Computers and Education: Artificial Intelligence 8 (2025) 100432

Contents lists available at ScienceDirect

Computers and Education: Artificial Intelligence

journal homepage: www.sciencedirect.com/journal/computers-and-education-artificial-intelligence

A real-time AI tool for hybrid learning recommendation in education: Preliminary results ☆

Chaman Verma

Eötvös Loránd University, Department of Media & Educational Technology, Faculty of Informatics, Budapest, 1053, Hungary

ARTICLE INFO

Keywords:
Gradio
Hybrid learning
Prediction
Classification
Stratified cross-validation
Support vector machine
Smote-NC

ABSTRACT

In recent years, owing to previous pandemics, online and offline education have collaboratively influenced students' academic experiences globally. This mixed-methods strategy offers both benefits and drawbacks for students. This study created an innovative AI tool utilizing the Support Vector Machine (SVM) algorithm on primary samples of Hungarian informatics students to assess their suitability for adopting hybrid learning in their studies. This paper also explained the strength of the model with Shapley Additive exPlanations (SHAP) values of the Explainable Artificial Intelligence (XAI) method. The SVM parameters were suggested by Grid Search Cross Validation (GSCV) to provide an optimal classifier with maximum accuracy. The trained model has been tested and validated with new realistic samples created with a non-parametric approach, Kernel Density Estimation (KDE). The newly created test data sets correspond to the actual data distribution of the data set and preserve the relationships between classes of characteristics, as validated by the Kolmogorov–Smirnov (KS) test ($p < 0.05$). The average precision cross-validation of training and testing the data set is 0.9881 and 0.9757, respectively, for classifying the target class of the hybrid learning data set. Additionally, the model achieved the highest f1-score, precision, and recall with values of 0.98. The Area Under Curve (AUC) is calculated to be 1, indicating a high value. According to the SVM coefficient, Overall Satisfaction (OS), Long Term Solution (LRS), and Feeling Happiness (FH) were the most influential attributes, with positive coefficients of 0.78, 0.49, 0.47 and 0.41 respectively. The negative coefficients for the Challenging Group task (CGT), and Obstacle (OBST), are -0.42, and -0.16, respectively. The SHAP explained the SVM model strength and voted for OS, FH, LRS, and CGT in the decision. The study's findings can help students and institutional administration decide whether to switch to hybrid education based on pros, cons, and other recommended features.

1. Introduction and related work

Online and in-person training are combined in hybrid learning, and students' learning behaviors determine outcomes Ding and Zhou (2023). Hybrid learning offers greater effectiveness than purely online education. As a result, numerous universities are exploring the creation of hybrid courses, programs, and degree options to leverage its advantages Ramadhani et al. (2023). Hybrid learning is closely linked to collaborative learning, continuous education, sustainability, and the protection of the environment Hermitaño-Atencio et al. (2023). Studies on e-learning and how happy students are have looked at different flexible and hybrid learning methods, and they found that students liked mixed-ability classes, which helped them get more involved. Positive trends emerged regarding student engagement, satisfaction with hybrid learning, and the relationship between lecturer support and engagement Hussein et al. (2020). The quality of e-learning services significantly impacted Indonesian students' satisfaction, with user happiness playing a critical role in their overall experience Saputri et al. (2022). For graduate students, blended learning and in-person classes contributed positively to their satisfaction levels (Chung and Gao (2021)). In Thailand, students expressed high satisfaction with hybrid learning during the study period ($\mu = 4.206, \sigma = 0.74$) Purahong et al. (2021). Arabian students adapted well to online learning, with hybrid learning during the pandemic enhancing their self-esteem and satisfaction (Senior (2022)). In Chile, no

☆ This work is supported by the Digital Education Development Competence Center under the project: 2022-1.1.1-KK-2022-00023, which project is financially supported by the National Research, Development and Innovation Office, Hungary.

E-mail address: chaman@inf.elte.hu.
URL: https://www.inf.elte.hu/en/staff/chaman-verma.

https://doi.org/10.1016/j.caeai.2025.100432
Received 20 January 2025; Received in revised form 9 May 2025; Accepted 26 May 2025
Available online 2 June 2025

significant differences were observed in learning outcomes between face-to-face and remote learners Paiva et al. (2021). South African university students expressed satisfaction with their learning management systems, particularly in addressing their immediate academic needs Mlitwa and Ogundaini (2022). Additionally, machine learning algorithms were used to analyze student intentions based on factors such as perceived risk, communication ease, usability, time and location flexibility, and gender Trivedi et al. (2022).

Computer science and AI employ data and algorithms to forecast data trends in machine learning. Predicting numerous academic dataset traits is also common Venkatesan and Priya (2015). SVM, a prominent classification approach, combines margin hyperplane and kernel computation theories. The best hyperplane is found by measuring the margin and finding its most excellent point. A linear classification hyperplane and kernels that map data characteristics from lower to higher dimensions are used in linear SVM Findiana et al. (2020). SVM has shown high prediction accuracy (75.72%) in predicting student academic performance, dropout, and learning styles Assegie et al. (2024). A two-class SVM accurately assessed teachers' teaching abilities in the resource design process and the integration of machine learning into traditional education Cui (2024). With 84% accuracy, SVM also helped categorize online class student comments on their involvement in the upcoming semester Kurniawan and Wahyuni (2021). The SVM analysis revealed that most tweets on ChatGPT exhibit favorable or neutral feelings, while a small fraction express negative attitudes from students Tubishat et al. (2023). The SVM algorithm determines whether to suggest general education content to learners Sun (2021).

Previous research did not adequately consider the perspectives on hybrid learning in educational decision-making. Additionally, earlier studies did not employ machine learning algorithms to offer recommendations for hybrid learning environments. Furthermore, researchers have not identified any real-time tools to explore the innovative aspects of hybrid learning and provide guidance to students. Moreover, researchers did not utilize understandable artificial intelligence to thoroughly examine opinions on hybrid learning. Selecting the SVM has proven advantageous for making accurate predictions on educational datasets, as demonstrated by previous research. The present work draws inspiration from the previous SVMM findings in education datasets. In addition, this research has incorporated the SHAP technique into the SVM. Before this, the SVM algorithm was utilized to forecast the combination of lab and theoretical classes in hybrid learning Verma et al. (2024). This paper focuses on developing a new predictive model, Future Hybrid Learning Mode (FHLM), using the SVM algorithm as preliminary research.

The study has seven primary sections. Refer to Section 1 for an introduction and recent research. Section 2 outlines the statement of the problem, and Section 3 is about the contribution. Detailed methodology, framework, dataset preprocessing, testing, tuning parameters, and SVM description can be found in Section 4. Section 5 debates the results, and Section 6 explains the strength of the proposed SVM model with SHAP. Section 7 discusses the findings of experiments effectively. Section 8 presents implications. Section 9 explores the configuration and development environment of a novel real-time application. The study's results are presented in Section 10.

2. Problem statement

A structured educational technique that combines classroom and internet learning is hybrid learning. It combines the benefits of in-person education—synchronous engagement, peer cooperation, and quick feedback—with internet technologies' flexibility, accessibility, and customization. Hybrid learning meets modern learner needs with institutional approaches for flexibility and resilience, unlike reactive and transient emergency remote instruction. After COVID, students are seeking learning options that support their health, remote access, and time management demands, making hybrid learning more important. Institutions must provide inclusive, scalable, and successful learning environments. Hybrid learning does not work for all students or situations. Some students excel, while others struggle with low satisfaction, challenges, and interaction. Without data-driven tools to assess a student's hybrid learning preparedness, educators and administrators make judgments based on assumptions or generic regulations. Accordingly, the following primary research questions have been formulated to guide the investigation:

- RQ:1 How can we use SVM to identify and assess important features of hybrid learning?
  This frames the use of SVM as a robust, accurate method for classification and feature weighting in a hybrid learning dataset.
- RQ:2 How can we design a web-based system to recommend hybrid learning based on student responses?
  It addresses the design and implementation of an AI-driven decision-support system that directly analyzes individual feedback.
- RQ:3 What are the main benefits and challenges influencing the adoption of hybrid learning?
  It guides the identification of key features (e.g., satisfaction, happiness, benefits, and challenges) derived from student responses.
- RQ:4 How can interpretable methods (SHAP) help explain key factors in hybrid learning recommendations?
  It highlights the role of SHAP (SHapley Additive exPlanations) in making AI decisions transparent, thus increasing trust and usefulness in academic decisions.

3. Contribution

This study introduces a novel real-time SVM-FHLM AI-based web application designed to recommend hybrid learning strategies based on student feedback. The application effectively addressed the binary classification challenge to determine whether hybrid learning should be recommended for future use. It offers a validated prediction model, an interpretable user interface, and data-driven insights into the key features of hybrid learning that influence its adoption. It explores novel features such as satisfaction, happiness, benefits, challenges, and university support or assistance related to hybrid learning. The model has been validated with KDE techniques to make it more robust, generalized with the tool, and explained with SHAP techniques, which were used to elucidate the critical aspects of predictors of hybrid learning.

The findings suggest that hybrid learning could be a viable, long-term solution. Students can incorporate a hybrid learning strategy into their academic activities, such as alternating weekdays between online theory and offline lab practice. Students were pleased and satisfied with the time management techniques they employed to attend two classes at once. They also believed that hybrid learning was especially useful for unwell people. It was also discovered that students' parents favored the hybrid learning mode in the event of future severe pandemics like COVID-19, and students themselves agreed that hybrid learning is the greatest option for them as a secure online learning opportunity. Challenges included isolation from the authentic study environment, diminished student contact, disturbances caused by microphones and video cameras, and difficulties in managing group or team-based assignments, which were also considered vital predictors in the study.

