ISSN No:-2456-2165

Performance Comparison of kNN, Random Forest and SVM in the Prediction of Cervical Cancer from Behavioral Risk

Ali Degirmenci
Department of Electrical and Electronics Engineering
Ankara Yıldırım Beyazıt University
Ankara, Turkey

Abstract:- Cervical cancer is one of the most common vital diseases that still seriously affects women worldwide. Early detection of it may not be possible due to late onset of symptoms, community norms, unavailable healthcare facilities, and medical cost. Computer aided diagnostic tools have shown very successful results in the early diagnosis of diseases in recent years. Especially the developments in computer technology have increased the success of machine learning-based methods. This study presents and analyzes 3 different machine learning based algorithms (k nearest neighbor, support vector machines (SVM), and random forest) to predict cervical cancer. Hyperparameter optimization of algorithms is performed by exhaustive grid search and k-fold cross validation is used to increase the reliability of the results. Among the benchmarked methods, the best performance was obtained with the SVM method with the sigmoid kernel, and the accuracy, precision, recall, and F1-score metrics were 0.9274, 0.9093, 0.8410, 0.8565, respectively.

Keywords:- *Cervical Cancer; kNN; SVM; Random Forest; Computer Aided Diagnosisinsert.*

I. INTRODUCTION

Cervical cancer is the fourth most common cancer type among women, and malignancy develops in the woman's cervix in this type of cancer. Common types of cervical cancer are squamous cell carcinoma (70%) and adenocarcinoma (25%). The major cause of cervical cancer is the human papillomavirus (HPV). Smoking, a weak immune system, and a family history of cervical cancer are among the factors that increase the incidence of cervical cancer [1].

As in other types of cancer, early diagnosis is very important in cervical cancer and increases the success of treatment. Regular pap smear test and HPV vaccination are effective in cervical cancer diagnosis and prevention. However, the number of women affected by this disease is still high. The estimated number of women who died of cervical cancer in 2020 is 341,831. According to Cancer Facts and Figures 2022, there are more than 14000 cases of cervical cancer to be diagnosed and 4,280 deaths are expected in the US [2].

The fact that machine learning-based methods yield more successful results with each passing day has paved the way for their application to different areas such as electronic education [3], medical [4, 5], warfare [6], meteorology [7], economy [8], outlier detection [9-11]. Recently, significant progress has also been made in biomedical engineering. Machmud and Wijaya collected the cervical cancer dataset from behavior and applied Naive Bayes and logistic regression methods to predict cervical cancer [12]. According to the results, the naive bayes method achieves 91.67% accuracy, while the logistic regression method gives 87.5% accuracy. Tarakci and Ozkan compared the performance of *k* nearest neighbors (kNN) and weighted kNN (WkNN) in five different data sets (heart failure clinical records, raisin grains, rice cammeo osmancik, breast cancer coimbra, and cervical cancer behavior risk) [13]. In both methods, *k* was chosen as 10 and the Euclidean distance metric was used. In the kNN method, the weights of the samples were determined by calculating the inverse of the distance. Although no significant performance differences were observed in the experiments, kNN performed better on the rice cammeo osmancik, raisin grains data sets while underperforming in the other data sets. Gamara et al. used artificial neural networks in prediction [14]. The data set is divided into train (70%), validation (15%), and test (15%) sections. The performance of the algorithm in training and validation data is 100%, but its performance in test data is 90.9%. Akter et al. applied random forest, decision tree, and eXtreme Gradient Boosting (XGBoost) methods to predict cervical cancer from behavior [15]. Correlation coefficient demonstrating the relationship between multiple variables was calculated for the features. The highest correlation was found to be 0.85 and therefore they decided not to remove any feature in model construction. Feature importance was also made with XGBoost and the top three features that had the larger effects on the model were determined as intention_ aggregation, perception_ vulnerability, and empowerment_ desires. Alphan used logistic regression, J48, Naïve Bayes, Bayesian network, random tree, kNN, random forest, and support vector machines (SVM) methods on the same data set [16]. Analyzes were made with the WEKA tool and SVM method obtained the highest accuracy with 91.67% among the compared methods. Ghanem et al. developed an association rules-based classification method, named SIA, and compared the performance of this method with Naive Bayes, radial basis neural networks, decision tree, J48, and simple CART on the

6 different real-world data sets (Wisconsin breast cancer, cervical cancer behavior risk, Wisconsin prognosis breast cancer, EEG eye state, Wisconsin diagnosis breast cancer, and Haberman's survival) [17]. Ratul used 11 machine learning methods including SVM, kNN, decision tree, CatBoost, AdaBoost, Gaussian Naive Bayes, random forest, gradient boost classifier, multilayer perceptron (MLP), random forest, and XGBoost [18]. Among the compared methods, the best accuracy score was obtained with the MLP method (0.9333). Cicek suggested a Web application made via the Scikit-learn and Dash libraries [19]. This application includes logistic regression, Gaussian Naive Bayes, decision trees, SVM, AdaBoost, XGBoost, random forest, and LightGBM classification methods and allows comparison between them. The results of the random forest method in the cervical cancer behavior data set for accuracy, precision, recall, and F1-score performance metrics were 94.44%, 75.00%, 100%, and 94.44%, respectively.

