Journal of Clinical Epidemiology 175 (2024) 111531

Journal of Clinical Epidemiology

ORIGINAL RESEARCH

Implementation of a dynamic model updating pipeline provides a systematic process for maintaining performance of prediction models

Kamaryn T. Tanner[a,*], Karla Diaz-Ordaz[b], Ruth H. Keogh[a]

[a]*Dept of Medical Statistics, London School of Hygiene and Tropical Medicine, London, WC1E 7HT, UK*
[b]*Dept of Statistical Science, University College London, London, WC1E 6BT, UK*

Accepted 4 September 2024; Published online 12 September 2024

Abstract

Objectives: We describe the steps for implementing a dynamic updating pipeline for clinical prediction models and illustrate the proposed methods in an application of 5-year survival prediction in cystic fibrosis.

Study Design and Setting: Dynamic model updating refers to the process of repeated updating of a clinical prediction model with new information to counter performance degradation. We describe 2 types of updating pipeline: "proactive updating" where candidate model updates are tested any time new data are available, and "reactive updating" where updates are only made when performance of the current model declines or the model structure changes. Methods for selecting the best candidate updating model are based on measures of predictive performance under the 2 pipelines. The methods are illustrated in our motivating example of a 5-year survival prediction model in cystic fibrosis. Over a dynamic updating period of 10 years, we report the updating decisions made and the performance of the prediction models selected under each pipeline.

Results: Both the proactive and reactive updating pipelines produced survival prediction models that overall had better performance in terms of calibration and discrimination than a model that was not updated. Further, use of the dynamic updating pipelines ensured that the prediction model's performance was consistently and frequently reviewed in new data.

Conclusion: Implementing a dynamic updating pipeline will help guard against model performance degradation while ensuring that the updating process is principled and data-driven.

Keywords: Clinical prediction models; Dynamic updating; Model updating; Survival analysis; Cystic fibrosis

1. Introduction

Clinical prediction models provide patients and clinicians with an estimated risk of a health outcome based on individual characteristics [1]. The output may be used to identify higher-risk individuals and inform medical decision making. As external factors change, the performance of prediction models may decline over time [1,2]. This is often manifested as calibration drift—the deterioration of the model's calibration over time [3,4]. The introduction of new treatments or newly measured risk factors may also render a prediction model out of date.

Numerous methods have been proposed and compared for one-time updating of prediction models [2,5—10], which include recalibration and Bayesian updating, but no single technique has been found to be best across settings with different updating sample sizes and population shifts. Regardless of the technique selected, after updating, model performance may again deteriorate as the disease, treatments and/or population mix changes. Dynamic updating is a strategy for combating this by repeatedly updating the prediction model, making use of new data and information. Although dynamic updating has been shown to be promising [2,9,11], there is a lack of guidance on how to implement a dynamic updating pipeline. A pipeline provides a systematic process for determining whether to update a prediction model based on performance metrics, significance of changes and availability of new predictors.

Funding: Ruth H Keogh was funded by UK Research and Innovation (Future Leaders Fellowship MR/S017968/1, MR/X015017/1). Karla Diaz-Ordaz is funded by a Royal Society-Wellcome Trust Sir Henry Dale fellowship, grant number 218554/Z/19/Z.

* Corresponding author. London School of Hygiene & Tropical Medicine Faculty of Epidemiology and Population Health, Medical Statistics, Keppel Street, London WC1E 7HT, UK.

E-mail addresses: kamaryn.tanner1@lshtm.ac.uk; kamaryn.tanner@columbia.edu (K.T. Tanner).

https://doi.org/10.1016/j.jclinepi.2024.111531

What is new?

Key findings

- Dynamic updating of clinical prediction models enables predictive performance to be maintained over time in the presence of changes in population mix, environmental factors, and treatment options.

What this adds to what is known?

- We describe 2 types of dynamic updating pipelines based on proactive and reactive updating, and provide guidance on their implementation.
- Both pipelines are illustrated, each with different implementation choices, for prediction of 5-year survival in cystic fibrosis over 10 periods and we show that they can guard against performance degradation.

What is the implication and what should change now?

- We encourage the use of a dynamic updating pipeline instead of ad hoc updating as it ensures continual performance review and a data-driven approach to prediction model updating.

In this study, we describe and illustrate methodological pipelines for dynamic evaluation and updating of a clinical prediction model. In Section 2, we draw from both the machine learning and model updating literature to introduce 2 updating pipelines and enumerate the steps and choices to be made when implementing a dynamic updating process. Section 3 provides background on the motivating example, survival prediction for people with cystic fibrosis (CF), a genetic life-shortening disease. The pipelines are illustrated in Section 4 using data from the UK CF Registry [12]. After fitting an initial model for 5-year survival prediction using data on characteristics recorded in 2005 and follow-up through 2010, we retrospectively track the dynamic updating steps that would have been implemented under our 2 pipelines as new data became available annually over the period 2011-2021. Results are reported in Section 4 and we conclude with a Discussion in Section 5.

2. Materials and methods

2.1. Overview of dynamic model updating

The focus is on the setting in which a prediction model is developed for use within a population which is evolving over time, as is the case for prediction models developed within a specific patient population. We consider an initial prediction model f_0, developed using development dataset P_0 during period $u = 0$. We plan to repeatedly update f_0 as the population in which it was developed evolves. Let P_0 contain data from $(t\ _1, t_0]$. After time t_0, new data are acquired during period $u = 1$ to form dataset P_1 containing data from $(t_0, t_1]$. P_1 may contain a single new data point, several years of data, or anything in between. The length of any period u will depend on perceived need for an update, resource availability, procedures for retrieving new data and the ability of users of the prediction model to adapt to changes. Let f_u* denote the "current" prediction model at the start of period $u + 1$. New data P_{u+1} may be used to update f_u^* via one of $k = 1...K$ updating methods, (eg, refitting, recalibration, Bayesian updating), resulting in prediction model f_{u+1}^k. We let method $k = 1$ denote "no update," so that one candidate model is the current model. The K candidate models are evaluated based on selected performance metrics (such as discrimination and calibration measures), and we let $m_v\left(f_{u+1}^k, P_{u+1}\right)$ be the value of the vth performance metric for model f_{u+1}^k evaluated on data P_{u+1}. Performance metrics and methods for their comparison are discussed further below.

2.2. Proactive Vs reactive updating

In a *proactive* updating strategy, updates are applied in each period u and the best performing model from a set of updating methods becomes the current prediction model. This strategy may be appropriate when frequent changes to the model can be tolerated and when incorporation of new data is a priority. Under this strategy, no check is made to see if the current model's performance has deteriorated. In period $u + 1$, new data are acquired, P_{u+1}, and the current prediction model, f_u^*, is updated using each of the K updating techniques to obtain candidate updating models, $f_{u+1}^1, f_{u+1}^2, \ldots, f_{u+1}^K$. The performance of each candidate model is evaluated in the new data using each of the V evaluation metrics. Because models are both updated with and evaluated in P_{u+1}, optimism should be accounted for using test-train split, bootstrapping or cross-validation [13,14] (see Table 1). The model identified as the best performer is selected to become the new current model and predictions are made using this model until the next update (See Fig 1, left panel).

