Stabilizing Calibration of Clinical Prediction Models in Non-Stationary Environments: Methods Supporting Data-Driven Model Updating

By

Sharon Elizabeth Davis

Dissertation

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

in

Biomedical Informatics

October 31, 2019

Nashville, Tennessee

Approved:

Michael E. Matheny, M.D., M.S., M.P.H.

Robert A. Greevy, Jr., Ph.D.

Thomas A. Lasko, M.D., Ph.D.

Colin G. Walsh, M.D., M.A.

ACKNOWLEDGEMENTS

I am grateful for the enthusiastic support of my committee. Together, my committee members created an environment that encouraged my academic curiosity, facilitated my successful research, fostered my increasing independence, and prepared me to confidently undertake the next phase of research career. My advisor, Dr. Michael Matheny, has been a patient, generous, and candid mentor for the past six years. I am a better and more confident researcher thanks to his thoughtful advice on taking intentional steps to challenge myself and grow professionally. Dr. Tom Lasko has been an engaged and creative collaborator, as well as a source of practical perspective on project scoping. Dr. Robert Greevy provided critical statistical insight, with an excitement for interesting statistical challenges that kept me optimistic. Dr. Colin Walsh helped bridge the technical and practical aspects of my research, serving as a reminder of how informatics and predictive analytics play out in the wider, non-research context. I am also thankful for the contributions of two previous committee members – Dr. Guanhua Chen and Dr. Christopher Fonnesbeck – who supported the initial phases of organizing and planning my dissertation.

The faculty, trainees, and staff of the Department of Biomedical Informatics created a wonderful community in which to work and grow. I particularly want to thank Cindy Gadd, Gretchen Jackson, Rischelle Jenkins, Mark Frisse, Josh Smith, Lina Sulieman, and Bryan Steitz.

This work was made possible by the generous support of multiple funding agencies. I could not have returned to graduate school without the support of a training grant from the National Library of Medicine (grant number: 5T15LM007450-15). Time and effort of my committee members and key collaborators were supported by the U.S. Department of Veterans Affairs Health Services Research & Development (grant numbers: VA HSR&D IIR 11-292 and 13-052) and the National Institutes of Health (grant number BCHI-R01-130828).

Finally, I am thankful for the unwavering support of my family. My eldest son, Elliot, who sweetly assured me that it was okay for me to be “on call” when I worked during playtime in the last push to finish and defend this dissertation. My husband, Jeff, who never questioned my decision to pursue a doctoral degree, even in my moments of uncertainty. As our lives completely changed over the past few years, Jeff’s consistent support and encouragement truly made this work possible.

TABLE OF CONTENTS

Page

Appendix

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Adam – Adaptive moment estimation algorithm

Adwin – Adaptive windowing

AKI – Acute kidney injury

AUC – Area under the receiver operating curve

CUSUM – Cumulative sum

DDM – Drift detection method

ECI – Estimated calibration index

EDDM – Early drift detection method

EHR – Electronic health record

E_{max} – Maximum absolute difference

EWMA – Exponentially weighted moving average

FN – False negative

FP – False positive

ICI – Integrated Calibration Index

IQR – Interquartile range

L1 – L1-regularized regression

LR – Logistic regression

LSR – Logarithmic scoring rule

NN – Neural network

O:E – Observed to expected outcome ratio

RF – Random forest

SPC – Statistical process control

VLAD – Variable life-adjusted

CHAPTER 1

INTRODUCTION AND OVERVIEW

Clinical prediction models are developed to support patient and provider decision-making,[1, 2] assist in resource allocation,[3] and adjust quality metrics for acuity[4-6] across an array of clinical specialties and settings.[4, 6-11] Opportunities to deploy prediction models in support of patient-level decision-making are arising as the adoption of advanced electronic health records (EHRs) accelerates.[4, 12-17] At the same time, our understanding of the challenges of incorporating predictive analytics into clinical care is rapidly evolving, requiring new methods and evidence-based recommendations.

One challenge central to the long-term, prospective application of prediction models is the continuously evolving nature of clinical environments and the resulting tendency of model performance to deteriorate over time.[10, 11, 18-23] Patient mix may change gradually or quickly as populations age, new facilities bring new populations to a health system, or models are transported across clinical settings.[24-27] Predictor-outcome associations may shift along with practice patterns or the healthcare process model, such as changes in clinical guidelines, provider experience, coding practices, measurement accuracy, EHR interfaces, data entry workflows, and data definitions.[10, 11, 27, 28] Such changes impact model performance, particularly in terms of calibration, in ways that vary in magnitude and form depending on the model's underlying learning algorithm.[18, 19, 29]

Models underlying population health management, quality assessment, and clinical decision support applications require a high degree of accuracy and developers must be responsive to any degradation in performance. As a result, updating strategies to sustain performance are becoming critical components of model implementations. A range of updating methods are available to correct performance drift, from simple recalibration to full model revision (i.e., refitting) and even model extension with the incorporation of new predictors.[10, 11, 26, 27, 30] These updating methods vary in complexity, data requirements, and analytical resource demands.[10, 11, 26, 27, 30] While a lack of model updating can harm the performance and utility of predictions, common default model maintenance strategies, such as regularly scheduled model refitting,[27, 31, 32] may be inefficient or even detrimental.[10, 11, 22] Such predefined updating plans also fail to account for variations in the response of different learning algorithms to changes in clinical environments, which may impact the timing, extent, and form of shifts in model accuracy.[18, 19, 29]

Despite literature documenting performance drift[18-20, 23, 29] and the availability of multiple updating methods,[10, 26, 30, 33] guidance is needed to better inform the appropriate timing and methods for model updating in practice. Recommendations on the design of model maintenance protocols are limited and provide little insight into how differences between learning algorithms may impact updating requirements. Best practices must balance the amount of available evidence in new patient data, the desire to avoid overfitting, and the reliability of the predictions on which users depend. Additionally, as the volume and complexity of prediction models implemented in production EHR and ancillary clinical systems expands, automated surveillance procedures that can be deployed on a collection of active models are needed.

Specific Aims

The central objective of this dissertation is to develop a suite of methods supporting data-driven model updating strategies and active model surveillance systems in order to consistently retain model calibration over time for both regression and machine learning models. While performance drift may affect models predicting categorical, continuous, or time-to-event outcomes, in this work we focus on dichotomous categorical outcome models. With this in mind, we will pursue the following specific aims.

Aim 1. To describe dynamic calibration curves that provide on-going insight into model performance. In order to understand whether and to what extent performance drift may be impacting the utility of a prediction tool, it is critical to efficiently maintain an up-to-date representation of model performance as it changes over time. By continuously updating as data accumulates, our method for dynamic calibration curves is designed to visualize evolving forms of calibration and support calculation of detailed performance metrics reflecting current model accuracy. We hypothesized that dynamic calibration curves will quickly shift curves in response to changes in model performance, capturing the new form of calibration across the range of predictions without requiring users to fully refit the curve on a new batch of recent data.

Aim 2. To design a calibration drift detection algorithm that alerts users to deteriorating model performance. In order to support timely response to performance drift, we leverage the dynamic calibration curves developed in Aim 1 to construct a calibration drift detection system. Using an adaptive windowing monitor,[34] we designed a system that both provides a data-driven approach to initiate model updating as performance declines and inform the updating process with insight into defining an updating dataset. We hypothesized that our calibration drift detection system would 1) detect multiple forms and speeds of performance drift, and 2) report a window of observations with similar performance characteristics to that occurring at the time of drift detection.

Aim 3. To develop a model-agnostic testing procedure to select between competing updating methods. In order to correct performance drift and maintain generalizability of updated models, we define a nonparametric testing procedure that selects between available updating methods while controlling for sample size and overfitting. The objective of our testing procedure is to recommend simple updates that provide comparable or superior performance to more sophisticated updating methods. We seek to address limitations of other tests[33] by developing a procedure that is customizable and widely applicable to models for categorical outcomes regardless of the underlying learning algorithm. We hypothesize that applying the recommendations of our testing procedure, as opposed to a predefined model refitting strategy, will minimize model adjustments, improving prospective model performance, and lead to more stable performance over time.

This work promotes a shift away from inefficient and potentially sub-optimal “one-size fits all” updating strategies.[18, 19, 29, 33] The methods developed here can be used to tailor model updates to address the requirements of specific use cases and prediction models. Together they lay the ground work for the design of automated, EHR-embedded prediction model surveillance procedures that promote the long-term performance and utility of prediction models underlying a variety of informatics applications for decision support and population management.

Dissertation Roadmap

In this dissertation, we begin in Chapter 2 with background on performance drift and the state of the art of both model updating and drift detection, highlighting key advantages and limitations in the existing literature. Chapter 3 describes the concept of a data-driven model surveillance and updating system, with an emphasis on how such a system fits into the existing prediction model lifecycle.

Chapter 4 presents a method for constructing dynamic calibration curves to maintain a current understanding of model calibration at any given time. We use incremental gradient descent with an adaptive learning rate to continuously learn up-to-date calibration curves without having to repeatedly refit curves on a sliding window of recent observations. This approach to model validation is designed for use with streaming data, making it well-suited for clinical environments continuously managing new patients and care encounters. These dynamic curves not only allow the estimations of stringent calibration metrics on the fly, but also provide a visualization of the evolving nature of model calibration.

Leveraging these dynamic calibration curves, we then propose and evaluate a calibration drift detection system in Chapter 5. This data-driven method seeks to alert users to any deterioration in model performance in order to support a timely response and restoration of acceptable levels of accuracy. Our detector, built on the adaptive windowing method,[34] is applicable to dichotomous outcome models regardless of the underlying learning algorithm, making the approach relevant for systems managing suites of diverse prediction models. To further support model managers, we designed the calibration drift detection system to provide actionable alerts by including information on a window of recent data that may be appropriate for updating, if required.

Chapter 6, based on a study published in the Journal of the American Medical Informatics Association,[35] develops a nonparametric testing procedure that recommends updating methods while minimizing overfitting and accounting for uncertainty associated with the updating sample size. The testing procedure permits customization to meet use case requirements and is widely applicable to both parametric and nonparametric models. We illustrate the properties of our procedure on both simulated scenarios of population shifts that impact clinical use cases and two models developed and applied over time to Department of Veterans Affairs inpatient admissions.

In Chapter 7, we explore the implications of using a data-driven strategy to guide selection of updating methods and the impact of underlying model learning algorithms. Based on a paper accepted for publication in the 2019 Proceedings of the AMIA Annual Symposium,[36] this chapter compares three scheduled updating strategies—retention of the original model, predefined model refitting, and test-based updating with the recommendations of our nonparametric testing procedure. These strategies are applied across multiple years of data on hospitals admissions in national populations for which calibration drift and variability in drift by learning algorithm has been previously documented.[18, 19] We assess differences in discrimination and calibration over time under each updating strategy, as well as whether and how the learning algorithm underlying the model impacts updating requirements and accuracy.

Finally, in Chapter 8, we bring these methods together to discuss how they can be integrated into the data-driven model surveillance and updating system described in Chapter 3. We consider the advantages and limitations of these methods, the contributions of this work, and remaining methodological gaps warranting future research.

CHAPTER 2

BACKGROUND

With the increasingly widespread incorporation of advanced predictive analytics into electronic health records and healthcare applications,[4, 12, 14, 15] our understanding of the challenges presented by their use in clinical care is rapidly evolving. We focus on one such challenge, that of deteriorating model accuracy as patient populations and the processes of clinical care shift over time.[10, 11, 20-23] In order to support patient safety, user confidence, and clinical utility, strategies to restore and sustain model accuracy are becoming critical components of predictive analytics implementations. Here we review how and why performance of prediction models changes over time; current methods for and approaches to model updating; existing data-driven updating techniques; and opportunities for improvement.

The Concern of Performance Drift

Performance of prediction models is commonly measured along two dimensions – discrimination (i.e., the ability to separate populations with and without the outcome or to correctly rank-order observations by risk) and calibration (i.e., the agreement between individual predicted and true probabilities).[37] While discrimination focuses on whether a model typically assigns higher predicted probabilities to observations with the outcome than observations without the outcome, it does not consider whether those predicted probabilities are well-aligned with observed outcome rates (i.e., calibrated). Although both facets of model performance are important to consider when evaluating a new model, they may not be equally important in all contexts.[23, 37-40] For use cases aiming is to stratify individuals by risk category, discrimination may suffice; for use cases presenting personalized predicted probabilities in support of decision-making, calibration becomes critical.[4, 7, 11, 22, 23, 38, 41] ***Erroneous patient-level risk estimates produced by miscalibrated models may lead to over-confidence, inappropriately alter treatment choices, or misappropriate resources.***[4, 23, 40, 42] For example, patients may be dissuaded from pursing potentially effective treatments when presented with elevated estimates of complication risk or may elect to undergo difficult treatments when presented with inflated estimates of negative disease prognosis.[23, 42] Even in the case of risk stratification, however, model calibration can significantly impact our understanding for those patients near clinically meaningful cut-points, emphasizing the importance of aligning predicted probabilities with true risk across the range of patient risk.[39]

Focusing on different dimensions of model performance impacts our understanding of the stability of performance as models are deployed over extended timeframes. In temporal validation studies, stable discrimination of clinical prediction models has been documented up to 20 years after model development.[18-20, 23, 29, 43-47] Calibration, on the other hand, has been observed to be quite susceptible to deterioration. A review of temporal calibration studies revealed calibration of clinical prediction models deteriorates over time, typically in the direction of overprediction and in many cases within the five years after model development.[18-20, 23, 43, 48-52] Figures 1 and 2 illustrates this pattern of stable discrimination alongside drifting calibration for models predicting hospital-acquired acute kidney injury and 30-day mortality after hospital admission in national cohorts of admissions to Department of Veterans Affairs facilities.[18, 19] Such performance drift has implications for the reliability of predictions, user trust in predictive applications, and model utility. For example, Minne *et al* documented the consequences of calibration drift on quality assessments, finding assessments of mortality rates among intensive care units to be overly optimistic as a result of uncorrected calibration drift.[20]

Figure 1. Annual performance by learning algorithm of a model for hospital-acquired acute kidney injury. Adapted from data in Davis *et al*.[19]

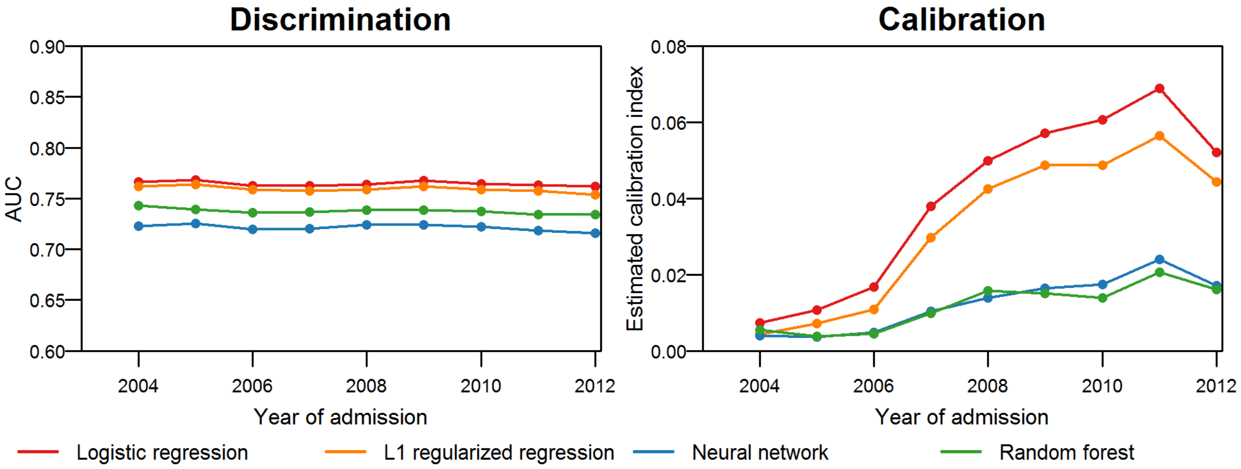
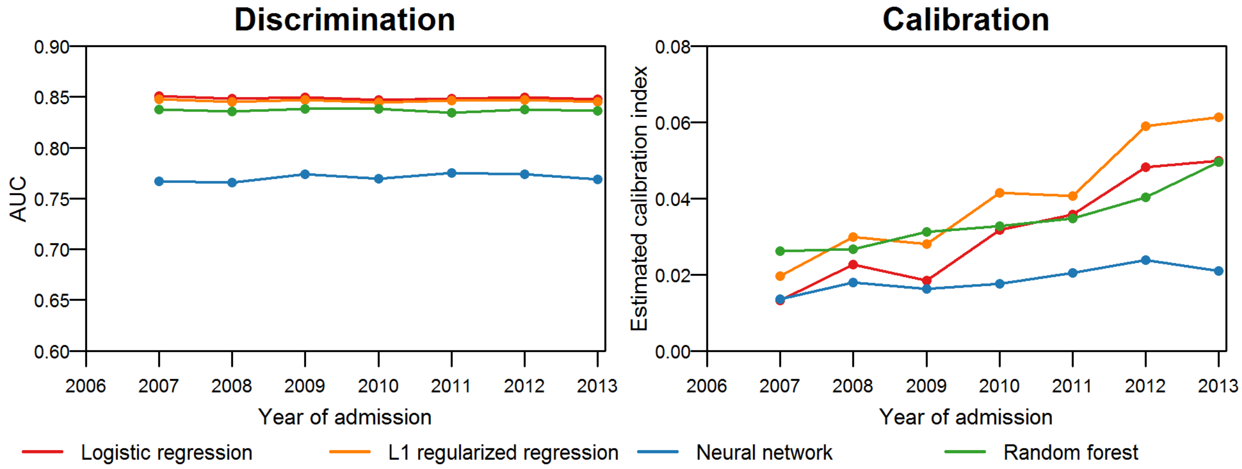


Figure 2. Annual performance by learning algorithm of a model for 30-day mortality after hospital admission. Adapted from data in Davis *et al.*[18]



Performance drifts over time as a result of models being deployed in non-stationary clinical environments where differences arise over time between the population on which a model was developed and the population on which that model is applied. This may include shifts in outcome rates, patient case mixes, and predictor-outcome associations.[10, 11, 23, 25] Data shifts that impact model performance are complex and can stem from the patient, provider, care process, or administrative domains (see Table 1).[10, 11, 27, 28] They may evolve gradually, for example when patient populations experience demographic shifts or new practice

Table 1. Example ways in which populations, clinical practice, and clinical data may change over time to impact prediction model performance.

Populations	Clinical Practice	Information
• Demographics composition[10, 23, 24] • Risk factors distributions[18, 23, 24] • Outcome incidence[23, 53] • Care access/utilization[24-26]	• Treatment patterns/ preferences[22, 26, 53] • Clinical guidelines[54] • Workflows/processes[55] • Provider experience[10, 56] • Scientific insights[10]	• Clinical information system design[57, 58] • Coding practices[59, 60] • Data definitions[10, 26, 55] • Measurement patterns and accuracy[10, 55, 61]

patterns emerge among care providers. They may also occur suddenly, as may be the case when new facilities join a health system or models are transported across clinical settings.[24-27] Predictor-outcome associations may change in expected ways under new clinical guidelines or unexpected ways as information systems and workflows evolve.[10, 11, 24, 27, 28]

Anticipating when and in what form performance drift may arise as a result of data shifts is challenging. Limited research has directly studied the link between performance drift and temporal changes in patient populations or clinical environments. However, the available evidence suggests patient case mix, outcome rates, and predictor-outcome associations do not shift in isolation, and complex, simultaneous shifts may be typical.[18, 19, 23, 62, 63] Studies also reveal prediction models based on common regression and machine learning algorithms are all susceptible to calibration drift. ***The form, degree, and speed of that calibration drift, however, varies by learning algorithms and data shift circumstances.***[18-20, 29] All models methods are susceptible to changes in the underlying event rate, while shifting case mix and predictor-outcome associations may have a greater impact on regression than machine learning approaches.[18, 19] ***These findings highlight the need to tailor the response to performance drift around model features and the environment in which models are applied***.

State of Model Updating

A Spectrum of Updating Methods

For predicted probabilities to be meaningful and potentially useful in clinical care, predictions must be highly accurate and reliable.[4, 41, 64] Thus systems for responding to performance drift and returning performance to acceptable levels are required. Inadequate performance of clinical prediction models commonly prompts researchers to develop entirely new models.[11, 22, 53] As a result, many models are published for the same outcome,[10, 53, 65] creating numerous competing models and complicating broad implementation. This approach also neglects information from previous modeling efforts and often utilizes smaller datasets than the original model.[10, 11, 22] Alternatively, ***a variety of model updating methods, varying in comprehensiveness, are available, many of which retain and extend knowledge from previous modeling efforts.***[10, 11, 24, 26, 27, 30]

Table 2 described a series of increasingly detailed updating methods that are widely applicable to dichotomous outcome models developed with a variety of learning algorithms. These methods vary in their ability to address different aspects of calibration drift, as well as

Table 2. Overview of common, widely applicable model updating techniques ordered by complexity/degree of model transformation.

	Method	Description	Issues addressed
0	Retention of original model	The original model is unchanged.	None
1	Intercept correction	Predictions from the original model are adjusted based on a logistic model with only an intercept.	Systematic over/underprediction
2	Linear logistic recalibration	Predictions from the original model are adjusted based on a logistic model defining a linear relationship between predictions and outcomes.	Over/underfitting and systematic over/underprediction
3	Flexible logistic recalibration	Predictions from the original model are adjusted based on a logistic model allowing nonlinear (e.g., spline, polynomial) relationships between predictions and outcomes.	Complex miscalibration varying in form and magnitude across range of prediction
4	Partial association adjustment	A combination of logistic recalibration and re-estimation of select predictor coefficients (parameter models only)	Complex miscalibration with special attention to known changes in variable relationships or definitions
4	Model refitting/ re-estimation	The model is re-estimated on new data with no changes to variable definitions or model form. Hyperparameters may be retuned.	Complex miscalibration, including that due to predictor-outcome association changes
5	Model revision / extension	The model is built on new data with possible changes to predictor set and/or model parameterization.	Complex miscalibration including that due to predictor-outcome association changes or omitted variables

their complexity and data requirements. Recalibration techniques (i.e., methods 1-4 in Table 2) retain information in existing models and improve generalizability, making these approaches preferable to model rebuilding and model revision when recalibration is sufficient to improve performance to acceptable levels.[11, 24, 26, 30] Common recalibration methods include intercept correction and linear logistic recalibration.[10, 11, 26, 30] These methods are based on mean and weak calibration metrics, which are limited in the forms of miscalibration they can detect.[39] Such approaches may thus not be able to correct for complex miscalibration that varies in direction and/or magnitude across the range of predicted probabilities. Flexible logistic recalibration can provide more nuanced correction for models in which calibration varies in magnitude and form across the range of predicted risk.[66] Although able to correct for more complex forms of miscalibration, flexible logistic recalibration has yet to be widely implemented. For those cases in which local knowledge indicates specific predictor definitions have changed or the literature highlights new predictors warranting inclusion, partial association adjustment combining both recalibration and estimation of select additional coefficients may be appropriate.[24, 26] Rebuilding the original or an extended model with new data may be required in response to substantial data shifts, significant care process changes, or critical new biological or system insights.[24]

Limitations of Current Updating Strategies

Despite recommendations emphasizing a consideration of recalibration prior to refitting a model,[11, 24, 26, 30] current updating protocols often call for regularly scheduled model refitting on an annual or biannual basis.[27, 31, 32] This baseline approach requires users make critical assumptions regarding the form and pace of performance drift. ***While a lack of model updating can harm the performance and utility of predictions, common prescribed updating strategy may be inefficient or even detrimental.***[10, 11, 22, 35]

By defaulting to refitting models at each update point, this strategy presumes data shifts warrant abrupt forgetting and retraining of all previously learned associations. As noted above, not only does this approach neglect information gleaned from previous modeling efforts, but can also lead to overfit models that lack generalizability, especially when updating datasets are smaller than development cohorts.[10, 11, 22, 33] Such predefined updating plans also fail to account for variations in the response of models trained with different learning algorithms to changes in clinical environments, which may impact the extent and form of changes in model accuracy[18, 19, 29] Recalibration may be more appropriate than refitting for models in clinical use when equivalent or improved performance can be achieved by the former. At the other extreme, refitting a model may not sufficiently correct performance if care processes have been modified, variable definitions or measurement accuracy have changed, or new predictors are available.[24] For these reasons, the selection of updating methods likely requires more guidance and flexibility than current typical updating strategies provide.

Prescheduled updating protocols, paying little or no attention to model performance between scheduled maintenance periods, may not be sufficient to ensure stable model accuracy. Performance drift occurs at variable speeds due to both the rate of change in populations and clinical practice, as well as how quickly different models respond to such changes.[18, 19] During periods of rapid performance drift, waiting for scheduled updating points may allow for unacceptably long durations of reduced accuracy in the interim. On the other hand, during periods of relatively stable performance, scheduled updates may result in unnecessary refitting of well-performing models, possibly reducing model generalizability and reliability. Implementing methods to detect performance drift would support triggered, data-driven model updating that responds to performance drift as it occurs, improving model stability with efficient model updating.

Alternative Strategies and Their Limitations

Thus far, our discussion has focused on updating as a means to restore the performance of static prediction models. Online learning algorithms, which continuously update models as new observations become available,[28, 67, 68] stand as an alternative to periodic updating of static models with either predefined or data-driven strategies. By incorporating changes in the environment as they occur, online models may provide more stable performance over time compared to static models.[27, 28, 62, 67, 68] While online models have been applied to health outcomes, such continuously updated models have yet to be incorporated into clinical tools.[27, 28, 63]

The shift to an online paradigm is not straightforward for clinical use cases. Documenting the performance characteristics of continuously changing online models will require the development of new validation techniques.[28, 67] Methodological innovation will also be required to enable online versions of additional learning algorithms, particularly for increasingly popular deep learning models. As was the case with the introduction of machine learning, the incorporation of new modeling techniques into clinical applications is accompanied by a need for user education to ensure acceptance and understanding. A fundamental change in the structure of clinical prediction models from a static to a dynamic online context will require similar investment.[67] From a policy perspective, the regulatory framework governing the implementation of online models in clinical settings is in early development and continues to evolve.[69] As a result of these larger challenges to the implementation of continuously updated prediction models, we focus further discussion and methods development within the current, static modeling context.

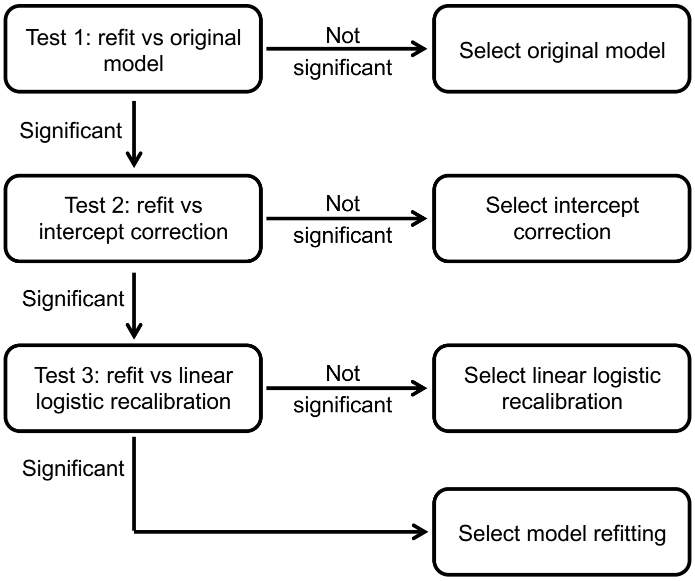
Data-Driven Updating

The variety of possible causes of and responses to performance drift warrant flexible and customizable model updating strategies. Data-driven methods guiding when and how to respond to deteriorations in model performance may promote long-term model utility while also addressing the requirements of specific use cases and prediction models.

Techniques Informing the Method of Updates

Given the variety of available model updating methods, there is a need for guidance regarding the selection of the most appropriate method in a given performance drift scenario. Vergouwe *et al*[33] recently described a closed testing procedure to select between updating methods with the aim of balancing the amount of available evidence in new observations and the desire to avoid overfitting. Using a series of likelihood ratio tests and assuming model refitting is the gold standard updating approach, the closed testing procedure selects the simplest updating method providing a fit similar to model refitting. Figure 3 illustrates this closed testing process.

Figure 3. Vergouwe *et al*'s closed testing procedure to select among updating methods.


Test 1: refit vs original model
Not significant
Select original model
Significant
Test 2: refit vs intercept correction
Not significant
Select intercept correction
Significant
Test 3: refit vs linear logistic recalibration
Not significant
Select linear logistic recalibration
Significant
Select model refitting

This testing procedure is limited in a number of ways. First, the approach assumes the user is interested in updating a model built using logistic regression. However, clinical prediction models are increasingly being developed with nonparametric and semiparametric machine learning and regression techniques.[4, 12-16] As our prior work revealed, all modeling methods are susceptible to performance drift, and the form of drift varies across modeling methods due to differences in their robustness to changes in clinical environments.[18, 19] The closed testing procedure also exhibits too strong a preference for model refitting, recommending this method even when refitting does not provide performance advantages over recalibration.[70] We observed this testing procedure's preference for refitting in our own work as well (see Chapters 6 and 7). These results appear to stem from a lack of explicit correction for overfitting and the assumption that a refit model is always the leading choice. In case of small updating samples, a refit model may be overfit and falsely appear to outperform other updating methods. ***Taken together, these limitations suggest a need for a more general method to select between updating techniques that can be widely applied regardless of a model's learning algorithm and without presuming simpler updates will never outperform refitting.***

Techniques Informing the Timing of Updates

The literature on concept drift (i.e., changes in distributions of predictors and outcomes) has long recognized the need to detect and respond to deteriorations in the performance of prediction models. A variety of data-driven concept drift detection algorithms have been developed to track performance and trigger model updating.[34, 68, 71-73] This research has typically focused on classification problems and thus drift detection studies have concentrated on identifying increasing rates of misclassification.[68, 72-74] For example, spam filters use prediction models to learn patterns distinguishing message types and are judged by their ability to accurately label new messages rather than their ability to predict the probability of whether each message is spam. These models experience drift as spam generators regularly change their approach to avoid detection. Drift detection algorithms aim to identify increases in the frequency of spam emails entering inboxes or legitimate messages being sent to junk folders. This translates into identifying changes in model discrimination. For clinical prediction models, however, we are more interested in identifying calibration drift than discrimination drift. Calibration is both more susceptible to drift[18, 19, 23, 29] and more critical to clinical decision-making applications.[20, 39-42, 46]

While several major methods, including the drift detection method (DDM)[75] and early drift detection method (EDDM),[72] are designed to track Bernoulli distributed error metrics and may not be easily extended to the calibration setting, other methods may be more amenable to tracking calibration metrics. Statistical process control charts (SPC),[76] and extensions such as cumulative sum (CUSUM)[68] and exponentially weighted moving average (EWMA) charts,[73] may be applicable to calibration drift detection. SPC methods have been implemented for a variety of healthcare applications—including tracking outcome rates,[77, 78] device safety,[77, 79, 80] quality improvement,[77, 81-84] and model performance.[29, 85] Prior work evaluating performance drift with SPC methods has focused on retrospective forensic evaluations of model deterioration.[22,65] Our searches did not revealed studies using these methods to trigger model updating in response to calibration drift. One variation of SPC, variable life adjusted years (VLAD) charts,[86] does track calibration using differences in observed and expected outcomes; however, this crude measure of calibration may not be sensitive enough to capture the diversity of calibration drift patterns. Other SPC methods may be able to detect changes in calibration, but do not necessarily provide guidance on what recent data may be relevant for subsequent model updating. Adaptive windowing (Adwin)[34] may be the most relevant drift detection method for identifying calibration drift. Although originally described using classification error, the Adwin algorithm does not presume users are interested in tracking a Bernoulli distributed metric[34] and may thus be extensible to calibration metrics. Adwin also inherently provides a window of recent data that may be suitable for updating in response to any detected drift.[34] We discuss the Adwin method in more detail in Chapter 5.

While ***existing drift detection algorithms were not designed with calibration in mind or may be limited in their direct applicability to surveilling clinical prediction models***, these methods do offer insight into how we might design a calibration drift detection system for clinical use cases. We briefly discuss some of these insights here and will revisit them in subsequent methods development chapters.

Predict-diagnose-update scheme

Most drift detection approaches are variations on the predict-diagnose-update scheme.[68, 72, 73] As new observations arrive, the active model makes a prediction. Once the outcome is observed, the detector evaluates the stability of the process and alerts the user to any significant change in performance. When the detector identifies a change, the model is updated in response. This framework is equally well-suited for surveilling models for changes in misclassification or calibration.

Patterns of drift

Studies of drift detection methods have explored multiple dimensions of data shift and resulting performance drift. Model performance may change abruptly, incrementally, or seasonally. The magnitude of change may be small or large.[68] Each drift detection method is better suited to detect certain forms of drift than others.[68, 87] For example, EDDM is more successful at identifying gradual changes than DDM,[72] and EWMA charts can detect smaller changes than traditional SPC charts.[73] With their focus on classification rather than probability, many drift detectors seek to simply recognize increases in error rates.[68] On the other hand, as we design a calibration-focused drift detector, we need to carefully consider the ways in which calibration may change. In our prior work exploring calibration over time, we noted some models experienced changes in the proportion of observations in calibrated regions of the probability scale, while other models had a relatively consistent proportion of observations in calibrated areas but experienced changes in the magnitude of over/underprediction among miscalibrated observations.[18, 19] The ability of our calibration drift detection system to address various speeds, magnitudes, and forms of calibration drift will be a key consideration.

Data longevity

The influence of each observation over time—both in terms of information content and storage requirements—varies across drift detection methods. Methods may be equally weight (e.g., DDM), abruptly forgot (e.g., sliding window), or gradually downweight (e.g., EWMA) older observations as newer observations accumulate.[68] This choice can influence how quickly changes are identified under different forms of drift.[68] In addition, many drift detection algorithms are designed to be used with streaming data, where observations arrive sequentially and storing all observations in memory is impractical or impossible.[68, 72, 73] Thus, algorithms commonly track and evaluate summary measures of performance rather than metrics that require all observations be available for distributional comparisons.[68] Some methods (e.g., DDM and EDDM) that avoid storing all observations in memory will store chunks of data during warning periods when drift is suspected.[68] This approach may be especially useful for informing the construction of updates sets when drift is detected.[68]

False alarms and speed of detection

The drift detection literature also highlights the importance of balancing false alarms, missed detections, and detection delays. The error tolerance parameter of each method informs the relative risk of false alarms and time from the start of performance drift to detection of the drift. However, even with careful parameter setting, methods may be more or less susceptible to false alarms under different speeds and magnitudes of performance drift.[68, 74, 87] New detection methods have often been developed in response to specific scenarios in which existing methods struggled. For example, EDDM was designed to detect gradual drift in response to limitations of DDM's performance in such cases.[72] A calibration drift detector should be designed to correctly trigger under common forms of miscalibration.

Methodological Gaps and Opportunities for Improvement

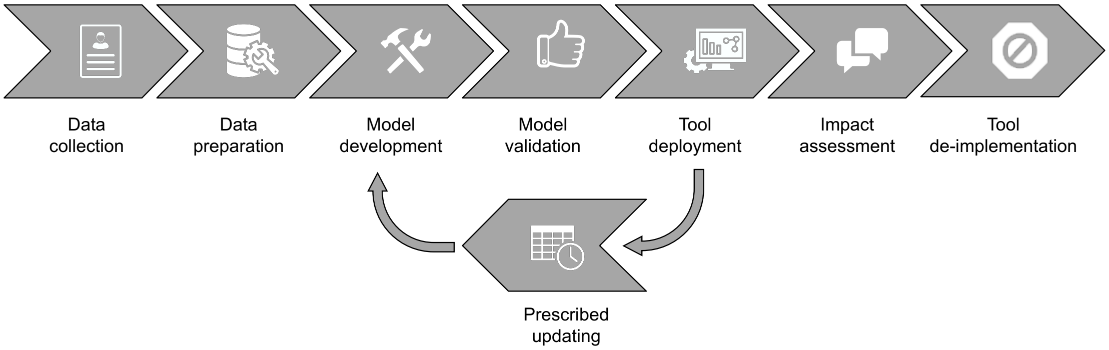
As the volume, complexity, and variability of prediction models implemented in health systems grows, data-driven updating policies could support model developers and managers as they endeavor to provide stable and accurate model performance. Data-driven updating strategies tailored to detect and respond to performance drift will become key components of automated surveillance systems underlying a variety of informatics applications. As highlighted in this chapter, current data-driven methods do not fully address the requirements of clinical use cases. New methods are needed to inform when and how to update clinical prediction models in order to respond to performance drift in a timely manner and promote prospective performance of updated models. These data-driven methods should be applicable to models regardless of the underlying learning algorithm and should be customizable to those aspects of model performance most relevant to each use case. We seek to develop these essential methods and evaluate their properties in a variety of settings.

CHAPTER 3

A FRAMEWORK FOR DATA-DRIVEN MODEL UPDATING

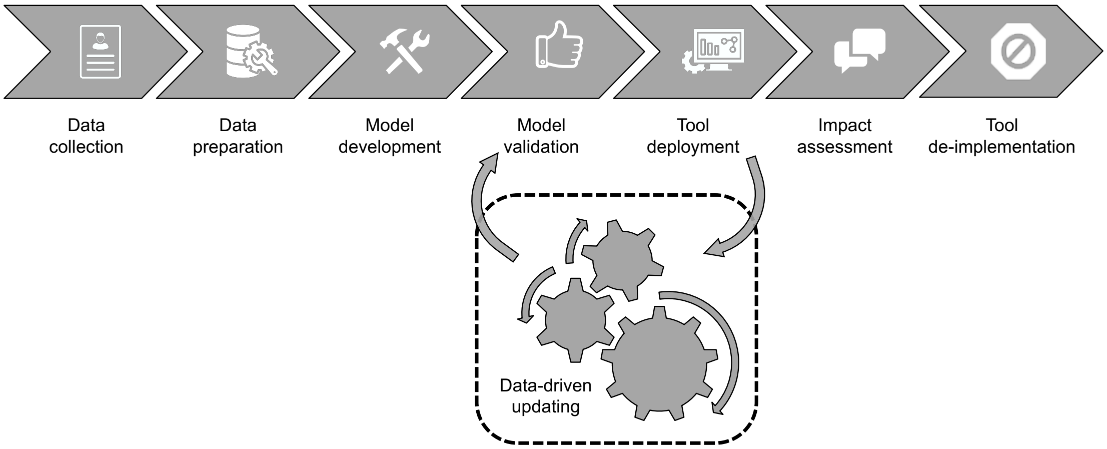
With increasing recognition of the need to maintain clinical prediction models over time, updating strategies to sustain performance are becoming critical components of the clinical modeling process. Figure 4 illustrates how common, prescribed model updating protocols integrate with the model development and implementation process. Plans calling for scheduled model refitting of deployed models create a straightforward cycle within the modeling process. However, as described in the previous chapter, such updating strategies neglect information learned from prior modeling efforts, are often more subject to overfitting than original models,[10, 11, 22] may not temporally align scheduled updating points with the speed of performance drift, and ignore the varying susceptibility of learning algorithms to changes in clinical environments.[18, 19, 29]

Figure 4. The clinical prediction modeling process with a predefined updating strategy.



In response to these limitations, we offer the revised clinical prediction modeling process in Figure 5. Using a data-driven updating strategy, incoming data on new patients and clinical encounters guide the updating process for deployed prediction models as performance deteriorates. A data-driven updating approach not only tailors the timing of updating, but also the means by which models are updated. Rather than assuming refitting a model on a new batch of recent data is always the best approach, a data-driven system considers model refitting as well as updating methods that integrate existing model insights with information in recent data. Under both the original and revised modeling processes, updated models will require

Figure 5. The clinical prediction modeling process with a data-driven updating strategy.



validation to ensure clinically acceptable performance is restored prior to continued model deployment.

A data-driven updating approach to model maintenance requires a suite of new methods that learn from evolving patient data streams. Through the series of studies described in this dissertation, we seek to develop the methods required to enable this data-driven model updating cycle. We address four key questions that any data-driven updating system must consider:

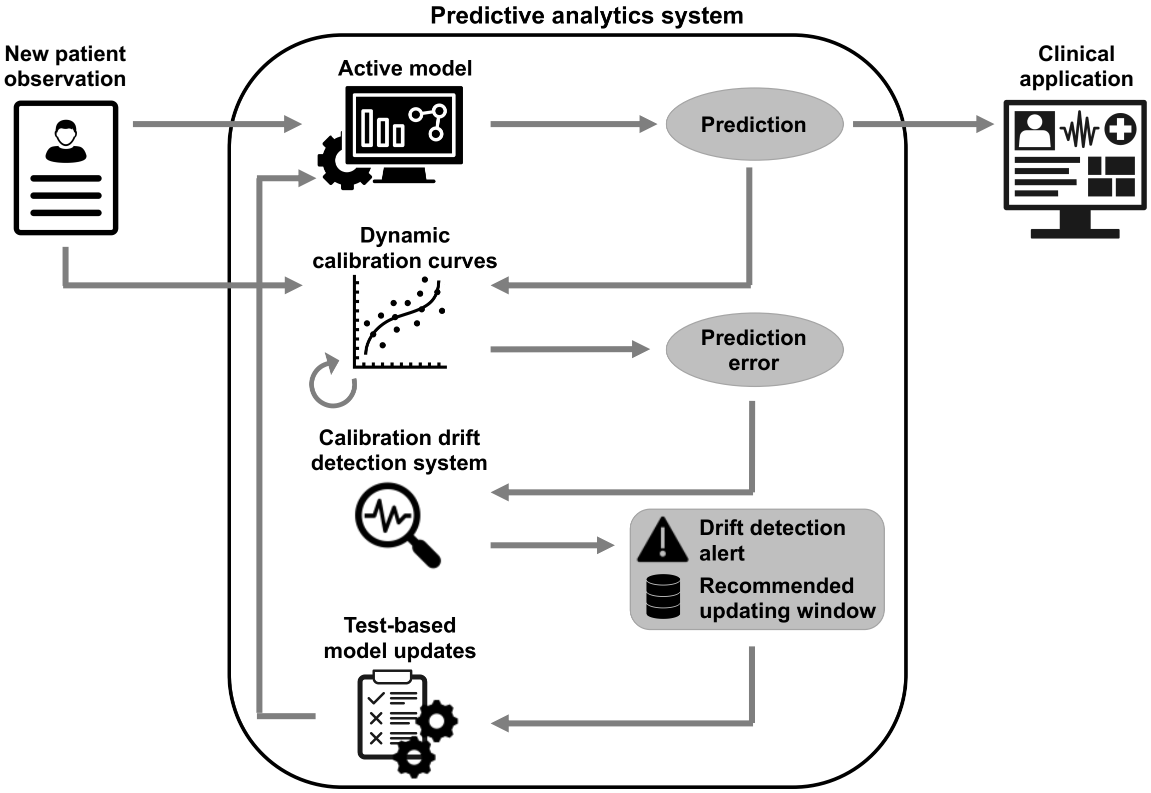
1. How is model performance evolving over time?
2. When has performance drifted significantly such that updating may be warranted?
3. What window of recent data should be used to update the model once drift is detected?
4. What is the best updating method to apply in order to improve performance while maintaining generalizability of the model for future patients?

We propose methods to address each question. First, dynamic calibration curves assess model performance in real-time, providing visualizations of changing model performance and supporting calculation of up-to-date stringent calibration metrics. Second, our calibration drift detection system alerts users to significant changes in model calibration and indicates a window of recent data they may be appropriate for model updating in response to this performance drift.

Finally, our nonparametric testing procedure evaluates competing updating methods to recommend model adjustments that improve performance while promoting model generalizability and applicability to subsequent patients. By using the accumulation of observed data to answer each question rather than making assumptions regarding the timing and form of performance drift, our data-driven methods may support more consistent, reliable, and efficient clinical prediction.

In Figure 6, we integrate these methods into a conceptual model for a data-driven active model surveillance and maintenance system. Such a system is built on the predict-diagnose-update scheme central in the drift detection literature.[68, 72, 73] When data on a new observation, or patient in our case, arrives, a prediction is generated using the current, active version of the prediction model. The error of this prediction is then estimated from the current dynamic calibration curve, which is subsequently updated once the observation's outcome becomes available. The prediction error is submitted to the calibration drift detection system which monitors the distribution of prediction error over time, triggering an alert when a change in the

Figure 6. Conceptual model of a data-driven active model surveillance and maintenance system.



error distribution is observed. In addition to alerting the user to performance drift, the detector notes the set of recent data that appears to have a consistent prediction error distribution and may be a good candidate for use in subsequent updating. When performance drift is signaled, the testing procedure is initiated and compares available updating methods using data from the suggested window to train each updating method. The test-recommended updating method is applied and this revised model becomes the new active model to be applied as new patients continue to arrive.

While the data-driven methods we develop here can be used in conjunction, as illustrated in Figure 6, we can also conceive of independent use cases for each method. For example, dynamic calibration curves may be implemented for visualization or metric monitoring in a dashboard without integrated testing and alerting. While the calibration drift detection system we describe in Chapter 5 is tightly linked to dynamic calibration curves, implementing the system with an alternative calibration metric removes this dependency. Similarly, our testing procedure to recommend updating methods could be implemented as needed or on a schedule rather than in response to a detected drift in performance.

Each of our data-driven methods is designed with three key features in mind. The methods should be ***practical*** in that they accomplish their stated goals without undue computation or analytic resource burdens. The methods should be ***generalizable*** to any categorical prediction model, regardless of the underlying learning algorithm. The methods should be ***customizable*** to meet the unique needs of diverse clinical use cases. We will return to these requirements in the final chapter to evaluate the strengths and limitations of the methods as developed through the course of this dissertation.

CHAPTER 4

DYNAMIC CALIBRATION CURVES FOR CONTINUOUS MODEL EVALUATION

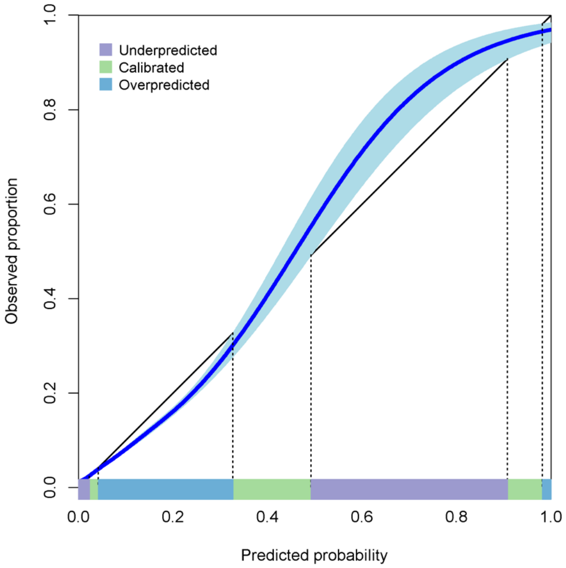
In this chapter, we propose a method for constructing dynamic calibration curves to provide on-going, up-to-date insight into model performance as it evolves over time. We bring together methods for graphical model validation[39] and continuous learning from streaming data[68] to provide continuous model assessment while minimizing both computational demands and assumptions regarding the time horizon of performance drift. These dynamic calibration curves not only allow the estimation of stringent calibration metrics in real-time without batching observations, but also provide visualizations of the evolving nature of model calibration over time.

Static Calibration Curves

Calibration curves are a graphical representation of model performance across the range of predicted probability. For categorical outcome models, calibration curves are developed by regressing observed outcomes on some function of the predicted probabilities.[88-90] Under the Cox recalibration framework, such curves were initially parameterized with linear associations between outcomes and predictions; however, calibration curves have since been extended to support nonlinear, flexible associations that better highlight variability of model performance in different regions of risk.[39] Nonlinear curves may parameterize the logistic calibration model with loess smoothers, splines, or polynomials.[39, 89-91]

Figure 7 provides an illustrative calibration curve. In a plot of the proportion of observations experiencing the outcome of interest against the predicted probability of that outcome, the 45° line represents perfect calibration, or perfect agreement between observed and predicted event rates. In the case of a calibrated model, the calibration curve would align with this ideal. In the common case of imperfect calibration, the calibration curve indicates how accurately a given prediction might reflect the true outcome rate among a group of similar observations. In regions where the curve falls above the ideal calibration line, model-based predictions are too low, underpredicting the probability of the outcome. Conversely, in regions where the curve falls below the ideal calibration line, the model overpredicts the probability of the outcome.

Figure 7. Illustrative calibration curve highlighting regions of calibration, overprediction, and underprediction.



While visualizing performance with calibration curves can provide important insight, graphical comparisons of multiple calibration curves, as may be generated by repeated temporal model validations, is difficult.[89, 90] Not only can it be challenging to overlay multiple curves, but graphical representations may not reveal key aspects of performance. Predicted probabilities are not uniformly distributed between 0 and 1, thus we need to consider the intersection of calibration curves and data distributions. Models that appear calibrated over only a small region may perform quite well in practice if most observations fall within this region. Similarly, highly miscalibrated regions may receive too much weight in a purely visual assessment if the region is only relevant to a few observations.[89] As a result, metrics have been developed to summarize these curves in variety of ways, including:

- Maximum absolute difference (E_{max})[92] – the maximum absolute difference between predicted probabilities and calibration curve fitted observed probabilities
- Estimated calibration index (ECI)[90] – mean squared difference between predicted probabilities and calibration curve fitted observed probabilities
- Integrated calibration index (ICI)[89] – mean absolute difference between predicted probabilities and calibration curve fitted observed probabilities

Both graphical representations and summary metrics stemming from calibration curves provide detailed, stringent assessments of model calibration[39] and are critical to clinical applications utilizing patient-level predictions.[39] Stringent calibration evaluations based on nonlinear calibration curves ensure models have a net benefit greater than or equal to treat-all or treat-none strategies, thus ensuring predictions are nonharmful to clinical decision-making.[39] These detailed assessments of performance are also critical to the model updating process as common, weaker calibration metrics (e.g., observed-to-expected outcome ratios and Cox recalibration intercepts/slopes) may conceal critical differences in performance across models and over time.[39] For example, in our previous studies exploring the interaction between learning algorithms and performance drift, we found learning algorithms to be variably susceptible to drift in ways that only become apparent in curve-based evaluations.[18, 19] Thus making calibration curves simple and readily available to model users, model managers, and model surveillance tools is crucial to establishing and maintaining useful clinical prediction applications.

