Ref. Ares(2021)2700855 - 22/04/2021

PROJECT DELIVERABLE REPORT

Project Title:

Advanced personalised, multi-scale computer models preventing osteoarthritis

SC1-PM-17-2017 - Personalised computer models and in-silico systems for well-being

Deliverable number	**9.1**
Deliverable title	Evaluation Toolbox Documentation
Submission month of deliverable	38
Issuing partner	HULAFE
Contributing partners	ALL
Dissemination Level (PU/PP/RE/CO):	
Project coordinator	University of Nicosia (UNIC)
Tel:	**+357 22 841 528**
Fax:	**+357 22 357481**
Email:	**felekkis.k@unic.ac.cy& giannaki.c@unic.ac.cy**
Project web site address	www.oactive.eu

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 777159

Revision History

Version	Date	Responsible	Description/Remarks/Reason for changes
1.0	2020/06/29	HULAFE	First Draft
1.1	2021/02/08	HULAFE	Review of First Draft
2.0	2021/02/21	CERTH	Second Draft
2.1	2021/04/20	LJMU, HULAFE	Review of Second Draft

Table of Contents

Abbreviations

3D	Three Dimensional
10FCV	Ten-fold Cross Validation
ADL's	Activities of Daily Living
AI	Artificial Intelligence
ANN	Artificial Neural Networks
AR	Augmented Reality
AUC	Area Under Curve
DoFs	Degrees of Freedom
DS	Development Stages
DTs	Decision Trees
EMG	Electromyography
EoMs	Equations of Motion
FS	Feature Selection
GBM	Gradient Boosting Model
GRF	Ground Reaction Forces
ICT	Information and Communications Technology
ID	Inverse Dynamics
IIR	Infinite Impulse Response
IK	Inverse Kinematics
IMU's	Inertial Sensors
JRA	Joint Reaction Analysis
JSM	Joint space narrowing on Medial compartment
JSN	Joint space narrowing
JSON	JavaScript Object Notation
KAM	Knee Adduction Moment
KL	Kellgren and Lawrence
KNN	k-Nearest Neighbours
KOA	Knee Osteoarthritis
KPI	Key Performance Indicators
Light GBM	Light Gradient Boosting Machine
LIME	Local Interpretable Model-Agnostic Explanations
LR	Logistic Regression
MLP-ARD	Multilayer Perceptron with Automatic Relevance Determination
MRI	Magnetic Resonance Imaging
ML	Machine Learning
MLP	Multilayer Perceptron
ms	Milliseconds
OA	Osteoarthritis
OAI	Osteoarthritis Initiative
OS	Operating System
R&D	Research and development
RF	Random forests
RFE	Recursive Feature Elimination
RMSE	Root Mean Square Error
SHAP	SHapley Additive exPlanations
SO	Static Optimization
SVM	Support Vector Machines
VE	Virtual Environment
VR	Virtual Reality
WP	Work Package
XGBoost	eXtreme Gradient Boosting

1. Summary

This report refers to Deliverable 9.1, which relates to the OActive WP 9, "Technology assessment and full system validation", specifically to Task 9.2, "Validation in big data registries".

This deliverable describes the evaluation rationale, roadmap, methods, and processes that are being used for the evaluation and the validation of the OACTIVE system.

2. Introduction

The objective of WP9 is to validate the integrated OACTIVE system by employing a comprehensive methodology that involves: (i) Clinical studies in human populations and (ii) validation of the system using big data registries. The ethical, legal, and social challenges that need to be met in order for the scientific advances to be responsibly applied will be finally investigated.

2.1. Purpose and Scope

For the validation for the tool that is being developed during the lifespan of OACTIVE project, 320 subjects are being recruited in 3 different countries (Spain-HULAFE, Greece-ANIMUS and Cyprus-Apollonion Private Hospital) involving patients that may develop OA, athletes and elderly people with developed OA. The aim of the clinical studies is to collect data, examine the relationship between the various risk factors generated by the different information sources and the clinical diagnosis (physical examination of clinicians).

The data collected from each patient is divided in different data subsets: (i) training data: data that is being used for building the personalised models, (ii) fine-tuning data: data used for further optimising the models and (iii) validation data: data utilised for testing the efficiency of the trained and fine-tuned patient-specific models. In addition, the proposed AR-based treatments will be evaluated towards the goal of personalized medicine in the cases of athletes and elderly people by modifying the gait pattern and/or proposing carefully selected exercises.

The integrated OACTIVE hyper-model is being validated using the big data registries from the OAI, which has collected substantial amounts of imaging, lifestyle, and biochemical data and other complementary data streams on the healthy subjects, and patients affected by OA. In OACTIVE, we will combine all of this information allowing for the first time the simultaneous exploration of multiple risk factors in big human populations involving thousands of patients. The knowledge (data) extracted here will then serve as input for the integrated computer hyper- models that will be constructed in work package 6. In task, the big data methodology, developed in WP6, will search through massive amounts of information, analysing it to predict outcomes for individual patients. That information will include data from past treatment outcomes with the outcome not only to predict but also to reveal surprising associations in data that our human brains would never suspect.

3. The OACTIVE system development

The development of the OACTIVE system is a many years procedure, in which many different techniques and resources were used in order to create the current form of the entire system. Specifically, the OACTIVE system has a duration of 36 months (3 years) and is divided into four phases which include the following activities: R&D activities (WP2-6), Validation activities (DEMO) (WP7-9), Dissemination and Exploitation Activities (DEC) (WP10), Project management activities (PM) (WP1).

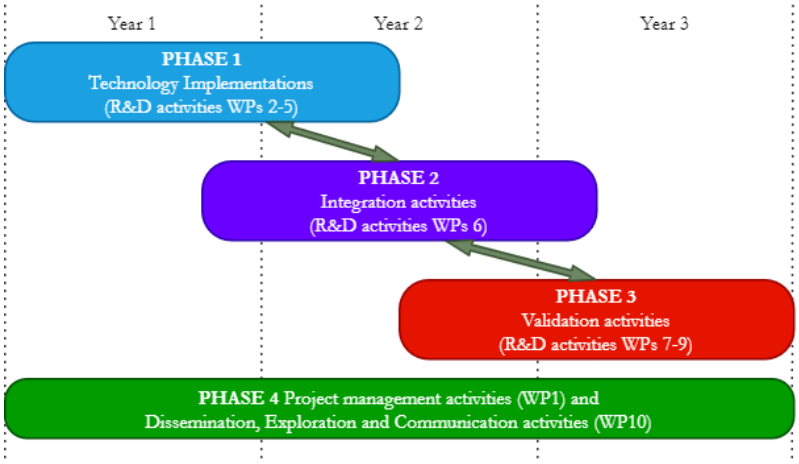

Figure 3.1 OACTIVE work plan

Phase 1: Technology generation and experimentation: This phase of the project includes all the R&D activities for the development of the OACTIVE personalised models implemented at various scales along with the design / development of the intervention module. The phase is finalised when all the developed models have been designed and tested in the laboratory and are ready for integration (WP2-5).

Phase 2: Integration of the developments from Phase 1 using big data: This phase involves the integration of all the developed technologies of Phase 1 (including mechanistic / phenomenological models, output information sets from various scales such as biological, social and behavioural). Big data and machine learning technologies will play a key role being the integrator of the various information sets as developed in Phase 1. Each model will be fine-tuned with the rest, and minor modifications are expected in order to optimise all the submodules to operate as a single integrated multi-scale hyper-model. In order to achieve this, the integration process runs in parallel with Phase1 giving constant feedback for modification for each sub-model.

Phase 3: Validation of the OACTIVE system. The aim of this phase is to validate the integrated OACTIVE system in both clinical studies/trials and big data registries. Clinical studies will also offer vital input to make any necessary adjustments before deploying the system in humans. Big data registries will be used to verify the efficiency of OACTIVE in a large human population. The accumulated results will give feedback to Phase 2 in order to monitor the required actions and perform an evaluation of the *Key Performance Indicators* (KPIs).

Phase 4: Project Management and Dissemination, Exploitation, and Communication Activities: This Phase runs the whole duration of the project in order to keep track of all the involved activities (R&D, DEMO) and take action when required. This will ensure the smooth

progress of all the R&D and demonstration activities as well as efficient planning for dissemination of the results throughout the duration of each phase.

These phases are divided into 10 work packages in total for a better management of the workload within the project, as it is briefly described below.

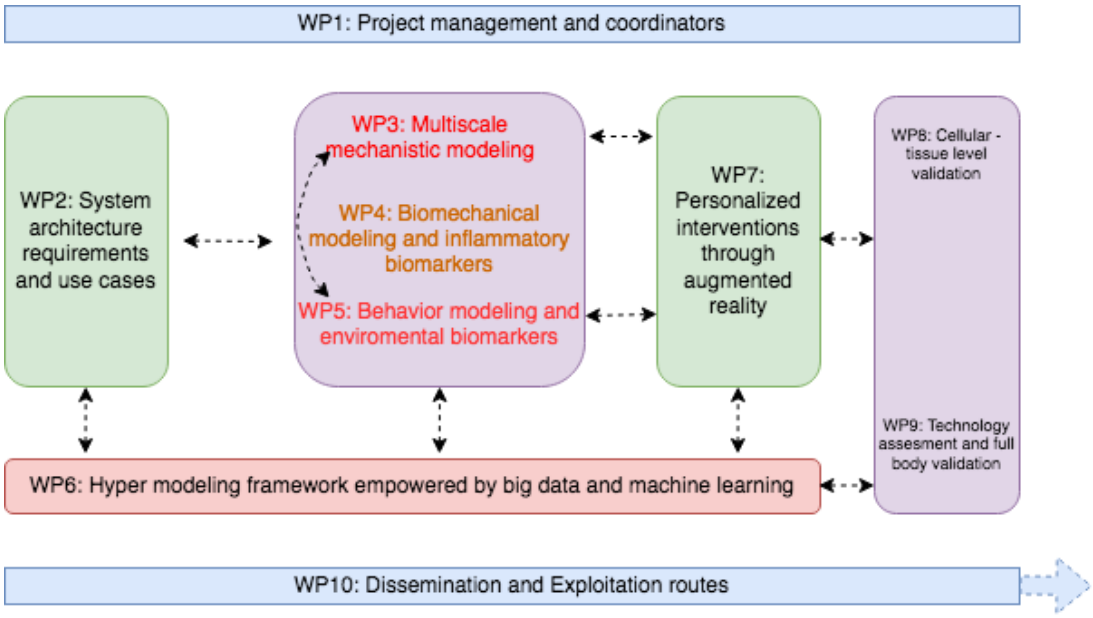

Figure 3.2 OACTIVE Work Packages diagram

WP1- Project Management: WP1 includes the administrative management, the quality management and the management of knowledge, IPR issues, Ethical, Legal and Safety Management as well as Open Research Data Management. This WP runs during the whole life of the project and interacts with all other WPs.

WP2- System Architecture Requirements and Use cases: WP2 is focused on the overall needs, architecture and system\ specifications of the OACTIVE infrastructure taking into account all medical, regulatory, and technological perspectives. It analysed and specified the requirements, restrictions and defined high-level needs delivering a number of representative use cases and clinical studies to highlight its novelties. This WP ran during M1-M6.

