AI in education

MDPI

*Article*

# Explainable Machine Learning for Student Performance Prediction

**Yu Lu [1,*], Avinash Shashikala Rajendra [1], Jun Zhang [1,2] and Tian Zhao [1]**

[1] Department of Computer Science, College of Engineering and Applied Science, University of Wisconsin–Milwaukee, Milwaukee, WI 53211, USA; avinash@uwm.edu (A.S.R.); junzhang@uwm.edu (J.Z.); tzhao@uwm.edu (T.Z.)

[2] Department of Electrical Engineering, College of Engineering and Applied Science, University of Wisconsin–Milwaukee, Milwaukee, WI 53211, USA

* Correspondence: lu5@uwm.edu

**Abstract**

Early identification of at-risk students is crucial for timely pedagogical intervention. Determining which assessments instructors should prioritize is complicated by the fact that different eXplainable-AI (XAI) methods can produce conflicting rankings for the same predictive model. We develop a framework combining a sequential GRU model with two complementary XAI techniques, Gradient SHAP (attribution) and DiCE (counterfactuals), and evaluate it in a foundational Data Structures and Algorithms course. The framework produces predictions and explanations for every prefix length throughout the semester and quantifies inter-method agreement and intra-method stability using three disagreement metrics. Intersecting the top-$k$ features identified by both methods isolates a compact subset of assessments whose predictive role is confirmed across two fundamentally different explanation mechanisms. We interpret this cross-method agreement as a heuristic that increases confidence in identified features relative to single-method results, though not as evidence of causal validity. For individual students, the framework uses the intersection of the two types of explanations when it is non-empty; otherwise, the instructor chooses between SHAP's diagnostic view and DiCE's prescriptive view, with an optional check against the top-$k$ list. The resulting guidance is less susceptible to method-specific biases than analyses relying on a single method.

**Keywords:** student performance prediction; explainable artificial intelligence; explanation disagreement

## 1. Introduction

Understanding how learning analytics can enhance educational outcomes is increasingly essential for stakeholders (Alsariera et al., 2022; Rehman et al., 2024). While conventional AI models prioritize predictive performance, understanding prediction rationale is equally important for evidence-based decision-making. Combining model prediction with interpretability through eXplainable Artificial Intelligence (XAI) enables more informed instructional strategies (García-Martínez et al., 2023). Post-hoc methods achieve this by explaining prediction models externally: a model is first trained, then post-hoc methods interpret its decision-making process (Schwalbe & Finzel, 2024).

In educational contexts, machine learning models have successfully processed time-series datasets containing chronologically organized records like grades, attendance, and assignment completion rates. Both recurrent architectures (e.g., GRU, RNN) (He et al.,



Academic Editor: Yupei Zhang

Received: 2 March 2026
Revised: 24 April 2026
Accepted: 12 May 2026
Published: 1 June 2026

2020) and non-recurrent models (Adnan et al., 2022) have proven effective for early identification of at-risk students. Numerous studies have applied diverse predictive models alongside XAI techniques such as SHAP, LIME, and counterfactual explanations to interpret predictions of student performance across various educational settings, including individual courses (Alwarthan et al., 2022; Jang et al., 2022; Smith et al., 2022), first-year programs (Albreiki et al., 2022; Cagliero et al., 2021), virtual learning environments (Chen et al., 2022; Ujkani et al., 2024), and massive open online courses (Swamy et al., 2022).

However, post-hoc explanations often appear counterintuitive or ambiguous due to data complexity, model capability, method design, and implementation decisions. For instance, Hoq et al. (2023) found SHAP revealed unexpected patterns requiring additional mixture model analysis. More broadly, different explanation methods can yield conflicting results: Swamy et al. (2022) reported significant variations in feature importance rankings across five XAI methods, while Jang et al. (2022) observed substantial divergences between global and local SHAP interpretations. These disagreements extend beyond a single method family. Kommiya Mothilal et al. (2021) explored both attribution-based methods (SHAP, LIME) and counterfactual explanations (DiCE, WachterCF), finding frequent disagreements in feature importance rankings while emphasizing the complementary nature of these approaches. Pawlicki (2023) further demonstrated that SHAP explanations are sensitive to data perturbations such as feature shuffling, missing values, and Gaussian noise, raising concerns about reliability when data quality fluctuates. In a comprehensive investigation, Krishna et al. (2025) documented widespread explanation disagreements across both perturbation-based and gradient-based methods, finding that practitioners frequently relied on ad hoc heuristics to resolve conflicts, and advocated for principled quantitative frameworks to assess explanation discrepancies.

Among existing work, Tiukhova et al. (2024) is the closest related study. They combine drift analyses with SHAP-based feature-importance rankings to assess the stability of student-success prediction models across multiple academic years and to identify predictors that remain stable across cohorts. However, their analysis relies on a single attribution-based explanation approach and does not evaluate sequential explanation stability, meaning whether explanations remain consistent as additional assessments become available over time, nor does it benchmark SHAP against alternative explanation approaches, such as counterfactual explanations, to quantify cross-method disagreement. Two related questions therefore remain underexplored in the prior studies we have reviewed: how different types of explanation methods agree or diverge on sequential educational data under expanding-input settings, and how each method's own rankings stabilize as additional assessment information arrives.

The consequences of this gap are not purely methodological. In educational contexts, feature-importance rankings are used to decide where instructors focus limited attention—which assessments receive additional review, which students receive targeted tutoring, and how curricula are revised across semester offerings of the same course. When two equally reasonable explanation methods disagree on which assessments matter most, interventions grounded in one method's output may be systematically misdirected relative to interventions grounded in the other. Understanding the degree and structure of this disagreement is therefore a prerequisite for the responsible use of learning analytics in teaching practice, not only a technical question about explanation algorithms. This study treats the reliability of explanations as itself an educational concern, and the framework we develop is designed to support actionable, student-specific guidance for timely intervention with at-risk students, with the choice of target assessments grounded in agreement between two explanation methods rather than in any single method's output.

We address this gap through three design choices. First, we adopt an expanding-input setting in which the model is evaluated at every prefix length from 5 to 42, producing a trajectory of feature rankings across the semester rather than a single ranking at the final prediction point. This design is dictated by the structure of early at-risk identification, where the model must produce predictions and explanations from each partial assessment sequence available during the semester, and where the usefulness of an intervention depends on how early it can be issued. Second, we quantify explanation consistency from two complementary perspectives. Inter-method agreement measures whether Gradient SHAP and DiCE identify the same influential assessments at each prefix; intra-method stability measures whether a ranking derived from a partial semester survives the arrival of later evidence. Both are computed using three disagreement metrics (Feature Agreement, Rank Agreement, and Rank Correlation) introduced by Krishna et al. (2025), and both speak to a concern that instructors face in practice: whether an explanation produced from incomplete information will continue to hold as the semester progresses. Third, we translate cross-method verification into a concrete pedagogical workflow for individual student intervention. When the two methods' explanations agree for an at-risk student, the shared assessments carry both diagnostic and prescriptive interpretations and serve directly as intervention targets. When they disagree, the framework treats the two views as complementary rather than competing, and provides a fallback procedure in which the instructor selects one view and may optionally cross-reference it against the global-level top-$k$ list. Because the overall workflow is built on both methods rather than either alone, the resulting guidance reflects features less susceptible to the method-specific biases that single-method analyses cannot detect.

In implementing these design choices, we predict student success or failure in a computer science course using assessment score sequences from four semesters. We tune a GRU model through exhaustive grid search and apply Gradient SHAP (Lundberg & Lee, 2017) and DiCE (Wachter et al., 2017–2018). Pairing the two follows Kommiya Mothilal et al. (2021), who treat attribution and counterfactual methods as methodologically complementary. When both methods identify the same assessments as influential, we read their agreement as increased confidence in those assessments relative to either method alone. We do not treat it as evidence that the assessments causally determine outcomes. Two explanation methods can converge because of shared statistical patterns in the data rather than because the identified assessments actually drive student success or failure.

These considerations motivate three research questions that organize the remainder of this study:

- RQ1. How do inter-method agreement between Gradient SHAP and DiCE, and intra-method stability within each method, evolve as additional assessment scores become available during the semester?
- RQ2. Which assessments are consistently identified as most influential by both explanation methods, and how robust is this identification to the choice of explanation method?
- RQ3. How can the explanations from Gradient SHAP and DiCE be combined to support per-student guidance—for instructors planning intervention and for students tracking their own progress—both when the two methods agree and when they do not?

The contributions of this study are threefold:

- Sequential explanation-agreement analysis: We characterize how inter-method agreement and intra-method stability evolve with prefix length, extending the XAI disagreement framework of Krishna et al. (2025) from fixed-input to expanding-input settings and treating intra-method stability as an analysis distinct from cross-method comparison.

- Empirical comparison of explanation methods in an educational setting: Using three complementary disagreement metrics, we quantify the agreement between Gradient SHAP and DiCE applied to a GRU model of student outcomes, and we use the resulting intersection of top-ranked features to identify assessments whose predictive role is robust to choice of explanation method.
- Pedagogical framework: We translate the dual-method approach into per-student guidance for at-risk students, supporting both instructor-led intervention and student self-tracking. The framework prescribes intervention targets from the intersection of the two explanations when it exists, and provides a single-view fallback otherwise, with optional cross-reference to the global-level top-$k$ list.

## 2. Materials and Methods

This section outlines the datasets, preprocessing, modeling, and evaluation procedures. It also describes the explanation methods (Gradient SHAP and DiCE) and the feature-selection strategy used to assess explanation agreement.

### *2.1. Dataset Description*

This study uses interaction data from CompSci 351 (Data Structures & Algorithms) at the University of Wisconsin–Milwaukee, collected across four consecutive semesters and comprising 244 student sequences in total. The course is a required component of the Computer Science and Computer Engineering undergraduate curricula. As a conceptually demanding gateway course whose outcomes affect students' progression through later coursework, it is a setting where early, explainable prediction of at-risk students is practically valuable, which motivates its choice for this study.

The course was taught by the same instructor across all four semesters, using the same syllabus, the same set of assessment categories (Homework, Online Activity, Lab Exercise, Midterm, Final), the same category weights, and the same release schedule relative to the semester timeline (i.e., each assessment was released in the same week of the term across semesters). Course topics are summarized in Table 1, and the weekly release schedule together with category weights is summarized in Table 2. Within each assessment slot, the problems were equivalent in substance across semesters, testing the same concepts at the same difficulty level against the same grading rubric. For Homework and Lab Exercise assessments, which are predominantly programming tasks, this equivalence is particularly tight: problems sharing the same specification require essentially the same solution structure, with variation limited to inputs, test cases, and problem wording. Such surface-level modifications were used across semesters to mitigate answer-sharing while preserving the underlying task.

The analyzed student populations differ across semesters. The majority of students in each semester took the course for the first time, with a small minority of returning students who had previously been enrolled. Each semester's enrollment is therefore largely a fresh sample of the undergraduate population taking this course.

Each student record consists of a sequence of 42 chronological assessment scores (Online Activities, Lab Exercises, Homework items, Midterm, and Final) together with a final letter grade (A through F with plus/minus modifiers). Table 2 illustrates the temporal relationships among assessments and their categories. The dataset uses a "wide" layout in which each row represents one student's complete assessment sequence, differing from the traditional time-series "long" format.

**Table 1.** Typical Schedule of Topics in Recent Semesters.

| Week Number | Topic |
|---|---|
| 1 | Abstract Data Types |
| 2 | Dynamic Arrays |
| 3 | Iterators |
| 4 | Linked Lists |
| 5 | Linked List Variations |
| 6 | Generics, Linked Iterators |
| 7 | Stacks and Queues |
| 8 | Midterm Exam |
| 9 | Binary Search Trees |
| 10 | Navigating Trees |
| 11 | Maps |
| 12 | Hashing |
| 13 | Graphs |
| 14 | Sorting |
| 15 | Heaps |
| 16 | Review and Final Exam |

**Table 2.** Data Schema. The table uses a folded layout to clearly illustrate the relationships among assessments, such as the week each assessment was released and its corresponding category.

| Category | Online Activity | Lab Exercise | Homework | Exam |
|---|---|---|---|---|
| Weight (%) | 10 | 10 | 40 | 40 |
| Value Type | Nonnegative Int. | Nonnegative Int. | Nonnegative Int. | Nonnegative Int. |
| Maximum | 10 | 10 | 30 | 200 |
| Week Number | | | | |
| 1 | | Lab Exercise 1 | Homework 1 | |
| 2 | Activity 2 | Lab Exercise 2 | Homework 2 | |
| 3 | Activity 3 | Lab Exercise 3 | Homework 3 | |
| 4 | Activity 4 | Lab Exercise 4 | Homework 4 | |
| 5 | Activity 5 | Lab Exercise 5 | Homework 5 | |
| 6 | Activity 6 | Lab Exercise 6 | Homework 6 | |
| 7 | Activity 7 | Lab Exercise 7 | Homework 7 | |
| 8 | | | | Midterm |
| 9 | Activity 8 | Lab Exercise 8 | Homework 8 | |
| 10 | Activity 9 | Lab Exercise 9 | Homework 9 | |
| 11 | Activity 10 | Lab Exercise 10 | Homework 10 | |
| 12 | Activity 11 | Lab Exercise 11 | Homework 11 | |
| 13 | Activity 12 | Lab Exercise 12 | Homework 12 | |
| 14 | Activity 13 | Lab Exercise 13 | Homework 13 | |
| 15 | Activity 14 | | Homework 14 | |
| 16 | | | | Final |
| Final Course Outcome Levels/Labels: A, A−, B+, B, B−, C+, C, C−, D+, D, D−, F | | | | |

#### 2.1.1. Data Preprocessing

Following standard machine learning pipelines, we performed essential data preprocessing to understand the dataset and prepare it for model development. Although typical preprocessing may involve extensive integration, imputation, cleaning, transformation, and feature engineering, our dataset is relatively small, collected over a short time span, and contains simple data types with no missing records. We therefore adopted a streamlined preprocessing strategy consisting of the following steps:

1. Creation of the categorical target variable:

   To formulate the problem as binary classification, we defined a response variable "Pass/Fail," which indicates whether a student is likely to pass (assigned a value of "1") or fail (assigned a value of "0"). We grouped the final grades into two categories: stu-

dents with a final grade of C or higher were classified as "Pass," while those with grades lower than C were classified as "Fail," following conventional pass-fail boundaries[1].

2. Retention of numeric scores:

   We retained all numerical assessment scores (Online Activities, Lab Exercises, Homework items, Midterm, and Final) and omitted the final categorical grades from the feature set.

3. Feature scaling:

   Before exploratory data analysis (EDA) and model building, we applied min-max scaling to normalize all feature values to the range [0, 1]. This scaling aids EDA methods and benefits algorithms such as neural networks that are sensitive to variations in feature scales.