Unfortunately, providing up-to-date calibration curves in a streaming data environment, such as that of clinical information systems, can become challenging. In order to provide a visualization and assessment of the current performance of a prediction model, we would need a means of fitting logistic calibration curves on-demand using a batch of recent observations. Defining an appropriate window of recent data requires users to both consider the sample size needed for building the logistic curve and anticipate the speed of performance drift. Ideally, each calibration curve would be constructed using a window of data during which model performance is stable. Given the complexity of clinical environments and the variable sensitivity of models to data non-stationarity, defining such a window is not straightforward and there may not be an appropriate rule of thumb to be applied across models and time. Even if one could define an appropriate window size, fitting a calibration curve upon arrival of each new observation could become burdensome for high volume, high velocity data streams.

Dynamic Calibration Curves

In order for on-going assessment of prediction models using calibration curves and stringent calibration metrics to be feasible, methods to avoid assumptions about the speed of performance drift and repetitive model building are necessary. We propose applying online learning methods to continuously maintain calibration curves, providing up-to-date representations of current model performance characteristics.

Traditional offline modeling builds a model on a batch of data and applies that model to new data.[68] Such models are static unless proactively updated with a new batch of observations.[68] In contrast, online models continuously update in response to the arrival of new data.[93, 94] As each new observation or small batch of observations becomes available, the current model integrates the new information into a revised version of the model. Online approaches are well-suited to high volume data streams because they achieve this continuous learning while avoiding the need to both retain all observations in memory and fully retrain on the expanded batch of data.[68, 93] Adaptive learning methods extend the online learning framework to allow models to react and evolve in response to changes in the data environment.[68]

Such methods are highly applicable to the challenges of maintaining up-to-date calibration curves. New patients and patient encounters constitute continuous data streams entering clinical information systems and being processed by the prediction models within these systems. As a results of shifting patient populations, care practices, and clinical environments these data streams are non-stationary. Given this setting, adaptive online learning methods are a promising approach to maintaining updated, evolving calibration curves.

Online Gradient Descent

Gradient descent can be applied to develop models of many varieties, including logistic regression models such as those underlying calibration curves. Gradient descent estimates model parameters by incrementally adjusting parameters toward those values that minimize error.[95] In repeated iterations, estimates for each observation are constructed using current parameter values and the gradient of a loss function is evaluated with these estimates. Parameter are then adjusted based on the gradient value proportional to some learning rate. By repeating this process multiple times, parameter estimates step toward optimal values that minimize loss.[95]

In its basic form, batch gradient descent, observations are processed together at each step.[95] Alternatively, stochastic gradient descent processes one randomly selected observation at a time without retraining on the entire dataset.[95, 96] Relaxing the random ordering requirement, stochastic incremental gradient descent applies the streaming online learning context by updating parameters estimates with each newly arriving observation.[13] By processing observations in temporal order, as model performance changes over time, the loss function will reflect this change and respond by stepping parameters toward newly optimal values. [96, 97] In this way, incremental gradient descent serves as an adaptive online learning algorithm.

The learning rate, a hyperparameter defining how much weight is given to the current observation, influences how quickly incremental gradient descent adapts in non-stationary environments.[96, 97] Small learning rates minimize the influence of new observations and provide more weight to prior data, which can slow adaptation toward newly optimal parameter values. Conversely, large learning rates can allow large changes in parameter values at each iteration, leading to noisy models.[96, 97]

In the case of adaptive learning for non-stationary models with anticipated performance drift, a constant learning rate may not be appropriate. We would prefer to learn more quickly during periods of change and more slowly during periods of stability.[96] The adaptive moment estimation algorithm (Adam) optimizes the model by scaling each parameter's learning rate (or step size) by the exponentially weighted moving averages of the gradient and squared gradient.[98] This adaptive learning approach makes Adam well-suited for use with non-stationary data streams.[98] Adam is also fast, computationally efficient, and widely implemented in machine learning applications.[99] It is thus this variation of gradient descent that we utilize to construct dynamic calibration curves.

Curve Specification

Applying the Adam optimization algorithm to streaming patient data, we employ and update a dynamic calibration curve for each new observation as follows:

1. For t > 0, calculate the predicted probability (p_t) of the outcome of interest using the active prediction model.
2. Provide relevant calibration assessments for p_t as required based on the logistic calibration curve defined with coefficients $\boldsymbol{\beta}_{t-1}$. This may include visualization of the current calibration curve with performance at p_t highlighted or an observation-level prediction error. For example, we may calculate the absolute difference between predicted and observed probabilities, where the observed probability ($\hat{p}_t$) is defined as the fitted value of the current calibration curve such that $\hat{p}_t = f(p_t)\boldsymbol{\beta}_{t-1}$ where $f(p_t)$ is a user-specified nonlinear expansion of the p_t.

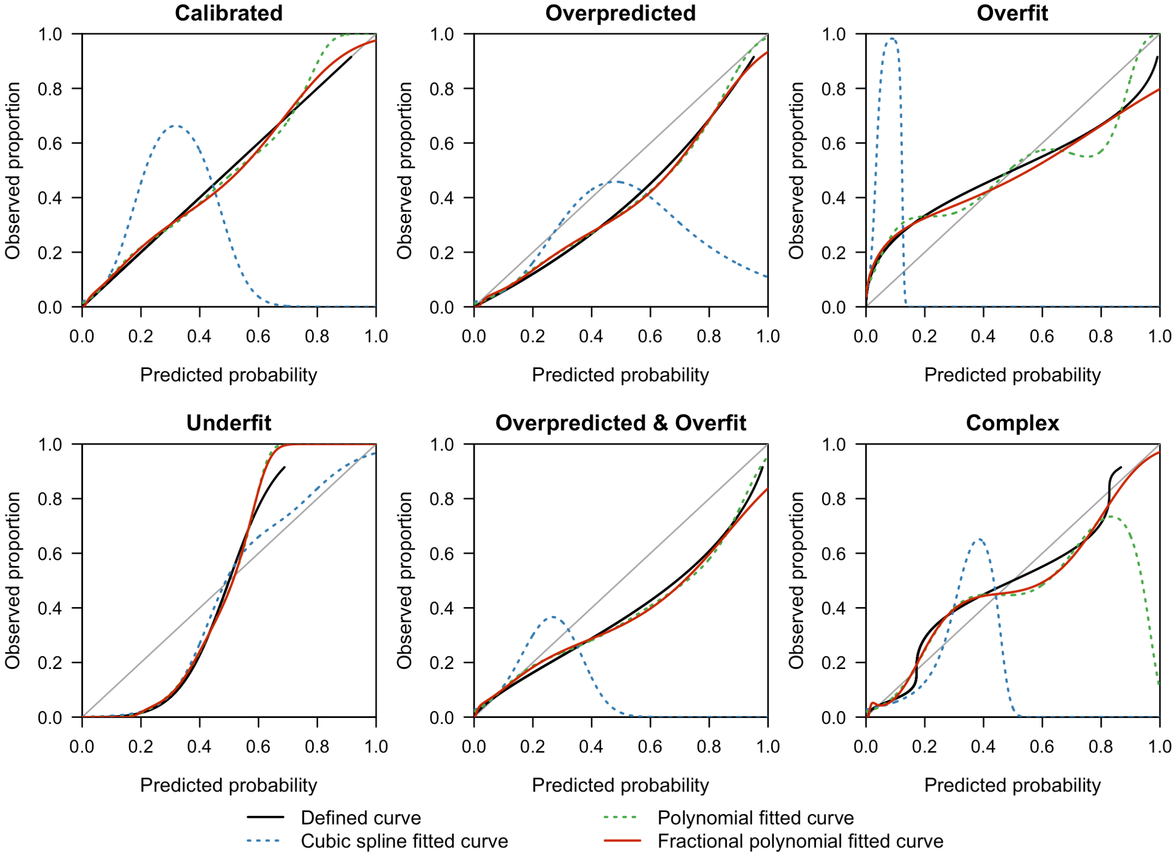
3. Once the outcome (y_t) is observed, conduct one iteration of Adam with p_t and y_t as inputs to update the coefficients of the logistic calibration curve from $\boldsymbol{\beta}_{t-1}$ to $\boldsymbol{\beta}_t$.

This approach requires parameterization of the nonlinear association between predictions and outcomes (i.e, $f(p_t)$) and initial parameter values (i.e., $\boldsymbol{\beta}_0$).

Curve parameterization

The logistic regression defining a nonlinear flexible calibration curve may take multiple forms.[39, 89-91] As examples, Figure 8 illustrates calibration curves fit with 5-knot restricted cubic splines, 5-degree polynomials, and a 5-degree fractional polynomials. We elected to define a default parameterization for dynamic calibration curves with fractional polynomials. This parameterization avoids the concern that the knots of splines may require repositioning over

Figure 8. Illustrative examples of nonlinear parameterizations of calibration curves fit to multiple forms of miscalibration, with emphasis on selected fractional polynomial parameterization.



time and better captures complex nonlinear associations than traditional polynomials (see Figure 8).[100] To select the form of fractional polynomials for our default parameterization we implemented a closed testing procedure that compares possible combinations of fractional polynomials of degree 0, ±0.5, ±1, ±2, and 3.[100, 101] By definition, fractional polynomials of degree 0 indicate transformation of a variable x into $\ln(x)$. Any repetition of a degree indicates the appropriate transformation of variable x should be multiplied by $\ln(x)$.[100] For example, if the fractional polynomial assigned to a variable x is $p =${0,2,2}, the form becomes $\ln(x) + x^2 + \ln(x)x^2$.

We evaluated fractional polynomial combinations with up to 5-degrees for several illustrative forms of miscalibration. Although the same parameterization was not selected across all cases, we observed $p =$ {0.5, 0.5, 0.5, 0.5, 0.5} generally performed well despite not matching the original parameterization of the defined curves. Figure 8 displays each form of miscalibration considered and the fit of a curve using this parametrization. Users may implement other parametrizations if desired, including alternative fractional polynomials combinations, traditional polynomials, or splines.

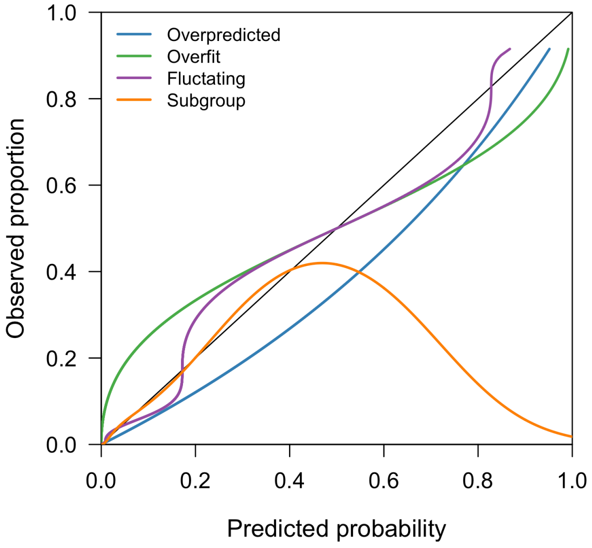
Curve initialization

Adam requires initial values for each curve parameter. Randomly generated values may suffice for some use cases; however, for dynamic calibration curves, we can provide more informative starting points. All prediction models will have been validated prior to implementation and before any subsequent ongoing assessment with dynamic calibration curves begins. We recommend leveraging information from such validation datasets to initialize the coefficients of logistic dynamic calibration curves. This can be achieved by fitting a calibration curve defined with the preferred parameterization on the validation data using general linear modeling methods. The coefficients from this model would then serve as $\boldsymbol{\beta}_o$.

Illustrative Examples

To illustrate the evolution of dynamic calibration curves as model performance drifts over time, we simulated a population in which the true probability of a binary outcomes was known and predicted probabilities followed known forms of miscalibration. To reflect the notion that most patients are low risk with a skew for relatively rare high risk patients, we generated the true probabilities from a skewed Beta(1.25, 5) distribution. For each observation, the outcome was generated by comparing true probabilities to random values generated from a uniform [0,1] distribution. If the random value was less than or equal to the assigned probability, then observation was assigned $Y = 1$, otherwise the observation was assigned $Y = 0$. Predicted probabilities were constructed by transforming the true probabilities to create overprediction (Cox intercept = -0.6), overfitting (Cox slope = 0.5), miscalibration that fluctuated over the range of probability, or miscalibration resulting from a subset of low risk observations being systematically overpredicted. The defined calibration curves resulting from these transformations are displayed in Figure 9.

Figure 9. Simulated forms of miscalibration.



From this population, we simulated 1,000 timeseries transitioning from a calibrated context to each form of miscalibration. Each series included 5,000 observations generated from a calibrated context and a subsequent 5,000 observations generated from the miscalibrated context. We recorded fitted values from the dynamic calibration curves after each observation in the timeseries. In addition to visualizing the progression of curves over the timeseries, we calculated the proportion of the true calibration curve represented by the dynamic calibration curve after each observation was processed. For each of 5,000 observations in a randomly sampled evaluation set, we estimated the fitted value of the dynamic calibration curve at each timepoint in each timeseries. Across the 1,000 simulations for each scenario, we determined whether the 95% sampling intervals of these fitted values included the true fitted value of the true defined calibration curve at the relevant timepoint. This approached allowed us to determine the proportion of the current true calibration curve represented by the current dynamic calibration curve, weighted by the distribution of predicted probabilities. This focuses our attention on the areas of the calibration curve most relevant to the data and the probably ranges we may reasonably expect data to be available for learning the calibration form.

In Figures 10-13, we plot the evolution of dynamic calibration curves using varying Adam step sizes. During the pre-drift period, curves did not diverge far from initial values for smaller step sizes, but exhibited more variability around the true association when step size increased

Figure 10. Dynamic calibration curves for timeseries abruptly transitioning from a calibrated context to an overpredicted context after 5,000 observations by Adam step size.

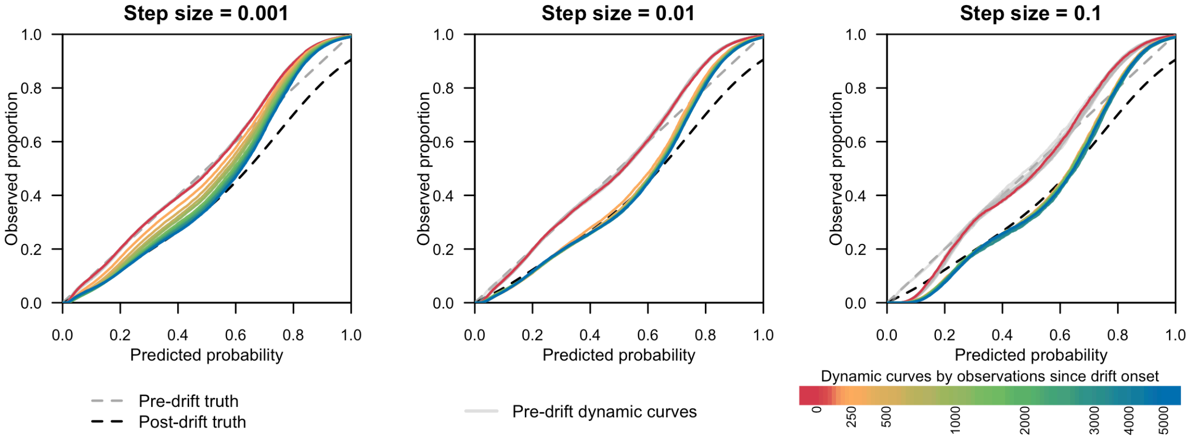


Figure 11. Dynamic calibration curves for timeseries abruptly transitioning from a calibrated context to an overfit context after 5,000 observations by Adam step size.

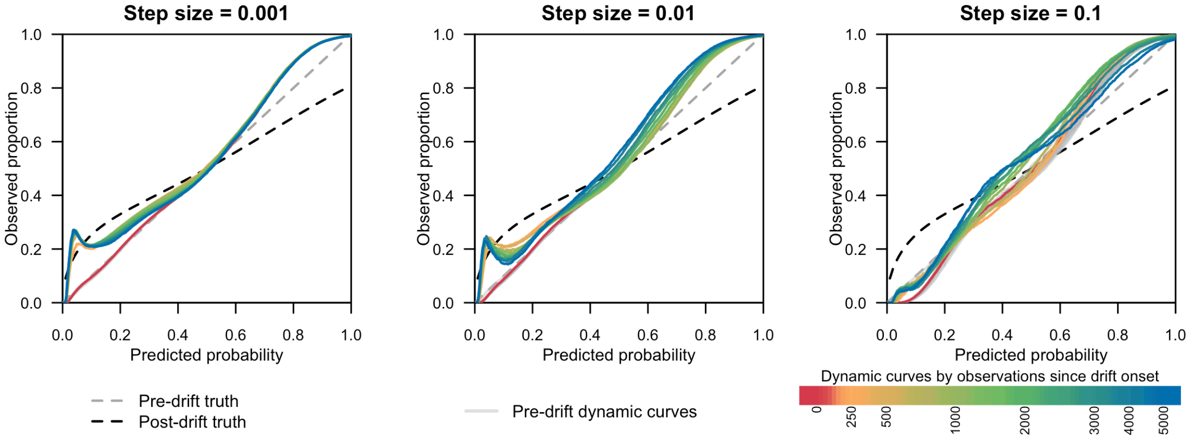


Figure 12. Dynamic calibration curves for timeseries abruptly transitioning from a calibrated context to a context with calibration fluctuating around the ideal line after 5,000 observations by Adam step size.

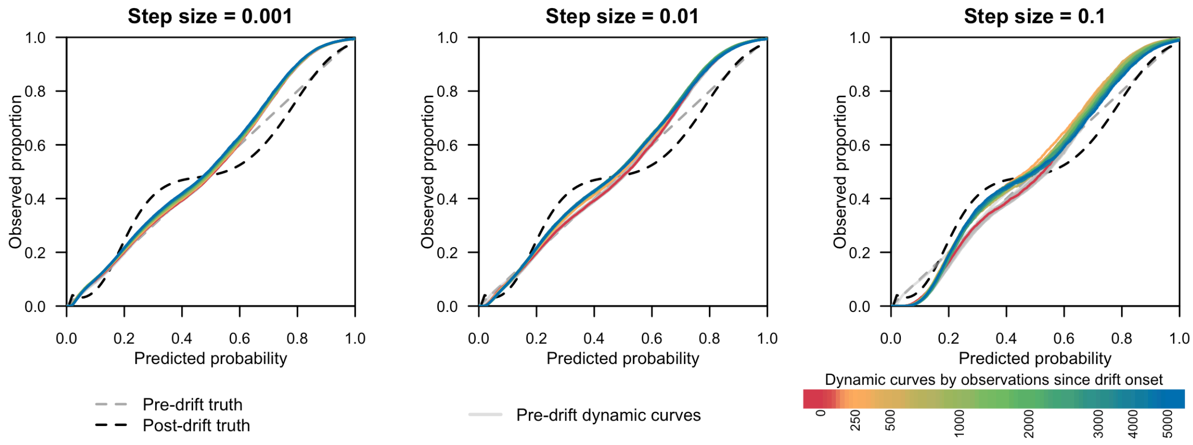
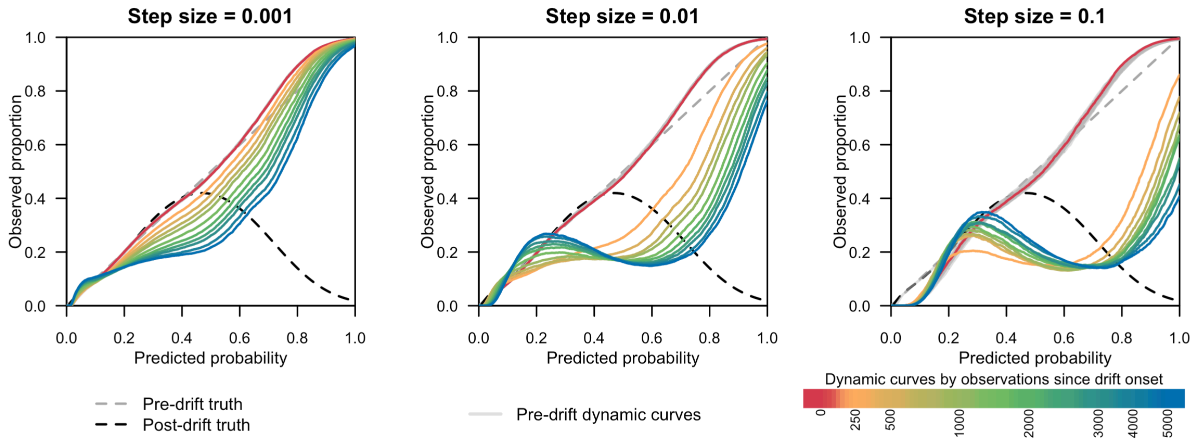


Figure 13. Dynamic calibration curves for timeseries abruptly transitioning from a calibrated context to a context in which a subgroup of low risk observations were assigned high predictions after 5,000 observations by Adam step size.



to 0.1. After drift onset, however, the curves quickly shifted in response to changes in calibration. For the default step sizes of 0.001, the sampling interval of the dynamic curves represented at least 95% of the true calibration curve within approximately 600 observations for the transition to an overpredicted setting context and within 150 observations for the transition to an overfitted setting context (see Figure 14). For overprediction, the curves illustrated the new

post-drift calibration setting except for the highest range of probability (see Figure 10). The post-drift curves for the overfit setting did not visually align with the true calibration curve (see Figure 11), they did highlight a change in performance and the sampling interval of the curve indicated the dynamic curves represented the true calibration relationship in data-dense regions (see Figure 14).

For more complex forms of post-drift miscalibration, the Adam step size impacted the performance of the dynamic calibration curves. Using the default step size of 0.001, the true calibration curve for the miscalibrated subgroup scenario was not well represented by the dynamic calibration curves (see Figure 13 and 14). However, increasing the step size to 0.01 resulted in the sampling interval of the dynamic curves representing at least 95% of the true calibration curve within approximately 1,000 observations. The dynamic calibration curves were least responsive for the transition to miscalibration that fluctuated across the range of probability, particularly when step sizes were small. As a visualization tool, these curves did not

Figure 14. Proportion of the true calibration curve represented by the dynamic calibration curve, weighted by the distribution of predicted probabilities for multiple lengths of the stable pre-drift period (step size=0.001).

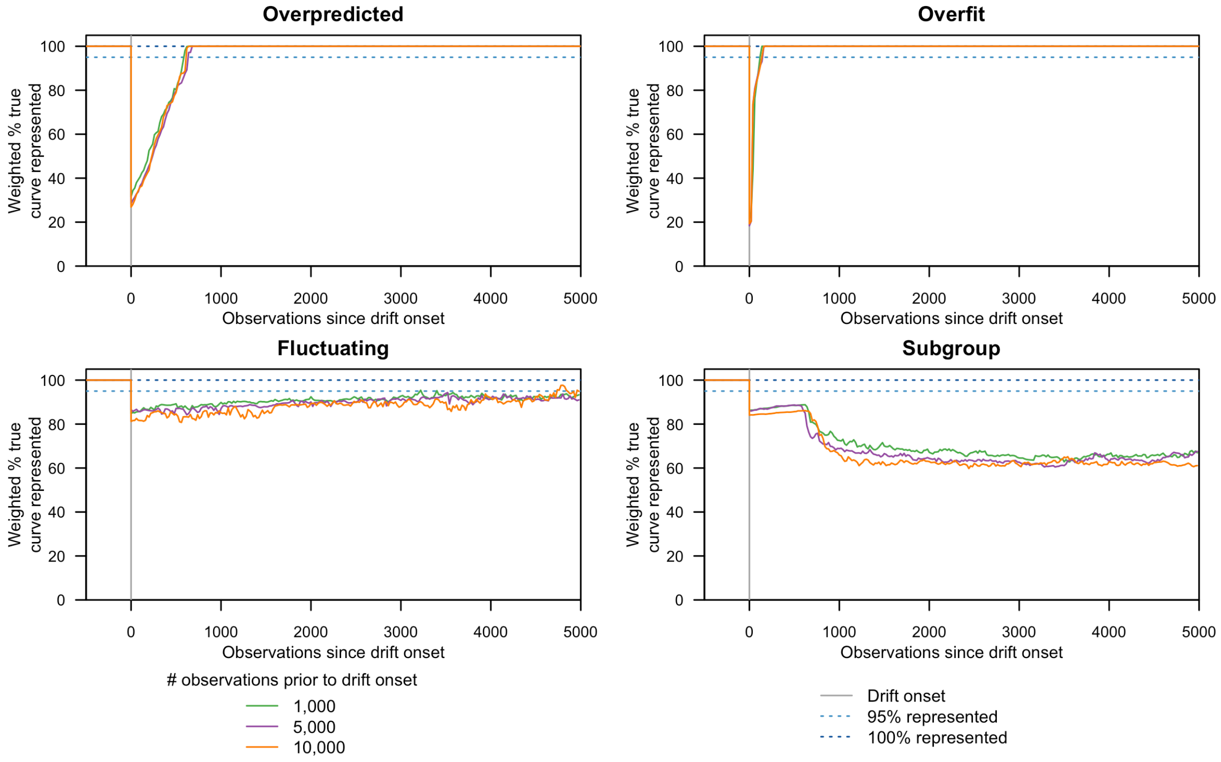
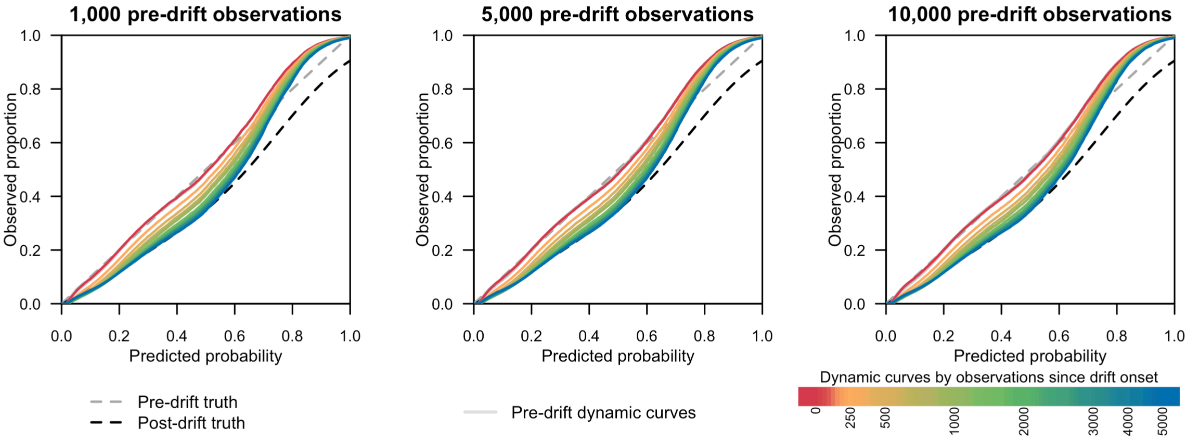


Figure 15. Impact of the length of the stable pre-drift period on the evolution of dynamic calibration curves for a transition to overprediction (Adam step size = 0.001).



appear to progress toward the post-drift true calibration curve (see Figure 12) until step size increased to 0.1. Nevertheless, even at small step sizes, the proportion of the true curve represented by the dynamic curves remained above 80% after drift onset and slowly increased to 95% over the 5,000 post-drift observations (see Figure 14). This seeming discrepancy is due to the fluctuating calibration curve not deviating far from the ideal calibration line in high density, low probability regions.

For all post-drift calibration scenarios, abbreviated or extended periods of stability prior to drift onset did not delay the response of the dynamic calibration curves. As an example, the progression of the dynamic calibration curves after an abrupt change to an overpredicted context following 1,000, 5,000 or 10,000 calibrated observations are shown in Figure 15.

Discussion

Utilizing continuous learning, we are able to maintain an ongoing assessment of model calibration. Rather than repeatedly refitting calibration curves with batches of recent data, dynamic calibration curves incorporate information from new observations into previously learned associations within the logistic calibration model. Using the Adam method for adaptive learning, the calibration curves shift in response to changes in model performance among new observations.

This approach to continuous calibration assessment has several advantages over static calibration curves for applications applying prediction models in non-stationary environments. Our method avoids the need to define appropriate batches of recent data for constructing calibration curves and, therefore, does not requires users to anticipate the pace of performance drift. Dynamic calibration curves also reduce computational requirements when calibration curves are desired for each new observation, especially in the case of high volume and high velocity data streams. Our method is generalizable to the variety of prediction models based on diverse learning algorithms and can support customizable curve parameterizations.

The implementation of dynamic calibration curves presented here has limitations that warrant further consideration and research. The dynamic curves shifted to highlight changes in performance, but did not necessarily capture the defined forms of miscalibration. This was particularly true for complex miscalibration and regions of the probability range with sparse data. The step size parameter of the Adam algorithm was influential in how well the dynamic curves represented complex miscalibration. Further investigation could provide guidance on tuning this parameter. In addition, alternative continuous learning approaches, such as dynamic logistic regression,[62] should also be considered and may improve the accuracy of post-drift curves.

Conclusion

We described a method to continuously monitor model calibration on streaming data using dynamic calibration curves updated as data accumulates . As opposed to periodic model validations, this method can reveal performance drift as it occurs. With additional tuning, dynamic calibration curves could be used to efficiently calculate observation-level stringent calibration metrics in real-time or visualize evolving calibration patterns. While providing insight into calibration over time, dynamic calibration curves do not indicate significant change in performance or alert users to performance drift. In the following chapter, however, we explore how methods for dynamic calibration curves can support methods aimed at identifying significant drift.

CHAPTER 5

A DRIFT DETECTION APPROACH TO TRIGGER MODEL UPDATING

Scheduled model updating is a common scheme in model maintenance protocols.[27, 31, 32] This approach, although simplifying planning, requires users to prespecify an anticipated rate of model deterioration. In practice, the frequency of scheduled updates may not align well with patterns and timing of changes in clinical populations or environments. For example, in a case study presented in Chapter 7, annually refitting a model for 30-day mortality model did not result in performance gains beyond that achieved by less frequent updating. Furthermore, the learning algorithm underlying prediction models influences the magnitude and speed of performance drift, even among models applied to the same population.[18, 19] Scheduled updating protocols may thus be inefficient during periods of relative model stability or allow for interim periods of uncorrected performance drift during phases of more rapid population shifts.

As an alternative to scheduled updating points, we present a data-driven approach to initiate model updating. Building on drift detection methods from the classification modeling literature[68] and leveraging the dynamic calibration curves we developed in Chapter 4, we propose and evaluate a calibration drift detection system that seeks to identify deterioration in performance and alert users when a model may require attention. Our detector is intentionally designed to be applicable regardless of the underlying learning algorithm, making the approach relevant for systems managing suites of diverse prediction models. To further support model managers, we designed the calibration drift detection system to provide actionable alerts that also return information on a window of recent data that may be appropriate for updating, if required.

Designing a Calibration Drift Detection System

Overview

Concept drift detection is an established area of research providing methods to identify changes in the performance of prospectively applied prediction models.[68] Common drift detection algorithms are model-independent[68] and can be incorporated into a model surveillance system regardless of the learning algorithm underlying the models being tracked. However, much drift detection research has focused on identifying changes in misclassification rates. This focus on discrimination rather than calibration does not provide a sufficiently nuanced assessment of model performance for many clinical use cases.[20, 39-42, 46] Statistical process control charts, which are more flexible in the error metrics they can track, have been applied retrospectively to evaluate calibration drift[29, 85] rather than prospectively to assess calibration drift in real-time. See Chapter 2 for more background on the state of concept drift detection and limitations of existing algorithms.

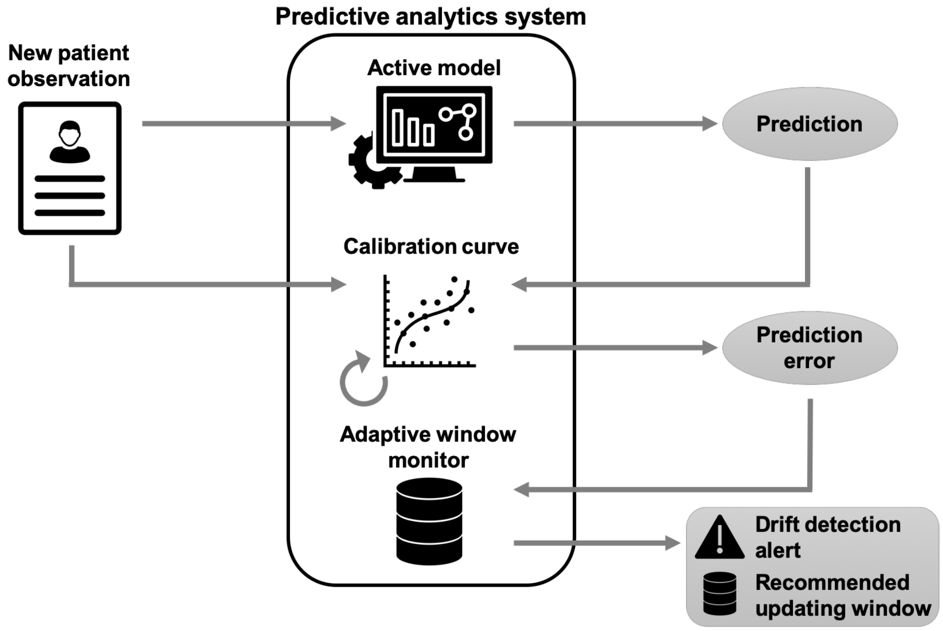
We sought to develop a new calibration-focused drift detector by building on prior research in the concept drift detection space. Our calibration drift detection system is an online performance tracking method that alerts users in real-time when performance drift is identified and provides guidance on what set of recent data might be appropriate for responding to that drift. We implement the adaptive window approach[34] to monitor mean predictive error using a detailed, up-to-date assessment of performance based on dynamic calibration curves. When a significant change in predictive error is observed, the detector alerts users to the presence of drift and returns a window of recent observations that appear to have been generated after the change point.

For an overview of the flow of data in our calibration drift detection approach, please refer to Figure 16, which represents a portion of our overall conceptual model. A prediction using the current, active prediction model is generated as new patient data becomes available. The error of this prediction is estimated from the dynamic calibration curve updated as of the previous observation. This dynamic calibration curve is further updated once the current observation's outcome becomes available in order to prepare for the arrival of subsequent patient observations. The current error value is submitted to the adaptive windowing monitor which checks for a change in performance (as described below), triggering an alert when appropriate. In addition to alerting the user to drift, the detector returns a window of recent data with an internally consistent error distribution, which may be a good candidate for use in any subsequent updating process. Although our detector is designed to motivate and support model updating, for the purposes of this chapter, we focus on detecting calibration drift and leave the response to an alert to later discussion.

The Adaptive Window Method

We leveraged the adaptive windowing (Adwin)[34] approach to drift detection (see Figure 17 for details). Adwin aims to maintain a window (W) of recent data which appears to be generated from a stable generating process. As new observations arrive, they are added to the

Figure 16. Calibration drift detection schematic.



head of the current window. Sliding divisions of W into a pair of subwindows (i.e., W_1 containing newer data and W_0 containing older data) allow for a sequence of comparison between a growing set of older data and a shrinking set of newer data. If a significant difference between a pair of subwindows is discovered, Adwin shrinks the current window by dropping the older data (W_0). This continues until no subwindow differences remain. In this way, the current window only retains data that appears to be from a single, current generating process. Anytime the window shrinks, the process has identified drift and the window of remaining data may be from a stable population and thus appropriate for updating the model, as needed, to restore performance.

Adaptive windowing has several advantages that are well-suited for our calibration drift detection system. First, the algorithm is designed with streaming data in mind. Observations are processed individually, avoiding the need to make assumptions regarding appropriate batch sizes as required by methods such as common statistical process control charts. As new observations arrive, the algorithm immediately checks for drift and integrates the observation into the current window, storing sufficient statistics about W rather than the entire stream of data.[34] This approach allows adaptive windowing to process new observations quickly and conserve memory in high data volume settings. Extensions to the original adaptive windowing algorithm also support parallel processing, minimize computational requirements, and account for delays between prediction generation and outcome observation.[34, 102] Second, although originally designed and described in a misclassification context, adaptive windowing only requires that the error metric be bounded.[34] Several model accuracy and calibration metrics could thus be tracked by an adaptive window implementation. Additionally, adaptive windowing evaluates the current state of the input data process based on the received observations rather than requiring users to prespecify an expected in-control (i.e., stable) distribution of the data.[34]

Figure 17. Details of the adaptive windowing method defined in Bifet and Gavaldà (2007).[34]

Adwin algorithm:

```
Initialize window $W$
**For** each $t > 0$
    **Do** $W \leftarrow W \cup \{x_t\}$ (i.e., add new observations to head of $W$)
        **Repeat** Drop elements from the tail of $W$
            **Until** $|\hat{\mu}_{W_0} - \hat{\mu}_{W_1}| \leq \epsilon_{cut}$ holds
                for every split of $W$ into $W = W_0 \cdot W_1$
```

Using a normal approximation, $\epsilon_{cut} = \sqrt{\frac{2}{m} \cdot \sigma_W^2 \cdot \ln\frac{2}{\delta'}} + \frac{2}{3m}\ln\frac{2}{\delta'}$ where

m is the harmonic mean of n_0 and n_1 (i.e., the size of subwindows W_0 and W_1)

$n = n_0 + n_1$ and $\delta' = \frac{\delta}{\ln{(n)}}$

Requirements:

- x_t is a bounded error metric scaled to the $[0,1]$ interval
- x_t are independent for each t

Parameters:

- $\delta \in (0,1)$

Implications:

- Drift detected when W shrinks
- Whether drift is detected or not, once processing finishes for each t, retained W will be composed of data from a stable generating process

Specification of a Calibration Drift Detection System

For our drift detection system, we utilize adaptive windowing to detect changes in a stringent measure of calibration based on flexible, nonlinear calibration curves. We selected a curve-based metric to align our detector with the clinical decision-making context. Calibration metrics based on flexible calibration curves ensure models have a net benefit greater than or equal to treat-all or treat-none strategies, thus ensuring predictions are nonharmful to clinical decision-making.[39] The fitted value of a calibration curve at a given predicted probability provides an estimate of the observed probability of the outcome among patients with similar predicted risk.[89] For each observation, we thus define the predictive error (i.e., x_t in the adaptive window definition presented in Figure 17) as the absolute difference between the predicted probability (p_t) and the fitted value of the calibration curve ($\hat{p}_t$) (see Figure 18).[89] Leveraging the dynamic calibration curves developed in the previous chapter, we base $\hat{p}_t$ on an up-to-date calibration curve without rebuilding the curve for each new observation. The data evaluated by the adaptive window-based calibration drift detection system are thus defined as follows for the observation at time t:

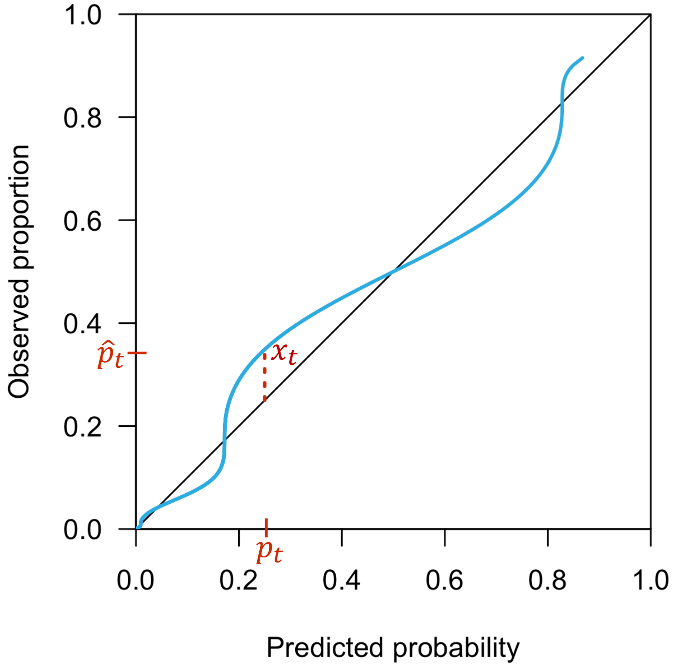
$$x_t = |\hat{p}_t - p_t|$$

where p_t is generated from the active prediction model and $\hat{p}_t$ is estimated using the most recent coefficients of the dynamic calibration curve ($\boldsymbol{\beta}_{t-1}$) and the user-defined nonlinear expansion of the predicted probability ($f(p_t)$).

$$\hat{p}_t = f(p_t)\boldsymbol{\beta}_{t-1}$$

This metric is bounded on the [0,1] interval as required by the adaptive windowing algorithm and is interpretable as the absolute distance between the calibration curve and the perfect calibration line. In addition to evaluating x_t with the adaptive windowing approach, once x_t has been calculated and the outcome at time t recorded, we update the dynamic calibration curve prior to the arrival of an observation at time $t + 1$. See Figure 16 above for the overview of this process.

Figure 18. Illustrative example of curve-based predictive error. Red text highlights parameter values for calculating curve-based predictive error. The red dotted line illustrates the metric's (x_t) interpretation as the magnitude of deviation of the curve from the ideal at the current prediction (p_t).



Parameterization of the Adaptive Windowing Method

In addition to defining the calibration metric considered by the adaptive windowing algorithm, we must specify an error tolerance for detecting changes in performance. We are interested in a detector that balances the probability of false positives (i.e., detecting drift during periods of stable model performance) and the probability of false negatives (i.e., not detecting drift during periods of performance deterioration). The δ parameter in the adaptive windowing algorithm provides control over these probabilities. While statistical theory, detailed in the original adaptive windowing study, provides for theoretical bounds on the false positive and false negative rates based on δ, Bifet and Gavaldà acknowledged false positive rates were substantially lower than theory would suggest.[34] For the evaluations of our calibration drift detection system, we established $\delta = 0.05$, which sets the upper bounds of the false positive rate at the common Type I error threshold of 5%. For sensitivity, we repeated our evaluations with $\delta = 0.075$ and $\delta = 0.1$. See Appendix A for detailed results. In addition, we conducted simulations of stable model performance over extended timeseries to provide some guidance on reasonable ranges of δ in a variety of contexts (see Appendix B).

Evaluating the Adaptive Windowing Calibration Drift Detection System

We conducted simulation studies to evaluate the performance properties of our calibration drift detection system along several important dimension, including:

1. *False positives* – How frequently is performance drift detected incorrectly?
2. *False negatives* – How frequently does performance drift go undetected?
3. *Time to detection* – How long is the delay between the start of performance drift and detection?
4. *Post-detection window composition* – After detection, does the retained window include data relevant for updating?

Simulated Performance Drift Patterns

We simulated timeseries in which observations were initially generated by a calibrated model and over time observations shifted to being generated by one of 10 miscalibrated models. These timeseries were sampled from populations in which the true probability of a binary outcome was known and predicted probabilities followed known forms of miscalibration. To reflect the notion that many risk modeling applications have predictions that are clustered asymmetrically in low risk regions, and that risk models operating with clustered high risk observations would present similar challenges, we generated the true probabilities from a skewed Beta(1.25, 5) distribution, which enriched for low probability predictions. For each observation, the outcome was generated by comparing true probabilities to random values generated from a uniform [0,1] distribution. If the random value was less than or equal to the assigned probability, the observation was assigned $Y = 1$, otherwise the observation was assigned $Y = 0$. Predicted probabilities were constructed by transforming the true probabilities to create multiple calibration patterns, including over/underprediction, over/underfitting, combined overfitting and overprediction, miscalibration that fluctuated over the range of probability, and miscalibration resulting from a subgroup of low risk observations being substantially overpredicted. The predefined calibration curves resulting from these

Figure 19. Simulated patterns of miscalibration.

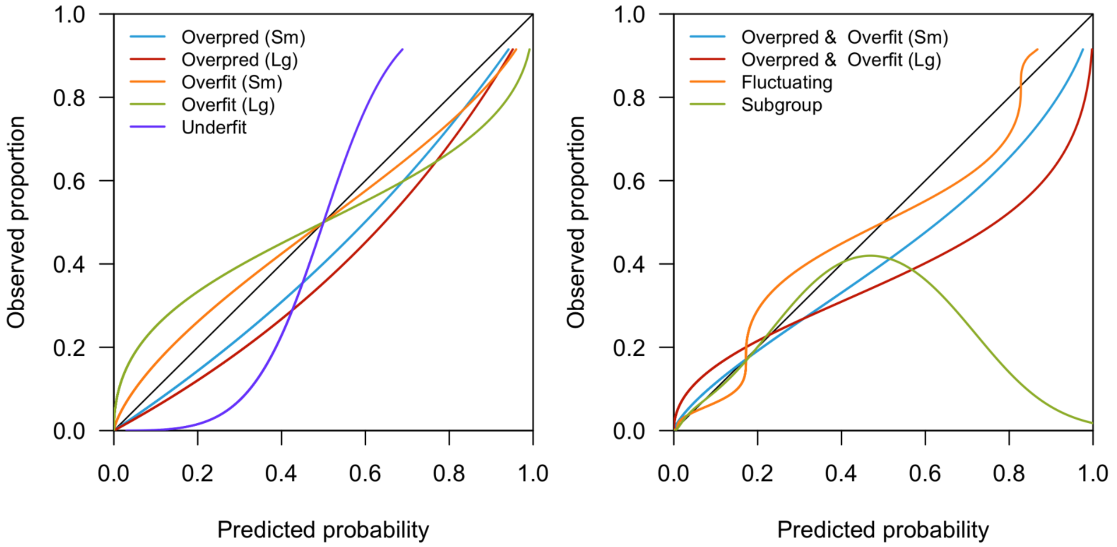
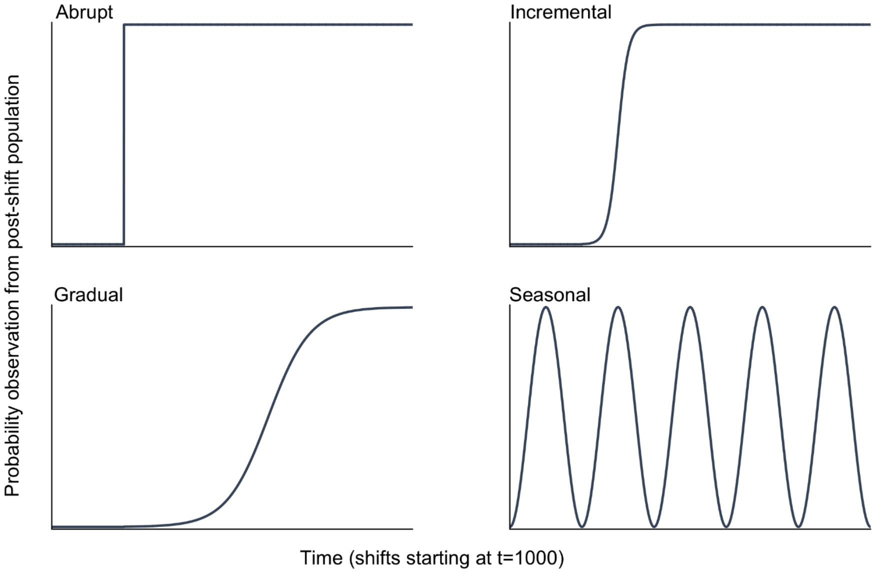


Figure 20. Temporal transition patterns for simulated timeseries.



transformations are displayed in Figure 19. See Appendix C for equations defining each transformation. An extreme case of random predictions was defined as well. Each time series included 5,000 ordered observations. The speed at which observations transitioned from calibrated to miscalibrated took four forms (see Figure 20) – an abrupt transition, a rapid transition over a short period, a gradual transition over an extended period, and a recurrent/seasonal transition in which observations transitioned back and forth between two calibration settings. With the exception of the recurrent/seasonal case, the first 1,000 observations in each series were generated from the population of calibrated predictions and temporal transitions began immediately following this stable period. The rapid pattern transitioned to miscalibrated predictions over 1,000 observations; the gradual pattern transitioned over 4,000 observations, only completing the transition at the end of the series. In the case of recurrent/seasonal temporal transitions, timeseries moved from calibrated to miscalibrated predictions and back to calibrated predictions every 1,000 observations.

In addition to defining the point of drift onset at *t*=1,000, we sought to determine a change point at which the pre- and post-drift populations were significantly different along each temporal transition to each form of miscalibration. For incrementally increasing mixing rates, we compared the mean predictive error between 1,000 randomly selected calibrated predictions and 1,000 observations randomly drawn from a mixture of calibrated and miscalibrated predictions. We identified the minimum mixing rate at which a significant difference ($p<0.05$) was recorded between the fully calibrated population and partially miscalibrated population. This process identified a significant mixing rate for each of the ten forms of miscalibration (see Table 3). We defined change points for each temporal transition pattern as the observation at which this mixing rate occurred.

Table 3. Mixing rates defining change points for transitions to each form of miscalibration.

Form of miscalibration	Mixing rate
Overpredicted (small)	0.425
Overpredicted (large)	0.275
Overfit (small)	0.5
Overfit (large)	0.275
Underfit	0.1
Overpredicted & overfit (small)	0.775
Overpredicted & overfit (large)	0.425
Fluctuating	0.8
Subgroup	0.15
Random	0.25

For each combination of temporal transition pattern and calibration change, we applied the calibration drift detection system to 1,000 timeseries as defined above. Initial values for each timeseries' dynamic calibration curve were estimated from a general linear model fit to a random sample of 500 calibrated predictions. Applying our calibration drift detection system to each timeseries, we documented whether and when drift was detected. This allowed us to examine the questions above with the following metrics:

- *False positives* – percent of iterations with a detection occurring prior to the start of the drift at observation $t = 1{,}000$
- *False negatives* – percent of iterations in which the detector failed to identify the drift prior to the end of the series
- *Time to detection* – number of post-drift observations prior to drift being detected
- *Lag to detection* – number of observations between the change point and drift detection
- *Post-detection window composition*
    - Relevancy of returned window – percent of detections including any pre-drift observations (i.e., $t < 1{,}000$) in the returned data window
    - Contamination of returned window – percent of data in returned windows occurring prior to drift onset

We further recorded the smoothed mean error over time for each timeseries using the exponentially weighted moving average approach.[73] This allowed us to examine how our error metric evolved over time in each setting and how this related to the point at which drift was detected.

