Original research

Computerised patient-specific prediction of the recovery profile of upper limb capacity within stroke services: the next step

Ruud W Selles,1 Eleni-Rosalina Andrinopoulou,2 Rinske H Nijland,3 Rick van der Vliet,4,5 Jorrit Slaman,6 Erwin EH van Wegen,7 Dimitris Rizopoulos,2 Gerard M Ribbers,4,6 Carel GM Meskers,7 Gert Kwakkel6,7

ABSTRACT

Introduction Predicting upper limb capacity recovery is important to set treatment goals, select therapies and plan discharge. We introduce a prediction model of the patient-specific profile of upper limb capacity recovery up to 6 months poststroke by incorporating all serially assessed clinical information from patients.

Methods Model input was recovery profile of 450 patients with a first-ever ischaemic hemispheric stroke measured using the Action Research Arm Test (ARAT). Subjects received at least three assessment sessions, starting within the first week until 6 months poststroke. We developed mixed-effects models that are able to deal with one or multiple measurements per subject, measured at non-fixed time points. The prediction accuracy of the different models was established by a fivefold cross-validation procedure.

Results A model with only ARAT time course, finger extension and shoulder abduction performed as good as models with more covariates. For the final model, cross-validation prediction errors at 6 months poststroke decreased as the number of measurements per subject increased, from a median error of 8.4 points on the ARAT (Q1–Q3:1.7–28.1) when one measurement early poststroke was used, to 2.3 (Q1–Q3:1.7–7.2) for seven measurements. An online version of the recovery model was developed that can be linked to data acquisition environments.

Conclusion Our innovative dynamic model can predict real-time, patient-specific upper limb capacity recovery profiles up to 6 months poststroke. The model can use all available serially assessed data in a flexible way, creating a prediction at any desired moment poststroke, stand-alone or linked with an electronic health record system.

INTRODUCTION

Predicting poststroke motor recovery is important to set realistic and feasible treatment goals, select effective therapies and determine adequate discharge policy and long-term management.1 In addition, prediction models can assist in clinical trial design to select patients that may benefit most for the treatment of interest; for example, by including only those patients that have the potential to benefit from the intervention under study.2 3 For health economics, prediction models allow benchmarking or help correct for casemix variability and for estimating the expected costs in healthcare.1

Until today, a large number of prediction models for poststroke upper limb motor recovery have been developed (see for a review, see reference 2). Most of these models are based on clinical markers, such as voluntary control of shoulder abduction and finger extension (SAFE model) early poststroke4 (for a review, see reference 5). Other models include neurophysiological markers such as intactness of the corticospinal pathway, measured with, for example, transcranial magnetic stimulation (TMS) (for review, see references 6 7).

Typically, current prediction models are developed with classic linear or logistic regression techniques. These models assume that candidate determinants and recovery outcomes are measured at predefined times poststroke; for example, determinants assessed at 48 hours poststroke are used to predict recovery at 6 months poststroke.2 Classic regression approaches, however, have several limitations. First, the accuracy of the prediction depends on the ability to assess patients at predefined timepoints poststroke.3 In clinical practice, measuring at predefined timepoints is challenging for many practical reasons. A second limitation of classic regression models is that they only predict one specific follow-up moment, for example, at 3 or 6 months poststroke. To predict an outcome at any given moment poststroke would require a large number of regression models. Third, a limitation of regression-based models is that a nonlinear time course of recovery is not taken into account, as regression models are based on only two timepoints (ie, baseline and outcome).

Since prospective cohorts of stroke recovery with serial measurements have shown both non-linearity and high interindividual variability of stroke recovery,1 8 as an alternative to regression models, mixed models can be used. Mixed models are ideally suited when the individual trajectory of recovery over time is influenced by characteristics that vary from patient to patient and explicitly account for the correlations between repeated measurements within each individual patient.5 In a mixed model, data from all assessments contribute to more precise estimates and characterise an individual patient’s recovery pattern over time.9 Thus, with mixed models, more precise predictions may
be achieved by including multiple repeated measurements from an individual patient, allowing patient-specific predictions with limits of uncertainty and taking all serially assessed information of that subject into account. In such a model, each new data point with new health status information will decrease the uncertainty of the individual prediction, taking into account changes in neurological status of the patient (‘dynamic prediction’).

In the current study, we aimed to develop an online-available mixed-effects prediction model for recovery in the first 6 months poststroke of the upper limb capacity, the ability to perform unilateral tasks like picking up a cup, transporting blocks, manipulating small objects, measured with the Action Research Arm Test (ARAT) scores. Subsequently, we investigated how this dynamic model can take time-dependent clinical improvement of patients into account as soon as new information is available. After comparing several model alternatives, we cross-validated the accuracy of the predictions and developed an online patient-specific model in which the predicted outcome can be used at stroke units and throughout the care chain for predicting the recovery profile of upper limb capacity.

**MATERIALS AND METHODS**

**Patients and measurements**

We combined data on upper extremity capacity recovery in first-ever ischaemic anterior circulation stroke patients from four early poststroke prospective cohort studies with repeated measurements in time: the EXPLICIT, EPOS, 4D-EEG, and EXPLORE cohorts.

All datasets contain repeated measurements of the ARAT scores after the stroke. Patients in the 4D-EEG and EXPLORE cohorts were recruited within 3 weeks poststroke. Patients in the 4D-EEG study and the EXPLORE study were measured in the first week poststroke and after 5, 12 and 26 weeks. In the EXPLICIT study, patients were measured 1, 2, 3, 5, 12 and 26 weeks poststroke and in the EPOS study in the first 3 days, at day 5 and 9 and after 6 months. All studies included adult patients with a monoparesis or hemiparesis within the first 72 hours after stroke onset, without disabling medical history and with no severe deficits in communication, memory, or understanding that impede proper measurement performance.

Patients received standard rehabilitation treatment according to the Dutch rehabilitation guidelines, which are in agreement with current international rehabilitation guidelines. In the EXPLICIT study (N=159), half of the patients with an unfavourable prognosis received electromyography-triggered neuromuscular stimulation and half of the patients with a favourable prognosis received modified constraint-induced movement therapy.

**Testing covariates**

In line with recent international recommendations, upper extremity capacity measured with the ARAT was used as the dependent variable in the model. The ARAT has 19 tasks divided into four subdomains: grasp, grip, pinch and gross movement. The ARAT score ranges from 0 to a maximal score of 57 points, and the ARAT has excellent clinimetric properties.