4. Method

4.1. Conceptual research framework

Fig. 1 illustrates the pictorial representation of the research. It shows the conceptual framework of the proposed research. At the outset, the primary dataset comprises records and features regarding hybrid learning from students' perspectives and has been preprocessed on a scale of 1 to 10. Subsequently, factor analysis was implemented to transform specific attributes into factors. Later, the minority class's records

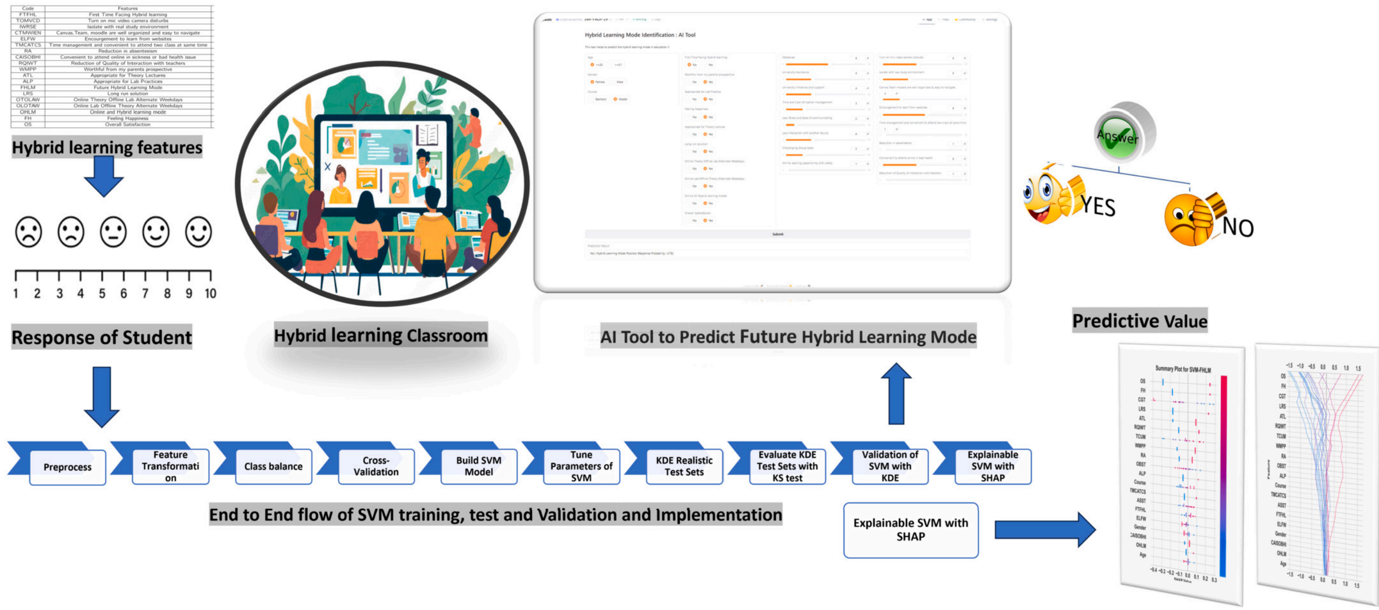


Fig. 1. Support Vector Machine-Based AI Tool: A Framework.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD
TOMVCD	IWRSE	CTMWIEN	ELFW	TMCATCS	RA	CAISOBHI	RQIWT	Age	Gender	Course	FTFHL	WMPP	FH	ALP	ATL	LRS	OTOLAW	OLOTAW	OHLM	OS	OBST	ASST	UNIS	TCUM	LSEC	LUWAF	OLOWS	CGT	FHLM
-0.65552	0.38196	-0.39201	-0.70215	0.83031	-1.22178	0.23217	0.59702	0	0	0	0	1	1	0	1	1	1	0	0	1	-0.10864	0.074	-1.01399	1.27883	-0.74856	-0.24584	0.47822	0.16267	1
-0.65552	-0.86465	-1.41331	-1.17503	1.23725	1.37943	0.80678	-0.60105	0	1	0	1	1	1	1	1	1	1	0	0	1	-0.92329	-1.78575	-1.2299	1.20111	0.3335	-0.8099	2.1992	0.30915	1
-0.65552	-0.86465	-0.90266	-0.70215	0.83031	-0.35471	-0.91706	-0.20169	0	0	0	1	1	1	1	1	1	1	0	0	1	-1.0284	-1.48485	-0.00984	-0.15196	-0.43964	-0.76122	0.20626	0.28087	1
0.14567	0.38196	0.11864	-2.59367	1.23725	1.37943	0.80678	0.19766	1	1	1	1	0	0	0	1	1	0	0	1	1	0.49651	0.86843	-2.82145	2.00954	1.61745	0.95759	-1.08807	-0.9609	1
-0.65552	-1.28018	0.62928	0.71648	-0.79743	-0.78825	0.80678	-1.39976	0	1	0	0	1	1	1	1	1	0	0	1	1	-1.35967	0.74442	0.80198	-0.59547	1.02591	-0.07509	-0.58519	-0.39615	1
0.94686	-1.28018	0.11864	-0.22927	1.23725	-0.78825	0.80678	-0.60105	1	1	1	1	1	1	1	1	1	0	0	1	1	-1.36084	0.72355	-1.43315	-1.08847	0.482	-0.57991	1.64854	0.75081	1
-0.25492	0.38196	-0.39201	1.18936	-1.6113	0.9459	0.23217	-0.20169	0	0	1	0	1	1	0	1	1	0	0	1	1	-0.18366	-0.04976	1.82815	-0.54864	0.65911	0.05294	0.28268	0.507	1
1.34746	1.21302	1.13993	1.18936	0.83031	0.51236	0.23217	0.59702	1	1	1	1	1	0	0	1	1	0	1	0	1	1.93302	-0.32231	1.43311	0.29635	-2.16292	-0.39389	0.70669	-0.02258	1
1.34746	1.21302	1.13993	1.18936	1.23725	1.37943	0.80678	0.19766	1	1	1	1	1	1	1	1	1	0	1	0	1	0.76638	0.57007	1.57442	0.95023	1.10127	0.33022	0.627	0.60397	1
-0.25492	1.21302	1.13993	1.18936	-0.79743	-2.08885	0.80678	0.19766	0	0	0	0	0	0	0	0	0	0	1	0	0	1.05076	1.69346	-0.80762	-0.73694	-1.54213	-0.28636	-0.92336	-0.18191	0
-1.05612	-1.69571	0.62928	0.2436	0.42338	-1.65532	-0.34244	0.19766	1	0	1	0	1	1	0	1	1	0	1	0	1	-1.26714	0.63634	0.37936	-0.40088	-1.61976	-1.2623	0.28231	-0.64029	1
0.94686	0.79749	0.11864	-1.17503	0.42338	-0.35471	-0.34244	-0.20169	0	1	0	1	1	0	0	1	1	0	1	0	1	-0.59276	1.16251	-0.71025	-0.06909	-0.86304	0.24307	-1.46925	0.19446	1
0.54627	-0.03358	1.13993	1.18936	-0.39049	0.07882	-1.49167	0.19766	0	1	1	0	0	0	0	0	0	0	0	1	0	0.35291	1.0074	1.74344	-0.65803	0.76676	0.77431	-2.07418	0.66856	0

Fig. 2. Preprocessed Dataset Input File.

were balanced using the Synthetic Minority Over-sampling TEchnique-Nominal Continuous (`Smote-NC`) Chawla et al. (2002). Standardized (z-score) features were excluded from factor analysis. After adjusting the hyperparameters, the cleaned data set was checked for accuracy using the stratified cross-validation method and trained with the SVM algorithm. The models' efficacy has been evaluated using a variety of classification metrics, and the model has been validated with KDE test sets. The SHAP scores of the explainable AI approach subsequently validated the performance of the SVM model. Features have been evaluated at the instance, class, and aggregate levels. This showcases the utilization of the SVM with the SHAP algorithm to determine the suitability of hybrid learning as a forthcoming trend. The anticipated outcomes will be either' 'Yes" or' 'No." As an early finding, the most important contribution of this research is to develop an AI web application/tool that identifies the hybrid learning mode possibility to adopt or not in higher education. Furthermore, the combination of an explicable artificial intelligence model (SHAP) and a standard machine learning model (SVM) to define the hybrid learning mode in the field of education is a unique contribution.

4.2. Dataset and preprocess

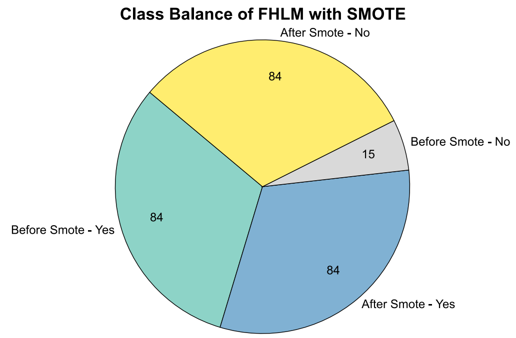
Table 1 shows that the dataset contains thirty features, one of which is a target, and the remaining twenty-nine features are also predictors. The dataset consists of eight factors and twenty-one features. Thirteen features are binary, while the remaining are standardized (Z-score). This study used ninety-nine students from one hundred and three primary data samples after removing outliers Verma and Illes (2023). The initial stage consisted of 99 instances. Implementing class balancing using the `Smote-NC` has resulted in 168 instances now available for experimentation.

Table 1
Dataset Description.

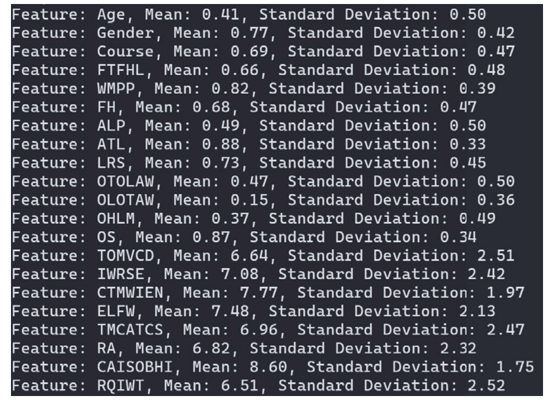
Particular	N
Target	1
Predictors	29
Factors	8
Features	21
Categorical	13
Standardized	16
Initial Instances	99
Final Instances	168

The reason is to apply `Smote-NC` because it was extended to handle nominal and continuous data types. Students studied informatics at Eötvös Loránd University. They studied informatics for graduation and post-graduation. Google Forms was used as the method for collecting samples Dataset (2022). The Likert scale (1–10) was used to test understanding, with one representing no agreement and ten representing complete agreement. No data was missing. IBM SPSS Version 29 performed factor analysis to create the most vital eight factors, and standardized values were used Verma (2024). The SVM algorithm has been implemented in the Python 3.12 language, and the final preprocess dataset input file can be seen in Fig. 2.

$$z - score = \frac{x - \mu}{\sigma} \quad (1)$$



(a) Class Balance



```
Feature: Age, Mean: 0.41, Standard Deviation: 0.50
Feature: Gender, Mean: 0.77, Standard Deviation: 0.42
Feature: Course, Mean: 0.69, Standard Deviation: 0.47
Feature: FTFHL, Mean: 0.66, Standard Deviation: 0.48
Feature: WMPP, Mean: 0.82, Standard Deviation: 0.39
Feature: FH, Mean: 0.68, Standard Deviation: 0.47
Feature: ALP, Mean: 0.49, Standard Deviation: 0.50
Feature: ATL, Mean: 0.88, Standard Deviation: 0.33
Feature: LRS, Mean: 0.73, Standard Deviation: 0.45
Feature: OTOLAW, Mean: 0.47, Standard Deviation: 0.50
Feature: OLOTAW, Mean: 0.15, Standard Deviation: 0.36
Feature: OHLM, Mean: 0.37, Standard Deviation: 0.49
Feature: OS, Mean: 0.87, Standard Deviation: 0.34
Feature: TOMVCD, Mean: 6.64, Standard Deviation: 2.51
Feature: IWRSE, Mean: 7.08, Standard Deviation: 2.42
Feature: CTMWIEN, Mean: 7.77, Standard Deviation: 1.97
Feature: ELFW, Mean: 7.48, Standard Deviation: 2.13
Feature: TMCATCS, Mean: 6.96, Standard Deviation: 2.47
Feature: RA, Mean: 6.82, Standard Deviation: 2.32
Feature: CAISOBHI, Mean: 8.60, Standard Deviation: 1.75
Feature: RQIWT, Mean: 6.51, Standard Deviation: 2.52
```

(b) Feature's statistics

Fig. 3. Instance Distribution and Statistics of Features.

Equation (1) represents the mathematical expression employed by the `StandardScaler()` function, where μ denotes the average value and σ represents the variability of the training samples.