False Positives

The rate of false positive detections during the stable pre-drift period are presented in Table 4. False positives were rare, with the rates well below the 5% threshold our δ might suggest. As δ increased to 0.1, the frequency of false positives increased but generally remained below 1% (see Appendix A). These findings are consistent with prior studies.[34] We note false positives were not relevant to the recurrent/seasonal transitions as this temporal pattern did not include an initial period of stability.

Table 4. Frequency of false positive (FP) and false negative (FN) detections by temporal transition speed and post-drift calibration setting.

Post-drift calibration setting	Abrupt		Rapid		Gradual		Recurrent/ seasonal	
	% FP	**% FN**	**% FP**	**% FN**	**% FP**	**% FN**	**% FP**	**% FN**
Overpredicted (small)	0.2	0.8	0.1	0.6	0.3	2.5	-	30.6
Overpredicted (large)	0.2	0	0.3	0	0.1	0	-	7.7
Overfit (small)	0.3	0.6	0.4	1.1	0.2	1.7	-	14.6
Overfit (large)	0.4	0	0.2	0	0.3	0	-	3.1
Underfit	0.8	0	0.3	0	0.1	0	-	0
Overpredicted & overfit (small)	0.2	16.2	0.3	14.4	0.4	21.6	-	59.5
Overpredicted & overfit (large)	0.3	0.8	0.4	0.7	0.1	0.7	-	15.5
Fluctuating	0.1	39.3	0.3	37.9	0.3	51.6	-	57.2
Subgroup	0.2	0	0.3	0	0.2	0	-	2.5
Random	0	0	0	0.1	0	0	-	4.0

False Negatives

For most drift scenarios, false negatives, missed opportunities to detect calibration drift, were infrequent, with rates under 3% (see Table 4). For all temporal transition patterns, false negatives were common among timeseries transitioning to the models with miscalibration that fluctuated around the ideal line and with combined modest overprediction and overfitting. These forms of miscalibration did not deviate far from calibration in the more densely populated low risk range. As a result, the magnitudes of change in calibration over time were small (see Figures 23 and 24). False negatives were most common under the recurrent/seasonal transition pattern. Drifts toward a relatively small magnitude of miscalibration were most susceptible to false negatives under this transitional pattern. False negative rate declined as δ increased (see Appendix A).

Time to Detection

Figures 21-25 display the distribution of detection points against the temporal error pattern and change point for each combination of temporal transition and post-drift miscalibration. Detection of calibration drift was fastest, in terms of the number of observations processed between drift onset and detection, for the abrupt transition to the more overfit setting (median time to detection=231). Time to detection was longest for the recurrent/seasonal transition to the models with miscalibration that fluctuated around the ideal line (median time to detection=3,246) and with combined modest overprediction and overfitting (median time to detection=3,297). Drifts toward these two forms of miscalibration consistently delayed detection, with more than 1,500 post-drift observations typically required for detections under all temporal transition patterns.

The delay between drift onset and drift detection varied by speed of temporal transition and degree of miscalibration in the post-drift setting (see Figure 26). Time to detection increased as the speed of transition slowed from abrupt to rapid to gradual. As highlighted in Figures 26 by the two variations of overfit models, the delay in and variability of time to detection increased as the magnitude of miscalibration decreased and the speed of transition slowed. For rapid transitions occurring over 1,000 observations, drift was detected during the transition period for those post-drift settings with the largest magnitudes of miscalibration, but not detected until the transition was complete in the case of modest overprediction, modest overfitting, and fluctuating miscalibration. Recurrent/seasonal transitions lead in the most variability in detection timing. In most cases, recurrent/seasonal drift required multiple cycles before detection. However, drift involving the more substantially overfit, the underfit, and the random models was typically detected before the first cycle of the recurrent/seasonal pattern was completed (i.e, median detection time <1,000 observations).

Lags in detection, in terms of the number of observations processed between the change point and drift detection, were shortest for the gradual change toward fluctuating miscalibration (median lag to detection=223) and abrupt change to the more overfit model (median lag to detection=231). Rapid changes toward fluctuating and combined modest overprediction and overfitting resulted in the longest lag in detection (median lag to detection=2,954). Within each post-drift calibration setting, lags were generally more consistent than time to detection across temporal transition speeds (see Figure 27). This is highlighted by transitions to the more overfit model in which the median lags ranged from 231 to 282 for abrupt

Figure 21. Error distribution and detection characteristics for timeseries transitioning between the calibrated model and models with varying magnitudes of overprediction.

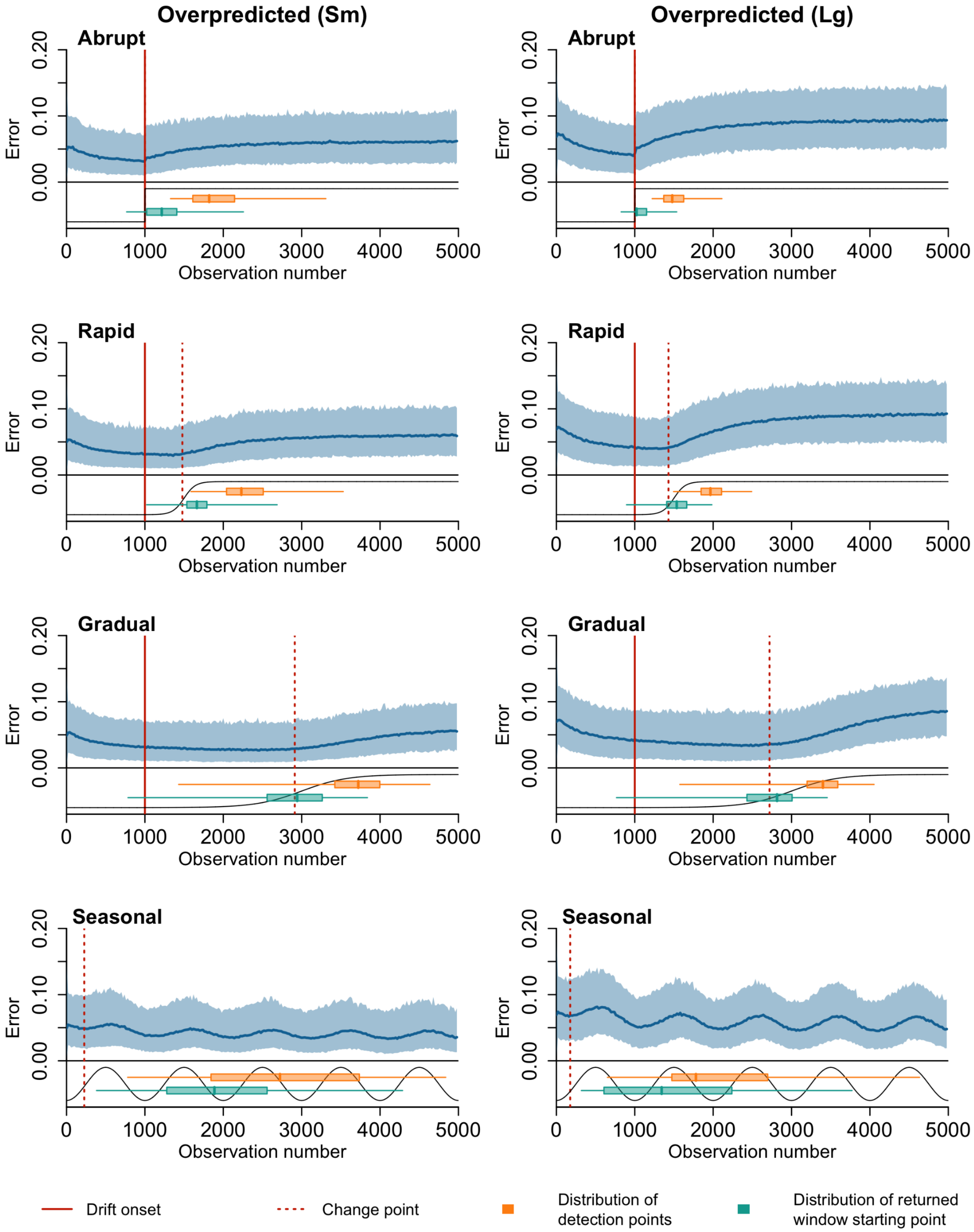


Figure 22. Error distribution and detection characteristics for timeseries transitioning between the calibrated model and models with varying magnitudes of overfitting.

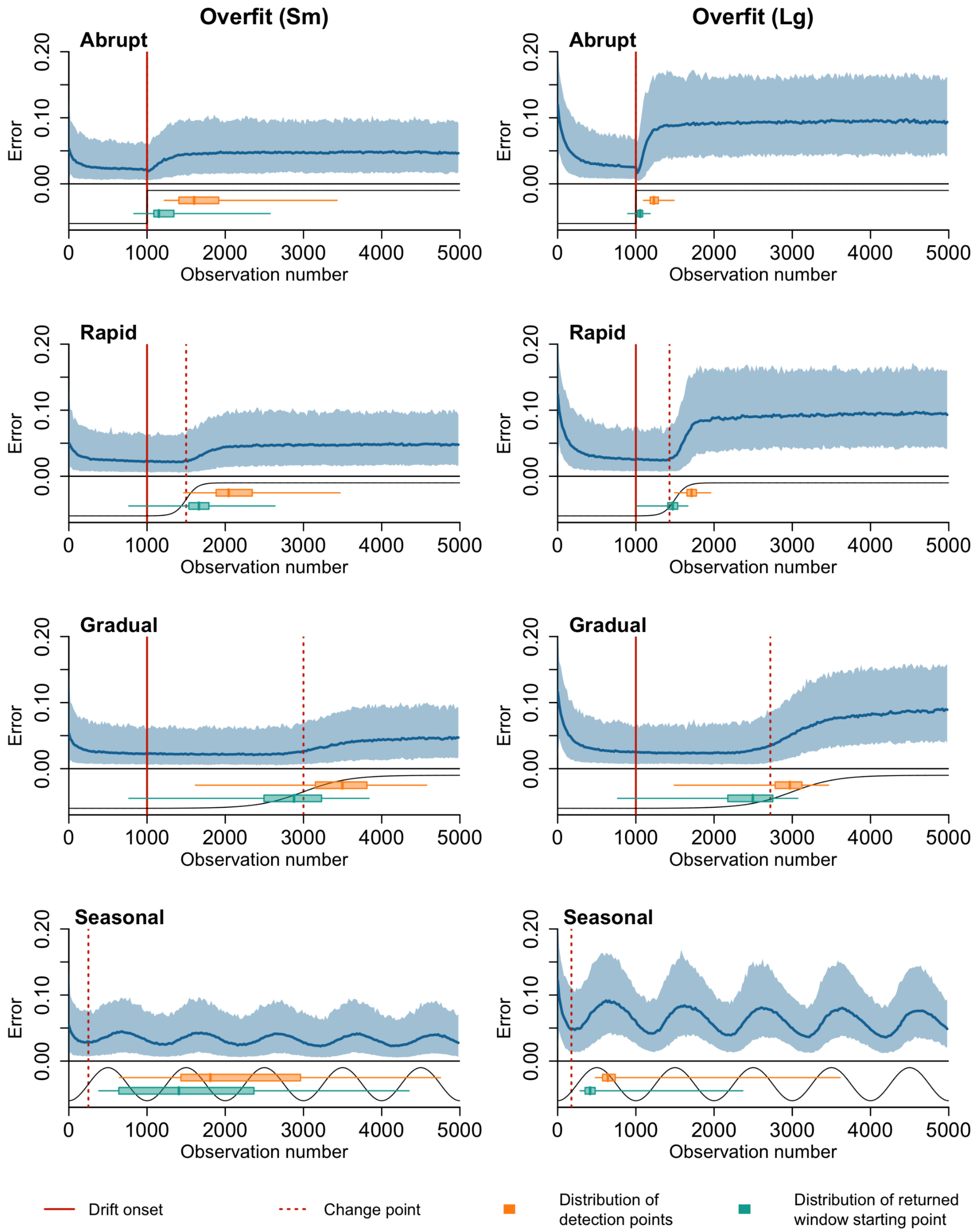


Figure 23. Error distribution and detection characteristics for timeseries transitioning between the calibrated model and models with varying magnitudes of combined overprediction and overfitting.

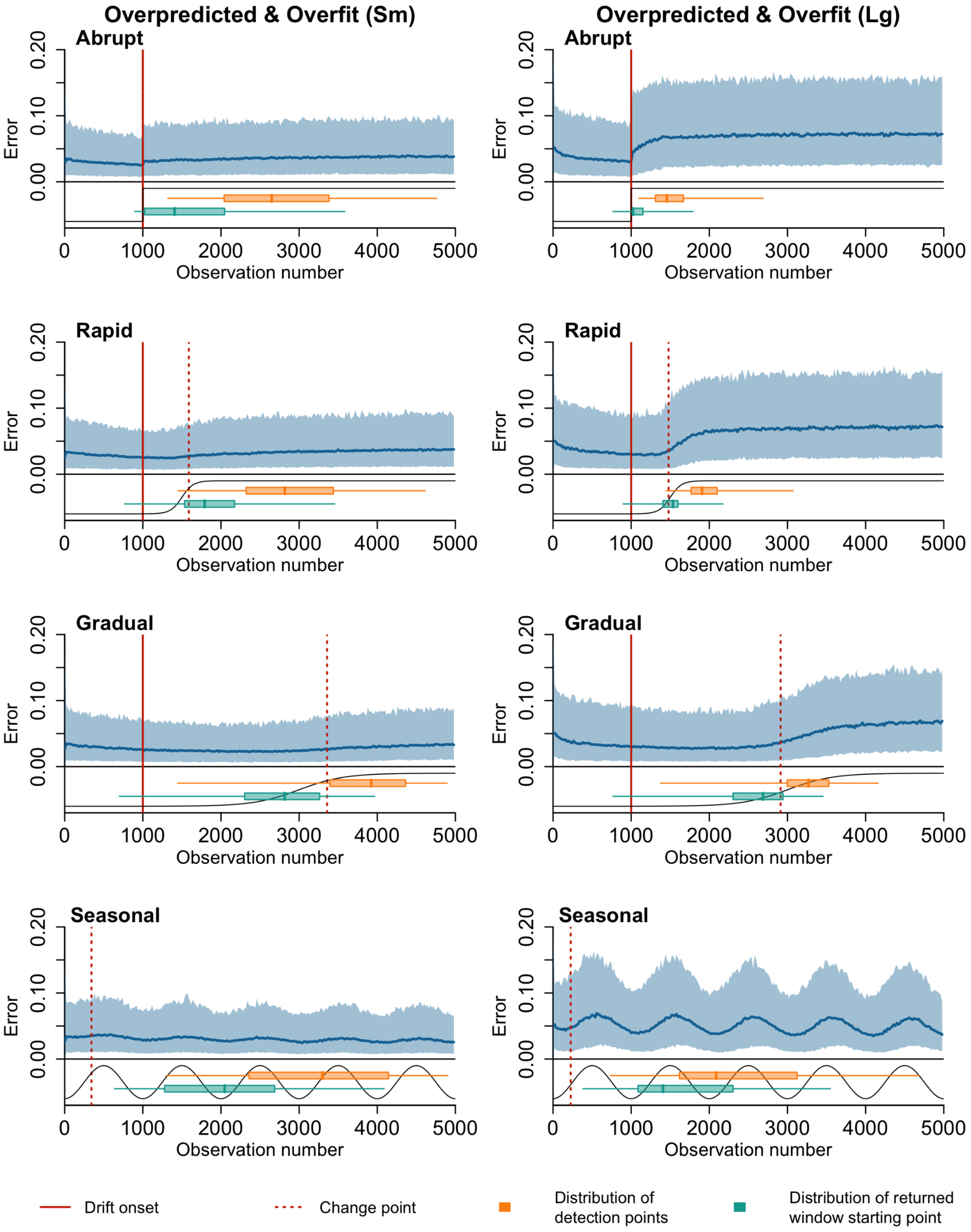


Figure 24. Error distribution and detection characteristics for timeseries transitioning between the calibrated model and a model with (left) miscalibration that fluctuated around the ideal line or (right) a subgroup of low risk observations was substantially overpredicted.

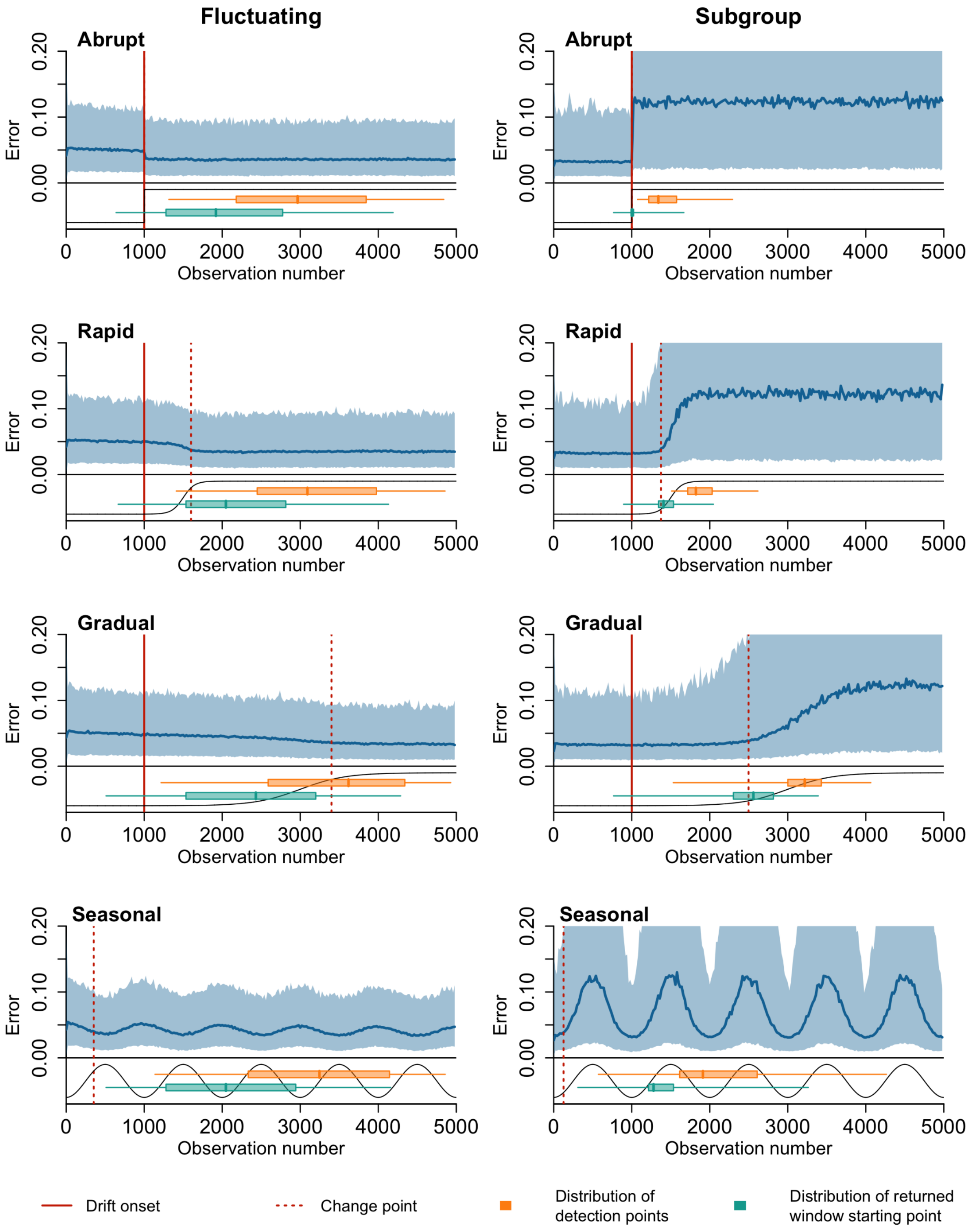


Figure 25. Error distribution and detection characteristics for timeseries transitioning between the calibrated model and (left) an underpredicted model or (right) a model with random predictions.

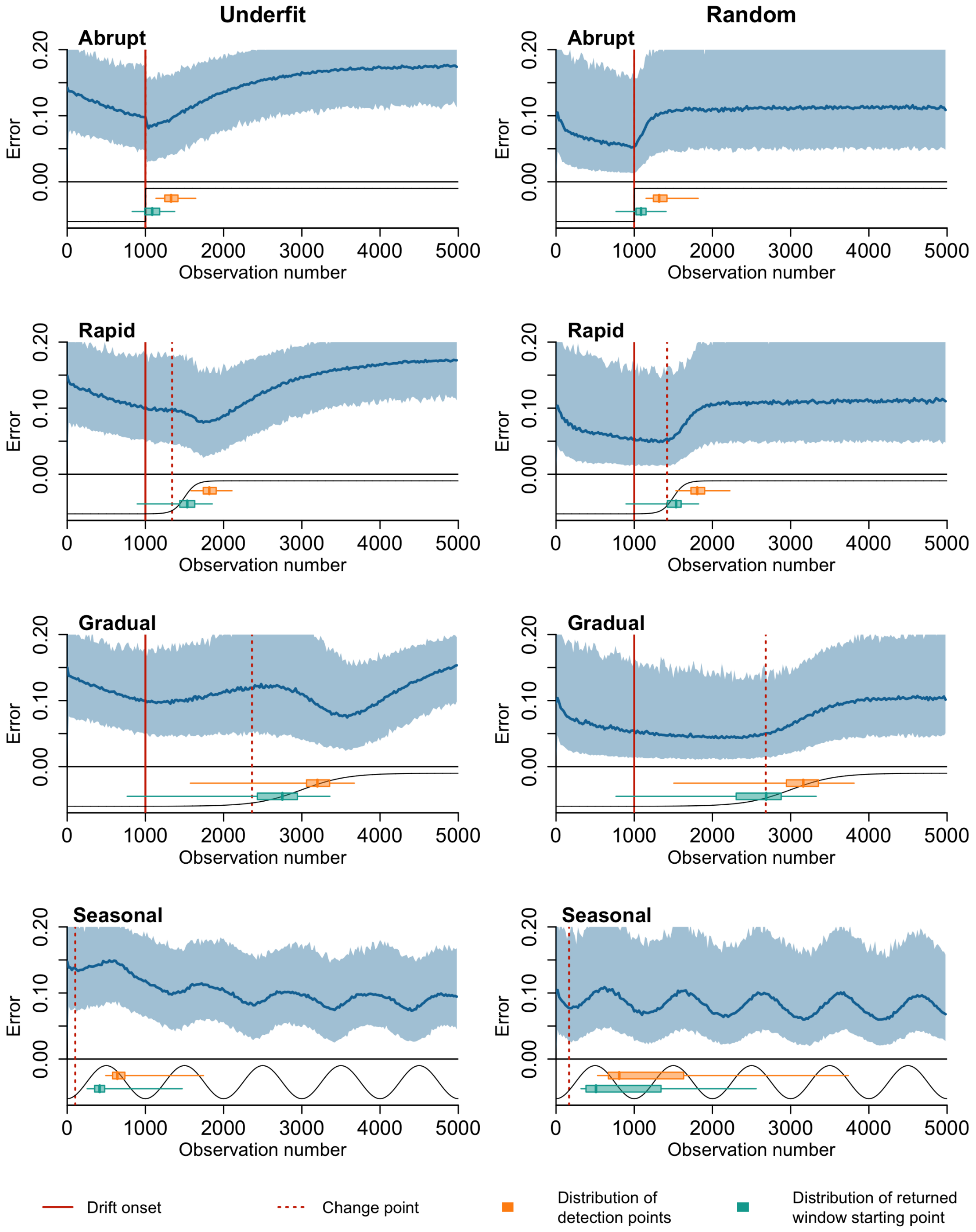


Figure 26. Time to detection as number of observations from drift onset to detection by speed and form of change.

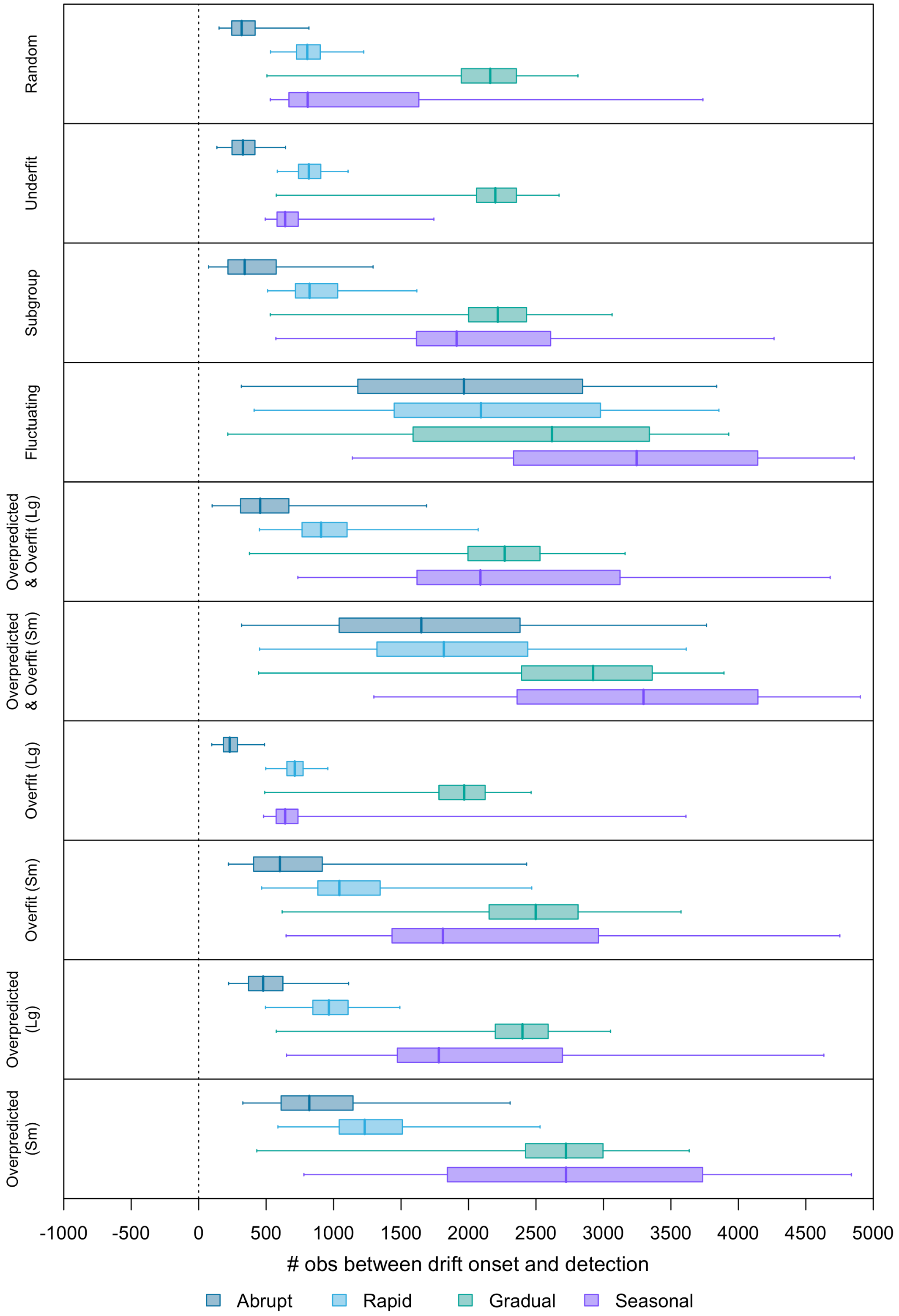
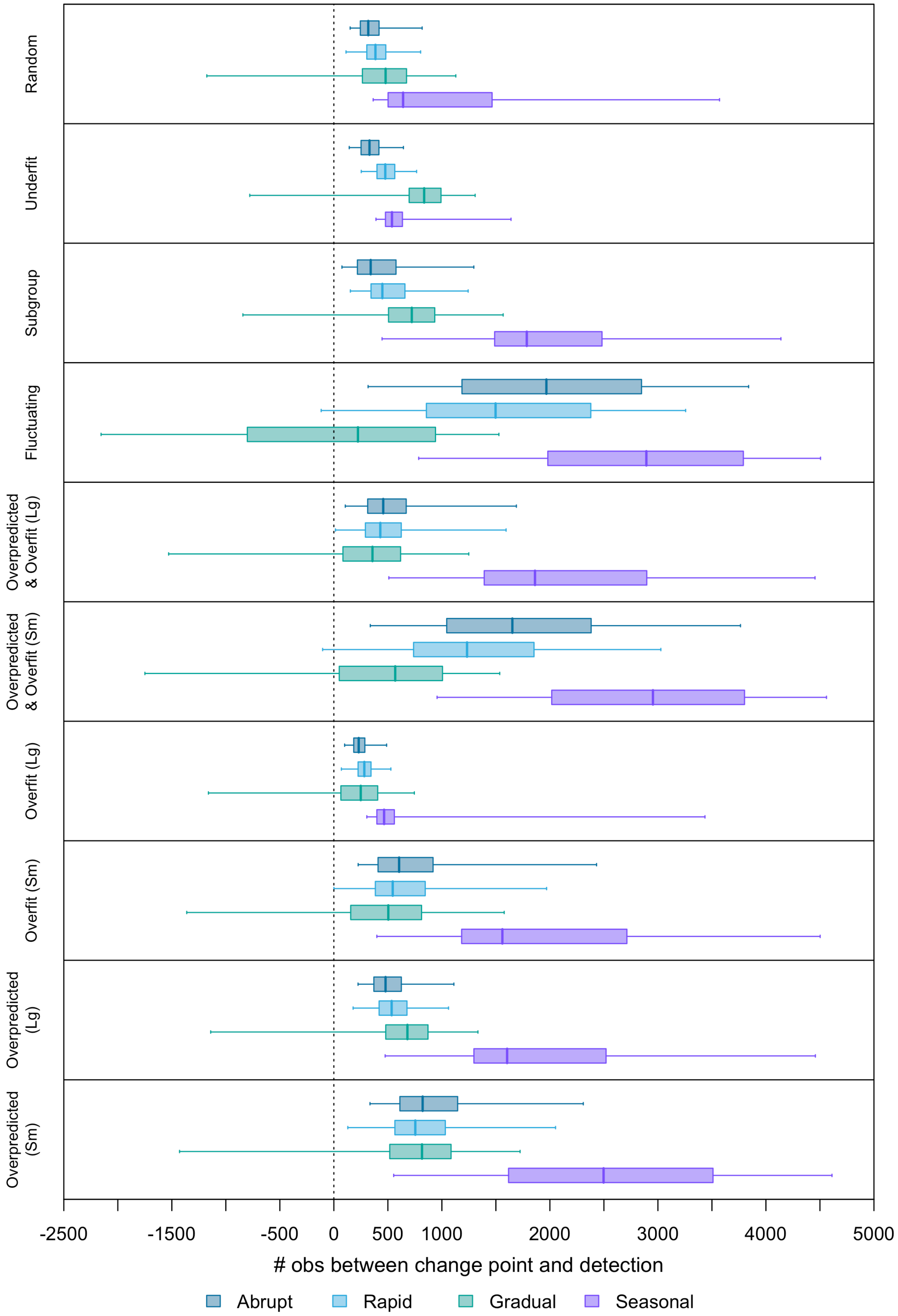


Figure 27. Lag to detection as number of observations from change point to detection by speed and form of change.



through gradual transitions. Corresponding median times to detection ranged from 231 to 1,969. In contrast to this pattern, transitions to fluctuating miscalibration and combined modest overprediction and overfitting exhibited longer lags to detection for faster transitions. In several cases, gradual transitions were detected prior to the identified change point. This was most common for the gradual transition towards fluctuating miscalibration and may reflect an accumulation of performance change over the extended period of transition.

Post-Detection Window Composition

The size of the data window returned by the calibration drift detector and the origins of the observations in the window are reported in Table 5. Window size increased as the speed of transition slowed and as the corresponding time to detection increased. Abrupt transitions resulted in pre-drift observations more frequently being included in the returned data window. In most cases, less than 20% of iterations for each post-drift calibration setting returned windows containing pre-drift observations. The major exception being an abrupt change to a model in which a subgroup of low risk observations was substantially overpredicted. In this case, pre-drift data was returned almost 50% of the time. For rapid and gradual temporal transitions, less than 10%, and often less than 5%, of detections included pre-drift observations in the returned window. For rapid and gradual temporal transitions, typically less than 2% of the observations in returned windows occurred prior to drift onset (see Figure 28). For abrupt transitions, typically less than 5% of the observations in returned windows occurred prior to drift onset. We note that there is no pre-drift period in the recurrent/seasonal case, and thus no possibility of the returned data window containing pre-drift observations.

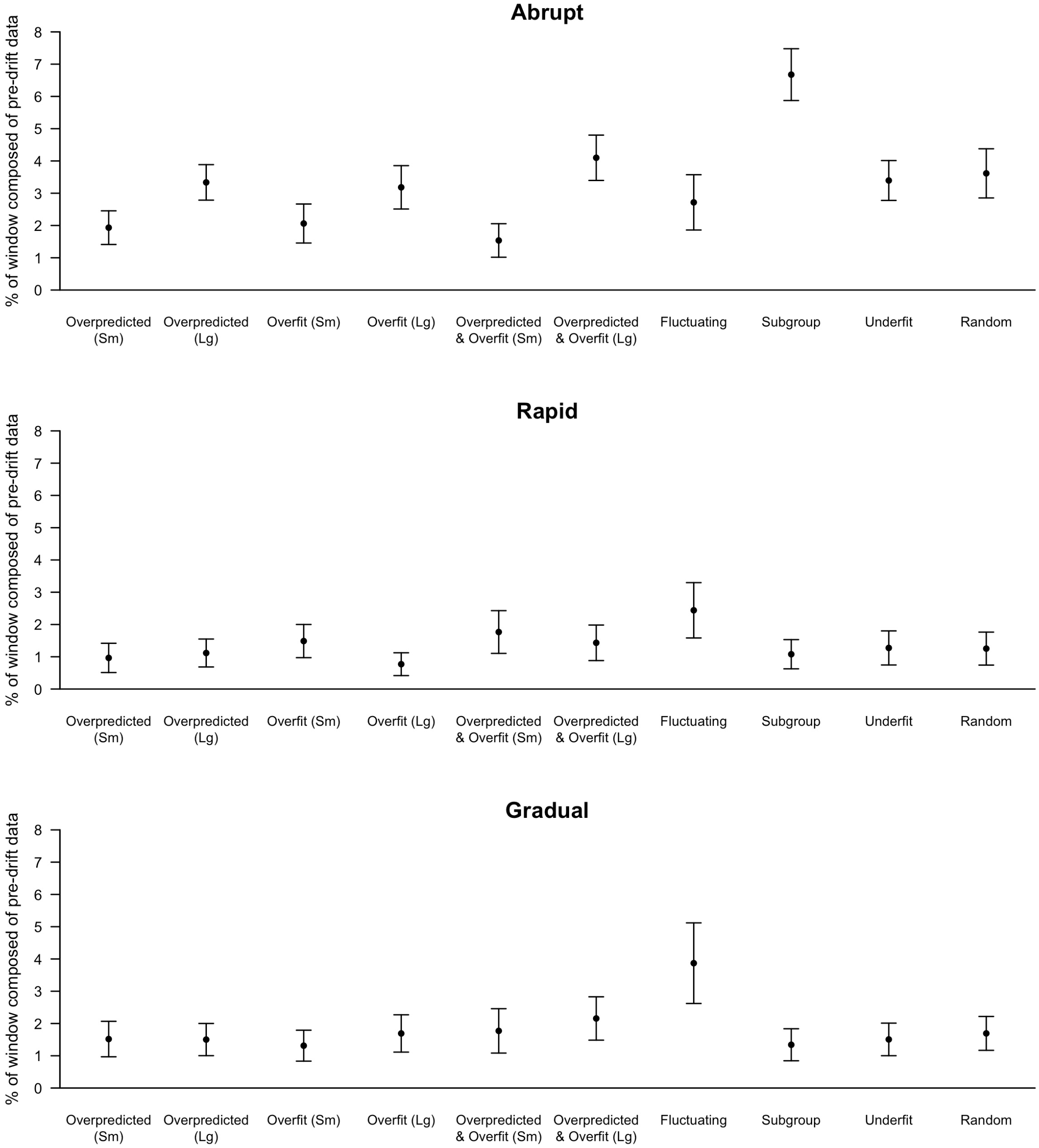
Discussion

To support timely, data-driven identification of performance drift in clinical prediction models, we developed a calibration drift detection system built on the adaptive windowing drift detection framework. This system, illustrated in Figure 16 above, integrates dynamically updated calibration curves into the adaptive windowing algorithm to support evaluating drift using a stringent metric of calibration (i.e., the absolute difference between predicted probabilities and fitted values from an up-to-date calibration curve). Our calibration drift detection system is designed to not only inform users when performance drift is identified, but also provide guidance on what data might be appropriate for responding to that drift. This

Table 5. Properties of retained data windows after drift detection.

Post-drift calibration setting	Transition pattern	Size (median & IQR)	% including pre-drift obs
Overpredicted (small)	Abrupt	559 (423, 790)	9
	Rapid	546 (423, 776)	2.4
	Gradual	716 (512, 976)	3.9
	Recurrent/Seasonal	661 (408, 1146)	-
Overpredicted (large)	Abrupt	394 (310, 497)	21.9
	Rapid	419 (341, 538)	3.2
	Gradual	575 (457, 831)	4.5
	Recurrent/Seasonal	418 (341, 764)	-
Overfit (small)	Abrupt	388 (268, 560)	6.8
	Rapid	421 (284, 623)	4.6
	Gradual	552 (370, 882)	4
	Recurrent/Seasonal	403 (269, 745)	-
Overfit (large)	Abrupt	171 (137, 222)	15.6
	Rapid	231 (184, 286)	2.5
	Gradual	436 (290, 652)	4.3
	Recurrent/Seasonal	223 (181, 278)	-
Underfit	Abrupt	229 (204, 262)	25.3
	Rapid	263 (229, 323)	3.1
	Gradual	453 (340, 653)	4.9
	Recurrent/Seasonal	243 (216, 271)	-
Overpredicted & overfit (small)	Abrupt	975 (699, 1429)	6.7
	Rapid	930 (637, 1315)	4.5
	Gradual	972 (695, 1332)	4.4
	Recurrent/Seasonal	1025 (620, 1497)	-
Overpredicted & overfit (large)	Abrupt	382 (264, 527)	24.3
	Rapid	414 (301, 565)	3.3
	Gradual	591 (439, 875)	5.1
	Recurrent/Seasonal	494 (346, 1066)	-
Fluctuating	Abrupt	782 (493, 1183)	8.4
	Rapid	823 (502, 1282)	7.6
	Gradual	981 (549, 1459)	10.4
	Recurrent/Seasonal	876 (514, 1412)	-
Subgroup	Abrupt	350 (235, 475)	49.9
	Rapid	400 (307, 517)	3.4
	Gradual	640 (502, 855)	3.8
	Recurrent/Seasonal	510 (360, 1085)	-
Random	Abrupt	234 (189, 305)	13.5
	Rapid	277 (216, 385)	3.1
	Gradual	492 (339, 733)	5.4
	Recurrent/Seasonal	271 (218, 390)	-

Figure 28. Proportion and 95% confidence interval of observations in the retained window generated prior to drift onset. Note, not relevant for recurrent/seasonal transitions in which there is no pre-drift period.



system is generalizable across prediction models based on diverse learning algorithms and can be customized with alternative bounded performance metrics.

Evaluating our calibration drift detection system across multiple simulated magnitudes, complexities, and speeds of calibration drift, we found the method accurately detected performance drift, minimizing both false positives and false negatives. This translates to avoiding alert fatigue due to false alarms during periods of stable model performance and avoiding missed opportunities to address model performance by neglecting to notice drifting performance. After drift onset, time to drift detection was associated with the speed and magnitude of calibration drift. Abrupt transitions were detected with the shortest delay. Gradual transitions required the most post-drift observations before drift could be detected. This observation is to be expected, as slower transitions from a calibrated to miscalibrated model evolve performance characteristics more slowly and require more observations before change can be distinguished from noise. Our evaluation of the lag between identified change points along each temporal transition and detection points provides a more fair comparison of any delay in detection across differing speeds of drift. Lags to detection for each form of post-drift miscalibration were generally consistent between abrupt, rapid, and gradual transitions. Recurrent/seasonal transitions lead to the most variable times and lags to detection, and in most cases required multiple cycles of drift prior to detection.

The delay from drift onset to detection and the lag from the change point to detection were also strongly related to the magnitude of post-drift miscalibration. Smaller changes in calibration required more data to be detected than did larger changes in calibration, even when the form of eventual miscalibration was similar. This finding is highlighted by the two variations of overprediction, as well as the two variations of overfitting. The distribution of predicted probabilities in the data and the variability of miscalibration across the range of probability were also critical to drift detection. For example, even in the case of substantial miscalibration in low density, high probability ranges, the detector was more likely to fail to detect or delay detection if miscalibration was more subtle in the more densely populated low probability range (e.g., the modestly overpredicted and overfit post-drift scenario). This may indicate a need for additional tuning of the step size in the Adam implementation of the underlying dynamic calibration curves from which error was estimated.

When drift is detected, our calibration drift detection system reports the detection and returns a window of recent observations that appears to be internally consistent based on the adaptive windowing assessment. If we are to use these returned windows to support model updating in response to the identified drift, then these windows should ideally only include data from the post-drift period. In our evaluations, settings with short temporal transitions most commonly included pre-drift data in the returned window. However, even in such cases the majority of simulated timeseries did not include pre-drift data in the returned window, and among those returned windows capturing pre-drift data, most observations were generated after drift onset. For the rapid and gradual transitions, drift was typically detected before the data had completely transitioned to the post-drift model. In such cases, returned data windows represented a transitional state rather than data from a new, stably miscalibrated setting. While updating with such data may improve model performance, it may also require subsequent or even periodic updating as performance continues to evolve. This may actually be most representative of how model updating would actually take place in ever-evolving clinical environments where model performance may never reach extended periods of stability. We are unable to comment on whether the returned windows are large enough to support model updating, as this would be dependent on the complexity of the model, the learning algorithm, and the degree of updating required to return the model to acceptable performance.

Our calibration drift detection system as presented here has several limitations. First, the adaptive windowing algorithm relies on a two-sided test. If we are only interested in detecting deteriorating model performance, we may be able to implement a more powerful test by adjusting the method to support one-sided analyses. Additionally, our system monitors model calibration or other user-preferred performance metrics. One could argue that we should instead be evaluating data streams for changes in predictor distributions and associations. Tracking these additional features may allow us to better recognize structural changes that could render a model unreliable (e.g. changes in data capture/coding) and require a tailored updating response such as model reparameterization or extension. However, while monitoring additional features of the data stream may be useful, it would not be sufficient in many cases. Unless changes in data stream features affect the accuracy of model predictions, such changes alone may not warrant model updating. Using our calibration drift detection system in combination with on-going model assessments, such as visualizations and summaries of dynamic calibration curves, model managers would have insights into any abrupt changes in performance that may signal structural issues in the input data stream and warrant further investigation. Further, maintaining open communication with clinical users can provide insight into critical clinical practice changes that may require substantive model adjustment that could be undertaken regardless of a drift detector's status.

Limitations of the current evaluations warrant continued investigation of the performance characteristics of our calibration drift detection system. With the exception of the recurrent/seasonal transition pattern, drift was preceded by a period of stable model performance. Findings may differ for timeseries with no initial stability or extended initial periods of stability. We only present results for timeseries moving away from calibration over time. Models are unlikely to be truly calibrated, even initially, and transitions between different forms of miscalibration may be easier or more difficult to detect. Additional insights may also come from exploring the performance of our calibration drift detection system in real clinical datasets where the timing and form of performance drift is uncertain.

Conclusion

Building on the dynamic calibration curves described in the previous chapter, we developed and evaluated a calibration drift detection system to provide data-driven guidance on when clinical prediction models may require updating. This system, generalizable irrespective of the learning algorithm on which categorical prediction models are built, supports alignment of model updating with the timing of performance drift. By updating models as performance deteriorates rather than on pre-determined schedules, model managers can avoid interim periods of insufficient model accuracy between scheduled updates and focus analytic resources on those models most in need of attention. Our calibration drift detection system also provides insight into a candidate updating set by returning a window of recent observations occurring after the point at which performance drift was identified. This system can be used to initiate predefined model updating strategies or in conjunction with data-driven methods to select updating methods. We explore methods for the latter approach in the following chapter.

CHAPTER 6

A NONPARAMETRIC TESTING PROCEDURE TO GUIDE UPDATING METHODS AND CORRECT PERFORMANCE DRIFT

In this chapter, we describe a new nonparametric testing procedure to recommend updating methods that minimizes overfitting, accounts for uncertainty associated with updating sample sizes, and is widely applicable to both parametric and nonparametric prediction models. We illustrate the properties of this testing procedure on both simulated scenarios of population shifts that impact clinical use cases and two case studies leveraging Department of Veterans Affairs inpatient admission data. Please note, large portions of this chapter were previously published in the Journal of the American Medical Informatics Association.[35]

A New Testing Procedure

Overview

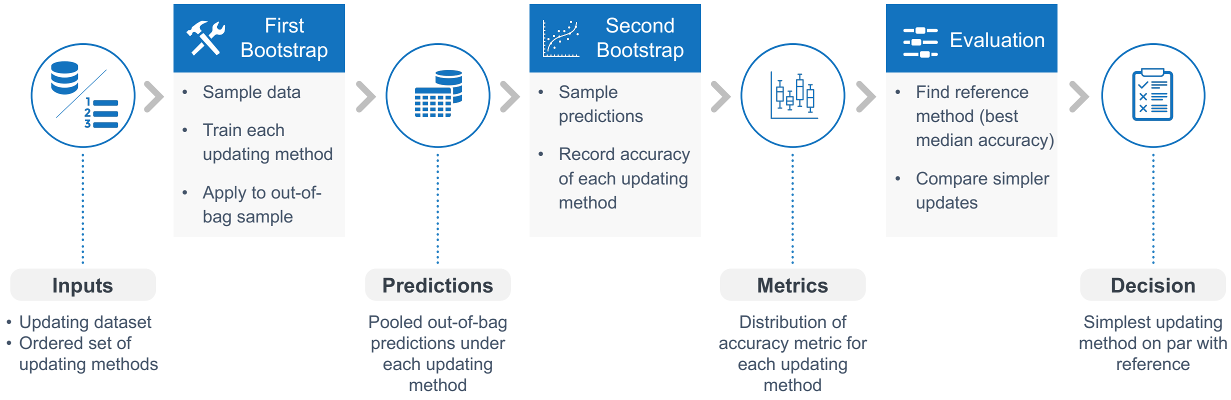
We sought to develop a testing procedure that recommends the simplest updating method that maximizes model performance in terms of accuracy, discrimination, or calibration. This procedure was designed to meet the following goals:

- prefer simple updating methods without compromising performance;
- work with any binary or categorical prediction model, regardless of the underlying learning algorithm;
- support customization to meet use case-specific requirements.

We pursued these objectives while also prioritizing a generalizable and extensible testing structure that avoids unnecessary assumptions regarding test inputs. The procedure calls for users to provide an existing categorical prediction model, a set of new observations to be used for updating, a set of updating methods order by complexity (or preference), and a scoring rule by which to compare updating methods. Observations in the updating set may comprise observations from a new clinical setting in which the model will be applied or observations accruing since the model was trained. Given these inputs, our procedure identifies the simplest (or most preferable) updating method that improves performance to comparable or superior levels than might be achieved with more sophisticated (or less preferable) updating methods.

Figure 29 provides an overview of the testing procedure's two-stage bootstrapping approach. We selected bootstrapping as the resampling method for both stages to maintain the sample size of the updating set for all assessments. The first bootstrapping stage minimizes the influence of overfitting on the procedure's recommendations by providing an out-of-bag set of updated predictions. The second bootstrapping stage utilizes these predictions to evaluate each updating method on samples of equal size to the updating set, incorporating uncertainty associated with the updating sample size into decision-making.

Figure 29. Simplified overview of our nonparametric testing procedure.

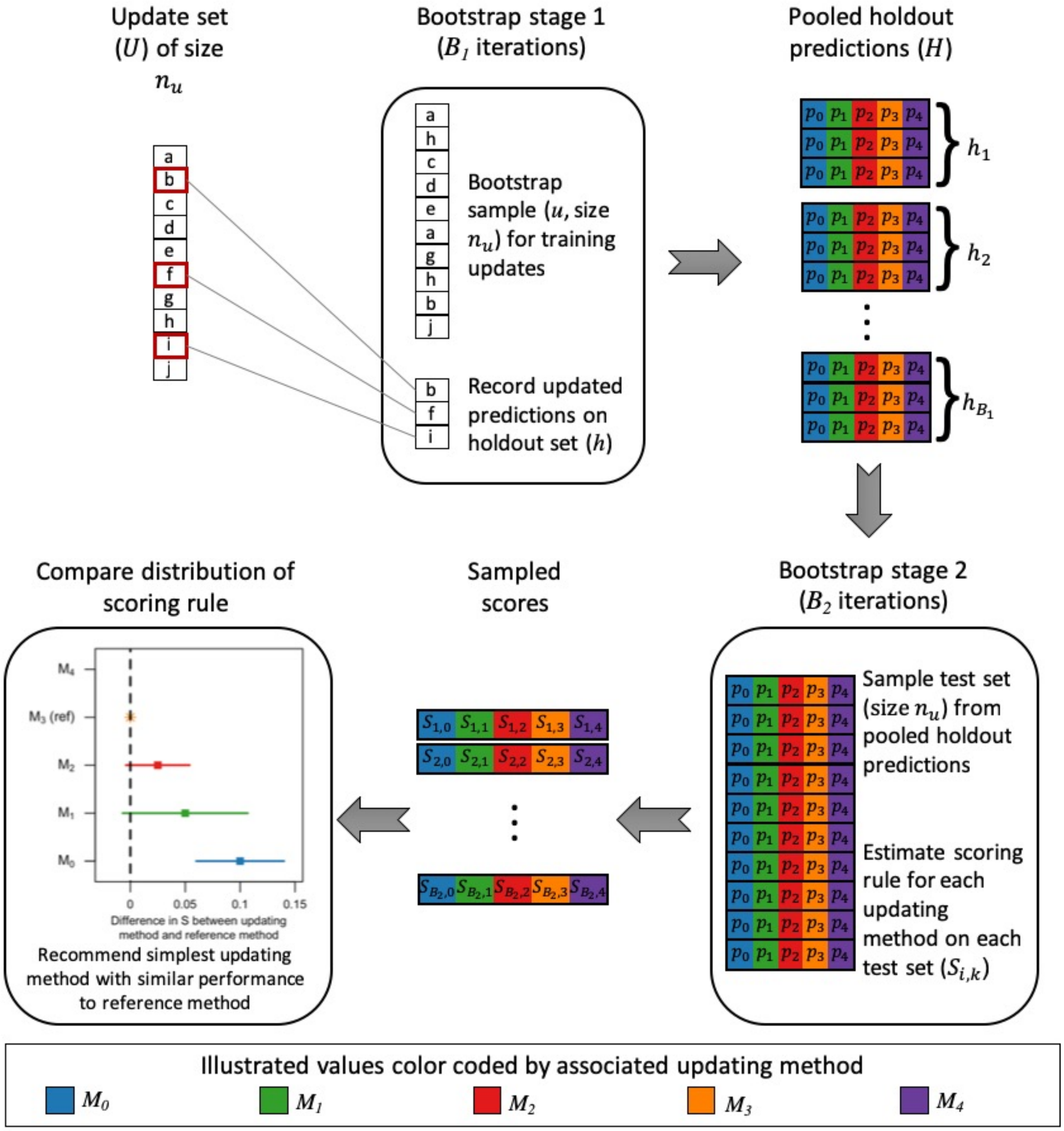

First Bootstrap
- Sample data
- Train each updating method
- Apply to out-of-bag sample
Second Bootstrap
- Sample predictions
- Record accuracy of each updating method
Evaluation
- Find reference method (best median accuracy)
- Compare simpler updates
Inputs
- Updating dataset
- Ordered set of updating methods
Predictions
Pooled out-of-bag predictions under each updating method
Metrics
Distribution of accuracy metric for each updating method
Decision
Simplest updating method on par with reference

Detailed Methodology

Given an update set (U) of size n_u and a current model (M_0), users define a set of updating methods as $M_1, M_2, \dots, M_m$, where methods are sorted by increasing statistical complexity or decreasing user preference. By default, the testing procedure includes retention of the original model as M_0, defining this approach the most preferable option. User-specified updating approaches may include commonly applicable methods (e.g., recalibration), as well as model-specific methods (e.g., reweighting the leaf nodes of each tree in a random forest model). See Chapter 2 for descriptions of several updating techniques.