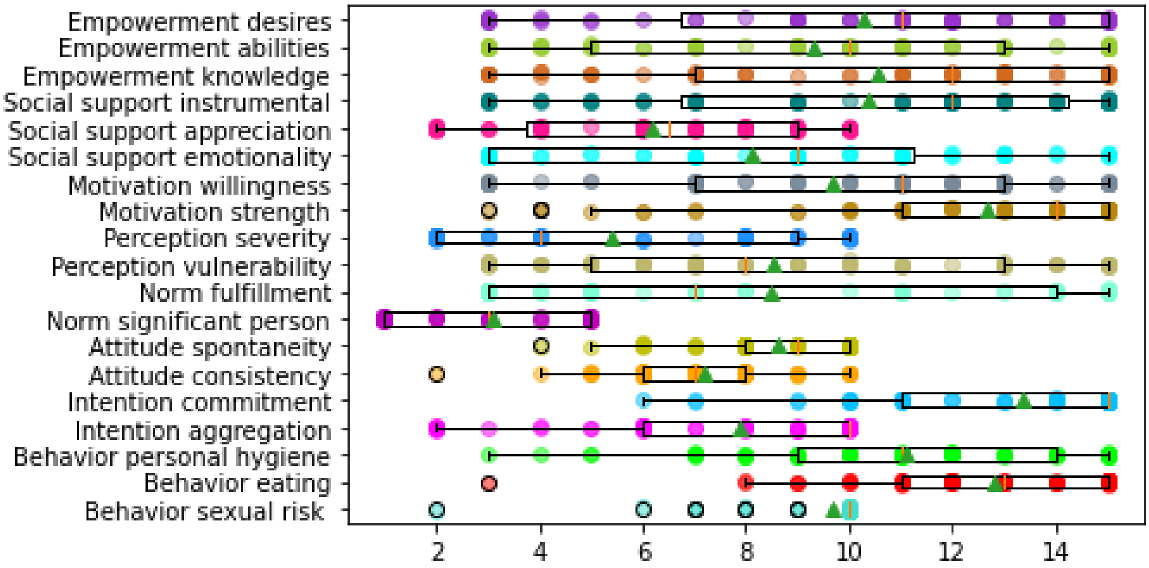


Fig 1:- Details of Features in The Data Set

Within the scope of this study, the early detection of cervical cancer based on behavior determinants is investigated with 3 different machine learning methods: kNN, random forest, and SVM. As can be understood from the detailed literature review, machine learning methods have been used before in this data set. However, in this study, the effect of the hyperparameters in each method is analyzed and the performance of these methods is tried to be increased. More importantly, while the success of the methods was determined by performance metrics, their objectivity and reliability were ensured by using the k-fold cross-validation method.

The structure of this paper is organized as follows. In section 2, detailed information regarding the data is presented. In Section 3, the machine learning methods used in this study are briefly explained. In Section 4, performance metrics along with the performance of the compared methods is detailly given. In section 5, concluding remarks and future aspects are given.

II. DATA SET

The "Cervical Cancer Behavior Risk" data set available on the UCI (University of California, Irvine) Machine Learning Repository website was used in the study [12, 20]. The samples are obtained from the 72 people living in Jakarta, Indonesia. Out of a total of 72 samples, 21 were positive (at risk) and the remaining 51 were negative (not at risk). The data set contains 19 features regarding cervix cancer behavior risk with class label with 1 and -1 as values, which means the instances "at risk" and "not at risk", respectively. Features are obtained from the seven determinants of behavior which are perception, attitude, empowerment motivation, intention, subjective norm, perception, and social support. The Attribute Characteristics of the Cervical Cancer Behavior Risk data set is integer. There is no missing value in the data set. The Box-and-whisker graph presenting the distribution and summary statistics for the features in the data set is shown in Figure 1.

III. METHODS

Three different ML-based classification methods (kNN, random forest, and SVM) used in the study are explained in the following subsections, respectively.

A. k Nearest neighbors (kNN)

kNN classifies test data according to the closest samples in the training data. It takes two user-defined hyperparameters, *k* and distance metric. The distance metric helps identify the closest samples in the data, and *k* indicates the closest sample number taken into account when determining the class of test samples. First, the distance between the query sample and the other samples in the data set is computed and then the *k* closest samples are determined. The class of the query sample is assigned to the most common class among the *k* nearest samples. The effect of *k* in the kNN method is shown in Figure 2. If $k = 3$ is selected, the class of the new sample is assigned to the blue square, and $k = 9$ to the orange diamond. Therefore, the *k* value should be adjusted appropriately for different data sets.

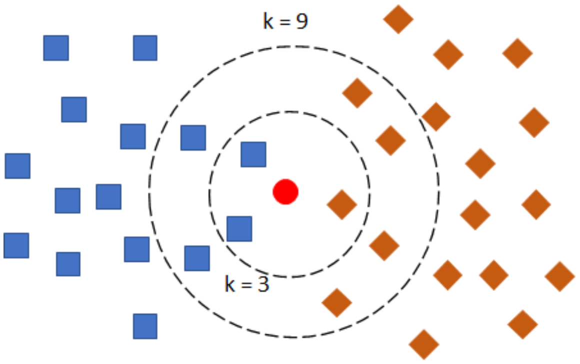


Fig 2:- The effect of k on performance in the kNN method

B. *Random Forest*

Random forest is an ensemble learning method that can be used in both classification and regression problems. In ensemble learning methods, instead of creating a single estimator, many estimators are created to increase the prediction accuracy. Bootstrap aggregation algorithm, abbreviated as bagging, is used as an ensemble learning method in the random forest. In the bagging, subsets are created from the original data set with-replacement and the data size of these subsets are equal to the original data set. Due to the with-replacement resampling, the subsets may contain the same samples more than once. Bootstrapping enables the creation of new training data sets. Decision tree method is used as a learner in the random forest. From these bootstrapped data sets, the subset of features is randomly used for splitting a node in the decision tree method, thus creating weak learners. These weak learners are then combined in the aggregation part. In random forest classification, the class of the new sample is estimated with the decision tree models created and majority voting is used to determine the final prediction of the method. The structure of the random forest is illustrated in Figure 3.