4. Data splitting:

   For model training and evaluation, we used a temporal splitting strategy. Data from the first three semesters (Spring 2021, Fall 2021, and Spring 2022) comprised the training and validation datasets, while data from the fourth semester (Fall 2022) was reserved as the testing set. This approach simulates real-world conditions in which models are applied to unseen data from a new semester. The validity of pooling the first three semesters into a single training-and-validation set is examined in Section 3.2.3, where a two-way mixed ANOVA tests whether mean scores differ systematically across semesters.

5. Assessment of class imbalance:

   Our dataset exhibits a moderate imbalance, as the "Pass" category is the majority and the "Fail" category is the minority. We computed the binary imbalance ratio (IR), defined as

$$\mathrm{IR} = \frac{N_1}{N_2}, \tag{1}$$

   where $N_1$ is the number of "Pass" cases and $N_2$ is the number of "Fail" cases. As shown in Table 3, the IRs for both the training and evaluation datasets are low, indicating that class imbalance is not a substantial issue (Silva & Zanchettin, 2015). We therefore did not apply any data-level, sampling-based imbalance handling methods such as SMOTE (oversampling minority groups) (Chawla et al., 2002) or Tomek Links (undersampling majority groups) (Tomek, 1976).

**Table 3.** Dataset Summary: Total counts, outcomes, and imbalance ratios (IR) for the entire dataset as well as for training, validation, and testing splits. "Fail" here follows the classification defined in Section 2.1.1: final grades below C, reflecting sub-competency performance rather than formal institutional course failure.

| Dataset | Set | Count | Pass | Fail | IR |
|---|---|---|---|---|---|
| 4 Semesters | Total | 244 | 163 | 81 | 2.01 |
| For Grid Search and Final Training | | | | | |
| First 3 Semesters | Training | 159 | 108 | 51 | 2.12 |
| | Validation | 18 | 12 | 6 | 2.00 |
| | Subtotal | 177 | 120 | 57 | 2.11 |
| 2022 Fall | Testing | 67 | 43 | 24 | 1.79 |

#### 2.1.2. Exploratory Data Analysis Procedure

To characterize relationships among assessment scores and to surface properties relevant to later modeling and explanation, we conducted three exploratory analyses. First, we computed pairwise Pearson correlations and mutual information to summarize relationships among assessments. Second, we evaluated the degree of multicollinearity

by applying two complementary diagnostics, the Variance Inflation Factor (VIF) and the Condition Index (CI), described in Section 2.1.3, since severe multicollinearity can degrade predictive accuracy and compromise feature-importance explanations. Third, we conducted a two-way mixed-design ANOVA (Hocking, 1973) with assessment as the within-subjects factor and semester as the between-subjects factor, to test for significant differences both across specific assessments and between academic terms. The ANOVA results are reported in Section 3.2.3.

#### 2.1.3. Feature Dependence Analysis Procedure

Multicollinearity arises when two or more input features are strongly linearly correlated, and it can undermine both Shapley-based and counterfactual explanation methods. In Shapley-based approaches like Expected Gradients (Erion et al., 2020), attributions are computed by marginalizing one feature at a time under an implicit separability assumption; when features are dependent, this assumption fails and attribution scores may be distorted (see Equation (13)). Counterfactual explanations seek the smallest change to a single feature that changes a prediction; under multicollinearity, the recommended intervention may target a correlated proxy rather than the true causal factor (see Equation (19)). Consequently, any feature importance ranking derived from these explanations can become unreliable. To quantify the degree of multicollinearity in our training data, we applied two complementary diagnostics: the Variance Inflation Factor (VIF), which measures how much a predictor's coefficient variance is inflated by its correlations with other variables, and the Condition Index (CI), which assesses overall collinearity within the feature matrix.

For a set of $k$ features, the VIF for predictor $X_j$ ($j = 1, \ldots, k$) is defined as

$$\mathrm{VIF}_j = \frac{1}{1 - R_j^2}, \tag{2}$$

where $R_j^2$ is the coefficient of determination from regressing $X_j$ on the other $k - 1$ predictors (Marcoulides & Raykov, 2019).

The CI diagnostic is given by

$$\mathrm{CI}_j = \sqrt{\frac{\lambda_{\max}}{\lambda_j}}, \tag{3}$$

where $\{\lambda_j\}_{j=1}^{k}$ are the eigenvalues of the predictors' correlation matrix and $\lambda_{\max} = \max_j \lambda_j$ (Callaghan, 2008).

#### 2.1.4. Statistical Testing Procedure Across Semesters and Assessments

We transformed the assessment data into a long format, in which each row corresponds to a student's score on a specific assessment. Each student's scores were arranged in 42 consecutive rows in chronological order, and the complete dataset was formed by concatenating these rows vertically. To support the statistical analysis, we added two identifier columns: "Semester", indicating the offering term, and "AssessmentName", indicating the assessment type. The rows were then sorted by "Student_ID" and "AssessmentName" to preserve temporal ordering while maintaining separate blocks for each semester.

Using this long-form dataset, we performed a two-way mixed ANOVA with "AssessmentName" as the within-subjects factor and "Semester" as the between-subjects factor. Our inference focused on three key statistics: the unadjusted $p$-value, the $p$-value corrected for repeated-measures effects (Abdi, 2010), and the partial eta-squared effect size $\eta_p^2$ (Pierce et al., 2004). These statistics allowed us to determine whether (a) different assessment types yielded statistically distinct mean scores within students, and (b) overall performance

shifted across semesters. The second test bears directly on our decision to pool data from the first three semesters into a single training-and-validation set (Section 2.1.1, preprocessing step on data splitting): a non-significant "Semester" main effect would support treating the pooled semesters as samples from distributions that do not differ systematically in mean score.

### 2.2. Formulation, Modeling, Training, and Evaluation

To identify at-risk students early, we employed a bidirectional Gated Recurrent Unit (GRU) model to predict whether a student would pass the course by the end of the semester. The model accepts input sequences of variable length. Specifically, for the first $k$ assessments of a student, the input consists of assessment scores $x_1, x_2, \ldots, x_k$, and the output is a binary label $y_k$ indicating "Fail" (0) or "Pass" (1). Formally,

$$h_k = \begin{cases} 0, & k = 0 \\ f(X_k, h_{k-1}), & \text{otherwise} \end{cases} \qquad (4)$$

$$y_k = g(h_k)$$

where $h_k$ is the hidden state at time $k$, $g(h_k)$ is the function mapping the hidden state to the prediction $y_k$, and $X_k = x_k$ with $k \leq 42$. Here, $y_k$ represents the predicted final course outcome after considering only the first $k$ assessments. The formula (4) can also be seen as a function of the prefix length $k$—the number of earliest assessments available at prediction time. At prefix length $k$ the model receives the partial sequence to predict the final course outcome of the student without access to later scores $x_{k+1}, x_{k+2}, \ldots, x_{42}$.

#### 2.2.1. Grid Search for Model Architecture and Training Configurations

We conducted a grid search on a fixed pair of training and validation (hold-out) datasets (obtained by stratified data splitting) to identify an optimal model architecture and training configuration set. Given our relatively small dataset, we constrained the search space as follows:

- Number of Hidden Layers: 1 to 3 layers.
- Number of Neurons (per Hidden layer):
  - For multi-layer networks: 8, 12, 16, and 24 neurons.
  - For single-layer networks: 56, 64, 72, 96, and 128 neurons.
- Activation Functions: ReLU and Tanh.
- Training Configurations: varying batch sizes, epochs, optimizers, multiscale learning rates, and $L_1$- and $L_2$-norm regularizers with multiple penalty factors.

For each hyperparameter combination, we conducted a single training and evaluation using a predetermined training–validation split, repeating the process five times with different random initializations. Each training instance used the full 42-element assessment sequences for all students in the training set, but the supervision signal was applied incrementally rather than only at the full-sequence level. Specifically, within each training epoch, the model was updated once at every prefix length $t \in \{1, 2, \ldots, 42\}$. For a given $t$, the input sequence was truncated to its first $t$ assessment scores before being processed by the bidirectional GRU, so that both the forward and the backward pass of the recurrence operated only on positions 1 through $t$. The model produced a prediction from the hidden state at position $t$, the binary cross-entropy loss was computed against the student's final Pass/Fail label, and a gradient step was taken. This prefix-expanded training procedure means that across an epoch, the model received 42 gradient updates per student, each targeting predictive accuracy at a specific prefix length, rather than a single gradient update

based only on the full sequence. Subsequently, we evaluated the model separately on the training and validation datasets at each prefix length $t = 1, \ldots, 42$, producing paired accuracy curves—one for training, one for validation—that reflect how well the model predicts final outcomes from progressively longer partial sequences.

Unlike conventional grid search methods that use cross-validation across multiple splits, our approach directly selects the model architecture–configuration pair that achieves the highest overall mean validation accuracy. This overall mean was computed by averaging the prediction accuracy scores obtained at each incremental input length on the fixed split. Employing cross-validation at this stage would have unnecessarily complicated subsequent analyses, as the model was later retrained on the same split and only a single model instance was preserved for subsequent prediction explanation on the test dataset. Consequently, we required that the chosen model demonstrate learning potential on the fixed split so that its prediction accuracy curve would serve as a baseline for further training attempts in the next stage.

#### 2.2.2. Final Model Training and Selection Using the Gompertz Function

Once the optimal model architecture and training configuration were identified, we retrained the model on the same fixed training–validation split. This retraining was repeated 20 times; for each run, a small-magnitude positive or negative adjustment was applied to the selected regularization penalty values while all other training configurations remained constant. The resulting pool of 20 candidate models shares a common architecture and training configuration but differs in the specific regularization strengths applied during training, providing a set of similarly-configured candidates from which a final model is chosen.

After each training run, we calculated the average prediction accuracy over the training–validation split separately for each prefix length $t = 1, 2, \ldots, 42$, producing a pair of prefix-indexed accuracy curves (one for training, one for validation) per candidate. We then filtered out candidates whose overall mean prediction accuracy on both the training and validation sets was below 80%, ensuring that the remaining candidates had demonstrated at least baseline predictive competence. The selection problem that followed was to choose, from among the surviving candidates, a single model for subsequent test-set evaluation and post-hoc explanation.

Standard scalar aggregations of the prefix-indexed accuracy curve, such as mean validation accuracy across prefixes, are ill-suited to this selection problem. Mean accuracy collapses the full trajectory into a single number and is insensitive to the shape of the curve: two candidates with identical mean accuracy may differ substantially in their usefulness for early prediction. One candidate's curve may oscillate between low and high values across prefixes, while another's may rise smoothly from a moderate initial accuracy to a high plateau; only the latter is a viable early-prediction system, yet mean accuracy cannot distinguish them. Similarly, a candidate whose accuracy rises quickly to a high asymptote is more useful for timely intervention than one whose accuracy rises slowly to the same asymptote, but this difference in growth rate is also invisible to mean accuracy. Because our application—early identification of at-risk students—depends on how the model's predictive power evolves with accumulating assessment information, the selection criterion must be sensitive to the shape of the accuracy curve, not only its average level.

To capture these trajectory-shape properties explicitly, we fitted the training accuracy curve of each surviving candidate to a parameterized Gompertz function (Waliszewski & Konarski, 2005). The Gompertz function is a smooth, monotonically non-decreasing parametric form that is widely used to describe growth trajectories in biological and learning-curve contexts. In our application, it serves as a compact summary of three properties of an

accuracy curve simultaneously: the asymptotic value to which the curve converges, the initial value of the curve at prefix length zero, and the rate at which the curve approaches its asymptote. The first corresponds to the accuracy plateau a well-trained model reaches given sufficient input; the second corresponds to the model's baseline predictive capacity with minimal information; and the third corresponds to how rapidly the model improves as additional assessment scores become available. A well-trained sequential model for early prediction should exhibit approximately monotonic improvement with accumulating information, and the Gompertz parameters encode the key features of such a trajectory. The smooth parametric fit tolerates local deviations from strict monotonicity—which are expected in practice due to finite-sample estimation noise—while rewarding the overall upward trend.

The Gompertz function is defined as

$$f(t) = a \exp\left[-b \exp(-ct)\right], \tag{5}$$

where $a > 0$ represents the curve's asymptotic maximum accuracy for large $t$, $b > 0$ (in conjunction with $a$) determines the initial value $f(0)$, and $c > 0$ governs how quickly the function approaches $a$. The parameters were estimated by non-linear least squares on the prefix-indexed accuracy values $t = 1, \ldots, 42$ for each candidate.

In selecting the final model, we prioritized candidates whose fitted Gompertz function on the training accuracy curve yielded a higher $c$-value, and among those we chose the candidate with the largest $a$-value[2]. This two-stage rule implements a preference for models that rise quickly to a high accuracy plateau, which is the behavior most relevant to early at-risk identification: a higher $c$ indicates that the model reaches useful predictive accuracy at earlier prefixes, enabling intervention earlier in the semester, while a higher $a$ indicates that the model's eventual performance is also high. The ordering of the two criteria—$c$ first, $a$ second—reflects the primacy of early-availability over late-plateau accuracy in this application; an institution with different priorities could reverse the ordering or use a weighted combination, and the same candidate pool could be re-ranked accordingly without re-running the selection procedure.

Although the procedure above is non-standard, its non-standardness reflects a property of the problem rather than a departure from methodological norms: standard scalar aggregations (e.g., mean accuracy) are not designed to represent trajectory-shape preferences, and selecting for such preferences requires a criterion that parameterizes the trajectory explicitly. The procedure is specifically designed for settings in which a single final model must be chosen for subsequent analysis and in which the model's behavior across the full range of partial inputs is itself the property of interest.

The chosen model defines the predictive component of the pipeline, but two operational decisions remain before it can be used to identify at-risk students and support explanation. First, a single prefix length must be identified as the anchor point for threshold selection and subsequent evaluation (Section 2.2.3). Second, the decision threshold that converts predicted probabilities into binary Pass/Fail classifications must be set (Section 2.2.4). We address these two decisions in turn.

#### 2.2.3. Anchor Prefix Selection

The anchor prefix is the specific $t$ at which threshold selection is performed and at which the final model's test-set performance is reported. Identifying this prefix is a different methodological decision from the Gompertz-based model selection in Section 2.2.2: the Gompertz criterion governs which candidate model to retain based on the shape of its full accuracy trajectory, whereas the anchor prefix selection described here identifies a

specific point on the already-selected model's trajectory at which downstream decisions can be anchored.

We adopted a learning-saturation criterion: the anchor prefix is defined as the smallest $t$ for which the model's training-set AUC reaches 99% of its maximum value over the full prefix range, with the 99% value itself a contextual choice that a different deployment setting could reasonably adjust. AUC is appropriate here because it is computed from prediction scores and evaluates how students are ranked by risk without requiring a fixed decision threshold, and because it is invariant to which class is designated positive—so the prefix-selection decision reflects the model's overall discriminative capability rather than sensitivity to Fail specifically, and the Fail-positive framing enters only at the threshold-selection step (Section 2.2.4). Saturation of the training-set AUC therefore indicates that the model has extracted nearly all of the ranking information the training data can provide, so that additional assessment scores beyond that point contribute little further discriminative value and the criterion identifies the earliest prefix at which learning from the observed sequence is essentially complete.