WP3- Multiscale mechanistic modelling: WP3 focuses on the creation of scalable subject-specific musculoskeletal biomechanical models to be used for simulations of able-bodied and pathological movement. This WP is running during M6-M41.

WP4- Biochemical modelling and inflammation biomarkers: The aim of this WP is to examine the relationship between biochemical markers for OA and clinical diagnosis. This WP is running during M6-M40.

WP5- Behaviour modelling and environmental biomarkers: WP5 aims to detect user's physical, mental and social behaviours and information that can be used for providing individualized diagnosis and recommendations for patient-specific treatments. This WP is running during M6-M41.

WP6- Hyper-modelling framework empowered by big data and machine learning: The objective of WP6 is to develop the hyper-modelling framework of OACTIVE which will

include data management mechanisms, development of data pre-processing algorithms, data mining techniques and the necessary ICT machine learning infrastructure, design and implementation of personalized predictive models, ontology-based framework and mechanisms for increased privacy and security. This WP is running during M13-M42.

WP7- Personalized intervention through augmented reality: The aim of WP7 is to develop AR tools for personalized interventions that will be used for the patient specific management of the condition. This WP is running during M6-M42.

WP8- Cellular-Tissue level validation: The objective of WP8 is to validate in vitro the relationship between cellular responses of osteochondral tissue and (a) biomarkers and imaging data (diagnostics), and (b) the tissue level mechanical activation during AR rehabilitation (therapy). This WP is running during M6-M42.

WP9- Technology assessment and validation: The objective of WP9 is to validate the integrated OACTIVE system by employing a comprehensive methodology that involves clinical studies in human populations and validation of the system using big data registries. This WP is running during M10-M42.

WP10- Dissemination and Exploitation: This WP involves all the dissemination and exploitation activities of the project e.g., the establishment and management of the project website, the production of the project's brochure, the organization of the OACTIVE workshop, continuous dissemination activities and project clustering activities as well as the development of the project's exploitation strategy.

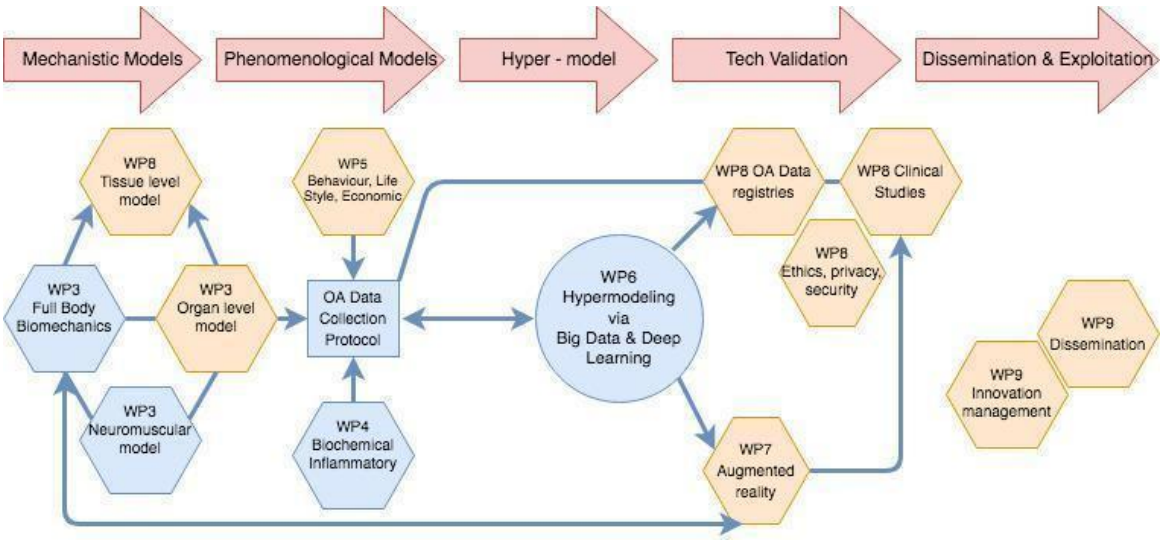

Figure 3.3 Schematic description of the OACTIVE system architecture and the relation with the WPs

4. Framework rationale

The aim of the OACTIVE project is to employ knowledge discovery techniques that are capable of extracting interpretable ruled-based knowledge from clinical time series, that will provide insights for the understanding of OA disease development and its progression. The extracted knowledge will be expressed in terms of IF-THEN rules utilizing only the significant information from the wealth of the acquired risk factors. The proposed approach makes it possible for the decision process of a trained model to be expressed as classification rules, which are more comprehensible to a human user than the classification process of the model which involves complex nonlinear mapping of the input data. Analysing, combing, and mining these rules to find hidden patterns in is a major challenge. For example, protein genetic polymorphisms or joint loading patterns which predict disease vulnerability or explain the disease process themselves. The insights will help design effective disease prevention measures or develop treatments.

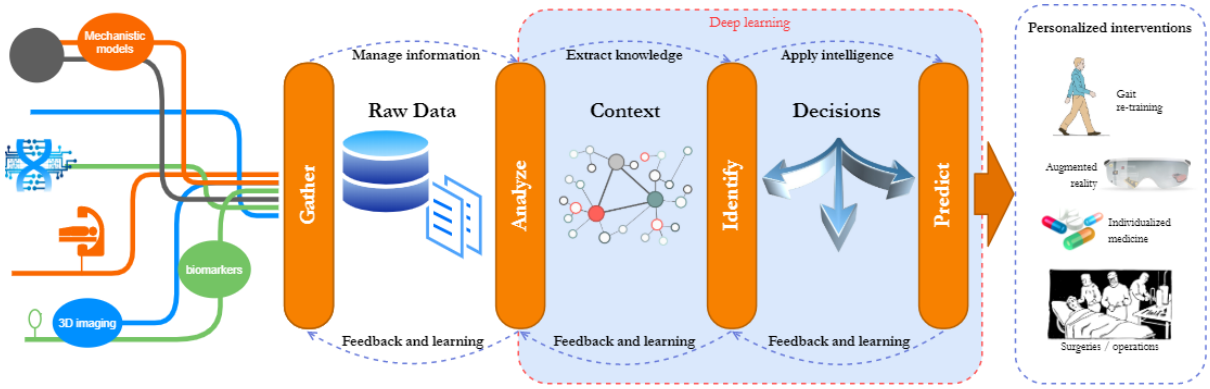

Figure 4.1: The OACTIVE system integration methodology

In Big Data research the most important tasks are to isolate high value features from the raw set of features (potentially irrelevant, redundant and noisy) and also to maintain the requirements in measurement and storage. In order to cope with the demanding problem of selecting significant risk factors, from a considerably extended database comprising records of high dimensionality, Computational efficient Feature Selection (FS) algorithms will be developed. The OACTIVE massive collections of data (thousands of subjects and risk factors) pose significant challenges for the development of the applied FS techniques. Addressing these demands, fast greedy FS algorithms will be developed that will search through candidate feature subsets guided by novel evaluation measures which captures the goodness of each subset, paying also attention to the complementary information between the selected factors. The goodness of each risk factor or subset will be independently and in combination with other complementary information sets. At different stages of the OA lifetime the FS will be applied to recognize risk factors subsets that are significantly correlated with the OA onset but also identify risk factors subsets associated with other development stages such as mild and moderate OA. Finally, significant post treatment patient-specific risk factors will be identified and will be used for the validation of the applied personalized interventions acting as a feedback mechanism to further improve/fine-tune the proposed interventions.

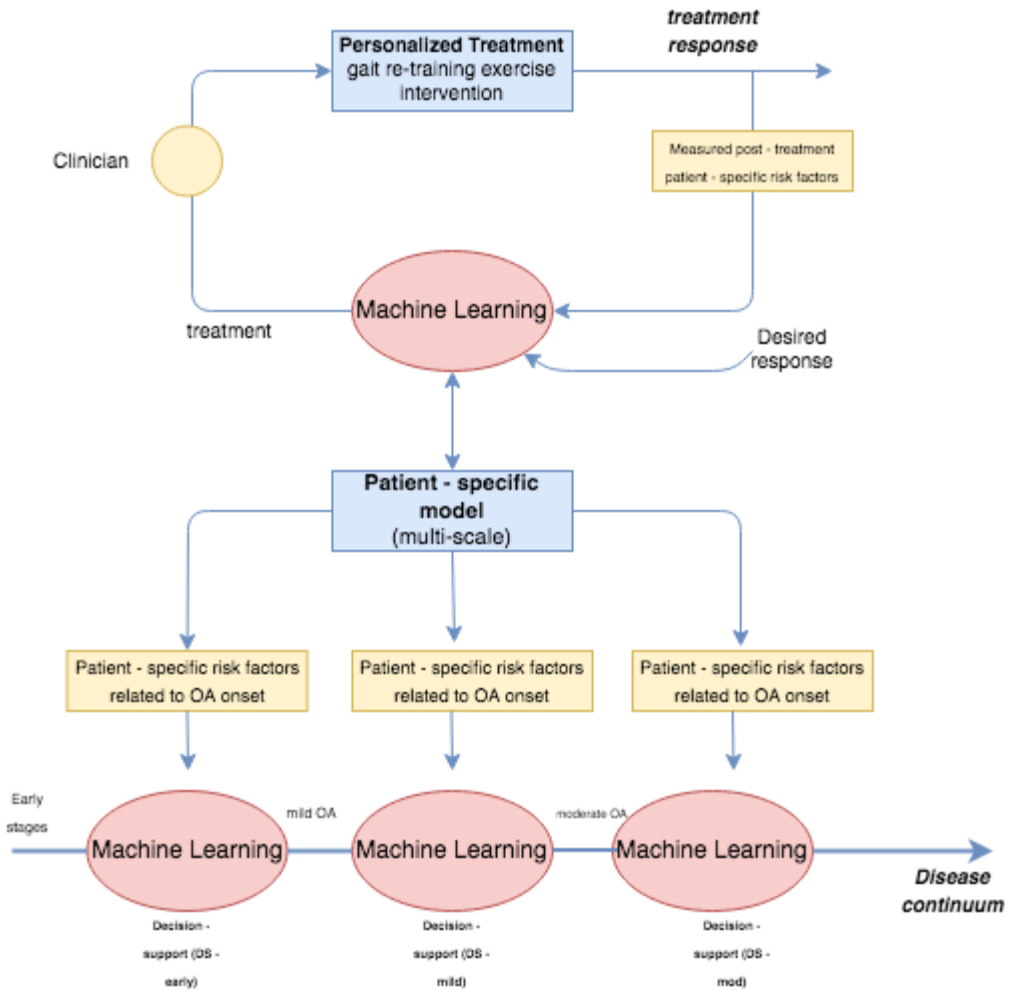

Figure 4.2: Architecture of the personalized predictive/ diagnostic modelling

OACTIVE prioritizes the development of a number of computational efficient 'local' predictive/diagnostic models that address specific OA stages in the disease continuum of a patient. Advanced pattern recognition models will be employed to model the OA disease onset and further progression (Figure 3.3). The training process of the models will be based on the significant risk factors recognized on the previous evaluation analysis. The outcome will be the generation of different local personalized decision models corresponding to different OA development stages (DS-early, DS-mild and DS-mod). Various classification models (such as artificial neural networks, support vector machines, decision trees and discriminant analysis) will be investigated for their appropriateness in providing accurate and robust decisions. The best model will be selected to accomplish the complex multi-class problem of OA diagnosis and severity assessment. Moreover, to analytically assess the information content of each risk factor family, the proposed model will be separately applied on every feature family. The final outcome (diagnosis) will be derived by applying fusion techniques on the individual decisions allowing biomedical researchers to investigate the influence of environmental factors on OA occurrence and their interactions with other health factors. Personalized decision support models will also be developed as part of a feedback-loop treatment mechanism where the models will be trained and updated using post-treatment patient-specific risk factors.