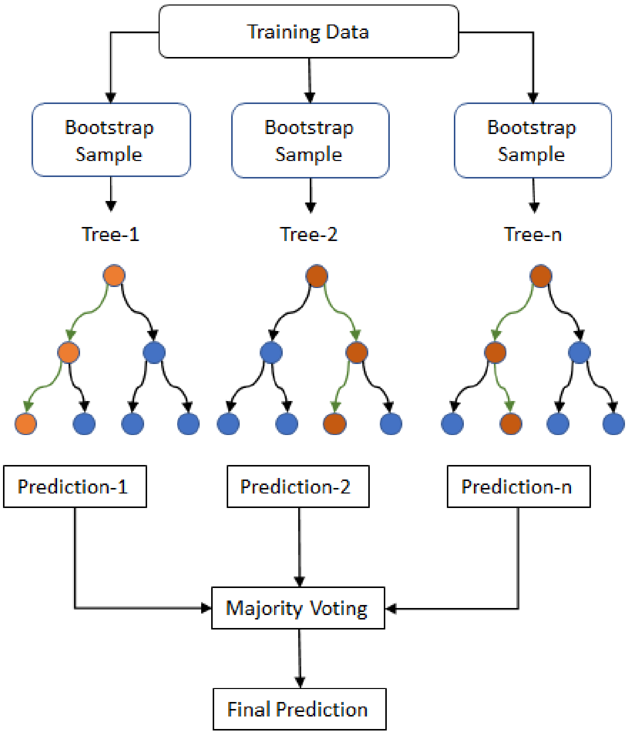
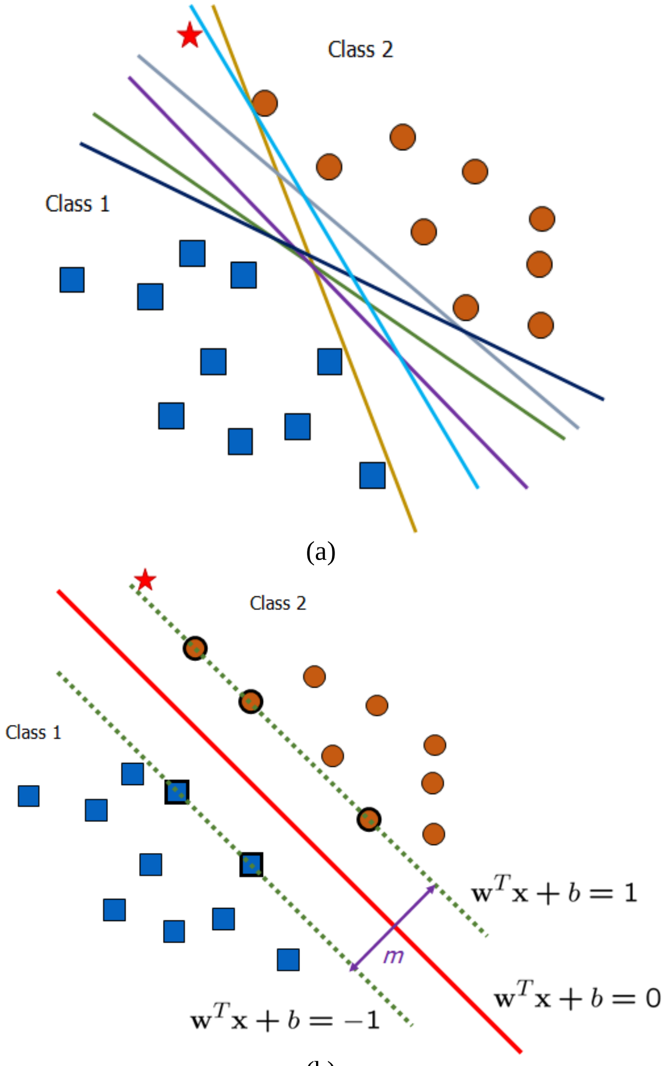


Fig 3:- The structural graph of the random forest

C. *Support Vector Machines (SVM)*

SVM is a powerful machine learning method that was originally created for classification problems and can also be used for regression problems [5, 21]. Considering linearly separable data sets, there are many hyperplanes that can separate classes without any misclassification errors. The multiple hyperplanes that can separate the data are shown in Figure 4(a). Although their ability to generalize on test data is good, their performance may be lower on unseen samples. As can be seen in Figure 4(a), the new sample, indicated by the red star, cannot be correctly classified with some hyperplanes. SVM aims to find the best decision boundary, also called the optimal hyperplane. In this way, the generalization capability of the model will be higher. SVM aims to find the best decision boundary between classes. The optimal hyperplane in the SVM is found by maximizing the margin, which is the perpendicular distance between the hyperplane and the closest samples in each class. The representation of the optimal hyperplane with the same dataset in Figure 4 (a) is shown in Figure 4 (b). As seen in Figure 4 (b), the query sample is perfectly classified by SVM.



