, Erez & Tassa, 2012*), Simscape (MathWorks, Inc.). The challenge remains in collating sufficient datasets of musculoskeletal measurements for creating complex musculoskeletal models and then in testing and validating these models across the full-range of motion to ensure their use in a wide range of applications.

The computational delays of solving the equations of motion governing limb dynamics have previously impeded the application of model-based prosthetic controllers. In particular, the evaluation of muscle force-velocity characteristic may require sub-millisecond latencies to decode accurately rapid movements computed by a physical engine, which requires additional time to execute (~1 ms in *Mansouri et al., 2017*). Our implementations demonstrated a clear speed advantage of the ANN over the LGB model (about 39× faster) and required about 1 ms on standard hardware. This performance was still about 15 times slower than our polynomial approximation (about 60 μs); yet, it provides a milestone for the future improvements due to the rapid development of both software and hardware solutions for ANNs. Further improvements in the performance of the ANN may be possible approaching the latencies appropriate not only for the feedforward computations, but also for predictive inverse computations inspired by theoretical neural transformations (*Wolpert & Ghahramani, 2000*). Another biomimetic feature of the ANN model is its potential solution for the necessity to increase computational complexity to accommodate the increase in the size of described structure, typically termed as "the curse of dimensionality". *Sobinov et al. (2020)* demonstrated that the typical exponential increase could be replaced with the linear increase in the required number of terms within the polynomial approximations. Here, the same structure of the ANN model accommodated accurate calculations for a subset and for the full set of 33 muscles. We hypothesize that as long as the additional simulated muscles are relatively similar in anatomical complexity to the muscles represented in our current model, the same structure of the ANN will be able to embed their dynamics without the increase in the number of nodes in each layer.

Limitations

Our approach described the kinematic transformations only and improves the calculation of instantaneous muscle forces, yet, it lacks the description of equations of motion. The full transformation to the intended joint and segment kinematics requires the forward simulation of segmental dynamics by a physics engine, for example, MuJoCo (*Todorov, Erez & Tassa, 2012*). Another minor limitation is the use of prior simplified model of muscle kinematics to supervise the training in our ML models. Small errors of about 1% are expected to propagate to this implementation. Given that models of this complexity often include hundreds of poorly validated or guessed parameters, the expected performance is within the "good enough" qualitative range for this model (*McGinley et al., 2009*). For example, in our development of biomechanical models with validated moment arm kinematics, we have discovered large discrepancies between models and experimental data (*Boots et al., 2020*). It is also expected that the individual morphological differences may exceed the numerical errors in our model. In the future work, we will examine the use of ML models in the description of subject-specific musculoskeletal transformations, where the current generic model can be used as the starting pre-trained model. Another current limitation is the demanding training dataset and also the slow execution of the ANN implementation, which is appropriate for real-time applications but is still slower than our previous polynomial approximation.

CONCLUSIONS

We demonstrate in this study the use of two ML methods for solving the posture-dependent changes in the musculoskeletal properties essential for the description of limb kinetics. The achieved execution accuracy was adequate with both ANN and LGB models and similar to the original polynomial model. ANN model was 39 times faster than LGB model computing muscle variables in 1.1 ms, which is appropriate for real-time control solutions.

ADDITIONAL INFORMATION AND DECLARATIONS

Funding

This work was supported by the National Institute of Health/the National Institute of Child Health and Human Development (R03#HD099426). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Grant Disclosures

The following grant information was disclosed by the authors:
National Institute of Health/The National Institute of Child Health and Human Development: R03#HD099426.

Competing Interests

Sergiy Yakovenko is a co-founder of Neurowired and an Academic Editor for PeerJ. Anton Popov is an employee of Ciklum.

Author Contributions

- Yaroslav Smirnov performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Denys Smirnov performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Anton Popov analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Sergiy Yakovenko conceived and designed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.

Data Availability

The following information was supplied regarding data availability:

The scripts and source data are available at figshare:

Yakovenko, Sergiy (2021): Solving musculoskeletal biomechanics with machine learning. figshare. Software. DOI 10.6084/m9.figshare.14130392.v4.

The results are available in Figures 3–5, and the corresponding data is available in open standard file format, JSON (https://en.wikipedia.org/wiki/JSON). The model parameters are available as code in TXT.