We included the following potential demographic and clinical covariates variables during the model development: age, gender, affected bodyside, dominant side, Bamford scale (lacrimal cerebral infarct/partial anterior circulation infarcts/total anterior circulation infarcts), and administration of alteplase (rt-PA). Neurological deficits were assessed with the National Institutes of Health Stroke Scale (NIHSS; range 0–42). NIHSS item 8 reflecting somatosensory deficit (range 0–2) and item 11 measuring extinction and inattention (range 0–2) were used as separate covariates. In addition, we used the O-letter cancellation test to assess visuospatial neglect using a cut-off of 2 missing O’s or more on the hemiplegic side.

As time-dependent variables, we included the Fugl Meyer Upper Extremity (FM-UE) motor score (range 0–66). In addition, the item voluntary finger extension (FE: no=0 or yes=1 or 2 points) of the FM-UE was used as a separate item for deriving the SAFE model. Upper and lower limb strength were measured with the Motricity Index leg and the Motricity Index arm score (range 0–99). The item ‘shoulder abduction’ was also used separately (six levels, range 0–33) for the SAFE model.

**Prediction model development**

For model fitting, we included all patients with at least three repeated measurements from which the first and last measurements were at least 12 weeks apart. We excluded patients for the model fitting who showed an FM-EU decrease beyond the upper limit of the clinically important difference for the FM-UE (7.25) in the first 6 months. We did, however, include these patients in the cross-validation to present a fair estimate of prediction accuracy.

As a first step in the model fitting, we compared mixed-effects models with different time structures to investigate whether a nonlinear time, compared with a linear time structure, was more

**Table 1** Summary of the five different model structures that were considered

<table>
<thead>
<tr>
<th>Model 1 (most extensive model)</th>
<th>Model 2 (all covariates, all main effects, only the significant interactions with time)</th>
<th>Model 3 (only significant main effects and significant interactions with time)</th>
<th>Model 4 (SAFE model)</th>
<th>Model 5 (minimal model)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time dependency of ARAT alone</td>
<td>Covariate: all main effects</td>
<td>Covariate: all interaction effects (with time)</td>
<td>Covariate: only significant main effects</td>
<td>Covariate: only significant interaction effects (with time)</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

For further description of the models, see prediction model development in the methods. The safe model was used as the final model for cross-validation.

ARAT, Action Research Arm Test; SAFE, Shoulder-Abduction-Finger Extension model.


appropriate to capture the mean recovery profile of the ARAT (fixed effects) and the subject-specific ARAT evolutions (random effects). For the models with nonlinear time structures, we used natural cubic splines with a different number of knots. The knots were placed at the quartile points in the case of 1, 2 and 3 knots and placed manually in the case of 6 knots.

Subsequently, we developed models with different fixed effect structures, starting from the most complex model with all covariates mentioned above (see also Table 1). More specifically: model 1. ARAT scores as a function of all available covariates with their main effects and interaction with time; model 2. ARAT scores as a function of all covariates with all their main effects but only the significant interactions with time; model 3. ARAT scores as a function of only the significant main effects and the significant interactions with time; model 4. ARAT scores as a function of time, the SAFE model and their interaction with time and model 5. ARAT scores as a function only of time.

Cross-validation and model comparison
We selected the optimal model from the above-mentioned models by comparing the prediction accuracy of the models. To express model accuracy, we calculated the absolute error between the predicted and the observed (true) ARAT measurement and presented the distribution of the errors using box-plots with the median, the IQR, the lower whisker presented as Q1 - 1.5 * IQR and the upper whisker presented as Q3 + 1.5 * IQR. Values outside the range (Q1 - 1.5 * IQR; Q3 + 1.5 * IQR) were considered outliers and were removed.

We performed a fivefold cross-validation procedure by splitting the data into five sets, fitted the models using four of them, and then predicting the recovery profiles for the patients from the fifth set. Although the model predicts the complete recovery profile and not only one future timepoint, to express prediction error, we compared predicted and actual recovery at 6 months poststroke. The predictions were obtained several times by adding an extra measurement for ARAT each time. We repeated this procedure 10 times.

For all analyses, R (V.3.6.1) software was used (freely available at https://cran.r-project.org). The packages nlme (V.3.1.144) and JMbayes (V.0.8.85) were used. In particular, we used the function lme() from the package nlme to fit the models and the function IndvPred_lme() from the package JMbayes to obtain the dynamic predictions.

Online patient-specific prediction visualisation
To allow clinicians and researchers to use the prediction model for individual stroke patients, we developed an online visualisation for the real-time prediction of the recovery profile of upper extremity development and the 68% and 95% prediction intervals, expressing the prediction uncertainty. For this, we use the shiny package in R to allow uploading a text file with recorded ARAT data and covariate data, which will automatically create a patient-specific prediction of the ARAT development over time.

Within our centre, we developed an infrastructure to couple this visualisation to data capture software (GemsTracker), so that recorded data from individual patients can be used to display the predicted outcome without further data entry or analysis. Finally, we published the source code of the prediction models web-based, allowing other centres to develop similar real-time prediction tools (see details statistics and code in online supplemental file 1).

RESULTS
Patients and measurements
The prospective data set consisted of 450 patients with a first-ever ischaemic hemispheric stroke (see table 2), of which 52% were males, mean age at baseline was 65 years, and mean follow-up days was 166 days. Patients were serially assessed with a median of six times (25% quantile: 4; 75% quantile: 8). The mean ARAT early poststroke was 14 (SD 19); the ARAT distribution was skewed by a large sample of patients with an ARAT of 0 or 1 scores early poststroke and a less equal distribution of ARAT scores throughout the rest of the range of scores (Q1=0, median is 1, Q3 is 29, maximum score 56).

Figure 1 shows all 6-month recovery profiles and highlights five individual patients with different recovery profiles. The figure illustrates both the heterogeneity and non-linearity of the recovery, with some patients showing little or no recovery, others reaching the maximum ARAT score within the first 3 weeks poststroke and again others starting with a relatively high
ARAT score already early after stroke and typically reaching the maximum ARAT score within a few weeks.

Prediction model development
Visual inspection of the individual repeated measurement profiles of the ARAT (see figure 1) indicated that the final model needs to allow for nonlinear evolutions over time. After comparing models with different nonlinear evolutions, we observed small differences between using two, three and six knots (See online supplementary figure 1). However, since the main goal was to use the model for predicting the recovery profile of the ARAT most precisely, we continued with the most flexible model based on six knots. This model also provided better predictions when checking several individual prediction plots.

Cross-validation prediction accuracy and model selection
Comparing the prediction accuracy of the different mixed-effects models and taking the first measurement of each patient as a predictor, we found mean cross-validation errors at 6 months poststroke of 10.1–105 ARAT points for the first, second and third model, 8.4 for the fourth SAFE model with only FE and SA and 17.3 for the fifth model that includes only time as a covariate (see Supplementary Figure 2 for more details). Based on these results, we decided to further continue with the fourth (SAFE) model for predicting the recovery profile of the ARAT.