Algorithm 1 SMOTE-NC (Nominal and Continuous Features).

Require: Minority dataset $\mathcal{D}_m$, synthetic samples N, neighbors k
Ensure: Synthetic dataset $\mathcal{D}_s$
Initialize $\mathcal{D}_s = \emptyset$
for each $x_i \in \mathcal{D}_m$ **do**
  Find k nearest neighbors of x_i
  for each synthetic sample to generate **do**
    Randomly pick neighbor x_{nn}
    for each feature f_j **do**
      if f_j is continuous **then**
        $x_{syn}[f_j] = x_i[f_j] + \delta \cdot (x_{nn}[f_j] - x_i[f_j])$, where $\delta \sim U(0,1)$
      else
        Set $x_{syn}[f_j]$ to most frequent value
      end if
    end for
    Add x_{syn} to $\mathcal{D}_s$
  end for
end for
return $\mathcal{D}_s$

The Nominal and Continuous Features (SMOTE-NC) algorithm is applied to balance the target class of hybrid learning Chawla et al. (2002). The algorithm of `SMOTENC class` describes that the input was minority class "No," majority class "Yes," and the initial number of samples N. The `SMOTENC()` function has been used with three parameters, `categorical_features`, `random_state=42`, and `k_neighbors=4`, for the dataset. The `imbalanced-learn` library has `over_sampling` module with `SMOTENC class` that has been used. Fig. 3 (a) depicts the distribution of records of the dataset before and after using the `Smote-NC`. Before the class balance, the category "Yes" contains 84 records, whereas the category "No" contains 15. Subsequently, the `Smote-NC` method increased the number of instances in the underrepresented class "No" to achieve a balanced distribution with the class "Yes." Fig. 3 (b) describes the statistical properties of 21 predictors. The features are with short abbreviations. The average and dispersion scores of predictors are mentioned, and it has been noted that there is less dispersion.

The sample adequacy was tested with the Kaiser-Meyer-Olkin (KMO) test, which calculates the proportion of common variance among variables Kaiser (1970), and sphericity was tested with the Bartlett test Bartlett (1937). The KMO test estimated 0.771, which is higher than 0.60, allowing us to implement a factor analysis approach. The Bartlett test of sphericity indicated that the variable correlation matrix

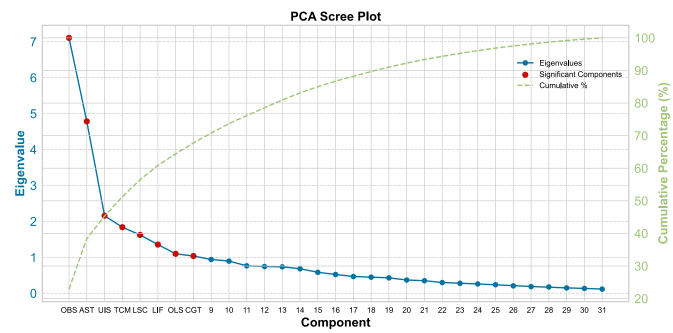
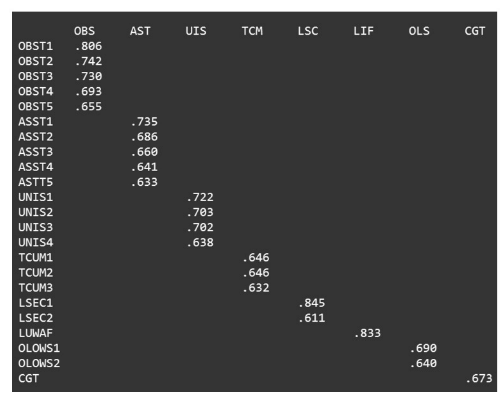


Fig. 4. Scree plot of PCA components.



	OBS	AST	UIS	TCM	LSC	LIF	OLS	CGT
OBST1	.806							
OBST2	.742							
OBST3	.730							
OBST4	.693							
OBST5	.655							
ASST1		.735						
ASST2		.686						
ASST3		.660						
ASST4		.641						
ASTT5		.633						
UNIS1			.722					
UNIS2			.703					
UNIS3			.702					
UNIS4			.638					
TCUM1				.646				
TCUM2				.646				
TCUM3				.632				
LSEC1					.845			
LSEC2					.611			
LUWAF						.833		
OLOWS1							.690	
OLOWS2							.640	
CGT								.673

Fig. 5. Factor loadings (Correlation).

is not an identity matrix, indicating that variables are connected and acceptable for factor analysis ($p < 0.05$).

Fig. 4 displays the scree plot with vital components with eigen score and cumulative percentage. A red circle on the blue line indicates the eight significant components. The highest eigen score of 7 in OBS significantly contributed to factor formation. The second robust component is AST, which exhibits five eigenvalues. The next three essential components are UIS, TCM, and LSC, each with an eigenvalue score of 2, while the next critical components are LIF, OLS, and CGT, with an eigenvalue score of 1. The cumulative percentage of variance for OBS is 23, whereas AST is 15, UIS is 7, TCM is 6, LSC is 5, LIF and OLS are both 4, and CGT is 3. The total cumulative variance is 68%.

Fig. 5 shows the factor loadings of the rotated component. A matrix represents the Pearson correlations between features and components. OBS and AST have five correlated features, respectively. UIS has four,

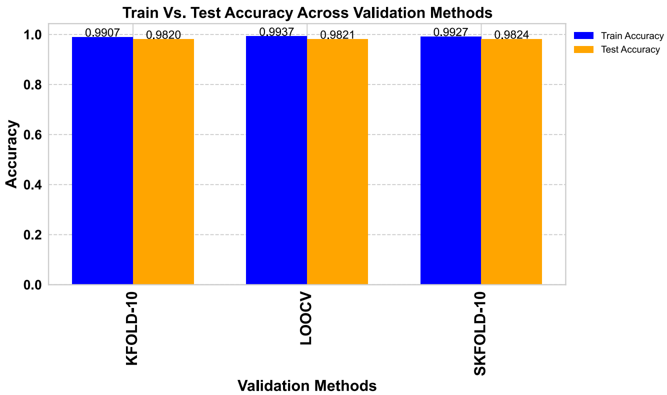


Fig. 6. Dataset Testing Techniques.

and TCM has three; LCS and OLS have two features correlated. CGT itself is a strong component. The value of correlation lies between 0.611 and 0.806. Consequently, It is identified no cross-loading issues. Hence, the indicated components are significant.

4.3. Testing approaches

A balanced dataset has undergone evaluation using four rigorous testing methodologies: k-fold Cross-Validation (KFOLD), Leave-One-Out Cross-Validation (LOOCV), and Stratified Cross-Validation (SKFOLD). Fig. 6 illustrates the comparative efficacy of several testing methodologies for test and training accuracies. K-Fold Cross-validation partitions the dataset into K equal or nearly equal segments (folds) Kohavi (1995). The SVM is trained on K-1 folds before testing on the remaining folds. Repeat this method for each fold. The K-Fold class is configured with the following parameters: `n_splits=10`, `shuffle=True`, and `random_state=41`. The mean testing accuracy is 0.9820, while the mean training accuracy is 0.9907, utilizing the K-Fold approach. Subsequently, the SKFOLD was implemented using Class `StratifiedFold` with parameters `n_splits=10`, `shuffle=True`, and `random_state=41`. The dataset comprises K folds, similar to K-fold. However, the partitioning process ensures that the class distribution in each fold matches the overall class distribution of the dataset efValverdeAlbacete2023. The mean testing accuracy is 0.9824, and the mean training accuracy is 0.9927 using the SKFOLD method. The mean testing accuracy is 0.9793, while the mean training accuracy is 0.9905, utilizing the SKFOLD method. Finally, the class `LeaveOneOut` implements LOOCV, which trains n times by utilizing one data point as the test set and n-1 samples as the training set. The mean testing accuracy is 0.9937, whereas the mean training accuracy is 0.9821. In this procedure, 167 training samples and 1 testing sample were in each set.

4.4. Validation with KDE

To do generalized testing on the trained SVM Model, the KDE non-parametric method is applied Silverman (1986) Bowen et al. (2021).

Standardized numeric characteristics with class-conditional KDE are used to generate new, realistic, and human-like samples, where both classes of the target feature FHLM are balanced. Three different groups were made, and the Kolmogorov–Smirnov (KS) test found important features ($p < 0.05$) in each group created by KDE (Table 2). It used to compare the distribution of a feature in the original dataset with the same feature in the KDE-generated synthetic humanized test sets. The distribution of each numeric feature is modeled by using a Gaussian KDE with a fixed bandwidth for each class label (FHLM = "No" and FHLM = "Yes"). Each feature had class-specific values extracted and a KDE model fitted to capture its empirical distribution. It sampled new synthetic values for each feature after training. Three balanced subsets

Algorithm 2 KDE-Based Class-Conditional Data Generation.

Require: Scaled numeric features X_{num}, categorical features X_{cat}, class labels y, samples per class n
Ensure: Synthetic dataset $D_{\text{synthetic}}$ with $2n$ samples
Initialize KDE model dictionary: KDE$[c][f]$ for $c \in \{0, 1\}$ and each numeric feature f
for each class $c \in \{0, 1\}$ **do**
  for each numeric feature f in X_{num} **do**
    $V \leftarrow$ all values of f where $y = c$
    Fit KDE model: KDE$[c][f] \leftarrow$ KDE_Fit(V)
  end for
end for
for each class $c \in \{0, 1\}$ **do**
  Initialize empty table T_{num} for numeric features
  for each numeric feature f **do**
    $T_{\text{num}}[f] \leftarrow$ KDE$[c][f]$.sample(n)
  end for
  $T_{\text{cat}} \leftarrow$ randomly sampled n rows from X_{cat} where $y = c$
  $T \leftarrow T_{\text{num}} \cup T_{\text{cat}}$
  Append target column T[FHLM] $\leftarrow c$
  Store T as D_c
end for
$D_{\text{synthetic}} \leftarrow D_0 \cup D_1$
Shuffle rows of $D_{\text{synthetic}}$
return $D_{\text{synthetic}}$

were created from 15 synthetic samples per feature, each class. A total of 90 realistic samples have been created. Categorical features were sampled without replacement from real class records to improve realism and structural consistency. After combining synthetic numeric and categorical characteristics, the class label was assigned. Finally, class-specific synthetic samples were combined and randomized to decrease ordering bias. The dataset has synthetic, statistically coherent samples that fit the distribution and preserve feature-class connections.

4.5. Hyperparameter tuning

The GSCV approach suggested the optimal parameters required for the SVM classification task. For this reason, the `GridSearchCV` class has been used in `scikit-learn`. Table 3 shows the best parameters for the SVM algorithm. The initial regularization parameter, denoted as `C=0.03`, was selected to be very tiny to get stronger regularization, and it targets a bigger margin for generalization. The second parameter, `class_weight=None`, is set because of the balanced class with `SMOTE-NC`. To utilize the linear kernel, a third kernel parameter was chosen: `kernel=linear`. For the linear kernel, it is assumed that the data is linearly separable and the decision boundary is straight. The fourth parameter, `tol=0.0001`, represents the tolerance level used to prevent overfitting. It helps to balance the accuracy and computing efficiency of the given dataset. The optimization can be stopped with high precision, based on the minimum improvement needed between iterations. The optimization procedure was conducted with the fifth parameter, `shrinking=True`, to ensure faster convergence and more efficient training. It also enhances computational efficiency and accelerates optimization. The final sixth parameter used was `probability=True`, which was employed to evaluate the likelihood of the "Yes" and "No" classes.