The selected prefix is used to determine the prediction threshold (Section 2.2.4), which is then fixed for the subsequent evaluations and analyses.

#### 2.2.4. Decision Threshold

Although the model encodes Pass as 1 and Fail as 0, the practical event of interest is course failure. We therefore treat Fail (at-risk) as the positive class and Pass (non-at-risk) as the negative class; under this convention, recall measures correct identification of at-risk students and specificity measures correct identification of non-at-risk students.

The decision threshold is a deployment choice rather than a model property. A common default is $\tau = 0.5$, appropriate when the two classes are roughly balanced, and the costs of the two error types are approximately equal. Neither condition is guaranteed in an intervention-oriented application, so we treat the threshold as a parameter selected against the actual costs the institution faces in deploying the model.

At-risk identification has two error types: missing a student who ultimately fails and therefore receives no intervention, and flagging a student who would have passed, generating unneeded outreach. When the cost of the first is taken to exceed the cost of the second, a natural response is to shift the threshold to increase recall. We accept this asymmetry—the consequences of missing a student headed toward course failure are real—but argue that in our deployment context it does not translate directly into a recall-maximizing threshold.

The binding constraint is not diagnostic sensitivity but instructional capacity. The course is supported by one instructor with 3 weekly office-hour hours and 2 teaching assistants each offering 3 weekly hours, totaling 9 h of scheduled support per week—sufficient for reactive support but not for proactive outreach to a large flagged set. Spreading that fixed capacity over a larger flagged set does not reach more students effectively; it only dilutes the depth of help each receives. A specificity-favoring threshold, therefore, concentrates scarce instructional resources on students for whom the model is most confident, trading a small reduction in population-level recall for preserved per-student intervention depth.

We implemented this reasoning through a specificity-targeted selection rule applied on the validation set at the anchor prefix from Section 2.2.3. Among candidate thresholds in $(0, 1)$, we retained those achieving at least 80% specificity on validation data, then selected from that subset the threshold with the highest recall for Fail. The chosen threshold was applied unchanged to all subsequent test-set evaluations and to the counterfactual explanation pipeline (Section 2.3). Gradient SHAP explains the model's output scores rather than a thresholded decision and is unaffected by this choice; DiCE counterfactuals depend on the target-class threshold and use the selected value.

The 0.80 specificity target is a contextual choice reflecting our instructional-capacity constraints; other deployment settings—larger teaching-assistant pools, dedicated early-alert staff, automated outreach systems—could reasonably apply the same procedure with a different target. What we retain is the principle that the threshold should be chosen against the institution's actual cost structure rather than defaulted to 0.5, and that the choice be transparent so the reported metrics can be interpreted in light of it.

### *2.3. Explanation*

The modeling and selection procedures in Section 2.2 produce a single trained model, an anchor prefix at which its behavior is evaluated, and a decision threshold that governs how its predictions are converted into Pass/Fail classifications. These components define the predictive system but do not reveal which assessments drive its predictions or how those predictions would change if individual assessment scores were different. To answer these questions, we generated post-hoc explanations using two complementary techniques: Gradient SHAP (Lundberg & Lee, 2017) and Diverse Counterfactual Explanations (DiCE) (Mothilal et al., 2020). Gradient SHAP was selected for its computational efficiency and smooth attributions on high-dimensional neural networks, leveraging model gradients to approximate Shapley values. DiCE generates diverse, minimally perturbed instances that flip the model's prediction, providing actionable insights into required feature changes. The two methods address complementary questions—Gradient SHAP asks how each feature contributes to the predicted probability, while DiCE asks what combinations of feature changes would flip the prediction—and their differing relationship to the decision threshold is relevant for our analysis: Gradient SHAP attributions are computed from model gradients and are independent of the threshold, whereas DiCE counterfactuals depend on the threshold and use the value selected in Section 2.2.4.

#### 2.3.1. Gradient SHAP

Gradient SHAP attributes each feature a scalar value reflecting its contribution to the model's predicted probability for a given instance, grounded in the classical Shapley value from cooperative game theory but adapted for continuous input spaces and differentiable models. The discrete Shapley value fairly allocates a model's output to individual features by averaging each feature's marginal contribution across all coalitions. To extend this idea to continuous input spaces, Integrated Gradients (IG) accumulates gradients along a straight-line path from a baseline input to the point of interest (Sundararajan et al., 2017). Expected Gradients generalizes IG by averaging over a distribution of baselines, thereby producing attributions that converge to Shapley values in expectation (Sundararajan et al., 2017); this algorithm is implemented in the SHAP library as Gradient SHAP (in short, Grad-SHAP). In what follows, we discuss the closed-form expressions for the Shapley value, IG, and Expected Gradients from the aspect of XAI, and then we derive the local- and global-importance vectors used in our analysis.

1. Shapley Value (Shapley, 1953) Let $N = \{1, \ldots, n\}$ index the full set of features, and let any subset $S \subseteq N$ denote a coalition of features. Let $\nu : 2^N \to \mathbb{R}$ be a worth function that assigns a scalar value to each coalition, with $\nu(\emptyset) = 0$ by convention. The Shapley value for feature $j$ is

$$\phi_j(\nu) = \sum_{S \subseteq N \setminus \{j\}} \frac{|S|!\,(n - |S| - 1)!}{n!} \times \left[\nu(S \cup \{j\}) - \nu(S)\right]. \tag{6}$$

where $\nu(N)$ is the worth (or gain) of the grand coalition $N$ that considers all features. One of four properties of the Shapley value is efficiency:

$$\sum_{j=1}^{n} \phi_j(\nu) = \nu(N) - \nu(\emptyset), \tag{7}$$

where $\nu(\emptyset)$ serves as the attribution baseline. In the classic SHAP, given specific instance $x = x^*$, $\nu(S)$ is evaluated as follows (Lundberg & Lee, 2017; Olsen & Jullum, 2024):

$$\nu(S) = \mathbb{E}[f(x)|x_S = x_S^*] \tag{8}$$

$$= \mathbb{E}[f(x_{\bar{S}}, x_S)|x_S = x_S^*] \tag{9}$$

where $x_S = \{x_j | j \in S \subseteq N\}$ and $x_{\bar{S}} = \{x_j | j \notin S \wedge j \in N\}$. Efficiency in this context is analogously expressed as

$$\sum_{j=1}^{n} \phi_j(x^*) = f(x^*) - \mathbb{E}[f(x)] \tag{10}$$

2. Integrated Gradients (Sundararajan et al., 2017) The idea of Integrated Gradients corresponds to the Aumann-Shapley (Aumann & Shapley, 1974) cost-sharing principle for differentiable models. For a baseline $x'$ and an input $x$, define the straight-line path $\gamma(\alpha) = x' + \alpha\,(x - x')$, with $\alpha \in [0, 1]$. The attribution for feature $j$ is

$$\phi_j^{\mathrm{IG}}(x, x') \;=\; (x_j - x_j') \int_0^1 \frac{\partial f\big(\gamma(\alpha)\big)}{\partial x_j}\, d\alpha. \tag{11}$$

If model $f$ is differentiable almost everywhere, then efficiency is expressed as

$$\sum_{j=1}^{n} \phi_j^{\mathrm{IG}}(x, x') = f(x) - f(x'). \tag{12}$$

3. Expected Gradients (Gradient SHAP) (Erion et al., 2020; Sundararajan & Najmi, 2020) Expected Gradients (EG) approximates Shapley values by averaging Integrated-Gradient samples over multiple baselines $x' \sim D$ and path points $\alpha \sim U(0, 1)$:

$$\phi_j^{\mathrm{EG}}(f, x) = \mathbb{E}_{x',\alpha}\Big[(x_j - x_j')\,\frac{\partial f\big(x' + \alpha(x - x')\big)}{\partial x_j}\Big]. \tag{13}$$

And efficiency is expressed as

$$\sum_{j=1}^{n} \phi_j^{\mathrm{EG}}(f, x) = f(x) - \mathbb{E}[f(x)]. \tag{14}$$

4. Local and Global Importances via Gradient SHAP For each instance $i$ at prefix length $t \in \{1, \ldots, 42\}$ (i.e., truncated to its first $t$ features), denoted $X_{i,t}$, the local importance of feature $j$ is the absolute Expected Gradients attribution:

$$h_{i,t}^{j} = \big|\phi_j^{\mathrm{EG}}\big(f, X_{i,t}\big)\big|, \quad j = 1, 2, \ldots, t. \tag{15}$$

Collecting these values yields the local importance vector

$$H_{i,t}^{q} = \big[h_{i,t}^{1}, \ldots, h_{i,t}^{t}\big]. \tag{16}$$

We then define the *global importance* over the dataset $D$ by averaging:

$$\begin{aligned} h^j_{D,t} &= \frac{1}{|D|}\sum_{i=1}^{|D|} h^j_{i,t}, \\ H^a_{D,t} &= \left[h^1_{D,t}, \ldots, h^t_{D,t}\right], \\ H^a_D &= \left\{ H^a_{D,t} \;\middle|\; t = 5, 6, \ldots, 42 \right\}, \end{aligned} \tag{17}$$

where $j = 1, 2, \ldots, t$ and the superscript $a$ indicates Gradient SHAP.

#### 2.3.2. Diverse Counterfactual Explanations

Given the final model $f$ and an input $x$, a counterfactual $c$ is any perturbation such that $f(c) \neq f(x)$, with $f(c)$ equal to the desired flipped outcome. DiCE (Mothilal et al., 2020) generates a diverse set of $k$ counterfactuals $C_k = \{c_1, \ldots, c_k\}$ by solving

$$C^*(x) = \arg\min_{C_k} \left[ \frac{1}{k}\sum_{z=1}^{k} \mathcal{L}\left(f(c_z), y\right) + \frac{\lambda_1}{k}\sum_{z=1}^{k} d(c_z, x) - \lambda_2 \operatorname{div}(C_k) \right]. \tag{18}$$

Here:

- $\mathcal{L}(f(c_z), y)$ is a loss function that encourages $f(c_z) = y$.
- $d(c_z, x)$ (e.g., the $L_1$ or $L_2$ norm) measures proximity to $x$, weighted by $\lambda_1$.
- $\operatorname{div}(C_k)$ quantifies pairwise distances among the $\{c_z\}$, weighted by $\lambda_2$ to promote diversity.

Objective function (18) generates multiple, minimally altered yet distinct counterfactuals that satisfy the flip constraint while remaining actionable.

Kommiya Mothilal et al. (2021) defined the necessity of a feature to obtain the current prediction as the fraction of times that changing the feature leads to a valid counterfactual example. For a single instance $i$ at prefix length $t$, $X_{i,t}$ the local importance of feature $j \in \{1, \ldots, 42\}$ evaluated on $k$ counterfactuals $C_k$ is defined as

$$h^j_{i,t} = \frac{1}{k}\sum_{z=1}^{k} \mathbb{1}\left[c^j_z \neq X^j_{i,t} \;\wedge\; f(c_z) \neq f(X_{i,t})\right]. \tag{19}$$

The local importance vector $H^b_{i,t}$, global importance vector $H^b_{D,t}$, and global importance set $H^b_D$ are defined similarly as in Equations (16) and (17), with a different superscript $b$ referring to DiCE.

### *2.4. Determine Important Features*

To identify the most influential features, we convert global importance vectors into global rank vectors. For each explanation method $m \in \{a, b\}$ (where $a$ denotes Gradient SHAP and $b$ denotes DiCE) and each prefix length $t \in \{5, \ldots, 42\}$, let

$$H^m_{D_1,t} = \left[h^1_{D_1,t}, \ldots, h^t_{D_1,t}\right] \tag{20}$$

be the global importance vector on the test set $\mathcal{D}_1$. We define the corresponding global feature rank vector

$$R^m_{D_1,t} = \operatorname{rank}\left(H^m_{D_1,t}\right) = \left[r^1_{D_1,t}, \ldots, r^t_{D_1,t}\right], \tag{21}$$

where $r^j_{D_1,t}$ is the rank of feature $j$ among the first $t$ features (a lower value indicates higher importance). Collecting these for $t = 5, \ldots, 42$ gives

$$R^m_{D_1} = \{ R^m_{D_1,t} \mid t = 5, \ldots, 42 \}, \tag{22}$$

a set of 38 rank vectors. For the case of local importance vector $H^m_{i,t}$, we can define local feature rank vector $R^m_{i,t}$ similar to (21) and local feature rank vector set $R^m_i$ similar to (22).

Next, for each feature $j \in \{1, 2, \ldots, 41\}$, we gather its ranks across all $R^m_{D_1,t}$ with $t \geq j$. To quantify both central tendency and stability, we compute the median rank and rank standard deviation for each $j \leq 41$:

$$M^m_j = \text{median}\{ r^j_{D_1,t} \mid t \geq j \}, \tag{23}$$

$$S^m_j = \sqrt{\frac{1}{43 - j} \sum_{t=j}^{42} (r^j_{D_1,t} - M^m_j)^2}, \tag{24}$$

omitting the case of $j = 42$ for computing standard deviation because only one rank exists.

An important feature should have both a low median rank (frequently top-ranked) and a low standard deviation (stable importance across different prefix situations). However, early features appear in more frequently—potentially biasing their statistics; their broader sampling nonetheless yields more precise estimates. Because our goal is early identification of at-risk students, we downweight any feature $j$ whose $M^m_j$ or $S^m_j$ exceeds that of the midterm feature (at prefix length $j = 21$). To ensure fairness, we compute $M^m_j$ and $S^m_j$ for features available at or before the midterm only over horizons $t \geq 21$ (i.e., at prefix length $t = 21, \ldots, 42$), so that each considered feature is evaluated over identical sequence lengths, with all considered features present in those sequences. In this case, we require that an important feature $j$ must satisfy the condition $M^m_j + S^m_j < M^m_{21}$.

### *2.5. Explanation Agreement*

To assess consistency, we compare feature rankings from Gradient SHAP (method $a$) and DiCE (method $b$), both within each method over time (*intra-method* agreement) and between methods at the same prefix length (*inter-method* agreement). We employ three quantitative metrics originally introduced by Krishna et al. (2025): feature agreement (FA) and rank agreement (RA), both bounded in $[0, 1]$, and rank correlation (RC), implemented via Kendall's $\tau$ and ranging in $[-1, 1]$.

Notation and Setup

Let

$$S = (\alpha_1, \alpha_2, \ldots, \alpha_{42})$$

be the sequence of features in order of first availability. For each method $c \in \{a, b\}$ and prefix length $t \in \{1, \ldots, 42\}$, define the rank vector for data instance $i$ in test set $D_1$

$$R^c_{i,t} = [r^1_{i,t}, \ldots, r^t_{i,t}],$$

where $r^j_{i,t}$ is the rank of $\alpha_j$ (a lower value means that a feature is more important). Additionally, when we compare rank vectors or rankings at two prefix lengths $t$ and $\tau$, we by default truncate both vectors to their first

$$k = \min(t, \tau)$$

elements, unless a common prefix length is explicitly specified for the comparison.

