1 Introduction

Joint moments are commonly calculated in biomechanics research and provide an indirect measure of muscular behaviors and joint loads (Richards et al., 2018). Joint moments have been used to understand motor control strategies (Toney and Chang, 2016), injury risk (Myer et al., 2015), disease progression (Henriksen et al., 2014); and also used to calculate measures that quantify the mechanical energetics of movement (Aleshinsky, 1986a, b).

Joint moments can be calculated using either a bottom-up or top-down approach. The more common bottom-up approach requires measuring ground reaction forces (GRF) from force plates (most commonly embedded in-ground) and partial-body segment kinematics captured using an optoelectronic system. The less common top-down approach requires capturing whole-body kinematics only using an optoelectronic system (Ren et al., 2008). Both approaches to calculating joint moments are limited by their lack of scalability. The requirement for force plates makes a bottom-up approach challenging to undertake outside the laboratory. Compared to the bottom-up approach, the top-down approach has greater limitations such as the greater influence of skin artefact and errors in body segment parameters (especially of the heavier trunk segment) (Ren et al., 2008).

To quantify joint moments outside the biomechanics laboratory, researchers have begun coupling kinematic-based features with machine learning to predict these measures (Johnson et al., 2019b; Liu et al., 2009; Stetter et al., 2020) – which we term, the Kinematic$_{ML}$ approach. Ideally, the Kinematic$_{ML}$ predictors should be derived from the most parsimonious number of kinematic segments which can be measured using either optoelectronic systems or inertial measurement units (IMUs). Existing studies have used the Kinematic$_{ML}$ approach during walking

(Wang et al., 2020), vertical jumping (Liu et al., 2009), running (Stetter et al., 2020), turning (Stetter et al., 2020), and side-step cutting (Johnson et al., 2019b; Stetter et al., 2020). ML methods range from shallow learning techniques such as artificial neural networks (ANN) with one/two "hidden" layers (Liu et al., 2009; Stetter et al., 2020) and boosting (Wang et al., 2020), to deep learning techniques such as deep neural networks (DNN) (Boswell et al., 2021; Wang et al., 2020). A limitation in most shallow learning methods for biomechanics is that they cannot accommodate time-varying variables as predictors and outcomes. To circumvent this limitation, researchers have opted to treat each value of a time-series as independent observations (Stetter et al., 2020), which ignores the inherent correlation in time-varying biomechanics data.

At a minimum, the performance of the $\text{Kinematic}_{\text{ML}}$ approach should be less than the intrinsic measurement error of traditional inverse-dynamics (< 0.31 Nm/kg) (Wilken et al., 2012). The predictive accuracy (root mean squared error [RMSE]) of the $\text{Kinematic}_{\text{ML}}$ approach ranged from 0.12 to 0.28 Nm/kg (Liu et al., 2009) and 0.26 to 1.13 Nm/kg using ANN (Stetter et al., 2020), and below 0.26 Nm/kg using either boosting or DNN (Wang et al., 2020). The biggest challenge associated with the $\text{Kinematic}_{\text{ML}}$ approach is the well-established issue that ML techniques are "data-hungry". Current studies have used data from a very small cohort of 10 (Liu et al., 2009) to a larger cohort of 106 participants (Wang et al., 2020). However, even though a sample size of 100 participants is considered large clinically, it pales in comparison to non-clinical ML research (e.g. millions of samples (Simonyan and Zisserman, 2015)). An issue when the sample size is small is that the ML model overfits and generalizes poorly for new observations.

A novel ML method to manage the issue of small sample sizes is transfer learning, an extension of deep learning (Weiss et al., 2016). Transfer learning takes advantage of "knowledge" from existing large pre-trained ML models, with the collected biomechanical data used for fine-tuning (Johnson et al., 2019b). Pre-trained models exist for various neural network architectures, such as a pre-trained VGG network, which was trained on 1.3 million ImageNet images and 1000 object classes (Simonyan and Zisserman, 2015). During walking, transfer learning achieved a Pearson correlation of 0.94 to 0.97 (Johnson et al., 2019b), which was similar to the correlation of 0.96 when using DNN (Wang et al., 2020) for predicting knee abduction/addition moments. However, transfer learning was superior to the performance of using ANN which achieved a correlation of 0.71 (Stetter et al., 2020). This suboptimal performance could be due to ANN not being designed to accommodate time-varying variables.