4.6. Support vector machine

SVM is a supervised machine learning method usually used for classification. It finds the best hyperplane to classify data. SVM works well in complex or high-dimensional class borders by maximizing the margin between data points of distinct classes. This paper employs a linear SVM classifier to ascertain whether a student's response facilitates hybrid learning. The future hybrid learning mode was the target class.

Vapnik developed the kernel-based SVM model, known for its strong discrimination and generalization. It works by maximizing the margin

Table 2
KS Test Results for KDE Realistic Sets.

Feature	K-Stat.	Set-A p < 0.05	K-Stat.	Set-B p < 0.05	K-Stat.	Set-C p < 0.05
ALP	0.43333	0.00020	0.32828	0.01034	0.39495	0.00097
ATL	0.47879	0.00002	0.54545	0.00000	0.54545	0.00000
Age	0.40000	0.00080	0.46667	0.00004	0.40000	0.00080
CAISOBHI	0.30000	0.02450	0.34444	0.00609	–	–
CGT	–	–	–	–	0.36970	0.00251
Course	0.36667	0.00280	0.35354	0.00446	0.48687	0.00002
FH	0.41010	0.00053	0.47677	0.00003	0.47677	0.00003
FTFHL	0.35657	0.00401	0.43333	0.00020	0.43333	0.00020
Gender	0.46667	0.00004	0.43333	0.00020	0.40101	0.00076
IWRSE	0.36667	0.00280	0.43333	0.00020	–	–
LRS	0.49394	0.00001	0.39394	0.00101	0.39394	0.00101
LSEC	–	–	0.34949	0.00512	0.30606	0.02047
OHLM	0.32626	0.01101	0.40000	0.00080	0.32626	0.01101
OLOTAW	0.44848	0.00010	0.46667	0.00004	0.51515	0.00000
OLOWS	–	–	–	–	0.29293	0.02997
OS	0.66869	0.00000	0.66869	0.00000	0.50202	0.00001
OTOLAW	0.39192	0.00109	0.30000	0.02450	0.36667	0.00280
RQIWT	–	–	0.28788	0.03466	–	–
TCUM	0.28182	0.04087	–	–	–	–
TOMVCD	0.34848	0.00531	–	–	–	–
WMPP	0.45152	0.00009	0.51818	0.00000	0.65152	0.00000

Table 3
Tuning Parameters.

Used Model	Tuned Parameters
SVM	C = 0.03, class_weight = None, kernel = 'linear', probability = True, shrinking = True, tol = 0.0001

between classes for decision-making Cortes and Vapnik (1995). SVM employs a hyperplane, also called the decision boundary, to separate data within a two-dimensional feature space created by mapping labeled inputs. The determination of this hyperplane for hybrid learning classification relies heavily on the optimal margin, which is defined by the closest support vectors. SVMs use training data to maximize performance in classifying patterns apart from input-output pairings, unlike conventional classification, models Sevindi (2020), Hinds and Joinson (2018).

$$f(\mathbf{x}) = \mathbf{x}^T\mathbf{w} + b = \sum_{i=1}^{n} x_i w_i + b = 0. \tag{2}$$

The equation (2) represents the decision function for a Support Vector Machine (SVM) with a linear kernel, defining the hyperplane in the feature space. Here, $\mathbf{x}$ is the input feature vector, $\mathbf{w}$ is the weight vector, b is the bias term, and x_i are the individual feature values. This equation determines the decision boundary where the classifier assigns an output of 0, separating the two classes' 'Yes" and' 'No." The dot product $\mathbf{x}^T\mathbf{w}$ calculates the weighted sum of features, while the bias b shifts the boundary.

Equation (3) is derived after dividing Equation (2) by $||\mathbf{w}||$ Kumar et al. (2021).

$$\frac{\mathbf{x}^T\mathbf{w}}{||\mathbf{w}||} = P_{\mathbf{w}}(\mathbf{x}) = -\frac{b}{||\mathbf{w}||}. \tag{3}$$

The projection of any point $\mathbf{x}$ onto the vector $\mathbf{w}$ along the plane is always given by $-\frac{b}{\|\mathbf{w}\|}$, where $\mathbf{w}$ represents the normal direction of the plane, and $\frac{|b|}{\|\mathbf{w}\|}$ is the distance from the origin to the plane. It is important to note that the equation of the hyperplane is not unique; $c\,f(\mathbf{x}) = 0$ represents the same plane for any constant c. The hyperplane divides the n-dimensional space into two regions. Specifically, the mapping function is defined as $y = \text{sign}(f(\mathbf{x})) \in \{1, -1\}$.

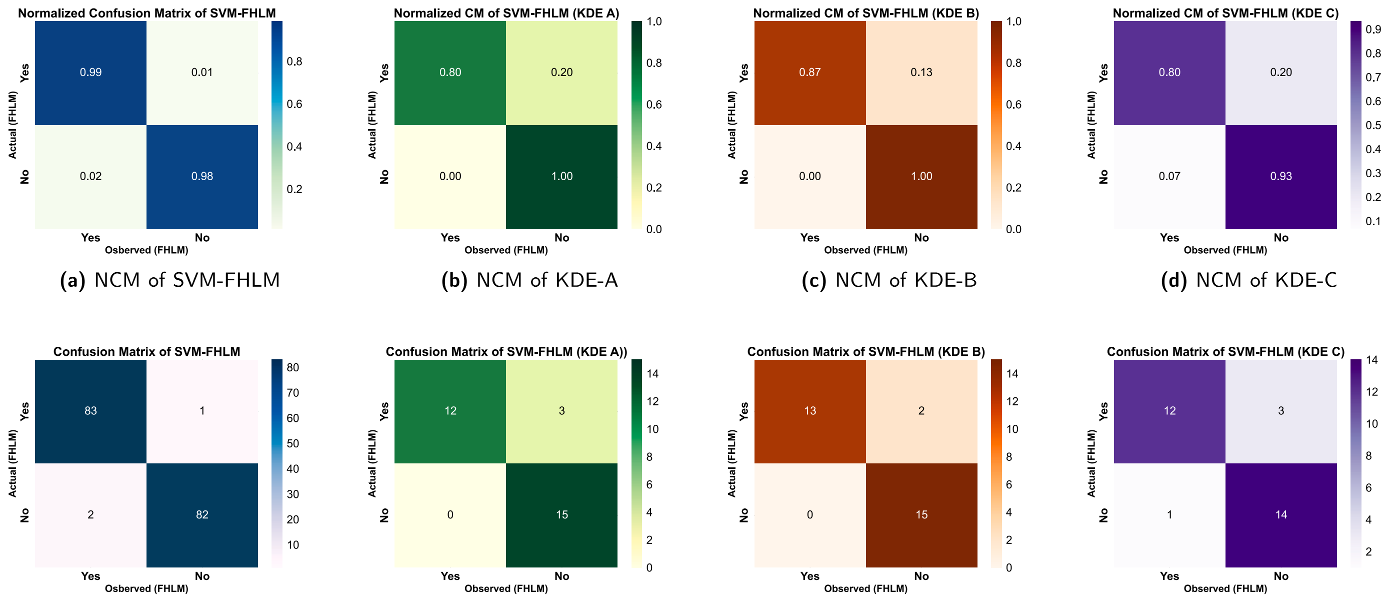
$$f(\mathbf{x}) = \mathbf{x}^T\mathbf{w} + b = \begin{cases} > 0, & y = sign(f(\mathbf{x})) = 1,\ \mathbf{x} \in P \\ < 0, & y = sign(f(\mathbf{x})) = -1,\ \mathbf{x} \in N. \end{cases} \tag{4}$$

In equation (4), $f(\mathbf{x}) = \mathbf{x}^T\mathbf{w} + b$ represents a linear classifier, where $\mathbf{x}$ is the input feature vector, $\mathbf{w}$ is the weight vector, and b is the bias term. The predicted label y is determined by the sign of the decision function $f(\mathbf{x})$, such that $y = \text{sign}(f(\mathbf{x}))$. If $f(\mathbf{x}) > 0$, the sample is classified as belonging to the positive class ($y = 1$), and if $f(\mathbf{x}) < 0$, it is classified as belonging to the negative class ($y = -1$). The decision boundary separates the two classes, with samples in class P having $f(\mathbf{x}) > 0$ and samples in class N having $f(\mathbf{x}) < 0$.

5. Result

5.1. Confusion matrix

Fig. 7 (a) presents a comparative analysis of both normalized and raw count confusion matrices for the base SVM-FHLM model and its three KDE-based variants: KDE-A, KDE-B, and KDE-C. Fig. 7 (a) displays the normalized confusion matrix, which visually represents the relative performance by considering actual and predicted values' proportions. The forecast of "Yes" achieved a perfect score of 0.98, whereas the prediction of "No" scored 0.99, which is a significant result in terms of classification. Fig. 7 (b) displays a fundamental confusion matrix that employs the actual number of records to calculate cross-validation accuracy. If the answer is "Yes" then 82 out of 168 records were successfully identified. 83 records were correctly categorized as "No" and 3 were misclassified. Further, the normalized confusion matrix of SVM-FHLM shows strong class separation, with 99% accuracy for 'Yes' and 98% for 'No', indicating a reliable baseline model. Fig. 7 (b) to Fig. 7 (d) reveal that all KDE-augmented models also perform well, with KDE-B achieving particularly high class-wise accuracy—87% for 'No' and a perfect 100% for 'Yes'—highlighting its balanced classification capability. Raw count confusion matrices in Fig. 7 (e) to Fig. 7 (h) further confirm the consistency of these findings. SVM-FHLM correctly classifies over 80 instances per class, while KDE-B maintains strong performance across unseen test datasets of 90 samples. KDE-A and KDE-C also show respectable results, but on smaller scales, suggesting differences in evaluation data. Overall, SVM-FHLM performed well on the three unseen datasets with KDE-based



(a) NCM of SVM-FHLM (b) NCM of KDE-A (c) NCM of KDE-B (d) NCM of KDE-C

(e) SCM of SVM-FHLM (f) SCM of KDE-A (g) SCM of KDE-B (h) SCM of KDE-C

Fig. 7. Normalized (NCM) and Simple (SCM) confusion matrices across models.

enhancements offering potential benefits in scenarios where density estimation refines class boundaries.

