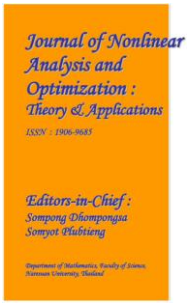
Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 15, Issue. 1 : 2024
ISSN : **1906-9685**



DESIGN AND DEVELOPMENT OF AN EFFICIENT

P. RAJENDRA KUMAR, Department of Information Technology, NRI Institute of Technology, Pothavarappadu (V), Agiripalli (M), Eluru (Dt)-521212
Y. NAVYA REDDY, Department of Information Technology, NRI Institute of Technology, Pothavarappadu (V), Agiripalli (M), Eluru (Dt)-521212
P. MOUNIKA SRI DEVI Department of Information Technology, NRI Institute of Technology, Pothavarappadu (V), Agiripalli (M), Eluru (Dt)-521212
I. NAVEENA Department of Information Technology, NRI Institute of Technology,Pothavarappadu (V), Agiripalli (M), Eluru (Dt)-521212

Abstract:
A rising number of people and businesses are using deep learning (DL) and machine learning (ML) to analyse vast volumes of data and provide insights that can be put to use. In medical practice, it is becoming more and more usual to use machine learning (ML) methods to predict major illnesses such as cancer, kidney failure, and heart attacks in their early stages. One of the most common illnesses affecting women is cervical cancer, which may be prevented with an early diagnosis. Cervical cancer is still the most common malignancy in women worldwide. Therefore, it is imperative to differentiate between the significance of cervical cancer risk factors in order to categorise prospective patients. Over 85% of women's early deaths globally are attributed to cervical cancer, making it one of the main causes of death for women in underdeveloped nations. Cervical cancer is connected with multiple risk factors. In this study, we created a prediction model that uses risk patterns from individual medical records and initial screening to forecast the prognosis of patients with cervical cancer. This work offers a clever method for using ML systems to predict cervical cancer. Predictive model selection (PMS), data preprocessing, pseudo-code, and research dataset comprise the four stages of the suggested research methodology. A variety of traditional machine learning techniques, such as decision trees (DT), logistic regression (LR), support vector machines (SVM), adaptive boosting, gradient boosting, random forests, K-nearest neighbours algorithm (KNN), and XGBoost, are tested and reported in the PMS section. The methods that yield the highest classification score for cervical cancer prediction include gradient boosting, adaptive boosting, decision tree (DT), and random forest (RF).

1. Introduction

Early-stage cervical cancer is symptomless, making it the fourth most common cause of cancer-related deaths among women. Cervical cancer can only be diagnosed with a few methods now available. This research introduces the support vector machine (SVM) technique for cervical cancer diagnosis. To further aid in the diagnosis of malignant cancer samples, two enhanced SVM techniques—support vector machine-recursive feature elimination and support vector machine-principal component analysis, or SVM-PCA—are put forth. 32 risk factors and 4 target variables—Hinselmann, Schiller, Cytology, and Biopsy—represent the cervical cancer data. The three SVM-based methods have each successfully diagnosed and classified each of the four targets. We next check our ranking result of risk variables with the ground reality and compare these three approaches. The superiority of the SVMPCA approach over the others is demonstrated. Artificial intelligence (AI), which helps human decision-makers make better decisions, has been used with Internet of Things (IoT) to enable significant improvements in cancer research. In order to forecast cancer genes that recur in the cervix, the Least Absolute Shrinkage and Selection Operator (LASSO) classifier was recently introduced. The first stage is gathering the lncRNA recurrent gene expression from Geo Datasets. This study synthesised pseudo-

computed tomography (CT) images step-by-step and built an anatomic semantic guided neural style transfer system. Twenty individuals with cervical cancer who were to get treatment were chosen based on their CT and US scans. The region growth approach was used to segment the foreground (FG) and background (BG) regions of the US photos, and three objective functions were developed for content, style, and contour loss. A convolution neural network-based local pseudo-CT image synthesis model was developed based on the two categories of areas. The enhanced technique offers a unique route for image guidance in cervical cancer brachytherapy and can create pseudo-CT images with excellent precision. Globally, women are affected by cervical cancer as the fourth most frequent malignant disease. Most of the time, cervical cancer is not detectable in its early stages. Numerous variables, including smoking, sexually transmitted infections, and the human papilloma virus, raise the risk of getting cervical cancer. It is difficult research to determine those elements and develop a classification model to determine whether or not the cases are cervical cancer. The purpose of this study is to use risk variables for cervical cancer to develop a classification model utilising the Random Forest (RF) classification technique, which includes the synthetic minority oversampling technique (SMOTE) and two feature reduction techniques: principal component analysis (PCA) and recursive feature removal.