Figure 2 illustrates the prediction errors at 6 months poststroke using the SAFE model. As shown, increasing the number of serial measurements decreased the absolute error of prediction of ARAT from 8.4 points on the ARAT (Q1–Q3: 1.7–28.1) to 2.3 (Q1–Q3: 1–7.2) when seven serial measurements were used for predicting the outcome at 6 months, of which generally one measurement as 6 months poststroke, one at 3 months poststroke, and the others between 2 days and 6 weeks poststroke.

Online patient-specific prediction visualisation
Figure 4 visualises predicted recovery profiles (with a 68% and a 95% prediction interval) using the SAFE model for two patients. The predicted recovery is nonlinear, especially in the patient with low initial ARAT scores. Patient 1 has a low ARAT score at baseline and a moderate predicted ARAT score at 6 months. Patient 2 has a higher ARAT score at baseline and the predicted score at 6 months is >50 points. Prediction plots of the same patients as well as two additional patients, including a 68% and a 95% prediction interval, are shown in online Supplementary Figure 3.

Online access to a real-time tool to predicting upper extremity recovery for an individual patient can be found at https://embio-statistics.shinyapps.io/DynamicPredictionARATapp/, including instructions on how to enter data from individual patients. This is also the model that, within our own setting, is coupled to an online structured data entry system called Profits), as a model of
Cerebrovascular disease

how more computationally complex prediction models can be coupled to electronic health record data or other types of online healthcare data collection systems.

Discussion

In the current study, we introduce an innovative mixed-effects model able to predict the patient-specific time course of upper limb recovery, including the associated limits of uncertainty within the first 6 months poststroke. We found that a model with only ARAT time course, FE and SA performed as good as models with more covariates for predicting the ARAT over time. In this final model, cross-validation prediction errors at 6 months poststroke decreased as the number of measurements per subject increased, from a median error of 8.4 points on the ARAT when one measurement early poststroke was used, to 2.3 for seven measurements. Linking this model with an electronic health record system at stroke units and rehabilitation wards will allow clinicians to obtain and visualise real time a patient’s recovery profile early poststroke.

In contrast to prognosis based on regression models, a major strength of current dynamic prediction modelling is that individual patient-specific neurological and functional recovery profiles are taken into account. By making the model accessible online and potentially connected to electronic health record systems, professionals can obtain predictions without needing to perform complex computations. At any given moment in time, such as in individual patient consultation or multidisciplinary team meetings, an individual prediction of the recovery profile of the ARAT may be generated, taking into account all available data points.

We found that a model with only time-dependent improvements in SA and FE predicted recovery as accurately as more complex models that included covariates such as age, gender, the Bamford classification, the NIHSS or the full FM-UE score, which is in line with the SAFE model. In addition, we found that the predictions become more accurate when repeated clinical measurements became available in the first 8 weeks post-stroke, which is in line with a recent model on FM-UE function and in line with the recent recommendations of the stroke recovery and rehabilitation round (SRRR) the Table group for clinical assessment. Finally, we showed that this dynamic model can be visualised and used online for clinicians working at stroke units and rehabilitation wards in an anonymised way by a simple text file as input, as shown in figures 3 and 4 and in the online application at https://emcbiostatistics.shinyapps.io/DynamicPredictionARATapp/.

Since the implementation of prediction models in clinical care depends on the ease of use, an important benefit of the present algorithm is that it can be an integral part of existing electronic health record systems at stroke units, allowing direct access to the model input variables. Since the current study also shows that serial measurements in time after discharge from stroke units have an important added value in the accuracy of individual predictions, preferably, patient-specific prediction modelling should be part of electronic health record systems that span the entire chain of care of stroke services. The current

Figure 3  Cross-validation accuracy of the fivefold cross-validation in for different levels of baseline (early poststroke) ARAT scores (left, middle and right panel) for the safe model as a function of the time of the last observed outcome (instead of the number of measurements used). Furthermore, we categorised the results by baseline ARAT group. The accuracy was defined as the difference between the predicted ARAT score at 6 months poststroke from the cross-validation and the measured ARAT score at the same time and displayed as median, (IQR: Q1=25th percentile and Q3=75th percentile), lower whisker presented as Q1−1.5 * IQR and upper whisker presented as Q3 +1.5 * IQR. The results can be different from figure 3 since some patients might have one measurement during the first week while other patients might have 2–3 repeated measurements in the first 6 months. ARAT, Action Research Arm Test.
Cerebrovascular disease

The current study predicts the 6-month outcome on the ARAT scale based on a population of first-ever, ischaemic hemispheric strokes measured from 2005 onwards in the Netherlands. The predicted time course shown is irrespective of differences in type and intensity of rehabilitation therapies applied, which were applied according to the Dutch Stroke guidelines. Currently, the recommended amount of usual exercise therapy is only twice 20 min per working day, which is probably insufficient to cause meaningful clinical differences in trajectories. However, one may assume that higher doses of task-specific exercise therapy may have caused beneficial effects and could serve in future models as one of the covariates.

In the present study, we focused on predicting the ARAT as a showcase for dynamic prediction modelling. The next step will be to build patient-specific models for predicting the time course of basic ADLs and walking ability, including community walking poststroke. These steps should be seen as essential for improving current discharge policy and stroke management within current stroke services. For this purpose, preferably, the data entry should be part of the existing stroke guidelines and executed by clinicians who are trained to apply the clinical measurements in a valid way.

A limitation of our study is that we were restricted to clinical time-dependent variables alone, such as synergism, strength, voluntary shoulder and FE. With that, our clinical model has room for improvement, especially in those patients with low baseline motor scores who may or may not show neurological