5.2. Classification report

Table 4 shows the classification report produced using `classification report()` function from the `klearn.metrics` module of the `scikit-learn` library. The precision for a class "No" is 0.98, indicating that 98% of occurrences labeled as "No" are true "No". The class "Yes" has a value of 0.99, indicating that 99% occurrences categorized as class "Yes" are indeed class "Yes". The recall for the class "No" is 0.99, indicating that all instances of the actual class "No" are perfectly classified, resulting in a 99% accuracy rate. The class "Yes" has an accuracy of 0.98, indicating that 98% of the instances belonging to the "Yes" class are correctly identified. The F1-score, the harmonic average of recall and precision, is 0.98 for both "No" and "Yes" classes. It gives a balanced measure of both recall and precision. A high accuracy of 0.98 indicates significant SVM model performance. Similar macro and weighted averages indicate a balanced dataset with comparable support for both classifications. Support for classes "No" and "Yes" is 84, indicating 84 instances in the dataset. The SVM-FHLM model performs well on KDE Set A despite generated noise. Class "No" had perfect precision (1.00) but somewhat lower recall (0.80), while class "Yes" scored 0.83 precision and perfect recall. Overall accuracy is 0.90, with macro average and AUC of 0.90 and 0.9733. Under noisy conditions, the model remains reliable but prioritizes true identification above false positive reduction. The model generalizes best in KDE Set B. Class "No" had perfect precision (1.00) and recall (0.87), whereas class "Yes" had 0.88 and perfect recall. AUC returns to 1.0000, matching the original model's discriminative power, and the accuracy is 0.93. This shows KDE Set B matches the training data distribution, making it easier for the model. The accuracy of KDE Set C reduces to 0.87, but the model still performs well. Class "No" has 0.92 precision and 0.80 recall, while class "Yes" has 0.82 and 0.93. The AUC remains high at 0.9822, indicating that the classifier can detect classes even in this more diversified or unseen data.

5.3. Receiver operating characteristic, gain curve, learning curve

Binary classification models are evaluated using the Receiver Operating Characteristic (ROC) curve, a plot that depicts the trade-off between the True Positive Rate (TPR) and the False Positive Rate (FPR) at different cutoffs. Fig. 8(a) depicts the comparative efficacy of the SVM-FHLM model alongside its KDE test sets with the help of ROC curves, which compare true and false positive rates at various threshold levels, separating Yes" and No" classes. The SVM-FHLM and KDE Set B attained flawless classification with an AUC of 1.00, exhibiting no false positives and ensuring total sensitivity. KDE Set C achieved an AUC of 0.98, indicating exceptional classification confidence despite challenging test conditions. KDE Set A demonstrated consistent performance, achieving an AUC of 0.97, indicative of a high degree of model correctness. Therefore, both realistic test samples proved the model's robustness and generalization power.

Fig. 8(b) shows that the SVM model achieved a maximum true positive rate of 99% within the range of 45%–50% of the data samples. The gain chart shows a significant increase above the reference line, followed by a leveling off. It underscores the models' predictive efficacy by juxtaposing the cumulative percentage of accurately predicted positives with the percentage of the sample population. The trained SVM-FHLM and KDE sets demonstrated a significant improvement over the random baseline, with KDE Sets B and C displaying especially pronounced increases, suggesting that a minor subset of data contributes to a substantial fraction of true positives. Although KDE Set A was slightly less effective in terms of gain compared to the other sets, it still demonstrated robust predictive capability. These results show that the SVM-FHLM model is very good at applying what it learned to new situations, and that using KDE in test scenarios helps maintain performance and proves the model can correctly identify true positives even when the data is disturbed or limited.

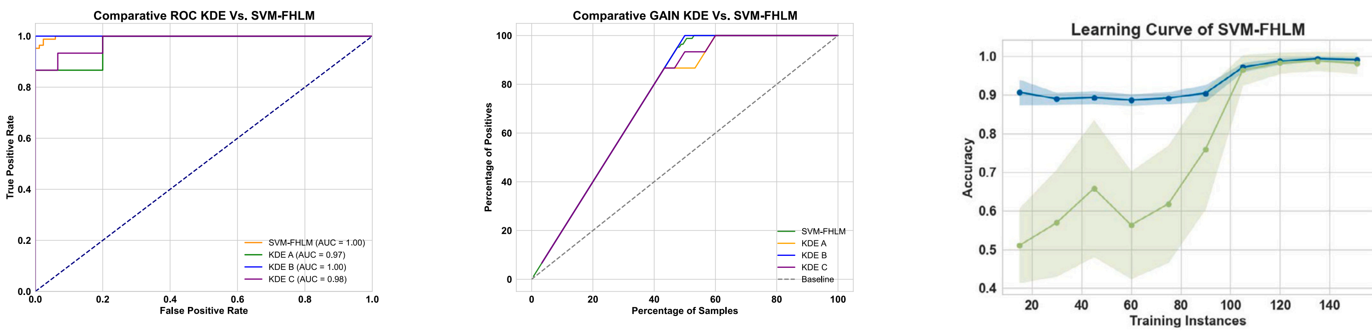
Fig. 8(c) shows the test and train accuracy learning curve with respect to the number of instances in the dataset. With 135 samples, the training accuracy is 97%, and the test accuracy is 99%. As the samples increase to 140, the accuracies of both models become increasingly similar. There appears to be no discrepancy between the SVM model's test and training accuracy. Therefore, the SVM model has been optimized to achieve maximum accuracy by using a sufficient amount of data and preventing overfitting.

5.4. SVM's feature coefficient

Fig. 9 displays the calculated coefficients of the decision function used by the linear kernel for classification. Ten features have negative

Table 4
Comparing Classification Report of SVM-FHLM and KDE test sets (A, B, C).

Model	Class	Precision	Recall	F1-Score	Support	Accuracy	Macro Avg	AUC
SVM-FHLM (CV)	No	0.98	0.99	0.98	84	0.98	0.98	1.0000
	Yes	0.99	0.98	0.98	84			
KDE Set A	No	1.00	0.80	0.89	15	0.90	0.90	0.9733
	Yes	0.83	1.00	0.91	15			
KDE Set B	No	1.00	0.87	0.93	15	0.93	0.93	1.0000
	Yes	0.88	1.00	0.94	15			
KDE Set C	No	0.92	0.80	0.86	15	0.87	0.87	0.9822
	Yes	0.82	0.93	0.88	15			



(a) FPR Vs TPR (b) Sample Vs Positive Instances (c) Learning curve

Fig. 8. Performance Curve: ROC curve, Gain chart, and Learning curve.

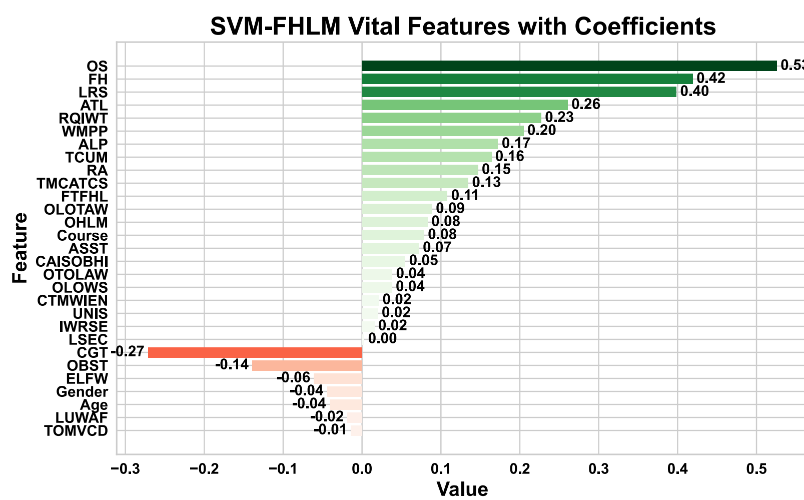


Fig. 9. SVM Coefficient of Feature.

coefficients, while eighteen have positive ones. These weights determine the decision boundary in the feature space. Red indicates negative coefficients, whereas green represents positive ones. The SVM model identified three hybrid learning influencer features—OS, FH, and LRS—with positive coefficients of 0.78, 0.49, and 0.47, respectively. Conversely, the features CGT, OBST, and LUWAF have negative coefficients of -0.42, -0.16, and -0.15, respectively. These features significantly influenced the classification of hybrid learning predictions for future study modes.

Table 5 compares the classification performance of the SVM-FHLM model, evaluated using 10-fold cross-validation, against three KDE-enhanced datasets: Set A, Set B, and Set C. The base model demonstrated strong predictive capabilities, with an average accuracy of 97.61% ± 1.13% (95% CI: 95.24%–99.40%), high precision (97.66% ± 1.67%), recall (97.56% ± 1.68%), and AUC (99.89% ± 0.09%). These results highlight the model's robustness when trained on clean, balanced data.

When evaluated on KDE-augmented test sets—designed to simulate real-world uncertainty through controlled noise—the model maintained solid performance. Set A achieved 90.22% ± 5.41% accuracy, 100% recall, and an AUC of 97.43% ± 2.39%, indicating the model's ability to identify all positive cases in noisy conditions. Set B further demonstrated robustness, with 93.12% ± 4.60% accuracy and perfect recall and AUC (100.00%), showcasing the model's generalizability across diverse data scenarios. While Set C exhibited lower accuracy and specificity, it still achieved a high AUC (98.27% ± 1.90%) and strong recall (93.30% ± 6.66%), reflecting the model's ability to detect true positives under increased noise. These results suggest that KDE-enhanced datasets serve as effective stress tests for assessing model stability. The SVM-FHLM model consistently retained high classification fidelity in perturbed settings. Its AUC remained stable across all KDE sets, underscoring its reliable discriminative capacity and suitability for real-world applica-

Table 5
Comparison of evaluation metrics of SVM-FHLM model with KDE test sets at 95% confidence intervals (CI) with dispersion (±).

Metric	SVM-FHLM (CV)	KDE Set A	KDE Set B	KDE Set C
Accuracy	0.9761 ± 0.0113 (95% CI: 0.9524–0.9940)	0.9022 ± 0.0541 (95% CI: 0.8000–1.0000)	0.9312 ± 0.0460 (95% CI: 0.8333–1.0000)	0.8681 ± 0.0639 (95% CI: 0.7333–0.9667)
Precision	0.9766 ± 0.0167 (95% CI: 0.9405–1.0000)	0.8364 ± 0.0882 (95% CI: 0.6471–1.0000)	0.8791 ± 0.0789 (95% CI: 0.7141–1.0000)	0.8255 ± 0.0931 (95% CI: 0.6364–1.0000)
Recall	0.9756 ± 0.0168 (95% CI: 0.9383–1.0000)	1.0000 ± 0.0000 (95% CI: 1.0000–1.0000)	1.0000 ± 0.0000 (95% CI: 1.0000–1.0000)	0.9330 ± 0.0661 (95% CI: 0.7692–1.0000)
Specificity	0.9766 ± 0.0166 (95% CI: 0.9405–1.0000)	0.8034 ± 0.1039 (95% CI: 0.5881–1.0000)	0.8625 ± 0.0898 (95% CI: 0.6471–1.0000)	0.8032 ± 0.1058 (95% CI: 0.5788–1.0000)
F1 Score	0.9759 ± 0.0115 (95% CI: 0.9503–0.9945)	0.9083 ± 0.0542 (95% CI: 0.7857–1.0000)	0.9337 ± 0.0462 (95% CI: 0.8332–1.0000)	0.8727 ± 0.0658 (95% CI: 0.7142–0.9744)
Kappa	0.9519 ± 0.0227 (95% CI: 0.9042–0.9881)	0.7994 ± 0.1073 (95% CI: 0.5867–1.0000)	0.8590 ± 0.0928 (95% CI: 0.6441–1.0000)	0.7305 ± 0.1288 (95% CI: 0.4495–0.9333)
MCC	0.9522 ± 0.0225 (95% CI: 0.9047–0.9882)	0.8189 ± 0.0912 (95% CI: 0.6443–1.0000)	0.8704 ± 0.0814 (95% CI: 0.6892–1.0000)	0.7422 ± 0.1238 (95% CI: 0.4708–0.9354)
G-Mean	0.9760 ± 0.0113 (95% CI: 0.9518–0.9946)	0.8944 ± 0.0591 (95% CI: 0.7669–1.0000)	0.9274 ± 0.0495 (95% CI: 0.8044–1.0000)	0.8633 ± 0.0675 (95% CI: 0.7119–0.9733)
AUC	0.9989 ± 0.0009 (95% CI: 0.9964–1.0000)	0.9743 ± 0.0239 (95% CI: 0.9138–1.0000)	1.0000 ± 0.0000 (95% CI: 1.0000–1.0000)	0.9827 ± 0.0190 (95% CI: 0.9321–1.0000)