2. Proposed System

Predictive model selection (PMS), training method, data preparation, and research dataset make up the various sections of the suggested research methodology. The model described in this research accomplishes certain necessary tasks in each level, thus it is evident that the architectural diagram has been divided into four phases. The Research Dataset section contains information on how research data are collected. How to clean up the dataset and prepare it for machine learning feeds is covered in the Data Preprocessing section. The PMS section displays the kind of prediction model that was chosen to forecast cervical cancer in this study. The Training Methods section lists the requirements for model training. Finally, we design the platform to provide an overall pipeline of cervical cancer prediction using the Python programming language. This research implements an algorithm that is better suited for the categorization of negative and positive cervical cancer diagnoses for clinical use. Cervical cancer can be diagnosed with the help of algorithms including decision tree, logistic regression, support vector machine (SVM), K-nearest neighbours (KNN), adaptive boosting, dradient boosting, random forest, and XGBoost. Class imbalance is a typical problem in ML when there are far fewer members of one class than the other(s). In random over-sampling, this method replicates examples from the minority class at random until the required balance is reached. Selecting an instance, determining its KNN, and creating additional instances along the line segments connecting them are its next steps. Random oversampling causes overfitting problems that SMOTE helps to overcome. The synthetic minority over-sampling approach (SMOTE) is utilised by SMOTE-Tomek in our study to over-sample, while Tomek Link is used to under-sample. Using statistical data, the filter approach ranks the features based on how relevant they are to the target variable. Algorithms that are more suitable for the clinical classification of positive and negative cervical cancer were identified by this study. These algorithms can be used to diagnose cervical cancer. When the model is first being trained, the system will be fed the training data. ML algorithms are then implemented. Finally, using the recently gathered data, prediction is done. The following are the main contributions of our work: The performance of the suggested model fared better than other models.

2.1Advantages of proposed system

Pruning, which produces a tree with fewer branches, will result in a smaller, more straightforward tree that will increase accuracy and avoid overfitting.

• Analysing how feature selection methods and machine learning models work together to predict cervical cancer.

• Uses a combination of SMOTE approach, meta-learner, and several ML techniques to predict cervical cancer.

Employing multiple feature selection strategies, the most significant and pertinent traits that enhance the performance of cancer detection are chosen.

• Based on the chosen features, our suggested approach performs better than previous works, according to the performance findings.

2.2 SYSTEM ARCHITECTURE

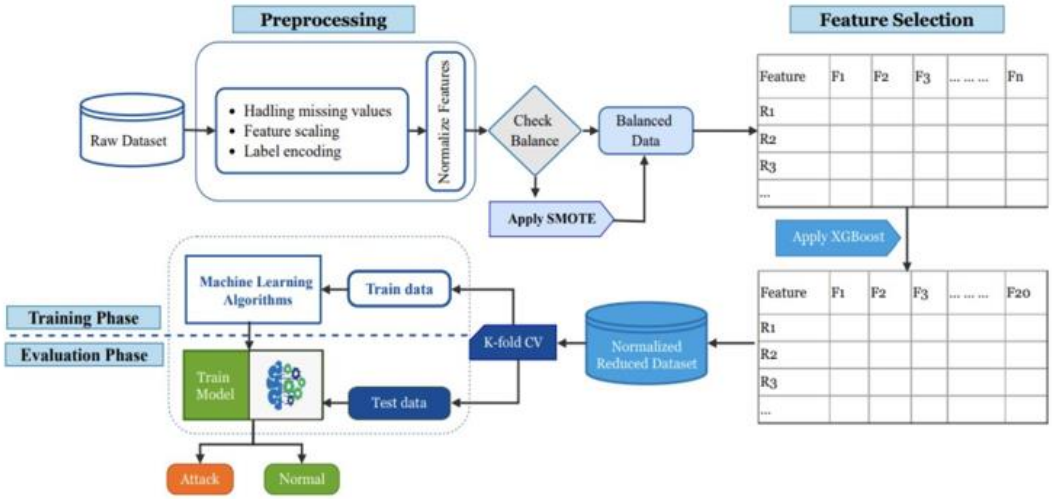


Figure.1. System architecture

2.3DATA FLOW DIAGRAM

1. Another name for the DFD is a bubble chart. A system can be represented using this straightforward graphical formalism in terms of the input data it receives, the different operations it performs on that data, and the output data it generates.
2. One of the most crucial modelling tools is the data flow diagram (DFD). The components of the system are modelled using it. These elements consist of the system's procedure, the data it uses, an outside party that communicates with it, and the information flows within it.
3. DFD illustrates the flow of information through the system and the various changes that alter it. This method uses graphics to show how information flows and the changes made to data as it goes from input to output
4. Another name for DFD is a bubble chart. Any level of abstraction can be utilised to portray a system using a DFD. DFD can be divided into phases that correspond to escalating functional detail and information flow.