The unordered top-$k$ feature set is

$$\mathrm{TF}(R,k) = \{\alpha_j \in S \mid j \in \{1,\ldots,|R|\} \wedge r^j \in R \wedge r^j \leq k\}.$$

The rank-lookup operator is

$$\mathrm{Rank}(R,\alpha_j) = r^j.$$

To obtain an ordered top-$k$ list, we define

$$\mathrm{TR}(R,k) = (\beta_1,\ldots,\beta_k),$$

where $\{\beta_1,\ldots,\beta_k\} = \mathrm{TF}(R,k)$ and

$$\mathrm{Rank}(R,\beta_1) \leq \mathrm{Rank}(R,\beta_2) \leq \cdots \leq \mathrm{Rank}(R,\beta_k).$$

1. Feature Agreement (FA) Feature agreement measures the overlap in the top-$k$ features (unordered):

$$\mathrm{FA}\big(R^c_{i,t},\, R^g_{i,\tau}\big) = \frac{\big|\mathrm{TF}(R^c_{i,t},k) \cap \mathrm{TF}(R^g_{i,\tau},k)\big|}{k}. \tag{25}$$

2. Rank Agreement (RA) Rank agreement considers how many shared top-$k$ features share the same ranks:

$$\mathrm{RA}(R^c_{i,t},R^g_{i,\tau}) = \frac{Q\big(R^c_{i,t},\, R^g_{i,\tau},\, k\big)}{k}, \tag{26}$$

where the numerator

$$Q\big(R^c_{i,t},\, R^g_{i,\tau},\, k\big) = \Big|\big\{\alpha \in (\mathrm{TF}(R^c_{i,t},k) \cap \mathrm{TF}(R^g_{i,\tau},k)) \;\big|\; \mathrm{Rank}(R^c_{i,t},\alpha) = \mathrm{Rank}(R^g_{i,\tau},\alpha)\big\}\Big|. \tag{27}$$

The function $Q\big(R^c_{i,t},\, R^g_{i,\tau},\, k\big)$ denotes the cardinality (or size) of the set of features that (i) appear among the top-$k$ in both local rack vectors $R^c_{i,t}$ and $R^g_{i,\tau}$, and (ii) occupy identical rank positions in the corresponding two rankings. In other words, $Q$ counts how many of the shared top-$k$ features also match exactly in their ordinal rankings.

3. Rank Correlation (RC) Rank correlation uses Kendall's $\tau$ (Kendall, 1938) on ordered top-$k$ sequences. We adapt the definition introduced by Tiukhova et al. (2024), which is originally presented in Krishna et al. (2025).
Let

$$A = \mathrm{TR}(R^c_{i,t},k), \quad B = \mathrm{TR}(R^g_{i,\tau},k). \tag{28}$$

Define

- $P$ = number of concordant pairs between $A$ and $B$,
- $Q$ = number of discordant pairs,
- $T$ = number of ties only in $A$,
- $U$ = number of ties only in $B$.

Then

$$\mathrm{RC}(R^c_{i,t},R^g_{i,\tau}) = \frac{P-Q}{\sqrt{(P+Q+T)\,(P+Q+U)}}. \tag{29}$$

## 3. Results

### *3.1. Experiment Setup*

All experiments were implemented in Python, using version 3.9 for post-hoc explanation and 3.10 for all other stages. Data preprocessing and exploratory data analysis were performed with scikit-learn 1.6.1, statsmodels 0.14.4, and pingouin 0.5.5, and our GRU models were developed with PyTorch 2.6. Hyperparameter optimization was managed via Weights & Biases (wandb 0.19.8) to conduct our grid searches. For explanation, we employed the SHAP library's GradientExplainer[3] without modification.

DiCE 0.11 was written against specific versions of NumPy and Pandas that predate the current releases of those libraries[4], and running it against newer versions can produce import errors or unexpected behavior. To restore the library's documented behavior, we pinned NumPy and Pandas to the versions specified in the DiCE repository's requirements file; no modifications were made to the DiCE algorithm itself. The exact pinned versions are recorded in the "requirements.txt" file in our code repository, which allows readers to reconstruct the Python environment used in this study.

To support methodological reproducibility—the application of our procedures to other datasets—our code repository includes the full training and selection pipeline, the explanation-generation scripts for both Gradient SHAP and DiCE, and the post-hoc analysis tools used to produce the figures in this paper. For the Gompertz-based model selection described in Section 2.2.2, the repository provides the global random seed for deterministic training, the sweep configuration for varying regularization penalties across retraining runs, and the Gompertz fitting routine with its initial parameter values and convergence settings.

The source code and environment specification described above are provided in the Supplementary Materials.

### *3.2. Exploratory Data Analysis*

#### 3.2.1. Correlation and Mutual Information

We computed Pearson correlation and mutual information for all pairs of assessment features. To facilitate comparison, we organized the features into three categories: Activity & Exam, Homework & Exam, and Lab Exercise & Exam. Pearson correlation captures linear association on a scale from $-1$ to $+1$. Mutual information, in contrast, quantifies the reduction in uncertainty (in nats or bits) one variable provides about another through entropy, thereby detecting both linear and nonlinear relationships.

The top row of Figure 1 shows that homework scores are more tightly interrelated than both activity and lab exercise scores. Pre-midterm homework correlates more strongly with the midterm exam, and all homework items exhibit stronger associations with the final exam. Lab exercises display the weakest pairwise associations and the lowest correlations with both the midterm and the final, indicating a comparatively minor role.

The bottom row is consistent with these findings and adds detail: activity and lab exercise features remain predominantly linear in their relationships (high Pearson correlation but low mutual information), whereas homework features also demonstrate relatively high nonlinear dependence. Homework assessments additionally have higher entropy, reflecting greater variability. Together, these results highlight the central importance of homework assessments, which is consistent with Homework's grading weight in Table 2, and confirm the promise or predictive potential of Homework assessments as early indicators for identifying students at risk of course failure.

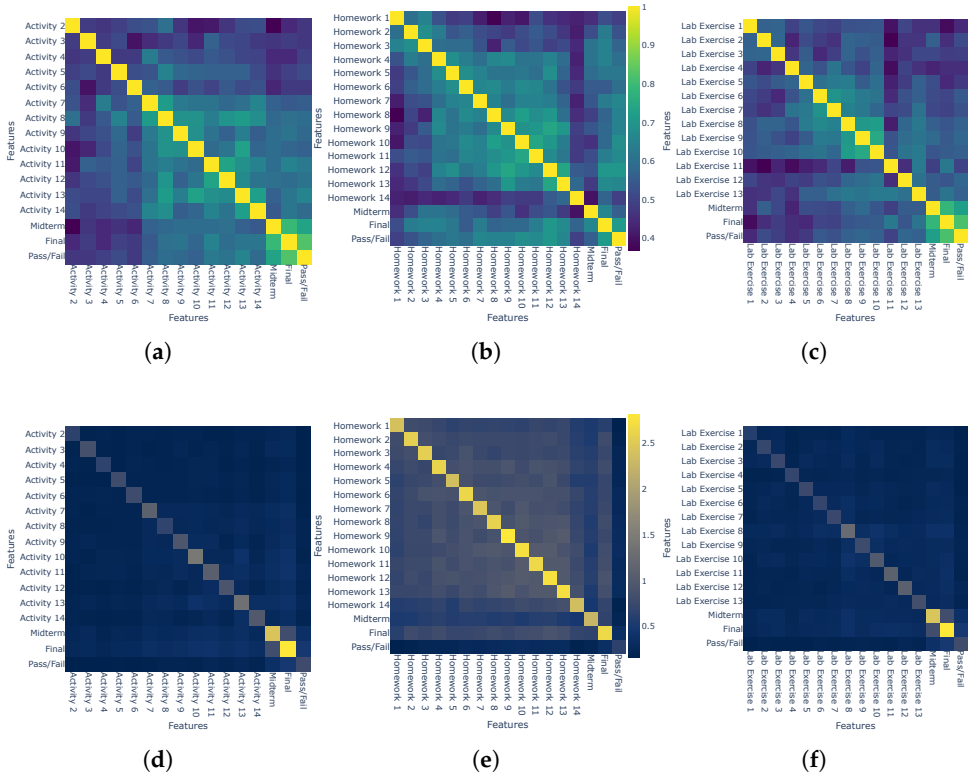
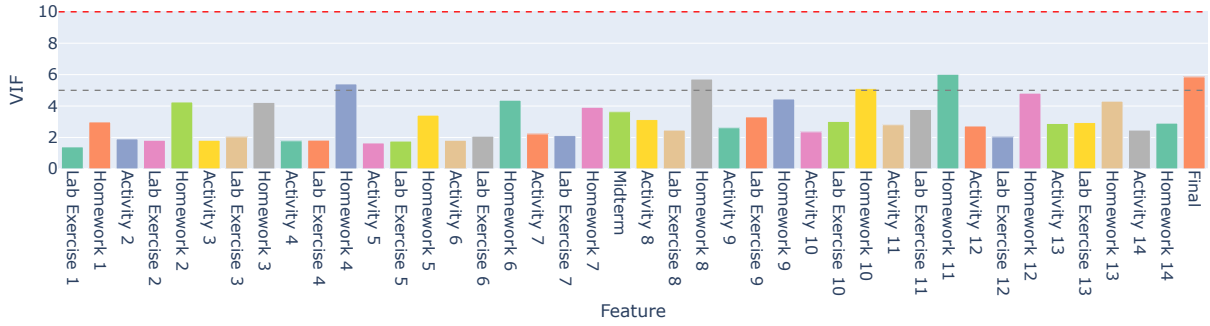


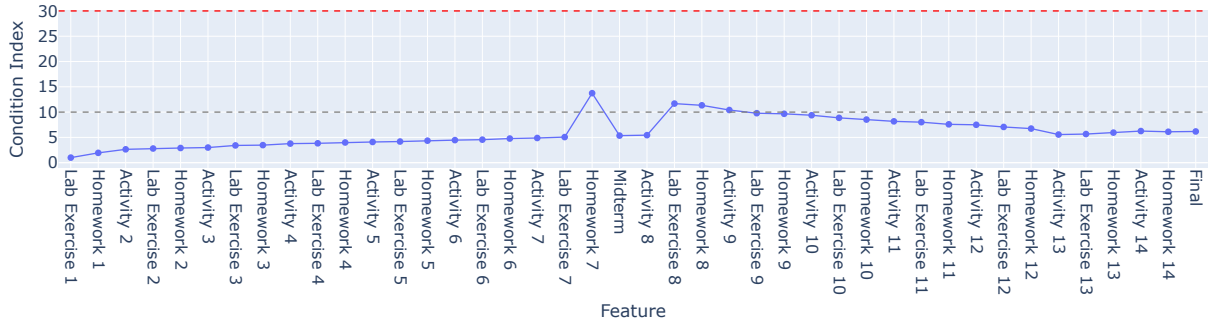

**Figure 1.** Feature correlation (top row) and mutual information (bottom row) between each assessment group and the midterm and final exams. (**a**–**c**) Correlation for activity, homework, and lab exercise, respectively. In these heatmaps, lighter colors (yellow) indicate higher correlation values, darker colors (dark blue) indicate lower correlation values, and green indicates intermediate values within the range [0.35, 0.85]. (**d**–**f**) The corresponding mutual information heatmaps, where lighter colors indicate higher mutual information and darker colors indicate lower values.

### 3.2.2. Feature Dependence Analysis

As introduced in Section 2.1.3, both Variance Inflation Factor (VIF) and Condition Index (CI) can quantify multicollinearity among features. When two or more features are strongly linearly correlated, it can compromise both Shapley-based and counterfactual explanation methods. Figure 2a displays all VIF values: most predictors lie below the common threshold of 5, and only five features fall between 5 and 6. This indicates a moderate level of collinearity for those predictors, while overall VIFs remain within acceptable limits (Marcoulides & Raykov, 2019). Figure 2b plots the CI values in the original assessment order. Here, most features exhibit CI below 10, while five lie between 10 and 15—a range generally interpreted as indicating moderate collinearity (Callaghan, 2008). Together, these diagnostics suggest that, although a handful of predictors show moderate interdependence, the overall level of multicollinearity in our training data is unlikely to compromise model stability or interpretability. These thresholds, however, address whether feature correlations destabilize OLS coefficient estimates. That concern differs from whether such correlations distort SHAP attributions or DiCE counterfactual targets, which we discuss in Section 4. Whether SHAP and DiCE remain well-behaved at the VIF and CI levels we report is not directly settled by these diagnostics, since they were calibrated against regression models rather than against XAI methods applied to neural networks.




(**a**)




(**b**)

**Figure 2.** Multicollinearity analysis. (**a**) Variance inflation factors (VIF) for each predictor. (**b**) Condition indices (CI) assessing collinearity severity.

#### 3.2.3. Statistical Significance Across Semesters and Assessments

Following the statistical testing procedure described in Section 2.1.4, the two-way mixed ANOVA results are summarized in Table 4. The main effect of "Semester" does not reach statistical significance ($p = 0.160$, $\eta_p^2 = 0.021$), indicating that mean scores did not differ systematically among the four academic terms. By contrast, "AssessmentName" exerts a highly significant effect, with $p_{GG} < 5 \times 10^{-183}$ under the Greenhouse–Geisser correction for non-sphericity.

**Table 4.** Two-way mixed ANOVA table: Sum of squares (SS), degrees of freedom (DF1, DF2), mean square (MS), F-statistic (F), uncorrected $p$-value (p-unc), Greenhouse–Geisser corrected $p$-value (p-GG-corr), partial eta-squared ($\eta_p^2$), Greenhouse–Geisser epsilon ($\varepsilon$), and sphericity assumption.

| Source | SS | DF1 | DF2 | MS | F | p-unc | p-GG-corr | $\eta_p^2$ | $\varepsilon$ | Sphericity |
|---|---|---|---|---|---|---|---|---|---|---|
| Semester | 6.5474 | 3 | 240 | 2.1825 | 1.7388 | 0.1597 | — | 0.0213 | — | — |
| AssessmentName | 158.9293 | 41 | 9840 | 3.8763 | 59.3068 | <0.0001 | $4.98 \times 10^{-183}$ | 0.1981 | 0.4577 | False |
| Interaction | 69.6837 | 123 | 9840 | 0.5665 | 8.6678 | <0.0001 | — | 0.0978 | — | — |

The non-significance of the "Semester" main effect directly justifies the pooling decision in the data-splitting step of Section 2.1: it establishes that the student populations across the first three semesters, used jointly as the training pool, are not statistically distinguishable in terms of assessment performance. This is a precondition for treating them as a unified training distribution rather than three heterogeneous populations.

### *3.3. Evaluation of Final Model*

This section addresses RQ2—the identification of assessments consistently recognized as most influential by both explanation methods. We first establish that the final model achieves accuracy sufficient to support trustworthy explanations, then use two complementary rank-based analyses to identify the influential assessments themselves.