To supplement the basic outline in Figure 29, the processes and flow of data through each of step of the testing procedure are illustrated in more detail in Figure 30. We begin by developing a pooled set of holdout predictions (H) via the first bootstrapping stage. For each of

Figure 30. Detailed illustration of our nonparametric testing procedure.


Update set (U) of size n_u
a
b
c
d
e
f
g
h
i
j
Bootstrap stage 1 (B_1 iterations)
a
h
c
d
e
a
g
h
b
j
Bootstrap sample (u, size n_u) for training updates
b
f
i
Record updated predictions on holdout set (h)
Pooled holdout predictions (H)
p_0 p_1 p_2 p_3 p_4
h_1
h_2
h_{B_1}
Bootstrap stage 2 (B_2 iterations)
Sample test set (size n_u) from pooled holdout predictions
Estimate scoring rule for each updating method on each test set ($S_{i,k}$)
Sampled scores
$S_{1,0}$ $S_{1,1}$ $S_{1,2}$ $S_{1,3}$ $S_{1,4}$
$S_{2,0}$ $S_{2,1}$ $S_{2,2}$ $S_{2,3}$ $S_{2,4}$
$S_{B_2,0}$ $S_{B_2,1}$ $S_{B_2,2}$ $S_{B_2,3}$ $S_{B_2,4}$
Compare distribution of scoring rule
M_4
M_3 (ref)
M_2
M_1
M_0
0
0.05
0.1
0.15
Difference in S between updating method and reference method
Recommend simplest updating method with similar performance to reference method
Illustrated values color coded by associated updating method
M_0
M_1
M_2
M_3
M_4

B_1 iterations, we randomly sample with replacement n_u observations from U, defining this sample as u. We construct a holdout set (h) with those observations from U not included in u. Predicted probabilities from M_0 are estimated for all observations in both u and h. Based on u, we calculate the adjustments required for updating methods M_1 through M_m. We apply these adjustments to h, resulting in a set of predicted probabilities based on the current model and each updating method for all observation in h. Holdout set predictions are pooled across bootstrap iterations to construct H. Here we set B_1 to be 100; however, fewer iterations may be permissible as long as H is large enough to capture variability in predictions and each observation is included in H with similar probability.

The performance of each updating method is evaluated on H via the second bootstrapping stage. For each of B_2 iterations, we randomly sample with replacement n_u observations from H and measure the performance of each updating method in this sample with the user-defined scoring rule (S). A variety of accuracy, discrimination, or calibration metrics may be applicable here, with selection dependent on those aspects of performance most relevant for a given use case. To enable stable quantile estimates of the scoring rule for each updating method, we set B_2 to be 1,000. This process results in a set of $S_{i,k}$ where $i = 1, 2, \dots, B_2$ indexes the iteration and $k = 0,1, \dots, m$ indexes the updating method.

Finally, we define M_r as the updating method for which the median S is closest, in terms of absolute value, to the scoring rule's ideal value. No other method will have significantly better performance than M_r as their accuracy cannot be significantly closer to the scoring rule's ideal value; however, other methods may exhibit similar performance with a score that is not significantly different than that of M_r. Since our procedure aims to recommend the simplest (or most preferable) updating method that does not compromise performance, we need only consider whether any simpler (or more preferable) methods perform comparably to M_r. If M_r is the current model (M_0), then no comparisons are needed and the procedure recommends retaining the current model. Otherwise, starting with $k = 0$, we estimate the percentile-based 100$(1 - \alpha)$% confidence interval for the paired difference in S between M_k and M_r. When this interval contains 0, indicating no significant difference, the procedure recommends M_k. When this interval does not contain 0, we increment k and repeat until a recommendation is made or $k = r$, in which case M_r is recommended.

We do not correct for multiple comparisons in this final step of the procedure because we do not seek to control the familywise error rate of rejecting one or more null hypotheses of no difference between models. Rather, we seek to identify significant differences with a standard correction for uncertainty that does not depend on the number of comparisons being made. Operationally, users can control the stringency of this correction uniformly through setting α in the 100$(1 - \alpha)$% confidence interval. We are also not comparing all methods to M_r simultaneously. Instead, we are filtering options and defining pairwise comparisons based on predefined preferences. For similar reasons, we encourage using the 100$(1 - \alpha)$% confidence interval framework over a hypothesis testing framework. However, the latter approach is equally easy to execute. Rather than forming an interval from the $\frac{\alpha}{2}$ and $1 - \frac{\alpha}{2}$ quantiles of the distribution of differences, the user would estimate q, the quantile represented by 0. A "p-value" equal to $2 * \min(q, 1-q)$ is compared to α where being less than α is equivalent to 0 being excluded from the interval.

Default Parameterization

To investigate the properties and performance of our testing procedure, we defined default values for the customizable aspects of the testing procedure. These values are widely generalizable and may serve as a baseline implementation of the testing procedure for users not wishing to pursue customization. We specified the set of updating methods as intercept correction (M_1), linear logistic recalibration (M_2), flexible logistic recalibration (M_3), and model refitting (M_4). Intercept correction and linear logistic recalibration are common approaches correcting systematic over/under prediction and overfitting, respectively.[26, 30] Flexible logistic recalibration extends the linear logistic recalibration approach to allow nonlinearity in the association between outcomes and baseline predictions, potentially correcting more complex forms of miscalibration.[66] Each of these updating methods may be applied to any categorical prediction model. See Chapter 2 for further detail on these updating techniques. We specified S as the Brier score. This quadratic scoring rule measures model accuracy by incorporating both discrimination and calibration.[37, 103] As the Brier score tends towards 0 with increasing accuracy, M_r is the updating method with the minimum median Brier score. We investigated the sensitivity of the testing procedure to the choice of scoring rule in a sensitivity analysis in which we replaced the Brier score with a logarithmic scoring rule.

Simulation Study

We conducted a simulation study to characterize the performance of our testing procedure under population shifts that may impact model performance in clinical settings.[18, 19, 104, 105] Such shifts involve changes in outcome prevalence, distributions of risk factors (i.e., case mix), and predictor-outcome associations. In the presence of each form of population shift, we updated a logistic regression model with recommendations both from our testing procedure and from a baseline testing procedure proposed by Vergouwe and colleagues.[33] Using sequential likelihood ratio tests, Vergouwe *et al*'s closed testing procedure selects the simplest updating method providing a fit similar to model refitting. See Chapter 2 for additional detail. We documented updating recommendations and compared performance under these recommendations with alternative updating methods.

Methods

A single model development population was generated and used to train the logistic models considered for updating by the testing procedures. This population included 100,000 observations with 32 covariates generated from multivariate normal, gamma, binary, Poisson, and multinomial distributions, reflecting a variety of predictor types that may be observed in clinical datasets. Details of predictor distributions are provided in Appendix D. These covariates and several interactions among them served as predictors for two reference logistic regression models, one with 10 degrees of freedom (df) and another with 40. The logistic regression models were defined by the following equation:

$$\begin{aligned} P(Y=1|X) = [1 + exp\{-(&\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 x_5 + \beta_6 x_6 + \beta_7 x_7 + \beta_8 x_8 \\ &+ \beta_9 x_9 + \beta_{10} x_{10} + \beta_{11} x_{11} + \beta_{12} x_{12} + \beta_{13} x_{13} + \beta_{14} x_{14} + \beta_{15} x_{15} + \beta_{16} x_{16} \\ &+ \beta_{17} x_{17} + \beta_{18} x_{18} + \beta_{19} x_{19} + \beta_{20} x_{20} + \beta_{21} x_{21} + \beta_{22} x_{22} + \beta_{23} x_{23} + \beta_{24} x_{24} \\ &+ \beta_{25} x_{25} + \beta_{26} x_{26} + \beta_{27} x_{27} + \beta_{28} x_{28} + \beta_{29} x_{29} + \beta_{30} x_{30} + \beta_{31} x_{31b} + \beta_{32} x_{31c} \\ &+ \beta_{33} x_{32b} + \beta_{34} x_{32c} + \beta_{35} x_1 x_{23} + \beta_{36} x_5 x_{22} + \beta_{37} x_{11} x_{12} + \beta_{38} x_{12} x_{13} \\ &+ \beta_{39} x_6 x_{26} + \beta_{40} x_3 x_{28})\}]^{-1} \end{aligned}$$

where

x_{31b} = dummy variable for 2nd level of X_{31}

x_{31c} = dummy variable for 3rd level of X_{31}

x_{32b} = dummy variable for 2nd level of X_{32}

x_{32c} = dummy variable for 3rd level of X_{32}

For the model with df =10, coefficients for select variables were set to 0, reducing the model form to

$$P(Y=1|X) = [1 + exp\{-(\beta_0 + \beta_2 X_2 + \beta_3 X_3 + \beta_5 X_5 + \beta_6 X_6 + \beta_8 X_8 + \beta_{16} X_{16} + \beta_{22} X_{22} + \\ \beta_{24} X_{24} + \beta_{29} X_{29} + \beta_{36} X_5 * X_{22})\}]^{-1}$$

Using coefficients defined as noted in Appendix D and intercepts adjusted to establish a population event rate of 25%, we calculated probabilities for each observation in the model development population under both the $df = 10$ and $df = 40$ models. A binary outcome under both models was defined by comparing these probabilities to random values generated from a uniform [0,1] distribution. If the random value was less than or equal to the assigned probability, the observation was assigned $Y = 1$, otherwise the observation was assigned $Y = 0$.

We constructed updating and evaluation populations under five population shift scenarios. Our simulated scenarios illustrate situations in which the testing procedure is applied to a fully shifted population rather than a gradually shifting population where observations are a mixture of the pre- and post-shift patterns. This may reflect updating after transporting a model to a new clinical setting or after a long delay. Shifted populations differed from the model development population in the following ways:

1. *No population shift* – predictors and outcomes generated in the same way was the development population.
2. *More prevalent outcome* – predictors generated in the same way was the development population; outcomes generated with an adjusted intercept.
3. *More homogenous case mix* – predictors generated from less variable distributions; outcomes generated in the same way was the development population.
4. *More heterogenous case mix* – predictors generated from more variable distributions; outcomes generated in the same way was the development population.
5. *Shift in predictor-outcome associations* – predictors generated in the same way was the development population; outcomes generated from models with adjusted coefficients.

For each population shift scenario, we simulated 200,000 observations with adjusted parameters, assigning half to the updating population and half to the evaluation population. Under the scenario of no population shift, these data were simulated with the same settings as the development population. To simulate event rate shift, we adjusted the intercept of the logistic models to increase the outcome prevalence from 25% to 30%. Observations for the more homogenous and heterogenous case mix scenarios were generated by decreasing and increasing the variability of predictor distributions, respectively. For the predictor-outcome association shift scenario, we adjusted half the logistic models' coefficients by 20%, with some increasing and others decreasing in strength of association. See Appendix D for additional details on adjustments.

We explored the impact of population shifts on updating recommendations under varying training (n_t= 1000, 5000, and 10000) and updating (n_u= 1000, 5000, and 10000) sample sizes. We expect larger n_t may lead to more robust, generalizable models that are more amenable to recalibration rather than requiring refitting under some scenarios. As larger n_u provide more information to support updating, we expect them to lead to more complex updating recommendations than smaller n_u under all population shift scenarios. We trained either the simple (df = 10) or complex (df = 40) logistic regression model on n_t observations sampled from the development population. We sampled n_u observations from both the updating and evaluation populations of each population shift scenario. To determine the recommended updating method, we applied our testing procedure to the updating sample. To document the impact of updating recommendations, we assessed the performance of each available updating method on the evaluation sample. This process was repeated 1,000 times for each combination of model complexity, n_t, and n_u.

Results

The updating recommendations of our testing procedure by population shift scenario, training sample size, and updating sample size for the df = 10 and df = 40 models are detailed in Tables 6 and 7, respectively.

When no population shifts occurred, our test generally recommended retaining the original model. As the updating samples increasingly outweighed training samples (i.e., $n_t \ll n_u$), model refitting became the primary recommendation. A similar pattern emerged when the event rate increased. In this case, intercept correction was recommended; however, a shift toward model refitting was apparent as the updating sample dominated the training sample. With small updating samples, test recommendations were split between not updating and intercept correction. Under both population shifts, the recommended updates provided superior or similar calibration to that achieved with more complex updating (see Figures 31-34). Recommendations for more complex updating when training samples were very small compared to updating samples (i.e., $n_t \ll n_u$) improved performance compared to the original model.

Table 6. Percent of iterations for which each updating methods was recommended by our nonparametric testing procedure under each simulated scenario, training sample size (n_t), and updating sample size (n_u) when df = 10.

		n_t = 1000			n_t = 5000			n_t = 10000		
Scenario	**Updating method**	n_u= 1000	n_u= 5000	n_u= 10000	n_u= 1000	n_u= 5000	n_u= 10000	n_u= 1000	n_u= 5000	n_u= 10000
No population shift	No update	99.8	60.8	14.9	100	99.3	93.1	100	99.7	98.5
	Intercept correction	0	5.9	4.5	0	0.7	4.5	0	0.3	0
	Linear recalibration	0.1	4.6	2.2	0	0	0	0	0	0
	Flexible recalibration	0	0.1	0.1	0	0	0	0	0	0
	Model refitting	0.1	28.6	78.3	0	0	2.4	0	0	1.5
Increased event rate	No update	52.9	0.7	0	53.3	0	0	63.4	0	0
	Intercept correction	46.1	52.1	15.4	46.7	99.7	95	36.6	99.7	99.7
	Linear recalibration	0.9	6.3	3.2	0	0	0.1	0	0.3	0.1
	Flexible recalibration	0	0.2	0.6	0	0	0	0	0	0
	Model refitting	0.1	40.7	80.8	0	0.3	4.9	0	0	0.2
Less variable case mix	No update	99.2	74.9	31.2	100	99.2	99.5	100	99.6	99.9
	Intercept correction	0	9.2	10.2	0	0.8	0.4	0	0.4	0.1
	Linear recalibration	0.8	3.8	7.1	0	0	0.1	0	0	0
	Flexible recalibration	0	0.2	0.4	0	0	0	0	0	0
	Model refitting	0	11.9	51.1	0	0	0	0	0	0
More variable case mix	No update	98.5	32.1	5.7	100	97.8	80.1	100	99.8	96.5
	Intercept correction	1	4	2.1	0	0.5	2.8	0	0.2	2.9
	Linear recalibration	0	3	1.9	0	0.5	0.4	0	0	0
	Flexible recalibration	0	0.1	0	0	0	0	0	0	0
	Model refitting	0.5	60.8	90.3	0	1.2	16.7	0	0	0.6
Association changes	No update	0	0	0	0	0	0	0	0	0
	Intercept correction	0.1	0	0	0	0	0	0	0	0
	Linear recalibration	0	0	0	0	0	0	0	0	0
	Flexible recalibration	0	0	0	0	0	0	0	0	0
	Model refitting	99.9	100	100	100	100	100	100	100	100

Table 7. Percent of iterations for which each updating methods was recommended by our nonparametric testing procedure under each simulated scenario, training sample size (n_t), and updating sample size (n_u) when df = 40.

		n_t = 1000			**n_t = 5000**			**n_t = 10000**		
Scenario	**Updating method**	**n_u= 1000**	**n_u= 5000**	**n_u= 10000**	**n_u= 1000**	**n_u= 5000**	**n_u= 10000**	**n_u= 1000**	**n_u= 5000**	**n_u= 10000**
No population shift	Not updating	94	7.7	0	100	99.7	87	100	100	100
	Intercept correction	1.2	1.3	0	0	0.2	0	0	0	0
	Linear recalibration	4.8	7.1	0	0	0.1	0	0	0	0
	Flexible recalibration	0	0.1	0	0	0	0	0	0	0
	Model refitting	0	83.8	100	0	0	13	0	0	0
Increased event rate	Not updating	43.6	0.1	0	43.4	0	0	48.1	0	0
	Intercept correction	48	5.2	0	56.6	99.2	81.7	51.9	100	100
	Linear recalibration	8.4	4.4	0	0	0.8	5.2	0	0	0
	Flexible recalibration	0	0	0	0	0	0	0	0	0
	Model refitting	0	90.3	100	0	0	13.1	0	0	0
Less variable case mix	Not updating	79.9	22.5	1	100	88.5	68.4	100	96.3	89.4
	Intercept correction	14.1	14.6	0.5	0	10.3	21.1	0	3.6	7
	Linear recalibration	6	20.7	1.6	0	1.2	5.2	0	0.1	3.6
	Flexible recalibration	0	0.1	0	0	0	0	0	0	0
	Model refitting	0	42.1	96.9	0	0	5.3	0	0	0
More variable case mix	Not updating	59	0	0	95.5	60.1	15.8	96.4	74.7	74.9
	Intercept correction	30.4	0	0	4.5	28.4	13.3	3.6	24.7	18
	Linear recalibration	8.2	0.2	0	0	1.7	7.8	0	0.4	3.5
	Flexible recalibration	0	0	0	0	0	0	0	0	0
	Model refitting	2.4	99.8	100	0	9.8	63.1	0	0.2	3.6
Association changes	Not updating	0	0	0	0	0	0	0	0	0
	Intercept correction	0	0	0	3.6	0	0	2.4	0	0
	Linear recalibration	0	0	0	0	0	0	1.1	0	0
	Flexible recalibration	0	0	0	0	0	0	0	0	0
	Model refitting	100	100	100	96.4	100	100	96.5	100	100

Figure 31. Brier scores in evaluation sets after predefined and test-recommended updates for simulated scenarios of no population change.

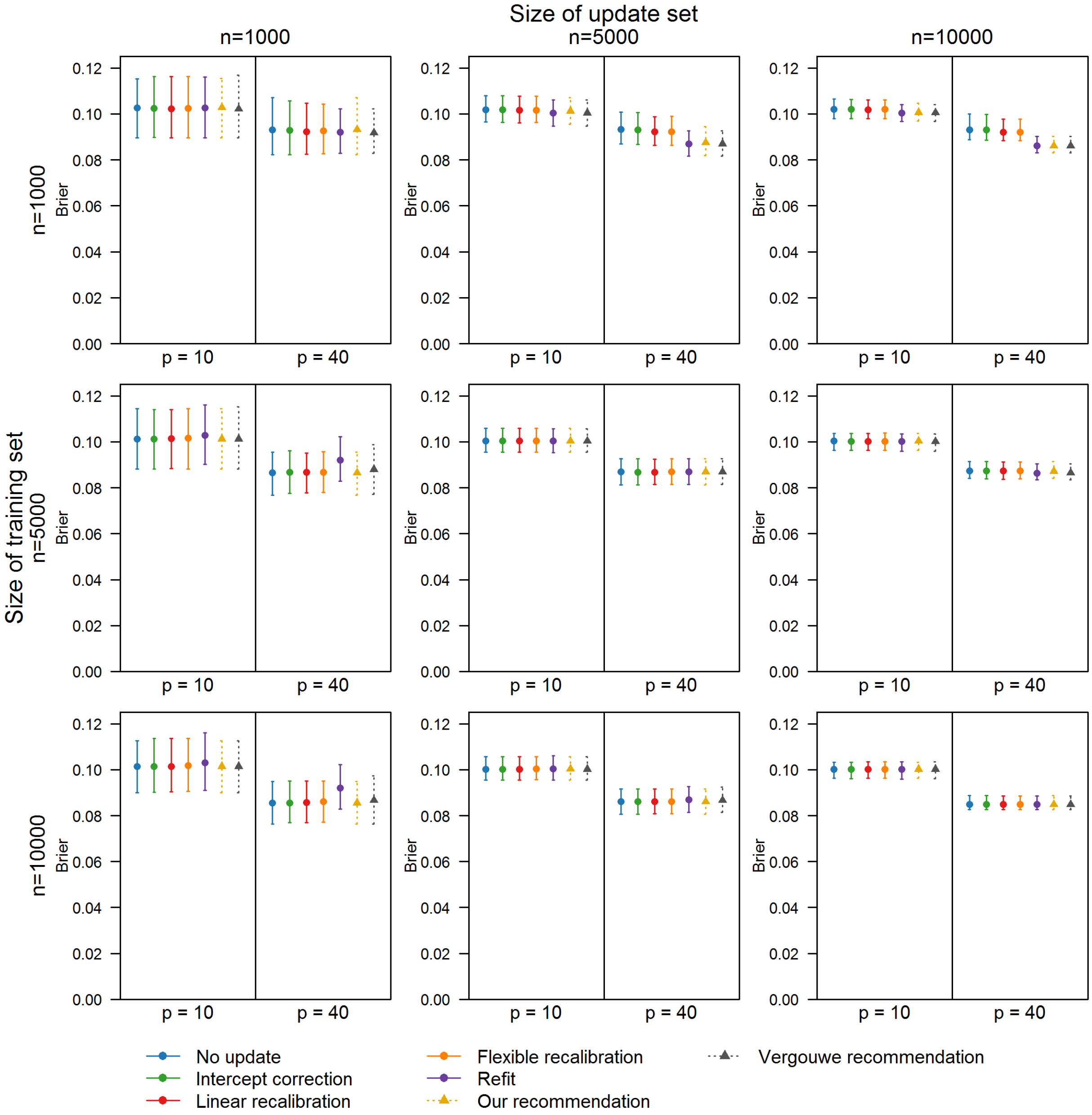


Figure 32. Estimated calibration index in evaluation sets after predefined and test-recommended updates for simulated scenarios of no population change.

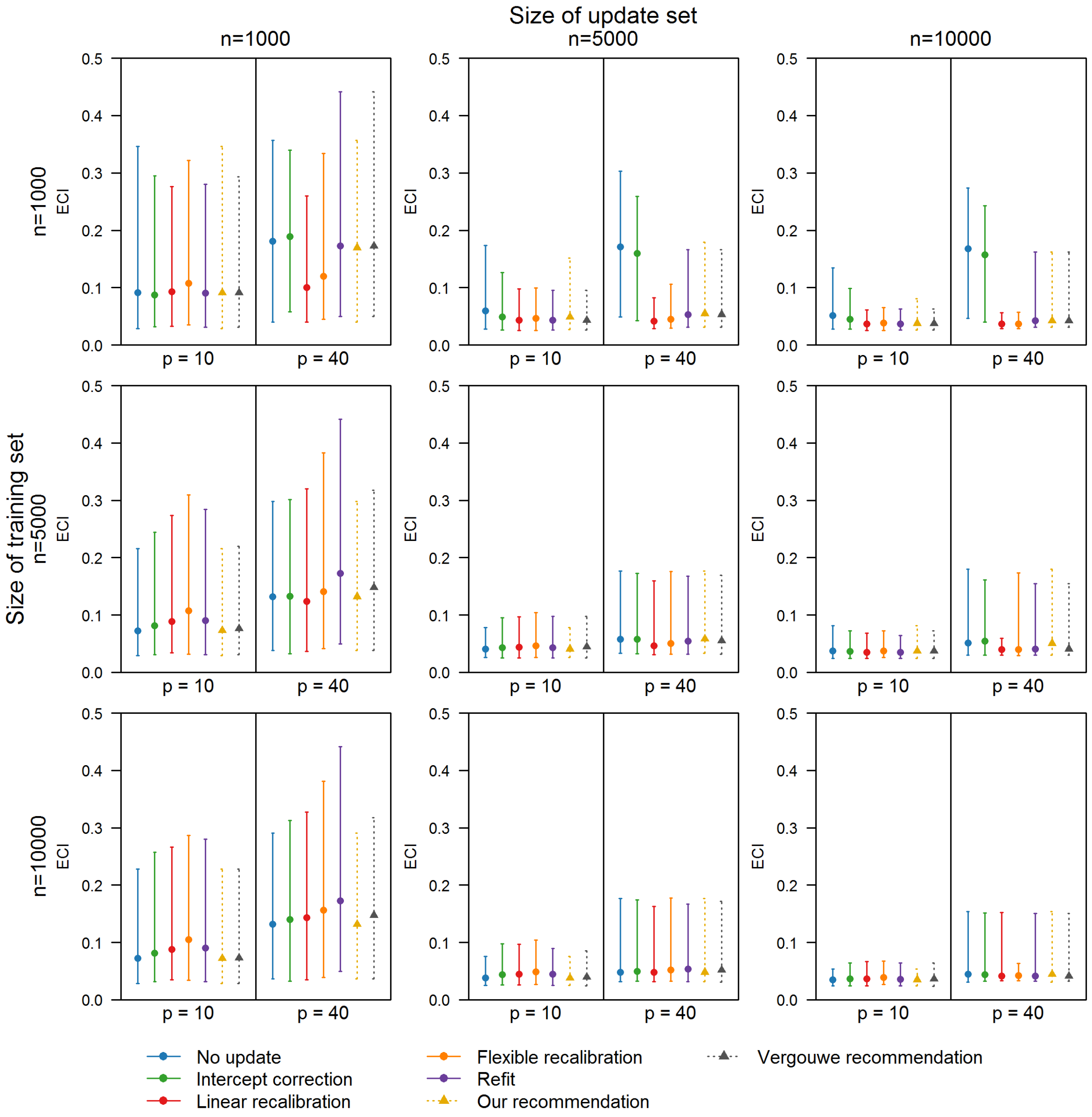


Figure 33. Brier scores in evaluation sets after predefined and test-recommended updates for simulated event rate change.

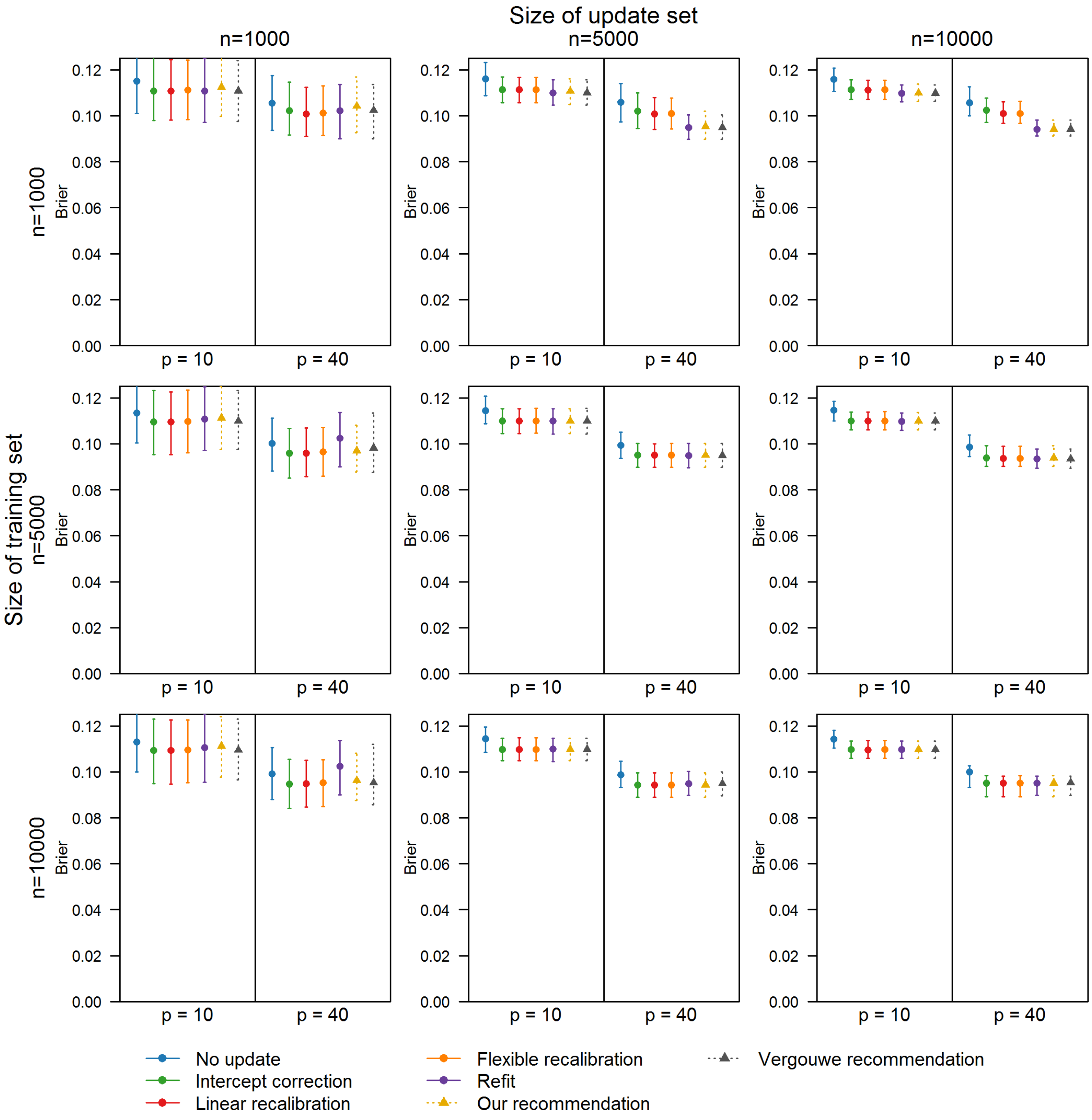


Figure 34. Estimated calibration index in evaluation sets after predefined and test-recommended updates for simulated event rate change.

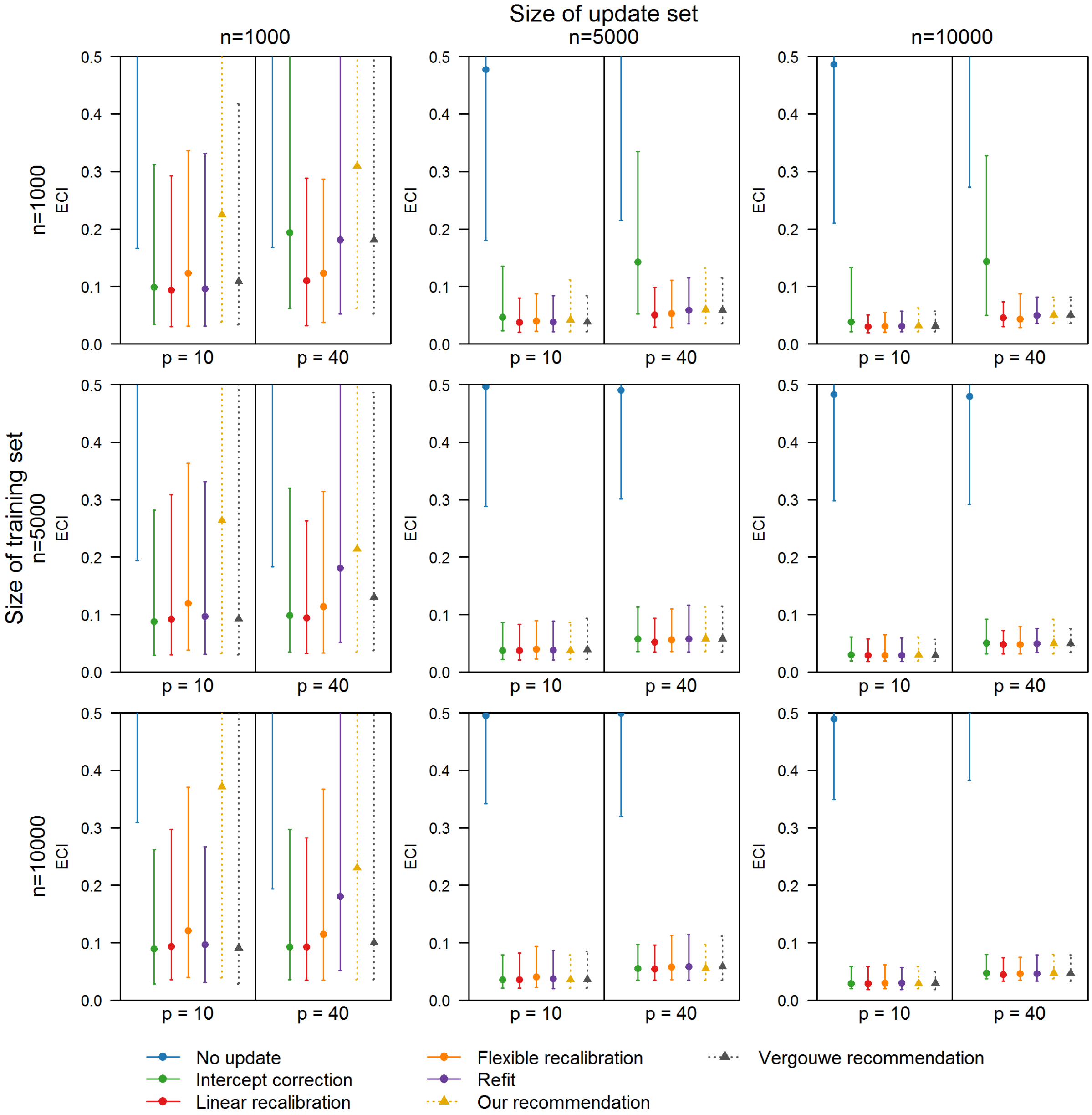


Figure 35. Brier scores in evaluation sets after predefined and test-recommended updates for simulated predictor-outcome association changes.

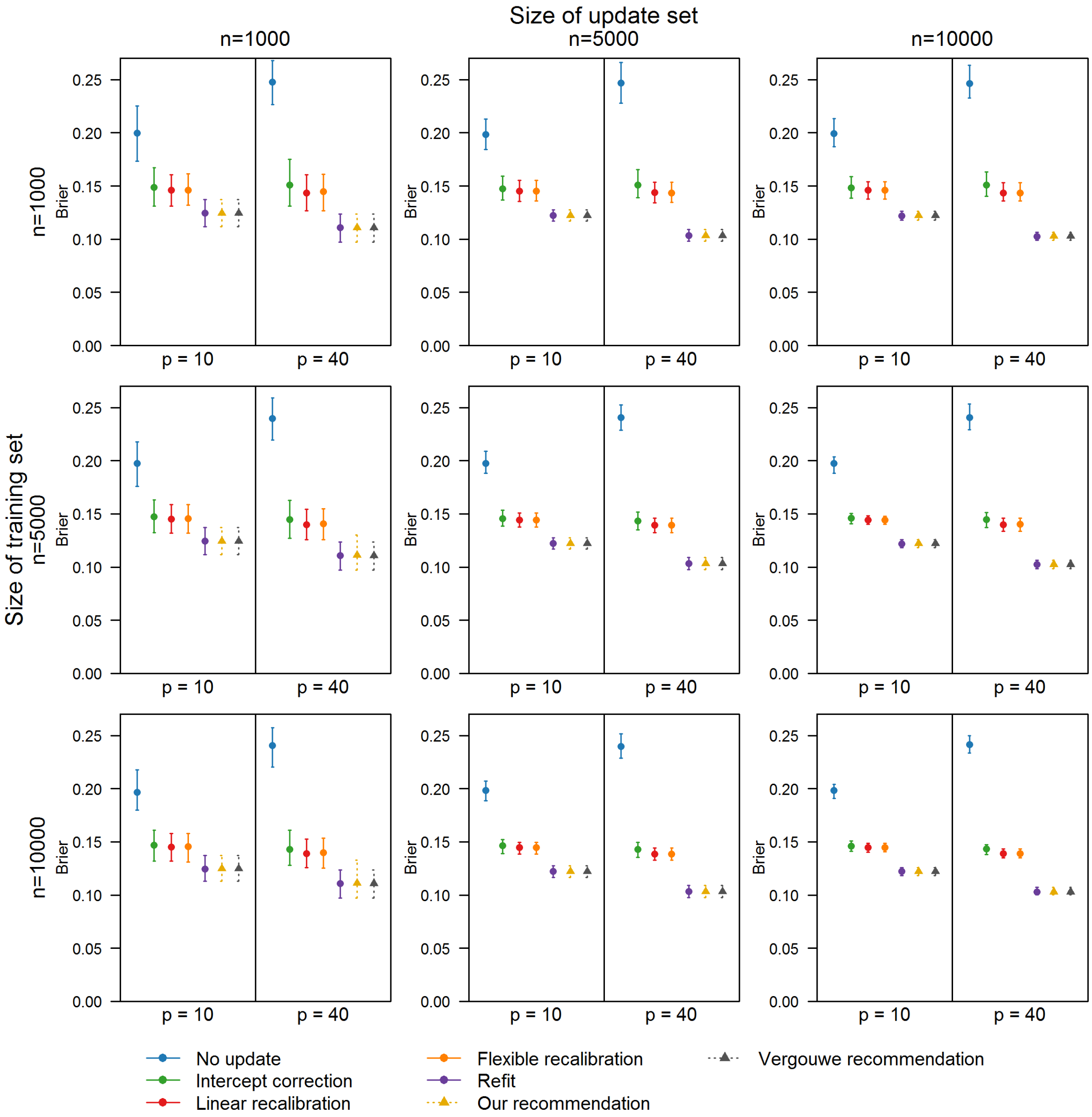


Figure 36. Estimated calibration index in evaluation sets after predefined and test-recommended updates for simulated predictor-outcome association changes.

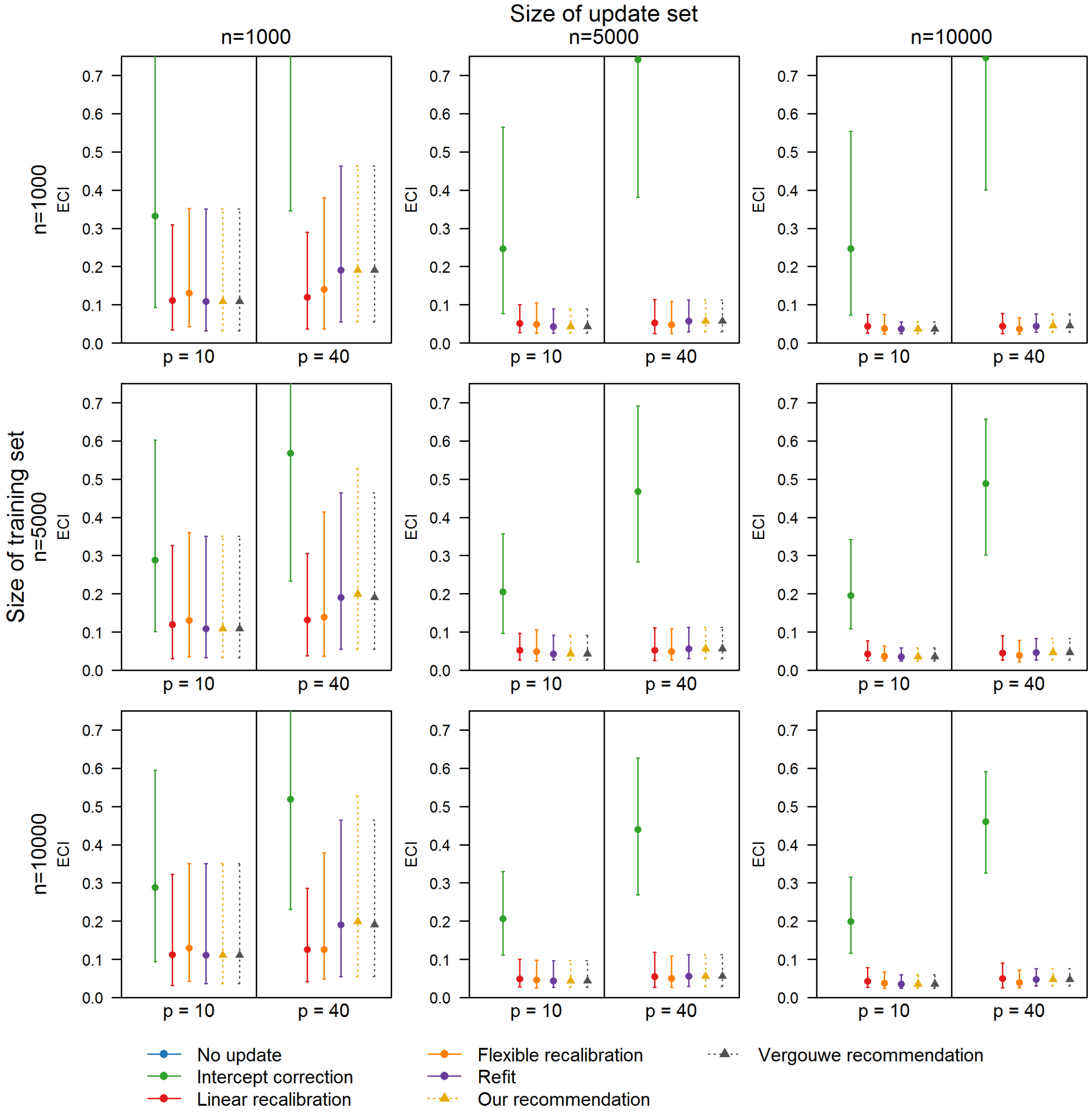


Figure 37. Brier scores in evaluation sets after predefined and test-recommended updates for simulated decrease in case mix variability.

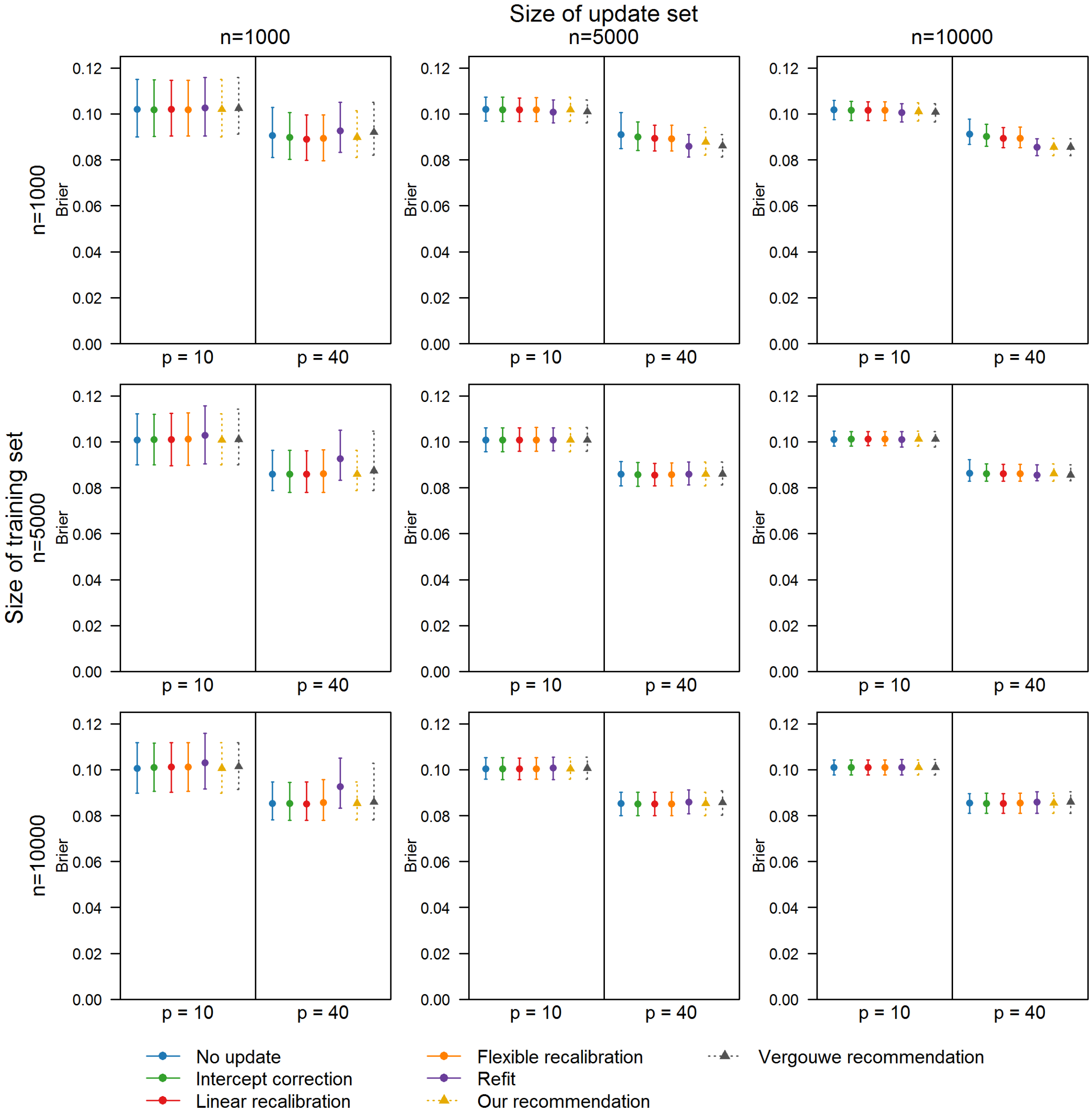


Figure 38. Estimated calibration index in evaluation sets after predefined and test-recommended updates for simulated decrease in case mix variability.

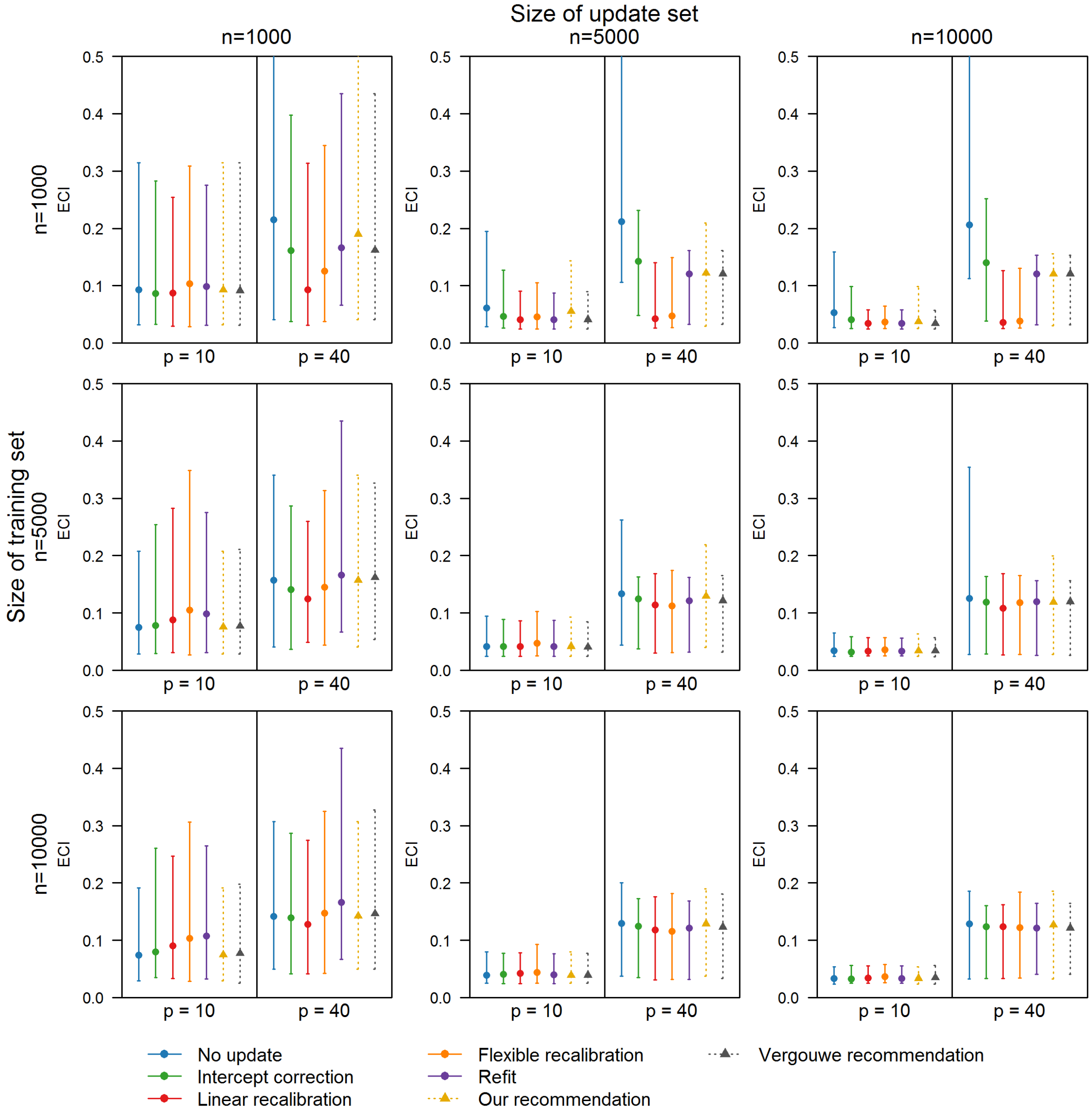


Figure 39. Brier scores in evaluation sets after predefined and test-recommended updates for simulated increase in case mix variability.