tions involving unseen data. The model achieved the highest accuracy (97.60% ± 1.13%), followed by KDE Set B (92.74% ± 4.10%) and Set A (89.44% ± 6.44%). G-Mean values remained stable across models, with Set C (86.33% ± 6.75%) maintaining a solid balance between sensitivity and specificity. The SVM-FHLM model also showed the highest agreement between predicted and actual labels ($\kappa = 0.9519 \pm 0.0227$). Among KDE sets, Set B led with $\kappa = 0.8580 \pm 0.0928$, followed by Set A (0.7994 ± 0.1073) and Set C (0.7305 ± 0.1288), indicating strong reliability despite noise. F1 scores further confirmed balanced performance: the SVM-FHLM model achieved 97.59% ± 1.15%, with Sets B (93.37% ± 4.05%) and A (90.36% ± 8.54%) remaining consistent, and Set C showing a reasonable F1 score of 87.27% ± 6.58%.

6. SVM explainability with SHAP

SHAP provides a unified framework for interpreting predictions by assigning a relevance score to each feature for a given prediction Lundberg and Lee (2017). This approach analyzes individual predictions and indicates whether each feature contributes to an increase or decrease in the predicted value.

By applying classic Shapley values from game theory, SHAP connects local explanations to fair and efficient credit allocation SHAP (2024). The `SHAP` library includes the `shap_values` function. The `shap.KernelExplainer` class utilizes this method to estimate SHAP values for test data.

$$\phi_i(SVM; x) = \sum_{z_0 \subseteq x_0} \frac{|z_0|!(M - |z_0| - 1)!}{M!} \times \left[SVM_x(z_0) - SVM_x(z_0 \setminus i)\right] \quad (5)$$

Equation (5) illustrates the SHAP value estimation. Here, $\phi_i(\mathrm{SVM}; x)$ denotes the SHAP score for attribute i in the SVM model, where x is the input value of that attribute. The term $\sum_{z_0 \subseteq x_0}$ indicates that z_0 is a subset of the feature set x_0. The expression $|z_0|!$ is the factorial of the subset size, $(M - |z_0| - 1)!$ is the factorial of the complement size, and $M!$ is the factorial of the total number of attributes. $SVM_x(z_0)$ represents the SVM's prediction when only the attributes in the subset z_0 are used, while $SVM_x(z_0 \setminus i)$ denotes the prediction when attribute i is excluded from the subset.

Fig. 10(a) presents a summary plot that compares the influence of each feature on both classes, revealing positive and negative associations with the target class. The x-axis shows log-odds values, and the y-axis lists features in descending order of importance. The vertical bar at the center separates the "Yes" class (right side) from the "No" class (left side). Colored dots represent individual data points, with red indicating higher logit values and blue indicating lower values, as shown by the color scale on the right. The feature `RQIWT` exhibits a strong influence on class "No" and a weaker influence on class "Yes." Three features—`OS`, `FH`, and `LRS`—show a more significant impact on class "Yes" than on class "No." Lower-ranking features such as `OLOWS`, `TOMVCD`, and `Age` contribute less to the distinction between classes. Thus, attributes related to Overall Satisfaction, happiness, and longer solutions appear organized and beneficial in the hybrid learning context.

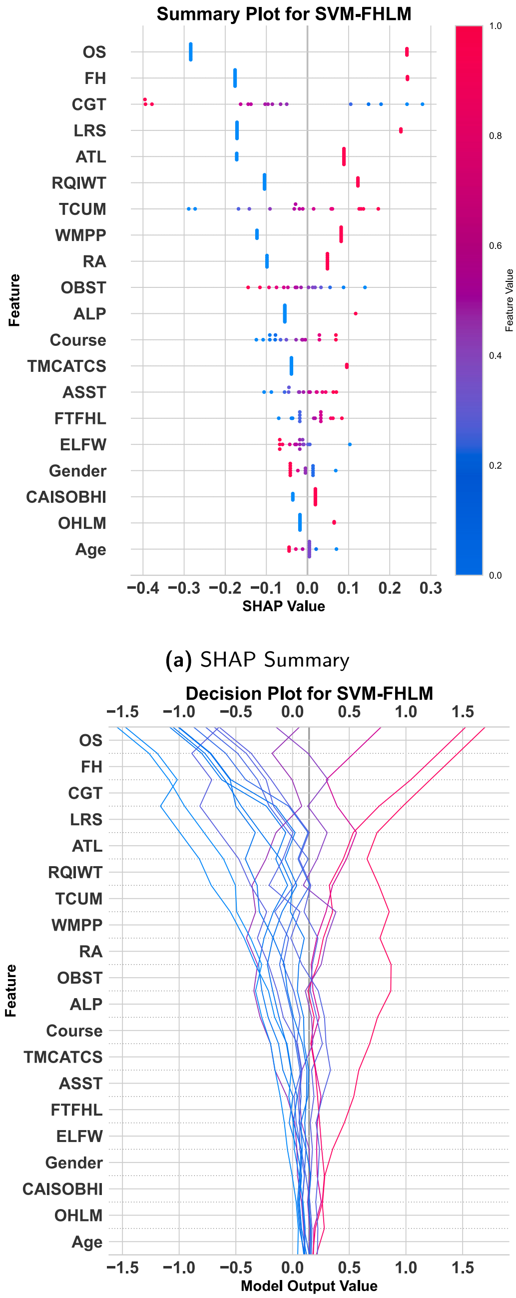
Fig. 10(b) displays the decision plot, with log-odds values on the top x-axis and feature names on the y-axis. Colored lines represent the prediction paths of individual observations, converging at their predicted values on the bottom x-axis. As the plot progresses, SHAP values for each feature are added to the model's base value. The features `FH`, `OS`, `LRS`, `CGT`, and `ATL` are shown to significantly influence the prediction.

The most important feature in OS (SHAP: 0.2561) in Fig. 11 (a), which relates to the training data, is followed by CGT (0.2213), FH (0.2065), LSR (0.2024), and ATL (0.1498). These qualities mostly concern measures of academic performance, hence supporting their importance in predicting FHLM classification results. By comparison, the least important characteristics were IWRS (0.0078), TMCATCS (0.0065), and UNIS (0.0061), which probably had little effect on the output of the model because of low variance or weak correlation with the goal.

Fig. 11 (b) shows SHAP scores of KDE set A. With OS (0.2744), CGT (0.2599), FH (0.2346), LSR (0.2021), and ATL (0.1505) exhibiting significant influence, the top five traits stay mostly stable with training. These numbers show that even with KDE-generated noise, the model still prioritizes the same significant factors. Closely reflecting their small part in the training phase, the least influential variables are IWRS (0.0109), TMCATCS (0.0099), and UNIS (0.0071).

The trend of prevailing features in Fig. 11 (c), which depicts KDE Set B, continues with OS (0.2972), CGT (0.2721), LSR (0.2305), FH (0.1862), and FTFHL (0.1560) placing at the top. Especially in this group, the SHAP values are somewhat higher, indicating more pronounced feature effects likely resulting from modified KDE distributions. Once more, the bottom three features—IWRS (0.0134), TMCATCS (0.0112), and UNIS (0.0082)—have little effect, implying consistent model behavior in de-emphasizing weaker predictors.

Fig. 11 (d) shows the SHAP study for KDE Set C. Leading features here are FH (0.2407), CGT (0.2399), LSR (0.2327), OS (0.2309), and ATL (0.1953), which emphasize the consistency of feature ranking across all test situations. The model keeps its emphasis on the fundamental academic features despite noise. SHAP value is consistently low for the bottom features—IWRS (0.0078), TMCATCS (0.0065), and UNIS (0.0064).



(a) SHAP Summary

(b) SHAP Decision

Fig. 10. (a) SHAP Summary (b) SHAP Decision.

Hence, SHAP analysis across training and KDE test sets shows that the model relies on realistic unseen samples, an interpretable subset of characteristics with good predictive value. The stability and interpretability of the SVM-FHLM model in multiple data settings are confirmed by the robust ranking of the importance of features.

Top predictors included `Overall Satisfaction (OS)` and `Feeling Happiness (FH)`, which state that emotional well-being is essential to learning. Student persistence and performance improve when they feel supported, engaged, and content. `Long-Term Solution (LRS)` importance also shows students' rising confidence that hybrid learning is a sustainable, even preferable, style of education beyond pandemics. `Challenging Group Tasks (CGT)`, `Reduction in Quality of Interaction with Teachers (RQIWT)`, and `Less Interaction with Other Faculty (LUWAF)` show that hybrid learning still struggles to promote collaboration and meaningful teacher-student discourse. These gaps propose rethinking group activities and how educators might be present in physical and digital places. The high ranking of `Appropriateness for Theory Lectures (ATL)` shows that students value hybrid delivery for specific subjects. The weight provided to `University Assistance (UIS)` and `University Initiative and Support (UIS)` emphasizes how much learners depend on institutional mechanisms to navigate these hindrances. Therefore, SVM-FHLM features don't merely inform a model; they highlight where pedagogy, technology, and student needs converge, giving educators a clearer route forward.

7. Discussion

In hybrid education, `OS` and `FH` highlight the importance of student motivation and emotional engagement. According to Self-Determination Theory Deci and Ryan (1985), motivation and learning outcomes are enhanced when autonomy, competence, and relatedness are supported. The flexibility of hybrid learning, which allows students to engage at their own pace and within their own environment, may foster intrinsic motivation. Furthermore, the perception of hybrid learning as a `LRS` influences students' views on the sustainability and adaptability of learning environments. Constructivist learning theories assert that learning is an active, contextualized process shaped by students' experiences and social interactions Vygotsky (1978); Bruner (1960).

The feature `CGT` highlights that collaboration becomes difficult in the absence of face-to-face interaction. This underscores how hybrid learning environments may hinder collaborative learning pedagogies, which depend on peer interaction, shared responsibility, and group cohesion—elements more challenging to sustain in digital formats Johnson and Johnson (1999). Results indicating a `RQIWT` and `LUWAF` further demonstrate the breakdown of relational and dialogic teaching and learning practices. These practices are foundational to the Community of Inquiry framework Garrison et al. (2000) and connectivist learning theory Siemens (2005), both of which emphasize the necessity of sustained teaching and social presence to facilitate meaningful learning in digital or blended settings.