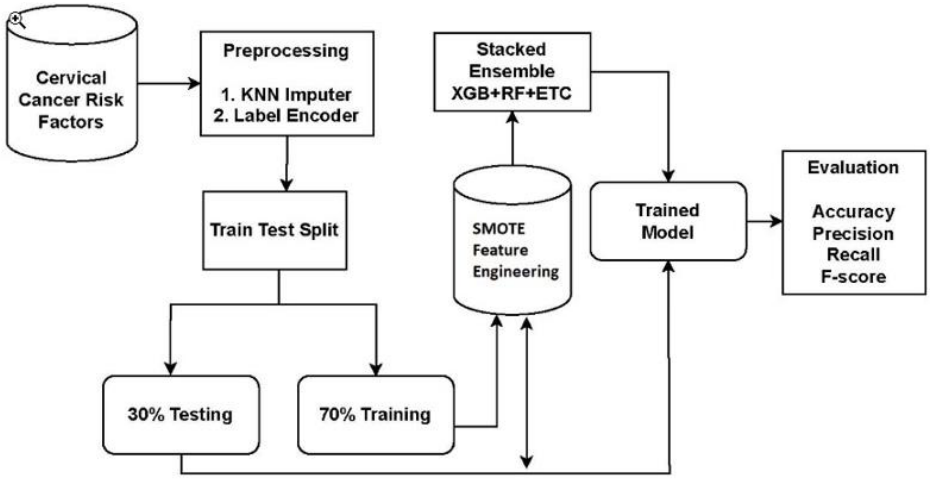


Figure.2. Dataflow diagrams

2.4 UML DIAGRAMS

Unified Modelling Language is known as UML. An industry-standard general-purpose modelling language used in object-oriented software engineering is called UML. The Object Management Group developed and oversees the standard

The intention is for UML to spread as a standard language for modelling object-oriented software. The two main parts of UML as it exists now are a notation and a meta-model. In the future, UML may also include other processes or methods that are connected to it.

2.5 Use case diagram

According to the Unified Modelling Language (UML), a use case diagram is a particular kind of behavioural diagram that is produced from and defined by a use case study. Its objective is to provide a graphical summary of the functionality that a system offers in terms of actors, use cases (representations of their goals), and any interdependencies among those use cases. A use case diagram's primary goal is to display which actors receive which system functionalities.

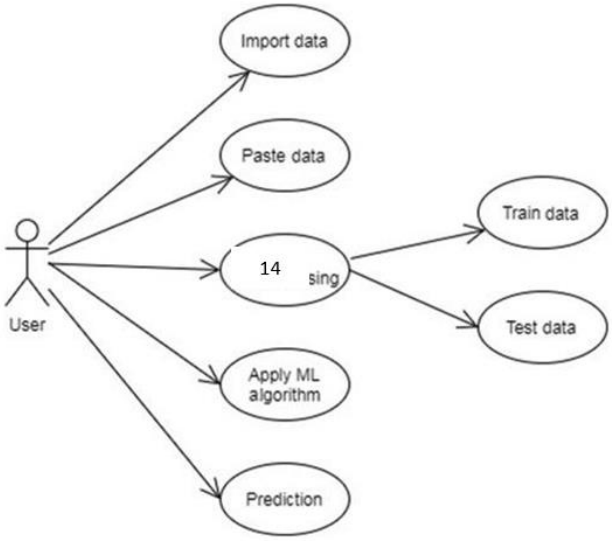


Figure.3. Usecase diagram

2.6 Class diagram

The use case diagram and the system's comprehensive design are both improved by the class diagram. The actors identified in the use case diagram are categorised into a number of related classes by the class diagram. It's possible that every class in the class diagram can perform certain functions. The "methods" of the class refer to these features that it offers. In addition, every class might possess specific "attributes" that allow for class uniqueness.

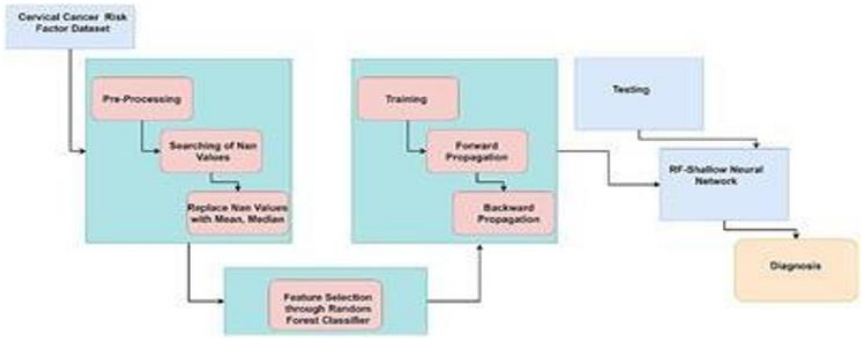


Figure.4. Class diagram

2.7Activity diagram:

The process flows in the system are captured in the activity diagram. Similar to a state diagram, an activity diagram also consists of activities, actions, transitions, initial and final states, and guard conditions.