Given that previous ML models were trained on different participants, with different inputs and sample sizes, it is difficult to accurately benchmark the relative merits of different ML algorithms in joint moments prediction. The present study aimed to compare the predictive accuracy of three different ML techniques that can accommodate both time-varying variables as predictors and as an outcome (ML method – functional regression [$ML_{fregress}$], a deep neural network built from scratch [ML_{DNN}], and transfer learning [ML_{TL}]). A previous study reported superior performance in transfer learning compared to shallow learning, in predicting both the three-dimensional shear and rotational moment values of GRF (Johnson et al., 2019a). This suggests that transfer learning may be the best ML technique when the sample size is limited regardless of the type of biomechanical outcomes investigated. Hence, we hypothesized that the predictive accuracy will be greatest for $ML_{TL}$and least for $ML_{fregress}$ across all nine lower-limb joint moment outcomes.

2 Methods

Data for this analysis came from three sources – a publicly available running dataset (Fukuchi et al., 2017), and two datasets from the lead author's research on load carriage running (Liew et al., 2016a; Liew et al., 2016b) (Figure. 1).

*** Insert Figure 1***

2.1 Study (Fukuchi) (Fukuchi et al., 2017)

Data for the current study came from a publicly available dataset on running (n = 28) in healthy adults (Fukuchi et al., 2017). Running assessment was performed using a dual-belt, force-instrumented treadmill (300 Hz; Bertec, USA), and the motion was captured with 12 optoelectronic cameras (150Hz; Motion Analysis Corporation, USA) (Fukuchi et al., 2017). Participants performed shod running across three fixed speeds of 2.5 m/s, 3.5 m/s, and 4.5 m/s (Fukuchi et al., 2017). Marker trajectories and GRF were collected for 30 s and the data were low passed filtered at a matched frequency of 12 Hz (4^{th} Order, zero-lag, Butterworth) (Kristianslund et al., 2012). Biomechanical modeling was performed in Visual 3D software (C-motion Inc., Germantown, MD, USA). A force plate threshold of 50 N was used to determine gait events of initial contact and toe-off. A seven-segment lower limb inertial model was created (Fukuchi et al., 2017).

2.2 Study one (Liew_study1)

Data came from a previously published work investigating the effects of load carriage on running biomechanics (n = 31) (Liew et al., 2016b). The protocol involved participants running across a 20 m runway, embedded with force platforms (AMTI, Watertown, MA), while carrying three load conditions (0%, 10%, 20% body weight (BW)) across three velocities (3.0 m/s, 4.0

m/s, 5.0 m/s) – the order of which was randomized (Liew et al., 2016b). Participants wore their running shoes and attire during the experiment. Each condition required five successful running attempts, which was defined as meeting the prescribed velocity within a ±10% variation, with no visible alteration to running gait (Liew et al., 2016b).

Kinematic data were captured using an 18 camera motion capture system (Vicon T-series, Oxford Metrics, UK) (250 Hz). GRF was measured using synchronized in-ground force plates (2000 Hz). Data processing was performed in Visual 3D. Marker trajectories and GRF were filtered at 18 Hz (4th order, zero-lag, Butterworth) (Robinson et al., 2014). A seven-segment lower limb inertial model was created (Liew et al., 2016b). A force plate threshold of 20 N was used to determine gait events of initial contact and toe-off.

2.3 Studies two and three (Liew_study2pre & Liew_study2post)

Data came from a project investigating the influence of strength training on load carriage on running biomechanics (n = 31) (Liew et al., 2016a). This dataset is independent from that reported in study one (above). Participants performed repeated overground running at a fixed velocity of 3.5 m/s (±10%) while carrying two load conditions (0%, 20% BW) (Liew et al., 2016a). A run-up distance of 20 m was given before the first force plate, and a tail-off distance of 10 m was given after the last force plate. Participants performed a minimum of five successful over-ground running trials at 3.5 m/s. Motion capture equipment, signal processing, and biomechanical modelling procedures were performed identical to *Liew_study1*.