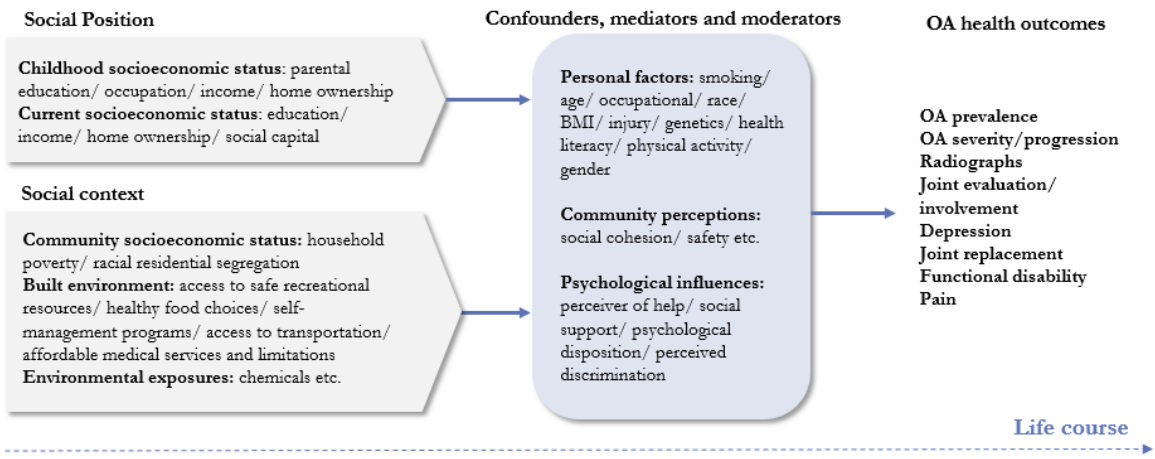

Figure 4.3: Socioeconomic risk factors considered in OACTIVE.

●

4.1. Evaluation Dataset Description

The OACTIVE project evaluation process consists of two different dataset collections. The first dataset collection is created from the OACTIVE project's clinical studies, while the second one is obtained from the Osteoarthritis Initiative database (OAI). The purpose of those collections is the training, evaluation and validation of the machine learning infrastructure of the OACTIVE project, with the ultimate goal of better and more consistent predictions.

For the created dataset collection, from the clinical studies of the OACTIVE project, the process involves data collection in 3 different countries (Spain-HULAFE, Greece-ANIMUS and Cyprus Apollonion Private Hospital) involving patients that may develop OA, athletes and elderly people with developed OA. The aim of the clinical studies is to collect data, examine the relationship between the various risk factors generated by the different information sources and the clinical diagnosis (physical examination of clinicians). These results are being used for the development of advanced computer modelling and simulation tools in order to be used in early diagnosis or prediction of further progression of OA. The procedures started in month 6 of the project and will last until the end of the project. In addition, the proposed AR-based treatments are being evaluated towards the goal of personalized medicine in the cases of athletes and elderly people by modifying the gait pattern and/or proposing carefully selected exercises.

- HULAFE (Spain): targeted population: subjects entering OA being at high risk of developing OA. Desired population size: >100; Acquired population size: 115.
- ANIMUS (Greece): targeted population: athletes in post-traumatic OA. Desired population size: >90; Acquired population size: 113.
- NIC (Cyprus): targeted population: elderly people with developing OA. Desired population size: >130; Acquired population size: 239.

This task lasted for more than 2 years so as to allow the collection over a long period of time covering potentially different periods of well-being and periods of illness within a patient-specific framework.

For the second dataset collection the data were obtained from the osteoarthritis initiative (OAI) database. Specifically, the current study only includes clinical data from: (i) The baseline; (ii) the

first follow up visit at month 12 and (iii) the next follow up visit at month 24 from all individuals being at high risk to develop KOA or without KOA. Eight feature categories were considered as possible risk factors for the prediction of KL as shown in Table 3.1. Furthermore, our study was based on Kellgren and Lawrence (KL) grade as the main indicator for assessing the clinical status of the participants.

Table 4.1: Main categories of the feature subsets considered.

Category	Description	Timeline of Visit		
		Baseline	12 Months	24 Moths
Subject characteristics	Anthropometric parameters including height, weight, BMI, abdominal circumference, etc.	●	●	●
Behavioural	Participants' social behaviour and quality level of daily routine	●	●	●
Medical history	Questionnaire data regarding a Participant's arthritis-related and general health histories and medications	●	-	-
Medical imaging outcome	Medical imaging outcomes (e.g., osteophytes and joint space narrowing)	●	-	-
Nutrition	Block Food Frequency questionnaire	●	-	-
Physical activity	Questionnaire results regarding leisure activities, etc.	●	●	●
Physical exam	Physical measurements of participants, including isometric strength, knee and hand exams, walking tests and other performance measures	●	●	●
Symptoms	Arthritis symptoms and general arthritis or health-related function and disability	●	●	●

The current study only includes clinical data from baseline from all individuals without or being at high risk to develop KOA in at least one knee. Data from the baseline (625 features in total) from nine feature categories were considered as possible risk factors for the prediction of joint space narrowing (JSN) as shown in Table 3.2.

Data from the OAI database was used in this work in order to validate our approach. This database was designed for 2 specific reasons: (i) to identify the factors that cause KOA, (ii) to promote the research in the area of KOA, which is going to create a better quality of life for patients with KOA.

The OAI database was launched in 2002, and its data is from patients in the ages 45-79 years old, either with symptomatic KOA, or being on the verge of developing it, in at least one knee. The study that produced this database had taken place in four medical centers in the US. In total 4796 patients were enrolled in the study, which lasted for 8 years. The most significant thing about this database is that it had more than 90% follow up for the first 4 years. In this paper though, we have not used all of the features. We have developed a voting system for assessing feature importance using only baseline data. WOMAC pain data from the first four visits was utilized to identify the different clusters of pain progression, whereas the selected feature subsets, as generated by the application of proposed FS methodology on baseline features, were used to train the ML models and finally produce the predictions.

Table 4.2: Main categories of the feature subsets considered in the proposed methodology.

Category	**Description**
Anthropometrics	Variables that describe measurements of participants including height, weight, BMI, abdominal circumference etc
Behavioural	Variables which describe the participants' social behaviour
Quality of life	Questionnaire results regarding the quality level of daily routine
Medical history	Questionnaire data regarding a participant's arthritis related and general health histories and medications
Medical imaging outcome	Variables which contain medical imaging outcomes (e.g., osteophytes and joint space narrowing)
Nutrition	Variables collected using the modified Block Food Frequency questionnaire
Physical activity	Questionnaire results regarding leisure activities etc
Physical exam	Variables which contain physical measurements of participants, including isometric strength, knee and hand exams, walking tests, and other performance measures

5. Methods and processes used for the evaluation and validation of the OActive system

The proposed model training method in OACTIVE is a Machine Learning method. Machine learning is divided into three main approaches: Supervised learning, Unsupervised Learning, and Reinforcement Learning. Supervised learning is the process of presenting an algorithm with example data and their desired outputs, which subsequently creates a model that can predict the responses of every data similar to the example. On the other hand, Unsupervised learning is the process where no labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself, by discovering hidden patterns in data, or a means towards an end. Finally, Reinforcement learning is the process by which a computer program interacts with a dynamic environment in which it must perform a certain goal (such as driving a vehicle or playing a game against an opponent). As it navigates its problem space, the program is provided feedback that's analogous to rewards, which it tries to maximize.

The validation activities of OACTIVE were focused on the Osteoarthritis Initiative database (http://www.oai.ucsf.edu/) and could be summarized as follows:

· Validation of interpretable models (*Logistic Regression, MLP*)

1. Risk of OA disease onset, identified at first clinical presentation in the data set from the OAI
2. Risk of further progression, by stratifying the baseline cohort with KL 0, 1 for the risk of progression to KL 2, 3, 4 within 7 years from baseline.

· Validation of ML models (*K-Nearest Neighbor, Support vector machines, Decision trees, Random Forest, XGboost, Naive Bayes*)

3. KOA diagnosis (healthy versus OA)
4. Prediction of KL progression (non-progressors versus progressors)
5. Prediction of pain (non-progressors versus progressors)
6. Prediction of joint space narrowing (non-progressors versus progressors)

· The proposed novel segmentation technique was also validated on MRI images acquired from the OAI initiative.

A variety of performance metrics was utilized including classification / regression accuracies, sensitivity, specificity, positive predictive value, AUC and confusion matrixes.

Given that ML can be used in many sensitive environments making life-changing decisions, OACTIVE took fairness issues into account to ensure that discriminatory behaviors toward certain groups or populations are omitted. In light of this , we proceeded in the first-ever validation of machine learning models with respect to fairness in the KOA classification research using two fairness criteria (demographic parity and balanced equalized odds). Finally, we performed explainability analysis (using LIME and SHAP libraries) to identify why specific decisions are made by the employed ML models highlighting the features that contributed more to the decisions made.

Task 9.2 is still ongoing and will be finished by the end of the project (at month 42). At the last time period of the project, the predictive capacity of the aforementioned AI models will be tested on the OACTIVE data. Specifically, only the features as have been selected by the knowledge discovery, will be utilized per data source. The trained AI models will be finetuned using a part of the OACTIVE dataset and the rest will be used to test the capacity of the models (using a variety of metrics, including accuracy, sensitivity, specificity and confusion matrix). Specifically, cross validation will be used to quantify the predictive capacity of the OACTIVE models. Two different validation approaches will be explored:

(i) the OACTIVE data will be used to test the performance of AI models that are trained on data from the OAI database, and

(ii) Selected data from the OAI database will be used to validate AI-empowered predictive models that are created using the OACTIVE data.