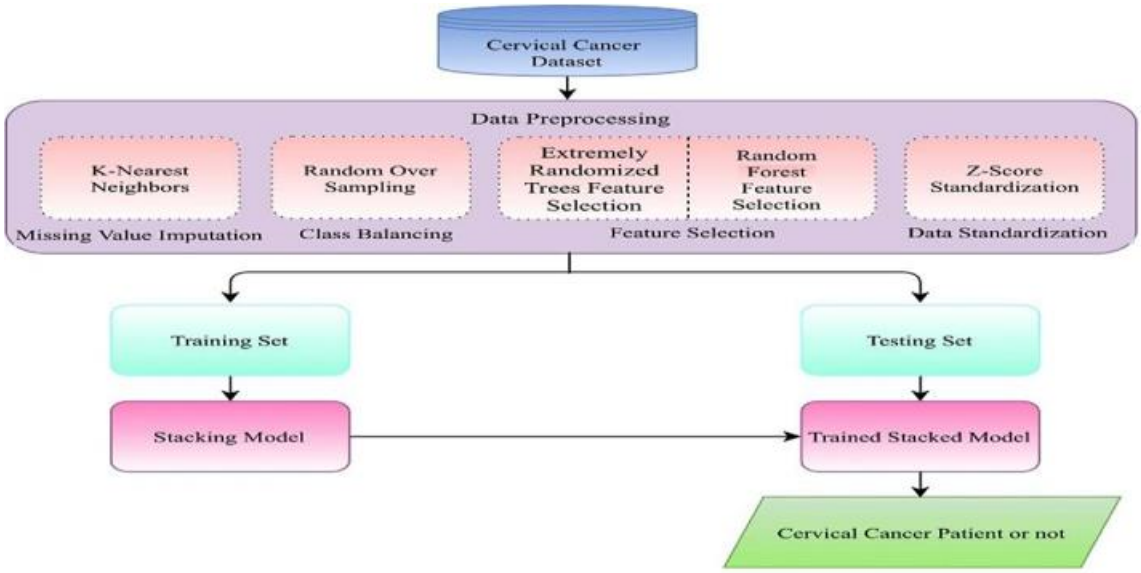


Figure.5. Activity diagram

2.8 Sequence diagram

The way various system items interact with one another is depicted in a sequence diagram. A sequence diagram's time-ordering is one of its key features. This indicates that a step-by-step representation of the precise order in which the items interacted is provided. In the sequence diagram, various objects communicate with one another by sending "messages".

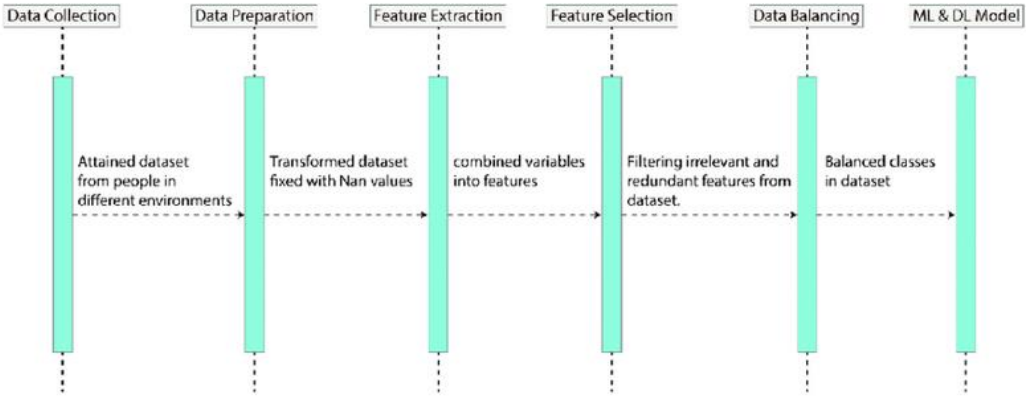


Figure.6. Sequence diagram

2.9 Collaboration diagram

A collaboration diagram groups together the interactions between different objects. The interactions are listed as numbered interactions that help to trace the sequence of the interactions. The collaboration diagram helps to identify all the possible interactions that each object has with other objects.