2.4 Common biomechanical processing

For all studies, three-dimensional (3D joint angle, velocity, acceleration, and internal moment, of the bilateral ankle, knee, and hip joints were collected. The joint angle was calculated using a Cardan flexion-abduction-rotation sequence (Cole et al., 1993). Joint velocity, acceleration, and moment were expressed in the proximal segment's reference frame (Schache and Baker, 2007). All biomechanical variables were time normalized to 101 data points within the stance phase of each lower limb. The joint moment was normalized to body mass (N/kg).

2.5 Machine learning

All analyses were conducted in R software (version 4.0.2) and Python (version 3.6.12), with associated codes and result found online (https://bernard-liew.github.io/2020_fun_regress/). The following packages were used: *refund* for functional regression (Goldsmith et al., 2020), *reticulate* which provides an R interface to Python (Ushey et al., 2021), *imager* for image preprocessing (Barthelme, 2020), *keras* (Allaire and Chollet, 2020) and *TensorFlow* (Allaire and Tang, 2020) for DNN.

2.5.1 Pre-processing

For all studies, multiple running steps within each subject-load-speed combination were averaged to produce one waveform per variable. There were 121 unique study-participant combinations (Fukuchi- 28 participants, Liew study 1 – 31 participants, Liew study 2 pre – 31 participants, Liew study 2 post – 31 participants). The data was split whereby 80% of the 121 study participants' data were used for model training, and a separate 20% was used for testing of the model's predictive performance. All predictors in the training set were demeaned and scaled to one standard deviation. All predictors in the test set were demeaned and scaled using the parameters from the training set.

For ML_{DNN} and ML_{TL}, the 3D time-series predictors in the training set were originally organized into a 4-dimensional array (490 [observations] × 101 [gait cycle] × 9 [3 joints & 3 variables] × 3 [axes]), whilst that of the testing set was organized similarly, apart from the number of observations different (120 × 101 × 9 × 3). For each outcome, the data was organized into a 2-dimensional matrix for both the training (490 × 101) and testing (120 × 101) sets. To leverage pre-trained image models for fine-tuning, the study's 3D kinematic predictors needed to be converted to a set of static color (Red, Green, Blue) images (Johnson et al., 2019b). This was achieved by mapping the 101 gait cycle points to the image height, the nine kinematic variables to the image width, and the axes of each kinematic variable to the image additive color model. The resultant 101 × 9-pixel images were warped to 150 × 150 pixels using cubic spline interpolation, to suit the input dimension of the pre-trained image model used in the present study.

2.5.2 Functional regression ($ML_{fregress}$)

A classical approach to deal with functional inputs (gait cycles) and functional outcomes (joint moments) is function-on-function regression (Scheipl et al., 2015). Initially motivated as a statistical regression technique, the approach is frequently used in machine learning (Liew et al., 2019; Rügamer et al., 2018), in particular with the combination with boosting (Brockhaus et al., 2020). We employed penalized function-on-function regression (*pffr*) by including the gait cycle of all 9 joint-variable combinations as a functional predictor and modeled each of the functional outputs (joint moments) using an individual model. Based on a cubic tensor product B-spline with marginal second differences penalties, the $ML_{fregress}$ model estimated the relationship between predictors and output using a two-dimensional smooth nonlinear function.

2.5.3 Deep neural network (ML_{DNN})

We designed a CNN with three convolution blocks followed by a network head that uses the features learned in the convolution to predict the 101-dimensional output (Table 1). Each convolution block consists of a 2D-convolution (32, 64, and 128 filters, all with kernel sizes of 3×3), a ReLU activation, batch normalization, a 2D-max pooling, and a dropout layer with a rate of 0.25. To generate the 101-dimensional output, an output layer with 101 units and linear activation was finally used. We trained the network using 200 epochs, a batch size of 16, "RMSprop" as the optimizer, and mean squared error (MSE) as the loss function between the observed and predicted outcomes.