REFERENCES

Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M, Kudlur M, Levenberg J, Monga R, Moore S, Murray DG, Steiner B, Tucker P, Vasudevan V, Warden P, Wicke M, Yu Y, Zheng X. 2016. TensorFlow: a system for large-scale machine learning. In: *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation*. Savannah, GA, USA, 21.

Alber M, Buganza Tepole A, Cannon WR, De S, Dura-Bernal S, Garikipati K, Karniadakis G, Lytton WW, Perdikaris P, Petzold L, Kuhl E. 2019. Integrating machine learning and multiscale modeling—perspectives, challenges, and opportunities in the biological, biomedical, and behavioral sciences. *NPJ Digital Medicine* **2(1)**:115 DOI 10.1038/s41746-019-0193-y.

Bernstein NA. 1967. *The co-ordination and regulation of movements*. Oxford: Pergamon.

Boots MT, Hardesty R, Sobinov A, Gritsenko V, Collinger JL, Fisher LE, Gaunt RA, Yakovenko S. 2020. Functional and structural moment arm validation for musculoskeletal models: a study of the human forearm and hand. *bioRxiv* DOI 10.1101/2020.05.29.124644..

Boots MT, Sobinov A, Gritsenko V, Mansoori BK, Fisher LE, Collinger JL, Gaunt RA, Yakovenko S. 2017. Scaling of musculoskeletal morphometry for human upper-limb models. In: *Society for Neuroscience Abstracts*. Washington, D.C.: Society for Neuroscience Abstracts.

Buchanan TS, Lloyd DG, Manal K, Besier TF. 2004. Neuromusculoskeletal modeling: estimation of muscle forces and joint moments and movements from measurements of neural command. *Journal of Applied Biomechanics* **20(4)**:367–395 DOI 10.1123/jab.20.4.367.

Castellini C, Van der Smagt P. 2009. Surface EMG in advanced hand prosthetics. *Biological Cybernetics* **100(1)**:35–47 DOI 10.1007/s00422-008-0278-1.

Charkhkar H, Christie BP, Triolo RJ. 2020. Sensory neuroprosthesis improves postural stability during sensory organization test in lower-limb amputees. *Scientific Reports* **10(1)**:6984 DOI 10.1038/s41598-020-63936-2.

Collinger JL, Wodlinger B, Downey JE, Wang W, Tyler-Kabara EC, Weber DJ, McMorland AJC, Velliste M, Boninger ML, Schwartz AB. 2013. High-performance neuroprosthetic control by an individual with tetraplegia. *The Lancet* **381(9866)**:557–564 DOI 10.1016/S0140-6736(12)61816-9.

Crouch DL, Huang H. 2016. Lumped-parameter electromyogram-driven musculoskeletal hand model: a potential platform for real-time prosthesis control. *Journal of Biomechanics* **49(16)**:3901–3907 DOI 10.1016/j.jbiomech.2016.10.035.

Cui L, Perreault EJ, Maas H, Sandercock TG. 2008. Modeling short-range stiffness of feline lower hindlimb muscles. *Journal of Biomechanics* **41(9)**:1945–1952 DOI 10.1016/j.jbiomech.2008.03.024.

Dantas H, Warren DJ, Wendelken SM, Davis TS, Clark GA, Mathews VJ. 2019. Deep learning movement intent decoders trained with dataset aggregation for prosthetic limb control. *IEEE Transactions on Biomedical Engineering* **66(11)**:3192–3203 DOI 10.1109/TBME.2019.2901882.

Debicki DB, Gribble PL. 2005. Persistence of inter-joint coupling during single-joint elbow flexions after shoulder fixation. *Experimental Brain Research* **163(2)**:252–257 DOI 10.1007/s00221-005-2229-6.

Downey JE, Brane L, Gaunt RA, Tyler-Kabara EC, Boninger ML, Collinger JL. 2017. Motor cortical activity changes during neuroprosthetic-controlled object interaction. *Scientific Reports* **7(1)**:16947 DOI 10.1038/s41598-017-17222-3.

Furui A, Eto S, Nakagaki K, Shimada K, Nakamura G, Masuda A, Chin T, Tsuji T. 2019. A myoelectric prosthetic hand with muscle synergy–based motion determination and impedance model–based biomimetic control. *Science Robotics* **4(31)**:eaaw6339 DOI 10.1126/scirobotics.aaw6339.