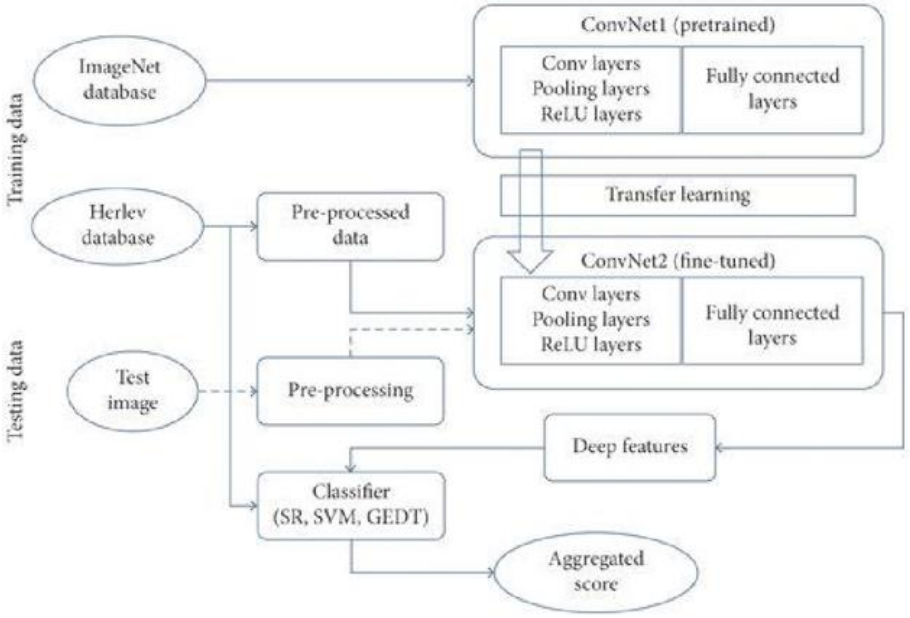


Figure.7. Collaboration diagram

2.10. Component diagram

The system's high-level components are represented by the component diagram. At a high level, this diagram shows which elements make up the system and how they are connected. A component diagram shows the parts removed from the system after it has completed the building or development stage.

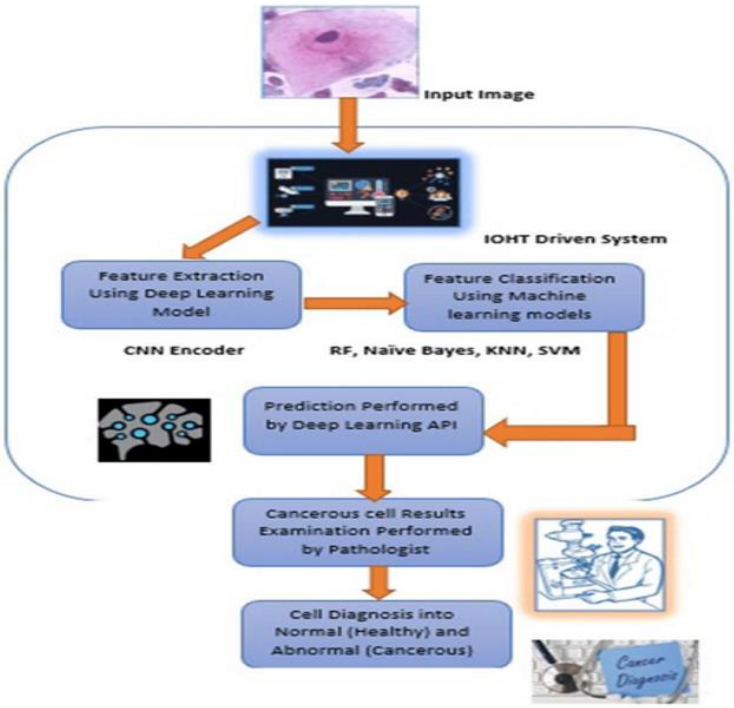


Figure.8. Component diagram

2.11. Deployment diagram:

The deployment diagram captures the configuration of the runtime elements of the application. This diagram is by far most useful when a system is built and ready to be deployed.

Figure.9. Deployment diagram

2.13 Software Testing Strategies:

The greatest strategy to make software engineering testing more effective is to optimise the approach. A software testing plan outlines the steps that must be taken in order to produce a high-quality final product, including what, when, and how. To accomplish this main goal, the following software testing techniques—as well as their combinations—are typically employed:

Static Examination:

Static testing is an early-stage testing approach that is carried out without really operating the development product. In essence, desk-checking is necessary to find errors and problems in the code itself. This kind of pre-deployment inspection is crucial since it helps prevent issues brought on by coding errors and deficiencies in the software's structure.