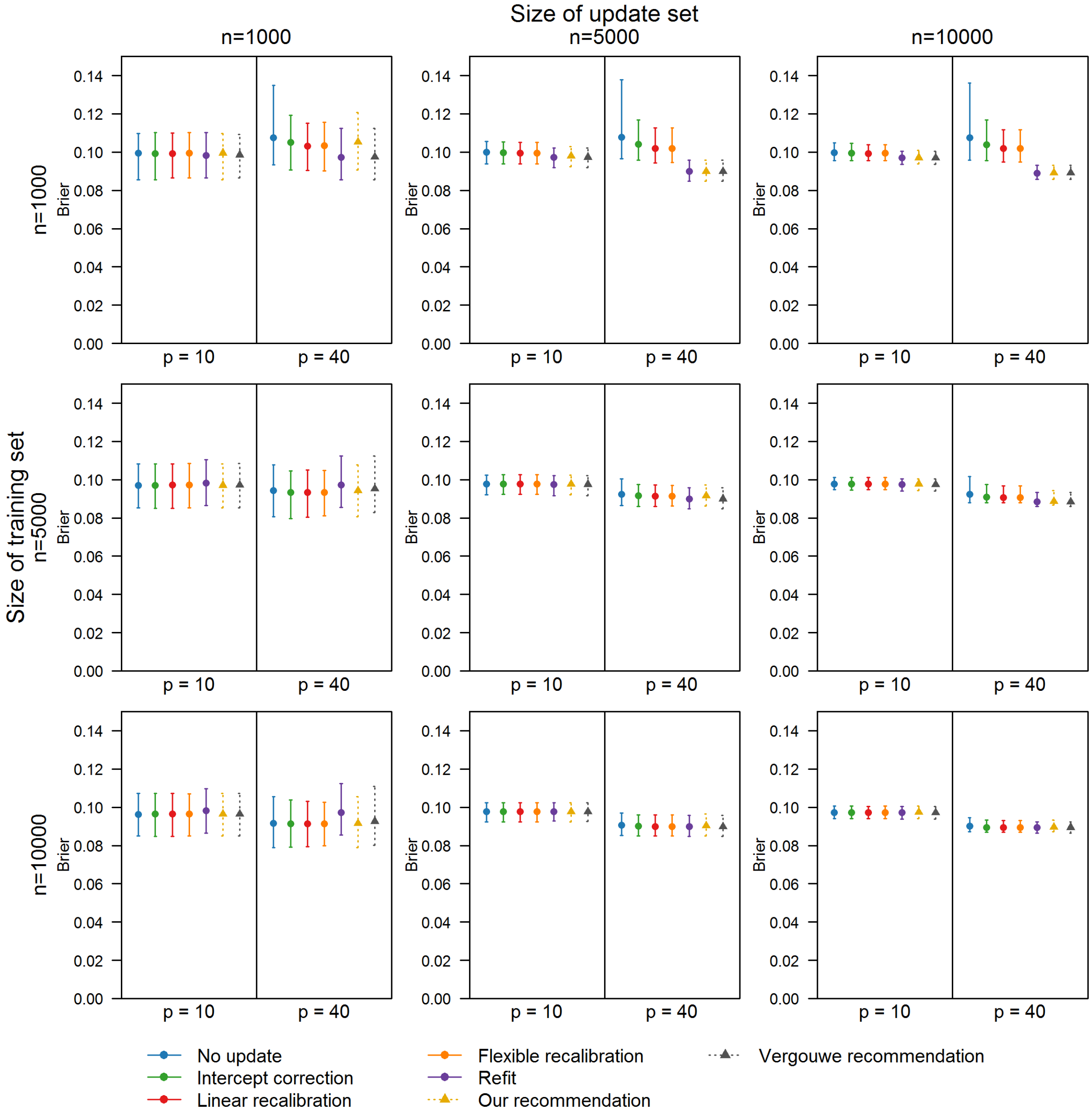
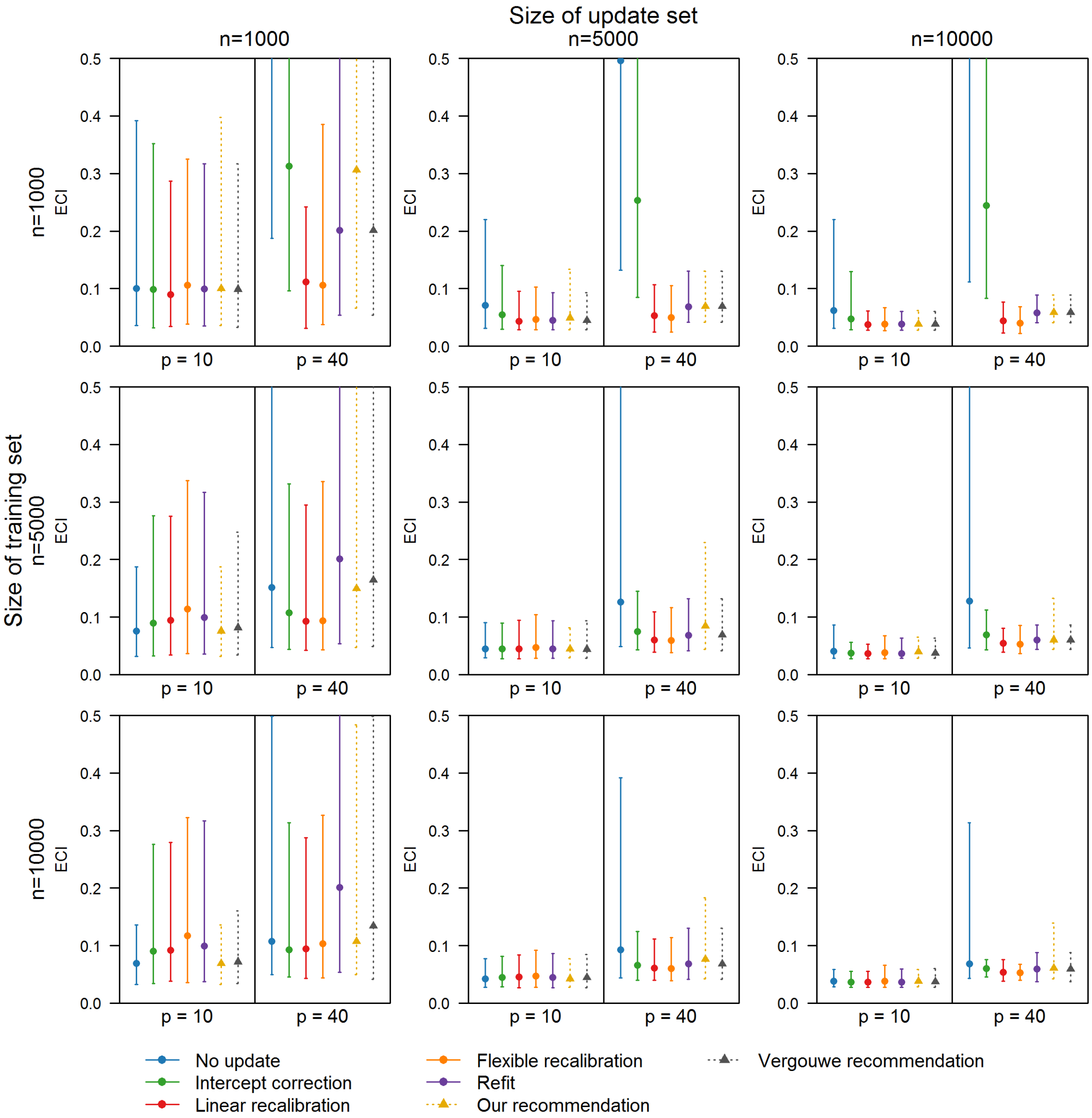


Figure 40. Estimated calibration index in evaluation sets after predefined and test-recommended updates for simulated increase in case mix variability.



In response to changes in predictor-outcome associations, our test recommended model refitting, regardless of the relative sizes of the training and updating samples. Refitting under predictor-outcome association shift improved accuracy compared to simpler updates, even when updating samples were smaller than training samples (see Figure 35-36).

Case mix shifts resulted in the most variable recommendations. When variability in case mix decreased between the training and updating populations, recommendations varied across the spectrum of updating methods. However, the overall trend was toward retaining the original model, particularly when updating samples included similar or smaller volumes of data than training samples. When n_t>1000, no significant improvement in performance was observed with updating, supporting the recommendation to retain the original model (see Figure 37-38).

With increasing variability in case mix, for the df = 10 model, refitting was recommended for updating samples of similar or larger size as training samples; however, not updating was the dominant recommendation for smaller update samples. Recommendations for the df = 40 model were primarily split between not updating and intercept correction, although refitting the model was recommended as updating samples grew larger than training samples. Calibration under the procedure's recommendations was generally less variable, but not significantly different, than that of the original model. More complex updates than those recommended did not provide additional improvement in performance (see Figure 39-40). For the smallest training samples, however, refitting with larger updating samples, as recommended, improved discrimination but not calibration compared to recalibration methods.

Comparison to baseline testing procedure

Updating recommendations based on Vergouwe *et al*'s closed testing procedure, extended to include a consideration of flexible logistic recalibration, are presented in Tables 8 and 9. Overall, recalibration recommendations were more variable and model refitting was recommended more often using Vergouwe's testing procedure compared to our testing procedure.

When no population shifts occurred between the development and updating populations, Vergouwe's procedure recommended refitting the model in the majority of cases, particularly for the df = 40 model. As with our testing procedure, the recommendation to refit a complex model when updating data outweighed the training data by 10 to 1 resulted in improved calibration compared to the original model. However, despite no differences in the training and updating populations, Vergouwe's testing procedure recommended refitting the model with an update

Table 8. Percent of iterations for which each updating methods was recommended by Vergouwe *et al*'s closed testing procedure under each simulated scenario, training sample size (n_t), and updating sample size (n_u) when df = 10.

		n_t = 1000			n_t = 5000			n_t = 10000		
Scenario	**Updating method**	n_u= 1000	n_u= 5000	n_u= 10000	n_u= 1000	n_u= 5000	n_u= 10000	n_u= 1000	n_u= 5000	n_u= 10000
No population shift	No update	45.4	0.7	0.2	87.6	37.7	16.7	95.3	65.3	40.2
	Intercept correction	5.1	0.8	0.1	3.3	7.1	3.4	1.2	7.1	10.4
	Linear recalibration	7.4	1.4	0.1	0.9	6.7	3	0	3.8	2.2
	Flexible recalibration	2.2	0.3	0.1	0.6	0.5	2.6	0	1.4	2.5
	Model refitting	39.9	96.8	99.5	7.6	48	74.3	3.5	22.4	44.7
Increased event rate	No update	2.5	0	0	5.9	0	0	10.6	0	0
	Intercept correction	44.8	1.6	0.3	85	37.8	22.8	84.6	71	41.8
	Linear recalibration	3.8	0.8	0	2.3	5.1	4.8	1.2	4.5	5.4
	Flexible recalibration	3.1	0.3	0	0.5	2.8	0.6	0	2	1.7
	Model refitting	45.8	97.3	99.7	6.3	54.3	71.8	3.6	22.5	51.1
Less variable case mix	No update	47	2.7	0.2	89.5	56.3	19.4	88.3	76	58.8
	Intercept correction	6.7	2.5	0.1	1.6	7.7	9.9	1.1	4.7	5.6
	Linear recalibration	11.3	3.6	1.2	1.2	7	11.1	2.4	1.7	4.1
	Flexible recalibration	0.8	0.4	0	1.4	1.4	4.7	1.2	1.5	3.7
	Model refitting	34.2	90.8	98.5	6.3	27.6	54.9	7	16.1	27.8
More variable case mix	No update	30.8	0.1	0	81.6	28.3	4.8	93.1	54.5	18.3
	Intercept correction	2.3	0.1	0	2.6	4.2	2.5	1.2	3.6	5.5
	Linear recalibration	3.7	0.6	0	4.1	2.2	0.4	0	2.5	4.5
	Flexible recalibration	0.3	0.1	0	0.2	0.7	2.3	0	1.3	2.4
	Model refitting	62.9	99.1	100	11.5	64.6	90	5.7	38.1	69.3
Association changes	No update	0	0	0	0	0	0	0	0	0
	Intercept correction	0	0	0	0	0	0	0	0	0
	Linear recalibration	0	0	0	0	0	0	0	0	0
	Flexible recalibration	0	0	0	0	0	0	0	0	0
	Model refitting	100	100	100	100	100	100	100	100	100

Table 9. Percent of iterations for which each updating methods was recommended by Vergouwe *et al*'s closed testing procedure under each simulated scenario, training sample size (n_t), and updating sample size (n_u) when df = 40.

		n_t = 1000			**n_t = 5000**			**n_t = 10000**		
Scenario	**Updating method**	**n_u= 1000**	**n_u= 5000**	**n_u= 10000**	**n_u= 1000**	**n_u= 5000**	**n_u= 10000**	**n_u= 1000**	**n_u= 5000**	**n_u= 10000**
No population shift	No update	3.5	0	0	70.6	3.9	0	78.9	26.7	0
	Intercept correction	1.2	0	0	3.5	0.6	0	2.4	1.4	0
	Linear recalibration	1.2	0	0	0	0.6	0	0	2.3	0
	Flexible recalibration	1.2	0	0	1.2	0.3	0	0	1.3	0
	Model refitting	92.9	100	100	24.7	94.6	100	18.7	68.3	100
Increased event rate	No update	0	0	0	4.7	0	0	9.5	0	0
	Intercept correction	3.5	0	0	60.8	2.3	0	80.9	24.5	7.2
	Linear recalibration	1.1	0	0	2.3	1	0	1.2	2.2	0
	Flexible recalibration	0	0	0	2.3	0.7	0	1.2	1.2	0
	Model refitting	95.4	100	100	29.9	96	100	7.2	72.1	92.8
Less variable case mix	No update	10.4	0	0	72.6	8.9	0	87.1	37.1	0
	Intercept correction	1.2	0	0	3.5	6.5	0	2.4	9.6	0
	Linear recalibration	10.6	0	0	2.4	5.9	2.7	1.1	5.3	3.6
	Flexible recalibration	1.1	0	0	1.2	0.7	0	0	1.2	0
	Model refitting	76.7	100	100	20.3	78	97.3	9.4	46.8	96.4
More variable case mix	No update	0	0	0	24.4	0	0	58.5	2.9	0
	Intercept correction	0	0	0	12.9	0.1	0	8.3	1.3	0
	Linear recalibration	0	0	0	3.5	0	0	4.7	0.9	0
	Flexible recalibration	0	0	0	2.4	0	0	1.2	0.1	0
	Model refitting	100	100	100	56.8	99.9	100	27.3	94.8	100
Association changes	No update	0	0	0	0	0	0	0	0	0
	Intercept correction	0	0	0	0	0	0	0	0	0
	Linear recalibration	0	0	0	0	0	0	0	0	0
	Flexible recalibration	0	0	0	0	0	0	0	0	0
	Model refitting	100	100	100	100	100	100	100	100	100

sample of just 1,000 observations in 18.7% of iterations when 10,000 observations had been used to train the df = 40 model. In such cases, no improvement in performance as a result of this more complex updating was observed (see Figures 31-32).

When the outcome prevalence was increased in the updating population, intercept correction was frequently recommended in those cases where $n_u \leq n_t$. An exception to this pattern emerged for the df = 40 model, for which refitting was recommended when updating sample sizes were equal to or larger than the training sample. When n_t and n_u were both set to 10,000 with the df = 10 model, the test was split between intercept correction and refitting. These recommendations did not improve performance beyond that which would have been achieved by always selecting intercept correction (see Figures 33-34).

As observed with our testing procedure, case mix shifts resulted in the most variable recommendations. For both the df = 10 and df = 40 models, refitting was the main recommendation when n_t = 1000 and updating samples were larger. For the scenario involving a more heterogenous case mix in the updating population, similar patterns were observed, particularly for the df = 10 model. With the df = 40 model, refitting was exclusively recommended when $n_u \geq n_t$. Discrimination and calibration were not significantly improved by these updating recommendations compared to the original model or less complexly updated models, with the exception of models trained on samples of n_t= 1000 (see Figures 37-40).

Vergouwe *et al*'s testing procedure exclusively recommended refitting the model when predictor-outcome associations had shifted between the training and updating populations, regardless of the relative sizes of the training and updating samples. Refitting the model after predictor-outcome association shifts improved accuracy compared to simpler updating methods, even in cases when $n_u < n_t$ (see Figure 35-36).

Case Studies

Methods

As illustrative examples on clinical data, we applied our procedure to two logistic regression models, one for 30-day all-cause mortality after hospital admission and another for hospital-acquired acute kidney injury (AKI). Each model was developed and updated with data on a national set of inpatient admissions to Department of Veterans Affairs facilities.[18, 19] Predictors, which were selected based on existing models from the literature, included demographics, vital signs, medications, laboratory values, diagnoses, admission characteristics,

Table 10. Case study populations.

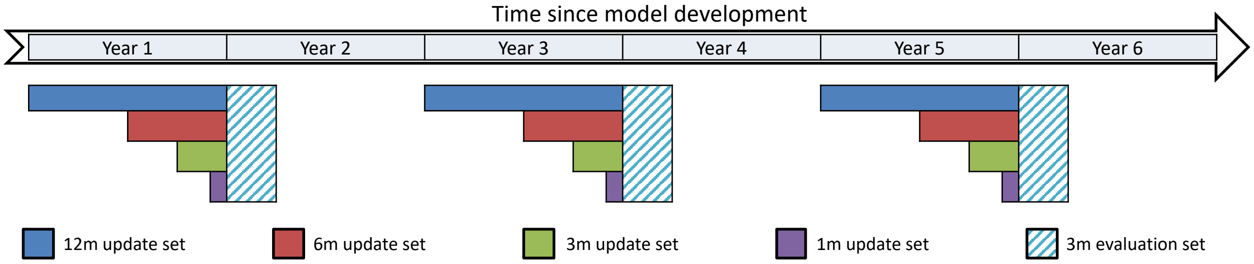
	Acute kidney injury dataset	**30-day mortality dataset**
Number of admissions	1,841,951	1,893,284
Study period	2003 - 2012	2006-2013
Outcome rate (%)	6.8	4.9
Age in years (mean and SD)	66.1 (13.0)	63.4 (14.0)
% Female	3.9	5.0
Race		
% White	75.3	72.1
% Black	19.3	19.7
% American Indian/Alaskan	0.9	1.2
% Asian/Pacific Islander	1.1	1.5
% Unreported	3.4	5.5
BMI at admission (mean and SD)	27.7 (7.5)	28.5 (7.2)

and healthcare utilization.[7, 8, 106-108] Both cohorts consisted of inpatient admissions to VA facilities that lasted at least 48 hours and for which the patient was at least 18 years of age. Admissions were excluded if the patient received hospice care or was admitted to a facility with fewer than 100 admissions per year or did not report key data to the central data warehouse. Outcome-specific data definitions and additional exclusion criteria were previously reported.[18, 19] The datasets and study population characteristics are summarized in Table 10.

Logistic regression models for both AKI and mortality experienced documented performance drift across several years.[18, 19] Drift of the AKI model accelerated four years after model development due to a complex mix of event rate, case mix, and predictor-outcome association changes.[19] Performance of the mortality model drifted more consistently over seven years as a result of steady event rate and case mix shifts.[18] We applied our testing procedure to assess the need for updating at multiple timepoints after development of each model. This study was approved by the Institutional Review Board and the Research and Development committee of the Tennessee Valley Healthcare System VA.

An illustration of the updating and evaluation framework for these case studies is provided in Figure 41. The mortality model was trained on admissions from 2006 (n=235,548)

Figure 41. Updating and evaluation scheme for case studies evaluating the nonparametric testing procedure.



and the AKI model on admissions from 2003 (n=170,675). We updated both models one, three, and five years after development, defining updating points at the end of 2007, 2009, and 2011 for the mortality model and at the end of 2004, 2006, and 2008 for the AKI model. Calibration of the mortality model steadily declined across this period,[18] whereas performance drift of the AKI model accelerated four years after development.[19] The 2003 AKI and 2006 mortality models were considered for updating at each time point without consideration of any prior updating recommendations. We applied the testing procedure with multiple definitions of the updating cohort, constructing updating sets with admissions in the prior 1, 3, 6, and 12 months. For simplicity, we refer to the 12-month updating set as a large update set, the 1-month updating set as a small update set, and the 3- and 6-month updating sets as moderate update set.

We documented performance of the original and updated models on a prospective evaluation set of admissions in the 3-months after each updating point, reflecting the notion that an updated model would ideally perform well immediately after updating. Calibration curves[109] were constructed for the original model and each updating method to characterize the 95% confidence interval of performance across the range of probabilities. Common discrimination and calibration metrics were also calculated.

For baseline comparison with the most applicable method in the literature, we also applied Vergouwe and colleagues' closed testing procedure.[33] As in the simulation study, we extended this testing procedure to incorporate flexible logistic recalibration.[66] We explored differences in updating recommendations and the impact of these recommendations on subsequent prospective performance.

Results

The updating recommendations of our nonparametric testing procedure and Vergouwe *et al*'s closed testing procedure are documented in Tables 11 and 12. Performance metrics for each updating method in the three months after updating are displayed in Figures 42 and 43, and corresponding calibration curves are presented in Figures 44-47. For clarity, we limited the detailed results in the main text to performance based on large and small update sets for the one and five year updating points. Full results are available in Appendix E.

For the AKI model, intercept correction was the most complex updating method recommended by our testing procedure. This change was recommended at all time points with large update sets, at most updating points with moderate update sets, and after 5 years with a small update set. One year out from model development, recalibration with intercept correction did not significantly improve calibration among inpatient admissions in the three months after updating (see upper left panel of Figure 44). Five years after model development, intercept correction with the large update cohort, as recommended, improved calibration over the original model and provided similar calibration to that of the refitted model. With small update sets, the calibration curves reflect poorer calibration after refitting compared to both the original and

Table 11. Updating recommendations of our nonparametric testing procedure by time since model development and size of updating set.

Time from development to updating	Large (12m) update set	Moderate (6m) update set	Moderate (3m) update set	Small (1m) update set
Acute kidney injury				
1 year	Intercept correction	Intercept correction	No update	No update
3 years	Intercept correction	Intercept correction	Intercept correction	No update
5 years	Intercept correction	Intercept correction	Intercept correction	Intercept correction
30-day mortality				
1 year	Flexible logistic recalibration	Flexible logistic recalibration	Flexible logistic recalibration	No update
3 years	Flexible logistic recalibration	Flexible logistic recalibration	Intercept correction	No update
5 years	Flexible logistic recalibration	Flexible logistic recalibration	Flexible logistic recalibration	Intercept correction

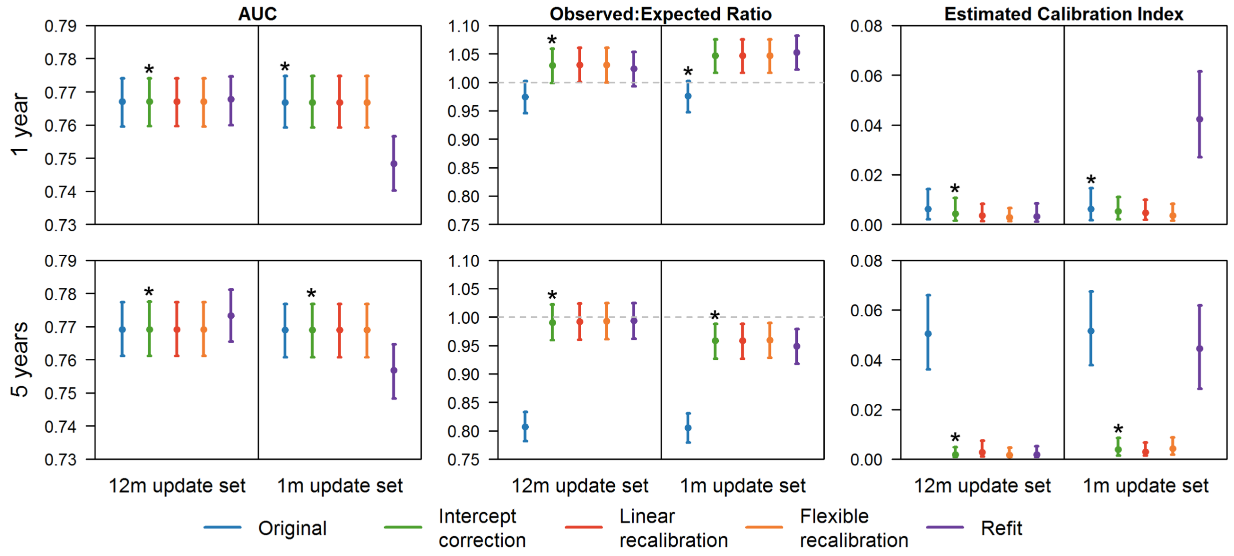
Table 12. Updating recommendations of Vergouwe *et al*'s closed testing procedure by time since model development and size of updating set.

Time from development to updating	Large (12m) update set	Moderate (6m) update set	Moderate (3m) update set	Small (1m) update set
Acute kidney injury				
1 year	Refit	Refit	Refit	Refit
3 years	Refit	Refit	Refit	No update
5 years	Refit	Refit	Refit	Refit
30-day mortality				
1 year	Refit	Refit	Refit	No update
3 years	Refit	Refit	Refit	Intercept correction
5 years	Refit	Refit	Refit	Refit

intercept-corrected models (see also Figure 42). At the five-year update point, intercept correction with the small update cohort, as recommended, improved upon the calibration of the original and refitted models. In each case, more complex recalibration than recommended did not provide additional performance improvements (see Figure 42).

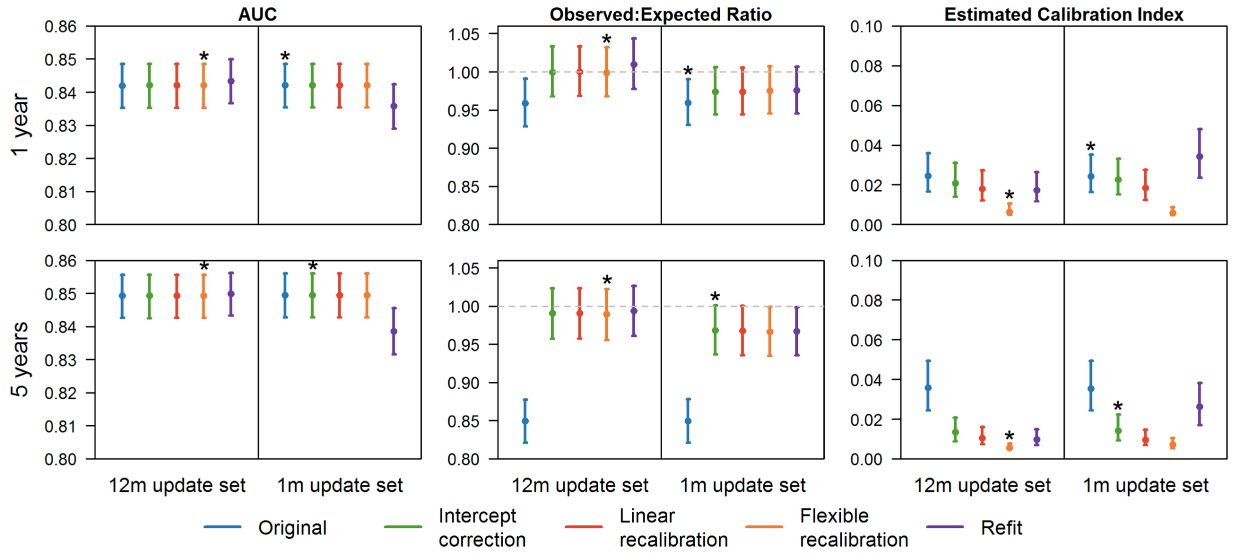
For the 30-day mortality model, large and moderate update sets prompted our nonparametric testing procedure to recommend flexible logistic recalibration both soon after model development and as time passed. The calibration curves associated with flexible logistic recalibration highlight well-calibrated predictions over a wide range of probabilities, while the curves for the original and refitted models highlight uncertainty and overprediction as predicted probabilities increase (see Figure 46). Simpler recalibration approaches did not provide as much improvement in performance as the recommended flexible logistic recalibration (see Figures 43 and 47). With small update sets and more time since model development, our testing procedure recommended intercept correction. Although calibration was somewhat improved with this update compared to the original and refitted models, the calibration curves reflect uncertainty in performance after updating which could have been further improved upon by more complex recalibration in this case (see Figures 43 and 47).

Figure 42. Performance of the acute kidney injury (AKI) model in the three months after updating with each updating method using large (12-month) and small (1-month) update sets.



* Recommended update per our nonparametric testing procedure

Figure 43. Performance of the 30-day mortality model in the three months after updating with each updating method using large (12-month) and small (1-month) update sets.



* Recommended update per our nonparametric testing procedure

Figure 44. Calibration curves in the three months after updating based on large (12-month) and small (1-month) update sets for the original acute kidney injury (AKI) model, the refit model, and the update recommended by our nonparametric testing procedure (if different; e.g., no update to original model recommended for bottom left panel).

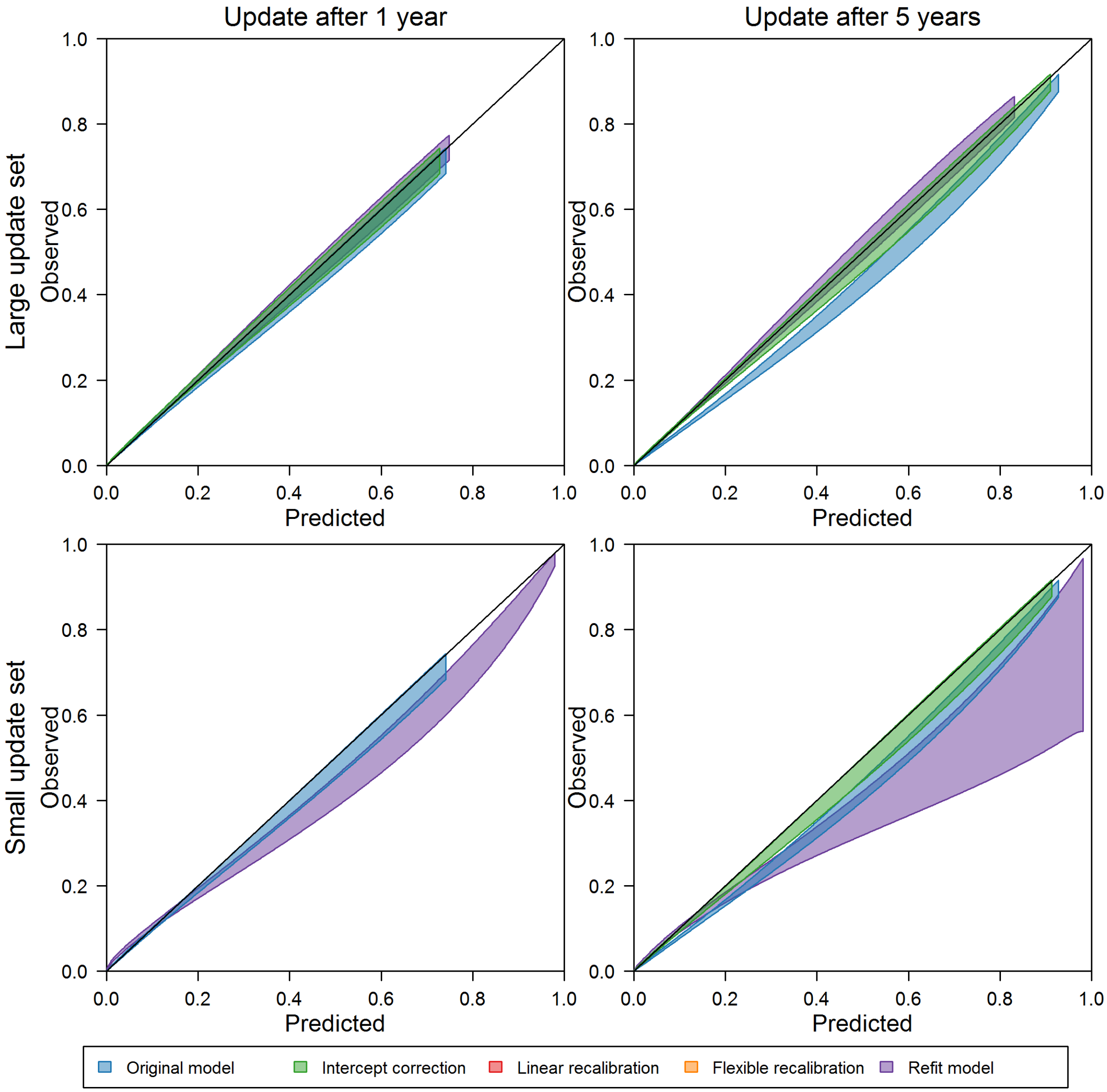


Figure 45. Calibration curves in the three months after updating based on large (12-month) and small (1-month) update sets for the acute kidney injury (AKI) model with varying degrees of recalibration.

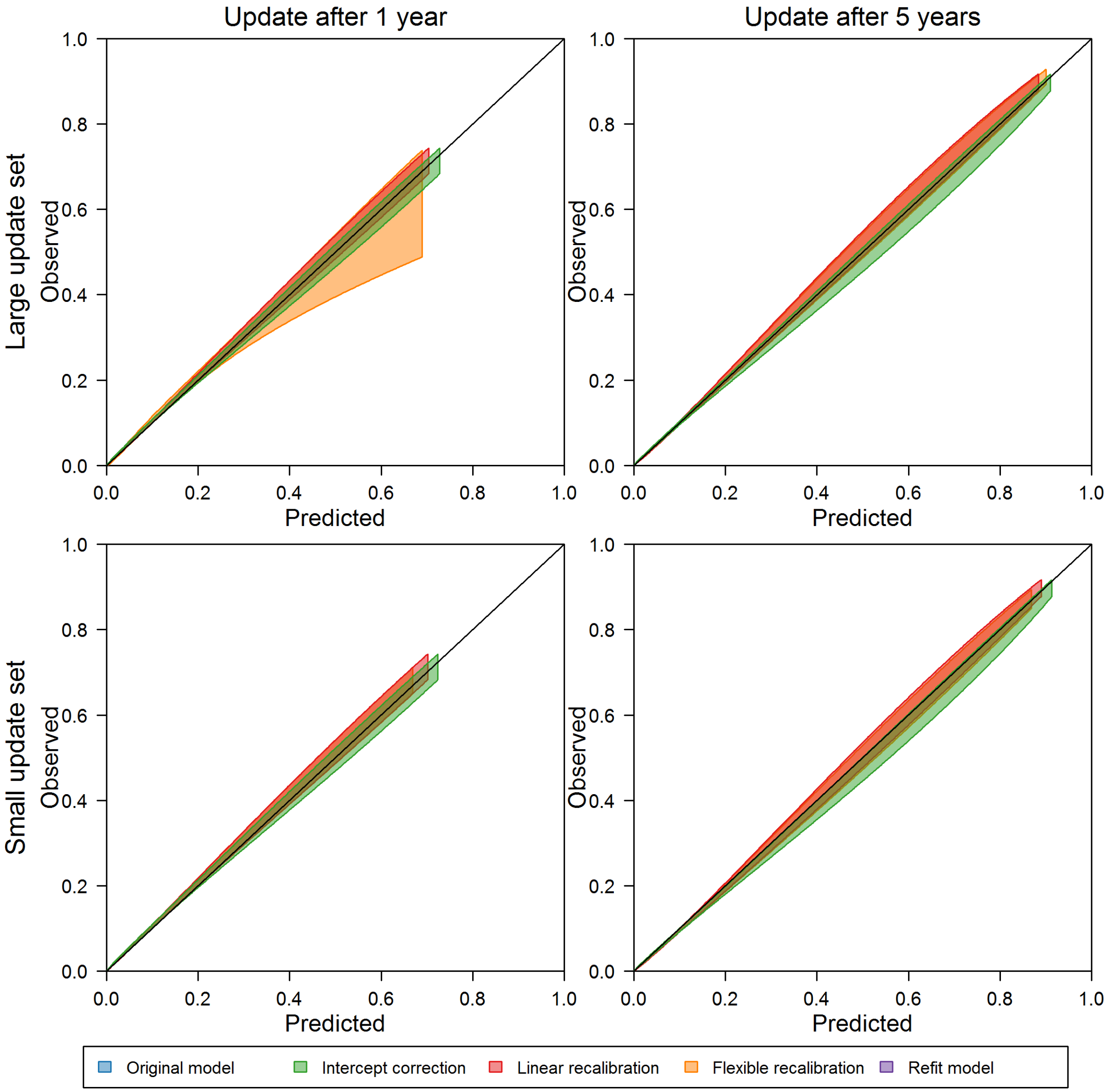


Figure 46. Calibration curves in the three months after updating based on large (12-month) and small (1-month) update sets for the original 30-day mortality model, the refit model, and the update recommended by our nonparametric testing procedure (if different; e.g., no update to original model recommended for bottom left panel).

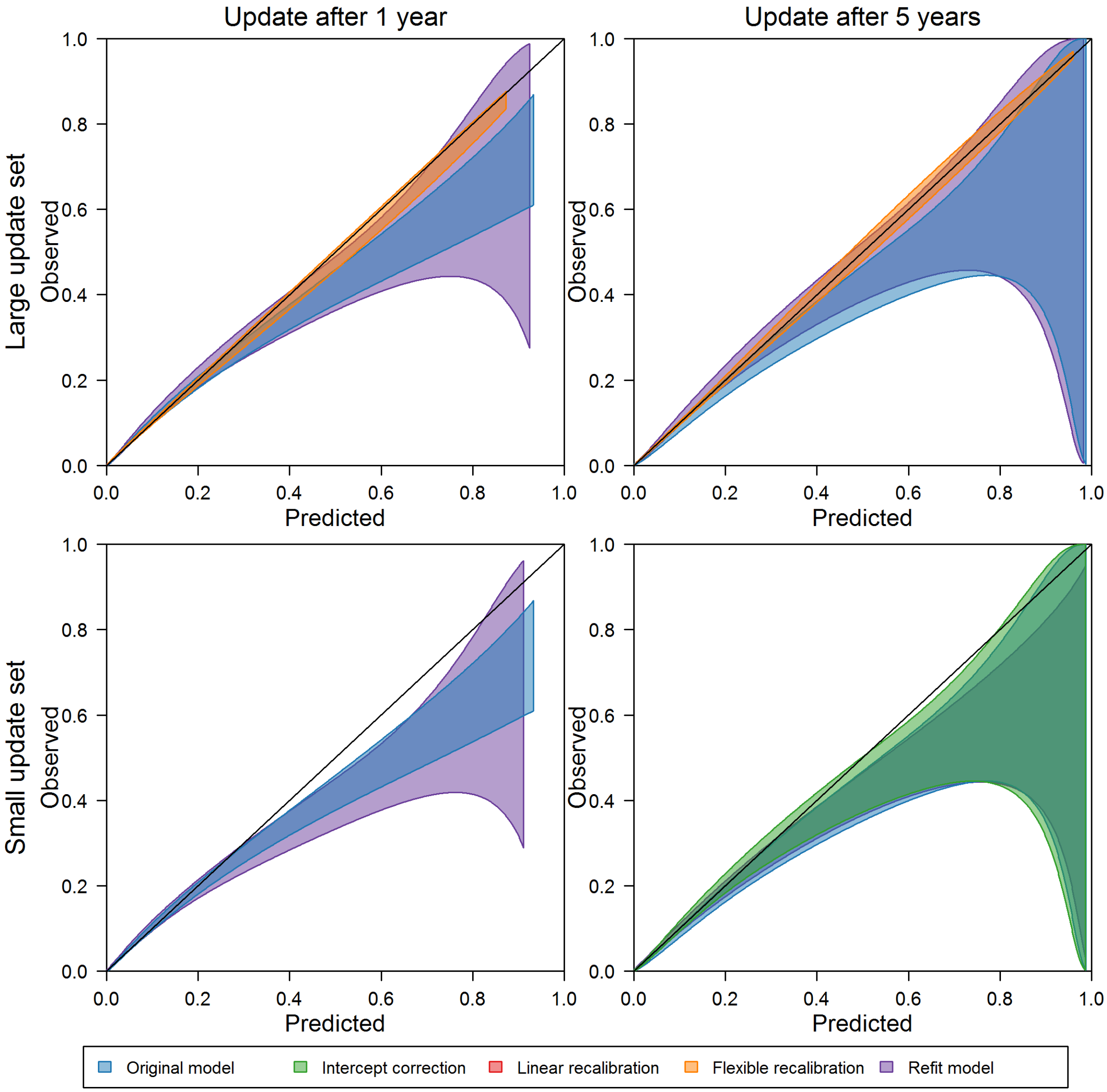
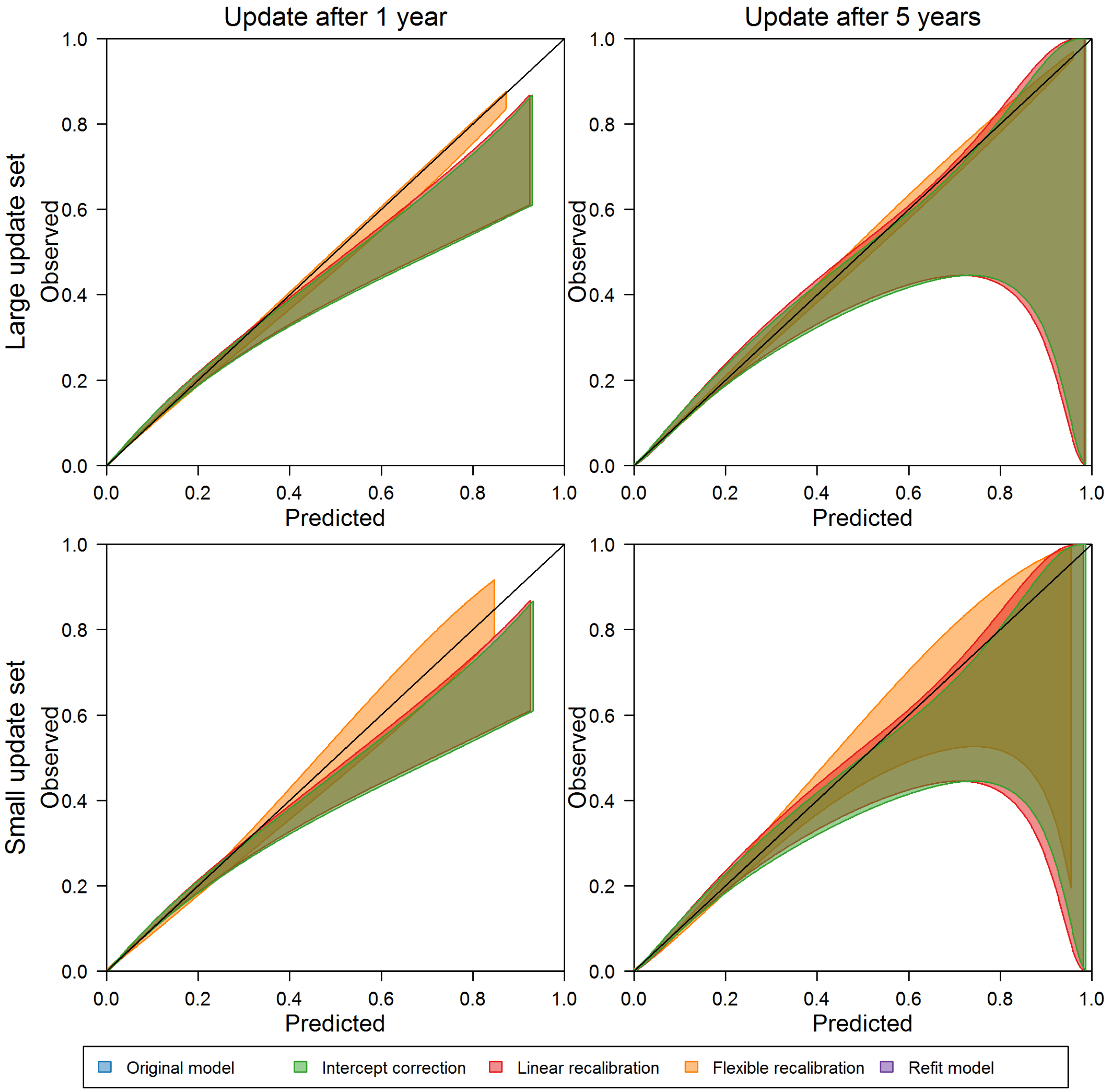


Figure 47. Calibration curves in the three months after updating based on large (12-month) and small (1-month) update sets for the 30-day mortality model with varying degrees of recalibration.



In contrast, Vergouwe *et al*'s closed testing procedure recommended refitting the AKI model in all but one case (see Table 12). Refitting did not significantly improve upon recalibration and in some cases resulted in diminished performance compared to simpler approaches (see Figure 42). The detrimental effect of refitting the AKI model with a 1-month update set is highlighted by the calibration curves in Figure 46, where the refit curve falls farther from the 45° calibration line than those of the original and intercept-corrected models.

Vergouwe *et al*'s closed testing procedure recommended refitting the 30-day mortality model with large and moderate update sets at all timepoints. With small update sets, this test recommended an increasing degree of updating over time, suggesting no update, intercept correction, and refitting after one, three, and five years, respectively. Refitting the model five years after development using one month of admissions resulted in inferior performance compared to recalibration and did not improve upon the original model (see Figure 43). With larger update sets, refitting, as recommended, did not significantly improve upon simpler recalibration approaches (see Figure 43).

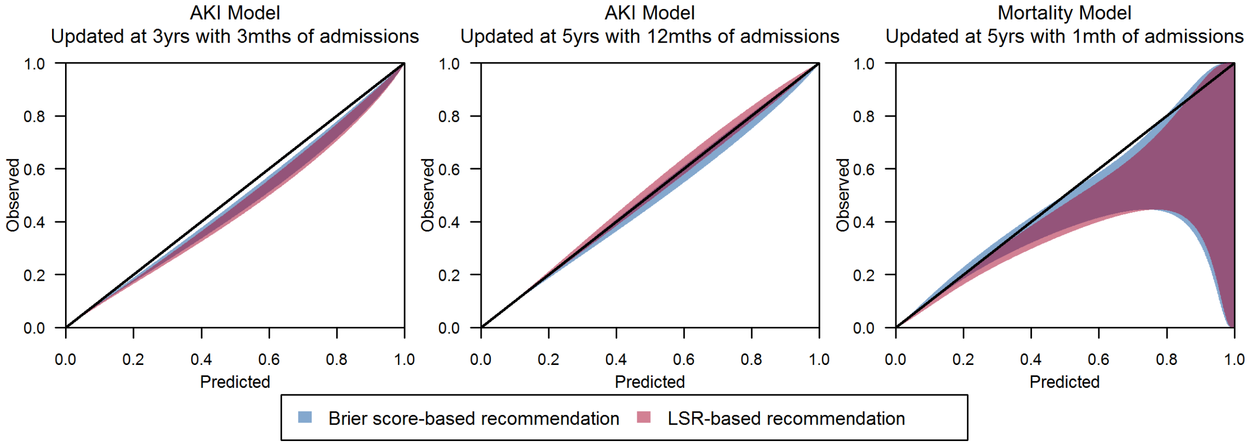
Sensitivity Analysis

One of the strengths of our nonparametric testing procedure is its customizable nature. Some users may prefer to focus on calibration metrics, utility metrics, or alternate accuracy metrics. To illustrate this customization and assess how the choice of accuracy metric impacts test recommendations, we ran both the simulation and case study analyses using a logarithmic scoring rule (LSR) rather than the quadratic Brier score. This change required adjusting the metric recorded in the second bootstrapping stage and defining M_r as the updating method that maximized rather than minimized the median $LSR_{i,k}$.

Table 13. Updating recommendations across all simulation iterations using the Brier score and logarithmic scoring rule.

		Brier score decision				
		No Update	Intercept correction	Linear recalibration	Flexible recalibration	Model refitting
LSR decision	No update	43,462	391	354	0	295
	Intercept correction	1,101	13,024	73	0	148
	Linear recalibration	246	197	432	1	1,104
	Flexible recalibration	0	3	3	4	27
	Model refitting	459	596	361	23	27,696

Figure 48. Calibration curves for case study scenarios in which the testing procedures evaluating the Brier score and logarithmic scoring rule provided differing recommendations.



In both the simulation and case studies, we observed strong agreement between the test recommendations based on the Brier and logarithmic scores. Across all simulation scenarios and parameter combinations, the two scores lead to the same recommendation 94.0% of the time (see Table 13). Within population shift scenarios, agreement ranged from 92.0% when the outcome prevalence was changed to 96.1% when no population shift occurred.

For the AKI model, using a 3-month update set at three years after development the Brier score-based test recommended intercept correction and the LSR-based test recommended retaining the original model. Using a 12-month update set after five years, the LSR-based test recommended model refitting, a more complex change than intercept correction as recommended by the Brier score-based test. Five years after the mortality model was developed, using one month of admissions in the updating process lead the LSR-based test to recommend retaining the original model, whereas the Brier score-based test recommended intercept correction. Figure 48 illustrates the differences in calibration curves for the three months after updating with these recommendations. The differing recommendations resulted in similar performance. For example, in the center panel of Figure 48, the calibration curves were almost mirrored across the ideal calibration line, with the Brier score-based update having a slight tendency toward overprediction and the LSR-based update having a slight tendency toward underprediction. However, both updated models were calibrated across the range of predictions. At other updating points and updating sample sizes, recommendations for updating the mortality and AKI models were unchanged.

Discussion

We described and evaluated a new nonparametric testing procedure to recommend prediction model updating methods that minimizes overfitting, accounts for uncertainty associated with updating sample size, incorporates a preference for simple updating, and is applicable regardless of the learning algorithm generating predicted probabilities. This testing procedure supports clinical prediction models underlying informatics applications for decision support, population management, and quality benchmarking, both when transporting such models to a new setting or applying them over time in evolving clinical environments.