In contrast, *reactive* updating begins by determining in each period u whether the current model requires an update based on some criteria. The performance of the current model f_u^* in the prior period's data P_u is compared to its performance in the new data P_{u+1}, based on metrics $m_v(f_u$*, $P_u)$ and $m_v(f_u$*, $P_{u+1})$, $v = 1,\ldots,V$. The metrics are compared to a set of predefined performance thresholds or "triggers", which specify how much the performance of the current model would have to deteriorate in the new data to warrant an update. If no triggers are flagged, the current model f_u^* is retained for another period; if one or more triggers are flagged, the current model is updated using the

Table 1. Considerations for proactive and reactive dynamic updating pipelines

Category	Description
Updating frequency	Updating frequency will depend on several factors: the frequency with which new data are acquired; the updating method chosen (e.g., a full refit requires more data than a Bayesian update); downstream users' tolerance to adjust to updates; and the extent and speed of changes in the environment. Proactive: More frequent updates with smaller incremental improvements. Updating may occur even if model is performing adequately. Reactive: Less frequent updates may have greater differences from previous model. Updated only when performance of current model falls below threshold. Performance of change detection methods may vary by circumstance [15].
Mix of old and new data	Models may be updated using newly acquired data only or a combination of old and new data, providing a larger dataset. Including older data will ''smooth'' the updates but may slow the incorporation of new trends [11]. Weights may be used to give more importance to recent observations [16]. Proactive: Including more old data will yield update candidates that are more similar to the current model. Reactive: Including more old data reduces the chances of the current model's performance falling below updating thresholds.
Methods for model updating	Dynamic model updating consists of repeated updating steps, which may include full refitting, recalibration, and Bayesian updating methods. Descriptions and comparisons of updating techniques can be found in [2,8] for binary outcome settings and in [9] for the time-to-event case. Proactive and Reactive: Both pipelines accommodate multiple updating methods.
Performance metrics and internal validation	Prediction models can be evaluated in terms of their calibration (e.g., calibration plot, calibration intercept and slope), discrimination (e.g., area under the receiver operating characteristic curve, C-index), and overall performance (e.g., R^2, Brier score) [1]. Because some performance evaluations are made in the same data used to update the model, internal optimism must be accounted for. Internal validation methods include a random-split sample approach, bootstrapping, cross-validation [13,14]. Because the split sample approach may be inefficient, the latter 2 are preferred but the choice needs to consider their computational complexity and compatibility with other modeling choices, such as multiple imputation [17]. Proactive and Reactive: Both allow for choice of appropriate metrics and validation methods.
Performance metrics comparison	Comparisons may be based on relative changes (e.g., an x% decrease in C-index) or deviations from target values (e.g., difference of calibration intercept from 0.0). Metrics points estimates can be directly compared, or hypothesis testing can be used to assess the significance of any differences, e.g., using a t-test applied to metrics estimated across bootstrap replicates. Multiple testing may be required.
Post—model selection inference	If methods include model selection (e.g., variable selection), then the model's performance may be optimistic in that data. Appropriate postselection inference should be used. Ignoring this would impact any decisions to update based on hypothesis testing. Proactive and Reactive: Same considerations apply for both pipelines
Non-inferiority margins and triggers	Non-inferiority margins to determine acceptable updates must be pre-defined (see Section 2.3). A data-driven option is to define the margin as some number of SDs from the current model's performance metric. Proactive: With wider non-inferiority margins, more update candidates will be deemed acceptable. To require updates to be superior on all performance metrics, margins can be set to zero. Reactive: Thresholds can be based on comparison of performance metrics in the old and new data, e.g. percentage change. Triggers may be informed by external information, such as a new predictor or a known external shock to the system.
Computation time	Computation time will increase with larger sample sizes and more updated candidates and modeling choices (e.g., Bayesian methods, multiple imputation, cross-validation). Proactive: Greater, all models are implemented and evaluated in the updating period. Reactive: Lower, e.g. no updating is required in an unchanging environment.

new data, P_{u+1} (See Fig 1, right panel). As in the proactive strategy, optimism must be accounted for (see Table 1).

Implementing a proactive or reactive dynamic updating strategy requires a number of methodological choices to be made. Some key decisions and a comparison of the pipelines are presented in Table 1. In Table 2, we consider several scenarios of change and how each pipeline can adapt to them.

2.3. *Selecting a candidate model update using performance metrics*

Performance of clinical prediction models is typically evaluated across multiple metrics including calibration, discrimination, and overall prediction error. Under a dynamic updating strategy, we may find that a candidate model is superior on one metric but inferior on another.

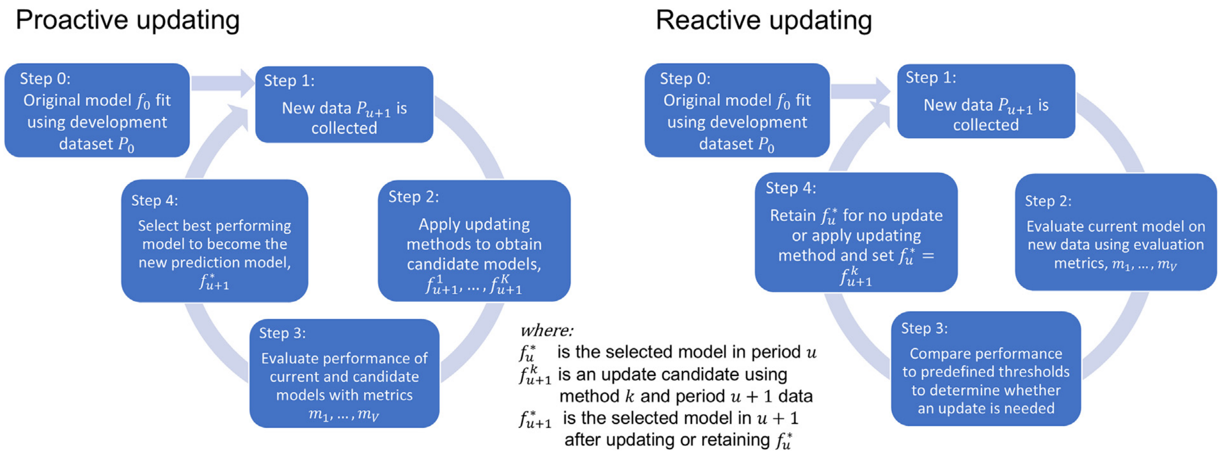
Proactive updating
Step 0: Original model f_0 fit using development dataset P_0
Step 1: New data P_{u+1} is collected
Step 2: Apply updating methods to obtain candidate models, $f_{u+1}^1, \dots, f_{u+1}^K$
Step 3: Evaluate performance of current and candidate models with metrics $m_1, \dots, m_V$
Step 4: Select best performing model to become the new prediction model, f_{u+1}^*
Reactive updating
Step 0: Original model f_0 fit using development dataset P_0
Step 1: New data P_{u+1} is collected
Step 2: Evaluate current model on new data using evaluation metrics, $m_1, \dots, m_V$
Step 3: Compare performance to predefined thresholds to determine whether an update is needed
Step 4: Retain f_u^* for no update or apply updating method and set $f_u^* = f_{u+1}^k$
where:
f_u^* is the selected model in period u
f_{u+1}^k is an update candidate using method k and period $u+1$ data
f_{u+1}^* is the selected model in $u+1$ after updating or retaining f_u^*