Ganzer PD, Colachis SC, Schwemmer MA, Friedenberg DA, Dunlap CF, Swiftney CE, Jacobowitz AF, Weber DJ, Bockbrader MA, Sharma G. 2020. Restoring the sense of touch using a sensorimotor demultiplexing neural interface. *Cell* **181(4)**:763–773.e12 DOI 10.1016/j.cell.2020.03.054.

Geethanjali P. 2016. Myoelectric control of prosthetic hands: state-of-the-art review. *Medical Devices: Evidence and Research Volume* **9**:247–255 DOI 10.2147/MDER.

George JA, Davis TS, Brinton MR, Clark GA. 2020. Intuitive neuromyoelectric control of a dexterous bionic arm using a modified Kalman filter. *Journal of Neuroscience Methods* **330(1)**:108462 DOI 10.1016/j.jneumeth.2019.108462.

Glorot X, Bengio Y. 2010. Understanding the difficulty of training deep feedforward neural networks. In: *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS)*. Chia Laguna Resort, Sardinia, Italy, 8.

Glorot X, Bordes A, Bengio Y. 2011. Deep sparse rectifier neural networks. In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. 1–9.

Gribble PL, Ostry DJ. 1998. Independent coactivation of shoulder and elbow muscles. *Experimental Brain Research* **123(3)**:355–360 DOI 10.1007/s002210050580.

Gribble PL, Ostry DJ. 1999. Compensation for interaction torques during single- and multijoint limb movement. *Journal of Neurophysiology* **82(5)**:2310–2326 DOI 10.1152/jn.1999.82.5.2310.

Gritsenko V, Hardesty RL, Boots MT, Yakovenko S. 2016. Biomechanical constraints underlying motor primitives derived from the musculoskeletal anatomy of the human arm. *PLOS ONE* **11(10)**:1–18 DOI 10.1371/journal.pone.0164050.

Gupta A, Katarya R. 2020. Social media based surveillance systems for healthcare using machine learning: a systematic review. *Journal of Biomedical Informatics* **108**:103500 DOI 10.1016/j.jbi.2020.103500.

Hartmann C, Boedecker J, Obst O, Ikemoto S, Asada M. 2013. Real-time inverse dynamics learning for musculoskeletal robots based on echo state gaussian process regression. In: *Robotics: Science and Systems VIII*. Sydney, NSW, Australia: University of Sydney, 113–120.

Heller BW, Veltink PH, Rijkhoff NJ, Rutten WL, Andrews BJ. 1993. Reconstructing muscle activation during normal walking: a comparison of symbolic and connectionist machine learning techniques. *Biological Cybernetics* **69(4)**:327–335 DOI 10.1007/bf00203129.

Hughes C, Herrera A, Gaunt R, Collinger J. 2020. Bidirectional brain-computer interfaces. *Handbook of Clinical Neurology* **168**:163–181 DOI 10.1016/B978-0-444-63934-9.00013-5.

Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q, Liu T-Y. 2017. LightGBM: a highly efficient gradient boosting decision tree. In: Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R, eds. *Advances in Neural Information Processing Systems*. Vol. 30. Red Hook: Curran Associates, Inc, 3146–3154.

Kiguchi K, Hayashi Y. 2012. An EMG-based control for an upper-limb power-assist exoskeleton robot. *IEEE Transactions on Systems, Man, and Cybernetics, Part B* **42(4)**:1064–1071 DOI 10.1109/TSMCB.2012.2185843.

Kingma DP, Ba J. 2017. Adam: a method for stochastic optimization. *arXiv. Available at http://arxiv.org/abs/1412.6980*.

Kuo AD. 1998. A least-squares estimation approach to improving the precision of inverse dynamics computations. *Journal of Biomechanical Engineering* **120(1)**:148–159 DOI 10.1115/1.2834295.

LeCun Y, Bengio Y, Hinton G. 2015. Deep learning. *Nature* **521(7553)**:436–444 DOI 10.1038/nature14539.

Liu MM, Herzog W, Savelberg HH. 1999. Dynamic muscle force predictions from EMG: an artificial neural network approach. *Journal of Electromyography and Kinesiology: Official Journal of the International Society of Electrophysiological Kinesiology* **9(6)**:391–400 DOI 10.1016/S1050-6411(99)00014-0.