(b)

Fig 4:- The decision boundary of the data linearly separable by (a) multiple hyperplanes, (b) optimal hyperplane (SVM)

The equation of the decision boundary is defined as

$$\boldsymbol{w}\cdot\boldsymbol{x}+b=0 \qquad (1)$$

where **w** is the weight vector perpendicular to the hyperplane, *b* is the scalar. In order to maximize margin, which is equal to the 1/**w**, the **w** must be minimized. The quadratic optimization problem of the SVM is defined as

$$\min_{w,b}\|\boldsymbol{w}\|^2$$

$$\text{subject to} \qquad (2)$$

$$y_i\left(\left(\boldsymbol{w}^T\boldsymbol{x}_i\right)-b\right)\ge 1,\ \ i=1,2,\ldots,l$$

The Lagrangian function is given by

$$L(\boldsymbol{w},b,\alpha)=\frac{1}{2}\|\boldsymbol{w}\|^2-\sum_{i=1}^{l}\alpha_i\left(y_i\left(\boldsymbol{w}^T\boldsymbol{x}_i-b\right)-1\right), \qquad (3)$$

$$\alpha \ \ge 0$$

where *l* is the number of samples and α is the lagrange multipliers. The dual form is obtained by differentiating with respect to the primal variables **w** and *b*.

$$\frac{\partial L_P}{\partial \boldsymbol{w}}=0 \Rightarrow \boldsymbol{w}=\sum_{i=1}^{l}\alpha_i y_i \boldsymbol{x}_i \qquad (4)$$

$$\frac{\partial L_P}{\partial b}=0 \Rightarrow \sum_{i=1}^{l}\alpha_i y_i=0 \qquad (5)$$

After substituting Equations 4 and 5 into Equation 3, the dual form becomes

$$\max_{\alpha} L(\alpha)=\sum_{i=1}^{l}\alpha_i-\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}y_i y_j \alpha_i \alpha_j \boldsymbol{x}_i^T\boldsymbol{x}_j,$$

$$\sum_{i=1}^{l}y_i\alpha_i=0, \qquad (6)$$

$$\alpha\ge 0$$

Along with the success of SVM in linearly separable data, it can also demonstrate high performance in nonlinearly separated data. It is enabled by the famously known kernel trick. With the kernel trick, data that cannot be separable in the input space is mapped to a higher dimensional feature space and becomes linearly separable. Although many application-specific kernel functions have been proposed in the literature, the commonly used kernel functions and hyperparameters are given in Table 1.

Kernel Name	Definition	Parameter
Linear	$x_i^T x_j + c$	None
Polynomial	$\left(x_i^T x_j + 1\right)^d$	d
Radial Basis Function	$e^{-\gamma\|x_i-x_j\|^2}$	γ
Sigmoid	$\tanh\left(\gamma\cdot x_i x_j + r\right)$	γ, r

Table 1:- Popular SVM Kernels

IV. RESULT AND DISCUSSION

D. *Performance Metrics*

Performance metrics allow us to evaluate the success of machine learning methods and make comparisons between them. In this regard, many performance measures have been proposed for different problems. Mean absolute error, mean square error, root mean square error, and R^2 are commonly used performance measures for regression problems. Accuracy, precision, recall, and F1-score are commonly used metrics in classification problems and are derived from the confusion matrix. The confusion matrix for binary classification problems is given in Table 2. In this table, True Positives (TPs) indicate the number of correctly predicted positive class instances; False Positives (FPs) give the number of negative instances incorrectly predicted as positive; False Negatives (FNs) give the number of positive instances incorrectly predicted as negative; True Negatives (TNs) indicate the number of correctly predicted negative class instances.

		Actual Values	
		Positive	Negative
Predicted Values	Positive	True Positives (TPs)	False Positives (FPs)
	Negative	False Negatives (FNs)	True Negatives (TNs)

Table 2:- Confusion Matrix for Classification

Accuracy gives the ratio of correctly predicted samples to the all samples in the data set. Accuracy of the model is computed as

$$Accuracy=\frac{TPs+TNs}{TPs+TNs+FPs+FNs} \qquad (7)$$

Precision measures the ratio of correctly classified positive class samples to all positively predicted samples and it is given by

$$Precision=\frac{TPs}{TPs+FPs} \qquad (8)$$

The recall, which is also named as true positive rate, returns the ratio of correctly predicted positive samples to all positive samples in the data set. Recall is defined as

$$Recall=\frac{TPs}{TPs+FPs} \qquad (9)$$

The F1-score computes the harmonic ratio of the precision and recall metrics. In the F1-score, both false positives and false negatives are considered in the performance measurement. For this reason, it can be preferred over the accuracy metric in irregularly distributed data sets. F1-score is computed as

$$F1-score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (10)$$

In machine learning, the dataset is split into training and test sets to build models and evaluate their performance. This separation is often done randomly and tends to give erroneous conclusions about the performance of the model. Because, based on the examples in the train set, the constructed machine learning model may overfit the data, giving good performance on the training set and poorer performance on the test set, or underfit the data, it can give poor results on both the training and the test set. To overcome this situation, k-fold cross validation technique is used. In this technique, the data set is randomly divided into *k* equal parts. One part is left to test the model and the remaining *k - 1* parts are used to build the model. This process is iterated *k* times and the success of the model is found by calculating the average scores of the performances in each iteration. Thus, all samples in the data set are used in both the training and testing phases of the machine learning method.