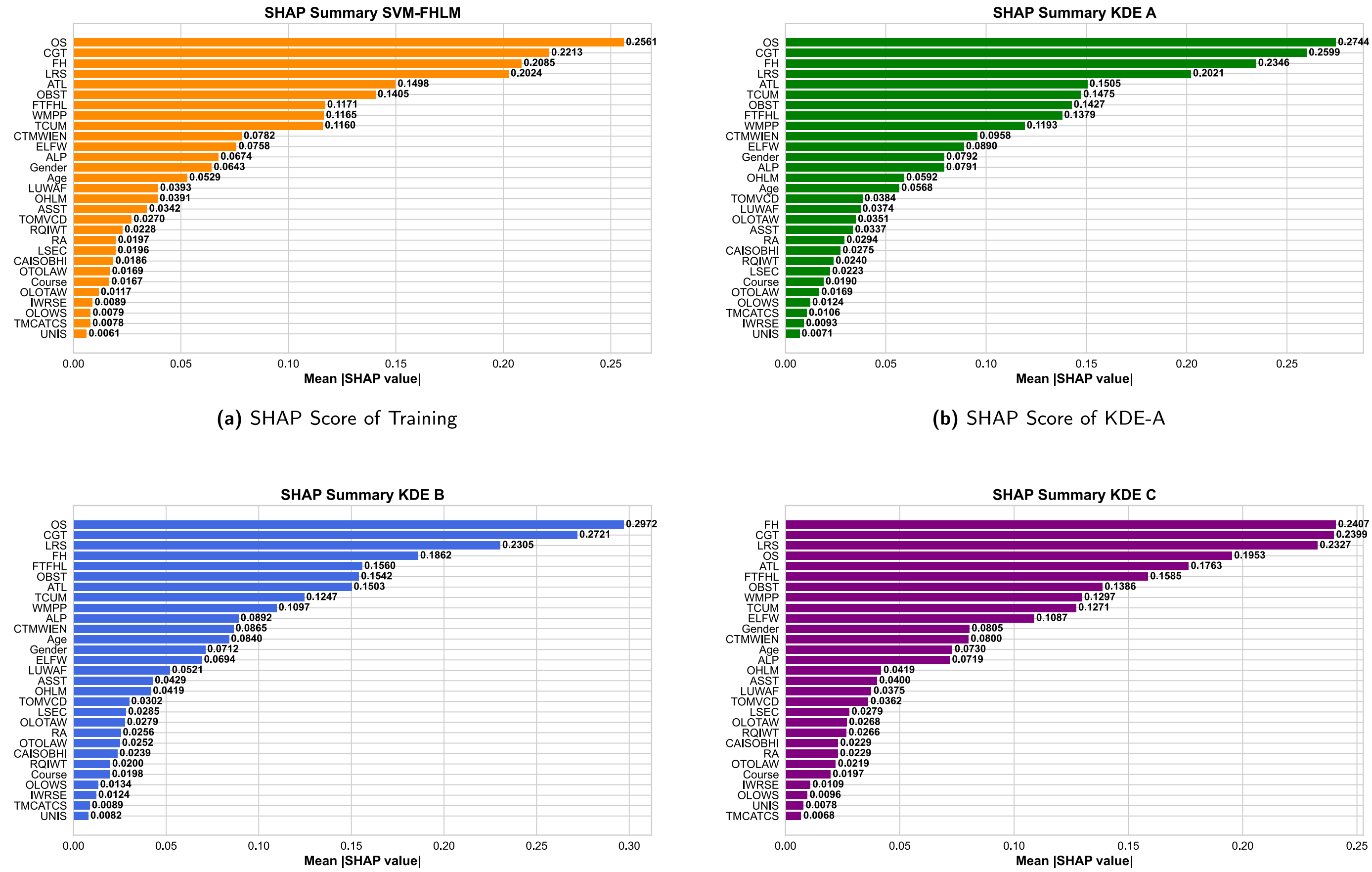
The feature `ATL` as a positive aspect reveals that students favor the hybrid model for theoretical instruction. This aligns with blended learning theory, which advocates combining face-to-face instruction with the flexibility and extended reach of online modalities Garrison and Kanuka (2004). The feature *First Time Facing Hybrid Learning (FTFHL)* introduces a reflective element, suggesting that encountering a new learning format can influence students' beliefs and learning strategies. According to Transformative Learning Theory, disorienting experiences, such as adapting to a hybrid education system, can trigger profound cognitive changes and perspective transformation Mezirow (1991).

The features `University Assistance (AST)` and `University Support (UNIS)` highlight the critical role of institutional infrastructure in fostering student success. Instructional scaffolding plays a central role in educational design by providing necessary support as learners confront new challenges Wood et al. (1976). Furthermore, Tinto's model of student integration and retention underscores the importance of intellectual and social support in promoting student engagement and persistence within higher education environments Tinto (1993).

These findings not only show that the SVM-FHLM AI model works well, but they also highlight how hybrid learning environments connect with important educational ideas. Bringing these ideas into the design of systems and the creation of educational policies could greatly improve the strength of teaching methods and focus on students in future hybrid learning programs.

8. Implication

This study's findings have meaningful implications for the design and support of hybrid learning in higher education. This study's conclu-



(a) SHAP Score of Training

(b) SHAP Score of KDE-A

(c) SHAP Score of KDE-B

(d) SHAP Score of KDE-C

Fig. 11. Comparison of SHAP scores for SVM-FHLM on (a) Training data, and KDE test sets: (b) KDE-A, (c) KDE-B, and (d) KDE-C.

sions affect higher education's hybrid learning design and support. `OS`, `FH`, and `LRS` show students' emotional engagement and perspective of hybrid learning as a feasible and sustainable approach, consistent with self-determination theory and constructivist pedagogy. `CGT` and `RQIWT` reveal weaknesses in collaborative dynamics and instructional presence, pointing out the need for stronger social and teaching presence, as emphasized in the Community of Inquiry framework. Positive reaction to `ATL` enables selective hybrid material delivery in blended learning methodologies. Finally, `UIS` and `AST` emphasize that hybrid student success depends on instructional design and effective support structures, which support instructional scaffolding and student retention. To be relevant and scalable, hybrid learning should be data-informed and based on strong pedagogical theory.

9. Framework configuration and deployment

The web application SVM-FHLM-29 Verma (2025) was built using the `Gradio` Python toolkit to design customizable user interfaces for statistical and machine learning models and different Python functions Gradio (2024). This paper used library `gradio` to access the `Interface`, `Blocks`, `Row`, `Column`, `Radio()`, `Slider` and `Markdown` classes, as well as `launch()` method to create and launch an application interface. Two files, `requirements.txt`, and `app.py`, were uploaded to `Hugging Face` website Face (2024), a popular real-time deployment web space.

Fig. 12 displays the real-time AI web application based on the experiment performed with eight sections. The web application user interface components are constructed using the extensive array of classes provided by the Gradio library. The `Blocks` class is a container for organizing and structuring various components. The `Markdown` class generates text over the components. The `Row` class organizes its items horizontally in a singular row. The `Column(scale=1)` arranges the components vertically in a column. `Radio` class generates a radio button component that enables users to select a single option from a set of predefined choices. The `Slider` class generates a slider component that enables users to select a numerical value within a defined range by manipulating a slider handle. The initial three components utilized the `Radio` class, while the other seven components were developed using the `Slider` class.

The first component takes input about the age, gender, and course level of study. The second component takes six significant inputs about hybrid learning satisfaction. For lab and theory classes, the third component receives three inputs regarding hybrid learning. The fourth component concerns seven challenges faced during hybrid learning. During hybrid learning, the fifth component solicits feedback regarding university support and assistance. The sixth component is about time and cost management in hybrid learning. The seventh component will take input about the benefits of hybrid learning, and the eighth component will display the results after clicking the submit button. The results will be based on the trained SVM model, which displays the classification results as eight instances of "Yes". The standard threshold is less than 0.5 for "No" responses and greater than 0.5 for "Yes" responses. The `classify ()` method processes the model's predicted values and assigns them to a specified class.

10. Conclusion

The present study integrated and explained the SVM model with SHAP to predict hybrid learning in education. Also, based on SVM, a real-time hybrid learning recommendation application has been devel-

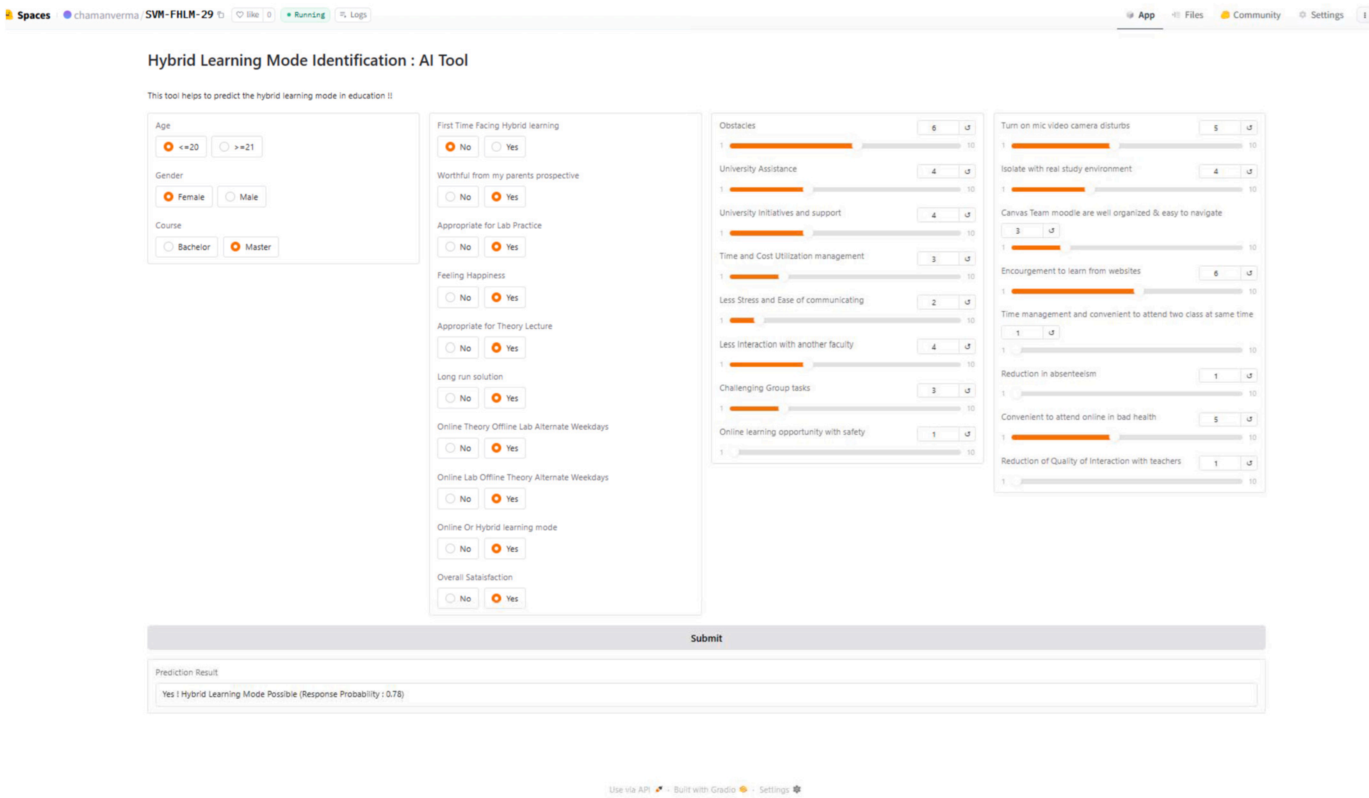


Fig. 12. A Real Time Hybrid Learning Identification Tool.

oped. The primary samples have been considered for it. Test accuracy was observed, and the best parameters were set to train the model using SVM hyperparameter optimization. The model has been trained with 99% accuracy and tested with 98% accuracy. Hence, SVM has been stabilized, and no significant variation has been found in both accuracies.