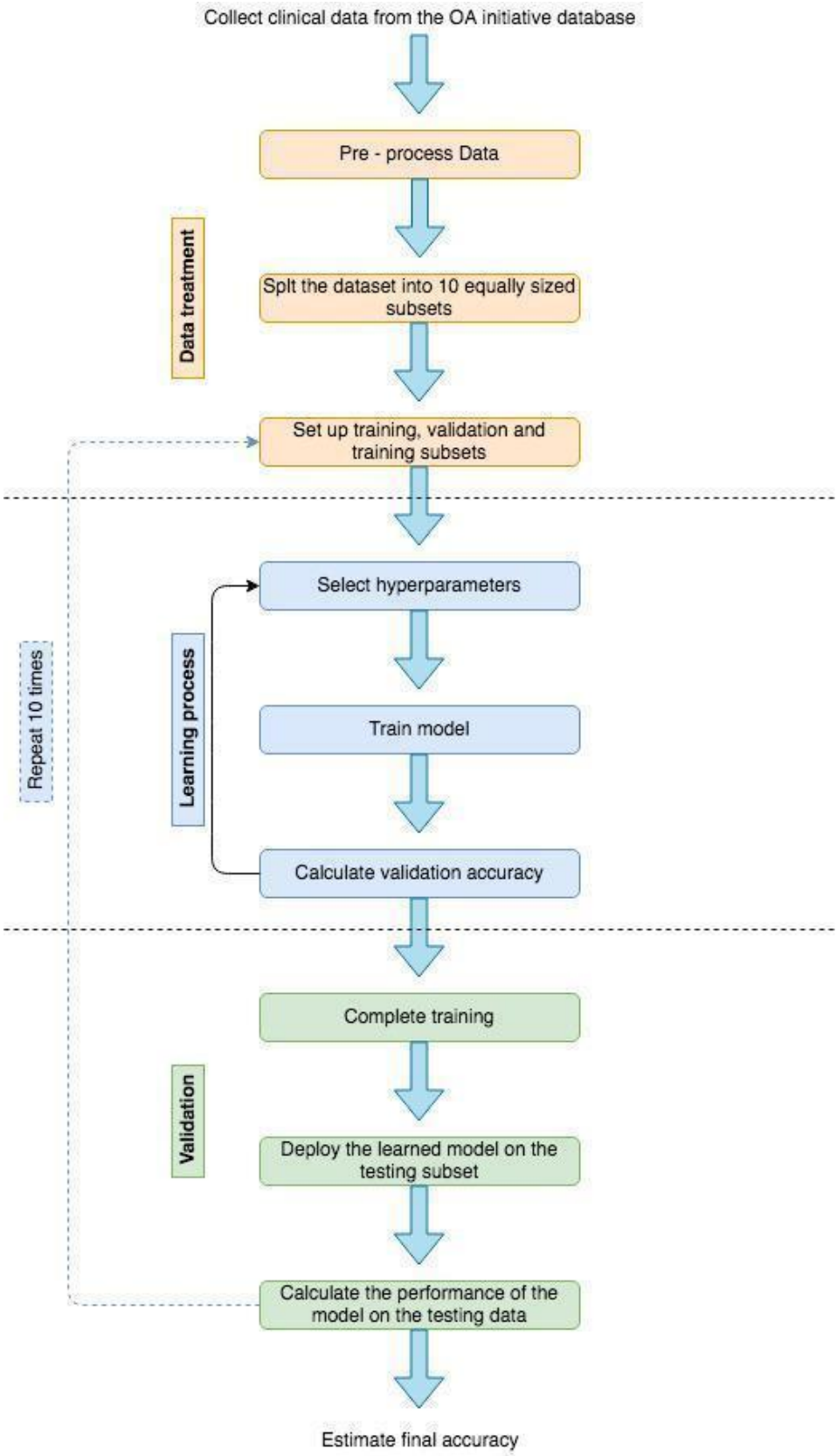

Figure 5.1: Flowchart of the proposed machine intelligence methodology

5.1. Data pre-process

To discover knowledge from medical data a common approach is to target the correct data, transform them via a pre-processing stage and recognize patterns in order to extract knowledge. Data mining techniques that are commonly used in the medical data and healthcare industry are association, clustering and classification.

A similar approach was developed in our study by taking advantage of the combination of descriptive and predictive techniques, such as clustering, feature selection and classification. The proposed machine learning methodology for predicting JSN consists of 4 main steps: (i) Data pre-processing; (ii) Data clustering; (iii) Feature Selection; and (iv) Data Classification. In the first step, data cleaning and normalization are performed to remove noise and bring all the variables to the same range. The normalized data are then clustered based on the joint space narrowing on Medial compartment (JSM) measures for the left and the right leg, respectively. Thus, the selection and extraction of features are realized based on the identified clusters (that are considered as classes in our case). Consecutively, the selected features are used to develop prediction models for KOA progression of patients (Figure 1).

The proposed, in this paper methodology, comprises of the following components: (i) a fitting technique for grouping/labelling of the data, (ii) a hybrid and robust Feature Selection technique employing a number of feature ranking algorithms to avoid bias, (iii) Machine Learning models for decision making and (iv) Validation.

5.2. Normalization

For the pre-processing of the data, we followed two procedures. The dataset has a lot of missing values, so we used mode imputation to handle them. We used this specific method because it is able to handle both numerical and non-numerical variables. We also standardized the features by subtracting the mean and scaling the values with respect to variance. This is a common requirement, because some ML algorithms behave badly, if the features are not normally distributed. Data cleaning was performed by excluding the columns with more than 20% missing values compared to the total numbers of samples. For the rest data, data imputation was implemented to replace missing values of the categorical or numerical variables by the mode (most frequent value) of the non-missing variables. Furthermore, standardization of a dataset is a common requirement for many ML estimators. Data was normalized to [0, 1] to build a common basis for the feature selection algorithms that follow.

5.3. Feature Selection

A hybrid feature selection methodology was employed consisting of filter, wrapper and embedded techniques, whereas feature ranking was decided on the basis of a majority voting system. Applying each technique separately, the order of the feature importance emerged from the frequency of feature appearance in the selection criteria. The features were ranked with respect to the votes received. Feature selection methodology combined the outcomes of six FS techniques: two filter algorithms (Pearson correlation and Chi-2), wrapper (with logistic regression) and three embedded ones (logistic regression L2, random forest and Light GBM). Feature ranking decided on the basis of a majority vote scheme as we performed all six FS techniques separately, each one resulting in a selected FS. A feature receives a vote every time it has been selected by one of the FS algorithms. We finally ranked all features with respect to the votes received.

Pearson Correlation is the most important correlation factor being based on the concept of linear relationship. If there is a linear dependence between two features, then their correlation coefficient is ± 1. If there is no dependence, the correlation coefficient is 0. However, if two variables are highly correlated among themselves, they provide redundant information regarding the target. Consequently, the second variable doesn't add additional information, so removing it can help to reduce the dimensionality. In this approach, we set the maximum number of the selected features to be 30.

Chi-squared independence test was applied to examine the relationship between two quality variables. The Chi-squared statistical test also works manually with nonnegative numerical and quantitative characteristics. The specific test compares the degree of agreement (or correlation) between the theoretical frequency and the actual frequency. The algorithm was decided to terminate at 30 selected features. The termination criterion was manually selected after a trial-and-error exploration process.

Recursive Feature Elimination (RFE) is a greedy optimization algorithm which aims to find the best performing feature subset. In each iteration, it creates models and keeps aside the best or the worst performing feature. Each next model includes a reduced number of features until all the features are exhausted. At the end, it ranks the features based on the order of their elimination. In this approach, the logistic regression classifier was selected to drive the elimination process whereas the termination criterion was also set to 30 features.

Logistic regression (L2 penalty) is an embedded method relying on regularized logistic regression models. Furthermore, this approach is based on small subsets of the full feature space by sampling at random this space. The sampling probability depends on the estimated feature relevance. In addition, the initial relevance of each feature is estimated according to a t-test ranking.

Random forests (RF) are a popular method for feature ranking, due to the fact that they require very little feature engineering and parameter tuning. But they come with their own limitations, especially when data interpretation is concerned. In high correlated data, strong features can end up with low scores and the method can be biased towards variables with many categories. In addition, this method rearranges stochastically all values of the features for each tree and uses the RF model to predict this permuted feature.

Light GBM is a gradient boosting framework that uses a tree-based learning algorithm. Specifically, Gradient based One-Side Sampling and Exclusive Feature Bundling are used to deal with large numbers of data instances and large numbers of features. Light GBM can handle the large size of data, it takes lower memory to run and is faster than Gradient Boosting Decision Tree.

The steps of the proposed feature selection are shown in steps below:

```
Step 1: All the features were normalized as we described in the Pre-processing
Section.
Step 2: Each one of the six FS techniques performed separately and as a result
create six feature subsets ...
Step 3: Main Loop that consists of 6 sub-steps.
```

```
        Step 3.1: For each feature j, we set v_j = 0, j = 1, …, M where M the total
        number of features.
        Step 3.2: Set j = 1
        Step 3.3: if feature j is selected in FS_i, then v_j = v_j + 1
        Step 3.4: Repeat step 3.3 for each one of the six FS techniques for i =
        1,…,6
        Step 3.5: Set j = j + 1
        Step 3.6: Terminate main loop if j > m otherwise go to step 3.3.
Step 4: Rank features to descending order with respect to v_j (that is the final
selection criterion)
End
```

5.4. Classification models

There is an intrinsic dichotomy in classification problems in the health domain in general. It could be argued that the only goal of a study should be to develop a system that is able to correctly attribute cases to classes, and in this case, we assume a "black-box" model of the system being developed (Multilayer Perceptron or Support Vector Machines, for example). Similar kinds of algorithms take some inputs and return some outputs; they can reach a quite high level of accuracy, but they will not enrich the human knowledge of the disease process under investigation. This is a key point in the biomedical context: clinicians often want to understand the way the classifier is behaving to judge its performance. This is a quite interesting perspective: underlying their interest there is the desire of gaining a deeper knowledge of the biological disease processes by interpreting the results returned by the system. This is a peculiar aspect of the biomedical field in which a percentage point in the classifier accuracy can decide the health and treatment implications of a patient. Another model is then needed to address these requirements. A second set of approaches provides a deeper insight into the problem adding a clear description of how the system arrived at the prediction. Such clear descriptions can be represented by IF-THEN classification rules and the process of rule extraction from a dataset is called rule induction. Several algorithms have been proposed for accomplishing the rule induction task, with C4.5, probably, being the most famous.

Classification consists of two steps: training and testing. During training a classification model is built based on the collected training data for generating classification rules. The accuracy of the model is estimated with the test data.

Various ML models were evaluated for their suitability in the task of KOA prediction. A brief description of these models is given below. We tested logistic regression which is likely the most commonly used algorithm for solving classification problems. Logistic regression models the probabilities for classification problems with two possible outcomes. It's an extension of the linear regression model for classification problems. The interpretation of the weights in logistic regression differs from the interpretation of the weights in linear regression, since the outcome in logistic regression is a probability between 0 and 1. We also evaluated decision trees (DTs) which are a non-parametric supervised learning method used for classification and regression. They are simple to understand and to interpret. DTs require little data preparation and perform well even if their assumptions are somewhat violated by the true model from which the data were generated. k-Nearest Neighbours (KNN) as well as non-linear support vector machines (SVM) algorithms, which can deal with the overfitting problems that appear in high-dimensional spaces.