Figure 4 Typical examples of the predicted ARAT recovery for two patients. The dotted vertical line represents the time of the last follow-up. The circles represent all the ARAT measurements available for that patient until that specific moment, while the solid line represents the predicted ARAT recovery. The shaded areas indicate the 68% (lighter shade) and 95% (darker shade) prediction intervals. From a clinical perspective, the errors in the cross-validation provide the best estimate of what the error in predicting the outcome for a new patient will be and may, therefore, be most clinically relevant. For each patient, the predicted recovery is illustrated at a first and a second time point, not necessarily corresponding to the first and second available measurements from a patient. The data of the same patients can be downloaded in the online APP to visualised predictions at all time points: https://emcbiostatistics.shinyapps.io/DynamicPredictionARATapp/. ARAT, Action Research Arm Test.
recovery. This is in line with previous studies indicating less accurate outcome prediction in patients with low FM-UE scores (<18 points) within the first week poststroke. We think that our current model may serve as a starting point for a more detailed stratification in recovery and rehabilitation trials, based on the patient's potential for neurological and functional recovery poststroke. We may further refine the identified subgroups of recovery and increase precision at an individual level by targeting big data, based on the recently recommended core set of clinical outcomes recently suggested by the SRRR the Table group59 and in Europe.60 In addition, taking other time-dependent clinical covariates into account, such as somatosensory deficits and visuospatial neglect,37 the precision of individual time courses can be further improved in particular for those with a low baseline score on FM-UE. In addition, future studies may also investigate the added value of early measured neurophysiological markers such as TMS,38 39 Diffusion Tensor Imaging (DTI),40 41 structural brain scans42 and normalisation of directional symmetry40 as well as intactness of somatosensory systems in EEG43 for improving the precision of individual time courses for the ARAT. Another limitation is that we only have time courses up to 6-month poststroke. While this is often considered a final outcome whereas no changes occur, several studies indicate that the ARAT score of these chronic stroke patients can be further improved with high doses of treatment.31 44 45 Such high dose treatment was not provided to our population and with such treatments or other successful interventions, predicted outcome can deviate from the outcome prediction provided by the current model. A strength of our model is that we predict the exact ARAT score as well as prediction uncertainty. For clinical purposes, this level of detail may not always be needed and classifications of the ARAT scores in four different levels have been proposed46; such classifications could be directly calculated from our outcome prediction. Future impact studies may indicate which level of detail is needed for clinical care and how knowledge of predicted outcome for individual influences impacts the decision making in stroke rehabilitation.

Author affiliations
1Rehabilitation Medicine & Plastic and Reconstructive Surgery, Erasmus MC - University Medical Center Rotterdam, Rotterdam, Netherlands
2Biosistatistics, Erasmus MC - University Medical Center Rotterdam, Rotterdam, Netherlands
3Rehabilitation Research Centre, Reade, Amsterdam, Netherlands
4Rehabilitation Medicine, Erasmus MC - University Medical Center Rotterdam, Rotterdam, Netherlands
5Neuroscience - University Medical Center Rotterdam, Erasmus MC, Rotterdam, Netherlands
6Rijndam Rehabilitation Center, Rotterdam, Netherlands
7Rehabilitation Medicine, Amsterdam UMC - Location UMC, Amsterdam Movement Sciences, Amsterdam, Netherlands

Correction notice This article has been corrected since it appeared Online First. Middle initials have been added to several authors.

Twitter Ruud W Selles @ruudselles

Acknowledgements We thank Aukje Andringa, Mique Saes, Caroline Winters and Sarah Zanbliet for their contribution to data-collection in the 4DEEG and Explore projects.

Contributors RWS, E-RA, GK: Study concept and design, drafting manuscript. E-RA, DR, RWS and RvDv: statistical modeling, interpretation and manuscript revision. RHJN, JSS, ELHW, GMR and CGMM data collection, data management, model interpretation and clinical interpretation and manuscript revision.

Funding This work was supported by ZonMw (grant no. 10-10400-98-008 and grant no. 104003008), Rijndam Rehabilitation Center (5201570) and Amsterdam Movement Sciences (2017). Data collection in the four prospective cohorts was supported by a number of grants, that is, the Netherlands Organization for Health Research and Development (ZonMw grant No. 89000001); the European Research Council (ERC) under the European Union’s Seventh Framework Program (FP/2007-2013/ERC Advanced grant no. 291339), the Dutch Society of Physical Therapy (grant no. 33368) and the Dutch Brain Foundation (Hersenstichting Nederland, grant no. F2011(1)-25).

Competing interests None declared.

Patient consent for publication Not required.

Ethics approval All studies were approved by the medical ethics committees of the VU University Medical Center (NL 47079 029 14) and the Leiden University Medical Center (NL39223.058.12).

Provenance and peer review Not commissioned; externally peer reviewed.

Data availability statement Data are available on reasonable request.

Supplemental material This content has been supplied by the author(s). It has not been vetted by BMJ Publishing Group Limited (BMJ) and may not have been peer-reviewed. Any opinions or recommendations discussed are solely those of the author(s) and are not endorsed by BMJ. BMJ disclaims all liability and responsibility arising from any reliance placed on the content. Where the content includes any translated material, BMJ does not warrant the accuracy and reliability of the translations (including but not limited to local regulations, clinical guidelines, terminology, drug names and drug dosages), and is not responsible for any error and/or omissions arising from translation and adaptation or otherwise.

Open access This is an open access article distributed in accordance with the Creative Commons Attribution 4.0 Unported (CC BY 4.0) license, which permits others to copy, redistribute, remix, transform and build upon this work for any purpose, provided the original work is properly cited, a link to the licence is given, and indication of whether changes were made. See: https://creativecommons.org/licenses/by/4.0/.

ORCID iD Ruud W Selles http://orcid.org/0000-0002-2432-6756

REFERENCES


Rondina JM, Park C-H, Ward NS. Brain regions important for recovery after severe post-stroke upper limb paresis. *J Neurol Neurosurg Psychiatry* 2017;88:737–43.


SUPPLEMENTAL MATERIAL

Computerized patient-specific prediction of the recovery profile of upper limb capacity within stroke services: The next step

Supplementary material

Statistical analysis additional information:

Mixed-effects models account for the serial evaluations of clinical parameters over time. These models can work with unbalanced datasets (i.e., datasets with an unequal number of follow-up measurements between subjects and varying times between repeated measurements of each subject), and they explicitly take into account that measurements from the same patient are more correlated than measurements from different patients.

One important aspect to investigate in mixed-effects model, is the shapes of the patient-specific evolutions over time. It is, therefore, important to postulate a mixed-effects model that appropriately captures nonlinear evolutions.