*** Insert Table 1***

2.5.4 Transfer learning (ML_{TL})

In this study, we used the VGG-16 model (Visual Geometry Group, Oxford, UK) that was pre-trained on 1,000,000 images dataset from ImageNet and achieved state-of-the-art results in object recognition (Simonyan and Zisserman, 2015). The VGG-16 model contains 13 convolutional layers and three fully connected layers. We added to the convolutional layers a series of fully connected, dropout, and batch normalization layers (Table 2). We froze the weights of the 13 convolutional layers of the VGG-16 model and only fine-tuned the weights of the added layers. We trained the network using 200 epochs, a batch size of 16, "RMSprop" as the optimizer, and mean squared error as the loss function.

*** Insert Table 2***

2.6 Predictive accuracy

Accuracy was quantified by comparing the nine joint moments in the test set, against their predicted values using one absolute index - both RMSE; and two relative indices - relative RMSE (relRMSE) expressed as a percentage (%) of the average peak-to-peak amplitude for the outcomes (Ren et al., 2008), and Pearson correlation coefficient (cor) (Johnson et al., 2019a; Johnson et al., 2019b).

$$RMSE = \sqrt{\frac{\int_0^T [u_{obs}(t) - u_{pred}(t)]^2 dt}{T}} \qquad (1)$$

$$relRMSE = \frac{RMSE}{0.5[\sum_{i=1}^{2}\left(max_{0<t<T}\left(u_i(t)\right) - min_{0<t<T}\left(u_i(t)\right)\right)]}\ x\ 100\% \qquad (2)$$

where T represents the stance duration between initial contact and toe-off, $u_{obs}(t)$ represents the value at the t^{th}time point of the observed outcome, $u_{pred}(t)$ represents the value at the t^{th}time point of the predicted outcome, and i represents either the observed or predicted outcomes.

3 Results

Basic descriptive characteristics of the cohort can be found in Table 3. The mean waveform plots of all kinematic and kinetic variables of the entire dataset can be found in the supplementary material (SM Fig. 1). The observed and predicted mean waveform for each of the nine outcomes are presented in Fig. 2. Prediction performance was generally the best using ML_{DNN}, and the worse using $ML_{fregress}$(Table 4). The average RMSE (minimum to maximum) for $ML_{fregress}$ was 0.31 Nm/kg (0.16-0.54 Nm/kg), ML_{DNN} was 0.14 Nm/kg (0.06-0.25 Nm/kg), and ML_{TL} was 0.18 Nm/kg (0.07-0.31 Nm/kg) (Table 4). The average relRMSE (minimum and maximum) for $ML_{fregress}$ was 27.4% (13.0-57.0%), ML_{DNN} was 15.1% (5.0-37.0%), and ML_{TL} was 18.6% (7.0-41.0%) (Table 4). The average correlation (minimum and maximum) for $ML_{fregress}$ was 0.82 (0.2-0.98), ML_{DNN} was 0.93 (0.64-1.00), and ML_{TL} was 0.89 (0.52-0.99) (Table 4).

*** Insert Figure 2***

*** Insert Table 3***

*** Insert Table 4***

On average across all outcomes, ML_{DNN} improved RMSE, relRMSE, and correlation by 0.16 Nm/kg, 12.3%, and 0.11, compared to $ML_{fregress}$, respectively (Table 4). ML_{DNN} improved RMSE, relRMSE, and correlation by 0.04Nm/kg, 3.4%, and 0.04, compared to ML_{TL}, respectively (Table 4). ML_{TL} improved RMSE, relRMSE, and correlation by 0.13 Nm/kg, 8.9%, and 0.07, compared to $ML_{fregress}$, respectively (Table 4).

4 Discussion

In the present study, we compared three different ML techniques to predict lower-limb joint moments during running. Contrary to our hypothesis, ML_{DNN} was the best performing technique, ML_{TL} came second, and $ML_{fregress}$ was the poorest performing technique.

In the present study, the predictive performance of our ML_{TL}approach matched that of a previous study (Johnson et al., 2019b), despite differences in the predictors used (e.g. marker accelerations), and type of pre-trained model used for transfer learning. Johnson et al. (Johnson et al., 2019b) reported a relRMSE of 7.8-13.5% for flexion-extension, 25.3-31.7% for adduction-abduction, and 24.3-27.3% for internal-external rotation knee moments. This was close to our prediction performance achieved using ML_{TL}, which was 8%, 31% and 19%, respectively for the knee moments (Table 2). A previous study reported that ML_{TL}outperformed a shallow learner (partial least square) in predicting GRF values (Johnson et al., 2019a), a finding replicated in the present study.