The final GRU model comprised a single hidden layer with 56 ReLU-activated units. We evaluated its accuracy separately on the training, validation, and test sets by computing prediction accuracy at each prefix length $x = 1, 2, \ldots, 42$, generating three distinct curves. Figure 3 plots accuracy versus prefix length (i.e., number of observed features) for the training, validation, and testing datasets. The alignment of the training and test curves demonstrates that the model generalizes well to unseen semester data. The validation curve remains below the training curve, indicating that the model has not fully captured the joint distribution of the first three semesters (the training set). This modest discrepancy is expected, given the limited sample size and the intrinsic difficulty of modeling diverse assessment-score trajectories of different students. Nonetheless, it suggests that any overfitting is minimal.

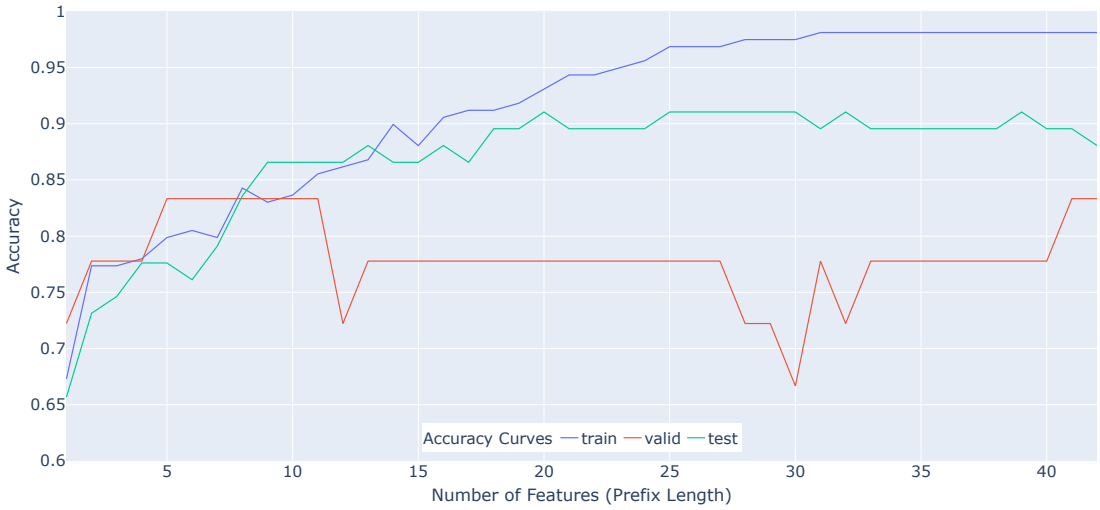



**Figure 3.** Accuracy curves evaluated on training, validation, and testing datasets.

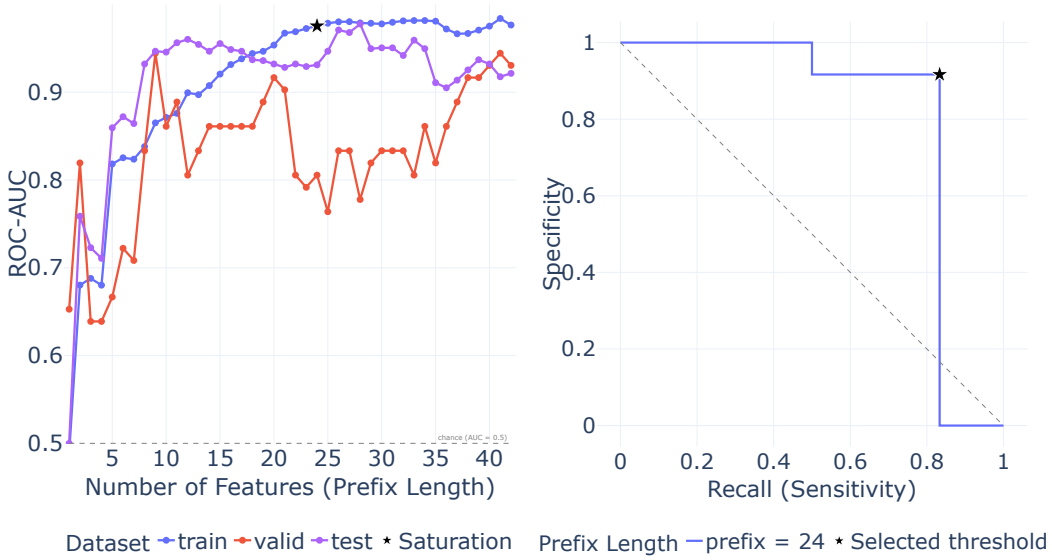
#### 3.3.1. Anchor Prefix Selection and Decision Threshold

The downstream results in Sections 3.3.2 and 3.3.3 are evaluated at a fixed decision threshold $\tau$ applied across all prefix lengths. Selecting $\tau$ requires first identifying a single prefix at which to examine the validation-set trade-off between specificity and recall—the anchor prefix defined in Section 2.2.3. This two-step structure separates the question of *where* on the semester to calibrate from the question of *how* to convert the model's continuous scores into a binary decision there.

Applying the training-AUC saturation criterion from Section 2.2.3, we selected the anchor prefix as the smallest $x$ at which training-set ROC-AUC reaches 99% of its maximum. Figure 4a plots the AUC trajectory across prefixes for the training, validation, and test sets; the criterion yielded $x = 24$, marked by a star on the training curve.

With the anchor prefix fixed at $x = 24$, we next selected the decision threshold on the validation set. Following the specificity-targeted rule of Section 2.2.4, we restricted attention to thresholds achieving validation-set specificity of at least 0.80 and, within that feasible set, chose the threshold maximizing recall for Fail. Figure 4b shows this trade-off at the anchor prefix; the selected threshold was $\tau = 0.793$, with realized validation specificity of 0.833.

At the operating point ($x = 24$, $\tau = 0.793$), the final model attained test-set recall of 0.7917, specificity of 0.9767, $F_1$-score of 0.8636, and accuracy of 0.9104. Figure 5 plots each of these four metrics across all prefix lengths so that the anchor-prefix values can be seen in context.




(**a**) ROC-AUC by prefix length. (**b**) Threshold selection at $x = 24$.

**Figure 4.** Operating-point selection for the final model. (**a**) ROC-AUC as a function of prefix length (denoted by $x$) for the training, validation, and test sets; the star marks the training-curve saturation point at $x = 24$ (99% of peak), defining the anchor prefix. (**b**) Specificity vs. recall on the validation set at $x = 24$; the star marks the threshold chosen by the specificity-floor rule (max recall subject to specificity $\geq$0.80), yielding $\tau = 0.793$. Recall treats at-risk students (Fail) as the positive class.

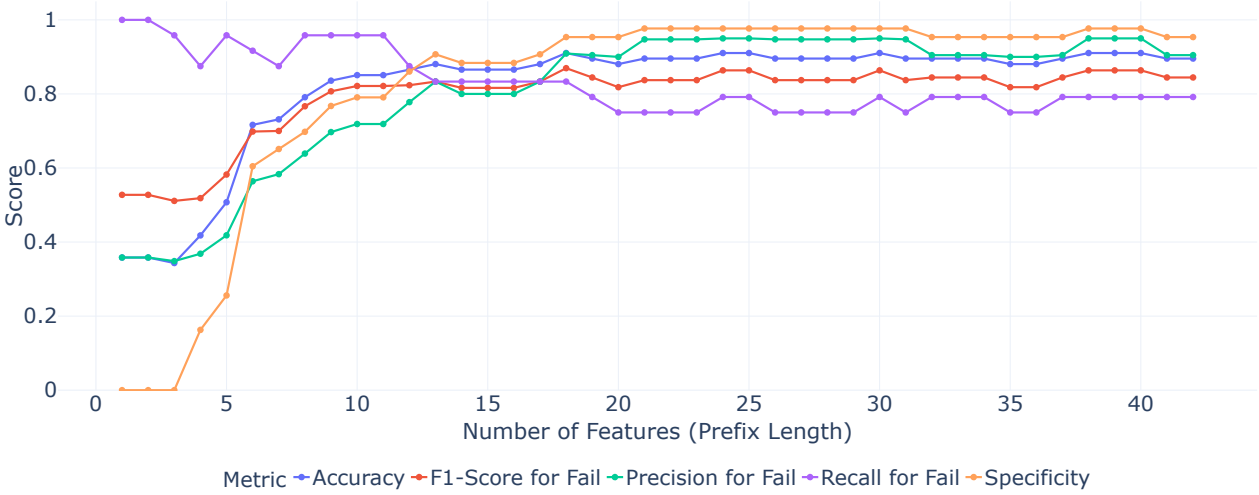


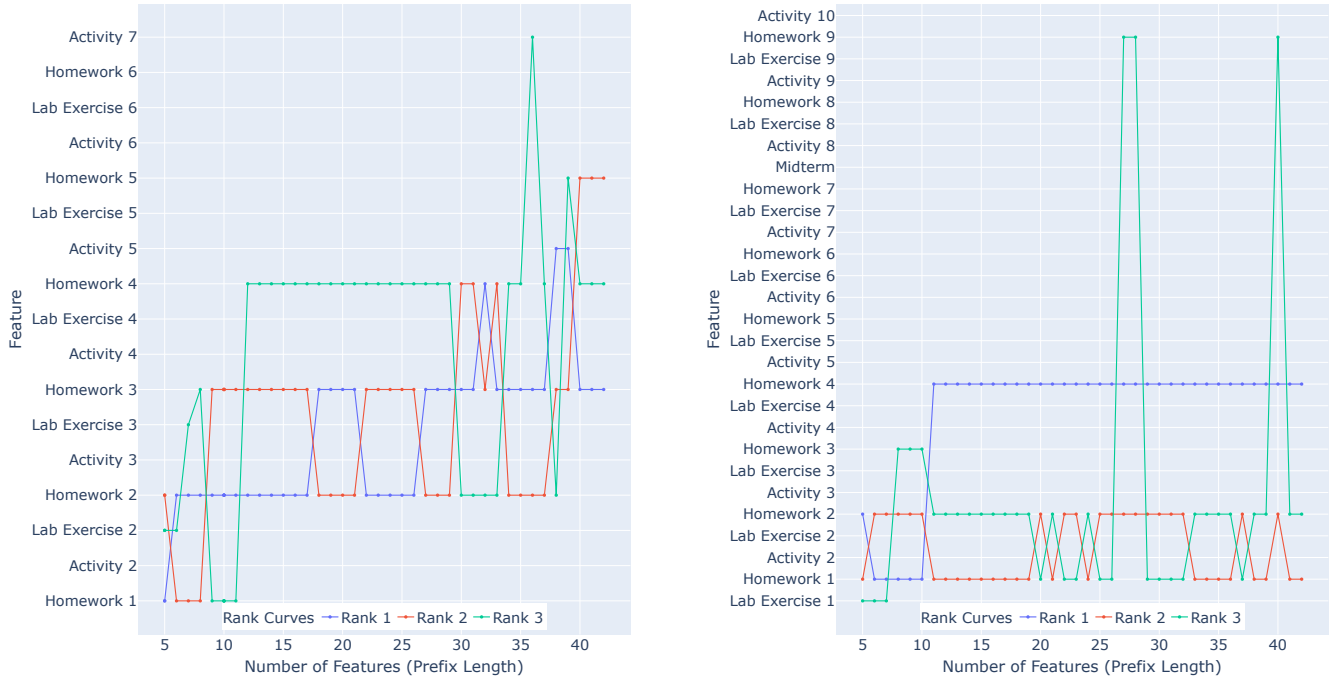

**Figure 5.** Classification metrics on the test set as a function of prefix length, evaluated at the selected decision threshold $\tau = 0.793$. Precision, recall, and $F_1$-score treat Fail (at-risk) as the positive class; specificity is the recall of the Pass class; accuracy is class-symmetric and is reported without a positive-class designation.

The test-set specificity of 0.9767 exceeds both the 0.80 validation target and the validation-set realized 0.833, confirming that the specificity-floor rule generalizes beyond the selection set. The complementary false-positive rate of approximately 0.023 corresponds to a single false positive among the 43 actual-Pass students in the test set, while the recall of 0.7917 indicates that 5 of the 24 actual-Fail students—approximately one in five—were not flagged. This asymmetry is the intended output of the specificity-floor rule, which prioritizes clearance of non-at-risk students. The nine weekly hours of instructional support described in Section 2.2.4 comfortably absorb the 20 flagged students in this cohort (19 true positives plus 1 false positive), satisfying the capacity constraint that originally motivated the rule with a wide margin.

The metrics reported at this operating point define the primary classification summary for the intervention scenario studied here. The cross-prefix analyses in the remainder of this subsection serve as sensitivity diagnostics showing how the model's behavior evolves across the full semester; they should be interpreted with respect to the anchor-prefix operating point rather than as alternative deployment configurations.

#### 3.3.2. Top-Rank Curves

As a first view of the influential assessments, we plot three separate "rank-$k$" curves in a two-dimensional space where the x-axis denotes the number of features (i.e., prefix length) and the y-axis indicates the chronological assessment category, tracking how the top three features (as determined on the test set) shift when more assessments become available. Figure 6 shows these curves based on Gradient SHAP explanations and on DiCE explanations. In Figure 6a (Gradient SHAP), all three curves lie below the midterm exam label (just above *Activity 7* on the y-axis). Before $x = 21$ (the midterm cutoff), the curves are relatively stable; after $x = 21$, they exhibit increased fluctuation as later assessments become available. Although additional post-midterm scores introduce ambiguity into the ranking, none of the top-3 curves crosses above the midterm mark. It indicates that the three most influential features remain among the pre-midterm assessments.




(**a**) (**b**)

**Figure 6.** Rank-$k$ curves for $k = 1, 2, 3$. (**a**) Gradient SHAP. (**b**) DiCE.

By contrast, in Figure 6b (DiCE), the rank-1 curve stabilizes after $x = 11$, while the rank-2 and rank-3 curves oscillate between early homework (*Homework* 1 or *Homework* 2) and later assessments (e.g., *Homework* 9, available from $x = 27$). These jumps imply that DiCE attributes similar importance to those assessments when they first appear, though this does not diminish their individual relevance.

Together, these analyses from both Gradient SHAP and DiCE confirm that the three most critical assessments occur early in the semester and well before the midterm. It supports the feasibility of early prediction of student course success or failure.

#### 3.3.3. Median Rank and Rank Standard Deviation

The rank-$k$ curves give a prefix-by-prefix view; to refine the identification into a single, aggregated answer, we examined each feature's median rank and rank standard deviation across prefix lengths, as introduced in Section 2.4. Figure 7a plots these two statistics for every feature, with ranks derived from the absolute values of Gradient SHAP attributions. A feature is possibly important when it possesses a low median rank value in the importance ranking with a small rank standard deviation. The former characteristic indicates frequent appearance near the top of the importance ordering, and the latter characteristic reflects stable importance across various numbers of observable features (or prefix lengths). Because early assessments appear in more rank vectors, their estimates are both more precise and potentially overestimated. Using *Midterm* as a reference, we observe in Figure 7a that nearly all homework assessments fall below and to the left of the midterm marker, consistent with greater and more stable importance.