E. Experimental Results

In this study, the performances of kNN, random forest and SVM methods compared in cervical cancer behavior risk estimation take hyperparameters. Hyperparameters have a significant impact on the performance of machine learning methods and more importantly they have to be set before training the machine learning method. The hyperparameters that give the best performance on a particular data set may demonstrate much lower results when the data set is changed. Therefore, they need to be tuned for different datasets. In this case, the exhaustive grid search method is the common and simplest technique used in hyperparameter optimization. In exhaustive grid search, models are created with all possible combinations of hyperparameters in the user-defined search space. Then, the performance of the models is determined by cross validation. The hyperparameters that achieve the best performance from the results are determined.

Another factor that affects the performance of machine learning methods is that the collected features vary in values or units or magnitudes. Distance and gradient descent-based methods are highly affected by feature scaling, while tree-based methods are rather insensitive. Feature(s) with large-range dominate the result of machine learning methods compared to small-range features. Thanks to scaling, the adverse effects of highly variable size feature(s) on machine learning methods are reduced. Because after feature scaling all features have the same weight, in other words they are all treated equally. The commonly used feature scaling technique is normalization. It is also called min-max scaler. In this technique, each feature in the data set is scaled to be between 0 and 1. Normalization is defined as

$$x_i^{'} = \frac{x_i - \min(X)}{\max(X) - \min(X)} \quad (11)$$

where x_i is the original value and $x_i^{'}$ is the normalized value. max(X) and min(X) represent the maximum and minimum values of this feature, respectively. In this study, min-max scaler is applied to the data set in preprocessing.

To make the benchmarking of the machine learning methods more accurate, the k-fold cross-validation algorithm with *k = 5* was used and repeated 10 times. The effect of hyperparameters on the success of the model is visualized with the results of the accuracy performance metric to facilitate examination of the methods being compared.

The kNN method takes the parameter *k* as the number of nearest neighbors and searches the range between 1 – 35 in increments of one. The result of the kNN is shown in Figure 5. In kNN, the highest score is obtained at *k = 2*, with increasing *k* values, the performance gradually decreases. The drop in performance is much greater when *k* is higher than 26.

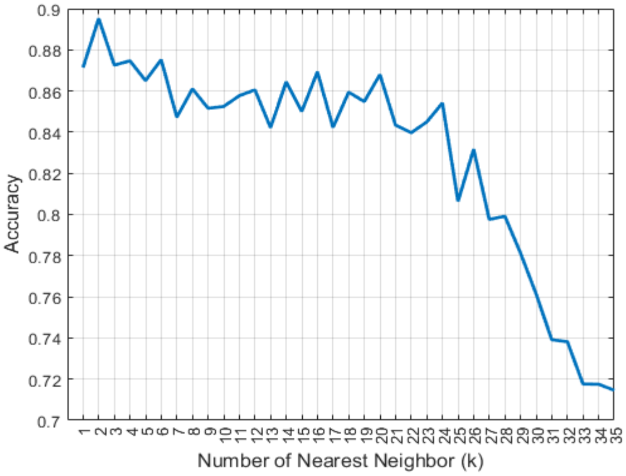


Fig 5:- The accuracy of kNN with in range 1-35

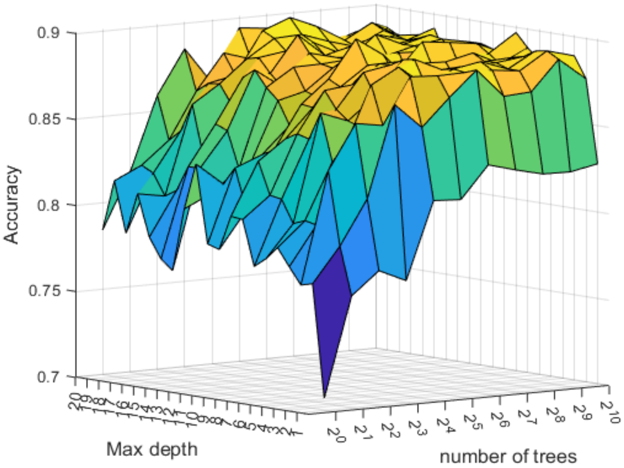


Fig 6:- Accuracy change in random forest related to max_depth and number of trees

The number of trees and the maximum depth of the tree (max_depth) hyperparameters are searched in the random forest, while the remaining hyperparameters are set to criteria = "gini", *min_samples_split = 2*, *max_features = "sqrt"*. The *max_depth* hyperparameter is searched in the range between 1 and 20 with one increment, and the number of trees is searched for $\left\{2^0, 2^1, 2^2, \ldots, 2^{10}\right\}$. The accuracy of the random forest is shown in Figure 6. The highest score in random forest obtained at *max_depth = 3*, and *number of trees* = 2^7. Performance in the hyperparameter range of *max_dept* = $\{3, 20\}$ and *number of trees* = $\left\{2^3, 2^{10}\right\}$ varies relatively small.