Surprisingly, a custom-designed DNN (ML_{DNN}) resulted in consistently superior predictive performance than ML_{TL}, despite the relatively small sample size of the present study. In the present ML_{TL}, we used a single pre-trained image model (VGG16) (Simonyan and Zisserman, 2015), rather than comparing different pre-trained models on prediction performance. The CaffeNet model used in previous studies (Johnson et al., 2019a; Johnson et al., 2019b) has only five convolutional layers but up to 60 million parameters (Krizhevsky et al., 2012), compared to the presently used VGG16 model (Simonyan and Zisserman, 2015), which has 13 convolutional layers and up to 14 million parameters. In image classification studies, the advantage of deeper networks is that they can learn features at various levels of abstraction, making them ideal in generalizing their predictions to an external context. A study comparing ML_{TL}using three pre-trained models (CaffeNet, AlexNet, and GooLeNet) to predict GRF, reported a difference in relRMSE between models of 3-5% (Johnson et al., 2019a). Given that the present study reported that relRMSE of ML_{TL} was between 2% and 9% greater than that of ML_{DNN}, using a different pre-trained image model may enhance the predictive performance of transfer learning.

Another reason that $ML_{DNN}$was superior toML_{TL}, could be that the pre-trained image weights were adding noise to the model. This may not be surprising given that pre-trained image models have been trained on images of non-biomechanical objects (e.g. animals) (Simonyan and Zisserman, 2015). A previous study reported improvements in knee joint moments' prediction when two levels of fine-tuning occurred (e,g. relRMSE of 29.5%) using two independent biomechanics datasets, rather than one level (e,g. relRMSE of 31.7%) (Johnson et al., 2019b). It may be that pre-trained weights should be derived from a model trained specifically on biomechanics data for transfer learning to leverage upon, during the fine-tuning process. Given that open-access biomechanics datasets are increasingly common (Fukuchi et al., 2018; Fukuchi et al., 2017), these data could potentially be leveraged to build a preliminary biomechanics-specific DNN model, before the actual fine-tuning using the experimental dataset. Future investigations are warranted to evaluate the differences in prediction performance using transfer learning using a non-biomechanics-specific, biomechanics-specific but on different movement tasks, and a biomechanics-specific and movement-specific pre-trained model.

For any ML technique, pre-processing plays an influential role in determining its predictive performance. Presently, we encoded our 3D time-series predictors into static images using cubic spline interpolation (Johnson et al., 2019a; Johnson et al., 2019b). The purpose of interpolating our data was so that it fitted the input dimensions of the VGG16 image model used for transfer learning. By "stretching" the data to fit the required input dimension, noise could have been introduced into our predictors. It is anticipated that the greater the amount of "stretching" required, the greater the level of noise introduced into the data. In the present study, our 101×9-pixel image was warped to a dimension of 150×150, which could mean that the width of our image could have been significantly distorted by noise. Other methods of encoding time-series into images could include transforming time series into polar coordinates via Gramian Angular Field (GMAF) (Wang and Oates, 2015). A previous study that used GMAR to encode inertial measurement unit (IMU) time-series data into images for activity recognition, reported achieving an accuracy of more than 98% (Boukhennoufa et al., 2021). Future studies comparing different time-series pre-processing techniques should be performed in evaluating its impact on using DNN in predicting joint moments.

It may be that because we investigated straight-line running, the signal-to-noise ratio of sagittal plane moments is higher than non-sagittal plane moments, making it easier to predict.

That was why we found that sagittal plane moments were generally predicted more accurately than non-sagittal plane moments. Whether the performance errors in any ML approaches are acceptable would be dependent on factors such as the joints, joint axes, direction of motion, and the intended usage of the outcome. Until there is an established magnitude of clinically significant errors in joint moment measurement, more accurate methods must be pursued in future studies.