In order to assume nonlinear evolutions for the ARAT outcome, we used natural cubic splines. Splines are piecewise polynomials where we split the stroke recovery up into intervals (defined by the knots). The knots define the positions at which the piecewise polynomials meet within the recovery profile. A larger number of knots allows for a more flexible model, leading to more flexible predictions. In our application, the knots were placed at the quartile points in the case of 1, 2, and 3 knots and placed manually in the case of 6 knots.
Supplementary Figure 1: Predicted and observed values for 12 randomly-selected patients. The circles represent the observed values, while the lines the predicted values assuming different splines structure. The fitting was better when more than one knot was used, but there are small differences observed between 2, 3, and 6 knots. However, since our main goal was to predict outcome most accurately, we continued with the more flexible model with 6 knots.
Supplementary Figure 2: Cross-validation errors of the 5 different prediction models (see Prediction Model Development section) as a function of the number of measurements that are used. The accuracy was defined as the absolute difference between the predicted ARAT score at 6 months poststroke from the cross-validation and the measured ARAT score at the same time and displayed as median, interquartile range (IQR: Q1 = 25th percentile and Q2 = 75 percentile), lower whisker presented as Q1 - 1.5 * IQR and upper whisker presented as Q2 + 1.5 * IQR. As can be seen, the cross-validation errors for models 1-4 are very similar and only model 5 is clearly less accurate.
Supplementary Figure 3: the same predictions as shown in Figure 2, but with two additional patients. The data of the same patients can be downloaded in the online app to visualized predictions at all time points:

https://emcbiostatistics.shinyapps.io/DynamicPredictionARATapp/
R code:

#########################
# Packages to install and load #
#########################
install.packages("nlme")
install.packages("splines")
install.packages("lattice")
library(nlme)
library(splines)
library(lattice)

############################################
# Functions needed for the dynamic predictions #
############################################

# The following functions were taken from the JMbayes package.
# Small changes have been made.

IndvPred_lme <- function(lmeObject, newdata, timeVar, times = NULL, M = 200L,
interval = c("confidence", "prediction"),
all_times = FALSE,
level = 0.95, return_data = FALSE, seed = 1L) {
  if (!inherits(lmeObject, "lme") && !inherits(lmeObject, "lmeComponents"))
    stop("Use only with 'lme' or 'lmeComponents' objects.
")
  interval <- match.arg(interval)
  if (inherits(lmeObject, "lme")) {
    data <- lmeObject$data
    formYx <- formula(lmeObject)
    mfX <- model.frame(terms(formYx), data = data)
    TermsX <- attr(mfX, "terms")
    formYz <- formula(lmeObject$modelStruct$reStruct[[1]])
    mfZ <- model.frame(terms(formYz), data = data)
    TermsZ <- attr(mfZ, "terms")
    idVar <- names(lmeObject$modelStruct$reStruct)
    betas <- fixef(lmeObject)
    sigma <- lmeObject$sigma
    D <- lapply(pdMatrix(lmeObject$modelStruct$reStruct), "*", sigma^2)[[1]]
    V <- vcov(lmeObject)
    times_orig <- data[[timeVar]]
    times_orig <- times_orig[!is.na(times_orig)]
  } else {
    formYx <- lmeObject$formYx
    TermsX <- lmeObject$TermsX
    formYz <- lmeObject$formYz
    TermsZ <- lmeObject$TermsZ
    idVar <- lmeObject$idVar
    betas <- lmeObject$betas
    sigma <- lmeObject$sigma
    D <- lmeObject$D
    V <- lmeObject$V
    times_orig <- lmeObject$times_orig
  }
  BMJ Publishing Group Limited (BMJ) disclaims all liability and responsibility arising from any reliance placed on this supplemental material which has been supplied by the author(s)

# drop missing values from newdata
all_vars <- unique(c(all.vars(TermsX), all.vars(TermsZ)))
newdata_nomiss <- newdata[complete.cases(newdata[, all_vars]), ]

mfX_new <- model.frame(TermsX, data = newdata_nomiss)
X_new <- model.matrix(formYx, mfX_new)

mfZ_new <- model.frame(TermsZ, data = newdata_nomiss)
Z_new <- model.matrix(formYz, mfZ_new)

na_ind <- attr(mfX_new, "na.action")
y_new <- model.response(mfX_new, "numeric")

if (length(idVar) > 1)
  stop("the current version of the function only works with a single grouping variable.
")
if (is.null(newdata[[idVar]]))
  stop("subject id variable not in newdata.")
id_nomiss <- match(newdata_nomiss[[idVar]], unique(newdata_nomiss[[idVar]]))
n <- length(unique(id_nomiss))

modes <- matrix(0.0, n, ncol(Z_new))
post_vars <- DZtVinv <- vector("list", n)

for (i in seq_len(n)) {
  id_i <- id_nomiss == i
  X_new_id <- X_new[id_i, , drop = FALSE]
  Z_new_id <- Z_new[id_i, , drop = FALSE]
  Vi_inv <- solve(Z_new_id %*% tcrossprod(D, Z_new_id) + sigma^2 * diag(sum(id_i)))
  DZtVinv[i] <- tcrossprod(D, Z_new_id) %*% Vi_inv
  modes[i, ] <- c(DZtVinv[i] %*% (y_new[id_i] - X_new_id %*% betas))

  t1 <- DZtVinv[i] %*% Z_new_id %*% D
  t2 <- DZtVinv[i] %*% X_new_id %*% V %*% crossprod(X_new_id, Vi_inv) %*% Z_new_id %*% D

  post_vars[i] <- D - t1 + t2

}

fitted_y <- c(X_new %*% betas) + rowSums(Z_new * modes[id_nomiss, , drop = FALSE])

if (is.null(times)) || !is.numeric(times)) {
  times <- seq(min(times_orig), max(times_orig), length.out = 100)
}

id <- match(newdata[[idVar]], unique(newdata[[idVar]]))
last_time <- tapply(newdata[[timeVar]], id, max)
times_to_pred <- lapply(last_time, function (t)
  if (all_times) times else times[times > t])
id_pred <- rep(seq_len(n), sapply(times_to_pred, length))

newdata_pred <- right_rows(newdata, newdata[[timeVar]], id, times_to_pred)
newdata_pred[[timeVar]] <- unlist(times_to_pred)

mfX_new_pred <- model.frame(TermsX, data = newdata_pred, na.action = NULL)
X_new_pred <- model.matrix(formYx, mfX_new_pred)

mfZ_new_pred <- model.frame(TermsZ, data = newdata_pred, na.action = NULL)
Z_new_pred <- model.matrix(formYz, mfZ_new_pred)

predicted_y <- c(X_new_pred %*% betas) +
  rowSums(Z_new_pred * modes[id_nomiss, , drop = FALSE])

set.seed(seed)
betas_M <- MASS::mvrnorm(M, betas, V)

modes_fun <- function (betas) {
  t(mapply("%*%", DZtVinv, split(y_new - X_new %*% betas, id_nomiss)))
}