The experiment was repeated for the analysis under a fair comparison, where only assessments available on or before the midterm are evaluated, and all ranks are recomputed from a common set of explanations in which every such feature should be present. Even after eliminating the sampling-frequency advantage, *Homework* 1–6 still occupy the region of lowest median rank and smallest variability (Figure 7b).

Taken together, Figure 7a,b demonstrate that *Homework* 1–6 are consistently the most influential assessments in predicting student outcomes.

Figure 7c presents the DiCE-based scatter plot analogous to Figure 7a. Here, feature *Midterm* achieves a lower (better) median rank and a smaller rank standard deviation than in the Gradient SHAP results. Moreover, every feature whose median rank and standard deviation both fall below those of the midterm is a homework assessment, which indicates that these consistently outrank the midterm in importance.

Figure 7d repeats this analysis but restricts consideration to assessments available at or before the midterm (prefix lengths $t \leq 21$). Under this constraint, the midterm's rank standard deviation drops to zero. This confirms that at least one assessment following Midterm exceeds Midterm in importance, while the median rank value of Midterm remains higher than several homework assessments. As in Figure 7b, all homework assessments exhibit lower median ranks than the midterm and minimal variability.

The dominance of homework assessments in both methods' rankings warrants closer inspection, since homework carries the largest category weight (40%) in the course's final grade and might therefore be expected to dominate feature importance mechanically. The course's grading rubric, however, assigns equal weight to every assessment within a category: all 14 homework items contribute equally to the final grade, as do all online activities and all lab exercises. Pure recovery of this structure would produce roughly equal importance within each category, but neither method does. Both Gradient SHAP and DiCE rank earlier homework items generally above later ones, so early homeworks receive more effective weight in driving the model's predictions than their nominally equal grading weight would imply, and the two methods disagree on the precise ordering. The same separation appears across the two 10%-weight categories: DiCE consistently ranks Activities above Lab Exercises (Figure 7d), with individual within-category deviations—Activity 2 sitting among Lab items, Lab Exercise 1 sitting near Activity items—confirming that the rankings reflect item-level predictive signal rather than category-level grading weight. That this within-category differentiation emerges under mechanisms as different as gradient-based attribution and counterfactual perturbation argues against a shared structural artifact.

Taken together, Figure 7 identifies *Homework* 1–4 and *Homework* 6 as the most influential early assessments across both explanation methods. These five assessments emerge as

the key predictors, underscoring the feasibility of reliable early prediction of student course outcomes.

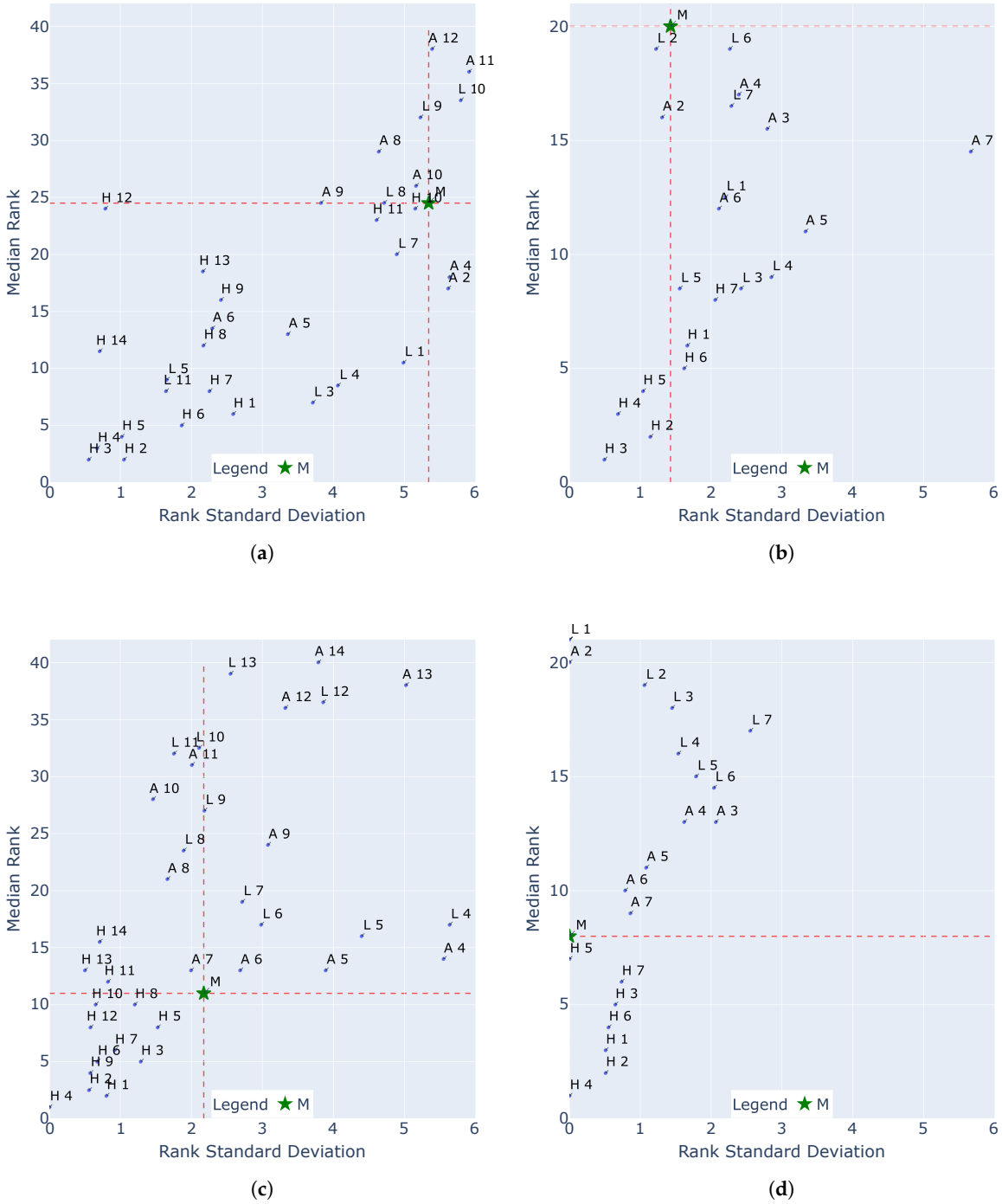



**Figure 7.** Median Rank vs. Rank Standard Deviation. (**a**) Gradient SHAP, all assessments ($t = 5, \ldots, 42$). (**b**) Gradient SHAP, pre-midterm assessments ($t = 21, \ldots, 42$). (**c**) DiCE, all assessments ($t = 5, \ldots, 42$). (**d**) DiCE, pre-midterm assessments ($t = 21, \ldots, 42$). Feature categories: A, Activity; H, Homework; L, Lab Exercise; M, Midterm.

### 3.3.4. RQ2 Summary

The intersected shortlist is robust to the choice of explanation method precisely because it is identified independently by an attribution-based method (Gradient SHAP) and a counterfactual-based method (DiCE), and the agreement of two methods grounded in dif-

ferent mechanisms provides stronger evidence for the predictive role of these assessments than either method alone could provide.

### *3.4. Measuring Explanation Agreement*

This section addresses RQ1—the evolution of explanation agreement as additional assessment scores become available during the semester. RQ1 has two complementary facets, each addressed in a separate subsection below. Intra-method agreement (Section 3.4.1) measures whether each method's rankings stabilize over time within the method itself; inter-method agreement (Section 3.4.2) measures whether the two methods agree with each other at any given prefix length. We quantify both by computing three metrics—Feature Agreement (FA), Rank Agreement (RA), and Rank Correlation (RC)—for the Gradient SHAP (method $a$) and DiCE (method $b$) explanations (see Section 2.5). Each metric captures a distinct aspect of concordance between two feature-importance rankings aggregated globally across test instances.

#### 3.4.1. Intra-Method Agreement

Intra-method agreement addresses the first facet of **RQ1**: how reliably each method's rankings at early prefix lengths survive the arrival of later evidence. Concretely, for each instance $i$ from test set $D_1$ and prefix length $t \in \{1, \ldots, 42\}$, we compare the partial rank vector $R^c_{i,t}$ with the "final" rank vector $R^c_{i,42}$. We then average over all $|\mathcal{D}_1|$ test instances to obtain

$$\overline{\mathrm{FA}}_t = \frac{1}{|\mathcal{D}_1|} \sum_{i=1}^{|\mathcal{D}_1|} \mathrm{FA}\big(R^c_{i,t},\ R^c_{i,42}\big), \tag{30}$$

with analogous definitions for $\overline{\mathrm{RA}}_t$ and $\overline{\mathrm{RC}}_t$, where $c \in \{a, b\}$. In what follows, we use "FA curve," "RA curve," and "RC curve" for brevity.

In Figure 8a,b, FA for both methods rises quickly toward 1, indicating that early partial rankings increasingly resemble the ultimate full-prefix ordering. This behavior reflects the well-trained GRU's inference capacity to capture stable feature patterns as more information is provided. Gradient SHAP's FA curve achieves above 0.5 at prefix length $x = 20$, which mathematically means

$$1/|D_1| \sum_{i \in D_1} |TF(R^a_{i,20}, 20) \cap TF(R^a_{i,42}, 20)| \geq 10.$$

It indicates that generally, with the benefit of hindsight, one half of the first 20 assessments in chronological order also appear in a student's final top-20 assessment ranking (see Figure 8a). Likewise, DiCE's FA curve achieves approximately 0.5 at prefix length $x = 20$, and we can give a similar interpretation.

Gradient SHAP's RC begins near 0.2, which is substantially below DiCE's, and then climbs to a plateau around 0.4 from $x = 15$ onward. At the same time, it shows a narrow, decreasing standard-deviation band, indicating a growing but moderate pairwise concordance (Figure 8a). DiCE's RC begins around 0.55 and oscillates mildly above 0.6, with a broader but slowly contracting band, implying higher average pairwise concordance but greater variability (Figure 8b).

DiCE's RA initiates at approximately 0.38 and fluctuates modestly around this level until prefix length $x = 20$ (Figure 8b). Thus, on average, about seven of the first 20 assessments occupy the same positions in the final full-prefix ranking. Across the entire prefix range, DiCE's RA consistently exceeds that of Gradient SHAP, indicating that DiCE better preserves the exact ranking positions of assessments within the top-$k$ set. However, its wider standard deviation band reveals greater inter-instance variability, implying that the subset of early top-$k$ assessments is less uniformly stable across students.

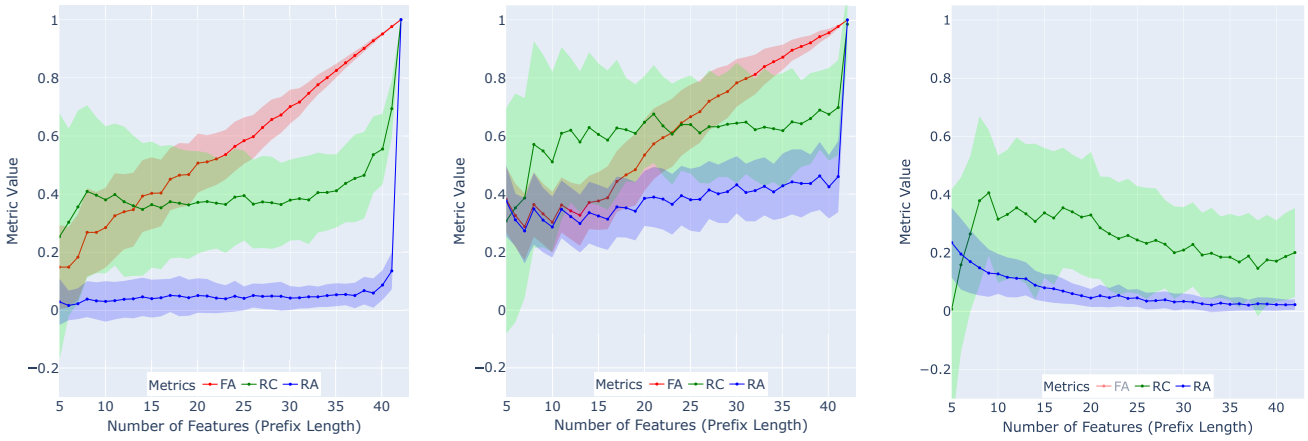



**Figure 8.** Feature agreement (FA = $\overline{\mathrm{FA}}_t$), rank agreement (RA = $\overline{\mathrm{RA}}_t$), and rank correlation (RC = $\overline{\mathrm{RC}}_t$) across prefix lengths. Shaded regions denote $\pm$ one standard deviation around each mean trajectory. (**a**) Gradient SHAP. (**b**) DiCE. (**c**) Inter-method comparison.

To further validate our earlier findings, we compute each agreement metric at a set of fixed prefix lengths $k \in \{5, 10, 15, 20, 25, 30\}$. For any two original prefix lengths $t$ and $\tau$, we define the effective comparison length

$$m = \min(k, t, \tau),$$

and calculate $\overline{\mathrm{FA}}_t$, $\overline{\mathrm{RC}}_t$, and $\overline{\mathrm{RA}}_t$ over the first $m$ features. Like before, we compare the partial rank vector $R^c_{i,t}$ with the "final" rank vector $R^c_{i,42}$, and thus $\tau = 42$ (see Equation (30)). This procedure allows us to assess ranking stability and consistency in the top-$m$ membership at regular checkpoints. Figure 9 shows the resultant curves, where we can observe a clear difference between the performance of the two methods (without standard-deviation shading for clarity).

Under Gradient SHAP, the curves FA_5, FA_10, FA_15, and FA_20 remain largely indistinguishable until after Midterm (prefix length $x = 21$), implying that reliable discrimination among feature subsets only emerges in later stages (Figure 9a). In other words, Gradient SHAP exhibits limited confidence in ranking the assessments beyond the first five during the early course phase, even though assessment scores become available incrementally.

By contrast, DiCE separates the FA_$k$ curves much earlier. The FA_5 trajectory rises to 1 rapidly, indicating that DiCE identifies the eventual top-five assessments with a stronger certainty from the outset (Figure 9d). Although decisiveness diminishes as the fixed prefix length $k$ increases, DiCE still outperforms Gradient SHAP: the FA_15 and FA_20 curves branch away from the common trajectory well before their respective $k$-values are reached, whereas the FA_25 and FA_30 curves diverge later.

A method with weak decisiveness would cause each FA_$k$ curve to peel off precisely at prefix length $x = k$; a method with complete decisiveness would keep all curves nearly coincident. DiCE's behaviour lies between these two extremes, demonstrating moderate stability in retaining its core feature set across prefix lengths, which is generally superior to Gradient SHAP.

The difference between Figure 9b,e is consistent with the gap in RC curves observed in Figure 8a,b. Across all prefix lengths, DiCE's RC_$k$ curves remain consistently higher than those of Gradient SHAP, indicating a more stable pairwise feature ordering.