Figure 1. Summary of proactive and reactive updating strategies. Proactive updating begins by updating the current prediction model, while reactive updating begins by determining whether the current model requires an update. In proactive updating, for each performance metric and each updating method, the performance of the current model in the new data is compared to the performance of the updating candidates in the new data, that is, m_v (f_u^*, P_{u+1}) is compared to m_v (f_{u+1}^k, P_{u+1}). In reactive updating, the performance of the current model is compared in last period's data and in the new data, that is, m_v (f_u^*, P_u) is compared to m_v (f_u^*, P_{u+1}). (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

Choosing based on one criterion alone may lead to substantial performance degradation in other criteria over time.

To facilitate comparison such that improvement in one metric does not lead to unacceptable worsening in others, we adopt acceptability graphs as described in [21]. An update from f_u^* to f_{u+1} is defined as "acceptable" if f_{u+1} is both noninferior on all performance metrics to f_u^* and superior on at least 1 metric. Noninferiority margins are specified for each metric. A sample acceptability graph for 2 metrics (C-index and calibration intercept) is shown in Figure 2 where the noninferiority margin for both metrics is 0.01. The shaded region represents the values of the metrics that are superior to Model 1 on at least one dimension and noninferior on the other. Model A is not an acceptable update because the calibration intercept is not noninferior; B is not acceptable because it is not superior on either dimension; C is superior to Model 1 on calibration intercept and noninferior on C-index, and is therefore an acceptable update. A noninferiority margin of 0 means that only superior models will be accepted.

3. Application to the UK Cystic Fibrosis Registry dataset

3.1. Overview

There is a rich literature on survival prediction models in CF with models fit to data from the US, UK, Canada and Europe using statistical techniques including Cox regression, landmarking, and machine learning [22–26]. At the same time, new treatments and improved standards of care have led to dramatic increases in expected survival. In the last decade, the median predicted survival age increased by over 10 years to 56 years (UK Cystic Fibrosis [27]). With the increasing use of disease-modifying therapies, continued improvements in survival are expected and existing prediction models will require updating [18].

We illustrate proactive and reactive updating pipelines for a model predicting 5-year survival in CF using data from the UK CF Registry, which collects data at annual reviews. This model's aim is prediction not explanation. To showcase the diversity possible in updating pipelines, our proactive updating approach uses Bayesian candidate update models; in the reactive updating pipeline we obtain model updates, where needed, by considering recalibration and refitting to allow for both changes in the baseline hazard and rapid updating of the coefficients. Figure 3 shows process flows for both pipelines.

3.2. Study population & data preparation

The study population was comprised of all individuals in the UK CF Registry with an annual review between January 1, 2005, and December 31, 2016, who were aged 5.5 years or more at the time of the annual review. This resulted in 74,338 annual review records for 9933 unique individuals. Of these, 2058 had the composite event of death or transplant up to the end of follow-up at December 31, 2021.

We created the initial model development dataset, P_0, using predictor information from annual reviews in calendar year 2005 and death/transplant information up to December 31, 2010. For performance evaluation and prediction model updating, we created 11 additional datasets, P_1,...P_{11}, using data on predictors from years 2006, 2007,...,2016 with follow-up respectively through year end 2011, 2012,...,2021. The initial model f_0 was fit using P_0 and similar predictors to the model of [23] (see

Table 2. Scenarios of change and their implications for the proactive and reactive updating pipelines

Scenario	Description
External shock. Eg, the introduction of a new therapy that dramatically improves survival outcomes such as cystic fibrosis transmembrane conductance regulator modulator therapies in CF [18].	The new treatment can be included in both pipelines using either refitting or Bayesian updating. A key limitation is the amount of new data available on outcomes for people receiving it. Information from clinical trials could be incorporated into the prior in Bayesian updating. Refitting may not be possible early in the treatment rollout if few individuals receiving the treatment have had the event [9]. Proactive: Data related to the new treatment are naturally incorporated with each update. Analysts may choose to use only/mostly new data for the updates to speed adaptation of the prediction model. Reactive: Slower to respond because performance of the current model may not decrease during rollout of the new treatment due to lack of data for the treated. A trigger could be applied to force an update based on external information.
Changes in the baseline risk, e.g., as in the EuroScore I model [2], or in patient mix.	Changes in baseline risk could manifest themselves as calibration drift, which can be addressed in both pipelines without a full refit using either intercept recalibration or Bayesian updating. When calibration is the primary consideration, analysts may wish to incorporate visual assessment of calibration plots for a more complete assessment of calibration. If calibration drift is expected from external knowledge, analysts may require updates to have superior calibration and/or lower calibration trigger thresholds. Case mix changes, which can occur over time or when a model is applied to a different population, may require updating the predictor coefficients. In both pipelines, Bayesian updating allows knowledge from the previous model to be applied to new case mix data. Proactive: Baseline risk changes likely occur gradually over time. Proactive updating will make gradual changes to the model at each update, with the goal of preventing calibration from drifting too far. Reactive: Speed of adaptation will depend both on threshold size and on speed of change in the environment. Based on inspection of a calibration plot, an update trigger could be set.
Data quality increases or decreases, e.g., due to changes in data collection techniques or changes in the extent of missingness over time.	Changes in data quality, e.g., data definitions or accuracy of certain measurements, could affect performance, resulting in updates under both pipelines. Missingness in predictors requires consideration in terms of how missing data are treated both at the model fitting stage and at the evaluation stage. Techniques such as multiple imputation and Bayesian methods to address missing data have been developed [17,19], but increase computational complexity. Implications for updating pipelines will depend on whether data quality changes are temporary or permanent, and whether they are apparent from external knowledge or more subtle. For temporary quality issues, analysts may consider increasing the ratio of old to new data used for updating. Proactive: Use of only/mostly new data for the updates could speed adaptation of the prediction model to a known new level of data quality. Reactive: Known changes in quality could be used to define update triggers.
Changes are burdensome for users. Eg, models used to identify high-risk groups of people (as with COVID-19 pandemic shielding lists) or primary care models requiring change across an entire health system.	Opting for recalibration will maintain the rank order of inividuals in terms of risk while allowing for calibration to be adjusted. Bayesian methods with high values of the forgetting factor will have regression parameters constrained to be more similar to the current model [9,20]. Proactive: May not be the best choice as it results in more frequent updates. Non-inferiority margins could be set to zero to require the updates to be superior on all metrics. Reactive: Thresholds can be defined so that the model is only updated when performance change would have important clinical implications.