As is desirable based on statistical theory, the testing procedure displayed a preference for more complex updates as the updating sample size increased or as the training and updating populations became increasingly disparate (see Table 14 for general patterns of recommendations). Our findings reflect the concept that when more information is available to support updating (i.e., update sets are large), more complex updating may be appropriate. For example, when our simulated update set was 10 times larger than the training set, our testing procedure most often recommended refitting the model, even in scenarios for which there were no differences between training and updating populations. This pattern is reassuring as we intuitively would want to refit a model when substantially more information is available for learning associations, even if we do not expect associations to have changed. For models trained on small datasets, updating recommendations varied when update sets were equally small, highlighting uncertainty in both the original and adjusted models. Generally, in those situations in which the updating and training sets provided the same volume of data, no population shift resulted in recommendations to retain the original model, case mix shift resulted

Table 14. Patterns of our nonparametric testing procedure's updating recommendations.

Situation	Most common recommendation
Changes in outcome prevalence, similar volumes of training and updating data	Intercept correction
Changes in case mix, similar volumes of training and updating data	No updating or recalibration
Changes in predictor-outcome associations, regardless of sample sizes	Refit the model
Updating set substantially larger than training set	Refit the model

in recommendations to retain the original model or conduct modest recalibration, and outcome rate shifts resulted in recommendations of intercept correction. Reassuringly, when predictor-outcome associations shifted between training and updating populations, our testing procedure predominantly recommended refitting regardless of the training or updating sample sizes.

The case study results similarly highlighted recommendations for more complex updating as population shift increased and updating sample sizes grew. With large updating samples, our testing procedure recommended updating at each timepoint considered and suggested more complex recalibration compared to recommendations based on smaller updating samples. As calibration of the original models deteriorated over time, our testing procedure responded by recommending recalibration even when updating samples were limited. By recommending recalibration to varying degrees rather than refitting the models, the testing procedure allowed us to avoid refitting in cases where this more data-intensive updating approach would have provided no benefit to or even harmed prospective performance.

Our testing procedure's recommendations were frequently different from those provided by Vergouwe and colleagues' closed testing procedure. Vergouwe *et al*'s testing procedure commonly recommended model refitting, even for simulations involving no population shifts and update sets substantially smaller than training sets. These refitted models resulted in either similar or inferior performance to that achieved through recalibration. This highlights the importance of controlling for overfitting and avoiding the assumption that model refitting is the ideal updating methods against which other methods should compete.

We have described our testing procedure as filtering based on any statistically significant difference in performance between updating methods. However, the procedure can be adjusted to filter based on clinically significant differences in performance between updating methods. This would be achieved by adjusting the final step of the procedure to consider a user-specified minimum difference in S between M_r and simpler updating methods. Although we have used 0 to find any difference in accuracy, small differences in the scoring metric may not be associated with clinically meaningful differences in performance and may result in users questioning the value of updating. Methods for defining clinically meaningful differences in performance as measured by various scoring rules remain an open area of research.

While our analyses focus on logistic regression models, the testing procedure is applicable to any categorical model, as the method makes no assumptions regarding the learning approach and relies only on observed and predicted values. In addition to applications involving other dichotomous outcome models, extension to multiclass models is straightforward by providing an appropriate scoring rule (e.g., the multiclass definition of the Brier score).

Although the updating methods implemented in these analyses are generalizable to models regardless of underlying learning algorithms, users may tailor the set of updating methods to be considered based on use case-specific needs and preferences. The only requirement for defining a custom set of updating methods is that users provide an order of complexity/preference among the included methods, which may require careful consideration for cases lacking a natural ordering. Some users may also prefer to optimize a different scoring metric than that implemented in these analyses. Such an adjustment is easily made by replacing the Brier score in the second bootstrapping stage. In sensitivity analyses using the logarithmic scoring rule, we observed strong agreement with the test recommendations based on the Brier score.

Our testing procedure supports periodic updating of static models. Alternatively, online learning algorithms continuously update as new observations become available, incorporating changes in the environment as they occur.[28, 67, 68] Such models have been applied to health outcomes, but have yet to be incorporated into clinical tools.[27, 28, 63] The shift to an online paradigm is not straightforward for clinical use cases, as new validation methods are required [28, 67] and the regulatory framework for implementing dynamic models is evolving.[69]

There are several key limitations of our testing procedure and the evaluations presented here. The case study highlights a conservative nature to our testing procedure, which may be a limitation for certain use cases. When only a small sample was available to update the 30-day mortality model, our testing procedure recommended not updating one year after model development and only minimal recalibration after five years. Calibration assessment based on admissions in the three months following these update points indicated that flexible recalibration, as recommended with larger update sets, could have provided additional improvement in calibration. While we view the decision to recommend less complex updating as a benefit given the relatively small size of the updating set in this example, the requirements of some use cases may view any improvement in calibration to be desirable and the test's recommendation as a missed opportunity. As a second limitation, the first bootstrapping stage may be computationally expensive, particularly for complex models. Although advancements in computational resources continue to reduce computation times, a refinement to the number of iterations in the first bootstrap stage may be warranted. Finally, updating per the test's recommendation, or any of the considered methods, may not result in sufficient improvement in model performance to warrant continued clinical application of the model. Users should evaluate performance of updated model to determine if clinically acceptable performance is achieved or whether model extension or alternative models may be required.

Conclusions

As clinical prediction models continue to be developed and deployed in complex, ever-changing environments, maintenance of these models will become increasingly crucial to their utility. Models underlying population health management, quality assessment, and clinical decision support applications require a high degree of accuracy and developers must be responsive to any degradation in performance. We described a new testing procedure to support data-driven updating of categorical prediction models, with the intent to increase the long-term sustainability of those models in a continuously evolving clinical environment. Our procedure encourages small corrections when only a small amount of new data is available, and graduates to recommending full model retraining when the new dataset is large enough to support it. The procedure is applicable to models developed with either biostatistical or machine learning approaches, and is customizable to user needs and preferences.

CHAPTER 7

COMPARISON OF SCHEDULED PREDICTION MODEL PERFORMANCE UPDATING PROTOCOLS

In this chapter, we extend our investigation of the nonparametric testing procedure and highlight a key strength of the method by applying the testing procedure to four common learning algorithms – ordinary logistic regression, L1-regularized logistic regression (i.e., lasso), random forest, and neural network. Guidance on the design of model updating policies is limited, and there is limited exploration of the impact of different policies on future model performance and across different model types. We address this knowledge gap by exploring whether the long-term performance of clinical prediction models is improved through a data-driven approach to scheduled model maintenance. We compare three annual updating strategies—retention of the original model, predefined model refitting, and application of recommendations of the nonparametric testing procedure. We assess differences in discrimination and calibration over time under each updating strategy, as well as whether and how the learning algorithm underlying the model impacts updating requirements and accuracy. Please note, large portions of this chapter will be published in the 2019 Proceedings of the American Medical Informatics Association Annual Symposium.[36]

Methods

Study Population and Initial Models

Here we expand on the case studies described in Chapter 6 by exploring performance drift and model updating for learning algorithms beyond ordinary logistic regression and across sequential updating points. See pages 83-84 for a brief overview of the datasets. This study was approved by the Institutional Review Board and the Research and Development committee of the Tennessee Valley Healthcare System VA.

We developed models for hospital-acquired AKI and 30-day mortality after hospital admission among patients admitted to Department of Veterans Affairs facilities using four common learning algorithms—logistic regression (LR), L1-regularized logistic regression (L1), random forests (RF), and neural networks (NN). For each outcome, models were developed using a common set of predictors and a single year of admissions data (2003 for AKI and 2006 for mortality). Information on admissions in subsequent years (2004-2012 for AKI and 2007-2013 for mortality) were collected for both updating and validation. Previous work revealed both the AKI and mortality models experiences performance drift across these study periods, with the timing and extent of drift varying by learning algorithm (see Figures 1 and 2).[18, 19] The LR and L1 models were most susceptible to calibration drift, with consistent deterioration over time of the mortality model and deterioration of the AKI model that accelerated four years into the validation period.[18, 19] Similar patterns of performance drift were observed for the corresponding RF models, although to a lesser degree than the LR and L1 models.[18, 19] For the mortality model, the NN model did not experience significant changes in calibration over time.[18] Complex combinations of decreasing event rates, evolving patient case mixes, and changing predictor-outcome associations were associated with these performance patterns.[18, 19]

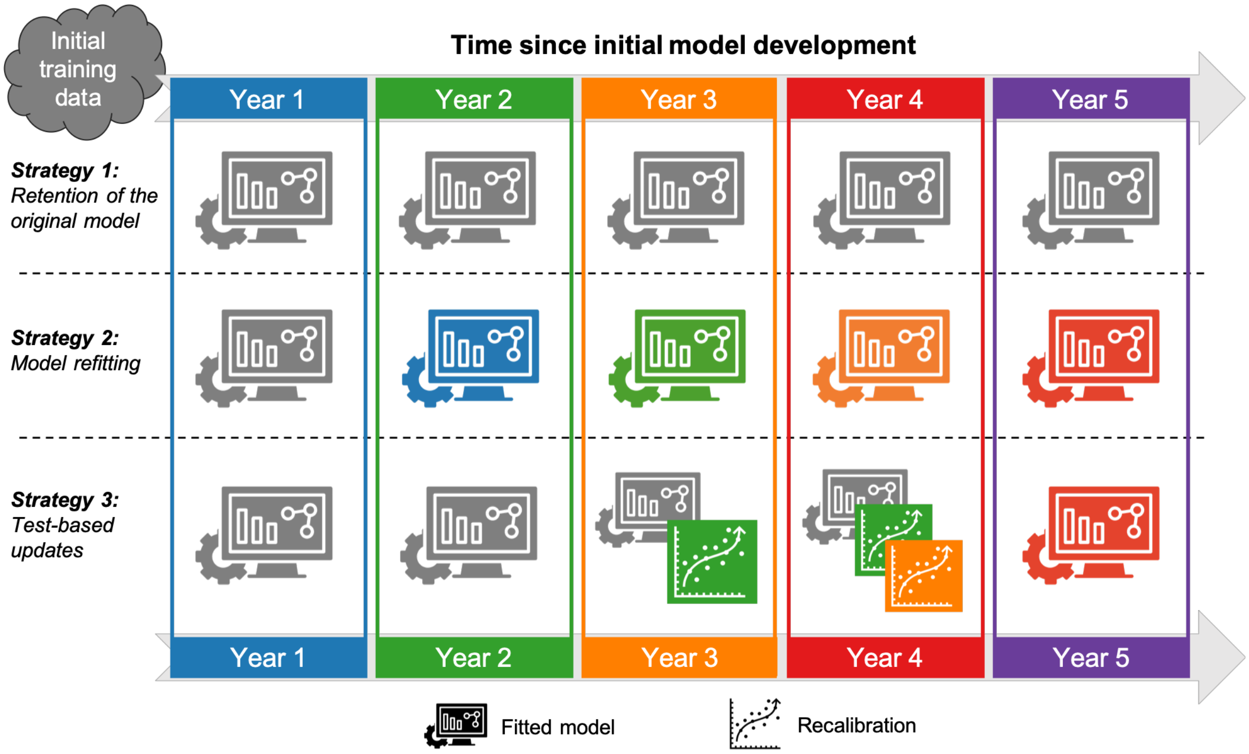
Scheduled Updating Strategies

Following a common model maintenance timeline in practice,[27, 31, 32] updating was undertaken for all models on an annual basis for each year following model development. Updating was based on admissions accrued over the prior 12 months and applied to admissions in the following 12 months. For the AKI model, this resulted in updating points at the end of 2004 through 2011, with admissions in 2012 serving as the validation data for the 2011 updates. For the mortality model, updates occurred at the end of 2007 through 2012, with 2013 admissions serving as the validation set for the final update.

Across these scheduled updating points, we implemented three competing strategies to update the LR, L1, RF, and NN models over time. An illustrative overview of these updating strategies is presented in Figure 49, highlighting the data underlying each model and the data on which each model was applied. As a baseline, we retained the original models developed on the first year of admissions and applied these models to all subsequent admissions. The second updating strategy called for annually refitting each model using all admissions that accrued over the prior 12 months. Hyperparameters for the L1, RF, and NN models were tuned annually using 5-fold cross-validation. Admissions in each year were assigned predicted probabilities based on the prior year's models. The third strategy selected an updating approach for each model based on our nonparametric testing procedure.[35] The testing procedure was implemented using the Brier score to selected between the retention of the current model, intercept correction, linear logistic recalibration, flexible logistic recalibration, or model refitting. For these analyses, we did not consider any additional model-specific updating approaches.

Updating sequences were retained over multiple years as needed, allowing updates to build on any prior adjustments to the model (see Figure 49). For example, a model based initially on Year 0 admissions was applied to Year 1 admissions. At the end of Year 1, the testing procedure recommended either continued use of the existing model, adjustment of the existing model through recalibration, or replacement of the model with a newly refit model. This updated version of the model was used to generate predictions for Year 2. Following Year 2, the testing procedure considered whether any additional updates to the model as adjusted after Year 1 were warranted, not whether to adjust the original Year 0 model. If additional updating was recommended by the test, those changes were applied in addition to the existing Year 1 adjustments. At any point, if the test recommended refitting the model, then all previous models and sequences of adjustments were replaced by a new model moving forward.

Figure 49. Overview of updating strategies applied over multiple updating points. Icons indicate the active model applied to observations in each time period and are color-coded to correspond with the data on which the model was built/updated. In this example, the test-based strategy recommended recalibration at the end of Year 2 and Year 3, as well as model refitting at the end of Year 4.


Initial training data
Time since initial model development
Year 1
Year 2
Year 3
Year 4
Year 5
Strategy 1: Retention of the original model
Strategy 2: Model refitting
Strategy 3: Test-based updates
Year 1
Year 2
Year 3
Year 4
Year 5
Fitted model
Recalibration

Evaluation of Scheduled Updates

We assessed the influence of each updating strategy on the long-term performance of the LR, L1, RF, and NN models. Under each updating strategy, models were assessed for both discrimination (area under the receiver operating curve, AUC[110]) and calibration (calibration curves, observed to expected outcome ratio, Cox intercept and slope, and estimated calibration index).[37, 39, 109] We evaluated overall and monthly performance of the models under each strategy across the entire validation and updating period. We further compared the updating requirements of the different learning algorithms based on the extent and timing of test-recommended model adjustments.

Results

Test-Based Updating Recommendations

Test-based updating recommendations for all models are noted in Tables 15 and 16. In each case, some adjustment of the original model was recommended after one year. The degree of recommended updating at this first updating point varied from intercept correction to model refitting. Following this initial update, the sequence of updating recommendations varied by outcome and learning algorithm, both in terms of timing and method.

For the AKI models, each model experienced periods during which annual updating was recommended and other periods during which no additional updates were recommended for a number of years. After initial recalibration at the end of 2004, the testing procedure advised periodic intercept correction for both the NN and RF models, although the timing of these additional corrections did not align. More substantial adjustments were undertaken with the LR and L1 models, with multiple instances of recalibration and each model being refit once during the study period.

After initial updates, the testing procedure generally recommended less frequent updating of the models for 30-day mortality. The NN model was an exception to this pattern, with annual model refitting being recommended. For the L1 model, the recalibration adjustments incorporated after the first year were maintained until the fifth year after model development, at which point the testing procedure recommended an additional intercept correction. The RF model was further updated only in the third year after model development. Continued updating across the study

Table 15. Annual updating recommendations for the acute kidney injury models by learning algorithm.

Update set	LR	L1	NN	RF
2004 admissions	Intercept correction	Intercept correction	Linear logistic recalibration	Flexible logistic recalibration
2005 admissions	No change	Linear logistic recalibration	No change	No change
2006 admissions	Linear logistic recalibration	No change	No change	Intercept correction
2007 admissions	Flexible logistic recalibration	Intercept correction	Intercept correction	Intercept correction
2008 admissions	No change	Refit	No change	No change
2009 admissions	No change	Linear logistic recalibration	Intercept correction	No change
2010 admissions	Intercept correction	No change	No change	No change
2011 admissions	Refit	No change	No change	Intercept correction

Table 16. Annual updating recommendations for the 30-day mortality models by learning algorithm.

Update set	LR	L1	NN	RF
2007 admissions	Flexible logistic recalibration	Flexible logistic recalibration	Refit	Linear logistic recalibration
2008 admissions	No change	No change	Refit	No change
2009 admissions	Intercept correction	No change	Refit	Linear logistic recalibration
2010 admissions	No change	No change	Refit	No change
2011 admissions	Intercept correction	Intercept correction	Refit	No change
2012 admissions	No change	No change	Refit	No change

period was recommended for the LR model, with the testing procedure recommending some degree of recalibration every other year.

Model Performance Over Time

Performance of the four model variations under each updating strategy over the entire validation and updating period is reported in Tables 17 and 18 for AKI (2004-2012) and mortality (2007-2013), respectively. Discrimination was not significantly different across updating strategies ($p<0.05$), with the exception of the mortality NN model for which refitting (and the test-based strategy) increased the AUC from 0.77 to 0.80.

For each learning algorithm, annually refitting improved calibration compared to retaining the original model across the entire study period ($p<0.05$). Further improvement in calibration was achieved using test-recommended updates ($p<0.05$)—the exception being the NN mortality model for the test-based strategy reduced to refitting. Differences in calibration across updating strategies were highlighted by the calibration curves and most apparent when focusing on the lower risk portion of the curves where over 95% of observations occurred (see Figures 50 and 51). In calibration plots, perfect calibration is represented by the bisecting 45° line at which predicted probabilities equal observed proportions. For both outcomes, the refitting and test-based strategies corrected overprediction of the original models in the lower risk, shifting the calibration curves upward toward the bisector. Calibration curves for the mortality LR model under the test-based updating strategy captured more of the ideal calibration line than either the refit or original models. None of the updating strategies resulted in calibration across a large range of probabilities for the mortality RF model, and the calibration curves of all three strategies follow similar patterns. However, both the refitting and test-based updating strategies moved the calibration curves for this model closer to the bisector for the risk range where most observations fell. In the densely populated risk range, although the magnitude of miscalibration of the mortality L1 model was similar between the refitting and test-based updating strategies, the refitting approach erred toward underprediction, while the test-based strategy erred toward overprediction. Calibration curves for the AKI models under the test-based strategy captured more of the bisector than did corresponding curves under the refitting strategy. However, differences between these strategies were small in magnitude.

Table 17. Overall performance of acute kidney injury models by learning algorithm and annual updating strategy.

Model	Updating Strategy	AUC	O:E	Cox Intercept	Cox Slope	ECI
LR	No updating	0.764 [0.763, 0.766]	0.846 [0.841, 0.850]	-0.313 [-0.327, -0.299]	0.943 [0.937, 0.950]	0.036 [0.034, 0.039]
	Refitting	0.770 [0.768, 0.771]	0.973 [0.968, 0.978]	-0.085 [-0.100, -0.070]	0.976 [0.969, 0.982]	0.004 [0.003, 0.004]
	Test-based	0.766 [0.765, 0.768]	0.957 [0.952, 0.962]	-0.104 [-0.121, -0.089]	0.976 [0.969, 0.982]	0.004 [0.003, 0.005]
L1	No updating	0.759 [0.758, 0.761]	0.831 [0.826, 0.835]	-0.175 [-0.191, -0.159]	1.020 [1.012, 1.027]	0.028 [0.027, 0.030]
	Refitting	0.765 [0.763, 0.766]	0.978 [0.973, 0.983]	0.077 [0.051, 0.100]	1.046 [1.034, 1.056]	0.003 [0.002, 0.003]
	Test-based	0.762 [0.761, 0.764]	0.972 [0.967, 0.977]	-0.052 [-0.079, -0.03]	0.991 [0.979, 1.001]	0.002 [0.002, 0.002]
RF	No updating	0.737 [0.736, 0.739]	0.967 [0.962, 0.972]	-0.371 [-0.390, -0.351]	0.851 [0.843, 0.859]	0.008 [0.007, 0.009]
	Refitting	0.739 [0.737, 0.740]	1.063 [1.057, 1.069]	-0.337 [-0.359, -0.316]	0.825 [0.817, 0.834]	0.004 [0.003, 0.004]
	Test-based	0.738 [0.736, 0.739]	0.971 [0.966, 0.976]	-0.117 [-0.138, -0.097]	0.963 [0.955, 0.972]	0.002 [0.002, 0.002]
NN	No updating	0.722 [0.72, 0.723]	0.917 [0.912, 0.922]	-0.212 [-0.230, -0.196]	0.950 [0.942, 0.956]	0.010 [0.009, 0.012]
	Refitting	0.726 [0.724, 0.728]	0.991 [0.985, 0.996]	-0.124 [-0.143, -0.107]	0.951 [0.943, 0.958]	0.002 [0.002, 0.003]
	Test-based	0.722 [0.721, 0.724]	0.977 [0.971, 0.982]	-0.033 [-0.052, -0.015]	0.997 [0.989, 1.004]	0.001 [0.001, 0.002]

Table 18. Overall performance of 30-day mortality models by learning algorithm and annual updating strategy.

Model	Updating Strategy	AUC	O:E	Cox Intercept	Cox Slope	ECI
LR	No updating	0.849 [0.847, 0.850]	0.876 [0.871, 0.882]	-0.227 [-0.242, -0.214]	0.970 [0.964, 0.976]	0.029 [0.027, 0.032]
	Refitting	0.850 [0.849, 0.851]	0.981 [0.975, 0.987]	-0.052 [-0.068, -0.038]	0.987 [0.982, 0.993]	0.011 [0.010, 0.012]
	Test-based	0.849 [0.847, 0.850]	0.953 [0.947, 0.959]	-0.073 [-0.089, -0.059]	0.994 [0.987, 1.000]	0.004 [0.003, 0.005]
L1	No updating	0.846 [0.845, 0.847]	0.815 [0.810, 0.821]	-0.221 [-0.237, -0.207]	1.014 [1.008, 1.021]	0.038 [0.036, 0.041]
	Refitting	0.846 [0.845, 0.848]	0.936 [0.930, 0.942]	0.005 [-0.011, 0.020]	1.038 [1.032, 1.044]	0.010 [0.009, 0.012]
	Test-based	0.846 [0.845, 0.847]	0.937 [0.932, 0.942]	-0.081 [-0.097, -0.066]	0.999 [0.993, 1.005]	0.005 [0.005, 0.006]
RF	No updating	0.837 [0.835, 0.838]	0.842 [0.837, 0.848]	-0.031 [-0.059, -0.006]	1.080 [1.068, 1.091]	0.033 [0.031, 0.035]
	Refitting	0.837 [0.836, 0.838]	0.950 [0.943, 0.956]	0.127 [0.096, 0.153]	1.082 [1.070, 1.094]	0.026 [0.024, 0.028]
	Test-based	0.837 [0.836, 0.838]	0.939 [0.933, 0.945]	-0.035 [-0.061, -0.010]	1.017 [1.006, 1.028]	0.019 [0.017, 0.021]
NN	No updating	0.770 [0.768, 0.772]	0.914 [0.908, 0.920]	-0.187 [-0.205, -0.171]	0.965 [0.959, 0.971]	0.018 [0.017, 0.020]
	Refitting	0.800 [0.798, 0.802]	0.991 [0.984, 0.997]	-0.104 [-0.122, -0.087]	0.961 [0.955, 0.967]	0.004 [0.003, 0.004]
	Test-based	0.800 [0.798, 0.802]	0.991 [0.984, 0.997]	-0.104 [-0.122, -0.087]	0.961 [0.955, 0.967]	0.004 [0.003, 0.004]

Figure 50. Overall calibration of acute kidney injury models by learning algorithm and updating strategy. Left panels display calibration curves across the range of predictions produced by each model; right panels zoom in on calibration curves for predicted probabilities below 25%, which includes over 95% of all observations.

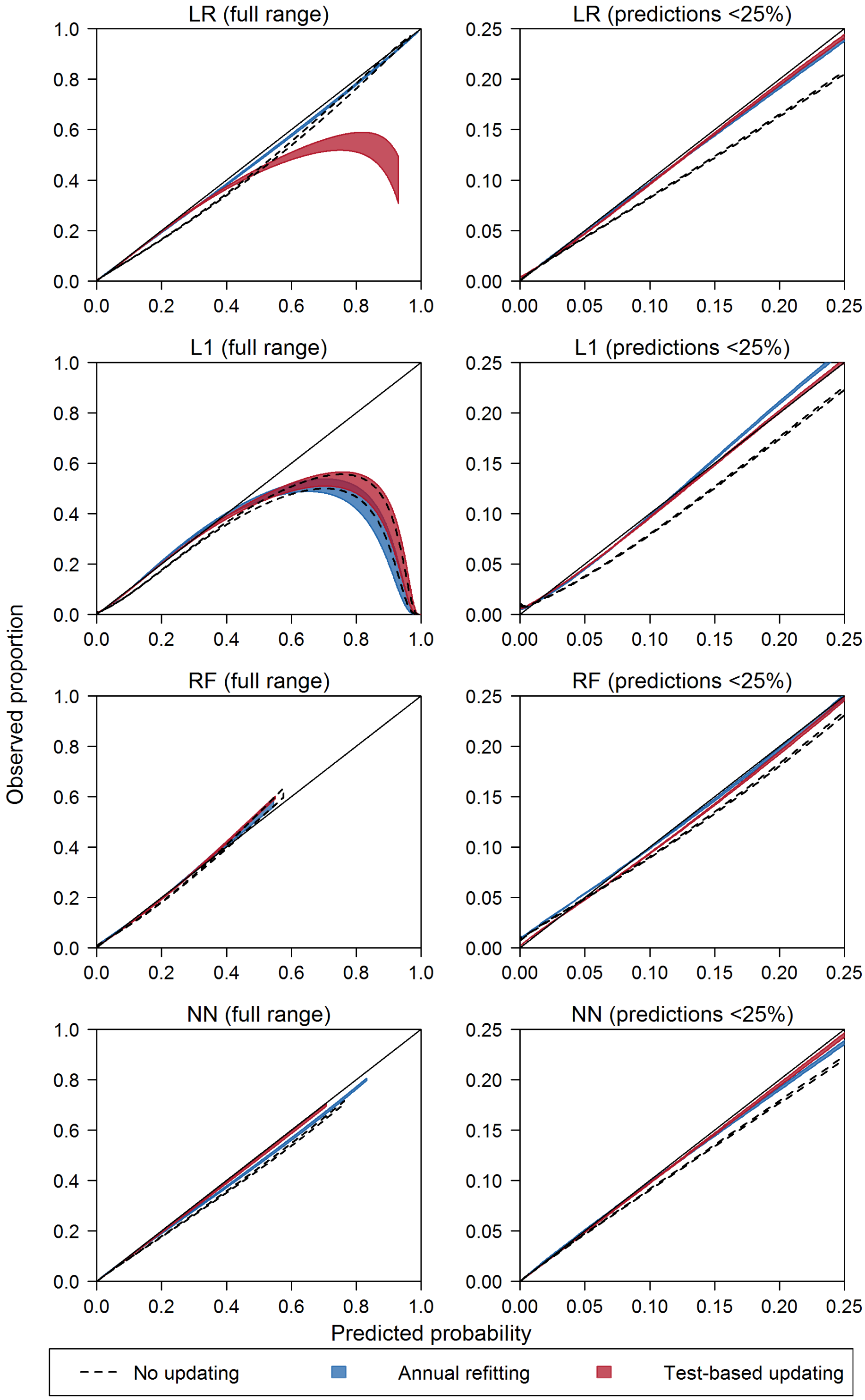
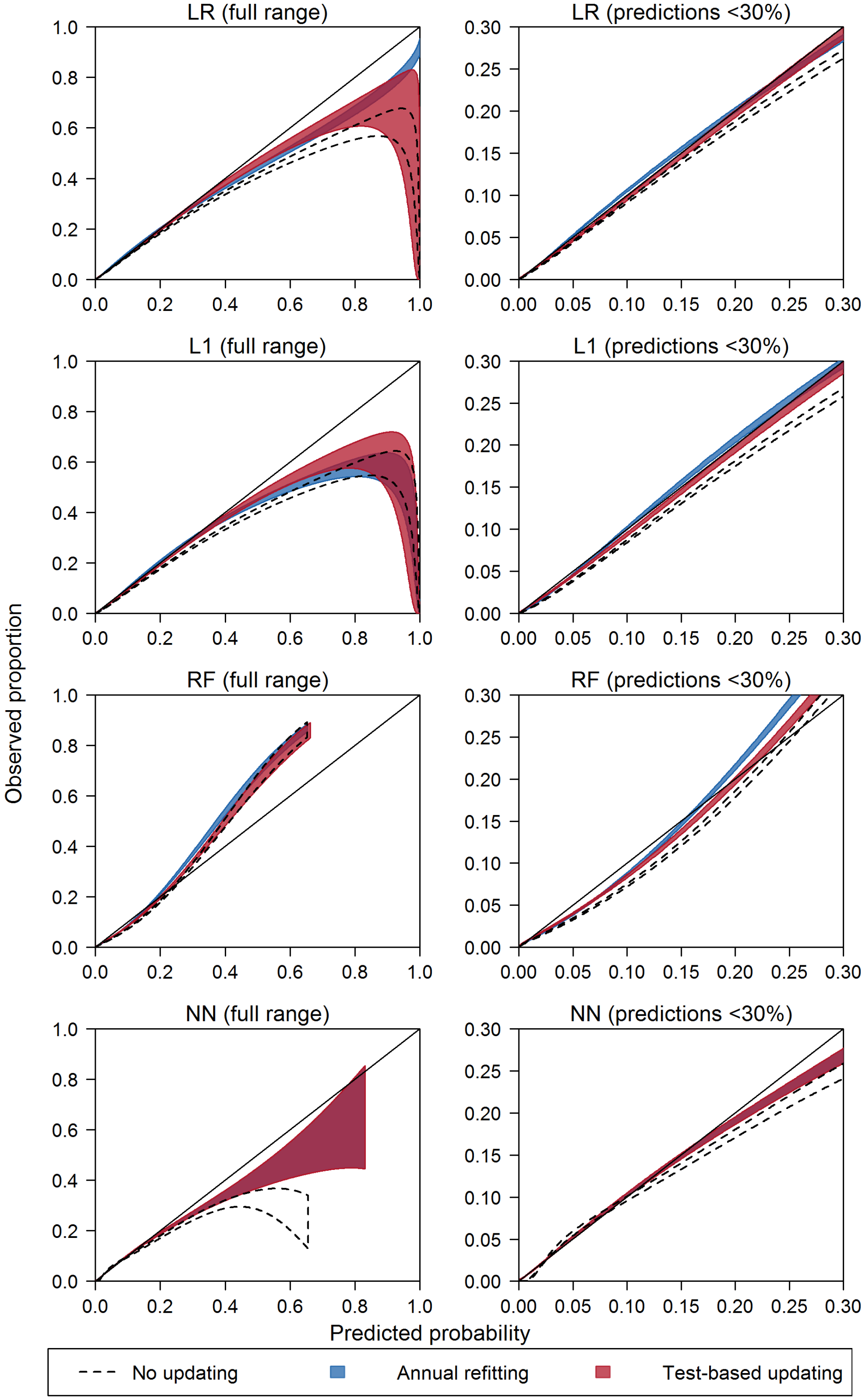


Figure 51. Overall calibration of 30-day mortality models by learning algorithm and updating strategy. Left panels display calibration curves across the range of predictions produced by each model; right panels zoom in on calibration curves for predicted probabilities below 30%, which includes over 95% of all observations.



Figures 52 and 53 display monthly calibration by learning algorithm and updating strategy using the estimated calibration index (ECI). This stringent measure of calibration decreases toward 0 as calibration improves.[39, 90] Without updating, ECIs increased over time, with the magnitude of the overall increase and monthly fluctuations varying by both learning algorithm. Both refitting and test-based updates improved calibration compared to the original model in the years following initial model development. For both outcomes, the RF model was an exception to this pattern. In some months, calibration under the refitting strategy was similar or inferior to that of the original RF model. This was particularly true for the mortality models (see Figure 53). The mortality RF model trained on 2008 admissions performed poorly relative to the other updating strategies when applied to 2009 admissions; the AKI mortality model trained on data from 2007 performed poorly on 2008 admissions. Although calibration of the original mortality NN model was stable over time compared to the other mortality models, annually refitting the mortality NN model improved calibration and reduced month-to-month variability in performance. For the mortality models, monthly ECIs under the test-based updating

Figure 52. Estimated calibration index (0 under perfect calibration) of acute kidney injury models over time by learning algorithm and updating strategy. Dotted vertical lines highlight points at which the testing procedure recommended recalibration.

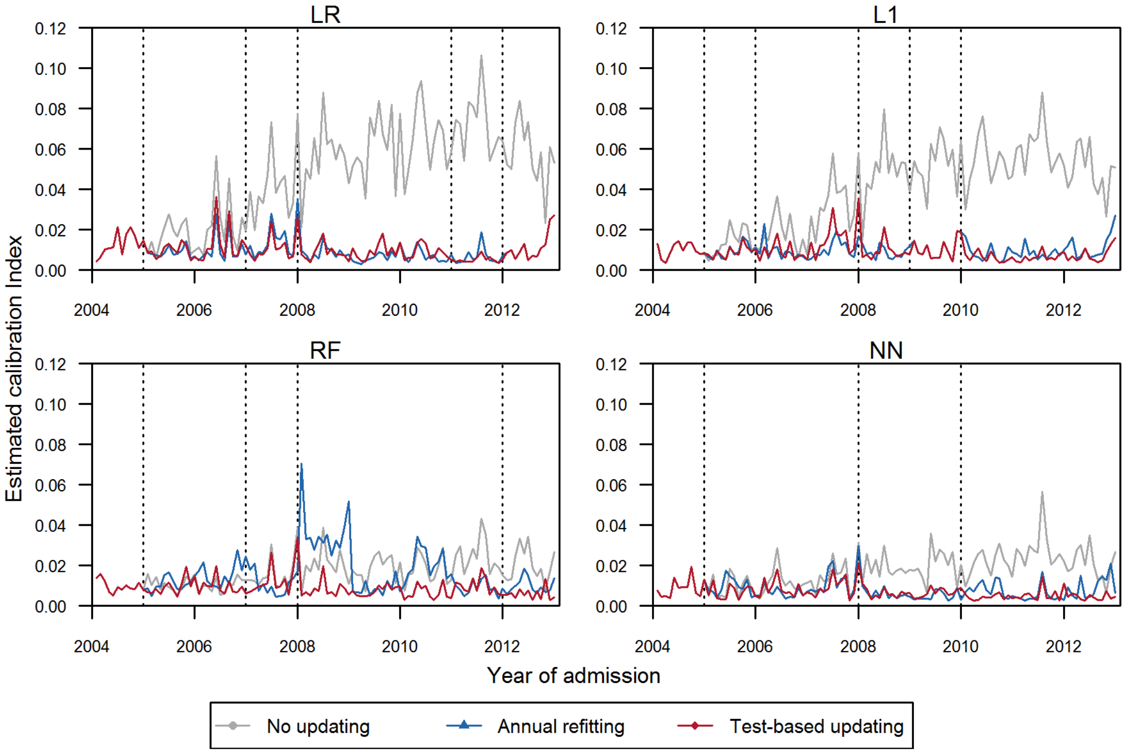
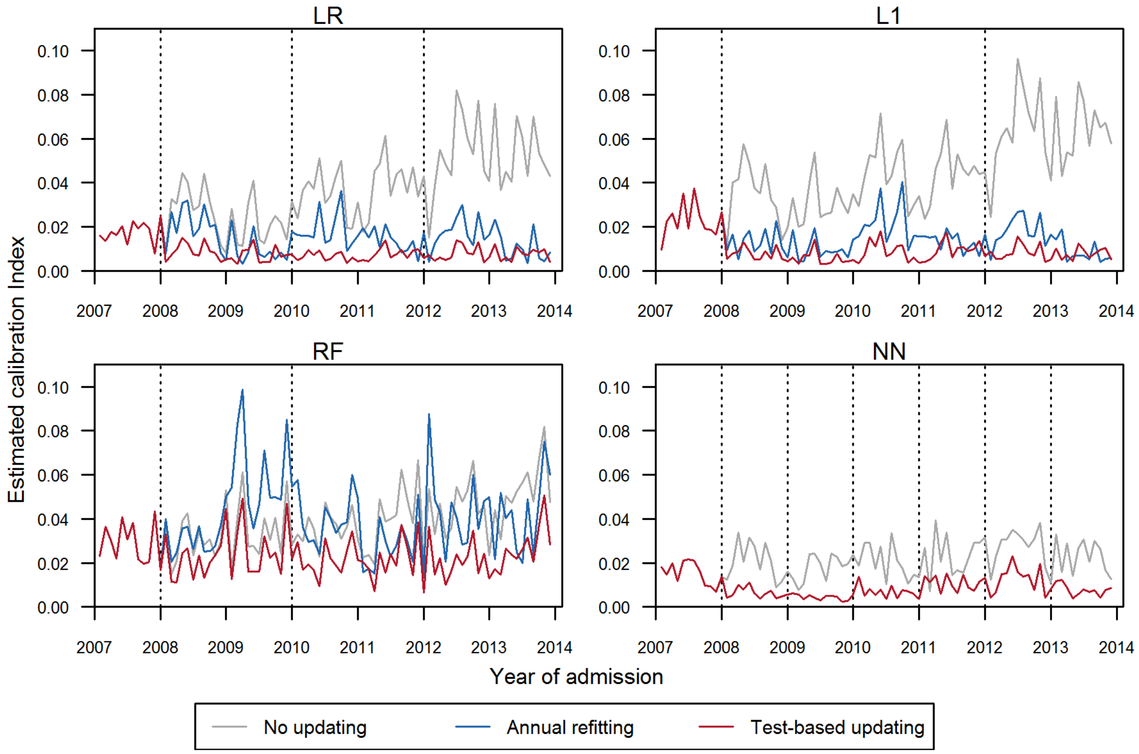


Figure 53. Estimated calibration index (0 under perfect calibration) of 30-day mortality models over time by learning algorithm and updating strategy. Dotted vertical lines highlight points at which the testing procedure recommended recalibration.



strategy were generally lower and less variable compared to ECIs under the refitting strategy. For the AKI models, monthly ECIs under the test-based updating strategy were similar to those observed for the refitting strategy, with the exception of the RF model. For those points at which the testing procedure recommended updating the AKI L1 model, ECIs over the prior 12 months (i.e., performance among those admissions serving as the update set guiding the test recommendations) appeared to increase in either magnitude (e.g., 2005 and 2007) or variability (e.g., 2009). A similar pattern was generally not apparent prior to those points at which the testing procedure recommended updating other models. Nevertheless, calibration improved immediately after these updates.

Discussion

We evaluated the impact of three competing updating strategies on performance of models for hospital-acquired AKI and 30-day mortality after hospital admission over several years following initial model development. In addition to common strategies of retaining the original model or routinely refitting, we included a new data-driven strategy based on our nonparametric testing procedure for selecting among competing updating methods. This testing procedure is applicable regardless of the learning algorithm underlying the model, allowing our study to compare updating requirements of parallel LR, L1, RF, and NN models.

Updating requirements varied across learning algorithms, both in terms of the timing and extent of updates. One year after model development, the nonparametric testing procedure recommended updating of all four models for both outcomes. These initial adjustments lead to immediate improvements in calibration in the months following the update. Subsequent updating recommendations were varied, with each updating method being recommended at least once and different methods being recommended for different learning algorithms even at the same timepoint. Interestingly, the most significant and frequent updating was recommended for the mortality NN model, which exhibited the least performance drift over time. The testing procedure recommended refitting each year due to quite small improvements in the Brier score (~0.0001) compared to other updating approaches. As the Brier score takes into consideration both discrimination and calibration, the improvement in both dimensions of performance that resulted from refitting the NN model may have driven this recommendation. Refitting of the other mortality models impacted calibration, but did not significantly improve discrimination.

Some form of updating was warranted for all models. Retaining the original model over the course of the study period resulted inferior calibration compared to either routine refitting or test-based updating. Calibration measures of the original mortality NN model did not exhibit significant trends indicative of performance drift over the course of the study.[18] Nevertheless, refitting this model each year, either as planned or as recommended by the testing procedure, still improved overall calibration, reduced month-to-month variability in calibration, and improved discrimination. For the other models, test-based updating improved upon the simple refitting strategy. Refitting corrected performance drift in the mortality LR and L1 models. Test-based updating recommendations, however, resulted in lower overall ECIs (i.e., better calibration) and less month-to-month variability in performance compared to refitting. This was generally observed for the AKI LR and L1 models as well. However, in some cases, calibration of the AKI L1 was less stable on a month-to-month basis under the test-based strategy compared to the refitting approach (e.g., 2007 admissions). Refitting the RF models improved overall calibration compared to the original model for both outcomes, but resulted in inferior calibration than the original model over shorter periods and did not correct performance drift in the mortality case. On the other hand, the test-based strategy avoided performance drift of the mortality RF model and exhibited fewer periods of instability observed under the refitting strategy for both outcomes.

In some cases, differences in calibration metrics between updating strategies were small and may not be clinically meaningful in practice. Whether these improvements are clinically meaningful, in addition to being statistically significant, is an important consideration and an open question for model comparison and impact assessment work. Although small in magnitude, the improvements in calibration under the test-based strategy compared to the refitting strategy highlight how recalibration may be sufficient, or even superior, to the standard practice of undertaking more substantial change by refitting. In addition, impact from recalibration is most likely to occur when patients are scored near user-defined cut-points that are clinically relevant, and assessment of clinically meaningful risk category reclassification anchors around what proportion of patients are near the cut-points (and change classification after calibration degradation). Future work could explore the impact of differing updating strategies on reclassification metrics.

These findings underscore this dissertation's central theme – the need for data-driven maintenance plans for clinical prediction models. A "one-size fits all" updating strategy will not suffice for all models. We cannot assume a new model built on recent data will be more generalizable to and perform better in the next cohort of patients than an existing model, even when large datasets, such as those in this study, are used for updating. For example, although calibration improved over the entire study period by regularly refitting the mortality RF model, the model built on 2008 admissions did not improve upon, and may have actually performed worse, than the original mortality RF model when applied in 2009. Similarly, we should not assume refitting is superior to simpler updating through recalibration. The intermittent recalibrations recommended by our testing procedure lead to better performance across the study period than routine refitting, both overall and on a month-to-month basis. Tailoring updating methods through data-driven updating strategies may, therefore, extend the accuracy and subsequent utility of prediction models beyond what might be achieved through simpler maintenance plans. We note, however, that these results may be sensitive to the volume of data available for updating, and further investigation regarding the impact of sample size is warranted.

Our results also highlight differences in the frequency with which models require updating. Despite being applied to the same data and therefore exposed to the same shifts in patient populations and clinical environments, the LR, L1, RF, and NN models required updating at different timepoints. With the exception of the mortality NN model, updating on an annual basis was not indicated and annual refits did not provide additional benefits over less frequent updates. Thus, we may experience inefficiencies under model maintenance plans requiring updates on a pre-planned regular basis. On the other hand, prescheduled updating plans may neglect to update models in a timely manner, allowing periods of performance drift to go unnoticed and uncorrected. The cost of interim periods of reduced model accuracy may be difficult to assess as the prediction errors may impact patient outcomes, user confidence, and clinical efficiency. As health systems seek to implement clinical prediction more broadly and begin managing many prediction models, additional data-driven methods to determine when models require attention may be necessary and would complement maintenance strategies implementing test-based updating methods. We addressed this methodological gap with the calibration drift detection system described in Chapter 5.

There are several limitations of the analyses presented here. We evaluated the three updating strategies in two clinical use case leveraging VA data. Exploring how these updating strategies perform on models subject to additional patterns of shift in the clinical environment would provide more generalizable understanding. In this study, we limited the nonparametric testing procedure to consider five updating methods – retention of the existing model, intercept correction, linear logistic recalibration, flexible logistic recalibration, and model refitting. These updating methods are common and applicable across models; however, additional updating methods, some of which may be specific to certain learning algorithms, could easily be incorporated into the testing procedure.[35] The availability of additional updating methods may impact when and how the test-based strategy adjusts models over time. Further, we did not explore the impact of sample size. Both the AKI and mortality datasets included on average over 180,000 and 235,000 admissions per year, respectively. The volume of data available for constructing updates could have important impacts on both the refitting and test-based updating strategies. For small samples, overfitting becomes more of a concern for the refitting strategy, while overly conservative updates may be a concern for the test-based strategy. We also acknowledge that the test-based updating strategy may be computationally intensive. Leveraging advances in computational resources and refining the number of bootstrap iterations considered in the first bootstrapping stage may reduce any computational burden. Tailoring the number of bootstrap iterations may also allow users to match statistical significance to clinically relevant magnitudes of change. Finally, all three of the updating strategies considered here may be inappropriate in the presence of significant changes in clinical practice or record systems that may render existing prediction models invalid. Any updating strategy must be flexible, both in terms of timing and approach, in response to such situations.

Conclusion

We illustrated the use of a new data-driven updating strategy for clinical prediction models based on a variety of underlying learning algorithms, comparing this strategy to two baseline approaches in which models are either never updated or regularly refit on recent observations. The test-based updating strategy conservatively adjusted models by recommending intermittent recalibration rather than repeated model refitting in most cases. Despite making limited adjustments to the models, the test-based updating strategy lead to more highly calibrated predictions than either of the baseline strategies. The test-based approach also highlighted differences in the updating requirements of common biostatistical and machine learning models, both in terms of the extent and timing of updates. Data-driven updating strategies, such as the test-based approach presented here, can both support implementations of new models transported across clinical settings and serve as a key component of automated model surveillance systems, such as that described in Chapter 3.

CHAPTER 8

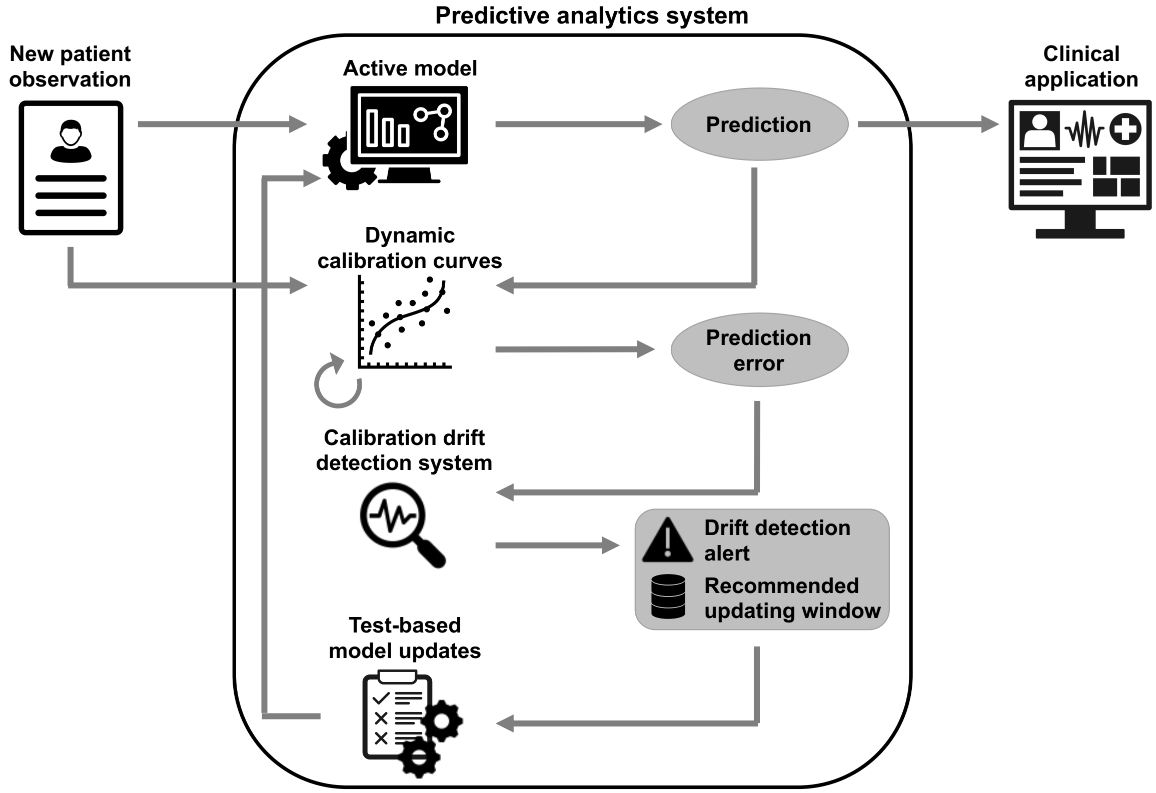
CONTRIBUTIONS AND CONCLUSIONS

Highly accurate predictions are critical to the success and safety of population health management, quality assessment, and clinical decision support tools employing prediction models.[4, 41, 64] Erroneous patient-level risk estimates produced by miscalibrated models may lead to over-confidence, inappropriately alter treatment choices, or misappropriate resources.[4, 23, 40, 42] As electronic health record-enabled risk prediction models are increasingly employed in healthcare applications, there is growing awareness of the need to address the tendency of model performance to deteriorate over time.[18-20, 23, 33, 43, 48-52] Common predefined updating strategies fail to account for variations in the response of models to changes in clinical environments which may impact the timing, extent, and form of drift in accuracy.[18-20, 29] Our work provides additional evidence that simple “one-size fits all” updating strategies do not effectively maintain consistent model performance and responds to these concerns by developing a framework and set of algorithms to maintain performance of risk models over time.

Innovation

As an alternative to prescriptive updating strategies, we proposed an active, data-driven model surveillance and updating system that may be embedded within electronic health record systems to promote the long-term reliability and utility of clinical prediction tools (see Figure 54). This system would accumulate evidence from the stream of data on new clinical encounters, allowing the system to identify and respond to performance drift as it occurs. We developed a suite of methods forming the necessary components of such a data-driven updating approach, ensuring the methods were applicable to categorical models based on both regression and machine learning techniques. Key features of these methods are noted in Table 19. We first developed the notation of dynamic calibration curves to maintain an evolving understanding of recent model performance. Leveraging these dynamic calibration curves, we built a calibration drift detection system to trigger model updating as performance declines and inform the updating process with insight into selecting updating datasets. Finally, we defined a nonparametric testing procedure to evaluate available updating methods and recommend simple updating method that improves subsequent model performance. Each method was

Figure 54. Conceptual model of a data-driven, active model surveillance and maintenance system.



designed to be widely applicable to categorical outcome models regardless of the underlying learning algorithm and customizable to meet the needs of diverse clinical use cases.