In the classification setting, the KNN algorithm essentially boils down to forming a majority vote between the K most similar instances to a given "unseen" observation. Similarity is defined according to a distance metric between two data points. A popular one is the Euclidean distance method. Furthermore, SVMs are a set of supervised learning methods used for classification, regression and outlier's detection. They are effective in high dimensional spaces and still effective in cases where the number of dimensions is greater than the number of samples. The ensemble technique Random Forest was also evaluated using DT models as weak learners. RF classifier creates a set of decision trees from randomly selected subsets of the training set. It then aggregates the votes from different decision trees to decide the final class of the test object. XGBoost and naive Bayes algorithms were also considered. XGBoost model is a sum of CART (tree) learners which try to minimize the log loss objective and the scores at leaves. These scores are actually the weights that have a meaning as a sum across all the trees of the model. Furthermore, they are always adjusted in order to minimize the loss. Moreover, naive Bayes methods are a set of supervised learning algorithms based on applying Bayes' theorem with the "naive" assumption of conditional independence between every pair of features given the value of the class variable. Naive Bayes learners and classifiers can be extremely fast. The decoupling of the class conditional feature distributions means that each distribution can be independently estimated as a one-dimensional distribution.

Machine Learning models were explored for their suitability in predicting pain progression on feature subsets of varying dimensionality, in order to see which one produces the best results. In this subsection we give a brief overview of the models that were employed in order to tackle the pain prediction problem.

k-Nearest Neighbours: it is a non-parametric, lazy learning algorithm. The classification prediction of a sample datapoint, is achieved with the use of data, which are class-separated. The algorithm presumes that similar data points are close to each other. More specifically, this algorithm loops over every datapoint in the data and calculates the distance between every datapoint and the chosen datapoint. The distances are sorted in an ascending order and then the algorithm chooses the first k entries.

Naive Bayes: Naive Bayes is a probabilistic classifier that uses the Maximum A Posteriori decision rule in a Bayesian setting and is included in supervised learning. The main idea behind this method is the Bayes Theorem. Bayes theorem approximates the probability of an event given the probability of a past event. The Naive Bayes predicts membership of probabilities for every class, such as the probability that the given data point belongs to a particular class. The data point belongs to the class with the highest probability score.

Support Vector Machines: it is an algorithm which finds a line that separates the data points that belong to different classes. The data points that are closest to the line play a crucial role in the learning process e (the so-called support vectors). Then the distance between the line and every datapoint is calculated, with an overall target to maximize the distance between classes. In case a non-linear separation is needed, kernels are applied in order to project the data points into higher dimensional spaces.

Logistic Regression (LR): it is a mathematical model that describes the relationship of data to a

dichotomous dependent variable. The model is based on the logistic function,

$$f(x) = \frac{1}{1 + e - x}, where\ x \in (-\infty, +\infty)\ and\ 0 \leq f(x) \leq 1$$

Thus, regardless of the value of x the model is designed to describe the data with a probability in the range of 0 and 1 in an A-shaped graph.

Decision Trees: It is one of the most famous algorithms for supervised learning for classification problems. It uses a lot of if-then-else decision rule statements in order to come to a decision. Its structure is a branch structure which breaks the data into data subsets, and then it produces decision and leaf nodes. Every node has a minimum of two branches, and every leaf node is for classification or a decision prediction.

Random Forest: It is an algorithm consisting of many decision trees algorithms. Its characteristics are the randomness in the sampling of data points when building the trees; and the randomness in the feature's subsets, when splitting nodes. Every tree in the algorithm learns from a random sample of data. These samples of data are being used several times by the trees, which means that the trees take them with replacement. So, every tree has high variance because of this fact, but the random forest has lower variance overall. It is worth noting that the decisions are the average of the predictions of all the trees in the random forest.

Gradient Boosting model GBM: It belongs in the family of the decision trees. GBM identifies and uses weak learners to produce strong learners through an additive, gradual and sequential process. A modified version of the initial training data set is fitted to develop a new tree. Let (x_i, y_i) for $i = 1, \dots, n$ be the training set, $L(y, F(x))$ be the loss function and M be the number of iterations.

XGBoost: XGBoost or eXtreme Gradient Boosting, is a parallel tree boosting that solves data science problems in a fast and accurate way. After constructing the boosted trees, the algorithm calculates the importance score of every feature of the dataset. This score is an indicator of how useful is its feature to the construction of the trees inside the algorithm. The calculation of this score is achieved by the amount that each feature point split improves the performance for the model for the data that the node is responsible for. A popular measure of performance is the Gini index which selects the split points. More specifically the Gini coefficient is a statistic which quantifies the amount of inequality that exists in a population. It is a number between 0 and 1, with 0 representing perfect equality and 1 perfect inequality. XGBoost in fact ranks the features of the data by comparing them to each other.

Multilayer Perceptron (MLP): It belongs in the category of Artificial Neural Networks (ANN) and it is the most common neural network. MLP is based on a supervised training procedure to generate a nonlinear model for prediction. MLP consists of layers, such as the input layer, output layer and hidden layers. Thus, MLP is a layered feedforward neural network where the information is transferred unidirectionally from the input layer to output layer through the hidden layers. In Figure 4a, a simple neuron perceptron is presented with a single layer where all inputs connect with only one output. Let $x_{i_{i=0}}^{n}$ be the input, such as features or variables, and $w_{i_{i=0}}^{n}$ be the weights of the neuron. The weighting step consists of three steps: (i) the multiplication of features with

the corresponding weight, $x_i w_{i_{i=0}}^{n}$; (ii) their sum, $z = \sum_{i=0}^{n} \quad x_i w_i$, and (iii) the transfer step where the output y is produced by the application of an activation function f to the sum, $y = f(z)$. Commonly used activation or transfer functions are the unit step (Heaviside), linear or logistic (sigmoid).

5.5. Clustering models

The proposed methodology was applied on the case study for the prediction of JSN in patients with KOA and it was implemented using the OAI dataset. The pre-processed data for the right and left leg was clustered and the groups of patients with/without JSM progression within the dataset were identified. To accomplish this task, K-Means, K-Medoids, Hierarchical clustering and Gaussian mixture models were employed in a comparative analysis. Davies Bouldin index was used to evaluate the optimal number of clusters in order to discriminate patients into groups but also to identify the magnitude of the variation in the patients' JSM measures. The parameters of the proposed clustering methods are presented below.

- *K-Means* → City block distance, 5 replicates
- *K-Medoids* → City block distance, 5 replicates
- *Hierarchical* → Agglomerative cluster tree, Chebychev distance, farthest distance between clusters, 3 maximum number of clusters
- *Gaussian mixture models* → using the Expectation-Maximization algorithm

Regarding the clustering process, 4 clusters were identified in most of the cases representing patients with zero, low, medium and high alterations in JSM measures, respectively. A class size imbalance problem was observed with Cluster 1 being significantly bigger than the rest three clusters for both left and right legs. We should notice that for the right leg some patients with low JSM alterations were erroneously grouped in the cluster with the stable or non-infected patients. To overcome these problems, we decided to perform clustering with only 2 clusters. Among the four potential clustering approaches, we adopted the results from the K-Means method since a better discrimination among the patient groups was achieved.

5.6. Evaluation and validation

The vision of OACTIVE project is the development of a holistic framework that envisages to consider individual/patient-specific information in a multi-scale approach. Computer modelling will be combined with simulations aggregating various information sets and inputs from models such as full body, organ and tissue level mechanistic models along with behaviour, lifestyle, environmental and other biochemical biomarkers of systemic health. The ultimate objective is to make a significant leap forward in developing patient-specific predictive and preventive computer-based models of the physiological systems at various scales, combined with data on population statistical variability and simulation tools, for understanding the development and progression of OA. Applying these models at the individual patient level and thus being able to predict outcomes more accurately at different stages of the disease, would help clinicians make informed decisions regarding the potential necessity of appropriate treatment at each stage and through time, and will

lead to the development of individually tailored preventative measures or treatments to maximize the efficacy of the intervention.

Ten-fold cross validation (10FCV) was used to evaluate the effectiveness of the learned classification models. The dataset was split into 10 subsets, called folds. The train-test method was applied iteratively by using each of the 10 folds for testing, while the learning model was trained with the remaining nine. The performance was calculated by averaging the individual ten test scores. To achieve a fair comparison between the different approaches, hyperparameter selection was performed for each one of the investigated machine learning algorithms. A validation subset was held out from the training sets (a randomly selected 10%) as a criterion for selecting the optimum hyperparameters by means of a grid search process.

To predict the JSN in KOA patients, six different prediction models were employed and compared separately at each leg, including Gradient Boosting, Logistic Regression, MLP, Naïve Bayes Gaussian, Random Forest and SVM. We validated the results by performing a 70%-30% train-test split. Learning of the algorithms was achieved on the stratified version of the train and the final performance was calculated on the test data. The evaluation of the models was performed on the presented medical dataset. For most of the models, hyper parameter tuning was realized with grid search and 3-fold cross validation. The test size was set to 30% with normalization upon the features. The models were evaluated in feature subsets of increasing dimensionality from 5 to 155 features with a step size of 5, from 155 to 325 features with a step size of 10, from 325 to 415 features with a step size of 15, from 415 to 475 features with a step size of 20 and from 475 to 625 features with a step size of 25.

Finally, the evaluation and in existence the optimization of the performance results is measured by Accuracy, Confusion Matrix, Sensitivity, Specificity and Area Under Curve (AUC).

Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition:

$$Accuracy = \frac{No\ of\ correct\ predictions}{Total\ number\ of\ predictions}$$

Confusion Matrix as the name suggests gives us a matrix as output and describes the complete performance of the model. If we have a binary classification problem. We have some samples belonging to two classes: YES or NO. Also, we have our own classifier which predicts a class for a given input sample.

Table 5.1: Example Confusion Matrix

	Predicted: No	**Predicted: Yes**
Actual: No	TrueNegative	FalsePositive
Actual: Yes	FalseNegative	TruePositive

There are 4 important terms:

- TruePositives : The cases in which we predicted YES and the actual output was also YES.
- TrueNegatives : The cases in which we predicted NO and the actual output was NO.
- FalsePositives : The cases in which we predicted YES and the actual output was NO.
- FalseNegatives : The cases in which we predicted NO and the actual output was YES.

Accuracy for the matrix can be calculated by taking average of the values lying across the "main diagonal":

$$Accuracy = \frac{TruePositive + TrueNegative}{TotalSamples}$$

Sensitivity (True Positive Rate): Sensitivity is a measure of the proportion of actual positive cases that got predicted as positive (or true positive). Sensitivity is also termed as Recall. This implies that there will be another proportion of actual positive cases, which would get predicted incorrectly as negative (and, thus, could also be termed as the false negative). This can also be represented in the form of a false negative rate. The sum of sensitivity and false negative rate would be 1.

$$Sensitivity = \frac{TruePositive}{FalseNegative + TruePositive}$$

Specificity (True Negative Rate): Specificity is defined as the proportion of actual negatives, which got predicted as the negative (or true negative). This implies that there will be another proportion of actual negative, which got predicted as positive and could be termed as false positives. This proportion could also be called a false positive rate. The sum of specificity and false positive rate would always be 1.

$$Specificity = \frac{TrueNegative}{TrueNegative + FalsePositive}$$

Area Under Curve (AUC) is a metric that is used widely for evaluation as it is used for binary classification problems. Classifier's AUC is equal to the probability that the classifier will rank a positive example, chosen randomly, higher than a negative one, chosen randomly. AUC has a range of $[0, 1]$. The greater the value, the better is the performance of our model.