CF, cystic fibrosis.

Appendix A). We allowed for new predictors to be added to the model over the 10-year updating process, as described in Appendix A.

3.3. Implementation of a proactive updating pipeline with Bayesian dynamic updating

A Bayesian dynamic updating approach combines information learned in the past (the prior distribution) with new information in the updating data (the likelihood). Here, the main model is a Weibull hazard model. Because there is missing predictor data, we include models for those variables in a fully Bayesian approach. Details of the Bayesian specification are in Appendix B. For periods $u > 0$, priors were derived from parameter estimates from the previous period, scaled by a forgetting factor, ξ, which specifies the uncertainty in the prior [20,28]. We consider Bayesian updates with 4 different values, $\xi = (0.7, 0.8, 0.9, 1.0)$. Two additional proactive pipelines using refitting and recalibration instead of Bayesian updating are described in Appendix D.

The performance of the previous period's selected model in the new data is compared to the performance of the candidate update models in the new data. For these comparisons, we randomly divided each updating dataset P_{u+1} into training P_{u+1}^{train} (75%) and holdout P_{u+1}^{hold} (25%) datasets, though cross-validation or bootstrapping could also have been chosen. The candidate update models f_{u+1}^{k} ($k = 1,\ldots,5$) are fit using P_{u+1}^{train} and all performance metrics are evaluated in P_{u+1}^{hold}. We calculated 4 performance metrics

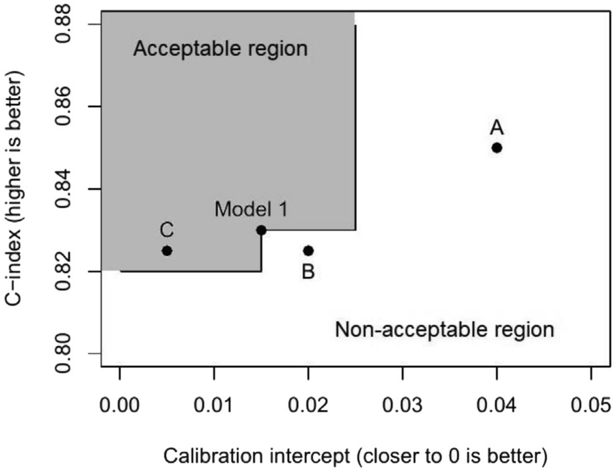

Figure 2. A sample acceptability graph for 2 performance measures. For a noninferiority margin of 0.1 for both C-index and calibration intercept, the shaded region indicates the values of those measures representing acceptable updates from Model 1. Of updated candidates A, B and C, only C is considered an acceptable update.

for 5-year survival prediction for each candidate update: Brier score, C-index, calibration intercept, and calibration slope.

To compare the performance between candidates, we used a hypothesis testing framework with a one-sided paired Wilcoxon signed rank test (significance level of 5%) and each performance metric was computed on 250 bootstrap samples of P_{u+1}^{hold}. To control for multiple testing across metrics and update candidates, we implemented Holm's procedure [29]. We used a data-driven approach to define noninferiority margins and set them at 1 SD from the value of the current model for C-index and calibration intercept/slope. Because Brier score values were close to zero, we set this margin at 0.005. If no candidate updates were superior on at least one metric and noninferior on all others, the current model was retained. If more than one candidate update was acceptable, we chose the one with best calibration intercept.

3.4. *Implementation of a reactive updating pipeline with recalibration and refitting*

In the reactive strategy, the candidate model updating methods were recalibration and refitting. After new data P_{u+1} was acquired, training P_{u+1}^{train} and holdout datasets P_{u+1}^{hold} were created by random 75%/25% split. If an update was required based on one or more triggers, f_u* was updated to f_{u+1}^* using P_{u+1}^{train} and performance of the updated model was computed in P_{u+1}^{hold}. If no update was triggered, f_u*_{+1} was set to f_u*.

We prespecified tolerance levels on absolute or relative performance of the current model in the new data that would trigger an update. Because the C-index had low variability and we had high tolerance for frequent updates, we set the C-index trigger at a drop of 2% or more. As Brier score values were close to zero, we set the threshold to be an increase of 0.01 or more in the new data, compared to the previous data. Because the exp(calibration intercept) estimates the ratio of observed to expected events, the trigger can be defined in terms of permitted deviation in this ratio. Here, if the calibration intercept or slope measured in the new data was more than 0.1 away from the target values of 0 and 1, respectively, an update was triggered. Finally, if a new predictor was introduced, this automatically triggered an update.

These trigger criteria were varied in 3 sensitivity analyses described in Appendix E.

Appendix C describes handling of missing data and Appendix F lists software used.

4. Results

4.1. *Proactive updating pipeline with Bayesian dynamic updating*

In each period, performance of the Bayesian updated model was compared to the performance of the current model. A summary of the results from the updating pipeline is shown in Table 3. In period 1, the initial model was retained, but in every subsequent period one of the candidate update models was chosen. The forgetting factor of the selected model (ξ) varied by period. Performance of the initial model representing no updating is provided for comparison. The C-index of the updated models was better than the initial model in all but 1 period and was better than the current model in all but 2 periods. Although the magnitude of the difference in any single period was small, less than 0.02, over the entire updating cycle the C-index of the final updated model was 0.08 greater than the initial model (See Figure 4). Because a calibration intercept closer to 0 is preferred, we compared the absolute values of the calibration intercept. The calibration intercept of the updated model was better than the current model by 0.10 or more in three updating periods and was better than the initial model in 8 periods.

4.2. *Reactive updating pipeline with recalibration and refitting*

After the initial model was fit, triggers were evaluated in each of the 10 annual updating datasets. Table 4 shows the triggers and updating results for each period for the current and updated models. In periods 1 and 10, there was a trigger for calibration intercept and the model was recalibrated. In periods 3, 8, and 9, new predictors were added that required a full refit and in period 2, C-index and calibration triggers fired. None of the triggers indicated that an update was needed for periods 4 to 7, so the current model was retained. The C-index of the updated model was equal to or better than both the current model and the initial model in each period (See Fig 5). The calibration