Dynamic Calibration Curves to Assess On-Going Performance

Calibration curves, based on regression of predicted probabilities on observed outcomes, provide insight into model performance across the range of prediction.[88, 90] Not only do these curves support visualization of the varying alignment between predicted and observed probabilities in different ranges of risk, they also support calculation of detailed calibration metrics. [37, 39, 89, 90] Decision analyses reveal that calibration metrics based on nonlinear calibration curves ensure predictions are nonharmful to clinical decision-making compared to treat-all or treat-none strategies.[39] Such calibration metrics are particularly susceptible to drift over time in response to the non-stationary nature of clinical environments.[18, 19]

Table 19. Overview of new data-driven methods for updating clinical prediction models.

Method	Features			Limitations
	Practicality	Generalizability	Customizability	
Dynamic calibration curves	• Simple implementation • Computationally efficient • Processes data as a stream • Supports both visualization and stringent calibration metrics	• Applicable regardless of underlying learning algorithm generating predicted probabilities	• Permits user-specified parameterization logistic curve	• Distribution of predicted probabilities may impact ability to visualize true form of calibration • Requires careful definition of curve parameterization and algorithm step size
Calibration drift detection system	• Simple implementation • Computationally efficient • Balances false alarms and delays in detection • Informs both timing of updates and definition of updating set	• Applicable regardless of underlying learning algorithm generating predicted probabilities • Detects performance change under multiple speeds of temporal transition	• Supports user-specified error metrics with bounded range	• Alerts in response to statistically significant changes in calibration not clinically relevant changes • Difficulty recognizing recurrent/seasonal performance change • Does not evaluate whether recommended updating set may be sufficient for effective updating
Updating Testing Procedure	• Improves prospective model performance • Promotes simple updates when feasible • Makes conservative recommendations when updating samples are limited	• Applicable regardless of underlying learning algorithm generating predicted probabilities • Avoids establishing a “gold standard” updating method assumed best in all cases	• Allows optimization of user-preferred scoring rule • Extends to consider additional updating methods of interest	• Multi-stage process adds complexity to implementation • May be computationally expensive when bootstrapping refitting of complex models • Does not guarantee recommended update achieves clinically acceptable performance

Surveillance of detailed model performance measures over time requires up-to-date calibration curves reflecting recent model performance. In order for on-going assessment with calibration curves to be feasible, we identified the need for an alternative to static curves which require assumptions about the speed of performance drift and repetitive curve construction. The dynamic calibration curve approach, described in Chapter 4, responds to this need by providing a computationally efficient method to progressively evolve logistic regression-based calibration curves to reflect recent model performance. Our method implements incremental gradient descent with an adaptive learning rate to immediately incorporate information on each new observation's predictive accuracy as data becomes available. The process by which predicted probabilities are generated is transparent to the dynamic calibration curve implementation, making this method generalizable to categorical prediction models regardless of the underlying learning algorithm. Dynamic calibration curves are easily customizable in terms of the form and degree of flexibility in the nonlinear relationship between predictions and outcomes. While we parameterized our curves with fractional polynomials, the incremental gradient descent approach easily supports alternative parameterizations, including splines, traditional polynomials, and other fractional polynomial combinations.

In our simulation studies, we found dynamic calibration curves responded quickly to changes in model performance, shifting curves to represent current rather than past model performance. Following an abrupt change from calibrated to overpredicted or overfit predictions, dynamic calibration curves shifted to capture the new form of calibration within approximately 600 and 150 observations, respectively. This swift evolution of the curves toward the new true form of calibration was observed after both brief and extended periods of initial performance stability. Dynamic calibration curves were best able to represent the changing relationship between predictions and outcomes in data-dense ranges of predicted probabilities. This may be sufficient when using calibration curves to calculate detailed metrics of predictive accuracy. However, this observation underscores the importance of incorporating data density and the range of predictions into visualizations based on dynamic calibration curves. We also note that performance drift toward more complex forms of miscalibration highlighted a need to refine the step size hyperparameter provided to the adaptive learning rate algorithm. Nevertheless, our method for dynamic calibration curves offers an on-going understanding of up-to-date model performance to support continuous model assessment within model surveillance tools.

A Calibration Drift Detection System to Trigger Updating

Common model maintenance protocols lay out a predefined schedule for updating.[27, 31, 32] This approach ensures all active models receive attention and are updated regularly. In practice, however, the frequency of scheduled updates may not align with the timing and speed of performance drift. Temporal misalignment between performance drift and updating schedules may leads to unanticipated intervals of inadequate model performance during periods of rapid population shifts, as well as inefficient prioritization of analytic resources during periods of relative model stability. We observed the latter in the case studies presented in Chapter 7. Our analyses indicated different models required updating at different frequencies and annually refitting each model provided no additional benefit over the testing procedure-recommended, less frequent updates.

We proposed triggered model updating as a data-driven alternative to scheduled updating protocols. Updating models on a timeline driven by the accumulation of evidence of performance drift in recent data may allow more timely correction of performance drift and, in turn, more stable performance characteristics. Monitoring calibration with continuous assessments, such as our dynamic calibration curves, is critical to understanding how accuracy may be deteriorating over time. However, data-driven model surveillance also requires a process to distinguish performance drift from natural performance variability. We constructed a calibration drift detection system to provide data-driven guidance on when clinical prediction models may require updating. Our system utilizes an adaptive windowing approach[34] to warn users if recent observations provide sufficient evidence of change in the distribution of a model's predictive error. This method not only alerts users to performance drift, but also specifies a candidate sample of recent data for developing updates in response. While our implementation monitors a detailed error metric by leveraging on-going insight into model performance from dynamic calibration curves, our calibration drift detection system can be tailored to monitor distributions of alternative error measures. As the adaptive windowing monitor only requires as input the value of each new observation's error, the system can be used to detect drift regardless of a model's underlying learning algorithm. Our drift detection approach can be used to initiate predefined model updating strategies or in conjunction with data-driven methods selecting updating methods.

We evaluated our calibration drift detection system's ability to identify change in performance and recommend updating samples under multiple magnitudes, complexities, and speeds of performance drift. The system generally avoided false alarms, minimizing both the risk of alert fatigue in the presence of stable model performance and missed opportunities for model improvement in the presence of deteriorating model performance. Seasonal performance drift was most difficult for the system to identify, with frequent false negatives in the case of small, recurrent/seasonal performance changes. However, we note that seasonal patterns can often be accounted for during model development and we anticipate longer term performance trends to be the dominant drivers of performance drift. Our system alerted quickly, in most cases within a few hundred observations, to statistically significant changes in the distribution of prediction errors. After the onset of performance drift, alerts were returned with samples of recent data representing the model's new calibration context. Contamination of the returned sample with outdated observations occurring prior to the start of performance drift was less than 3-5%. This is reassuring that our system provides insight into a window of recent data that is informative of the new environment and, therefore, useful for updating the model to better reflect current patterns of association and risk. However, a balance must be struck between the speed at which performance change is identified and the accumulation of relevant post-drift observations to support the updating process. Whether recommended updating samples are large enough to fully support model updating will be highly dependent on the complexity of the model and the degree of updating necessary to return the model to acceptable performance.

A Testing Procedure to Recommend Updating Methods

Whenever users decide to update a particular clinical prediction model – either in response to an alert from our calibration drift detection system, a scheduled updating plan, or the transportation of a model between clinical settings – refitting the model on new data should not be presumed to be the best approach. Model refitting neglects information from previous modeling efforts and often reduces generalizability as a result of overfitting on relatively small updating samples.[10, 11, 22] Recalibration techniques, on the other hand, build upon information already incorporated into existing models and improve generalizability, making these approaches preferable when recalibration is sufficient to improve performance.[11, 24, 26, 30] Despite recommendations emphasizing a consideration of recalibration prior to refitting,[11, 24, 26, 30] current updating protocols often simply call for model refitting.[27, 31, 32]

In order to provide guidance in selecting between competing updating methods, we developed a nonparametric testing procedure to provide data-driven recommendations. Using a two-stage bootstrapping approach, our testing procedure minimizes the influence of overfitting and accounts for uncertainty associated with updating sample sizes. The decision stage of the testing procedure incorporates a preference for simpler updates when more sophisticated techniques do not afford significant additional performance improvement. Our testing procedure is widely applicable to clinical prediction models developed with both parametric and nonparametric techniques; although computational demands of the first bootstrapping stage may be high for the most complex models. Users can easily customize the testing procedure to use case requirements by adjusting the performance metric on which recommendations are optimized and by incorporating additional updating methods of interest.

In a combination of simulation and case studies, we found our nonparametric testing procedure responded to both the degree of performance drift and the volume of updating data, resulting in improved prospective model performance. Recommended updating methods increased in complexity as training and updating populations became increasingly disparate. The testing procedure recommended simple updating methods when applied to small updating samples, and graduated to recommending full model refitting as the volume of updating data increased. For example, in simulations involving no differences between training and updating populations, our testing procedure recommended retention of the original model when updating samples were smaller or similar in size to training samples, but recommended refitting when updating samples grew to be 10 times larger training samples. In case studies of models for 30-day mortality and acute kidney injury, test recommendations also reflected differences in performance drift across learning algorithms. Despite being applied to the same data, models developed with different learning algorithms varied in terms of the frequency, timing, and extent of recommended updating. Applying the test-recommended updating methods resulted in immediate improvements in calibration in the months following update. Compared to both the original models and annually refitted models, updating with our testing procedure lead to better and more stable calibration over multiple years.

A Suite of Data-Driven Updating Methods for Model Surveillance

The methods we developed in this dissertation can be used to tailor the model updating process to the unique performance drift patterns and accuracy requirements of specific clinical prediction tools. Each of the three new methods have independent use cases. Dashboards monitoring active clinical prediction models may rely on dynamic calibration curves to provide an understanding of current model performance and how that performance has changed over time. Our calibration drift detection system may be implemented to alert model managers to those prediction models experiencing significant changes in performance and encourage timely intervention, even if local policy dictates updating methods. Without being triggered by a drift detection approach, our testing procedure may be employed to select between available updating methods when initially transporting a model across clinical setting or on a scheduled updating timeline.

In addition to their independent utility, these new data-driven methods promise to be most powerful when used in concert as illustrated in our conceptual model (see Figure 54). We anticipate a fully data-driven strategy, responding to performance drift as it occurs and using updating methods supported by information in recent observations, will outperform the current "one size fits all" state of the art approach. Future work will evaluate whether integrating our methods into such a data-driven updating strategy improves the stability and reliability of model performance over time. This will lay the ground work for automated, EHR-embedded model surveillance systems promoting the long-term performance and utility of prediction models underly a variety of informatics applications for decision support and population management.

Limitations

The work presented in this dissertation is limited in several dimensions that require further investigation in order to promote model reliability more broadly in practice.

Statistically Significant Drift May Not Align with Clinically Significant Drift

The data-driven methods we developed evaluate calibration and model updating from a statistical perspective rather than that of clinical utility. However, statistically significant changes in calibration may not translate directly to clinically important changes in model performance. For example, in the analyses presented in Chapters 7, while test-based updating statistically significantly improved performance over a refitting strategy, in some cases the differences in calibration metrics between updating strategies were small and may not be clinically meaningful in practice. Although stringent calibration may ensure predictions are nonharmful to clinical decision-making,[39] the magnitude of acceptable miscalibration and performance variability likely varies by use case. Understanding whether, when, and how performance drift affects the clinical utility of predictions for decision-making is key to establishing the value of model surveillance and updating strategies.

While defining and measuring clinically acceptable performance remains an open area of research,[111-114] the methods we developed in this dissertation are well-positioned to incorporate new insights from this domain as they develop. For example, performance drift is most likely to impact clinical utility when the accuracy of predictions in clinically-relevant decision risk regions and near classification cut-points deteriorates. As methods for defining clinically-relevant decision boundaries develop, one could imagine tailoring our calibration drift detection system to place more import on performance changes in these regions. Given clinically-relevant decision thresholds and classification cut-points, our nonparametric testing procedure could be implemented with a weighted scoring rule to emphasize accuracy in critical regions or with reclassification metrics to emphasize differences in risk categorization between updating methods. Future work could explore how updating plans and stability of model performance are influenced by incorporating clinical significance into our data-driven model surveillance and updating methods.

Data-Driven Methods May Not Correct All Performance Drift

Updating clinical prediction models with the guidance of data-driven methods, such as those presented here, may not sufficiently improve performance to fully correct for calibration drift or return model performance to clinically acceptable levels. This may be particularly true if care processes have been modified, variable definitions or measurement accuracy have changed, or scientific insights have generated new influential predictors.[24] Users should evaluate performance after model updating to determine if clinically acceptable performance is restored or whether models require either further modification or their use discontinued. Local knowledge of data managers and clinical users may provide critical perspective on these decisions by identifying unanticipated changes in variable definitions or evolving clinical understanding. Their insight may direct further updating of clinical prediction models with partial association adjustment, which combines both recalibration and estimation of select additional coefficients, or model extension.[24, 26] Thus, although data-driven model surveillance and updating systems can support predictive modeling teams in prioritizing their workload and resources, the human element remains critical to realizing and sustaining the benefits of clinical prediction tools.

Alternative Modeling Contexts Warrant Consideration

We have focused on updating methods for restoring the performance of static dichotomous categorical prediction models; however, other modeling frameworks will require similar maintenance guidance and corresponding methods development. Our methods for continuous model assessment and calibration drift detection are easily extensible to models for multinomial outcomes, as calibration curves and several metrics have established multiclass definitions.[33, 70, 103] The spectrum of updating methods – from recalibration to refitting to extension – is applicable to multiclass models, extending the relevance of our testing procedure to this modeling context. Assessing calibration and temporal performance of survival models for time-to-event outcomes, on the other hand, is more challenging[115] and warrants focused research effort. Similarly, as deep learning models become increasingly common and move toward implementation in clinical settings, additional work will be necessary to better understand the updating requirements and challenges of such models.

Online learning algorithms, which continuously update models as new observations become available,[28, 67, 68] should be considered as an alternative to periodic updating of static models with either predefined or data-driven strategies. Although online models have been applied to health outcomes, the shift to an online paradigm is not straightforward for clinical use cases. As noted in Chapter 2, we focused our work on the established static prediction model paradigm as implementing continuously updated online models will require new validation methods[28, 67] and are subject to an evolving regulatory framework.[69] We note, however, that the data-driven model surveillance framework we proposed can support clinical systems that implement both static and online prediction models. For example, our calibration drift detection system could be used to monitor the performance of online models to provide reassurance that the continuous updating process is successfully maintaining model performance and to highlight any deterioration that may indicate a breakdown in the flow of data to the model.

Feedback Loops Will Require Additional Methods

Current updating approaches and the new methods we presented through the course of this dissertation presume the data we observe for model evaluation and updating is without undue bias. However, if we find success achieving the goals of interventions based on clinical prediction tools, we may actually introduce bias that will require renewed consideration of model development and updating methods. This stems from the feedback loop that would be created by changes in provider or patient choices that arise from the use of predictions in the decision-making process.[116] If the treatment course of a patient is altered by the risk prediction in a clinical decision support tool and that choice impacts the patient's eventual outcome (ideally for the better), then the observed outcome will be biased by the availability of the prediction.

Without accounting for this feedback, the biased data available for monitoring and correcting model performance could lead us to falsely determine predictor-outcome associations have shifted and the model requires significant adjustment.[116] We thus need new methods to ensure models remain up-to-date while avoiding updating away the useful information that created the successful intervention. Such feedback loops pose new and interesting challenges that require more guidance on model development methods that incorporate interventions and motivate additional methods development for model evaluation and updating.[116] We foresee flexibility in our concept of an active, data-driven model surveillance system that will support the incorporation of new evaluation and updating methods as this area of research develops.

Clinical Implications

With increasingly widespread integration of advanced predictive analytics into electronic health records and healthcare applications,[4, 12, 14, 15] the challenges of maintaining clinical prediction tools over time will require increasing attention from healthcare administrators and health information managers. Reliable, accurate clinical prediction models can support complex decision-making, inform targeted interventions, and promote safety. Insufficient calibration of prediction models, on the other hand, can lead to misleading information with implications for sub-optimal care, misappropriate of resources, and risks to patient safety.[4, 23, 40, 42] For example, patients presented with inflated estimates of negative disease prognosis may choose to undergo difficult treatments that may not align with their values and may not have been their choice given more accurate risk estimates.[42] Similarly, quality assurance systems may misidentify underperforming units when risk predictions inadequately correct for patient risk profiles.[20] Unfortunately, we cannot rely on the initial performance characteristics of a newly developed clinical prediction model to be sustained over time without intervention. Clinical environments are everchanging, evolving in terms of patient populations, clinical practice, workflows, information systems, and scientific understanding.[10, 11, 24, 27, 28] As a result, the accuracy of prediction models deteriorates over time[18-20, 23, 43, 48-52] and effective strategies for model maintenance are becoming critical components of predictive analytics implementations.[4, 10, 11, 21, 117]

Our work responds to a need for informed model updating strategies to sustain the accuracy and utility of applications relying on clinical predictions. Model updating can restore model performance with significant consequences for clinical uses. For example, one study found recalibration to update an outdated clinical prediction model revealed quality assessments of intensive care units were overly optimistic using the outdated model, highlighting higher mortality rates than expected in 35% rather than 15% of units.[20] Our data-driven methods for performance monitoring and model updating are broadly applicable and customizable to the wide variety of clinical prediction models implemented in population health management, quality assessment, and clinical decision support tools. Our evaluations, based on both simulated data and large cohorts of national inpatient admissions data, showed our data-driven methods were able to identify changes in model performance as it occurred and to recommend updating methods that lead to more stable and more accurate performance than scheduled, non-data-driven updating strategies. We envisioned a data-driven active model surveillance system that integrates these methods within production clinical information systems to deliver more consistently accurate and reliable predictions. By promoting stable, accurate model performance, this work reinforces safety, user confidence, and clinical utility of clinical prediction tools.

Informatics Implications

The methods developed here enable and encourage the translation of informatics advancements in predictive analytics into clinical decision tools. We presented a conceptual model for an active, data-driven model surveillance system that not only illustrates the use case integrating our new methods, but also provides a framework for predictive model management more broadly. Our data-driven methods were developed with special attention to generalizability and customizability, recognizing and supporting the variable needs of diverse clinical informatics applications.

Throughout this dissertation, our work embraces and furthers the movement toward more consistent attention to model calibration, both in general and specifically to detailed, stringent calibration measures. Although current recommendations emphasize the importance of calibration for clinical use cases employing predictions in decision-making,[4, 22, 23, 38-42, 118] calibration remains underreported in validation studies[65, 119-121] and stringent measures of calibration have yet to become commonplace.[37, 41] Our method for dynamic calibration curves and our use of a detailed calibration curve-based metric in our drift detection system advance efforts to promote detailed calibration measures. In addition, our dynamic calibration curves create new opportunities to monitor calibration in data streams rather than cross-sectional batches of observations. This allows us to begin thinking differently about how model validations can leverage the immediate availability of data within electronic health record systems.

This work recognizes and supports diverse approaches to the development of clinical prediction models. Each of our data-driven methods are applicable regardless of the learning algorithms used to train prediction models. We consider this feature to be critical to any guidance on model surveillance and updating strategies. Comparative studies indicate different learning algorithms achieve superior performance for different outcomes and clinical settings.[106, 122-128] Newer modeling methods, such as deep learning and online learning, continue to evolve and are increasingly applied to clinical outcomes.[27, 28, 63, 129, 130] Clinical predictive analytics systems must thus be flexibility designed to implement and manage models based on a diverse set of regression and machine learning methods. Our data-driven methods for model monitoring and updating, which rely solely observed outcomes and predicted probabilities, are designed with this essential generalizability in mind. This ensures our methods and vision for a data-driven model surveillance system are relevant for information systems managing a suite of prediction tools and are well-positioned to support the evolving landscape of clinical prediction.

Enterprise-wide clinical predictive analytics systems must ensure model accuracy is sustained over time and be agile in handling the rapid innovation of predictive analytic methods. The data-driven, generalizable, and customizable nature of the methods developed through the course of this dissertation empower clinical predictive analytics systems to adhere to these requirements. A data-driven model surveillance and updating system cannot correct all forms of performance drift or ensure clinically acceptable model performance. However, such a system can support predictive modeling teams in prioritizing their workloads and analytic resources. This, in turn, empowers broad implementation of electronic health record-embedded clinical prediction applications and the translation of advances in predictive analytic methods into practical clinical informatics tools.

Conclusions

Clinical prediction models have long provided insight to support clinical decision-making by synthesizing information across complex, interacting risk factors. Advances in prediction methods and the embedding of models within electronic health records are creating new opportunities to deliver personalized predictions in a variety of informatics applications – from point-of-care clinical decision support to population management to quality assessment. As interest grows in translating the potential of clinical prediction into practice, strategies to sustain performance over time are becoming critical components of model implementations. Common, predefined model updating strategies fail to account for variations in the timing, extent, and form of change in the accuracy of prediction models over time. We developed a suite of methods supporting data-driven model updating strategies. We first defined the notation of dynamic calibration curves to maintain an evolving assessment of model performance. Leveraging these dynamic calibration curves, we constructed a calibration drift detection system to trigger model updating as performance declines and inform the updating process with insight into defining updating datasets. Finally, we developed a nonparametric testing procedure to select between available updating methods, including recalibration and model refitting. Acknowledging the varied and developing scope of clinical predictive analytics, each method is designed to be both generalizable and customizable. This work lays the ground work for electronic health record-embedded, data-driven model surveillance systems that enable a shift away from insufficient “one-size fits all” updating methods and strategies. Individually and in concert, these methods tailor the model updating process to the unique requirements of specific prediction models and clinical use cases. This work promotes the long-term utility of prediction models underlying a variety of clinical informatics applications, and prepares data-driven model updating strategies to incorporate future methodological advancements in predictive analytics.

APPENDIX A

CALIBRATION DRIFT DETECTION SYSTEM SIMULATION RESULTS BY δ VALUES

Accuracy of detection reported as percent of iterations. False positives (FP) are detections occurring during the initial stable 1000 observations. False negatives (FN) are the failure to detect a changed by the end of the series. Note, recurrent/seasonal transitions do not have a stable run-in period and thus no false positives by definition.

Table 20. Frequency of false positive and false negative detections by transition speed, post-drift calibration setting and δ.

Post-drift calibration setting		**$\delta = 0.05$**		**$\delta = 0.075$**		**$\delta = 0.1$**	
	Transition pattern	**% FP**	**% FN**	**% FP**	**% FN**	**% FP**	**% FN**
Overpredicted (small)	Abrupt	0.2	0.8	0.6	0.4	0.6	0.5
	Rapid	0.1	0.6	0.5	0.3	0.5	0.5
	Gradual	0.3	2.5	0.2	1.8	0.8	1.5
	Recurrent/Seasonal	-	30.6	-	23.3	-	21.3
Overpredicted (large)	Abrupt	0.2	0	0.4	0	0.2	0
	Rapid	0.3	0	0.8	0	0.4	0
	Gradual	0.1	0	0.4	0	0.5	0
	Recurrent/Seasonal	-	7.7	-	6	-	5.6
Overfit (small)	Abrupt	0.3	0.6	0.4	0.3	0.6	0.4
	Rapid	0.4	1.1	0.8	0.4	0.7	0.7
	Gradual	0.2	1.7	0.3	1.4	0.5	1.8
	Recurrent/Seasonal	-	14.6	-	12	-	11.9
Overfit (large)	Abrupt	0.4	0	0.8	0	1.3	0
	Rapid	0.2	0	0.3	0	0.8	0
	Gradual	0.3	0	0.5	0.1	0.7	0
	Recurrent/Seasonal	-	3.1	-	3	-	2.1
Underfit	Abrupt	0.8	0	0.7	0	1	0
	Rapid	0.3	0	0.7	0	0.9	0
	Gradual	0.1	0	0.4	0	0.5	0
	Recurrent/Seasonal	-	0	-	0	-	0.1
Overpredicted & overfit (small)	Abrupt	0.2	16.2	0.4	13.2	1.4	12.2
	Rapid	0.3	14.4	0.5	10.5	0.6	10.4
	Gradual	0.4	21.6	0.2	17.9	1.1	17.7
	Recurrent/Seasonal	-	59.5	0	54.8	0	49.5

Table 20 continued.

Post-drift calibration setting	**Transition pattern**	**$\delta = 0.05$**		**$\delta = 0.075$**		**$\delta = 0.1$**	
		% FP	**% FN**	**% FP**	**% FN**	**% FP**	**% FN**
Overpredicted & overfit (large)	Abrupt	0.3	0.8	0.6	0.5	1.3	0.5
	Rapid	0.4	0.7	0.3	0.4	0.4	0.1
	Gradual	0.1	0.7	0.5	0.2	1	0.4
	Recurrent/Seasonal	-	15.5	-	11.8	-	9.9
Fluctuating	Abrupt	0.1	39.3	0.7	34.7	0.4	31
	Rapid	0.3	37.9	0.5	34.8	0.9	30.7
	Gradual	0.3	51.6	0.4	45.4	0.8	40.6
	Recurrent/Seasonal	-	57.2	0	56.2	0	50.7
Subgroup	Abrupt	0.2	0	0.4	0	0.5	0
	Rapid	0.3	0	0.5	0	0.4	0
	Gradual	0.2	0	0.5	0	0.4	0
	Recurrent/Seasonal	-	2.5	0	0.3	0	1.2
Random	Abrupt	0	0	0.4	0.2	0.6	0.1
	Rapid	0	0.1	0.5	0	0.4	0.1
	Gradual	0	0	0.2	0.1	1.1	0
	Recurrent/Seasonal	-	4	-	3.3	-	1.7

Figure 55. Time to detection by speed of transition, form of change, and δ.

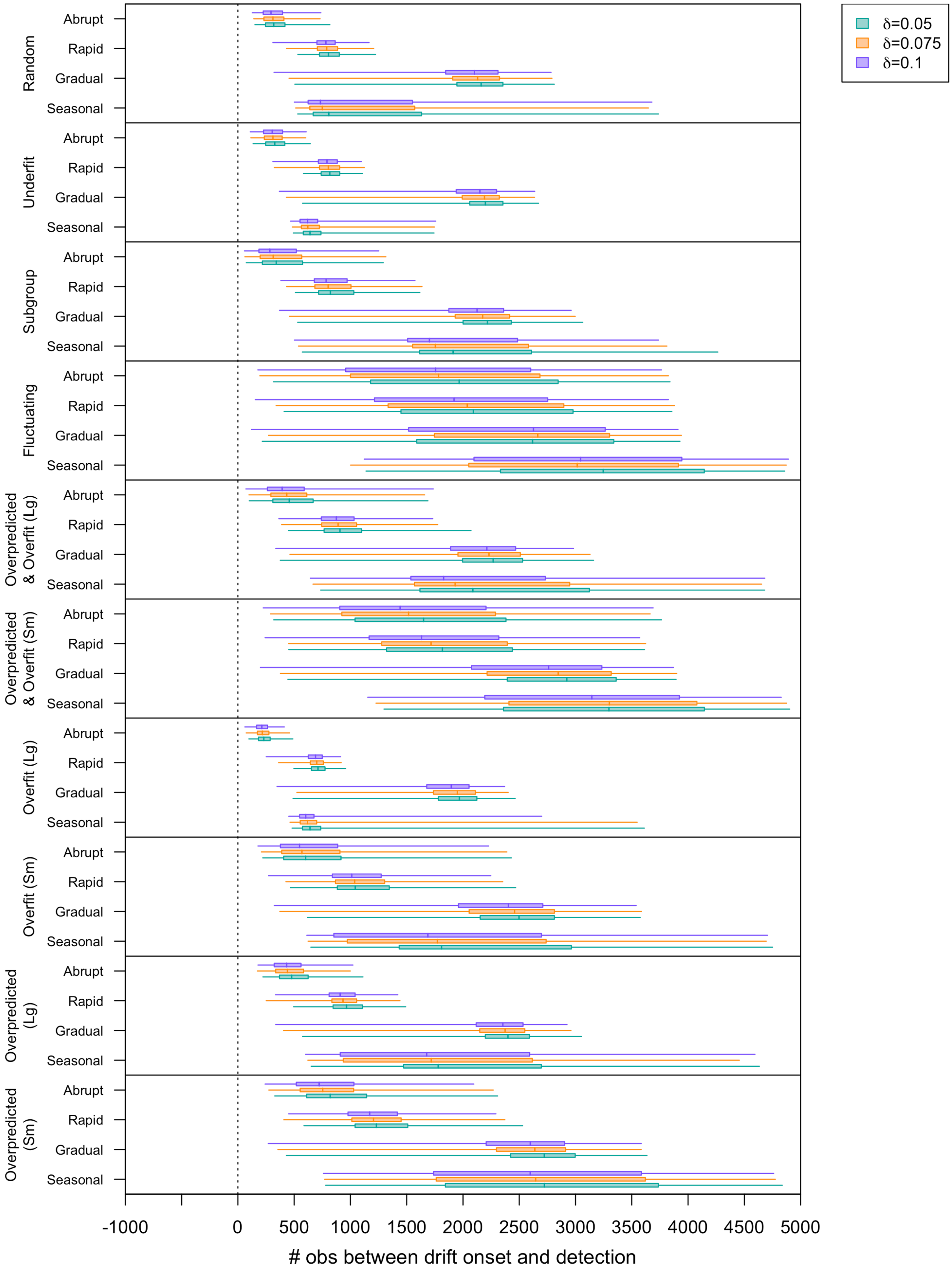


Figure 56. Lag to detection by speed of transition, form of change, and δ.

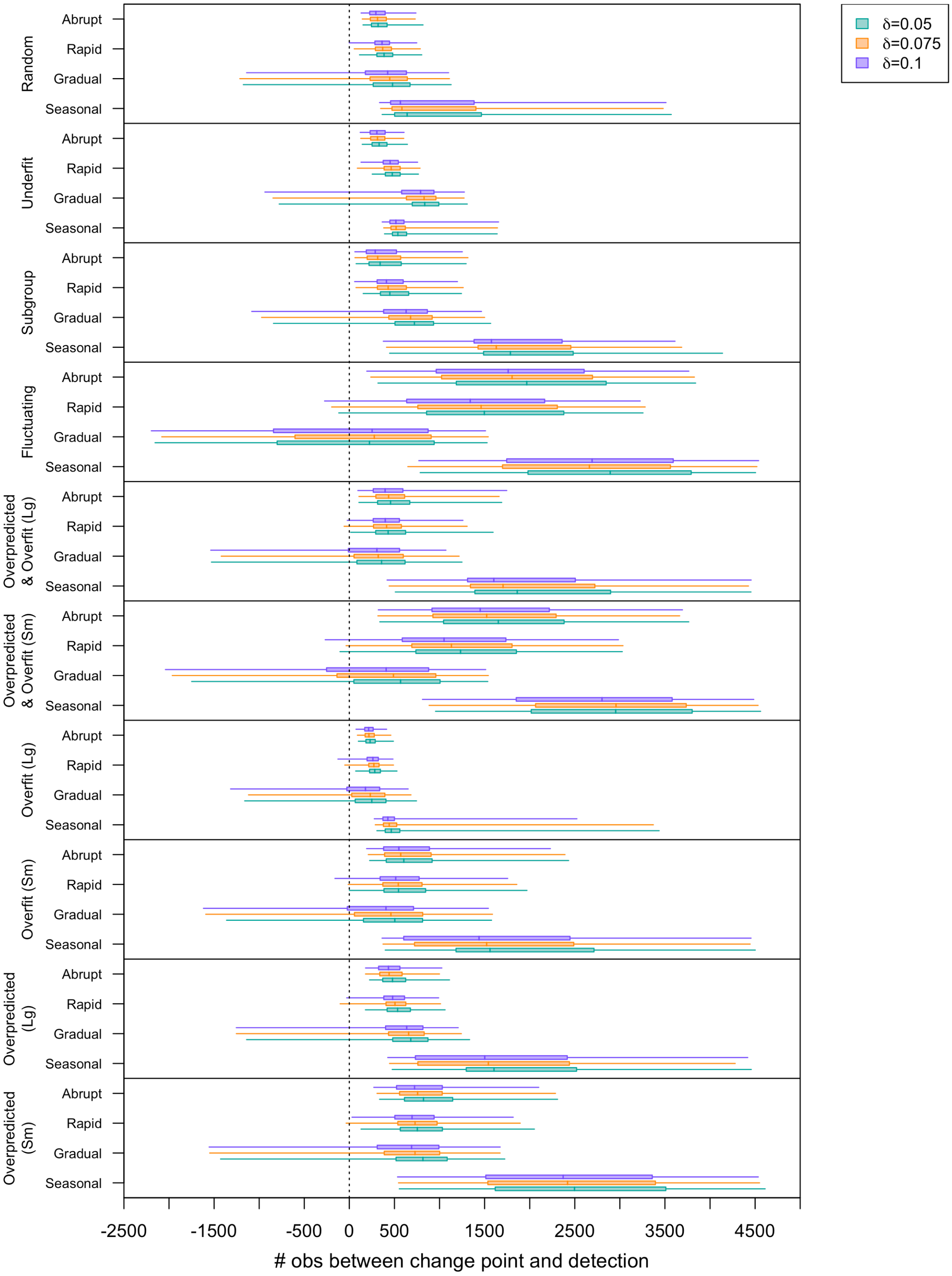


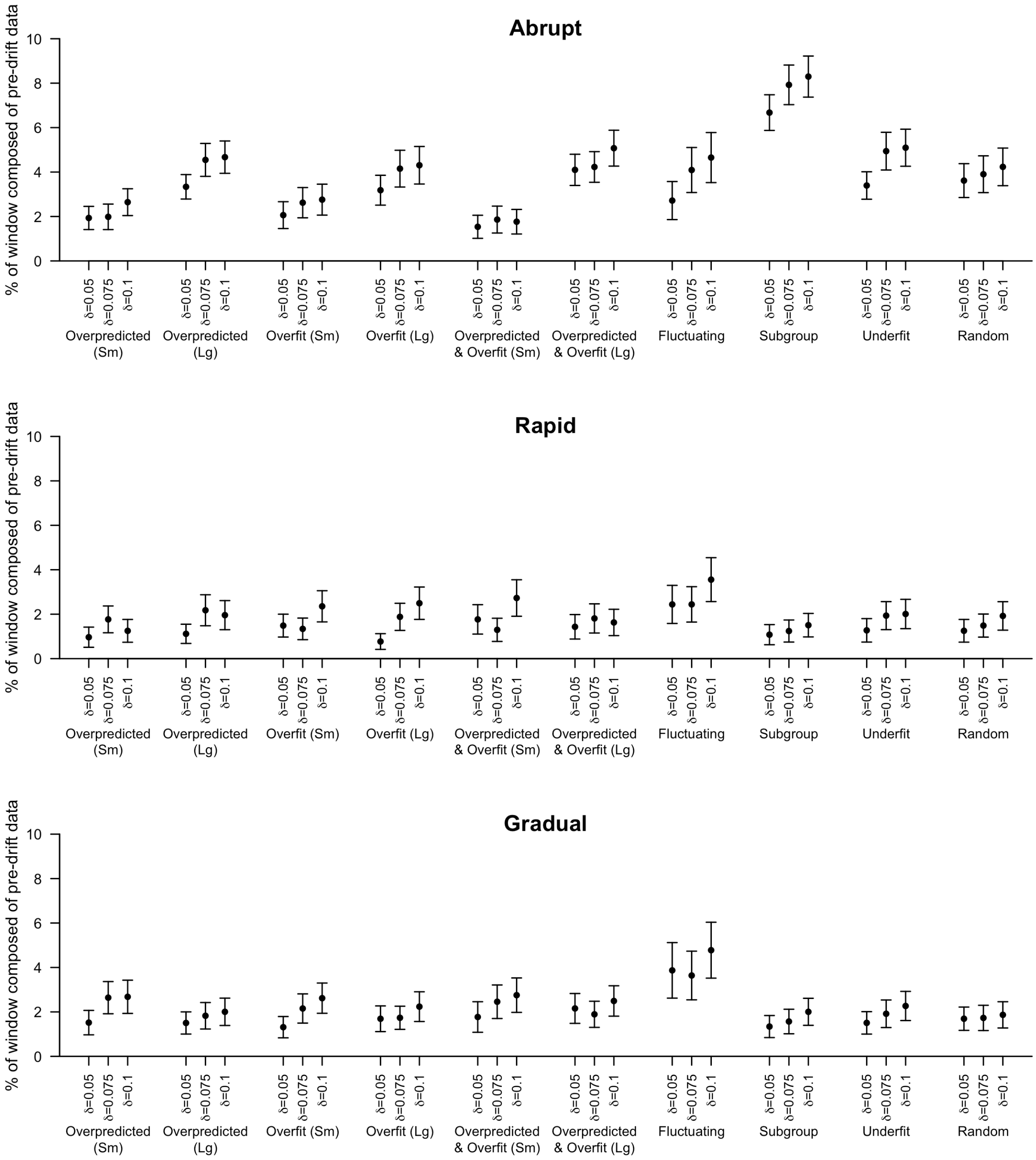
Table 21. Properties of retained data windows after drift detection by δ.
Window size reported as median and inter-quartile range. Window compositions reported as percent of detections for which pre-drift observations where included in the returned data window (% PD).

		$\delta = 0.075$		$\delta = 0.1$	
Post-drift calibration setting	**Transition pattern**	**Size**	**% PD**	**Size**	**% PD**
Overpredicted (small)	Abrupt	522 (394, 719)	9	505 (380, 719)	13
	Rapid	539 (430, 763)	4.1	521 (398, 744)	3.3
	Gradual	709 (489, 986)	6.8	692 (489, 988)	6.7
	Recurrent/Seasonal	555 (402, 1070)	-	538 (387, 1089)	-
Overpredicted (large)	Abrupt	368 (289, 464)	26.2	362 (280, 469)	27.2
	Rapid	396 (322, 516)	4.6	399 (315, 525)	4.9
	Gradual	592 (448, 849)	5	581 (436, 837)	5.6
	Recurrent/Seasonal	398 (317, 582)	-	379 (300, 529)	-
Overfit (small)	Abrupt	371 (259, 551)	8	346 (249, 524)	8.8
	Rapid	394 (270, 576)	3.8	384 (268, 555)	5.7
	Gradual	549 (364, 888)	5.4	548 (369, 856)	7.6
	Recurrent/Seasonal	385 (266, 687)	-	342 (249, 538)	-
Overfit (large)	Abrupt	163 (130, 210)	16.5	159 (127, 198)	15.8
	Rapid	227 (176, 302)	4.7	220 (170, 293)	5.5
	Gradual	430 (284, 644)	5.2	418 (273, 651)	5.2
	Recurrent/Seasonal	211 (172, 268)	-	204 (165, 253)	-
Underfit	Abrupt	220 (195, 258)	29.6	214 (190, 253)	29.9
	Rapid	254 (221, 321)	4.5	247 (217, 322)	4.8
	Gradual	448 (333, 656)	5.3	461 (332, 707)	6.4
	Recurrent/Seasonal	236 (211, 266)	-	225 (202, 259)	-
Overpredicted & overfit (small)	Abrupt	925 (644, 1308)	7.2	909 (603, 1317)	7.2
	Rapid	892 (618, 1236)	3.8	838 (587, 1138)	6.1
	Gradual	912 (648, 1296)	6.7	902 (592, 1264)	7.6
	Recurrent/Seasonal	1056 (566, 1462)	-	945 (555, 1398)	-
Overpredicted & overfit (large)	Abrupt	358 (258, 491)	24.8	339 (240, 481)	28
	Rapid	394 (283, 529)	4	399 (276, 537)	3.7
	Gradual	596 (426, 844)	5.2	563 (398, 813)	6.7
	Recurrent/Seasonal	461 (311, 1047)	-	430 (296, 883)	-
Fluctuating	Abrupt	686 (442, 1083)	12.8	714 (432, 1085)	13
	Rapid	806 (488, 1228)	7.3	766 (451, 1117)	9.5
	Gradual	941 (551, 1390)	10.3	907 (528, 1374)	11.9
	Recurrent/Seasonal	749 (436, 1261)	-	753 (412, 1166)	-

Table 21 continued.

		$\delta = 0.075$		$\delta = 0.1$	
Post-drift calibration setting	**Transition pattern**	**Size**	**% PD**	**Size**	**% PD**
Subgroup	Abrupt	320 (220, 473)	53.6	290 (210, 444)	56.4
	Rapid	392 (294, 505)	3.5	383 (289, 502)	4.1
	Gradual	604 (472, 834)	4.6	588 (456, 821)	5.2
	Recurrent/Seasonal	454 (337, 935)	-	421 (314, 735)	-
Random	Abrupt	229 (182, 295)	13.2	224 (173, 282)	15.8
	Rapid	264 (202, 371)	4.3	262 (203, 358)	4.6
	Gradual	467 (306, 722)	4.8	451 (303, 693)	5
	Recurrent/Seasonal	262 (210, 368)	-	263 (204, 363)	-

Figure 57. Proportion and 95% confidence interval of observations in the retained window generated prior to drift onset by δ. Note, not relevant for recurrent/seasonal transitions in which there is no pre-drift period.



APPENDIX B

SPECIFYING THE ADAPTIVE WINDOW ERROR TOLERANCE IN OUR CALIBRATION DRIFT DETECTION SYSTEM

In order to provide some insight into the influence of δ during periods of stable model performance, we generated non-transitioning timeseries for the calibrated model and each of the 10 miscalibrated models considered. Stable timeseries were generated from each model with n = {5000, 50000, 100000, 250000} observations. Over 1,000 iterations of each series length, we documented the proportion of iterations falsely detecting a change as δ increased from 0.01 to 0.2. The table below documents false alarm rates for each scenario. The minimum δ value for which the false alarm rate exceeded 0.05 is highlighted in bold.

Table 22. Proportion of iterations falsely detecting a change in stable timeseries by δ.

Post-drift calibration setting	Series length	δ = 0.05	δ = 0.1	δ = 0.15	δ = 0.2	δ = 0.21	δ = 0.22	δ = 0.23	δ = 0.24	δ = 0.25
Calibrated	5000	0	0	0	0.001	0.002	0.009	0.021	**0.497**	0.494
	50000	0	0	0	0.01	0.019	0.023	**0.072**	0.991	0.993
	100000	0	0	0	0.004	0.011	0.021	**0.074**	1	0.999
	250000	0	0	0	0.004	0.011	0.021	**0.074**	1	1
Overpredicted (small)	5000	0	0	0	0	0.003	0.008	0.043	**0.927**	0.93
	50000	0	0	0	0.003	0.004	0.009	**0.052**	1	1
	100000	0	0	0	0.001	0.002	0.008	**0.057**	1	1
	250000	0	0	0	0.001	0.002	0.008	**0.057**	1	1
Overpredicted (large)	5000	0	0	0	0	0.001	0.002	**0.077**	0.914	0.903
	50000	0	0	0	0.001	0.002	0.005	**0.092**	1	1
	100000	0	0	0	0	0	0.003	**0.079**	1	1
	250000	0	0	0	0	0	0.003	**0.079**	1	1
Overfit (small)	5000	0	0	0	0.001	0.003	0.008	**0.068**	0.997	0.998
	50000	0	0	0	0	0	0.005	**0.076**	1	1
	100000	0	0	0	0.002	0.004	0.011	**0.082**	1	1
	250000	0	0	0	0.002	0.004	0.011	**0.082**	1	1
Overfit (large)	5000	0	0	0	0.002	0.004	0.018	**0.167**	1	0.999
	50000	0	0	0	0.001	0.004	0.019	**0.172**	1	1
	100000	0	0	0	0.001	0.002	0.021	**0.186**	1	1
	250000	0	0	0	0.001	0.002	0.021	**0.186**	1	1
Underfit	5000	0	0	0	0	0	0.007	**0.089**	0.807	0.809
	50000	0	0	0	0.001	0.004	0.016	**0.135**	0.994	0.995
	100000	0	0	0	0.001	0.001	0.012	**0.113**	0.991	0.998
	250000	0	0	0	0.001	0.002	0.018	**0.125**	0.996	0.997

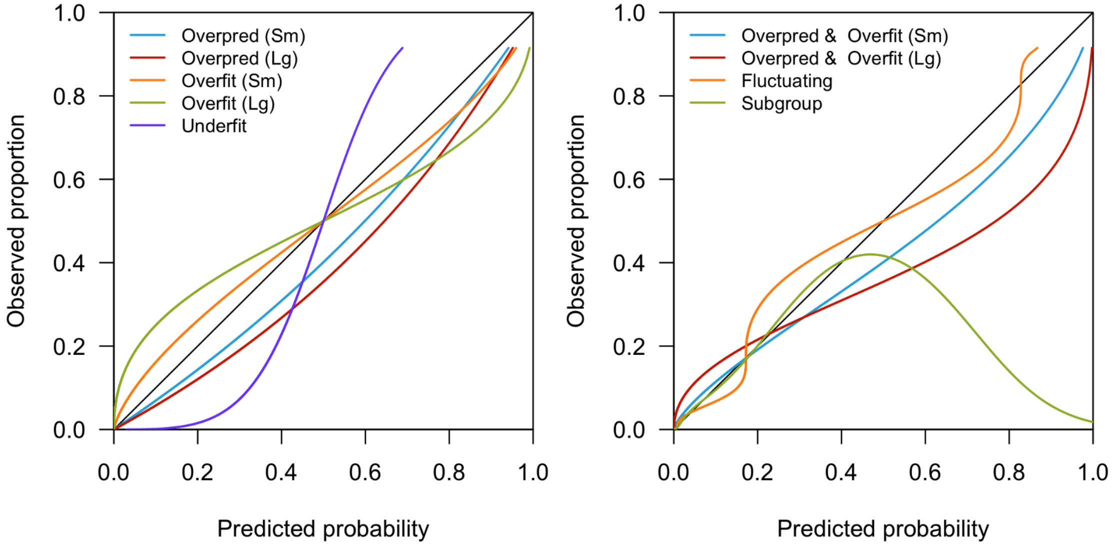
Table 22 continued.

Post-drift calibration setting	**Series length**	**δ = 0.05**	**δ = 0.1**	**δ = 0.15**	**δ = 0.2**	**δ = 0.21**	**δ = 0.22**	**δ = 0.23**	**δ = 0.24**	**δ = 0.25**
Overpredicted & overfit (small)	5000	0	0	0	0.001	0.002	0.008	**0.058**	0.62	0.641
	50000	0	0	0	0.001	0.004	0.014	**0.101**	0.999	0.999
	100000	0	0	0	0.002	0.007	0.024	**0.101**	1	1
	250000	0	0	0	0.002	0.005	0.022	**0.104**	1	1
Overpredicted & overfit (large)	5000	0	0	0	0.003	0.008	**0.053**	0.184	0.797	0.78
	50000	0	0	0	0.003	0.01	**0.056**	0.212	1	1
	100000	0	0	0	0.002	0.013	**0.059**	0.193	1	1
	250000	0	0	0	0.003	0.012	**0.053**	0.192	1	1
Fluctuating	5000	0	0	0	0.001	0.002	0.009	**0.067**	0.526	0.538
	50000	0	0	0.001	0.001	0.006	0.033	**0.18**	0.988	0.987
	100000	0	0	0	0.001	0.005	0.029	**0.158**	0.995	0.994
	250000	0	0	0	0.001	0.004	0.029	**0.14**	0.999	1
Subgroup	5000	0	0	0	0.009	0.016	0.03	**0.058**	0.168	0.157
	50000	0	0	0	0.04	**0.093**	0.188	0.314	0.875	0.899
	100000	0	0	0	0.045	**0.092**	0.192	0.353	0.984	0.981
	250000	0	0	0	0.042	**0.106**	0.206	0.369	1	1
Random	5000	0	0	0	0.015	0.034	**0.077**	0.225	0.821	0.841
	50000	0	0	0	0.028	**0.053**	0.129	0.324	1	1
	100000	0	0	0	0.025	**0.055**	0.124	0.3	1	1
	250000	0	0	0	0.025	**0.055**	0.124	0.3	1	1

Appendix C

SIMULATION STUDY DESIGN DETAILS FOR THE CALIBRATION DRIFT DETECTION SYSTEM

This appendix provides additional detail on the design of the simulation study described in Chapter 5. We simulated predictions with prespecified forms of miscalibration. The calibration curves associated with each form of miscalibrations are displayed in the figure below. Miscalibrated probabilities were constructed by transforming randomly generated true probabilities using the following equations:



1. Overprediction – Systematic overprediction was created by varying the intercept of the Cox recalibration equation.

$$\text{logit}(p_{true}) = \alpha + \beta * \text{logit}(p_{pred})$$

   A small degree systematic overprediction was created by setting β = 1 and α = - 0.4.
   A larger degree systematic overprediction was created by setting β = 1 and α = - 0.6.

2. Overfitting – Two levels of overfitting were created by varying the slope of Cox recalibration equation. A relatively small degree of overfitting was created by setting α = 0 and β= 0.75. A larger degree of overfitting was created by setting α = 0 and β= 0.5.

3. Combined overprediction and overfitting – Combined overprediction and overfitting was constructed using combinations of the intercept and slope values above. A moderate degree of combined overprediction and overfitting was defined with α = -0.4 and β= 0.75. A larger degree of combined overprediction and overfitting was defined with α = -0.6 and β= 0.5.
4. Underfitting – Underfitting was created by setting the coefficients of Cox recalibration equation to β = 3 and α = 0.
5. Fluctuating – Miscalibration was designed to fluctuate over the range of probability. This was achieved with the following equation:

$$\text{logit}(p_{true}) = \ 0.5 * \sin\left(2 * \text{logit}(p_{pred})\right) + \ \text{logit}(p_{pred})$$

6. Subgroup – Miscalibration was created by assigning substantially overpredicted probabilities to a subgroup of low risk observations. We randomly sampled 30% of observations with $p_{true} < 0.2$ and predicted probabilities were assigned to be $p_{true} + 0.7$.
7. Random – Predicted probabilities were defined by randomizing the set of true probabilities.

APPENDIX D

SIMULATION STUDY DESIGN DETAILS FOR THE NONPARAMETRIC TESTING PROCEDURE

This appendix provides additional detail on the design of the simulation study described in Chapter 6.

Predictor generation

Default case mix. The following predictor generation model was used to simulate the development population and the updating/evaluation populations for the no population shift, event rate shift, and predictor-outcome association shift scenarios.