The optimization process consists of the hyperparameter tuning of each algorithm. The hyperparameters settings for tuning are presented below:

- kNN:
    - Number of neighbours to use by default for k neighbours queries.
- *Naïve Bayes Gaussian: -*
- *SVM:*
    - The regularization parameter was tested on 0.001, 0.01, 0.1, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.

  - Kernel type was set to linear, polynomial, sigmoid and radial basis functions.
- *Logistic Regression:*
  - The inverse of regularization strength was tested on 0.001, 0.01, 0.1, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.
  - Algorithm to use in the optimization problem was set to 4 different solvers that handle L2 or no penalty, such as 'newton-cg', 'lbfgs', 'sag' and 'saga'.
  - A binary problem is fit for each label or the loss minimized is the multinomial loss fit across the entire probability distribution, even when the data is binary.
  - With and without reusing the solution of the previous call to fit as initialization.
- Decision Trees
  - The number of features to consider when looking for the best split.
  - The minimum number of samples required to split an internal node.
  - The minimum number of samples required to be at a leaf node.
- Random Forest:
  - The number of trees in the forest from 10 to 500 with 10 step size.
  - The maximum depth of the tree from 1 to 10 with 1 step size.
  - The minimum number of samples required to split an internal node: 2, 5 and 10.
  - The minimum number of samples required to be at a leaf node: 1, 2 and 4.
  - The number of features to consider when looking for the best split: $\sqrt{nfeatures}$ or $log2\ (nfeatures)$.
  - With and without bootstrap.
- *Gradient Boosting*:
  - The number of boosting stages to perform from 10 to 500 with 10 step size.
  - The maximum depth of the individual regression estimators from 1 to 10 with 1 step size.
  - The minimum number of samples required to split an internal node: 2, 5 and 10.
  - The minimum number of samples required to be at a leaf node: 1, 2 and 4.
  - The number of features to consider when looking for the best split: $\sqrt{nfeatures}$ or $log2\ (nfeatures)$
- XGBoost
  - Gamma: Minimum loss reduction required to make a further partition on a leaf node of the tree.
  - Maximum depth of a tree. Increasing this value will make the model more complex and more likely to overfit.
- MLP:
  - Hidden layers: The MLP consists of three or more layers (an input and an output layer with one or more hidden layers) of nonlinearly-activating nodes. Since MLPs are fully connected, each node in one layer connects with a certain weight to every node in the following layer.
  - Activator function: If a multilayer perceptron has a linear activation function in all neurons, that is, a linear function that maps the weighted inputs to the output of each neuron, then linear algebra shows that any number of layers can be reduced to a two-layer input-output model. In MLPs some neurons use a nonlinear activation function that was developed to model the frequency of action potentials,

or firing, of biological neurons.
  - Solver for weight optimization: Solver class represents a stochastic gradient descent-based optimizer for optimizing the parameters in the computation graph.
  - L2 penalty (regularization term) parameter: L2 regularization tries to reduce the possibility of overfitting by keeping the values of the weights and biases small.
  - The learning rate: Specifically, the learning rate is a configurable hyperparameter used in the training of neural networks that has a small positive value, often in the range between 0.0 and 1.0. The learning rate controls how quickly the model is adapted to the problem.

Hyperparameter selection was implemented to optimize the performance of our models and to avoid overfitting and bias errors. Each model was optimized with respect to a number of preselected hyperparameters that are presented below:

- for *XGBoost*
  - 'gamma': [0, 0.4, 0.5, 0.6]
  - 'maximal depth': [1, 2, 3, 4, 5, 6, 7, 8]
  - 'minimum child and weight': [1, 3, 4, 5, 6, 8]
- *for MLP:*
  - Hidden layers: (10,50,100), (50,100,150) and (100,200,400)
  - Activator function: Relu and $tanh$
  - Solver for weight optimization: Stochastic gradient descent, stochastic gradient-based optimizer proposed by Kingma, Diederik, and Jimmy Ba and an optimizer in the family of quasi-Newton methods.
  - L2 penalty (regularization term) parameter: 0.0001 and 0.05
  - Learning Rate: The learning rate schedule for weight updates was set as a constant learning rate given by the given number and as adaptive by keeping the learning rate constant to the given number as long as training loss keeps decreasing.
- for *KNN*
  - 'k-parameter': [5, 7, 9, 12, 14, 15, 16, 17]
- for *SVMs*
  - 'C': [0.001, 0.01, 0.1, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
  - 'kernel': ['linear', 'sigmoid', 'rbf', 'poly']
- for *Logistic Regression*
  - 'penalty': ['l1', 'l2']
  - 'C': [100, 10, 1.0, 0.1, 0.01]
- for *Decision Trees*
  - 'maximal features': ['auto', 'sqrt', 'log2']
  - 'minimum samples leafs': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]
  - 'minimum number of decision splits': [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]
- for *Random Forest*
  - 'criterion': ['gini', 'entropy']
  - 'minimum samples leaf': [1, 2, 3]
  - 'minimum samples split': [3, 4, 5, 6, 7]

- ‘number of estimators’: [10, 15, 20, 25, 30]

6. AR treatments

Personalized OA management can be thought of as the broad application of approaches that allow decision making to be based on the individual's specific test results and clinical factors rather than being based on global recommendations. Risks and benefits should be considered for each patient and therapy should be individualized wherever possible. Although established OA is currently an irreversible, chronic, disabling and painful condition that is not curable with existing treatments, research on the epidemiology of OA over recent decades has identified several risk factors that include biological, medical, environmental and social characteristics and influences. Most importantly, however, they include both non-modifiable and modifiable risk factors and this opens up the possibility of disease prevention before it is developed into an established incurable condition or slowing its progression and debilitating consequences once developed through appropriate interventions to modify the relevant risk factors. This is currently the most promising approach to tackle OA, but personalized disease prevention or treatment through modification of appropriate risk factors requires accurate and comprehensive risk prediction models. These must include all the main known risk factors for the development and progression of knee OA, but most importantly, the models must also capture the complex interactions between environmental, social and biological/medical factors. With such complete and accurate risk prediction models, it is possible to estimate the risk reduction by targeting one or several modifiable risk factors through patient-specific and effective treatments or interventions that can prevent the disease or slow down its progression and therefore have a major public health impact. Although such conventional risk prediction models and prevention strategies have been developed for other major diseases such as cardiovascular disease and cancer, there are only a few and quite limited models in OA such as the Nottingham knee OA risk prediction models, the OA Policy model (OAPol) or a limited factor joint replacement prediction model. Although there are some other OA prediction models reported in previous studies, the main criticism is that they are actually classification models of the disease rather than conventional risk prediction models that can be used for prevention. The most comprehensive and extensive risk prediction models by far are the Nottingham knee OA risk prediction models. These authors developed risk prediction models for incidence of radiographic and symptomatic knee OA and progression using conventional risk factors such as age, gender, BMI, family history of OA, occupational and sports participation risks, knee joint injury, and number of knee joint compartments affected by OA. Although the prediction outcomes are normally robust in various sensitivity analyses for these models, the range of risk factors included is very limited and future research should be performed to extend the efficiency of the modelling method in utilizing more information from the range of risk factors relevant to OA.

The OACTIVE project aims to revolutionize current practices for managing OA. These practices include nonpharmacological treatments such as providing patient education and self-management strategies, advising weight loss, strengthening programs, and addressing biomechanical issues. Oral analgesics and anti-inflammatories are pharmacological approaches that are commonly used and the literature overall supports that some of these medications can be helpful for managing OA in the short-term but are less effective for long-term management. Additionally, more prolonged use significantly increases the risk of serious associated side effects that are not too uncommon. Disease modifying OA drugs are being researched as a treatment modality to potentially halt or slow disease progression. Intraarticular injections are also implemented to manage OA ranging from corticosteroids to hyaluronans to more recently platelet rich plasma and even stem cells while several other injection therapies are presently being studied. The goal of developing new treatment strategies for OA, through the OACTIVE model is to prolong the need for total arthroplasty

which should be utilized only if other strategies have failed. Arthroscopy has been commonly used for many years to treat OA to address degenerative articular cartilage, however, several high-quality studies have shown that it is not a very effective treatment for the majority of cases and should generally not be considered when managing OA. Improving the management of OA requires a multi-faceted treatment approach along with continuing to broaden our understanding of this complex disease so that therapeutic advancements can continue to be developed with the goal of preventing further disease progression and even potentially reversing the degenerative process.

Table 6.1: State of the Art in Osteoarthritis treatments

Acronym	Objectives
ADIPOA2 (H2020-PHC-2014)	Clinical trial of autologous adipose-derived mesenchymal stromal cells (ASC) in the treatment of mild to moderate osteoarthritis.
Hy2Care (H2020-SMEINST-1-2016-2017)	Injectable hydrogels made of advanced nanomaterials that integrate the repair tissue
REGHA (H2020-SMEINST-1-2014)	This therapeutic approach is applied by local injection into the joint of a pharmacological active patented molecule stimulating chondrocytes proliferation, the only cells producing and maintaining the cartilage matrix.

In our approach for OA treatment all factors are taken under consideration and only when needed, in the proper stage of the condition, the suitable suggestion will be given. It is evident that in OA management not all treatments are suitable in all cases, patients have different phenotype, symptoms and progress. For that important reason, our OACTIVE for personalized, predictive and preventive management of OA is innovative and beyond the state of the art.

Augmented Reality (AR) superimposes a computer-generated image on a user's view of the real world. It not only preserves some benefits of leveraging Virtual Reality (VR) such as fully controlled setting and measurable feedback, but also needs less computation time to model the 3D environment. Moreover, to interact with a non-immersive VR setting (which is widely used in motor and cognitive rehabilitation), the subject needs to perform at least one extra transformation to translate the virtual world's coordinate to the body-centered coordinate, which could be challenging to elderly patients with cognitive difficulties. This is not needed in an AR setting. In AR, patients experience a more engaging and natural interaction since virtual objects can be manipulated in an intuitive and natural way to maximize learning activities of daily living (ADL's). The haptic feeling of the real objects could bring on a more natural interaction. There is consensus amongst therapists that as the interaction of patients with the physical environment is reduced, their ADL's recovery starts to deteriorate. Thus, an essential factor to successful recovery is to increase the patient's level of interaction with their environment. With the advances of AR interaction technology, AR games combine traditional digital games and physical activities providing alternative leisure opportunities for older adults. Different from traditional digital games that rely on joysticks or related controllers to receive players' feedback and signals, AR games are obviously more enjoyable by providing instant, positive feedback on current actions as well as a clear picture on long-term performance. Given their widespread availability and (relatively) inexpensive price tags, handheld gaming devices and mobile phones are now capable of supporting AR. An AR gamer environment motivates the individual and makes him/her train more often and for a longer period of time without getting tired. The continuous feedback provided by the AR therapeutic programs builds and strengthens the user's motivation.