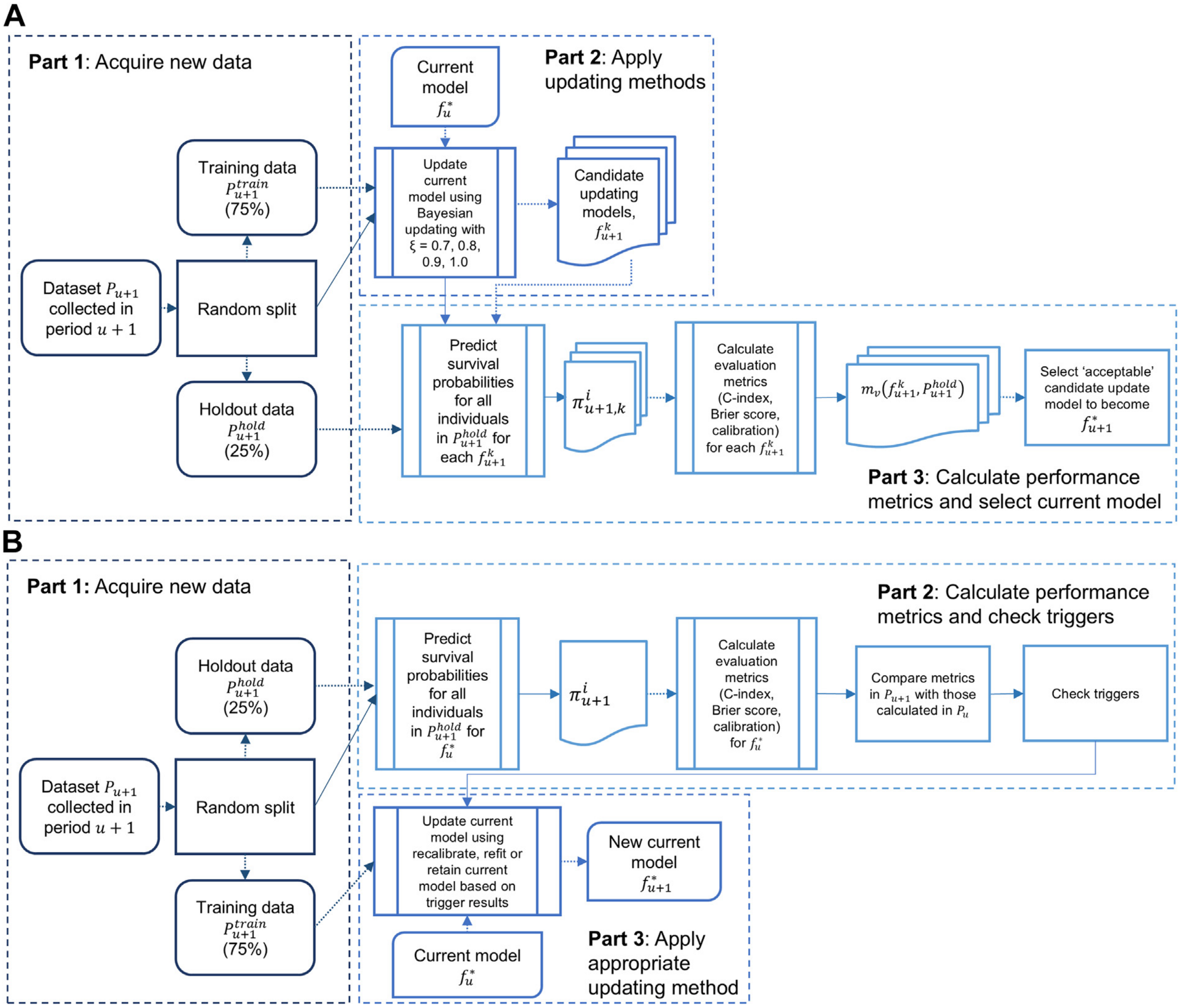
A
Part 1: Acquire new data
Dataset P_{u+1} collected in period $u+1$
Random split
Training data P_{u+1}^{train} (75%)
Holdout data P_{u+1}^{hold} (25%)
Current model f_u^*
Part 2: Apply updating methods
Update current model using Bayesian updating with ξ = 0.7, 0.8, 0.9, 1.0
Candidate updating models, f_{u+1}^k
Predict survival probabilities for all individuals in P_{u+1}^{hold} for each f_{u+1}^k
$\pi_{u+1,k}^i$
Calculate evaluation metrics (C-index, Brier score, calibration) for each f_{u+1}^k
$m_v(f_{u+1}^k, P_{u+1}^{hold})$
Select 'acceptable' candidate update model to become f_{u+1}^*
Part 3: Calculate performance metrics and select current model
B
Part 1: Acquire new data
Holdout data P_{u+1}^{hold} (25%)
Dataset P_{u+1} collected in period $u+1$
Random split
Training data P_{u+1}^{train} (75%)
Part 2: Calculate performance metrics and check triggers
Predict survival probabilities for all individuals in P_{u+1}^{hold} for f_u^*
π_{u+1}^i
Calculate evaluation metrics (C-index, Brier score, calibration) for f_u^*
Compare metrics in P_{u+1} with those calculated in P_u
Check triggers
Update current model using recalibrate, refit or retain current model based on trigger results
New current model f_{u+1}^*
Current model f_u^*
Part 3: Apply appropriate updating method

Figure 3. Process flow for a proactive updating pipeline (a) and a reactive updating pipeline (b) implemented with training/holdout data split to address internal optimism. Dotted arrows indicate inputs to the next step; solid arrows indicate the next step in the pipeline. The proactive pipeline used Bayesian model updating and the reactive pipeline used refitting and recalibration. Chosen evaluation metrics were the same for both pipelines. We illustrate the random-split sample approach we used for validation but cross-validation or bootstrapping may also be used (See ''Performance metrics and internal validation'' in Table 1). (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

intercept of the updated model was better than the current model in 8/10 periods and better than or equal to the initial model in 9/10 periods. In the second period, the model was refit in response to calibration slope and C-index triggers but the calibration intercept of the resulting updated model was worse (0.13) than that of the current model (0.9). This update was accepted nonetheless because it satisfied the noninferiority conditions. In period 9, an update was accepted despite having an inferior calibration intercept because we chose to automatically update when new predictors were present, regardless of noninferiority conditions. A sensitivity analysis of the trigger settings found poorer calibration intercepts when trigger thresholds were reduced (ie, lower threshold for needing an update) and when new predictors did not force refitting (Details in Appendix E). The estimated hazard ratios for this pipeline changed less frequently than the proactive updating pipeline but the changes tended to be larger in magnitude (Fig S1, Appendix E).