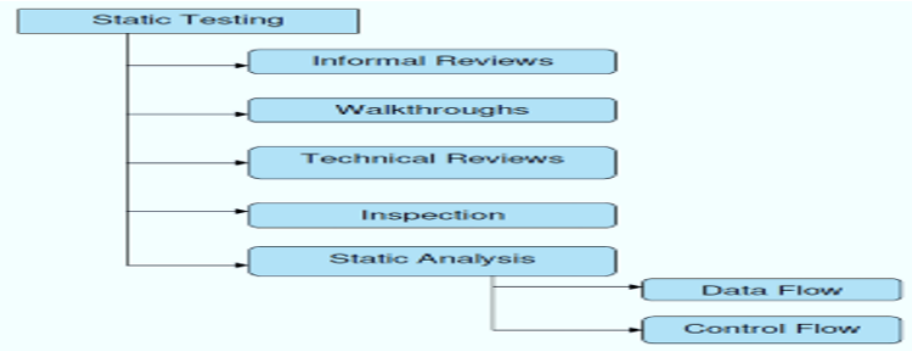


Figure.10. Static Testing

2.14 Structural Testing

Software cannot be tested efficiently unless it is run. White-box testing, another name for structural testing, is necessary to find and correct flaws and faults that surface during the pre-production phase of the software development process. Regression testing is being used for unit testing depending on the programme structure. To expedite the development process at this point, it is typically an automated procedure operating inside the test automation framework. With complete access to the software's architecture and data flows (data flows testing), developers and quality assurance engineers are able to monitor any alterations (mutation testing) in the behaviour of the system by contrasting the test results with those of earlier iterations (control flow testing).

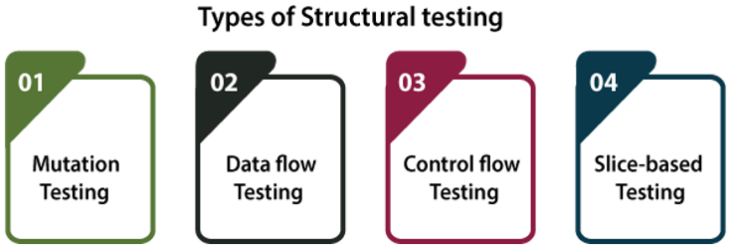


Figure.11. Structural Testing

2.15 Behavioural Testing

Rather than the mechanics underlying these reactions, the final testing phase concentrates on how the programme responds to different activities. Put differently, behavioural testing, commonly referred to as black-box testing, relies on conducting multiple tests, the majority of which are manual, in order to examine the product from the perspective of the user. In order to perform usability tests and respond to faults in a manner similar to that of ordinary users of the product, quality assurance engineers typically possess specialised information about a company or other purposes of the software, sometimes known as "the black box." If repetitive tasks are necessary, behavioural testing may also involve automation (regression tests) to remove human error. To see how the product handles an activity like filling out 100 registration forms on the internet, for instance, it would be better if this test were automated.