modes_M <- lapply(split(betas_M, row(betas_M)), modes_fun)
matrix_row <- function (m, i) m[, i, drop = FALSE]
modes_M <- lapply(seq_len(n), function (i) t(sapply(modes_M, matrix_row, i = i)))
b_M <- modes_M
for (i in seq_len(n)) {
  b_M[[i]] <- t(apply(modes_M[[i]], 1, MASS::mvrnorm, n = 1, Sigma = post_vars[[i]]))
}
n_pred <- length(predicted_y)
sampled_y <- matrix(0.0, n_pred, M)
for (m in seq_len(M)) {
  betas_m <- betas_M[m, ]
b_m <- t(sapply(b_M, function (x) x[m, ]))
  mean_m <- c(X_new_pred %*% betas_m) +
             rowSums(Z_new_pred * b_m[[id_pred, , drop = FALSE]])
sampled_y[, m] <- if (interval == "confidence") mean_m
  else morm(n_pred, mean_m, lmeObject$sigma)
}
low <- apply(sampled_y, 1, quantile, probs = (1 - level) / 2)
upp <- apply(sampled_y, 1, quantile, probs = 1 - (1 - level) / 2)
rm(list = ".Random.seed", envir = globalenv())
if (!return_data) {
  list(times_to_pred = times_to_pred, predicted_y = predicted_y,
       low = low, upp = upp)
} else {
  out_data <- rbind(newdata, newdata_pred)
  out_data$pred <- c(fitted_y, predicted_y)
  out_data$low <- c(rep(NA, length(fitted_y)), low)
  out_data$upp <- c(rep(NA, length(fitted_y)), upp)
  out_data[order(out_data[[idVar]], out_data[[timeVar]]), ]
}
right_rows <- function(data, times, ids, Q_points) {
  fids <- factor(ids, levels = unique(ids))
  if (!is.list(Q_points))
    Q_points <- split(Q_points, row(Q_points))
  ind <- mapply(findInterval, Q_points, split(times, fids))
  ind[ind < 1] <- 1
  rownams_id <- split(row.names(data), fids)
  ind <- mapply(`[`, rownams_id, split(ind, col(ind)))
  data[c(ind), ]
}

# Function for the dynamic plots #

DynPlots <- function(model.output = model.output, newdata, timeVar,
                      main_title = "Dynamic predictions"){
  # Load individual prediction -------------------------------
  data <- model.output$data
  formYx <- formula(model.output)
yOutcome <- formYx[[2]]
  IndvPrediction95 <- IndvPred_lme(lmeObject = model.output, newdata, timeVar,

times = NULL, M = 500, interval = "prediction", return_data = TRUE)

IndvPrediction68 <- IndvPred_lme(lmeObject = model.output, newdata, timeVar,
times = NULL, M = 500, interval = "prediction", return_data = TRUE, level = 0.68)

pred95 <- IndvPrediction95[which(!is.na(IndvPrediction95$low)),]
pred68 <- IndvPrediction68[which(!is.na(IndvPrediction68$low)),]
nopred <- IndvPrediction95[which(is.na(IndvPrediction95$low)),]

timeVariable <- pred95[[timeVar]]

# Generating plot -----------------------------------------------------
xyplot(pred ~ timeVariable , main = main_title, data = pred95,
type = "l", col = rgb(0.6769,0.4447,0.7114, alpha = 1), lty = c(1, 2, 2), lwd = 3,
ylim = c(0,65), xlim = c(0,200), ylab = list(yOutcome, cex = 1.5), xlab = list(timeVar, cex
= 1.5),
scales = list(x = list(cex = 1.3) , y = list(cex = 1.3)),
panel = function(x, y,  ...){
  panel.xyplot(x, y, ...)
  panel.polygon(c(pred95[,"Days"], rev(pred95[,"Days"])),
c(pred95[,"upp"], rev(pred95[,"low"])),
  border = NA,
col = rgb(0.6769,0.4447,0.7114, alpha = 0.2)
  panel.polygon(c(pred68[,"Days"], rev(pred68[,"Days"])),
c(pred68[,"upp"], rev(pred68[,"low"])),
  border = NA,
col =rgb(0.6769,0.4447,0.7114, alpha = 0.4))
  panel.points(x = nopred[[timeVar]], y = nopred[[yOutcome]], cex = 1.2, pch = 16, col =
  "black";
  panel.lines(x = rep(tail(nopred[[timeVar]], n = 1), 200), y = seq(-100, 100, length =
  200), col = "grey", lty = 3, lwd = 2)
}

#########################################
# Run the mixed-effects model #
#########################################

# ARAT: Action Research Arm test
# SA: shoulder abduction, defined as:
#   - no random movement
#   - random activity palpable, no movement visible
#   - random movement see able but not seeable in total movement range
#   - random movement across total movement range, not possible against resistance
#   - random movement agains resistance , but weaker than ‘healthy’ side
#   - normal strength in comparison with ‘healthy’ side
# FE: finger extension, defined as:
#   - none
#   - partial
#   - full
# Days: The day were the measurements of the ARAT, SA and FE were taken
# Number: is the identification number
# data: include all the variables mentioned above
model_SA_FE <- lme(ARAT ~ SA * ns(Days, knots = c(6, 11, 19, 34, 50, 91)) +
                   FE * ns(Days, knots = c(6, 11, 19, 34, 50, 91)),
                   data = data,
                   random = list(Number =
                        pdDiag(form = ~ ns(Days, knots = c(6, 11, 19, 34, 50, 91)))),
                   na.action = na.exclude,
                   control = lmeControl(maxIter = 1e8, msMaxIter = 1e8))

# Obtain the dynamic predictions
# Define a new patient or select one from the data set
nd <- data[data$Number == 1, ]
# Obtain the plot
DynPlots(model.output = model_SA_FE, newdata = nd, timeVar = "Days",
         main_title = "Patient")
SUPPLEMENTAL MATERIAL

Computerized patient-specific prediction of the recovery profile of upper limb capacity within stroke services: The next step

Supplementary material

Statistical analysis additional information:

Mixed-effects models account for the serial evaluations of clinical parameters over time. These models can work with unbalanced datasets (i.e., datasets with an unequal number of follow-up measurements between subjects and varying times between repeated measurements of each subject), and they explicitly take into account that measurements from the same patient are more correlated than measurements from different patients.

One important aspect to investigate in mixed-effects model, is the shapes of the patient-specific evolutions over time. It is, therefore, important to postulate a mixed-effects model that appropriately captures nonlinear evolutions.