5. Discussion

In this study, we described *proactive* and *reactive* pipelines for dynamic updating of clinical prediction models that allow different modeling and performance criteria choices. Implementation of such a pipeline requires decisions to be made about updating frequency, updating methods, performance criteria, and trigger/threshold settings. Although certain methods may suit particular

Table 3. Performance results from the proactive updating pipeline with Bayesian dynamic updating

Per	Selected	C-index			Brier score			Calibration intercept			Calibration slope		
u	Update	Initial	Prior	Update	Initial	Prior	Update	Initial	Prior	Update	Initial	Prior	Update
0[a]	Initial model	0.84	NA	NA	0.08	NA	NA	−0.10	NA	NA	0.90	NA	NA
1[b]	No update	0.87	0.87	0.87	0.08	0.08	0.08	0.13	0.13	0.13	1.01	1.01	1.01
2[c]	Using ξ = 0.7	0.82	0.82	0.82	0.09	0.09	0.08	0.13	0.13	−0.03	0.80	0.80	0.89
3	Using ξ = 0.8	0.82	0.88	0.88	0.06	0.06	0.09	0.02	−0.19	−0.08	1.00	1.12	0.99
4	Using ξ = 1.0	0.89	0.89	0.89	0.06	0.06	0.06	0.10	−0.03	0.07	1.04	0.95	1.00
5	Using ξ = 1.0	0.89	0.89	0.89	0.06	0.06	0.06	0.14	0.12	0.08	1.02	1.03	0.96
6	Using ξ = 0.8	0.89	0.90	0.90	0.06	0.05	0.05	0.11	−0.03	−0.09	1.05	0.96	0.91
7	Using ξ = 0.7	0.92	0.93	0.92	0.05	0.04	0.04	0.11	−0.04	0.06	1.14	1.04	1.03
8	Using ξ = 1.0	0.90	0.91	0.92	0.04	0.04	0.04	0.08	0.10	0.08	1.07	1.03	1.05
9	Using ξ = 0.7	0.91	0.92	0.92	0.03	0.03	0.03	−0.07	−0.13	−0.07	1.05	1.01	1.01
10	Using ξ = 0.7	0.91	0.92	0.92	0.03	0.03	0.02	−0.34	−0.32	−0.21	1.08	1.011	1.011

C-index, Brier score, and calibration intercept and slope are shown for each period for the ‘‘Initial’’ (not updated) model f_0, the ‘‘Prior’’ period’s model (the model to be updated) f^*_{u-1}, and this period’s selected ‘‘Update’’ model f^*_u. Using the row Period u = 2 as an example, performance metrics were calculated in the period 2 holdout data for the initial model; the period 1 model to be updated f^*_1; and the period 2 selected model (the best performing update candidate) $f^k_2 = f^*_2$. (See Table S1 for a description of the update datasets.).

[a] The initial model was fit to the development dataset in period u = 0. Updating has not yet begun.

[b] In period u = 1, the initial model was selected as the best performer of the update candidates, so it was retained. $f^*_1 = f_0$.

[c] In period u = 2, the period 1 model is updated with a Bayesian model with forgetting factor ξ = 0.7. This becomes the period 2 model f^*_2.

datasets and environments better than others, the most important step is taking the decision to move away from ad hoc updating and to implement a dynamic updating process. There is no set of implementation choices that will guarantee an optimal updating process. Rather, a good pipeline will consist of well thought-out decisions about the components to match the resources and data available.

In our illustrative example, we used a proactive pipeline using Bayesian updating methods and a reactive pipeline based on refitting and recalibration. However, we could have combined Bayesian updating with a reactive pipeline or refitting and recalibration with a proactive pipeline. Similarly, other choices for performance measures and their evaluation could be employed. In our example of updating

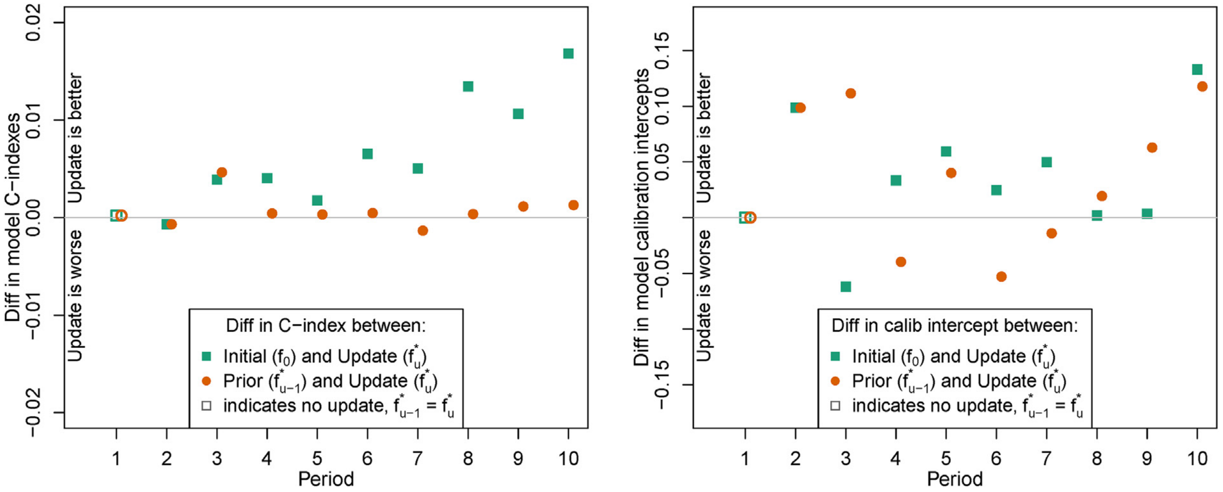

Figure 4. Results from the proactive updating pipeline with Bayesian dynamic updating. Performance difference of the updated model over the initial model (green squares) and over the prior period’s model (orange circles). Difference in C-index between the 2 indicated models is shown on the left and difference in absolute value of the calibration intercept between the 2 models is shown on the right. Unfilled squares and circles denote periods in which the model from period *u* - 1 was selected as the best performer in period *u*. Points above the horizontal line at 0 indicate that the updated model had superior performance to either the prior or initial model and points below 0 indicate that the updated model had inferior performance. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

Table 4. Performance results from the reactive updating pipeline with recalibration and refitting

Per		Selected	C-index			Brier score			Calibration intercept			Calibration slope		
u	Trigger	Update	Initial	Prior	Update	Initial	Prior	Update	Initial	Prior	Update	Initial	Prior	Update
0[a]	NA	Initial	0.84	NA	NA	0.08	NA	NA	−0.05	NA	NA	0.87	NA	NA
1[b]	calib int	Recal	0.87	0.87	0.87	0.08	0.08	0.08	0.17	0.17	0.09	0.98	0.98	0.96
2[c]	C-index, calib slope	Refit	0.82	0.82	0.82	0.09	0.09	0.09	0.18	0.09	0.13	0.77	0.76	0.81
3	new predictors	Refit	0.88	0.88	0.89	0.06	0.06	0.06	0.08	0.03	−0.03	0.96	1.00	0.96
4	none	No update	0.88	0.89	NA	0.06	0.06	NA	0.16	0.03	NA	0.99	0.93	NA
5	none	No update	0.89	0.89	NA	0.06	0.06	NA	0.20	0.10	NA	0.97	0.94	NA
6	none	No update	0.89	0.90	NA	0.06	0.05	NA	0.17	0.00	NA	1.00	0.95	NA
7	none	No update	0.92	0.93	NA	0.05	0.04	NA	0.17	0.00	NA	1.09	1.07	NA
8	new predictors	Refit	0.90	0.91	0.91	0.04	0.04	NA	0.15	0.03	0.15	1.02	1.05	1.01
9	new predictors	Refit	0.91	0.92	0.92	0.03	0.03	0.03	0.01	−0.03	0.02	1.00	0.98	0.98
10	calib int	Refit	0.90	0.92	0.92	0.03	0.02	0.02	−0.26	−0.21	−0.09	1.03	1.07	1.06

C-index, Brier score, and calibration intercept and slope are shown for each period for the ''Initial'' (not updated) model f_0, the ''Prior'' period's model (the model to be updated) f^*_{u-1}, and this period's ''Update'' model f^*_u if an update was indicated by the triggers. Using Update 2 as an example, two different triggers fired leading to an update by refitting. Performance metrics calculated in the period 2 holdout data are shown for the initial model; the period 1 model to be updated f^*_1 ; and the updated model fit to the period 2 training data f^k_2 . (See Table S1 for a description of the update datasets.