$[X_1, X_2, \ldots X_{15}]^T \sim N_{15}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$where all $\mu_i = 0$ and $\boldsymbol{\Sigma}$ is the symmetric covariance matrix defined as follows:

	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{12}	X_{13}	X_{14}	X_{15}
X_1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X_2		4	0.25	0	0	0	0	-1	0	0	0	0	0	0.5	0
X_3			9	0	0.5	0	0	0	0	0	0	0	0	0	-0.25
X_4				16	0	0	0	0	0	0	0	0	0	0	0
X_5					1	0	0	2.5	0	0	0	0	0	0	-0.5
X_6						4	0	0.5	0	0	0	0.25	0	1	0
X_7							9	0	0	0	0	0	0	0	0
X_8								16	0	0	0	0	0	0	0
X_9									1	0	1	-0.5	0	0	0
X_{10}										4	0	0	0	0	0
X_{11}											9	-0.25	0	0	0
X_{12}												1	0	0	0
X_{13}													4	0	0
X_{14}														1	0.25
X_{15}															4

$X_{16} \sim Gamma(2, 2)$

$X_{17} \sim Gamma(5, 1)$

$X_{18} \sim Gamma(2, 0.5)$

$X_{19} \sim Gamma(0.5, 1)$

$X_{20} \sim Gamma(1, 2)$

$X_{21} \sim Bernoulli(0.1)$

$X_{22} \sim Bernoulli(0.2)$

$X_{23} \sim Bernoulli(0.3)$

$X_{24} \sim Bernoulli(0.4)$

$X_{25} \sim Bernoulli(0.5)$

$X_{26} \sim Poisson(1)$

$X_{27} \sim Poisson(2)$

$X_{28} \sim Poisson(4)$

$X_{29} \sim Poisson(6)$

$X_{30} \sim Poisson(8)$

$X_{31} \sim Multinomial(0.25, 0.25, 0.5)$

$X_{32} \sim Multinomial(0.33, 0.33, 0.34)$

More homogenous case mix. The default predictor generation model used for the development population was adjusted as follows to simulate the updating/evaluation populations for the more homogenous/less variable case mix scenario.

Variances of the multivariate normal predictors $X_1, X_2, \ldots, X_{15}$ were adjusted to the values specified in the table below. Correlations among these predictors were not adjusted.

Table 23. Variance of multivariate normal predictors under case mix shift scenarios.

Variable	More Homogenous Case Mix	More Heterogenous Case Mix
X_1	0.5625	1.5625
X_2	2.25	6.25
X_3	5.0625	14.0625
X_4	9	25
X_5	0.5625	1.5625
X_6	2.25	6.25
X_7	5.0625	14.0625
X_8	16	25
X_9	0.5625	1.5625
X_{10}	2.25	6.25
X_{11}	9	14.0625
X_{12}	0.5625	1.5625
X_{13}	2.25	6.25
X_{14}	1	1.5625
X_{15}	9	6.25

Distributions of the following predictors were also adjusted to decrease variance in the population.

$X_{17} \sim Gamma(5, 0.25)$

$X_{18} \sim Gamma(2, 0.1)$

$X_{19} \sim Gamma(0.5, 0.5)$

$X_{26} \sim Poisson(0.5)$

$X_{27} \sim Poisson(1)$

$X_{28} \sim Poisson(2)$

$X_{24} \sim Bernoulli(0.2)$

$X_{25} \sim Bernoulli(0.1)$

More heterogenous case mix. The default predictor generation model used for the training population was adjusted as follows to simulate the updating/evaluation populations for the more heterogenous/more variable case mix scenario.

Variances of the multivariate normal predictors $X_1, X_2, \dots, X_{15}$ were adjusted to the values specified in Table 23. Correlations among these variables were also decreased. The adjusted covariance matrix was defined as follows:

	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{12}	X_{13}	X_{14}	X_{15}
X_1	1.5625	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X_2		6.25	0.2	0	0	0	0	-0.8	0	0	0	0	0	0.4	0
X_3			14.0625	0	0.4	0	0	0	0	0	0	0	0	0	-0.2
X_4				25	0	0	0	0	0	0	0	0	0	0	0
X_5					1.5625	0	0	2	0	0	0	0	0	0	-0.4
X_6						6.25	0	0.4	0	0	0	0.2	0	0.8	0
X_7							14.0625	0	0	0	0	0	0	0	0
X_8								25	0	0	0	0	0	0	0
X_9									1.5625	0	0.8	-0.4	0	0	0
X_{10}										6.25	0	0	0	0	0
X_{11}											14.0625	-0.2	0	0	0
X_{12}												1.5625	0	0	0
X_{13}													6.25	0	0
X_{14}														1.5625	0.2
X_{15}															6.25

Distributions of the following predictors were also adjusted to increase variance in the population.

$X_{17} \sim Gamma(5, 3)$

$X_{18} \sim Gamma(2, 1)$

$X_{19} \sim Gamma(0.5, 2)$

$X_{21} \sim Bernoulli(0.5)$

$X_{22} \sim Bernoulli(0.4)$

$X_{26} \sim Poisson(3)$

$X_{27} \sim Poisson(4)$

$X_{28} \sim Poisson(5)$

Outcome generation

Two logistic regression models were defined by the following equation:

$$(Y = 1|X) = [1 + exp\{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 x_5 + \beta_6 x_6 + \beta_7 x_7 + \beta_8 x_8 + \beta_9 x_9 + \beta_{10} x_{10} + \beta_{11} x_{11} + \beta_{12} x_{12} + \beta_{13} x_{13} + \beta_{14} x_{14} + \beta_{15} x_{15} + \beta_{16} x_{16} + \beta_{17} x_{17} + \beta_{18} x_{18} + \beta_{19} x_{19} + \beta_{20} x_{20} + \beta_{21} x_{21} + \beta_{22} x_{22} + \beta_{23} x_{23} + \beta_{24} x_{24} + \beta_{25} x_{25} + \beta_{26} x_{26} + \beta_{27} x_{27} + \beta_{28} x_{28} + \beta_{29} x_{29} + \beta_{30} x_{30} + \beta_{31} x_{31b} + \beta_{32} x_{31c} + \beta_{33} x_{32b} + \beta_{34} x_{32c} + \beta_{35} x_1 x_{23} + \beta_{36} x_5 x_{22} + \beta_{37} x_{11} x_{12} + \beta_{38} x_{12} x_{13} + \beta_{39} x_6 x_{26} + \beta_{40} x_3 x_{28})\}]^{-1}$$

x_{31b} = dummy variable for 2nd level of X_{31}
x_{31c} = dummy variable for 3rd level of X_{31}

x_{32b} = dummy variable for 2nd level of X_{32}
x_{32c} = dummy variable for 3rd level of X_{32}

For the model with df =10, coefficients for select variables were set to 0 (i.e., odds ratios set to 1), reducing the model form to $P(Y = 1|X) = [1 + exp\{-(\beta_0 + \beta_2 X_2 + \beta_3 X_3 + \beta_5 X_5 + \beta_6 X_6 + \beta_8 X_8 + \beta_{16} X_{16} + \beta_{22} X_{22} + \beta_{24} X_{24} + \beta_{29} X_{29} + \beta_{36} X_5 * X_{22})\}]^{-1}$

Using the coefficient values defined below, each observation in the training, updating, and evaluation populations were assigned a probability using both the df =10 and df =40 models. A binary outcome under both models was defined by comparing these probabilities to random values generated from a uniform [0,1] distribution. If the random value was less than or equal to the assigned probability, the observation was assigned $Y = 1$, otherwise the observation was assigned $Y = 0$.

Default coefficients. For the training population and the updating/evaluation populations under no population shift and case mix shift scenarios, model effects were defined as noted in Table 24. Intercepts were set to establish a population event rate of 25%.

Change in outcome prevalence adjustments. For the updating/evaluation populations under the event rate shift scenario, intercepts were set to establish a population event rate of 30%.

Predictor-outcome association shift adjustments. For the updating/evaluation populations under the association shift scenario, half of the odds ratios for predictors in each model multiplied by 120%. Revised model effects are noted in Table 24.

Table 24. Odds ratios for model effects under each predictor-outcome association scheme.

	Default associations		**Adjusted associations**	
Variable	**Model with df=10**	**Model with df=40**	**Model with df=10**	**Model with df=40**
X_1	1	1.25	1	1.5
X_2	1.1	1.1	1.1	1.1
X_3	1.25	0.95	1.25	1.14
X_4	1	1	1	1
X_5	1.5	0.5	1.8	0.6
X_6	0.9	0.9	0.9	0.9
X_7	1	1.05	1	1.26
X_8	0.75	0.75	0.9	0.75
X_9	1	0.9	1	1.08
X_{10}	1	1.25	1	1.25
X_{11}	1	1.05	1	1.26
X_{12}	1	1.1	1	1.1
X_{13}	1	0.5	1	0.6
X_{14}	1	1.05	1	1.05
X_{15}	1	1	1	1.2
X_{16}	0.5	0.5	0.6	0.5
X_{17}	1	1	1	1.2
X_{18}	1	1.05	1	1.05
X_{19}	1	1.1	1	1.32
X_{20}	1	0.95	1	0.95
X_{21}	1	2	1	2.4
X_{22}	1.1	1.1	1.32	1.1
X_{23}	1	0.75	1	0.9
X_{24}	0.9	0.9	0.9	0.9
X_{25}	1	1.5	1	1.8
X_{26}	1	1.1	1	1.1
X_{27}	1	1.01	1	1.212
X_{28}	1	0.95	1	0.95
X_{29}	2	1	2.4	1.2
X_{30}	1	1.75	1	1.75
X_{31b}	1	1.1	1	1.32
X_{31c}	1	1.75	1	1.75
X_{32b}	1	0.95	1	1.14
X_{32c}	1	1.25	1	1.25
$X_1 X_{23}$	1	1	1	1.2
$X_5 X_{22}$	1.1	0.95	1.1	0.95
$X_{11} X_{12}$	1	0.99	1	1.188
$X_{12} X_{13}$	1	1.01	1	1.01
$X_6 X_{26}$	1	0.99	1	1.188
$X_3 X_{28}$	1	1.025	1	1.025

APPENDIX E

DETAILED RESULTS OF THE CASE STUDIES FOR OUR NONPARAMETRIC TESTING PROCEDURE

Figure 58. Calibration curves in the three months after updating for the original acute kidney injury model, the refit model, and the recommended update (if different).

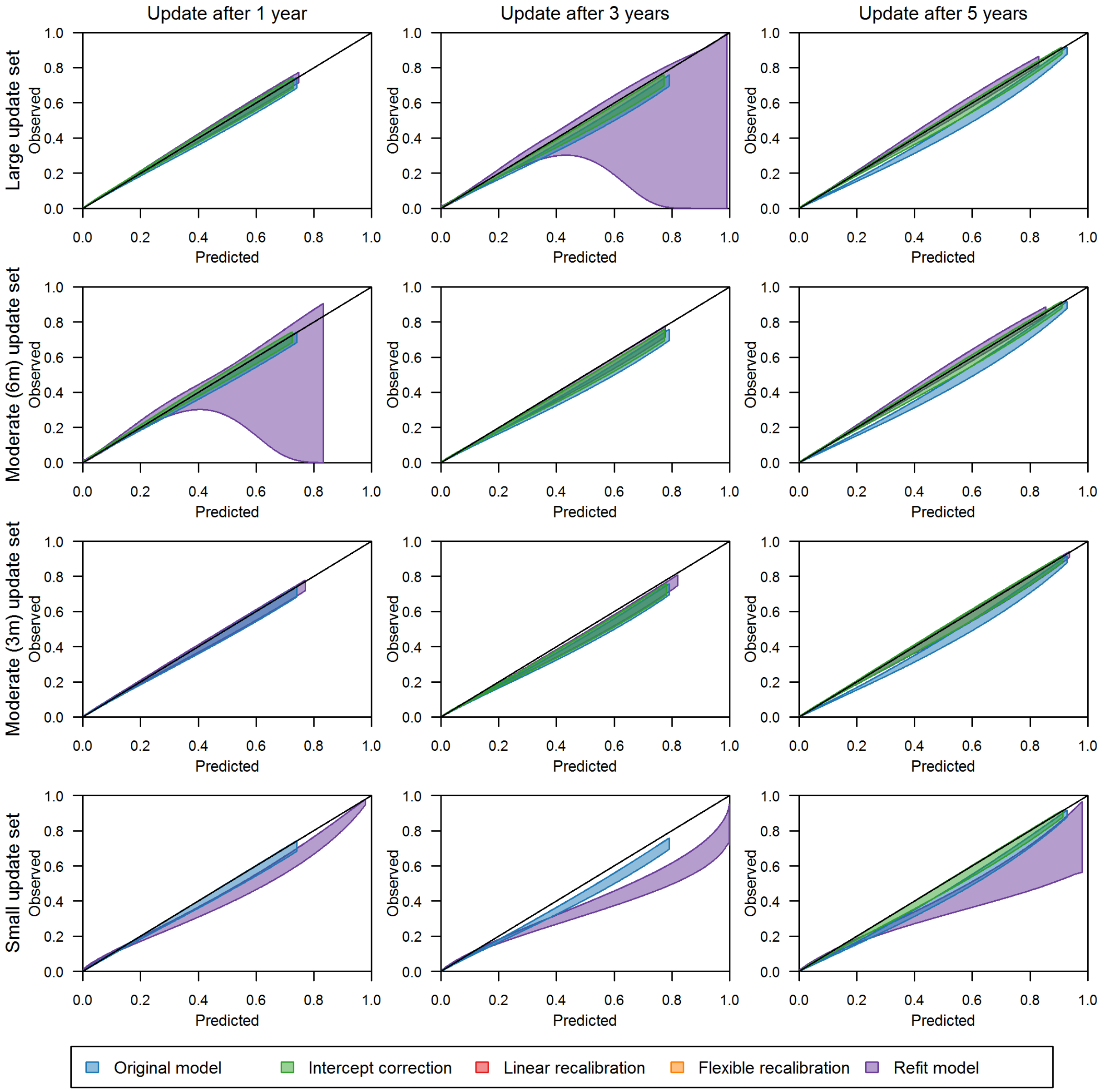


Figure 59. Calibration curves in the three months after updating the acute kidney injury model with three levels of recalibration.

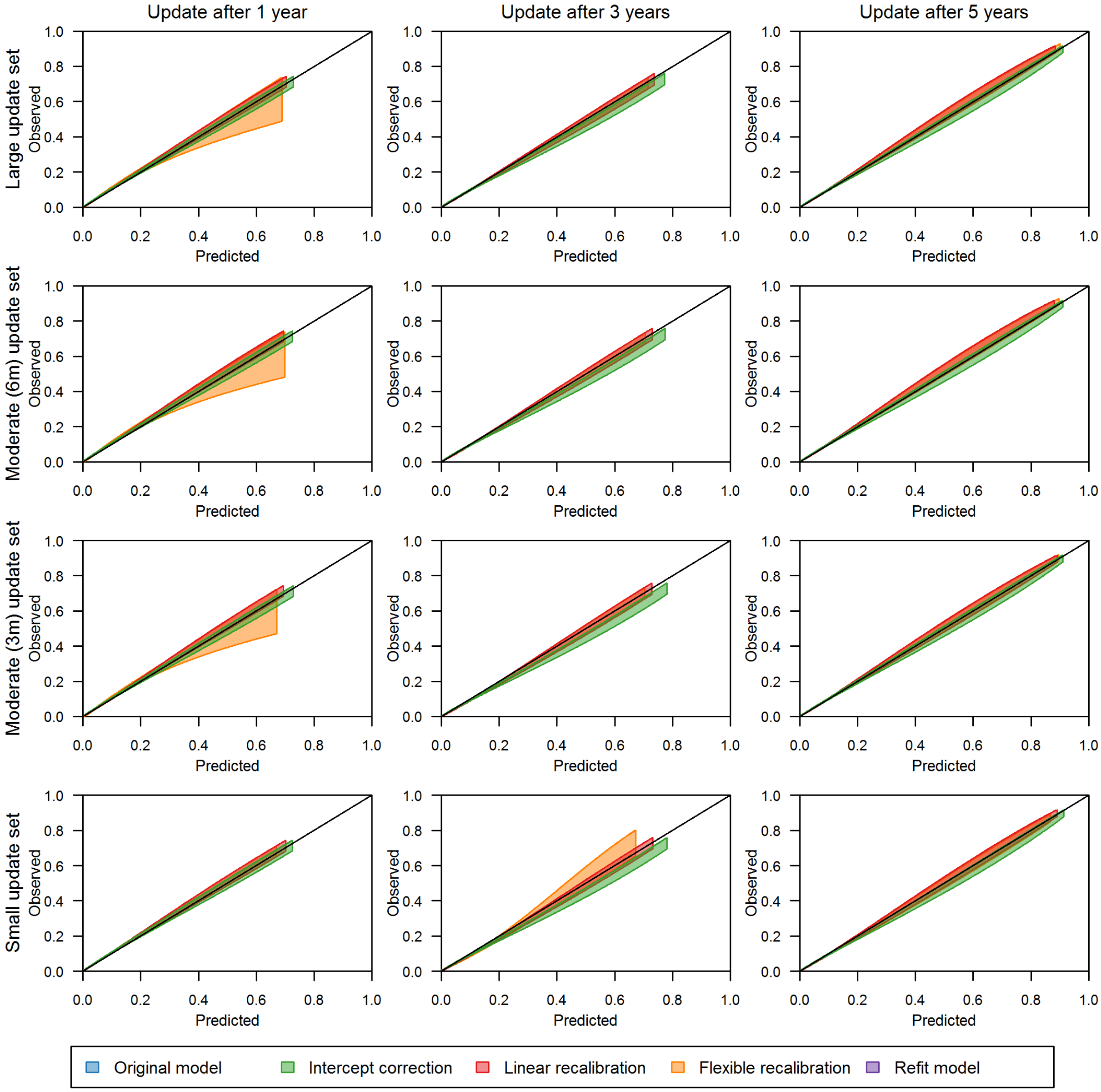


Figure 60. Calibration curves in the three months after updating for the original 30-day mortality model, the refit model, and the recommended update (if different).

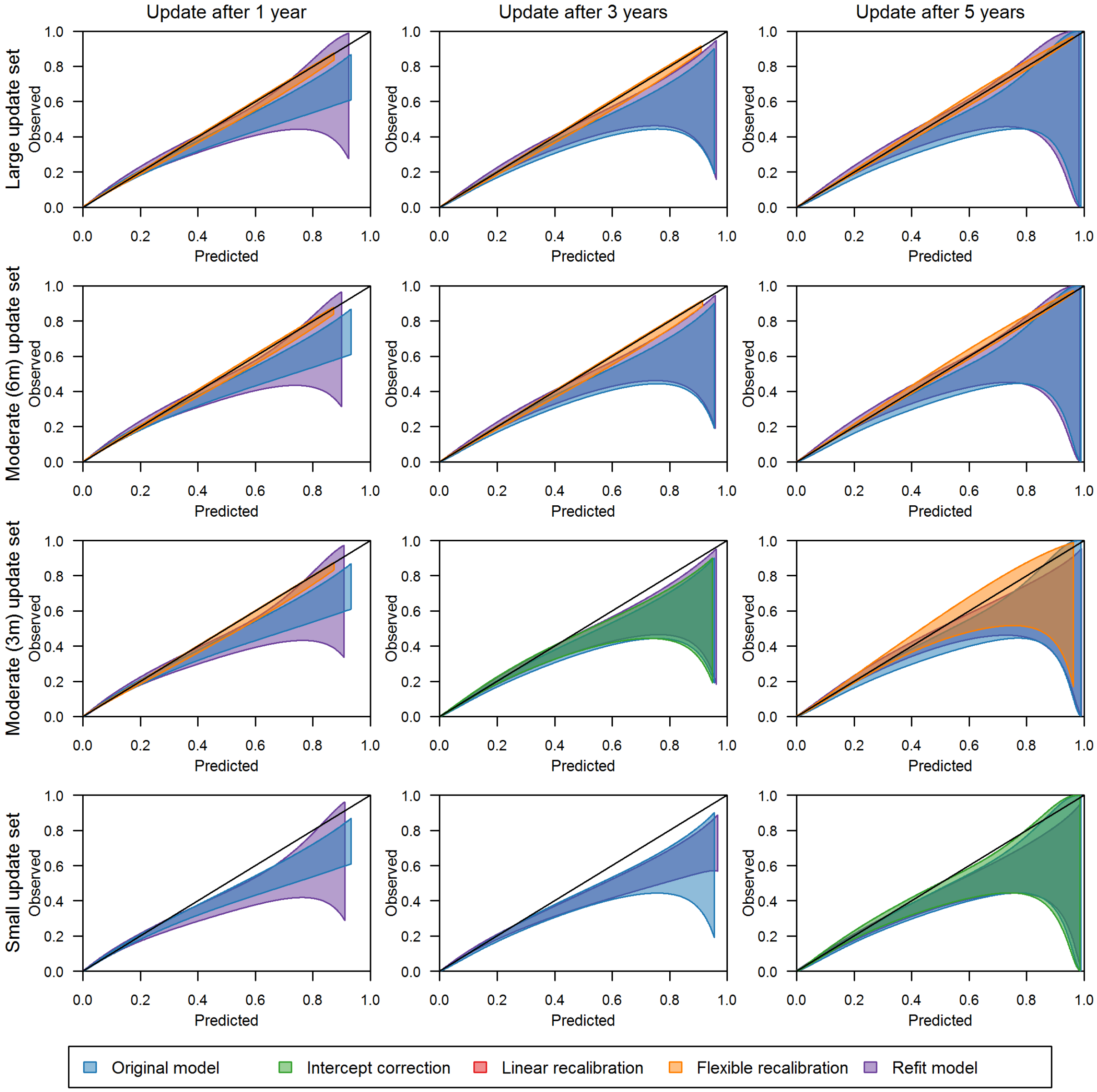
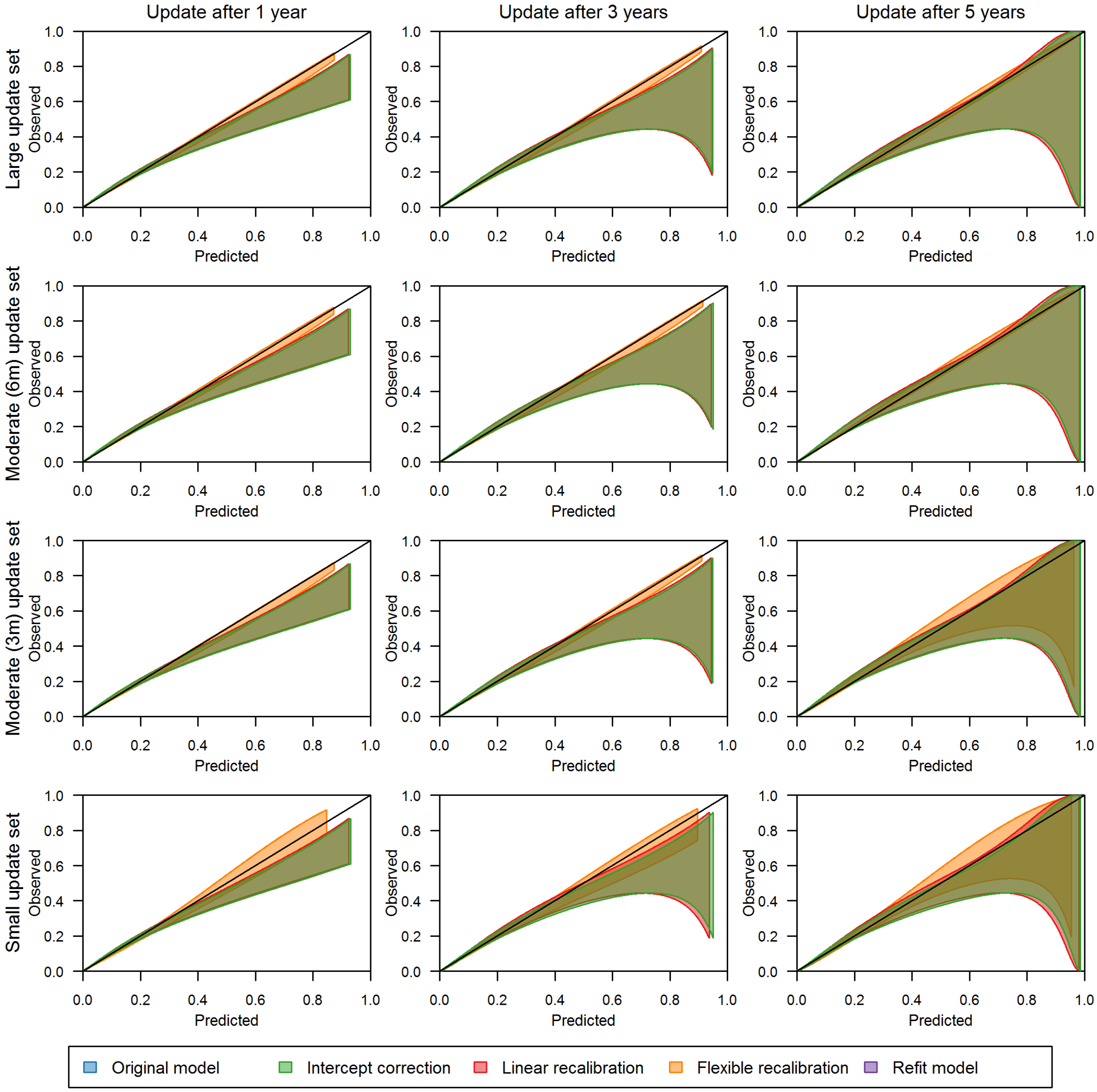


Figure 61. Calibration curves in the three months after updating the 30-day mortality model with three levels of recalibration.



REFERENCES

1. Hall, L.M., R.T. Jung, and G.P. Leese. 2003. *Controlled trial of effect of documented cardiovascular risk scores on prescribing.* BMJ. 326(7383): 251-2.

2. Feldman, M., R. Stanford, A. Catcheside, and A. Stotter. 2002. *The use of a prognostic table to aid decision making on adjuvant therapy for women with early breast cancer.* European Journal of Surgical Oncology 28(6): 615-9.

3. Amarasingham, R., et al. 2013. *Allocating scarce resources in real-time to reduce heart failure readmissions: A prospective, controlled study.* BMJ Quality & Safety. 22(12): 998-1005.

4. Amarasingham, R., R.E. Patzer, M. Huesch, N.Q. Nguyen, and B. Xie. 2014. *Implementing electronic health care predictive analytics: Considerations and challenges.* Health Affairs. 33(7): 1148-54.

5. Jarman, B., et al. 2010. *The hospital standardised mortality ratio: A powerful tool for dutch hospitals to assess their quality of care?* BMJ Quality & Safety. 19(1): 9-13.

6. Steyerberg, E.W., et al. 2013. *Prognosis research strategy (progress) 3: Prognostic model research.* PLoS Medicine. 10(2): e1001381.

7. Ohno-Machado, L., F.S. Resnic, and M.E. Matheny. 2006. *Prognosis in critical care.* Annual Review of Biomedical Engineering. 8: 567-99.

8. Matheny, M.E., et al. 2010. *Development of inpatient risk stratification models of acute kidney injury for use in electronic health records.* Medical Decision Making. 30(6): 639-50.

9. Kansagara, D., et al. 2011. *Risk prediction models for hospital readmission: A systematic review.* JAMA. 306(15): 1688-98.

10. Toll, D.B., K.J. Janssen, Y. Vergouwe, and K.G. Moons. 2008. *Validation, updating and impact of clinical prediction rules: A review.* Journal of Clinical Epidemiology. 61(11): 1085-94.

11. Moons, K.G., et al. 2012. *Risk prediction models: Ii. External validation, model updating, and impact assessment.* Heart. 98(9): 691-8.

12. Sajda, P. 2006. *Machine learning for detection and diagnosis of disease.* Annual Review of Biomedical Engineering. 8: 537-65.

13. Steyerberg, E.W., T. van der Ploeg, and B. Van Calster. 2014. *Risk prediction with machine learning and regression methods.* Biometrical Journal. 56(4): 601-6.

14. Pencina, M.J. and E.D. Peterson. 2016. *Moving from clinical trials to precision medicine: The role for predictive modeling.* JAMA. 315(16): 1713-4.

15. Parikh, R.B., M. Kakad, and D.W. Bates. 2016. *Integrating predictive analytics into high-value care: The dawn of precision delivery.* JAMA. 315(7): 651-2.

16. Kourou, K., T.P. Exarchos, K.P. Exarchos, M.V. Karamouzis, and D.I. Fotiadis. 2015. *Machine learning applications in cancer prognosis and prediction.* Computational and Structural Biotechnology Journal. 13: 8-17.

17. Goldstein, B.A., A.M. Navar, M.J. Pencina, and J.P. Ioannidis. 2017. *Opportunities and challenges in developing risk prediction models with electronic health records data: A systematic review.* Journal of the American Medical Informatics Association. 24(1): 198-208.

18. Davis, S.E., T.A. Lasko, G. Chen, and M.E. Matheny. 2017. *Calibration drift among regression and machine learning models for hospital mortality.* Proceedings of the AMIA Annual Symposium.

19. Davis, S.E., T.A. Lasko, G. Chen, E.D. Siew, and M.E. Matheny. 2017. *Calibration drift in regression and machine learning models for acute kidney injury.* Journal of the American Medical Informatics Association. 24(6): 1052-61.

20. Minne, L., et al. 2012. *Effect of changes over time in the performance of a customized saps-ii model on the quality of care assessment.* Intensive Care Medicine. 38(1): 40-6.

21. Steyerberg, E.W. 2009l. *Clinical prediction models: A practical approach to development, validation, and updating*. New York, NY: Spring.

22. Moons, K.G., D.G. Altman, Y. Vergouwe, and P. Royston. 2009. *Prognosis and prognostic research: Application and impact of prognostic models in clinical practice.* BMJ. 338: b606.

23. Hickey, G.L., et al. 2013. *Dynamic trends in cardiac surgery: Why the logistic euroscore is no longer suitable for contemporary cardiac surgery and implications for future risk models.* European Journal of Cardio-thoracic Surgery. 43(6): 1146-52.

24. Kappen, T.H., et al. 2012. *Adaptation of clinical prediction models for application in local settings.* Medical Decision Making. 32(3): E1-10.

25. Debray, T.P., et al. 2015. *A new framework to enhance the interpretation of external validation studies of clinical prediction models.* Journal of Clinical Epidemiology. 68(3): 279-89.

26. Janssen, K.J., K.G. Moons, C.J. Kalkman, D.E. Grobbee, and Y. Vergouwe. 2008. *Updating methods improved the performance of a clinical prediction model in new patients.* Journal of Clinical Epidemiology. 61(1): 76-86.

27. Siregar, S., et al. 2014. *Improved prediction by dynamic modelling: An exploratory study in the adult cardiac surgery database of the netherlands association for cardio-thoracic surgery.* Interactive Cardiovascular and Thoracic Surgery. 19: S8.

28. Jenkins, D.A., M. Sperrin, G.P. Martin, and N. Peek. 2018. *Dynamic models to predict health outcomes: Current status and methodological challenges.* Diagnostic and Prognostic Research. 2(23).

29. Minne, L., et al. 2012. *Statistical process control for monitoring standardized mortality ratios of a classification tree model.* Methods of Information in Medicine. 51(4): 353-8.

30. Steyerberg, E.W., G.J. Borsboom, H.C. van Houwelingen, M.J. Eijkemans, and J.D. Habbema. 2004. *Validation and updating of predictive logistic regression models: A study on sample size and shrinkage.* Statistics in Medicine. 23(16): 2567-86.

31. Hannan, E.L., K. Cozzens, S.B. King, 3rd, G. Walford, and N.R. Shah. 2012. *The new york state cardiac registries: History, contributions, limitations, and lessons for future efforts to assess and publicly report healthcare outcomes.* Journal of the American College of Cardiology. 59(25): 2309-16.

32. Jin, R., A.P. Furnary, S.C. Fine, E.H. Blackstone, and G.L. Grunkemeier. 2010. *Using society of thoracic surgeons risk models for risk-adjusting cardiac surgery results.* Annals of Thoracic Surgery. 89(3): 677-82.

33. Vergouwe, Y., et al. 2017. *A closed testing procedure to select an appropriate method for updating prediction models.* Statistics in Medicine. 36(28): 4529-39.

34. Bifet, A. and R. Gavaldà. Yearl. *Learning from time-changing data with adaptive windowing*. in *Proceedings of the 2007 SIAM international conference on data mining*. SIAM.

35. Davis, S.E., et al. 2019. *A nonparametric updating method to correct clinical prediction model drift.* Journal of the American Medical Informatics Association. Online August 12, 2019.

36. Davis, S.E., R.A. Greevy, T.A. Lasko, C.G. Walsh, and M.E. Matheny. 2019. *Comparison of prediction model performance updating protocols: Using a data-driven testing procedure to guide updating.* Proceedings of the AMIA Annual Symposium. (Accepted).

37. Steyerberg, E.W., et al. 2010. *Assessing the performance of prediction models: A framework for traditional and novel measures.* Epidemiology. 21(1): 128-38.

38. Matheny, M.E., L. Ohno-Machado, and F.S. Resnic. 2005. *Discrimination and calibration of mortality risk prediction models in interventional cardiology.* Journal of Biomedical Informatics. 38(5): 367-75.

39. Van Calster, B., et al. 2016. *A calibration hierarchy for risk models was defined: From utopia to empirical data.* Journal of Clinical Epidemiology. 74: 167-76.

40. Van Calster, B. and A.J. Vickers. 2015. *Calibration of risk prediction models: Impact on decision-analytic performance.* Medical Decision Making. 35(2): 162-9.

41. Shah, N.D., E.W. Steyerberg, and D.M. Kent. 2018. *Big data and predictive analytics: Recalibrating expectations.* JAMA. 320(1): 27-8.

42. Jiang, X., M. Osl, J. Kim, and L. Ohno-Machado. 2012. *Calibrating predictive model estimates to support personalized medicine.* Journal of the American Medical Informatics Association. 19(2): 263-74.

43. Mikkelsen, M.M., S.P. Johnsen, P.H. Nielsen, and C.J. Jakobsen. 2012. *The euroscore in western denmark: A population-based study.* Journal of Cardiothoracic and Vascular Anesthesia. 26(2): 258-64.

44. Harrison, D.A., et al. 2014. *External validation of the intensive care national audit & research centre (icnarc) risk prediction model in critical care units in scotland.* BMC Anesthesiology. 14: 116.

45. Barili, F., A. Capo, E. Ardemagni, F. Rosato, and C. Grossi. 2012. *Trend analysis of euroscore performance: A prospective tenyear experience.* Giornale Italiano di Cardiologia. 2: 171S.

46. Arvis, P., P. Lehert, and H.L.A. Guivarc. 2012. *Simple adaptations to the templeton model for ivf outcome prediction make it current and clinically useful.* Human Reproduction. 27(10): 2971-8.

47. Harrison, D.A., A.R. Brady, G.J. Parry, J.R. Carpenter, and K. Rowan. 2006. *Recalibration of risk prediction models in a large multicenter cohort of admissions to adult, general critical care units in the united kingdom.* Critical Care Medicine. 34(5): 1378-88.

48. Cook, D.A., et al. 2002. *Prospective independent validation of apache iii models in an australian tertiary adult intensive care unit.* Anaesthesia and Intensive Care. 30(3): 308-15.

49. Paul, E., M. Bailey, A. Van Lint, and V. Pilcher. 2012. *Performance of apache iii over time in australia and new zealand: A retrospective cohort study.* Anaesthesia and Intensive Care. 40(6): 980-94.

50. Rogers, F.B., et al. 2012. *Has triss become an anachronism? A comparison of mortality between the national trauma data bank and major trauma outcome study databases.* Journal of Trauma and Acute Care Surgery. 73(2): 326-31.

51. Madan, P., M.A. Elayda, V.V. Lee, and J.M. Wilson. 2011. *Risk-prediction models for mortality after coronary artery bypass surgery: Application to individual patients.* International Journal of Cardiology. 149(2): 227-31.

52. Hekmat, K., et al. 2005. *Daily assessment of organ dysfunction and survival in intensive care unit cardiac surgical patients.* Annals of Thoracic Surgery. 79(5): 1555-62.

53. Janssen, K.J., Y. Vergouwe, C.J. Kalkman, D.E. Grobbee, and K.G. Moons. 2009. *A simple method to adjust clinical prediction models to local circumstances.* Canadian Journal of Anesthesia. 56(3): 194-201.

54. Pencina, M.J., et al. 2014. *Application of new cholesterol guidelines to a population-based sample.* New England Journal of Medicine. 370(15): 1422-31.

55. Hripcsak, G. and D.J. Albers. 2013. *Next-generation phenotyping of electronic health records.* Journal of the American Medical Informatics Association. 20(1): 117-21.

56. Larcher, A., et al. 2019. *The learning curve for robot-assisted partial nephrectomy: Impact of surgical experience on perioperative outcomes.* European Urology. 75(2): 253-6.

57. Wright, A., et al. 2016. *Analysis of clinical decision support system malfunctions: A case series and survey.* Journal of the American Medical Informatics Association. 23(6): 1068-76.

58. Li, R.C., et al. 2018. *Impact of problem-based charting on the utilization and accuracy of the electronic problem list.* Journal of the American Medical Informatics Association. 25(5): 548-54.

59. Lindenauer, P.K., T. Lagu, M.S. Shieh, P.S. Pekow, and M.B. Rothberg. 2012. *Association of diagnostic coding with trends in hospitalizations and mortality of patients with pneumonia, 2003-2009.* JAMA. 307(13): 1405-13.

60. Danciu, I., et al. 2014. *Secondary use of clinical data: The vanderbilt approach.* Journal of Biomedical Informatics. 52: 28-35.

61. Pivovarov, R., D.J. Albers, G. Hripcsak, J.L. Sepulveda, and N. Elhadad. 2014. *Temporal trends of hemoglobin a1c testing.* Journal of the American Medical Informatics Association. 21(6): 1038-44.

62. McCormick, T.H., A.E. Raftery, D. Madigan, and R.S. Burd. 2012. *Dynamic logistic regression and dynamic model averaging for binary classification.* Biometrics. 68(1): 23-30.

63. Hickey, G.L., et al. 2013. *Dynamic prediction modeling approaches for cardiac surgery.* Circulation: Cardiovascular Quality and Outcomes. 6(6): 649-58.

64. Diamond, G.A. 1992. *What price perfection? Calibration and discrimination of clinical prediction models.* Journal of Clinical Epidemiology. 45(1): 85-9.

65. Wessler, B.S., et al. 2015. *Clinical prediction models for cardiovascular disease: Tufts predictive analytics and comparative effectiveness clinical prediction model database.* Circulation: Cardiovascular Quality and Outcomes. 8(4): 368-75.

66. Dalton, J.E. 2013. *Flexible recalibration of binary clinical prediction models.* Statistics in Medicine. 32(2): 282-9.

67. Su, T.L., T. Jaki, G.L. Hickey, I. Buchan, and M. Sperrin. 2018. *A review of statistical updating methods for clinical prediction models.* Statistical Methods in Medical Research. 27(1): 185-97.

68. Gama, J., I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia. 2014. *A survey on concept drift adaptation.* ACM Computing Surveys (CSUR). 46(4): 44.

69. U.S. Food & Drug Administration Center for Devices and Radiological Health, *Proposed regulatory framework for modifications to artificial intelligence/machine learning (ai/ml) - based software as a medican device (samd)*. 2019.

70. Van Calster, B., et al. 2017. *Validation and updating of risk models based on multinomial logistic regression.* Diagnostic and Prognostic Research. 1(2).

71. Nishida, K. and K. Yamauchi. Yearl. *Detecting concept drift using statistical testing*. in *International Conference on Discovery Science*. Springer.

72. Baena-Garcıa, M., et al., *Early drift detection method*, in *Fourth International Workshop on Knowledge Discovery From Data Streams*. 2006. p. 77-86.

73. Ross, G.J., N.M. Adams, D.K. Tasoulis, and D.J. Hand. 2012. *Exponentially weighted moving average charts for detecting concept drift.* Pattern Recognition Letters. 33(2): 191-8.

74. Chen, K., Y.S. Koh, and P. Riddle, *Tracking drift severity in data streams*, in *Australasian Joint Conference on Artificial Intelligence*. 2015, Springer. p. 96-108.

75. Gama, J., P. Medas, G. Castillo, and P. Rodrigues. Yearl. *Learning with drift detection*. in *Brazilian Symposium on Artificial Intelligence*. Springer.

76. Benneyan, J.C., R.C. Lloyd, and P.E. Plsek. 2003. *Statistical process control as a tool for research and healthcare improvement.* BMJ Quality & Safety. 12(6): 458-64.

77. Thor, J., et al. 2007. *Application of statistical process control in healthcare improvement: Systematic review.* BMJ Quality & Safety. 16(5): 387-99.

78. Benneyan, J.C., et al. 2011. *Illustration of a statistical process control approach to regional prescription opioid abuse surveillance.* Journal of Addiction Medicine. 5(2): 99-109.

79. Matheny, M.E., L. Ohno-Machado, and F.S. Resnic. 2008. *Risk-adjusted sequential probability ratio test control chart methods for monitoring operator and institutional mortality rates in interventional cardiology.* American Heart Journal. 155(1): 114-20.

80. Matheny, M.E., et al. 2011. *Evaluation of an automated safety surveillance system using risk adjusted sequential probability ratio testing.* BMC Medical Informatics and Decision Making. 11: 75.

81. Morton, A.P., et al. 2001. *The application of statistical process control charts to the detection and monitoring of hospital-acquired infections.* Journal of Quality in Clinical Practice. 21(4): 112-7.

82. Baker, A.W., et al. 2018. *Performance of statistical process control methods for regional surgical site infection surveillance: A 10-year multicentre pilot study.* BMJ Quality & Safety. 27(8): 600-10.

83. Seim, A., B. Andersen, and W.S. Sandberg. 2006. *Statistical process control as a tool for monitoring nonoperative time.* Anesthesiology. 105(2): 370-80.

84. Pimentel, L. and F. Barrueto, Jr. 2015. *Statistical process control: Separating signal from noise in emergency department operations.* Journal of Emergency Medicine. 48(5): 628-38.

85. Minne, L., et al. 2012. *Statistical process control for validating a classification tree model for predicting mortality--a novel approach towards temporal validation.* Journal of Biomedical Informatics. 45(1): 37-44.

86. Steiner, S.H. 2014l. *Risk-adjusted monitoring of outcomes in health care*. in *Statistics in action: A canadian outlook*. p. 225-41.

87. Gonçalves Jr, P.M., S.G. de Carvalho Santos, R.S. Barros, and D.C. Vieira. 2014. *A comparative study on concept drift detectors.* Expert Systems with Applications. 41(18): 8144-56.

88. Austin, P.C. and E.W. Steyerberg. 2014. *Graphical assessment of internal and external calibration of logistic regression models by using loess smoothers.* Statistics in Medicine. 33(3): 517-35.

89. Austin, P.C. and E.W. Steyerberg. 2019. *The integrated calibration index (ici) and related metrics for quantifying the calibration of logistic regression models.* Statistics in Medicine. 38(21): 4051-65.

90. Van Hoorde, K., S. Van Huffel, D. Timmerman, T. Bourne, and B. Van Calster. 2015. *A spline-based tool to assess and visualize the calibration of multiclass risk predictions.* Journal of Biomedical Informatics. 54: 283-93.

91. Nattino, G., S. Finazzi, and G. Bertolini. 2016. *A new test and graphical tool to assess the goodness of fit of logistic regression models.* Statistics in Medicine. 35(5): 709-20.

92. Harrell, F. 2001l. *Regression modeling strategies: With applications to linear models, logistic regression, and survival analysis*. Springer series in statistics. New York: Springer.

93. Quinonero-Candela, J., M. Sugiyama, A. Schwaighofer, and N. Lawrence. 2009l. *Dataset shift in machine learning*. Cambridge, MA: The MIT Press.

94. Laskov, P., C. Gehl, and S. Kruger. 2006. *Incremental support vector learning: Analysis, implementation and applications.* Journal of Machine Learning Research. 7: 1909-36.

95. Ruder, S. 2016. *An overview of gradient descent optimization algorithms.* arXiv preprint arXiv:1609.04747.

96. Miyaguchi, K. and H. Kajino. Yearl. *Cogra: Concept-drift-aware stochastic gradient descent for time-series forecasting*. in *Proceedings of the AAAI Conference on Artificial Intelligence*.

97. Losing, V., B. Hammer, and H. Wersing. 2018. *Incremental on-line learning: A review and comparison of state of the art algorithms.* Neurocomputing. 275: 1261-74.

98. Kingma, D.P. and J. Ba. 2014. *Adam: A method for stochastic optimization.* arXiv preprint arXiv:1412.6980.

99. Srinivasan, V., A.R. Sankar, and V.N. Balasubramanian. Yearl. *Adine: An adaptive momentum method for stochastic gradient descent*. in *Proceedings of the ACM India Joint International Conference on Data Science and Management of Data*. ACM.

100. Royston, P., G. Ambler, and W. Sauerbrei. 1999. *The use of fractional polynomials to model continuous risk variables in epidemiology.* International Journal of Epidemiology. 28(5): 964-74.

101. Royston, P. 2017. *Model selection for univariable fractional polynomials.* The Stata Journal. 17(3): 619-29.

102. Grulich, P.M., et al. Yearl. *Scalable detection of concept drifts on data streams with parallel adaptive windowing*. in *21st International Conference on Extending Database Technology (EDBT)*.

103. Brier, G.W. 1950. *Verification of forecasts expressed in terms of probability.* Monthly Weather Review. 75(1): 1-3.

104. Tsymbal, A. 2004. *The problem of concept drift: Definitions and related work.* Computer Science Department Trinity College Dublin. 106(2): 58.

105. Murphy-Filkins, R., D. Teres, S. Lemeshow, and D.W. Hosmer. 1996. *Effect of changing patient mix on the performance of an intensive care unit severity-of-illness model: How to distinguish a general from a specialty intensive care unit.* Critical Care Medicine. 24(12): 1968-73.

106. Cronin, R.M., et al. 2015. *National veterans health administration inpatient risk stratification models for hospital-acquired acute kidney injury.* Journal of the American Medical Informatics Association. 22(5): 1054-71.

107. Dalton, J.E., et al. 2011. *Risk quantification for 30-day postoperative mortality and morbidity in non-cardiac surgical patients.* Anesthesiology. 114(6): 1336-44.

108. Yale New Haven Health Services Corporation Center for Outcomes Research and Evaluation, *2015 condition-specific measures updates and specifications report hospital-level 30-day risk-standardized mortality measures - version 4.0*. 2015.

109. Nattino, G., S. Finazzi, and G. Bertolini. 2014. *A new calibration test and a reappraisal of the calibration belt for the assessment of prediction models based on dichotomous outcomes.* Statistics in Medicine. 33(14): 2390–407.

110. Hanley, J.A. and B.J. McNeil. 1982. *The meaning and use of the area under a receiver operating characteristic (roc) curve.* Radiology. 143(1): 29-36.

111. Hendriksen, J.M., G.J. Geersing, K.G. Moons, and J.A. de Groot. 2013. *Diagnostic and prognostic prediction models.* Journal of Thrombosis and Haemostasis. 11 Suppl 1: 129-41.

112. Reilly, B.M. and A.T. Evans. 2006. *Translating clinical research into clinical practice: Impact of using prediction rules to make decisions.* Annals of Internal Medicine. 144(3): 201-9.

113. Altman, D.G., Y. Vergouwe, P. Royston, and K.G. Moons. 2009. *Prognosis and prognostic research: Validating a prognostic model.* BMJ. 338: b605.

114. Vickers, A.J. and E.B. Elkin. 2006. *Decision curve analysis: A novel method for evaluating prediction models.* Medical Decision Making. 26(6): 565-74.

115. Royston, P. and D.G. Altman. 2013. *External validation of a cox prognostic model: Principles and methods.* BMC Medical Research Methodology. 13: 33.

116. Lenert, M.C., M.E. Matheny, and C.G. Walsh. 2019. *Prognostic models will be victims of their own success, unless.* J Am Med Inform Assoc.

117. Breck, E., S. Cai, E. Nielsen, M. Salib, and D. Sculley. 2016. *What's your ml test score? A rubric for ml production systems.*

118. Walsh, C.G., K. Sharman, and G. Hripcsak. 2017. *Beyond discrimination: A comparison of calibration methods and clinical usefulness of predictive models of readmission risk.* Journal of Biomedical Informatics. 76: 9-18.

119. Bouwmeester, W., et al. 2012. *Reporting and methods in clinical prediction research: A systematic review.* PLoS Medicine. 9(5): 1-12.

120. Collins, G.S., et al. 2014. *External validation of multivariable prediction models: A systematic review of methodological conduct and reporting.* BMC Medical Research Methodology. 14: 40.

121. Mallett, S., P. Royston, R. Waters, S. Dutton, and D.G. Altman. 2010. *Reporting performance of prognostic models in cancer: A review.* BMC Medicine. 8: 21.

122. Ross, E.G., et al. 2016. *The use of machine learning for the identification of peripheral artery disease and future mortality risk.* Journal of Vascular Surgery. 64(5): 1515-22 e3.

123. VanHouten, J.P., J.M. Starmer, N.M. Lorenzi, D.J. Maron, and T.A. Lasko. 2014. *Machine learning for risk prediction of acute coronary syndrome.* Proceedings of the AMIA Annual Symposium. 2014: 1940-9.

124. Singal, A.G., et al. 2013. *Machine learning algorithms outperform conventional regression models in predicting development of hepatocellular carcinoma.* The American Journal of Gastroenterology. 108(11): 1723-30.

125. Dreiseitl, S. and L. Ohno-Machado. 2002. *Logistic regression and artificial neural network classification models: A methodology review.* Journal of Biomedical Informatics. 35(5-6): 352-9.

126. Austin, P.C., D.S. Lee, E.W. Steyerberg, and J.V. Tu. 2012. *Regression trees for predicting mortality in patients with cardiovascular disease: What improvement is achieved by using ensemble-based methods?* Biometrical Journal. 54(5): 657-73.

127. van der Ploeg, T., D. Nieboer, and E.W. Steyerberg. 2016. *Modern modeling techniques had limited external validity in predicting mortality from traumatic brain injury.* Journal of Clinical Epidemiology. 78: 83-9.

128. Ennis, M., G. Hinton, D. Naylor, M. Revow, and R. Tibshirani. 1998. *A comparison of statistical learning methods on the gusto database.* Statistics in Medicine. 17(21): 2501-8.

129. Feng, R., M. Badgeley, J. Mocco, and E.K. Oermann. 2018. *Deep learning guided stroke management: A review of clinical applications.* Journal of Neurointerventional Surgery. 10(4): 358-62.

130. Ravi, D., et al. 2017. *Deep learning for health informatics.* IEEE Journal of Biomedical and Health Informatics. 21(1): 4-21.