This study is not without limitations. First, we did not perform hyperparameter tuning. Given that functional regression has no intrinsic hyperparameters, we wanted to compare all three models using their “default” settings. Hence, our findings can be said to provide a more conservative estimate of the predictive performance of deep and transfer learning models. Second, we used predictors derived from optoelectronic systems, which can still be time-consuming to use in the clinics. Wearable sensors or markerless motion capture represent the most clinically feasible methods of measuring body motions. Whether the performance of the $\text{Kinematic}_{\text{ML}}$ approach using these newer technologies would match that of traditional optoelectronic systems needs to be investigated.

5 Conclusions

DNN with or without transfer learning was superior in predicting joint moments compared to functional regression. A custom DNN model was superior to transfer learning, using an existing pre-trained image model. To leverage transfer learning in predicting biomechanical variables, a pre-trained biomechanics-specific image model may be needed. Synergising ML with kinematic inputs has the potential to allow an in-depth biomechanical analysis of movement data obtained in the field.

6 References

Aleshinsky, S.Y., 1986a. An energy 'sources' and 'fractions' approach to the mechanical energy expenditure problem--I. Basic concepts, description of the model, analysis of a one-link system movement. J Biomech 19, 287-93.

Aleshinsky, S.Y., 1986b. An energy 'sources' and 'fractions' approach to the mechanical energy expenditure problem--IV. Criticism of the concept of 'energy transfers within and between links'. J Biomech 19, 307-9.

Allaire, J., Chollet, F., 2020. keras: R Interface to 'Keras'. R package version 2.3.0.0. https://CRAN.R-project.org/package=keras.

Allaire, J., Tang, Y., 2020. tensorflow: R Interface to 'TensorFlow'. R package version 2.2.0. https://CRAN.R-project.org/package=tensorflow.

Barthelme, S., 2020. imager: Image Processing Library Based on 'CImg'. R package version 0.42.3. https://CRAN.R-project.org/package=imager.

Boswell, M.A., Uhlrich, S.D., Kidziński, Ł., Thomas, K., Kolesar, J.A., Gold, G.E., . . . Delp, S.L., 2021. A neural network to predict the knee adduction moment in patients with osteoarthritis using anatomical landmarks obtainable from 2D video analysis. Osteoarthritis and Cartilage 29, 346-56.

Boukhennoufa, I., Zhai, X., McDonald-Maier, K., Utti, V., Jackson, J., 2021. Improving the activity recognition using GMAF and transfer learning in post-stroke rehabilitation assessment. In Proceedings of the IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI).

Brockhaus, S., Rügamer, D., Greven, S., 2020. Boosting Functional Regression Models with FDboost. Journal of Statistical Software; Vol 1, Issue 10 (2020).

Cole, G.K., Nigg, B.M., Ronsky, J.L., Yeadon, M.R., 1993. Application of the joint coordinate system to three-dimensional joint attitude and movement representation: A standardization proposal. J Biomech Eng 115, 344-9.

Fukuchi, C.A., Fukuchi, R.K., Duarte, M., 2018. A public dataset of overground and treadmill walking kinematics and kinetics in healthy individuals. PeerJ 6, e4640.

Fukuchi, R.K., Fukuchi, C.A., Duarte, M., 2017. A public dataset of running biomechanics and the effects of running speed on lower extremity kinematics and kinetics. PeerJ 5, e3298.

Goldsmith, J., Scheipl, F., Huang, L., Wrobel, J., Chongzhi, D., Gellar, J., . . . Reiss, P., 2020 refund: Regression with Functional Data. R package version 0.1-23. https://CRAN.R-project.org/package=refund.

Henriksen, M., Creaby, M.W., Lund, H., Juhl, C., Christensen, R., 2014. Is there a causal link between knee loading and knee osteoarthritis progression? A systematic review and meta-analysis of cohort studies and randomised trials. BMJ open 4, e005368.

Johnson, W.R., Alderson, J., Lloyd, D., Mian, A., 2019a. Predicting Athlete Ground Reaction Forces and Moments From Spatio-Temporal Driven CNN Models. IEEE Trans Biomed Eng 66, 689-94.

Johnson, W.R., Mian, A., Lloyd, D.G., Alderson, J.A., 2019b. On-field player workload exposure and knee injury risk monitoring via deep learning. J Biomech 93, 185-93.

Kristianslund, E., Krosshaug, T., van den Bogert, A.J., 2012. Effect of low pass filtering on joint moments from inverse dynamics: Implications for injury prevention. J Biomech 45, 666-71.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems 25, 1097-105.