In order to assume nonlinear evolutions for the ARAT outcome, we used natural cubic splines. Splines are piecewise polynomials where we split the stroke recovery up into intervals (defined by the knots). The knots define the positions at which the piecewise polynomials meet within the recovery profile. A larger number of knots allows for a more flexible model, leading to more flexible predictions. In our application, the knots were placed at the quartile points in the case of 1, 2, and 3 knots and placed manually in the case of 6 knots.
Supplementary Figure 1: Predicted and observed values for 12 randomly-selected patients. The circles represent the observed values, while the lines the predicted values assuming different splines structure. The fitting was better when more than one knot was used, but there are small differences observed between 2, 3, and 6 knots. However, since our main goal was to predict outcome most accurately, we continued with the more flexible model with 6 knots.
Supplementary Figure 2: Cross-validation errors of the 5 different prediction models (see Prediction Model Development section) as a function of the number of measurements that are used. The accuracy was defined as the absolute difference between the predicted ARAT score at 6 months poststroke from the cross-validation and the measured ARAT score at the same time and displayed as median, interquartile range (IQR: Q1 = 25th percentile and Q2 = 75 percentile), lower whisker presented as Q1 - 1.5 * IQR and upper whisker presented as Q2 + 1.5 * IQR. As can be seen, the cross-validation errors for models 1-4 are very similar and only model 5 is clearly less accurate.
Supplementary Figure 3: the same predictions as shown in Figure 2, but with two additional patients. The data of the same patients can be downloaded in the online app to visualize predictions at all time points:

https://emcbiostatistics.shinyapps.io/DynamicPredictionARATapp/

![Graphs showing ARAT scores over time for Patient 1, Patient 2, Patient 3, and Patient 4.](image-url)
R code:

# Packages to install and load#
install.packages("nlme")
install.packages("splines")
install.packages("lattice")
library(nlme)
library(splines)
library(lattice)

# Functions needed for the dynamic predictions#

# The following functions were taken from the JMbayes package.
# Small changes have been made.

IndvPred_lme <- function(lmeObject, newdata, timeVar, times = NULL, M = 200L,
interval = c("confidence", "prediction"),
all_times = FALSE,
level = 0.95, return_data = FALSE, seed = 1L) {
  if (!inherits(lmeObject, "lme") && !inherits(lmeObject, "lmeComponents"))
    stop("Use only with 'lme' or 'lmeComponents' objects.\n")
  interval <- match.arg(interval)
  if (inherits(lmeObject, "lme")) {
    data <- lmeObject$data
    formYx <- formula(lmeObject)
    mfX <- model.frame(terms(formYx), data = data)
    TermsX <- attr(mfX, "terms")
    formYz <- formula(lmeObject$modelStruct$reStruct[[1]])
    mfZ <- model.frame(terms(formYz), data = data)
    TermsZ <- attr(mfZ, "terms")
    idVar <- names(lmeObject$modelStruct$reStruct)
    betas <- fixef(lmeObject)
    sigma <- lmeObject$sigma
    D <- lapply(pdMatrix(lmeObject$modelStruct$reStruct), "+", sigma^2)[[1]]
    V <- vcov(lmeObject)
    times_orig <- data[[timeVar]]
    times_orig <- times_orig[!is.na(times_orig)]
  } else {
    formYx <- lmeObject$formYx
    TermsX <- lmeObject$TermsX
    formYz <- lmeObject$formYz
    TermsZ <- lmeObject$TermsZ
    idVar <- lmeObject$idVar
    betas <- lmeObject$betas
    sigma <- lmeObject$sigma
    D <- lmeObject$D
    V <- lmeObject$V
    times_orig <- lmeObject$times_orig
  }

  if (inherits(lmeObject, "lme") & & inherits(lmeObject, "lmeComponents"))
    stop("Use only with 'lme' or 'lmeComponents' objects.\n")
  interval <- match.arg(interval)
  if (inherits(lmeObject, "lme")) {
    data <- lmeObject$data
    formYx <- formula(lmeObject)
    mfX <- model.frame(terms(formYx), data = data)
    TermsX <- attr(mfX, "terms")
    formYz <- formula(lmeObject$modelStruct$reStruct[[1]])
    mfZ <- model.frame(terms(formYz), data = data)
    TermsZ <- attr(mfZ, "terms")
    idVar <- names(lmeObject$modelStruct$reStruct)
    betas <- fixef(lmeObject)
    sigma <- lmeObject$sigma
    D <- lapply(pdMatrix(lmeObject$modelStruct$reStruct), "+", sigma^2)[[1]]
    V <- vcov(lmeObject)
    times_orig <- data[[timeVar]]
    times_orig <- times_orig[!is.na(times_orig)]
  } else {
    formYx <- lmeObject$formYx
    TermsX <- lmeObject$TermsX
    formYz <- lmeObject$formYz
    TermsZ <- lmeObject$TermsZ
    idVar <- lmeObject$idVar
    betas <- lmeObject$betas
    sigma <- lmeObject$sigma
    D <- lmeObject$D
    V <- lmeObject$V
    times_orig <- lmeObject$times_orig
  }
# drop missing values from newdata
all_vars <- unique(c(all.vars(TermsX), all.vars(TermsZ)))
newdata_nomiss <- newdata[complete.cases(newdata[all_vars]), ]
mfX_new <- model.frame(TermsX, data = newdata_nomiss)
X_new <- model.matrix(formYx, mfX_new)
mfZ_new <- model.frame(TermsZ, data = newdata_nomiss)
Z_new <- model.matrix(formYz, mfZ_new)
na_ind <- attr(mfX_new, "na.action")
y_new <- model.response(mfX_new, "numeric")

if (length(idVar) > 1)
  stop("the current version of the function only works with a single grouping variable.
"")
if (is.null(newdata[[idVar]]))
  stop("subject id variable not in newdata.
")
if (is.null(newdata[[idVar]]))
  stop("subject id variable not in newdata.
")
if (is.null(newdata[[idVar]]))
  stop("subject id variable not in newdata.
")

id_nomiss <- match(newdata_nomiss[[idVar]], unique(newdata_nomiss[[idVar]]))
n <- length(unique(id_nomiss))

modes <- matrix(0.0, n, ncol(Z_new))
post_vars <- DZtVinv <- vector("list", n)
for (i in seq_len(n)) {
  id_i <- id_nomiss == i
  X_new_id <- X_new[id_i, , drop = FALSE]
  Z_new_id <- Z_new[id_i, , drop = FALSE]
  Vi_inv <- solve(Z_new_id %*% tcrossprod(D, Z_new_id) + sigma^2 * diag(sum(id_i)))
  DZtVinv[[i]] <- tcrossprod(D, Z_new_id) %*% Vi_inv
  modes[i, ] <- c(DZtVinv[[i]] %*% (y_new[id_i] - X_new_id %*% betas))
  t1 <- DZtVinv[[i]] %*% Z_new_id %*% D
  t2 <- DZtVinv[[i]] %*% X_new_id %*% V %*% crossprod(X_new_id, Vi_inv) %*% Z_new_id %*% D
  post_vars[[i]] <- D - t1 + t2
}

fitted_y <- c(X_new %*% betas) + rowSums(Z_new * modes[id_nomiss, , drop = FALSE])