[a] The initial model was fit to the development dataset in period u = 0. Updating has not yet begun.

[b] In period u = 1, a calibration intercept trigger led to the initial model being updated via intercept recalibration.

[c] In period u = 2, triggers on both C-index and calibration slope led to the period 1 model being refit.

a model for prediction of 5-year survival in people with CF, we used an annual update cycle because this is how the data are released by the UK CF Registry. In some contexts where data may be acquired more frequently, it may be possible to update more frequently. There are trade-offs between sample size and speed, and different updating methods have different data requirements. Updating can use a combination of old and new data to increase sample size but this slows the pace of change of the model. The rate of updating must also be balanced between how rapidly the environment is changing and therefore, how quickly the current prediction model's performance is deteriorating, and how frequently downstream users of the prediction model can absorb changes. Abrupt changes to predicted outcomes are likely to be confusing to clinicians and patients. The sample splitting scheme (random, cross-

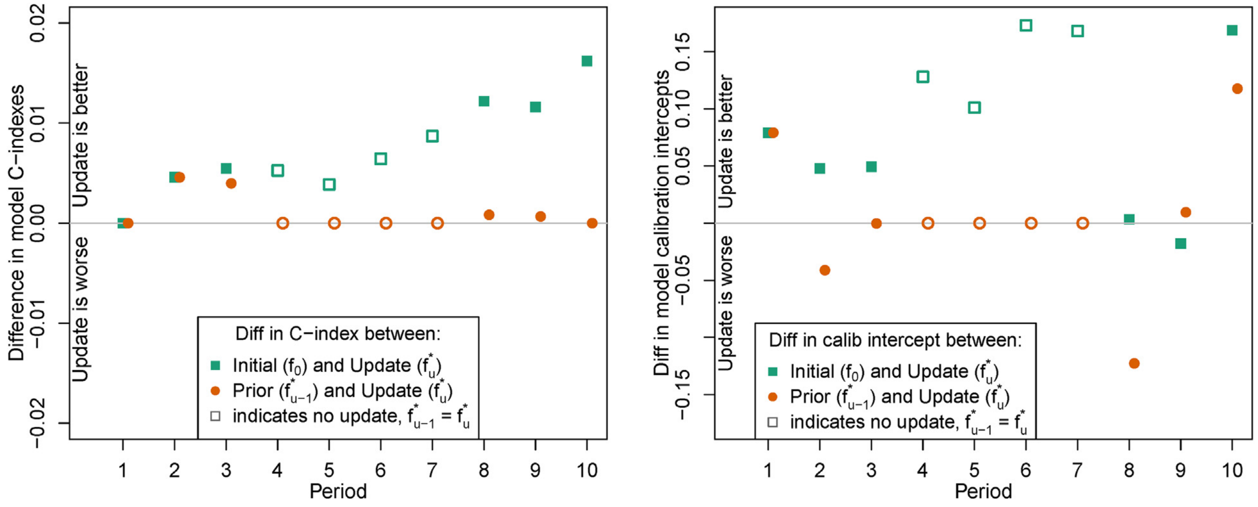

Figure 5. Results from the reactive updating pipeline with recalibration and refitting. Performance difference of the updated model over the initial model (green squares) and over the prior period's model (orange circles). Difference in C-index between the 2 indicated models is shown on the left and difference in absolute value of the calibration intercept between the 2 models is shown on the right. Unfilled squares and circles denote periods in which the model from period u - 1 was selected as the best performer in period u. Points above the horizontal line at 0.0 indicate that the updated model had superior performance to either the prior or initial model and points below zero indicate that the updated model had inferior performance. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

validation, bootstrapping) choice has implications on computing times, memory requirements, and model overoptimism, and needs to be carefully considered. Also, we focused on a pure prediction problem; there may be other considerations (eg, strong assumptions about confounding, focus on "causal" exposure during model development, etc.) for updating explanatory causal models used to quantify treatment effects or for policy making. The updating pipelines considered in this paper could also be useful in the emerging context of models for enabling prediction under interventions [30,31].

In the illustration of survival prediction in CF, when using a proactive updating strategy, the prediction model was updated more often than with the reactive strategy because there was no threshold determining whether to update or not. The reactive strategy had slightly better calibration during periods 4-7 when the model was not being updated compared to the proactive strategy where the model was updated in each of those periods. Overall, both pipelines led to improvements in calibration and discrimination in most periods in comparison to no updating.

The machine learning literature contains studies on continuously updating predictive models in the context of continuously generated streaming data. Much of this work was motivated by computational concerns when working with very large amounts of data, how to identify concept drift, and dealing with class imbalance [32,33]. In our clinical setting, it is more common for data to be released in batches after cleaning and anonymization. [21] Feng et al. proposed procedures for updating a machine learning−based prediction model with externally supplied candidate updates by limiting the error due to bad updates in a binary outcome setting using sensitivity and specificity. The methods here differ in that creating candidate updates is part of the pipeline.

New treatments and procedures are continually being discovered that change the risk of an outcome. In this environment, we expect the performance of clinical prediction models to deteriorate as changing risk causes calibration drift and new information must be accounted for. Having a dynamic updating pipeline in place ensures that performance metrics will be calculated on an ongoing basis and the decisions about when and how to update will be made in a timely, principled manner.

CRediT authorship contribution statement

Kamaryn T. Tanner: Writing − review & editing, Writing − original draft, Software, Methodology, Formal analysis, Conceptualization. **Karla Diaz-Ordaz:** Writing − review & editing, Methodology, Formal analysis, Conceptualization. **Ruth H. Keogh:** Writing − review & editing, Methodology, Formal analysis, Conceptualization.

Data availability

The authors do not have permission to share data.

Declaration of competing interest

There are no competing interests for any author.

Acknowledgments

The authors wish to thank Diana Bilton and Siobha'n Carr for providing clinical expertise that greatly enhanced the understanding of cystic fibrosis (CF) and the UK CF Registry dataset. The authors also thank people with cystic fibrosis and their families for consenting to their data being held in the UK CF Registry and NHS teams in CF centers and clinics for the input of data into the Registry. The UK CF Registry is sponsored and coordinated by the UK Cystic Fibrosis Trust.

Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jclinepi.2024.111531.

References

[1] Steyerberg EW. Clinical Prediction models: a practical approach to development, validation and updating. Springer; 2009.

[2] Hickey GL, Grant SW, Caiado C, Kendall S, Dunning J, Poullis M, et al. Dynamic prediction modeling approaches for cardiac surgery. Circ Cardiovasc Qual Outcomes 2013;6:649−58.

[3] Davis SE, Lasko TA, Chen G, Siew ED, Matheny ME. Calibration drift in regression and machine learning models for acute kidney injury. J Am Med Inf Assoc 2017;24:1052−61.

[4] Jenkins DA, Sperrin M, Martin GP, Peek N. Dynamic models to predict health outcomes: current status and methodological challenges. Diagn Progn Res 2018;2(23).

[5] Davis SE, Greevy RA, Fonnesbeck C, Lasko TA, Walsh CG, Matheny ME. A nonparametric updating method to correct clinical prediction model drift. J Am Med Inf Assoc 2019;26:1448−57.

[6] Feng J, Gossmann A, Sahiner B, Pirracchio R. Bayesian logistic regression for online recalibration and revision of risk prediction models with performance guarantees. J Am Med Inf Assoc 2022;29:1−12.

[7] Janssen KJ, Moons KG, Kalkman CJ, Grobbee DE, Vergouwe Y. Updating methods improved the performance of a clinical prediction model in new patients. J Clin Epidemiol 2008;61:76−86.

[8] Su TL, Jaki T, Hickey GL, Buchan I, Sperrin M. A review of statistical updating methods for clinical prediction models. Stat Methods Med Res 2018;27:185−97.

[9] Tanner KT, Keogh RH, Coupland CA, Hippisley-Cox J, Diaz-Ordaz K. Dynamic updating of clinical survival prediction models in a rapidly changing environment. Diagn Progn Res 2023;47:9−10e.

[10] Vergouwe Y, Nieboer D, Oostenbrink R, Debray TP, Murray GD, Kattan MW, et al. A closed testing procedure to select an appropriate method for updating prediction models. Stat Med 2017;36:4529−39.

[11] Schnellinger EM, Yang W, Kimmel SE. Comparison of dynamic updating strategies for clinical prediction models. Diagnostic and Prognostic Research 2021;5:1–10.

[12] Taylor-Robinson D, Archangelidi O, Carr SB, Cosgri R, Gunn E, Keogh RH, et al. Data resource profile: the UK cystic fibrosis registry. Int J Epidemiol 2018;47:1–7.

[13] Collins GS, Dhiman P, Ma J, Schlussel MM, Archer L, Van Calster B, et al. Evaluation of clinical prediction models (part 1): from development to external validation. BMJ 2024;384:e074819.

[14] Steyerberg EW, Harrell FE, Borsboom GJ, Eijkemans M, Vergouwe Y, Habbema JF. Internal validation of predictive models: Efficiency of some procedures for logistic regression analysis. J Clin Epidemiol 2001;54:774–81.

[15] Schnellinger EM, Yang W, Harhay MO, Kimmel SE. A comparison of methods to detect changes in prediction models. Methods Inf Med 2022;61:19–28.

[16] Vickers AJ, Kent M, Scardino PT. Implementation of dynamically updated prediction models at the point of care at a major cancer center: making nomograms more like netflix. Urology 2017;102:1–3.

[17] Carroll O. Strategies for imputing missing covariate values in observational data. In: PhD thesis. London, UK: London School of Hygiene and Tropical Medicine; 2022.

[18] Balfour-Lynn I, King J. CFTR modulator therapies - effect on life expectancy in people with cystic fibrosis. Paediatr Respir Rev 2022;42: 3–8.

[19] Wood AM, Royston P, White IR. The estimation and use of predictions for the assessment of model performance using large samples with multiply imputed data. Biom J 2015;57:614–32.

[20] McCormick TH, Raftery AE, Madigan D, Burd RS. Dynamic logistic regression and dynamic model averaging for binary classification. Biometrics 2012;68:1–19.

[21] Feng J, Emerson S, Simon N. Approval policies for modifications to machine learning-based software as a medical device: a study of biocreep. Biometrics 2021;77:31–44.

[22] Keogh RH, Seaman SR, Barrett JK, Taylor-Robinson D, Szczesniak R. Dynamic prediction of survival in cystic fibrosis: a landmarking analysis using UK patient registry data. Epidemiology 2019;30:29–37.

[23] Liou TG, Adler FR, Fitzsimmons SC, Cahill BC, Hibbs JR, Marshall BC. Predictive 5-year survivorship model of cystic fibrosis. Am J Epidemiol 2001;153:345–52.

[24] Nkam L, Lambert J, Latouche A, Bellis G, Burgel P, Hocine M. A 3-year prognostic score for adults with cystic fibrosis. J Cyst Fibros 2017;16:702–8.

[25] Stanojevic S, Sykes J, Stephenson AL, Aaron SD, Whitmore GA. Development and external validation of 1- and 2-year mortality prediction models in cystic fibrosis. Eur Respir J 2019;54.

[26] Tanner KT, Sharples LD, Daniel RM, Keogh RH. Dynamic survival prediction combining landmarking with a machine learning ensemble: Methodology and empirical comparison. J Roy Stat Soc Stat Soc 2021;184:3–30.

[27] UK Cystic Fibrosis Registry. UK cystic fibrosis registry 2022 annual data report. 2023. Available at: https://www.cysticfibrosis.org.uk/sites/default/files/2023-10/CFT_2022_Annual_Data_Report_FINAL_v8.pdf. Accessed June 25, 2023.

[28] Raftery AE, Karny M, Ettler P. Online prediction under model uncertainty via dynamic model averaging : application to a cold rolling mill. Technometrics 2010;52:52–66.

[29] Efron B, Hastie T. Computer Age Statistical Inference: Algorithms, Evidence, and Data Science, chapter 15: Large-Scale Hypothesis Testing and False-Discovery Rates. New York: Cambridge University Press; 2016.

[30] Lin L, Sperrin M, Jenkins DA, Martin GP, Peek N. A scoping review of causal methods enabling predictions under hypothetical interventions. Diagnostic and Prognostic Research 2021;5:1–16.

[31] van Geloven N, Swanson SA, Ramspek CL, Luijken K, van Diepen M, Morris TP, et al. Prediction meets causal inference: the role of treatment in clinical prediction models. Eur J Epidemiol 2020;35:619–30.

[32] Gama J, Zliobaite I, Bifet A, Pechenizkiy M, Bouchachia A. A survey on concept drift adaptation. ACM Comput Surv 2014;46:1–37.

[33] Hoens TR, Polikar R, Chawla NV. Learning from streaming data with concept drift and imbalance: an overview. Prog Artif Intell 2012;1:89–101.