SVM receives kernel hyperparameter and the number of hyperparameters SVM takes varies according to the selected kernel. In this regard, the most used kernels (linear, polynomial, sigmoid, and radial basis function (RBF)) are analyzed separately, and their results are visualized. Figure 7 shows the accuracy results of linear kernel SVM (linear-SVM) with regularization hyperparameter (C) values in the range $\left\{2^{-4}, 2^{-3}, 2^{-2}, \ldots, 2^{14}\right\}$. Linear-SVM performance degrades at C values less than 2^{-2}. However, the performance variation is very limited when the C value is in the range of $2^{-2} - 2^{14}$. Linear-SVM's performance is maximized at $C = 2^4$.

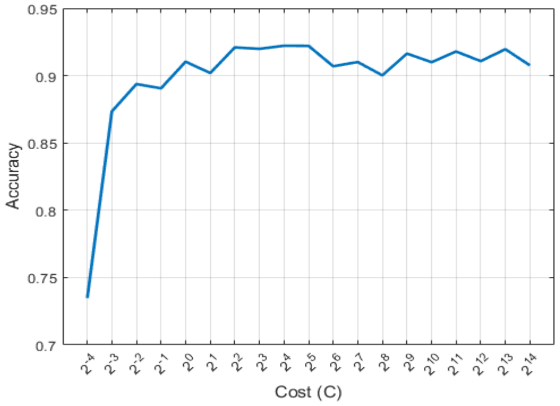


Fig 7:- Impact of C on linear kernel SVM

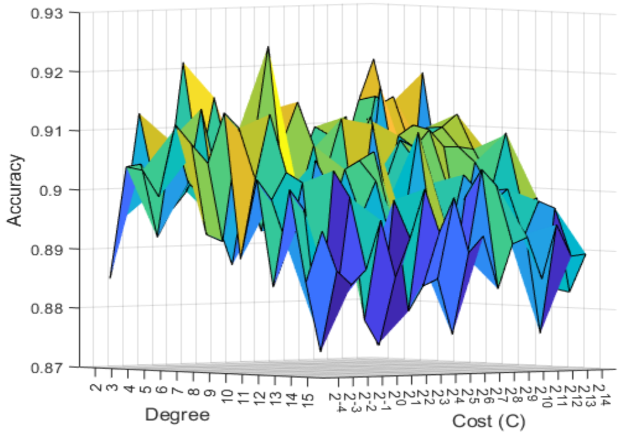


Fig 8:- Impact of degree and C hyperparameters on polynomial kernel SVM

In polynomial kernel SVM (polynomial-SVM), the cost hyperparameter is searched in the range $\left\{2^{-4}, 2^{-3}, 2^{-2}, \ldots, 2^{14}\right\}$ and the degree hyperparameter is searched in the range 2 to 15 with a one increment. Accuracy scores are shown in Figure 8. In polynomial-SVM, the best accuracy is obtained with $degree = 9,\ C = 2^{-1}$ hyperparameters (0.9250), while the lowest accuracy (0.8741) is obtained with $degree = 14,\ C = 2^{-3}$ hyperparameters. The difference between the highest and lowest accuracy is 5.5%.

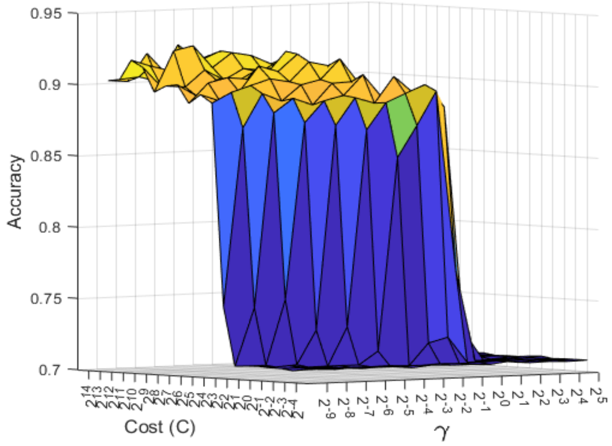


Fig 9:- Impact of C and γ hyperparameters on RBF kernel SVM

The RBF kernel SVM (RBF-SVM) takes the C and γ hyperparameters and is searched in the range $\left\{2^{-4}, 2^{-3}, 2^{-2}, \ldots, 2^{14}\right\}$ and $\left\{2^{-9}, 2^{-8}, 2^{-7}, \ldots, 2^{5}\right\}$, respectively. As can be seen from Figure 9, the accuracy scores are barely changed in the hyperparameters combinations of $C = 2^4 - 2^{14}$ and $\gamma = 2^{-9} - 2^0$, and scores are much lower outside this range. Also, the accuracy scores in the hyperparameters $C = 2^{-2} - 2^{-4}$ and $\gamma = 2^2 - 2^5$ are the lowest and are around 0.7. In RBF-SVM, the best score (0.9268) is achieved at $C = 2^{13}$ and $\gamma = 2^{-6}$.