Focusing on Figure 9c, we observe that all $\mathrm{RA}_k$ curves remain low and nearly flat across prefix lengths, except for a rise at $x = 41$ and $x = 42$ (and $x = 42$ is a redundant case,

as it is a self-comparison). This pattern indicates weak consistency in Gradient SHAP's exact top-$k$ rankings for every value of $k$. In contrast, Figure 9f reveals that DiCE displays graded stability: the RA_5 curve stays highest throughout, showing that DiCE reliably preserves both membership and ordering of the top-5 assessments even as additional scores become available, while the RA_$k$ curves for larger $k$ values exhibit progressively lower agreement.

In summary, the curves reveal that both methods converge in their overall feature importance profiles (FA) at different rates, they diverge in stability at finer scales: as additional assessments are added, Gradient SHAP and DiCE differ in how reliably they maintain pairwise feature order (RC) and the exact rank positions of features within the top-$k$ set (RA). If we relax the requirement on exact positions and consider only whether a feature belongs to the top-$k$, DiCE maintains higher agreement than Gradient SHAP across all prefix lengths. It also demonstrates the greater robustness of DiCE in retaining the core subset of important assessments even as rank positions change.

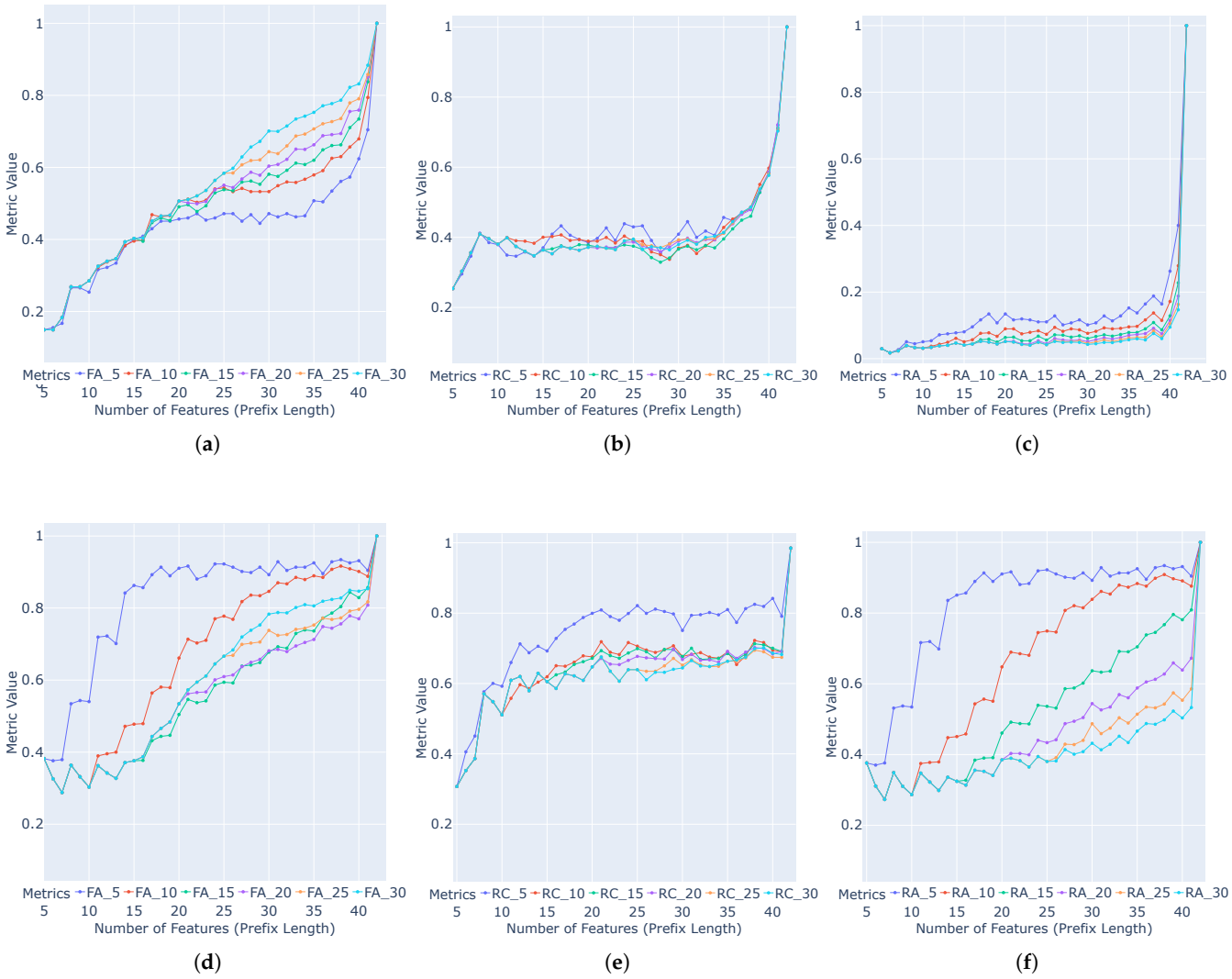



**Figure 9.** Comparison of Gradient SHAP and DiCE across three ranking agreement metrics. Top row: Gradient SHAP results for (**a**) feature agreement, (**b**) rank correlation, and (**c**) rank agreement. Bottom row: DiCE results for (**d**) feature agreement, (**e**) rank correlation, and (**f**) rank agreement. Each panel displays six curves corresponding to fixed prefix lengths $k = 5, 10, 15, 20, 25, 30$.

#### 3.4.2. Inter-Method Agreement

Inter-method agreement addresses the second facet of RQ1: whether Gradient SHAP and DiCE identify the same influential assessments at any given prefix. Concretely, we evaluate $\mathrm{RA}(R^a_{i,t}, R^b_{i,t})$ and $\mathrm{RC}(R^a_{i,t}, R^b_{i,t})$ at each prefix length $t \in \{1, \ldots, 42\}$, then average over $i$.

Figure 8c compares Gradient SHAP and DiCE rankings across prefix lengths with RA and RC; FA is omitted because it equals 1 at every prefix. Both curves decline as additional assessments are introduced, displaying growing divergence between the two explanation methods. RA curve, reflecting complete-order concordance, decreases monotonically to near zero by prefix length $x = 42$, denoting almost no agreement in overall rankings. RC curve, capturing pairwise concordance, falls more slowly and plateaus near 0.2, indicating persisting partial-order alignment. The nonzero RA values at early prefix lengths suggest that both methods consistently recognize the initial top-ranked assessments, although they diverge on the precise ranking positions of those assessments as more assessments are incrementally added to consider.

#### 3.4.3. Individual-Population Agreement

We computed both RA and RC at the full prefix length ($t = 42$, when all features are available). For each test instance $i \in \mathcal{D}_1$, we compared its local feature rank vector $R^c_{i,42}$ to the global rank vector $R^c_{\mathcal{D}_1,42}$ for each explanation method $c \in \{a, b\}$ (i.e., ranking disagreement between individuals and population): $RA(R^c_{i,42}, R^c_{D_1,42}), RC(R^c_{i,42}, R^c_{D_1,42})$. Figure 10 plots the resulting RA and RC values, using different colors and markers to indicate true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN), where the positive class is at-risk students (Fail); see Section 2.2.4.

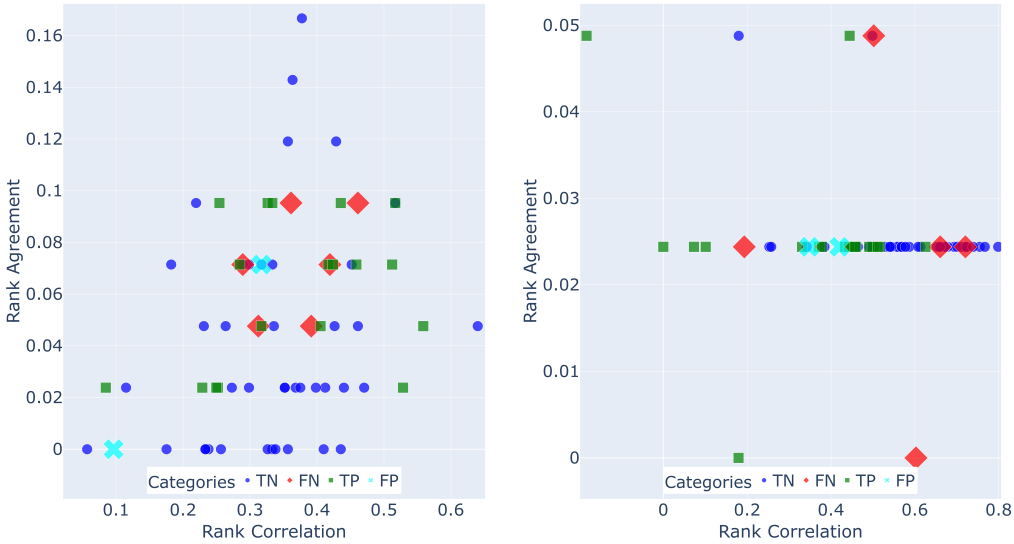



**Figure 10.** Rank Agreement vs. Rank Correlation. (**a**) Gradient SHAP. (**b**) DiCE. Markers indicate prediction categories: blue circle = TN, red diamond = FN, green square = TP, and cyan cross = FP. The positive class is at-risk students (Fail).

Under Gradient SHAP, RA values cover a relatively wide interval, reflecting a higher variability in how individual feature rankings align with the global ranking; its RC values occupy a slightly narrower range. By contrast, DiCE's RA values form a nearly horizontal band near 0.025, signifying that for virtually every instance, the exact rank positions of features diverge almost completely from the global ranking. Meanwhile, RC values span

a broader range than that of Gradient SHAP, revealing greater dispersion in pairwise feature orderings across instances. These distinct stability profiles reflect the explainers' fundamentally different mechanisms. Gradient SHAP derives attributions from model gradients, whereas DiCE constructs counterfactual examples through feature perturbation, a process that is inherently more sensitive to local variations in the feature space.

#### 3.4.4. RQ1 Summary

Taken together, Sections 3.4.1 and 3.4.2 answer RQ1: both methods' rankings stabilize over time within themselves, but the two methods do not converge toward each other as additional assessment scores arrive. Intra-method agreement rises with prefix length for both methods, indicating that early partial rankings increasingly resemble full-sequence rankings within each method. Inter-method agreement, by contrast, declines with prefix length, indicating that Gradient SHAP and DiCE diverge more as more information becomes available, even as each individually converges on its own final ranking. The individual-population analysis in Section 3.4.3 complements this answer by showing that within-method stability at the population level coexists with substantial individual-to-population divergence, which is most pronounced for DiCE. This divergence is a separate axis of agreement from the ones RQ1 addresses and is relevant to how population-level explanations should be interpreted at the individual-student level.

## 4. Discussion

The disagreement problem in XAI, where different explanation methods produce contradictory results for the same model, has been documented across multiple domains (Brughmans et al., 2024; Krishna et al., 2025; Mitruț et al., 2024). Relying on a single attribution technique can miss features that are either necessary or sufficient, and differences in feature importance rankings grow with feature dimensionality (Kommiya Mothilal et al., 2021). We therefore developed a dual-method filtering strategy that retains only assessment features ranked highly by both an attribution method (Gradient SHAP) and a counterfactual method (DiCE), following Kommiya Mothilal et al. (2021) in treating the two families as methodologically complementary. We interpret cross-method agreement as a heuristic that increases confidence in identified features relative to single-method results, not as evidence of causal validity: two methods that rest on different mechanisms can agree for reasons unrelated to the underlying causal structure, particularly when features are statistically dependent.

Using either method alone is problematic for the decisions our framework supports. Gradient SHAP achieves high feature agreement across prefix lengths (approaching 1) but rank agreement near 0, indicating substantial re-ordering as new evidence arrives; this volatility, driven by baseline shifts and input-dimensionality expansion, inflates the apparent importance of early-visible assessments and can direct instructor attention toward assessments whose rank position will not survive later evidence. DiCE has higher rank agreement but only identifies features through which minimal perturbations can flip the prediction, so assessments that predict outcomes in aggregate but fall outside minimal counterfactual sets are systematically under-credited. SHAP describes what the model is associated with; DiCE prescribes what can change its output. Intersecting the two at the population level yields a global-level top-$k$ shortlist more stable than SHAP's standalone rankings and broader than DiCE's minimal counterfactual targets; this global shortlist summarizes historical patterns across the student population and serves as a reference against which individual explanations can later be cross-checked.

For individual at-risk students, DiCE prescribes minimal score changes that would flip the predicted outcome, while SHAP's complementary attributions, visualized through stan-

dard waterfall or force plots, highlight which assessments most drive the current prediction. Paired with incremental re-forecasting as new scores arrive, this supports an iterative intervention loop: instructors direct tutoring and practice toward the assessments each method identifies as pivotal for a given student, and subsequent prefix-level predictions indicate whether the intervention has shifted the model's assessment of that student's trajectory. The remaining question is which assessments, specifically, should guide intervention for a given student at a given point in the semester. The answer depends on how the two methods' explanations agree with each other at the individual level and with the global-level top-$k$ list. We can adopt the following procedure. For a given at-risk student, we first intersect the SHAP and DiCE explanations; the intersection is non-empty when the two methods share at least one influential assessment, and empty when they share none.

- When the intersection is non-empty, the shared assessments carry both a diagnostic and a prescriptive interpretation, and the instructor can direct intervention toward them directly.
- When the intersection is empty, the instructor chooses between SHAP's attribution-based explanation (diagnostic view) and DiCE's counterfactual-based explanation (prescriptive view), and may optionally cross-reference the chosen view against the global-level top-$k$ list when that list exists.

Intersecting the two explanations is the first step. It ensures that the counterfactual prescriptions are derived from a model whose influential assessments have been independently confirmed by an attribution-based method. The resulting guidance reflects features less susceptible to method-specific biases than single-method analyses can detect. The global-level top-$k$ list plays a supporting role in the second branch, where it can further corroborate the view the instructor has chosen. However, the list assumes the student follows the population distribution, so its applicability to any particular student is a judgment the framework defers to instructor expertise. With an appropriate user interface, the same outputs can also serve students directly: a student could track their current risk status as new scores arrive, see from the SHAP attribution which past assessments most drive that status, and see from the DiCE counterfactual which assessments offer the smallest-effort path to reclassification (from at-risk to not-at-risk). This gives students concrete, self-directed targets for improvement rather than a single pass-or-fail signal, and turns the framework into a shared reference used by the instructor and students together.

### *4.1. RQ3 Summary*

Taken together, the preceding paragraphs answer RQ3 through two complementary mechanisms and a provision for the case in which they conflict. The dual-method procedure directs intervention at the intersection of the SHAP and DiCE explanations when that intersection exists for an at-risk student, and otherwise has the instructor choose one view and optionally consult the historical global-level top-$k$ list, subject to the population-distribution caveat noted above.