Mansouri M, Yakovenko S, Gritsenko V, Boots MT, Sobinov A, Beringer C, Boninger ML, Fisher LE, Collinger JL, Gaunt RA. 2017. Using biomimetic models and intramuscular EMG for control of myoelectric prostheses. In: *American Society of Biomechanics*. Boulder, Colorado1–2.

McGinley JL, Baker R, Wolfe R, Morris ME. 2009. The reliability of three-dimensional kinematic gait measurements: a systematic review. *Gait & Posture* **29(3)**:360–369 DOI 10.1016/j.gaitpost.2008.09.003.

McGovern A, Lagerquist R, John Gagne D, Jergensen GE, Elmore KL, Homeyer CR, Smith T. 2019. Making the black box more transparent: understanding the physical implications of machine learning. *Bulletin of the American Meteorological Society* **100(11)**:2175–2199 DOI 10.1175/BAMS-D-18-0195.1.

McKinney SM, Sieniek M, Godbole V, Godwin J, Antropova N, Ashrafian H, Back T, Chesus M, Corrado GC, Darzi A, Etemadi M, Garcia-Vicente F, Gilbert FJ, Halling-Brown M, Hassabis D, Jansen S, Karthikesalingam A, Kelly CJ, King D, Ledsam JR, Melnick D, Mostofi H, Peng L, Reicher JJ, Romera-Paredes B, Sidebottom R, Suleyman M, Tse D, Young KC, De Fauw J, Shetty S. 2020. International evaluation of an AI system for breast cancer screening. *Nature* **577(7788)**:89–94 DOI 10.1038/s41586-019-1799-6.

McNamee D, Wolpert DM. 2019. Internal models in biological control. *Annual Review of Control, Robotics, and Autonomous Systems* **2(1)**:339–364 DOI 10.1146/annurev-control-060117-105206.

Microsoft Corporation. 2020. LightGBM. *Available at https://lightgbm.readthedocs.io* (accessed 15 April 2020).

Muceli S, Farina D. 2012. Simultaneous and proportional estimation of hand kinematics from EMG during mirrored movements at multiple degrees-of-freedom. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **20(3)**:371–378 DOI 10.1109/TNSRE.2011.2178039.

Murphy AC, Muldoon SF, Baker D, Lastowka A, Bennett B, Yang M, Bassett DS. 2018. Structure, function, and control of the human musculoskeletal network. *PLOS Biology* **16(1)**:e2002811 DOI 10.1371/journal.pbio.2002811.

Natekin A, Knoll A. 2013. Gradient boosting machines: a tutorial. *Frontiers in Neurorobotics* **7**:21 DOI 10.3389/fnbot.2013.00021.

Patla AE. 1985. Some characteristics of EMG patterns during locomotion: implications for the locomotor control process. *Journal of Motor Behavior* **17(4)**:443–461 DOI 10.1080/00222895.1985.10735360.

**Richards BA, Lillicrap TP, Beaudoin P, Bengio Y, Bogacz R, Christensen A, Clopath C, Costa RP, De Berker A, Ganguli S, Gillon CJ, Hafner D, Kepecs A, Kriegeskorte N, Latham P, Lindsay GW, Miller KD, Naud R, Pack CC, Poirazi P, Roelfsema P, Sacramento J,

Saxe A, Scellier B, Schapiro AC, Senn W, Wayne G, Yamins D, Zenke F, Zylberberg J, Therien D, Kording KP. 2019. A deep learning framework for neuroscience. *Nature Neuroscience* **22(11)**:1761–1770 DOI 10.1038/s41593-019-0520-2.

Sartori M, Durandau G, Došen S, Farina D. 2018. Robust simultaneous myoelectric control of multiple degrees of freedom in wrist-hand prostheses by real-time neuromusculoskeletal modeling. *Journal of Neural Engineering* **15(6)**:066026 DOI 10.1088/1741-2552/aae26b.

Sartori M, Fernandez JW, Modenese L, Carty CP, Barber LA, Oberhofer K, Zhang J, Handsfield GG, Stott NS, Besier TF, Farina D, Lloyd DG. 2017. Toward modeling locomotion using electromyography-informed 3D models: application to cerebral palsy. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine* **9(2)**:e1368 DOI 10.1002/wsbm.1368.