6.1. AR treatment evaluation

Digital games hold significant promise for enhancing the lives of older adults, since they require progressively more accurate and more challenging judgments at higher speed, and the suppression of irrelevant information, which leads to positive neurological changes in the brain systems and improves the cognitive level of the patients. In addition, several clinicians and caregivers around the world suggest treating patients with virtual and/or augmented reality games/exercise programs that allow them to have fun while stretching their physical and cognitive capabilities. In VR games/rehabilitation programs the user is connected to the VR system as part of the input/output loop, allowing individuals to provide input to the virtual environment (VE) and experience the result of that input. VR games/rehabilitation programs using specialized interface devices (e.g., Wii video game, Nintendo Wii Fit Plus Essentials Workout Kit, My Fitness Coach, WiiFit, iDance, etc.) have been applied to improve motor skill rehabilitation of functional deficits including reaching, hand function and walking.

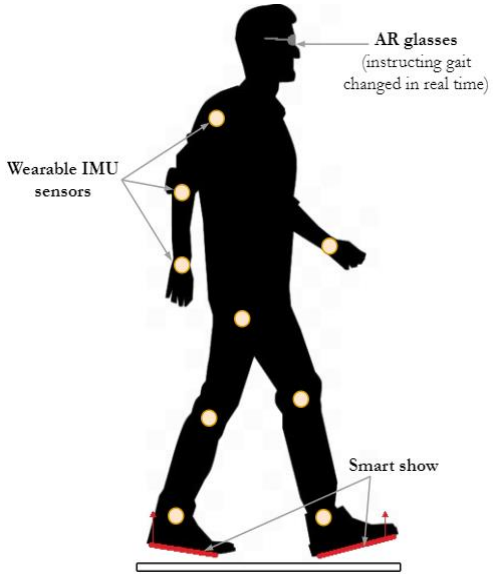

Figure 6.1: Augmented Reality based gait re-training system.

OACTIVE will rely on the AR gaming concept to enable personalized interventions, both at the clinical assessment and rehabilitation levels, which are usually not available with traditional OA treatment methods. It aims at exploiting haptic and vision technologies to provide OA patients with assistive visual and contact feedback while performing games/rehabilitation as well as medical staff with biomechanical indicators for assessment and clinical decision support. More specifically the AR games will be used for the treatment of OA by using the gait retraining method. In OA gait retraining is proving to be an effective treatment for correcting gait alterations. Current gait retraining methods for knee OA rely on the use of simple biomechanical models for calculating the external knee adduction moment (KAM) as a target variable to control during the gait retraining interventions. Decreasing the early stance peak KAM has been reported to also decrease pain, disease progression, and disease severity in OA patients. Recent studies have explored real-time visual and vibrotactile feedback to enable subjects to relearn their gait with reductions in KAM that ranged from 7% to 48%. Similar procedures are currently being developed and tested for treating hip OA. Retrained gaits with minimal in vivo tibiofemoral contact forces may be more effective than gaits with minimal KAM peaks to the treatment of OA condition because OA progression is directly related to tibiofemoral contact forces and only indirectly to KAM peaks. The availability of electromyography (EMG)-driven models that can predict accurate estimates of tibiofemoral forces in real time will offer the possibility of performing joint contact force-based gait retraining to any subject through AR games. Inertial sensors (IMU's) will be placed on lower limb segments to provide estimates of joint kinematics. Together with EMG, these data will be used within a surrogate contact model that includes muscle force estimates using an EMG-driven musculoskeletal model. Estimates of peak contact force or pressure are then used in a data-driven gait prediction model to provide a stimulus to the subject in real-time via AR game to alter their gait (Figure 6.1).

Personalized gait retraining intervention requires an estimation of the internal state of the musculoskeletal system. Current approaches rely mostly on kinematic measurements to evaluate the performance of a subject. However, a pure kinematic analysis is not enough to quantify the stress of the knee during movement, because one must also account for the forces that act on the system. In terms of dynamics, most studies are limited only to inverse dynamics analysis due to the increased computational burden required for calculating the muscle forces and joint reaction loads in real-time. The system developed in the OACTIVE project moves one step further by estimating the muscle forces and joint reaction loads in real-time. This section presents the architecture and implementation details of the underlying system. The term "internal state of the musculoskeletal system" refers to the quantities that cannot be measured non-invasively, which however can be determined from other experimental measurements, and are important for quantifying the performance of the subject. Examples include (i) the kinematics of the skeletal system (e.g., anatomical joint angles), (ii) joint loads, (iii) muscle forces, and (iv) joint reaction loads. This information can be used to determine whether a subject is walking in an optimal manner by examining his/her reaction loads at the diseased knee and provide real-time feedback that can aim in reducing the contact pressures.

Figure 4.2 presents a diagram of the different stages that are being solved in real-time in order to calculate the internal state of the musculoskeletal system in real-time from kinematic and dynamic measurements. The experimental measured positions (markers or orientations) are fed to the Inverse Kinematics (IK) module, which determines the generalized model coordinates that best match the experimentally recorded motion. The kinematic analysis is very important for the next stages. Following Inverse Dynamics (ID) is performed to calculate the generalized forces that satisfy the Equations of Motion (EoMs), provided any externally applied force. Static Optimization (SO) can be employed to estimate the required muscle activation and forces that satisfy both the motion and the physiological muscle constraints. This step can be further improved by using information from the EMG recordings if available. The muscle forces are essential for the calculation of the joint reaction loads that is the final stage of the system.

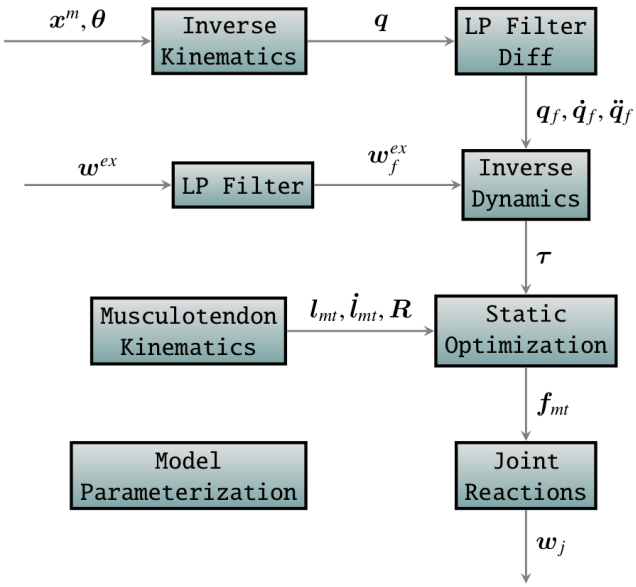

Figure 6.2: Architecture diagram of the different stages that are solved in order to determine the internal state of the musculoskeletal system in real-time.

In order to perform any kind of IK analysis one must obtain the motion of the body segments from the recorded kinematics. The motion can be obtained either by recording the position of the attached markers or by measuring the 6D motion of IMUs. The next step would be to determine the evolution of the generalized model coordinates that best match the experimentally recorded motion. The IK method goes through each time step of the recorded motion and computes the generalized coordinates which positions the model in a pose that best matches the experimental measurements. More formally, this is expressed as a weighted least squares problem, whose solution aims to minimize the following error.

$$\sum_{i}^{measure} w_i \left|\left|x_i^{exp} - x(q)\right|\right|^2$$

where w_i represents the weight for measure i, x_i^{exp} the experimental measure and $x_i(q)$ the model measure that depends on the pose. Each measure has a weight associated with it, specifying the influence of the particular term on the overall error. The least squares problem can be solved using a general quadratic programming solver. In terms of performance, our implementation is able to solve the IK for a model with 23 Degrees of Freedom (DoFs) in less than 0.5ms without compromising the accuracy of the solution ($< 2cm$ Root Mean Square Error (RMSE)).

The next stage (Figure 1 second level) is to provide a solution to the ID problem. In order to solve for the generalized forces that satisfy the EoMs one must first calculate the first and second derivative of the kinematics. Numerical differentiation is a challenging problem, especially when we deal with real-time application, where one does not know the value of the kinematics in future time instances (casual systems). The kinematics obtained from IK are noisy, resulting in bad approximations of higher order derivatives. Therefore, appropriate filtering must be implemented in order to reduce the noise artefacts without introducing lags and errors in the filtered signals. However, even if one filters the signal there will be always jitter in the higher order derivatives. To this end, we adopted the following strategy making use of the high sampling rates of the kinematics measurements in order to implement higher order IIR filters without introducing lags in the calculated quantities:

1. IK is solved in a separate thread (T1) configuring the motion capture system to operate with high sample frequencies and the solution of IK is stored in a thread safe buffer.
2. In a separate thread (T2) we solve the rest of the stages (ID, SO, and JRA).
3. Each time we proceed into T2 we read the buffer containing the solutions obtained by IK.
4. A low pass Butterworth filter of order > 50 is applying on the most recent kinematics data.
5. Then smooth splines are fitted in order to compute higher order derivatives.
6. Finally, we choose the sample delay in order to evaluate the kinematics and their derivatives. Low delay (2-3 samples) will result in low latency (lag) in the estimated quantities. Higher delay (non causal filter) results in larger latencies but permits better approximation of the higher order derivatives as compared to offline analyses.

Figure 4.3 depicts the results obtained by applying the aforementioned procedure to filter the kinematics and obtain higher order derivatives. For a comparison, a conventional 4th order Butterworth filter is applied to the signal (right column). The latter is the filtering adopted in the literature, which evidently, can lead to bad approximations of the higher order derivatives. The

increased complexity of the proposed filter satisfies the requirement for accuracy that can propagate to errors in the estimated quantities of the next stages.

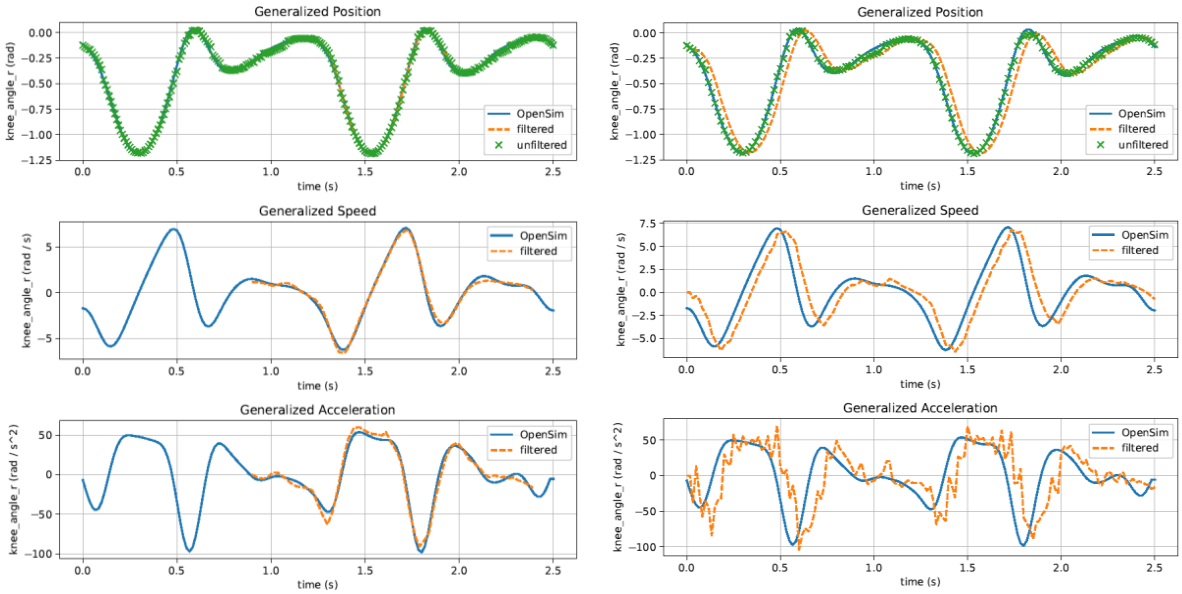

Figure 6.3: Calculated kinematics of the knee joint. Unfiltered kinematics from IK are filtered by the process outlined in this section. The filtered kinematics are compared against results obtained from OpenSim when performing the analysis in an offline mode, meaning that OpenSim can apply non-casual filters that can better remove the noisy artefacts in the signal. The figure on the right depicts what will be the actual result of the filtered signal if one used conventional 4th order low pass Butterworth filter.