### *4.2. Limitations*

Several limitations constrain how these findings should be interpreted. First, our study examines a single course with 244 student sequences across four semesters, and its assessments, grading structure, and student population shape which specific homework items emerge as influential. Therefore, the framework itself is scope-limited. It assumes a course structured as ours is: semester-long, with multiple unequally-weighted assessment categories released at known points across the term, and enough assessments to yield informative prefix trajectories. Substantial adaptation would be needed for courses with few scheduled assessments, such as project- or portfolio-based courses where the

grade rests on one or two large submissions. Second, our feature-dependence diagnostics place the data within a moderate-collinearity range that applied SHAP and counterfactual studies routinely tolerate, though the implications of that range for SHAP attributions and DiCE counterfactual targets in our setting are not directly established by these diagnostics. Dependence-aware Shapley variants (Aas et al., 2021; Olsen & Jullum, 2024) and causally constrained counterfactuals (Mahajan et al., 2020) have been proposed as partial remedies, but no single method has emerged as standard practice, and each involves trade-offs between fidelity, computational cost, and the strength of assumptions placed on the data-generating process. We did not use dependence-aware variants here because they are less mature as tooling than the standard implementations, and substituting them asymmetrically (for example, pairing a dependence-aware SHAP with original DiCE) would introduce a source of measured disagreement unrelated to the attribution-versus-counterfactual contrast our framework is designed to isolate. A fully symmetric substitution, using dependence-aware versions of both methods, is a natural direction for future work. Third, our agreement analysis is descriptive: it measures whether two methods converge, not whether they converge on causally correct features, and counterfactual explanations in an observational dataset with correlated assessments cannot by themselves establish causal effects on student outcomes.

### *4.3. Future Work*

Future work could extend the framework along four directions. A natural next step is to apply the framework to data from other courses and institutions, indicating whether the patterns observed here extend beyond the single course analyzed in this study. A second direction is to incorporate dependence-aware explanation methods, such as conditional Kernel SHAP, causal or asymmetric Shapley values, and causally constrained counterfactual generators, which would allow the cross-method comparison to proceed under less restrictive independence assumptions. Third, resampling-based stability selection becomes feasible with larger datasets or faster counterfactual methods and would provide a more principled complement to the midterm-referenced stability criterion we used; we did not adopt it here because DiCE counterfactual generation is computationally expensive relative to Gradient SHAP. Finally, evaluating real-world impact through instructor-led interventions informed by the framework's outputs would test whether cross-method verification translates into improved student outcomes, a claim our descriptive analysis cannot by itself establish.

## 5. Conclusions

This study applied a dual-method explanation framework, combining Gradient SHAP and DiCE, to a GRU model predicting student course outcomes from incremental assessment sequences in a foundational Data Structures and Algorithms course. Three findings answer the questions posed at the outset. As the semester progresses, both methods' rankings stabilize internally, though to different degrees, while the two methods increasingly disagree with each other on precise ordering (RQ1). A compact subset of early-semester homework assessments is consistently identified as most influential by both methods (RQ2). At the individual student level, the two explanations can be intersected to yield intervention targets that carry both a diagnostic and a prescriptive interpretation; when no such intersection exists, the instructor selects one view and may cross-reference it against the global-level top-$k$ list, producing guidance less susceptible to method-specific biases than single-method analyses. With an appropriate user interface, the same outputs can also serve students directly as a self-directed reference (RQ3).

Combining explanation methods grounded in distinct principles increases confidence in the identified features relative to single-method results, without establishing causal validity. We view this cross-method verification as a step toward more responsible use of learning analytics in teaching practice.

**Supplementary Materials:** The source code for this study is available at https://github.com/for-research-only/explainable-student-performance.git (accessed on 10 April 2025).

**Author Contributions:** Conceptualization, Y.L., J.Z. and T.Z.; methodology, Y.L., J.Z. and T.Z.; software, Y.L. and A.S.R.; validation, Y.L.; formal analysis, Y.L., J.Z. and T.Z.; investigation, Y.L.; data curation, A.S.R. and T.Z.; writing—original draft preparation, Y.L.; writing—review and editing, Y.L., J.Z. and T.Z.; visualization, Y.L.; supervision, J.Z. and T.Z.; project administration, T.Z.; funding acquisition, T.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work received support from the University of Wisconsin–Milwaukee (UWM) Foundation.

**Data Availability Statement:** The data analyzed in this study cannot be shared in any form, including de-identified or aggregated form, due to University of Wisconsin–Milwaukee institutional policies governing the use and disclosure of student educational records. These policies permit use only for the original study and do not allow data release on request.

**Acknowledgments:** The authors thank John Boyland for sharing the CompSci 351 course data and details of the course structure used in this study.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

| | |
|---|---|
| XAI | eXplainable Artificial Intelligence |
| SHAP | SHapley Additive exPlanations |
| DiCE | Diverse Counterfactual Explanations |
| GRU | Gated Recurrent Unit |

## Notes

1 The "Fail" label here reflects competency-level performance rather than institutional academic failure. The institution's formal failing grade is F (scores below 60, i.e., below D−), and grades in the D range (D−, D, D+) are institutionally passing; however, a D-range performance falls below the competency standard required to proceed confidently to subsequent courses that depend on Data Structures and Algorithms as a prerequisite, which is the intervention-relevant criterion for our early-prediction framework.

2 A tiebreaking rule was not required in our 20-run pool, as all surviving candidates had clearly distinct $c$ and $a$ values; should ties arise in future applications, we suggest using mean prediction accuracy across all prefix lengths as a secondary criterion. Additional implementation details are discussed in Section 3.1.

3 The version of the SHAP library we used is 0.47.0. See URL https://github.com/shap/shap, accessed on 10 April 2025.

4 We cloned the DiCE repository for compatibility inspection during integration with our experiment scripts. See URL https://github.com/interpretml/DiCE, accessed on 10 April 2025.

## References

Aas, K., Jullum, M., & Løland, A. (2021). Explaining individual predictions when features are dependent: More accurate approximations to Shapley values. *Artificial Intelligence*, *298*, 103502. [CrossRef]

Abdi, H. (2010). The greenhouse-geisser correction. *Encyclopedia of Research Design*, *1*, 544–548.

Adnan, M., Uddin, M. I., Khan, E., Alharithi, F. S., Amin, S., & Alzahrani, A. A. (2022). Earliest possible global and local interpretation of students' performance in virtual learning environment by leveraging explainable AI. *IEEE Access*, *10*, 129843–129864. [CrossRef]

Albreiki, B., Habuza, T., & Zaki, N. (2022). Framework for automatically suggesting remedial actions to help students at risk based on explainable ML and rule-based models. *International Journal of Educational Technology in Higher Education*, *19*(1), 49. [CrossRef]

Alsariera, Y. A., Baashar, Y., Alkawsi, G., Mustafa, A., Alkahtani, A. A., & Ali, N. (2022). Assessment and evaluation of different machine learning algorithms for predicting student performance. *Computational Intelligence and Neuroscience*, *2022*, 4151487. [CrossRef]

Alwarthan, S., Aslam, N., & Khan, I. U. (2022). An explainable model for identifying at-risk student at higher education. *IEEE Access*, *10*, 107649–107668. [CrossRef]

Aumann, R. J., & Shapley, L. S. (1974). *Values of non-atomic games*. Princeton University Press.

Brughmans, D., Melis, L., & Martens, D. (2024). Disagreement amongst counterfactual explanations: How transparency can be misleading. *TOP*, *32*(3), 429–462. [CrossRef]

Cagliero, L., Canale, L., Farinetti, L., Baralis, E., & Venuto, E. (2021). Predicting student academic performance by means of associative classification. *Applied Sciences*, *11*(4), 1420. [CrossRef]

Callaghan, K. J. (2008). Revisiting the collinear data problem: An assessment of estimator 'ill-conditioning' in linear regression. *Practical Assessment, Research, and Evaluation*, *13*(1), 5. [CrossRef]

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. [CrossRef]

Chen, H.-C., Prasetyo, E., Tseng, S.-S., Putra, K. T., Prayitno, Kusumawardani, S. S., & Weng, C.-E. (2022). Week-wise student performance early prediction in virtual learning environment using a deep explainable artificial intelligence. *Applied Sciences*, *12*(4), 1885. [CrossRef]

Erion, G., Janizek, J. D., Sturmfels, P., Lundberg, S., & Lee, S.-I. (2020). Improving performance of deep learning models with axiomatic attribution priors and expected gradients improving performance of deep learning models with axiomatic attribution priors and expected gradients. *arXiv*. [CrossRef]

García-Martínez, I., Fernández-Batanero, J. M., Fernández-Cerero, J., & León, S. P. (2023). Analysing the impact of artificial intelligence and computational sciences on student performance: Systematic review and meta-analysis. *Journal of New Approaches in Educational Research*, *12*(1), 171–197. [CrossRef]

He, Y., Chen, R., Li, X., Hao, C., Liu, S., Zhang, G., & Jiang, B. (2020). Online at-risk student identification using RNN-GRU joint neural networks. *Information*, *11*(10), 474. [CrossRef]

Hocking, R. R. (1973). A discussion of the two-way mixed model. *The American Statistician*, *27*(4), 148–152. [CrossRef]

Hoq, M., Brusilovsky, P., & Akram, B. (2023, July 11–14). *Analysis of an explainable student performance prediction model in an introductory programming course* [Conference Paper]. 16th International Conference on Educational Data Mining (EDM 2023), Bengaluru, India.

Jang, Y., Choi, S., Jung, H., & Kim, H. (2022). Practical early prediction of students' performance using machine learning and eXplainable AI. *Education and Information Technologies*, *27*(9), 12855–12889. [CrossRef]

Kendall, M. G. (1938). A new measure of rank correlation. *Biometrika*, *30*(1–2), 81–93. [CrossRef]

Kommiya Mothilal, R., Mahajan, D., Tan, C., & Sharma, A. (2021, May 19–21). *Towards unifying feature attribution and counterfactual explanations: Different means to the same end* [ACM Conference]. 2021 AAAI/ACM Conference on AI, Ethics, and Society, Virtual. [CrossRef]

Krishna, S., Han, T., Gu, A., Wu, S., Jabbari, S., & Lakkaraju, H. (2025). The disagreement problem in explainable machine learning: A practitioner's perspective the disagreement problem in explainable machine learning: A practitioner's perspective. *arXiv*. [CrossRef]

Lundberg, S., & Lee, S.-I. (2017). A unified approach to interpreting model predictions a unified approach to interpreting model predictions. *arXiv*. [CrossRef]

Mahajan, D., Tan, C., & Sharma, A. (2020). Preserving causal constraints in counterfactual explanations for machine learning classifiers preserving causal constraints in counterfactual explanations for machine learning classifiers. *arXiv*. [CrossRef]

Marcoulides, K. M., & Raykov, T. (2019). Evaluation of variance inflation factors in regression models using latent variable modeling methods. *Educational and Psychological Measurement*, *79*(5), 874–882. [CrossRef]

Mitruț, O., Moise, G., Moldoveanu, A., Moldoveanu, F., Leordeanu, M., & Petrescu, L. (2024). Clarity in complexity: How aggregating explanations resolves the disagreement problem. *Artificial Intelligence Review*, *57*(12), 338. [CrossRef]

Mothilal, R. K., Sharma, A., & Tan, C. (2020). Explaining machine learning classifiers through diverse counterfactual explanations. In *proceedings of the 2020 conference on fairness, accountability, and transparency* (pp. 607–617). Association for Computing Machinery. [CrossRef]

Olsen, L. H. B., & Jullum, M. (2024). Improving the sampling strategy in KernelSHAP improving the sampling strategy in KernelSHAP. *arXiv*. [CrossRef]

Pawlicki, M. (2023, October 9–13). *Towards quality measures for xAI algorithms: Explanation stability*. 2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA) (pp. 1–10), Thessaloniki, Greece. [CrossRef]

Pierce, C. A., Block, R. A., & Aguinis, H. (2004). Cautionary note on reporting eta-squared values from multifactor ANOVA designs. *Educational and Psychological Measurement*, *64*(6), 916–924. [CrossRef]

Rehman, N., Huang, X., Batool, S., Andleeb, I., & Mahmood, A. (2024). Assessing the effectiveness of project-based learning: A comprehensive meta-analysis of student achievement between 2010 and 2023. *ASR: Chiang Mai University Journal of Social Sciences and Humanities*, *11*(2), e2024015. [CrossRef]

Schwalbe, G., & Finzel, B. (2024). A comprehensive taxonomy for explainable artificial intelligence: A systematic survey of surveys on methods and concepts. *Data Mining and Knowledge Discovery*, *38*(5), 3043–3101. [CrossRef]

Shapley, L. S. (1953). 17. A value for n-person games. In H. W. Kuhn, & A. W. Tucker (Eds.), *Contributions to the theory of games (AM-28), Volume II* (pp. 307–318). Princeton University Press. [CrossRef]

Silva, E. J. R., & Zanchettin, C. (2015, October 9–12). *On the existence of a threshold in class imbalance problems*. 2015 IEEE International Conference on Systems, Man, and Cybernetics (pp. 2714–2719), Kowloon Tong, Hong Kong. [CrossRef]

Smith, B. I., Chimedza, C., & Bührmann, J. H. (2022). Individualized help for at-risk students using model-agnostic and counterfactual explanations. *Education and Information Technologies*, *27*(2), 1539–1558. [CrossRef]

Sundararajan, M., & Najmi, A. (2020, July 13–18). *The many shapley values for model explanation*. 37th International Conference on Machine Learning, Virtual.

Sundararajan, M., Taly, A., & Yan, Q. (2017, August 6–11). *Axiomatic attribution for deep networks*. 34th International Conference on Machine Learning (Vol. 70, pp. 3319–3328), Sydney, NSW, Australia.

Swamy, V., Radmehr, B., Krco, N., Marras, M., & Käser, T. (2022). *Evaluating the explainers: Black-box explainable machine learning for student success prediction in MOOCs.* International Educational Data Mining Society. [CrossRef]

Tiukhova, E., Vemuri, P., Flores, N. L., Islind, A. S., Óskarsdóttir, M., Poelmans, S., Baesens, B., & Snoeck, M. (2024). Explainable learning analytics: Assessing the stability of student success prediction models by means of explainable AI. *Decision Support Systems*, *182*, 114229. [CrossRef]

Tomek, IVAN. (1976). Two modifications of CNN. *IEEE Transactions on Systems, Man, and Cybernetics*, *SMC-6*(11), 769–772. [CrossRef]

Ujkani, B., Minkovska, D., & Hinov, N. (2024). Course success prediction and early identification of at-risk students using explainable artificial intelligence. *Electronics*, *13*(21), 4157. [CrossRef]

Wachter, S., Mittelstadt, B., & Russell, C. (2017–2018). Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harvard Journal of Law & Technology (Harvard JOLT)*, *31*(2), 841–888.

Waliszewski, P., & Konarski, J. (2005). A mystery of the Gompertz function. In G. A. Losa, D. Merlini, T. F. Nonnenmacher, & E. R. Weibel (Eds.), *Fractals in biology and medicine* (pp. 277–286). Birkhäuser-Verlag. [CrossRef]