As with RBF-SVM, the sigmoid kernel SVM (sigmoid-SVM) takes the C and gamma hyperparameters and is searched in the same range as RBF-SVM. RBF-SVM scores highest for hyperparameter combinations in the $\gamma = 2^{-9} - 2^{-4}$ and $C = 2^5 - 2^{14}$ ranges. Performance results are visualized in Figure 10. Especially after $\gamma = 2^{-3}$, accuracy scores decrease with increase in gamma hyperparameter. The lowest accuracy (0.5860) is achieved using hyperparameters with $C = 2^{11}$, $\gamma = 2^1$, while the highest score (0.9274) is achieved using hyperparameters with $\gamma = 2^{-9}$, $C = 2^{12}$. The difference between the highest and lowest accuracy score is 58.32%, indicating the importance of hyperparameter optimization.

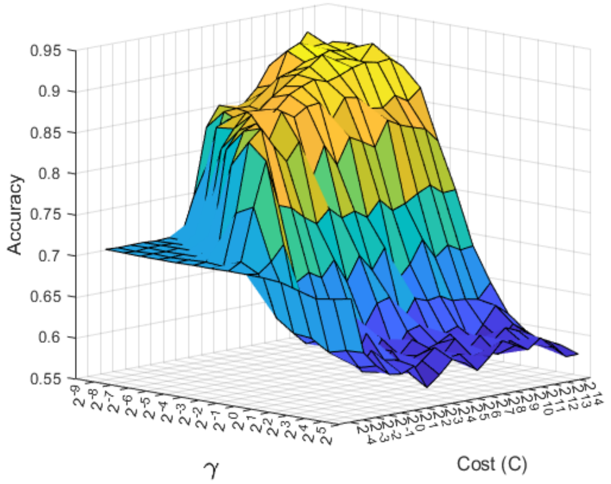


Fig 10:- Impact of C and γ hyperparameters on sigmoid kernel SVM

The results of the methods compared in accuracy, precision, recall, and F1-score performance metrics are given in Table 3. This table presents the results of the hyperparameters for which the methods obtained the highest accuracy values and the results of other performance metrics. The highest scores on all performance metrics are highlighted in bold. According to the results in Table 3, the best scores are obtained with sigmoid-SVM in all performance metrics except the recall metric. Polynomial-SVM achieved the highest score on the recall metric. The lowest scores on the accuracy and recall metrics are achieved in kNN, while in the precision and F1-score metrics they are in the random forest. Excluding the recall metric, the performance of SVM with different kernels is very close to each other. However, in the recall metric, performance differs by 10% in the SVM between the lowest and highest scores.

The performance results of the methods used in the literature with the same data set are shown in Table 4. Not available (NA) in Table 4 indicates that the performance result for this metric is not available or calculated from the paper. When the studies in the literature were compared with the accuracy metric, the studies in 15, 18 and 19 outperformed this study. Results in these studies are based on a training/testing split, so they may differ at each iteration, leading to performance degradation. The presented study outperformed studies in [13, 16, 17] using k-fold cross validation. It also outperformed studies in [12, 14] that used all data and training/validation/test splits.

	Accuracy	**Precison**	**Recall**	**F1-score**
kNN k=2	0.8952	0.8848	0.7458	0.7863
Random Forest max_depth = 3, number of trees = 2^7	0.9011	0.8975	0.7256	0.7830
linear-SVM $C = 2^{4}$	0.9222	0.8737	0.8269	0.8299
Polynomial-SVM $C = 2^{-1}, degree = 9$	0.9251	0.9022	**0.9022**	0.8536
RBF-SVM $C = 2^{13}, \gamma = 2^{-6}$	0.9268	0.8747	0.8196	0.8277
Sigmoid-SVM $\gamma = 2^{-9}, C = 2^{12}$	**0.9274**	**0.9093**	0.8410	**0.8565**

Table 3:- Table Styles

V. CONCLUSION

In this study, different machine learning methods, kNN, random forest, and SVM with different kernels, were compared for the estimation of cervical cancer behavioral risk. Minimum-max normalization was applied to reduce the dominating effect of the features and k-fold cross validation (with *k = 10, iteration = 10*) was used to increase the reliability of the results. In order to obtain the best performance in each benchmarking method, hyperparameter optimization was performed with exhaustive grid search technique. Performance measures of accuracy, precision, recall, and F1-score are employed to evaluate the benchmarked methods. According to the experimental results, the highest accuracy (0.9274) was obtained with sigmoid-SVM. Performances on accuracy, precision, and recall metrics are quite similar in SVM with four different kernels. However, the hyperparameters used to obtain the highest performances in the SVM method are quite different.

Paper	Validation	Method	Accuracy	Precision	Recall	F1-score
Machmud and Wijaya **[12]**	All data	Logistic regression	0.8750	0.8000	0.7619	0.7805
		Naive Bayes	0.9167	0.8947	0.8095	0.8500
Tarakci and Ozkan **[13]**	k-fold cross validation, k = 10	kNN	0.8470	0.8225	1.0000	.9026
		WkNN	0.8890	0.8644	1.000	.9272
Gamara et al. **[14]**	train (70%), validation (15%), and test (15%)	Artificial Neural Networks	0.9091	1.0000	0.8000	0.8889
Akter et al. **[15]**	train (80%), test (20%)	Decision Tree	0.9333	1.0000	0.9091	0.9524
		Random Forest	0.9333	0.9167	1.0000	0.9565
		XGBoost	0.9333	0.9333	1.0000	0.9655
Alphan **[16]**	k-fold cross validation, k = 10	SVM	0.9167	NA	0.9200	NA
Ghanem et al. **[17]**	k-fold cross validation, k = 5	SIA	0.8056	0.7833	0.7301	0.7558
Ratul et al. **[18]**	train (80%), test (20%)	MLP	0.9333	0.9091	1.000	0.9524
Cicek et al. **[19]**	train (75%), test (25%)	Random Forest	0.9444	0.7500	1.000	0.9444
This study	k-fold cross validation, k = 10	Sigmoid-SVM	0.9274	0.9093	0.8410	0.8565

Table 4:- Comparison of the Performance with the Literature

As a future study, more samples can be collected to improve the performance of machine learning methods. Ensemble learning methods can be used to improve performance.