Liew, B., Rugamer, D., De Nunzio, A., Falla, D., 2019. Interpretable machine learning models for classifying low back pain status using functional physiological variables, 2 ed, Mendeley Data, DOI: 10.17632/stbx779nt6.2.

Liew, B.X., Morris, S., Keogh, J.W., Appleby, B., Netto, K., 2016a. Effects of two neuromuscular training programs on running biomechanics with load carriage: A study protocol for a randomised controlled trial. BMC Musculoskelet Disord 17, 445.

Liew, B.X.W., Morris, S., Netto, K., 2016b. The effects of load carriage on joint work at different running velocities. J Biomech 49, 3275-80.

Liu, Y., Shih, S.M., Tian, S.L., Zhong, Y.J., Li, L., 2009. Lower extremity joint torque predicted by using artificial neural network during vertical jump. J Biomech 42, 906-11.

Myer, G.D., Ford, K.R., Di Stasi, S.L., Foss, K.D., Micheli, L.J., Hewett, T.E., 2015. High knee abduction moments are common risk factors for patellofemoral pain (PFP) and anterior cruciate ligament (ACL) injury in girls: is PFP itself a predictor for subsequent ACL injury? Br J Sports Med 49, 118-22.

Ren, L., Jones, R.K., Howard, D., 2008. Whole body inverse dynamics over a complete gait cycle based only on measured kinematics. J Biomech 41, 2750-9.

Richards, R.E., Andersen, M.S., Harlaar, J., van den Noort, J.C., 2018. Relationship between knee joint contact forces and external knee joint moments in patients with medial knee osteoarthritis: effects of gait modifications. Osteoarthritis Cartilage 26, 1203-14.

Robinson, M.A., Donnelly, C.J., Tsao, J., Vanrenterghem, J., 2014. Impact of knee modeling approach on indicators and classification of anterior cruciate ligament injury risk. Med Sci Sports Exerc 46, 1269-76.

Rügamer, D., Brockhaus, S., Gentsch, K., Scherer, K., Greven, S., 2018. Boosting factor-specific functional historical models for the detection of synchronization in bioelectrical signals. Journal of the Royal Statistical Society: Series C (Applied Statistics) 67, 621-42.

Schache, A.G., Baker, R., 2007. On the expression of joint moments during gait. Gait Posture 25, 440-52.

Scheipl, F., Staicu, A.-M., Greven, S., 2015. Functional Additive Mixed Models. J Comput Graph Stat 24, 477-501.

Simonyan, K., Zisserman, A., 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition.

Stetter, B.J., Krafft, F.C., Ringhof, S., Stein, T., Sell, S., 2020. A Machine Learning and Wearable Sensor Based Approach to Estimate External Knee Flexion and Adduction Moments During Various Locomotion Tasks. Front Bioeng Biotechnol 8, 9.

Toney, M.E., Chang, Y.H., 2016. The motor and the brake of the trailing leg in human walking: leg force control through ankle modulation and knee covariance. Exp Brain Res 234, 3011-23.

Ushey, K., Allaire, J., Tang, Y., 2021. reticulate: Interface to 'Python'. R package version 1.18-9006. https://github.com/rstudio/reticulate.

Wang, C., Chan, P.P.K., Lam, B.M.F., Wang, S., Zhang, J.H., Chan, Z.Y.S., . . . Cheung, R.T.H., 2020. Real-Time Estimation of Knee Adduction Moment for Gait Retraining in Patients With Knee Osteoarthritis. IEEE Trans Neural Syst Rehabil Eng 28, 888-94.

Wang, Z., Oates, T., 2015. Imaging time-series to improve classification and imputation, Proceedings of the 24th International Conference on Artificial Intelligence. AAAI Press, Buenos Aires, Argentina, pp. 3939–45.

Weiss, K., Khoshgoftaar, T.M., Wang, D., 2016. A survey of transfer learning. Journal of Big Data 3, 9.

Wilken, J.M., Rodriguez, K.M., Brawner, M., Darter, B.J., 2012. Reliability and Minimal Detectible Change values for gait kinematics and kinetics in healthy adults. Gait Posture 35, 301-7.