The trained model has been validated with KDE three test sets of 30 samples each, which also achieved more than 85% accuracy with 95% CI. This process helped make the model more robust and better generalize to unseen data. Additionally, we used the SHAP approach to explain the final trained model further, highlighting the novel features that significantly contributed to the classification task. The SVM model acquired the maximum possible F1 score, precision, and recall values, all of which were 0.98. According to the calculations, the AUC equals 1, which indicates a high value for accurate segregation between “Yes” and “No” responses. According to the SVM coefficient, the attributes that had the most impact were OS, LRS, FH, and OS. These attributes had positive coefficients of 0.78, 0.49, 0.47, and 0.41, respectively. The Challenging Group tasks (CGT) and the Obstacle (OBST) have negative coefficients, with the former being -0.42 and the latter representing -0.16. While the selection was being made, Shapley Additive Explanations (SHAP) explained the strength of the SVM model and voted for `OS`, `FH`, `LRS`, and `CGT`.

The current study used a single institution, a limited sample size, and confined dimension reduction with machine learning techniques. Also, the developed real-time website will be tested with the real responses of students and cross-validated with the current accuracy of the model. If needed, the future studies may incorporate techniques such as Xtream Gradient Boosting, Linear discriminant analysis, random forest, K-nearest neighbor, and Decision Tree with SHAP to improve the explanation. In addition, the following study will focus on improving prediction accuracy and investigating the strengths of attributes. A considerable model can also be deployed as an online Python web application for institutions.

CRediT authorship contribution statement

Chaman Verma: Conceptualization of this study, Objectives, Methodology, Manuscript writing and editing, Experiment, Software.

Declaration of competing interest

The author declare that he has no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This paper is supported by the Digital Education Development Competence Center under the project: 2022-1.1.1-KK-2022-00023, which the National Research, Development and Innovation Office, Hungary, financially supports. The study was approved by an ethical committee with ID: (IK/2831/1(2023). Informed consent was obtained from all participants, and their privacy rights were strictly observed. The data can be obtained by sending request e-mails to the corresponding author.

References

Assegie, T., Salau, A., Chhabra, G., Kaushik, K., & Braide, S. (2024). Evaluation of random forest and support vector machine models in educational data mining. In *2024 2nd international conference on advancement in computation & computer technologies (InCACCT)* (pp. 131–135). India: Gharuan.

Bartlett, M. S. (1937). Properties of sufficiency and statistical tests. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, *160*, 268–282. https://doi.org/10.1098/rspa.1937.0109.

Bowen, C., Liu, C., & Weninger, T. (2021). Generating high-quality synthetic tabular data using kernel density estimation. preprint, arXiv:2108.10936. Available at: https://arxiv.org/abs/2108.10936.

Bruner, J. S. (1960). *The process of education*. Harvard University Press.

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.

Chung, M., & Gao, Y. (2021). Online or in-person course? Meaningful discussion of learning effectiveness and satisfaction of microwave filter design course in graduate students. *IEEE Transactions on Education*, 1–8.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*, 273–297. https://doi.org/10.1007/BF00994018.

Cui, B. (2024). Evaluation of teaching capability of teachers based on two-class support vector machine. In *2024 third international conference on distributed computing and electrical circuits and electronics (ICDCECE)* (pp. 1–4). India: Ballari.

Dataset, H. L. (2022). Hybrid learning. Online. Available: https://forms.gle/wx1N6XewB8oAwTVa8. Verma C.

Deci, E. L., & Ryan, R. M. (1985). *Intrinsic motivation and self-determination in human behavior*. Springer.

Ding, H., & Zhou, Y. (2023). Behavior analysis and learning outcomes prediction model based on blended learning. In *2023 8th international conference on information systems engineering (ICISE), Dalian, China* (pp. 525–529).

Face, H. (2024). Hugging face documentation. https://huggingface.co/. (Accessed 12 December 2024).

Findiana, R., Yuniarno, E. M., & Endroyono (2020). Classification of graduates student on entrance selection public higher education through report card grade path using support vector machine method. In *2020 3rd international conference on information and communications technology (ICOIACT), Yogyakarta, Indonesia* (pp. 7–11).

Garrison, D. R., Anderson, T., & Archer, W. (2000). Critical inquiry in a text-based environment: Computer conferencing in higher education. *The Internet and Higher Education*, *2*, 87–105.

Garrison, D. R., & Kanuka, H. (2004). Blended learning: Uncovering its transformative potential in higher education. *The Internet and Higher Education*, *7*, 95–105.

Gradio (2024). Gradio documentation. https://www.gradio.app/docs. (Accessed 12 December 2024).

Hermitaño-Atencio, B., Ortiz-Vergara, M., Buleje-Agüero, G., Torres-Calixtro, J., & García-Rojas, V. (2023). Smart education and hybrid education in students of the national university of education in times of pandemic and post-pandemic. In *2023 IEEE Colombian conference on communications and computing (COLCOM), Bogota, Colombia* (pp. 1–8).

Hinds, J., & Joinson, A. N. (2018). What demographic attributes do our digital footprints reveal? A systematic review. *PLoS ONE*, *13*, 1–40.

Hussein, N., Aluwi, A., Mohd, I., & Omar, M. (2020). Enhancing students engagement through blended learning satisfaction and lecturer support. In *2016 IEEE 8th international conference on engineering education (ICEED)* (pp. 175–180).

Johnson, D. W., & Johnson, R. T. (1999). *Learning together and alone: Cooperative, competitive, and individualistic learning*. Allyn & Bacon.

Kaiser, H. F. (1970). A second generation little jiffy. *Psychometrika*, *35*, 401–415. https://doi.org/10.1007/BF02291817.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th international joint conference on artificial intelligence, IJCAI conference* (pp. 1137–1145).

Kumar, D., Verma, C., Dahiya, S., Singh, P., Raboaca, M., Illés, Z., & Bakariya, B. (2021). Cardiac diagnostic feature and demographic identification (cdf-di): An iot enabled healthcare framework using machine learning. *Sensors*, *21*, 6584.

Kurniawan, C., & Wahyuni, F. (2021). Sentiment analysis of online learning students feedback for facing new semester: A support vector machine approach. In *2021 7th international conference on education and technology (ICET), Malang, Indonesia* (pp. 1–6).

Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems (NIPS 2017), Long Beach, California, USA* (pp. 4765–4774).

Mezirow, J. (1991). *Transformative dimensions of adult learning*. Jossey-Bass.

Mlitwa, N. W. B., & Ogundaini, O. O. (2022). Determinants of e-learning user satisfaction at a South African university of technology. In *2022 IEEE 28th international conference on engineering, technology and innovation (ICE/ITMC) & 31st international association for management of technology (IAMOT) joint conference* (pp. 1–8).

Paiva, C., Cantarutti, C., & García Huidobro, R. (2021). Implementing and evaluating hybrid classes in a biostatistics course for dentistry. In *2021 machine learning-driven digital technologies for educational innovation workshop* (pp. 1–5).

Purahong, B., Anuwongpinit, T., Sithiyopasakul, S., Archevapanich, T., Sithiyopasakul, P., & Vichaiya, P. (2021). Hybrid learning during the covid-19 pandemic of engineering students at Kmitl, Thailand. In *2021 international STEM education conference (iSTEM-Ed 2021)* (pp. 1–4).

Ramadhani, D., Nasution, F. A., Ariffin, F. B., Bayuputra, M. R. A., Riady, Y., & Oktavia, T. (2023). The impact of hybrid learning on learning quality for higher education institution. In *2023 international conference on information management and technology (ICIMTech), Malang, Indonesia* (pp. 66–71).

Saputri, M., Utami, F., & Sari, D. (2022). The effectiveness of e-learning service quality in influencing e-learning student satisfaction and loyalty at telkom university. In *2022 international conference advancement in data science, E-learning and information systems (ICADEIS)* (pp. 1–5).

Senior, J. (2022). How to build on arab speaking students' positive experiences: Emergency remote learning during the covid-19 pandemic and into the future of learning. In *2022 international conference on business analytics for technology and security (ICBATS)* (pp. 1–6).

Sevindi, T. (2020). Investigation of social appearance anxiety of students of faculty of sport sciences and faculty of education in terms of some variables. *Asian Journal of Education and Training*, *6*, 541–545.

SHAP (2024). Shap documentation. https://shap.readthedocs.io/en/latest/. (Accessed 30 July 2024).

Siemens, G. (2005). Connectivism: A learning theory for the digital age. Retrieved from http://www.elearnspace.org/Articles/connectivism.htm.

Silverman, B. W. (1986). *Density estimation for statistics and data analysis*. London: Chapman and Hall.

Sun, X. (2021). Exploration of general education work in universities based on the feature extraction algorithm. In *2021 2nd international conference on education* (pp. 598–601). Xiamen, China: Knowledge and Information Management (ICEKIM).

Tinto, V. (1993). *Leaving college: Rethinking the causes and cures of student attrition* (2nd ed.). University of Chicago Press.

Trivedi, S., Sharma, A., Patra, P., & Dey, S. (2022). Prediction of intention to use social media in online blended learning using two step hybrid feature selection and improved svm stacked model. *IEEE Transactions on Engineering Management*, *9*, 1–16.

Tubishat, M., Al-Obeidat, F., & Shuhaiber, A. (2023). Sentiment analysis of using chatgpt in education. In *2023 international conference on smart applications, communications and networking (SmartNets), Istanbul, Turkiye* (pp. 1–7).

Venkatesan, N., & Priya, G. (2015). A study of random forest algorithm with implementation using weak. *International Journal of Innovative Research in Computer Science and Engineering*, *1*, 156–162.

Verma, C. (2024). Machine learning model for applicability of hybrid learning in practical laboratory. *Procedia Computer Science*, *235*, 1600–1607.

Verma, C. (2025). Svm-fhlm. https://huggingface.co/spaces/chamanverma/SVM-FHLM-29. (Accessed 7 May 2025).

Verma, C., & Illes, Z. (2023). Predicting student's satisfaction towards hybrid learning in informatics. In *2023 3rd international conference on innovative practices in technology and management (ICIPTM), Uttar Pradesh, India* (pp. 1–6).

Verma, C., Illes, Z., & Kumar, D. (2024). Tclpi: Machine learning-driven framework for hybrid learning mode identification. *IEEE Access*, *12*, 98029–98045.

Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes*. Harvard University Press.

Wood, D., Bruner, J. S., & Ross, G. (1976). The role of tutoring in problem solving. *Journal of Child Psychology and Psychiatry*, *17*, 89–100.