REFERENCES

[1]. Z. Ou, S. Lin, J. Qiu, W. Ding, P. Ren, D. Chen, ..., and P. Wu, "Single-Nucleus RNA Sequencing and Spatial Transcriptomics Reveal the Immunological Microenvironment of Cervical Squamous Cell Carcinoma," Advanced Science, 2203040, 2022.

[2]. Cancer Facts & Figures 2022. Atlanta: American Cancer Society; 2022.

[3]. E. Ersoy, and Ö. Karal, "Yapay sinir ag˘ları ve insan beyni," İnsan ve Toplum Bilimleri Araştırmaları Dergisi, vol. 1, no. 2, pp. 188-205, 2012.

[4]. A. Degirmenci, and O Karal, "Evaluation of kernel effects on svm classification in the success of wart treatment methods," Am. J. Eng. Res, vol. 7, pp. 238-244, 2018.

[5]. Ö. Karal, "Destek Vektör Regresyon ile EKG Verilerinin Sıkıştırılması," Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi, 2018.

[6]. D.T. Arık, Ö. Karal, and A.B. Şahin, "A Comparative Study of Artificial Neural Networks and Na¨ıve Bayes Techniques for the Classification of Radar Targets," Bitlis Eren Üniversitesi Fen Bilimleri Dergisi, vol. 9, no. 4, pp. 1779-1788, 2020.

[7]. M. Apaydın, M. Yumuş, A. Değirmenci, and Ö. Karal, "Evaluation of Air Temperature with Machine Learning Regression Methods Using Seoul City Meteorological Data," Pamukkale University Journal of Engineering Sciences, 2022.

[8]. Ş. Hatipoğlu, M.A. Belgrat, A. Degirmenci, and O. Karal, "Prediction of Unemployment Rates in Turkey by k-Nearest Neighbor Regression Analysis," In 2021 Innovations in Intelligent Systems and Applications Conference (ASYU), pp. 1-5, IEEE, October, 2021.

[9]. A. Degirmenci, and O. Karal, "Robust Incremental Outlier Detection Approach Based on a New Metric in Data Streams," IEEE Access, vol. 9, pp. 160347-160360, 2021.

[10]. A. Degirmenci, and O. Karal, "Efficient density and cluster based incremental outlier detection in data streams," Information Sciences, vol. 607, pp. 901-920, 2022.

[11]. A. Degirmenci, and O. Karal, "iMCOD: Incremental multi-class outlier detection model in data streams," Knowledge-Based Systems, 2022.

[12]. R. Machmud, and A. Wijaya, "Behavior determinant based cervical cancer early detection with machine learning algorithm," Advanced Science Letters, vol. 22, no. 10, pp. 3120-3123, 2016.

[13]. F. Tarakci, and I.A. Ozkan, "Comparison of classification performance of kNN and WKNN algorithms," Selcuk University Journal of Engineering Sciences, vol. 20, no. 2, pp. 32-37, 2021.

[14]. P.R.C. Gamara, R.Q. Neyra, and K.H.A. Recto, "Behavior-Based Early Cervical Cancer Risk Detection Using Artificial Neural Networks," In 2021 IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), pp. 1-6, November, IEEE 2021.

[15]. L. Akter, M. Islam, M.S. Al-Rakhami, and M. Haque, "Prediction of cervical cancer from behavior risk using machine learning techniques," SN Computer Science, vol. 2, no. 3, pp. 1-10, 2021.

[16]. K. Alpan, "Performance evaluation of classification algorithms for early detection of behavior determinant based cervical cancer," In 2021 5th international symposium on multidisciplinary studies and innovative technologies (ISMSIT), pp. 706-710, IEEE, October, 2021.

[17]. S. Ghanem, R. Couturier, and P. Gregori, "An Accurate and Easy to Interpret Binary Classifier Based on Association Rules Using Implication Intensity and Majority Vote," Mathematics, vol. 9, no. 12, 1315, 2021.

[18]. I.J. Ratul, A. Al-Monsur, B. Tabassum, A.M. Ar-Rafi, M.M. Nishat, and F. Faisal, "Early risk prediction of cervical cancer: A machine learning approach," In 2022 19th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), pp. 1-4, IEEE, May, 2022.

[19]. İ.B. Cicek, S.E.L. İlhami, F.H. YAĞIN, and C. Colak, "Development of a Python-Based Classification Web Interface for Independent Datasets," Balkan Journal of Electrical and Computer Engineering, vol. 10, no. 1, pp. 91-96, 2022.

[20]. D. Dua, and C. Graff, "UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]," Irvine, CA: University of California, School of Information and Computer Science, 2019.

[21]. O. Karal, "Maximum likelihood optimal and robust Support Vector Regression with lncosh loss function," Neural networks, vol. 94, pp. 1-12, 2017.