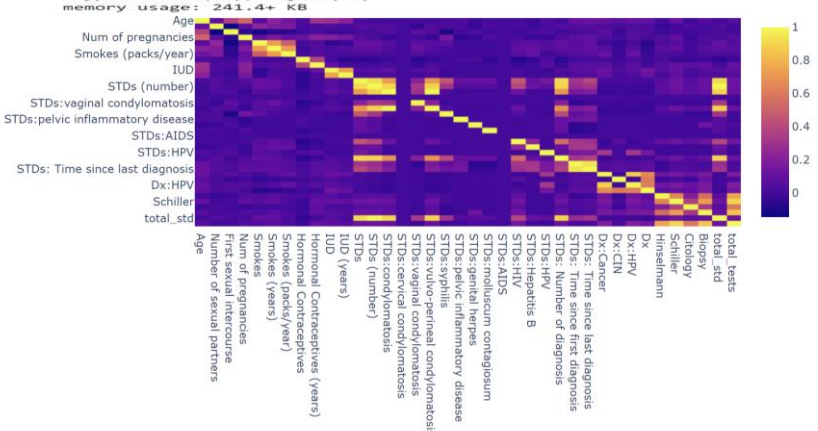
Figure.12. Behavioural Testing

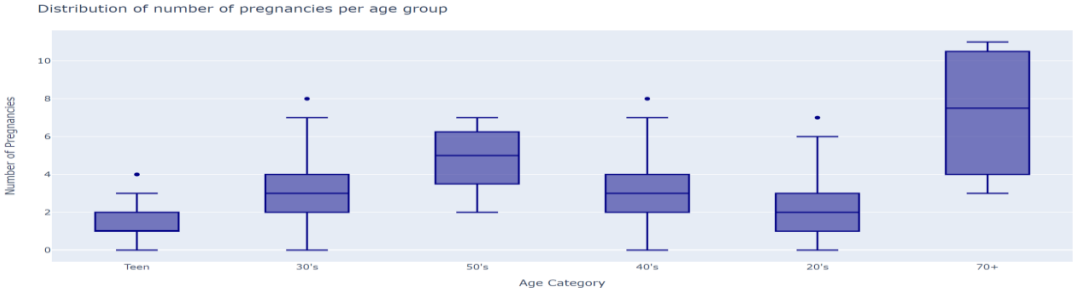
3. Results and Discussion

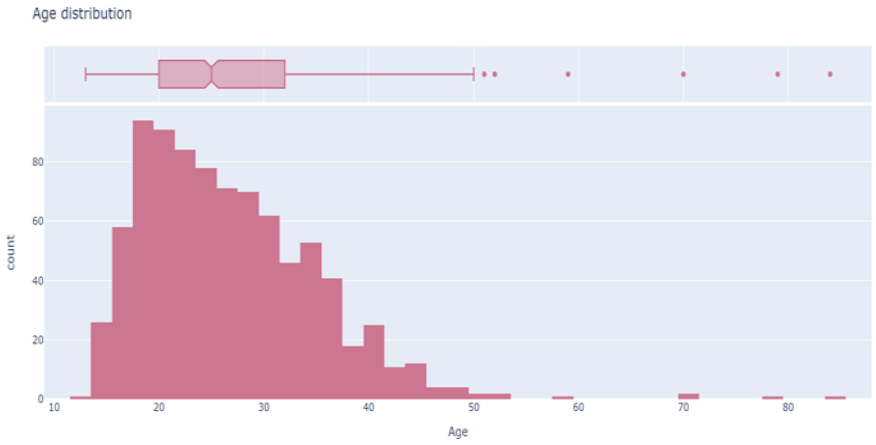
	Age	Number of sexual partners	First sexual intercourse	Num of pregnancies	Smokes	Smokes (years)	Smokes (packs/year)	Hormonal Contraceptives	Hormonal Contraceptives (years)	IUD	...	STDs: Time since first diagnosis	STDs: Time since last diagnosis	Dx:Cancer	Dx:CIN	Dx:HPV	Dx	Hinselmann	Schiller	Citology	Biopsy
0	18	4.0	15.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	...	?	?	0	0	0	0	0	0	0	0
1	15	1.0	14.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	...	?	?	0	0	0	0	0	0	0	0
2	34	1.0	?	1.0	0.0	0.0	0.0	0.0	0.0	0.0	...	?	?	0	0	0	0	0	0	0	0
3	52	5.0	16.0	4.0	1.0	37.0	37.0	1.0	3.0	0.0	...	?	?	1	0	1	0	0	0	0	0
4	46	3.0	21.0	4.0	0.0	0.0	0.0	1.0	15.0	0.0	...	?	?	0	0	0	0	0	0	0	0