For the given kinematics describing the movement of a model and any externally applied force, the ID method determines the generalized forces (e.g., net forces and torques) at each joint that satisfy the movement. More formally (see equation below), they are solved for the unknown τ provided q, $\dot{q}$ and $\ddot{q}$. Since q is calculated from IK, $\dot{q}$ and $\ddot{q}$ must be obtained using numerical differentiation. Discontinuities in the generalized coordinates can lead to numerical singularities during the evaluation of higher order derivatives (as discussed previously) and it is thus advised to apply filters to remove any undesirable artefacts. The following notation is used for describing the EoMs.

$$M(q)\ddot{q} + f(q,\dot{q}) = \tau$$

where $M \in R^{nxn}$ denotes the symmetric, positive definite joint space inertia mass matrix, n the number of models DoFs and $q, \dot{q}, \ddot{q}$ the joint space generalized coordinates and their derivatives with respect to time. The term f models all internal and external applied forces (e.g., gravity, Coriolis, GRF, etc.), whereas τ is the vector of applied generalized forces that actuate the model. Most of the quantities in the equations are a function of the generalized coordinates and derivatives, thus this dependency will be omitted for simplicity.

When ID is solved, it is preferred to ignore model constraints since a set of $q, \dot{q}$ and $\ddot{q}$ may not necessarily satisfy the constraint algebraic equations. As a result, if a model contains kinematic constraints ID may give rise to unreliable estimates of the generalized forces. In this case, ID-based methods may provide a suitable solution. Another important misconception is that ID does not necessarily depend on externally applied forces, however, it is common sense that they should be accounted for since their omission will significantly alter the result.

Moving to the solution of ID and as indicated in Figure 15, the external forces w^{ex} which are measured are low pass filtered. The filtered kinematics as well as filtered external forces are fed to the ID system which is also a very fast operation ($< 0.5ms$). As indicated in Figure 6.4 where we compare the result obtained from the real-time analysis to the one obtained by OpenSim in an offline mode, good agreement is observed.

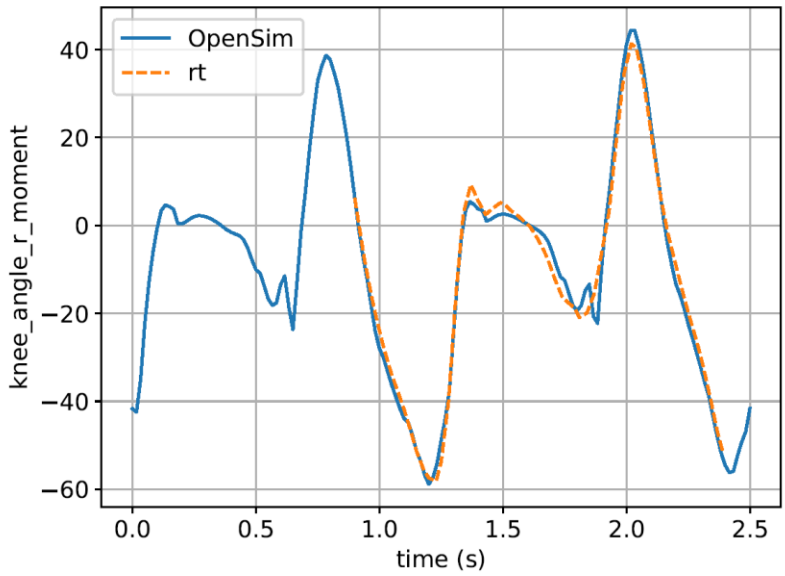

Figure 6.4: Estimated knee moments from ID. Comparison is made by performing an ID analysis through OpenSim in an offline mode, where the whole measured kinematics and external forces are available.

Static Optimization (SO) is a method for determining the muscle forces required to satisfy the motion and forces of the model. The OpenSim internal implementation requires 1s/iteration to solve the SO problem, which is unacceptable for real-time applications. There are many ways to implement and solve the SO problem. Here we will outline an approach which can result in a solution that can operate with 10ms delay for a model containing 92 muscles. The following optimization problem is being formulated.

$$\sum_{m=1}^{muscles} f_m^2$$

$$s.t. R(q)f_m = \tau and f_m \geq 0$$

where, $\boldsymbol{f}m$ is muscle force m, $\boldsymbol{R}$ is the moment arm matrix and $\boldsymbol{\tau}$ the generalized forces obtained from ID. The objective of the optimization is to compute the minimal muscle force $\boldsymbol{f}m$ needed so that the contribution of the muscle forces to the joint torques to be equal to the torques computed by ID for each time instance. It is called "static" optimization because the performance criterion (i.e., the cost index) is confined to quantities that can be computed at any instant in time during a simulation. The bottleneck of this formalization is (i) that one must calculate the moment arm matrix $\boldsymbol{R}(\boldsymbol{q})$ which is a time-consuming operation provided that the muscles have complex routing mechanisms, and (ii) the optimization must find a solution with less iterations to achieve real-time performance (preferably <7).

To address the first problem, we derived and precomputed a symbolic representation of the moment arm matrix as a function of the generalized coordinates. In order to derive a symbolic

representation, multivariate polynomial fitting was performed on samples of the muscle moment arm at different configurations. To reduce the complexity and improve the robustness of the fit, we determined the coordinates affecting each element in the moment arm matrix, by identifying the DoFs spanned by each muscle. The sampled and symbolically obtained moment arm of the vastus intermedius (a mono-articular muscle) at the knee joint as a function of the knee flexion angle and the moment arm of the hamstring muscle at the knee joint as a function of the hip and knee flexion angles are compared. The second problem is addressed by providing the previously obtained solution as an initial guess for the next optimization. Furthermore, one could adjust the convergence tolerance of the optimization in order to reduce the iterations needed to obtain a solution within a desired accuracy of digits. The close match between the muscle forces obtained by the proposed real-time optimization and the one obtained from OpenSim can be graphically depicted.

The visualization back-end was developed with the use of the Unity 3D game engine. Unity 3D was selected because it offers multiple tools for visualization and gamification and there is already support for multiple AR and VR devices, such as the Meta 2 AR headset. A fast communication between the real-time musculoskeletal analysis and the visualization process was necessary in order to develop a game with real-time interventions and feedback. To achieve that kind of demanding communication, especially in latency, a shared memory interface was developed. The shared memory interface utilizes the OS capabilities of mapping a memory block by one process and then sharing the same block with another process. Providing access to the same memory block, the data is available on both processes with almost no delay compared to communication techniques utilizing the network stack (i.e., sockets). The shared memory interface limitation is that the processes must run on the same computer. The implementation followed a triple buffer design in order to ensure that the interface is thread-safe, meaning there are no race conditions or memory inconsistencies.

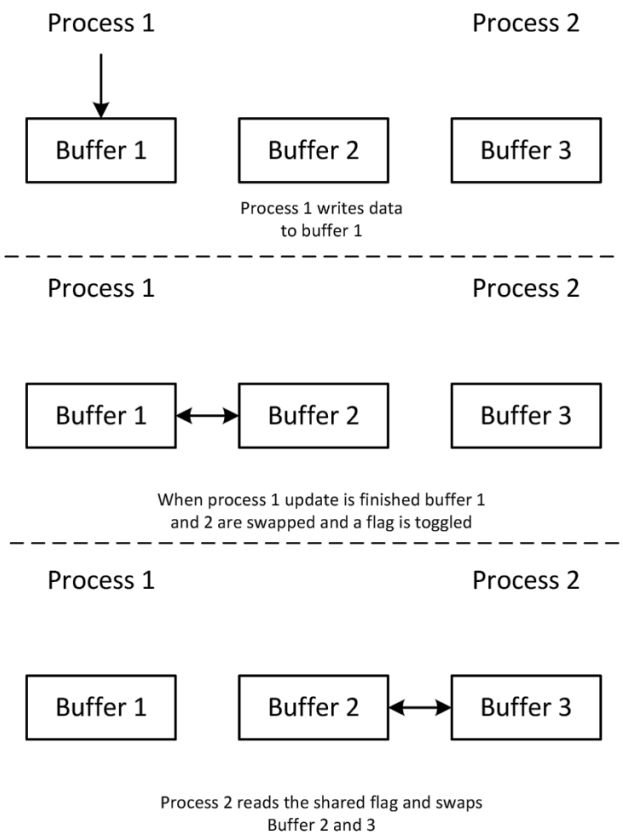

Figure 6.5: Triple buffer implementation flow diagram

The program flow follows the steps bellow:

1. The user starts the visualization process.
2. It bounds the memory block in order to share with the simulation process.
3. It starts the simulation process via an OS command with the map of the shared memory block as a parameter.
4. The simulation process performs some initialization steps and serializes the model structure (joints, bones and muscles) in a JSON format.
5. The simulation process provides the json to the visualization process.
6. The visualization process constructs the skeleton model along with the muscles by deserializing the json.
7. Then, asynchronously, the simulation process sends all the necessary data to the visualization process through the shared memory interface on each frame, and the visualization process updates the model until the user terminates the application.

7. Conclusions

D9.1 Evaluation Toolbox Documentation is a part of the OACTIVE project where the project's infrastructure, from data collection to model fitting and prediction tasks, is evaluated and validated. For this purpose this deliverable focuses on the presentation of the machine learning algorithms and the data collection scheme. The validation of the process strictly ensures the validity of our results. As a result, through the process of hyperparameter tuning for each classification and clustering task, we end up with the best possible evaluation results of the OACTIVE's predictions. The extracted information is used for personalized AR treatment plans for each patient individually. Finally, the evaluation and validation of the AR treatment plans are implemented through personalised treatment predictions, visualizations, analytics and alerts through a patient friendly environment.

8. Acknowledgment

This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 777159.

Disclaimer

The Horizon 2020 project has been made possible by a financial contribution by the European Commission under Horizon 2020 research and innovation programme. The publication as provided reflects only the author's view. Every effort has been made to ensure complete and accurate information concerning this document. However, the author(s) and members of the consortium cannot be held legally responsible for any mistake in printing or faulty instructions. The authors and consortium members reserve the right not to be responsible for the topicality, correctness, completeness or quality of the information provided. Liability claims regarding damage caused by the use of any information provided, including any kind of information that is incomplete or incorrect, will therefore be rejected. The information contained in this document is based on the author's experience and on information received from the project partners.

○