Sazli MH. 2006. A brief review of feed-forward neural networks. *Communications, Faculty of Science, University of Ankara* **50(1)**:11–17 DOI 10.1501/commua1-2_0000000026.

Sepulveda F, Wells DM, Vaughan CL. 1993. A neural network representation of electromyography and joint dynamics in human gait. *Journal of Biomechanics* **26(2)**:101–109 DOI 10.1016/0021-9290(93)90041-C.

Seth A, Hicks JL, Uchida TK, Habib A, Dembia CL, Dunne JJ, Ong CF, DeMers MS, Rajagopal A, Millard M, Hamner SR, Arnold EM, Yong JR, Lakshmikanth SK, Sherman MA, Ku JP, Delp SL. 2018. OpenSim: simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement. *PLOS Computational Biology* **14(7)**:e1006223 DOI 10.1371/journal.pcbi.1006223.

Shen L, Margolies LR, Rothstein JH, Fluder E, McBride R, Sieh W. 2019. Deep learning to improve breast cancer detection on screening mammography. *Scientific Reports* **9(1)**:12495 DOI 10.1038/s41598-019-48995-4.

Shi X, Wong YD, Chai C, Li MZ-F. 2020. An automated machine learning (AutoML) method of risk prediction for decision-making of autonomous vehicles. Epub ahead of print 1 July 2020. *IEEE Transactions on Intelligent Transportation Systems* 1–10 DOI 10.1109/TITS.2020.3002419.

Snoek J, Larochelle H, Adams RP. 2012. Practical Bayesian optimization of machine learning algorithms. In: *Advances in Neural Information Processing Systems*. **25**:2951–2959.

Sobinov A, Boots MT, Gritsenko V, Fisher LE, Gaunt RA, Yakovenko S. 2020. Approximating complex musculoskeletal biomechanics using multidimensional autogenerating polynomials. *PLOS Computational Biology* **16(12)**:e1008350 DOI 10.1371/journal.pcbi.1008350.

Song R, Tong KY. 2005. Using recurrent artificial neural network model to estimate voluntary elbow torque in dynamic situations. *Medical & Biological Engineering & Computing* **43(4)**:473–480 DOI 10.1007/BF02344728.

Ting J, Farina D, Weber DJ, Del Vecchio A, Friedenberg D, Liu M, Schoenewald C, Sarma D, Collinger J, Colachis S, Sharma G. 2019. A wearable neural interface for detecting and decoding attempted hand movements in a person with tetraplegia. In: *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. Berlin, Germany: IEEE.

Todorov E, Erez T, Tassa Y. 2012. MuJoCo: a physics engine for model-based control. In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Piscataway: IEEE, 5026–5033.

Valero-Cuevas FJ. 2015. *Fundamentals of neuromechanics*. London: Springer.

Wang Lin, Buchanan TS. 2002. Prediction of joint moments using a neural network model of muscle activations from EMG signals. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **10(1)**:30–37 DOI 10.1109/TNSRE.2002.1021584.

Wendelken S, Page DM, Davis T, Wark HAC, Kluger DT, Duncan C, Warren DJ, Hutchinson DT, Clark GA. 2017. Restoration of motor control and proprioceptive and cutaneous sensation in humans with prior upper-limb amputation via multiple Utah Slanted Electrode Arrays (USEAs) implanted in residual peripheral arm nerves. *Journal of NeuroEngineering and Rehabilitation* **14(1)**:121 DOI 10.1186/s12984-017-0320-4.

Wolpert DM, Ghahramani Z. 2000. Computational principles of movement neuroscience. *Nature Neuroscience* **3**:1212–1217.

Xia P, Hu J, Peng Y. 2018. EMG-based estimation of limb movement using deep learning with recurrent convolutional neural networks: EMG-BASED ESTIMATION OF LIMB MOVEMENT. *Artificial Organs* **42(5)**:E67–E77 DOI 10.1111/aor.13004.

Zajac FE. 1989. Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control. *Critical Reviews in Biomedical Engineering* **17**:359–411.

Zhang Y, Nieveen J, Wendelken S, Page D, Davis T, Bo APL, Hutchinson D, Clark GA, Warren DJ, Zhang C, Mathews VJ. 2017. Individual hand movement detection and classification using peripheral nerve signals. In: *2017 8th International IEEE/EMBS Conference on Neural Engineering (NER)*. Shanghai, China: IEEE, 448–451.