if (is.null(times)) || is.numeric(times)) {
  times <- seq(min(times_orig), max(times_orig), length.out = 100)
}

id <- match(newdata[[idVar]], unique(newdata[[idVar]]))
last_time <- tapply(newdata[[timeVar]], id, max)
times_to_pred <- lapply(last_time, function (t)
  if (all_times) times else times[times > t])

newdata_pred <- right_rows(newdata, newdata[[timeVar]], id, times_to_pred)
X_new_pred <- model.frame(TermsX, data = newdata_pred, na.action = NULL)
Z_new_pred <- model.frame(TermsZ, data = newdata_pred, na.action = NULL)
predicted_y <- c(X_new_pred %*% betas) +
  rowSums(Z_new_pred * modes[id_nomiss, , drop = FALSE])

set.seed(seed)
betas_M <- MASS::mvrnorm(M, betas, V)

modes_fun <- function (betas) {
  t(mapply("%*%", DZtVinv, split(y_new - X_new %*% betas, id_nomiss)))
}

modes_M <- lapply(split(betas_M, row(betas_M)), modes_fun)
matrix_row <- function (m, i) m[i, , drop = FALSE]
modes_M <- lapply(seq_len(n), function (i) t(sapply(modes_M, matrix_row, i = i)))
b_M <- modes_M
for (i in seq_len(n)) {
b_M[i] <- t(apply(modes_M[i], 1, MASS::mvrnorm, n = 1, Sigma = post_vars[i]))
}

n_pred <- length(predicted_y)
sampled_y <- matrix(0.0, n_pred, M)
for (m in seq_len(M)) {
betas_m <- betas_M[m, ]
b_m <- t(sapply(b_M, function (x) x[m, ]))
mean_m <- c(X_new_pred %*% betas_m) +
rowSums(Z_new_pred * b_m[id_pred, , drop = FALSE])
sampled_y[, m] <- if (interval == "confidence") mean_m
else morm(n_pred, mean_m, lmeObject$sigma)
}

low <- apply(sampled_y, 1, quantile, probs = (1 - level) / 2)
upp <- apply(sampled_y, 1, quantile, probs = 1 - (1 - level) / 2)

if (!return_data) {
list(times_to_pred = times_to_pred, predicted_y = predicted_y,
low = low, upp = upp)
} else {
out_data <- rbind(newdata, newdata_pred)
out_data$pred <- c(fitted_y, predicted_y)
out_data$low <- c(rep(NA, length(fitted_y)), low)
out_data$upp <- c(rep(NA, length(fitted_y)), upp)
out_data[order(out_data[[idVar]], out_data[[timeVar]]),]
}

right_rows <- function(data, times, ids, Q_points) {
fids <- factor(ids, levels = unique(ids))
if (!is.list(Q_points))
Q_points <- split(Q_points, row(Q_points))
ind <- mapply(findInterval, Q_points, split(times, fids))
ind[ind < 1] <- 1
rownams_id <- split(row.names(data), fids)
ind <- mapply("[", rownams_id, split(ind, col(ind))]
data[c(ind),]
}

# Function for the dynamic plots
DynPlots <- function(model.output = model.output, newdata, timeVar,
main_title = "Dynamic predictions"){

# Load individual prediction
data <- model.output$data
formYx <- formula(model.output)
yOutcome <- formYx[2]
IndvPrediction95 <- IndvPred_lme(lmeObject = model.output, newdata, timeVar,
times = NULL, M = 500, interval = "prediction", return_data = TRUE)

IndvPrediction68 <- IndvPred_lme(lmeObject = model.output, newdata, timeVar, 
times = NULL, M = 500, interval = "prediction", return_data = TRUE, level = 0.68)
pred95 <- IndvPrediction95[which(!is.na(IndvPrediction95$low)),]
pred68 <- IndvPrediction68[which(!is.na(IndvPrediction68$low)),]
nopred <- IndvPrediction95[which(is.na(IndvPrediction95$low)),]

timeVariable <- pred95[,timeVar]

# Generating plot -----------------------------------------------------
xyplot(pred ~ timeVariable , main = main_title, data = pred95, 
type = "l", col = rgb(0.6769,0.4447,0.7114, alpha = 1), lty = c(1, 2, 2), lwd = 3,
ylim = c(0,65), xlim = c(0,200), ylab = list(yOutcome, cex = 1.5), xlab = list(timeVar, cex = 1.5),
scales = list(x = list(cex = 1.3) , y = list(cex = 1.3)), 
panel = function(x, y, ...) {
  panel.xyplot(x, y, ...)
  panel.polygon(c(pred95[,"Days"], rev(pred95[,"Days"])), 
c(pred95[,"upp"], rev(pred95[,"low"])),
  border = NA,
  col = rgb(0.6769,0.4447,0.7114, alpha = 0.2))
  panel.polygon(c(pred68[,"Days"], rev(pred68[,"Days"])), 
c(pred68[,"upp"], rev(pred68[,"low"])),
  border = NA,
  col =rgb(0.6769,0.4447,0.7114, alpha = 0.4))
  panel.points(x = nopred[,timeVar], y = nopred[yOutcome], cex = 1.2, pch = 16, col = "black"),
  panel.lines(x = rep(tail(nopred[[timeVar]], n = 1), 200), y = seq(-100, 100, length = 200), col = "grey", lty = 3, lwd = 2)
})

###########################
# Run the mixed-effects model #
###########################

# ARAT: Action Research Arm test
# SA: shoulder abduction, defined as:
# - no random movement
# - random activity palpable, no movement visible
# - random movement see able but not seeable in total movement range
# - random movement across total movement range, not possible against resistance
# - random movement agains resistance , but weaker than 'healthy' side
# - normal strength in comparison with 'healthy' side
# FE: finger extension, defined as:
# - none
# - partial
# - full
# Days: The day were the measurements of the ARAT, SA and FE were taken
# Number: is the identification number
# data: include all the variables mentioned above
model_SA_FE <- lme(ARAT ~ SA * ns(Days, knots = c(6, 11, 19, 34, 50, 91)) + 
                    FE * ns(Days, knots = c(6, 11, 19, 34, 50, 91)), 
                    data = data, 
                    random = list(Number = 
                                    pdDiag(form = ~ ns(Days, knots = c(6, 11, 19, 34, 50, 
                                                      9)), 
                                    na.action = na.exclude, 
                                    control = lmeControl(maxIter = 1e8, msMaxIter = 1e8))

# Obtain the dynamic predictions #
# Define a new patient or select one from the data set
nd <- data[data$Number == 1, ]
# Obtain the plot
DynPlots(model.output = model_SA_FE, newdata = nd, 
          timeVar = "Days", 
          main_title = "Patient")