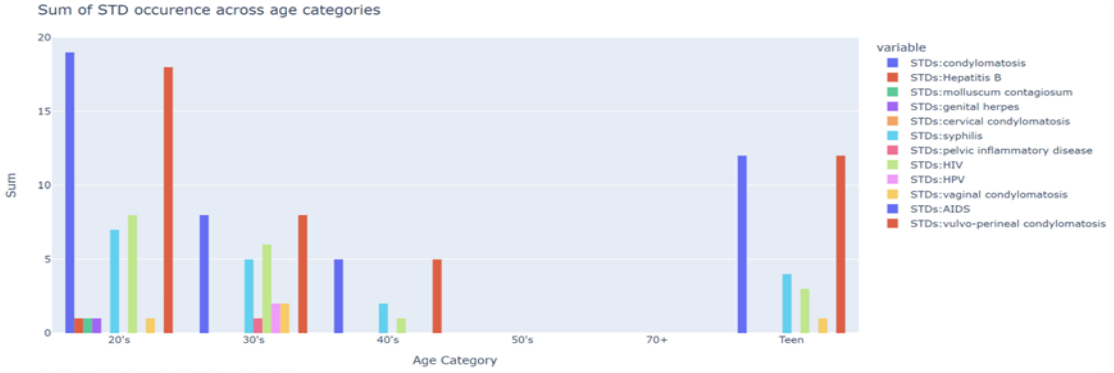
5 rows × 36 columns

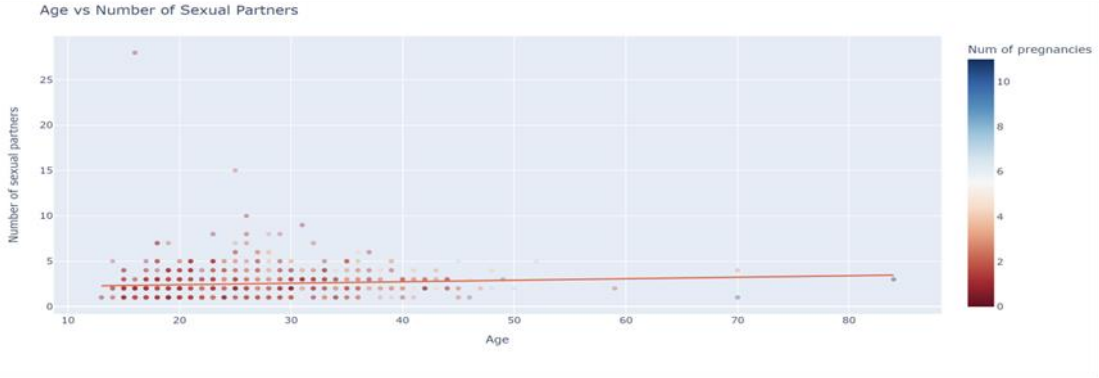
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 858 entries, 0 to 857
Data columns (total 36 columns):
 #   Column                              Non-Null Count  Dtype
---  ------                              --------------  -----
 0   Age                                 858 non-null    int64
 1   Number of sexual partners           858 non-null    object
 2   First sexual intercourse            858 non-null    object
 3   Num of pregnancies                  858 non-null    object
 4   Smokes                              858 non-null    object
 5   Smokes (years)                      858 non-null    object
 6   Smokes (packs/year)                 858 non-null    object
 7   Hormonal Contraceptives             858 non-null    object
 8   Hormonal Contraceptives (years)     858 non-null    object
 9   IUD                                 858 non-null    object
 10  IUD (years)                         858 non-null    object
 11  STDs                                858 non-null    object
 12  STDs (number)                       858 non-null    object
 13  STDs:condylomatosis                 858 non-null    object
 14  STDs:cervical condylomatosis        858 non-null    object
 15  STDs:vaginal condylomatosis         858 non-null    object
 16  STDs:vulvo-perineal condylomatosis  858 non-null    object
 17  STDs:syphilis                       858 non-null    object
 18  STDs:pelvic inflammatory disease    858 non-null    object
 19  STDs:genital herpes                 858 non-null    object
 20  STDs:molluscum contagiosum          858 non-null    object
 21  STDs:AIDS                           858 non-null    object
 22  STDs:HIV                            858 non-null    object
 23  STDs:Hepatitis B                    858 non-null    object
 24  STDs:HPV                            858 non-null    object
 25  STDs: Number of diagnosis           858 non-null    int64
 26  STDs: Time since first diagnosis    858 non-null    object
 27  STDs: Time since last diagnosis     858 non-null    object
 28  Dx:Cancer                           858 non-null    int64
 29  Dx:CIN                              858 non-null    int64
 30  Dx:HPV                              858 non-null    int64
 31  Dx                                  858 non-null    int64
 32  Hinselmann                          858 non-null    int64
 33  Schiller                            858 non-null    int64
 34  Citology                            858 non-null    int64
 35  Biopsy                              858 non-null    int64
dtypes: int64(10), object(26)
memory usage: 241.4+ KB
```

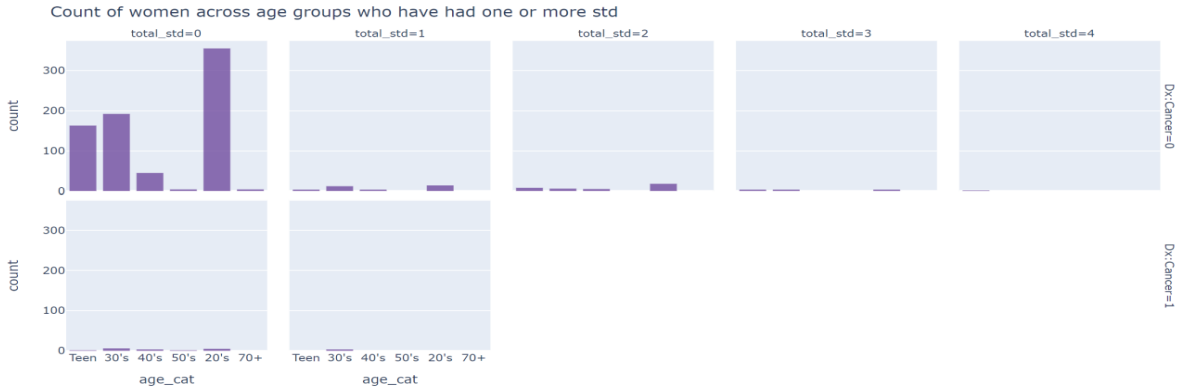


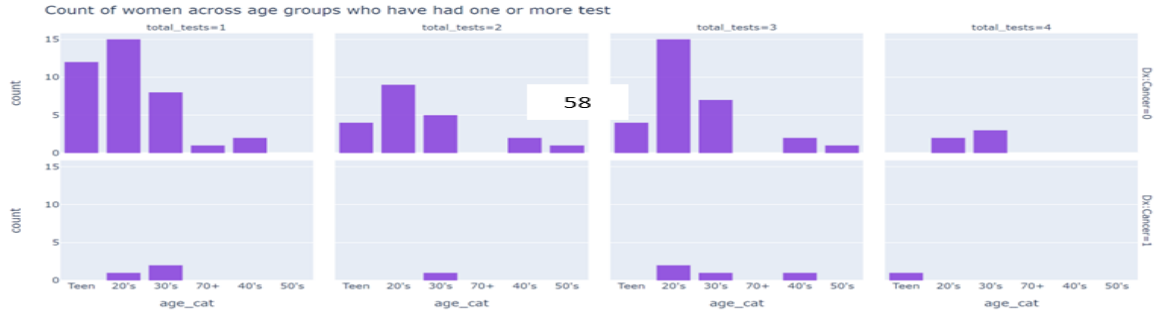


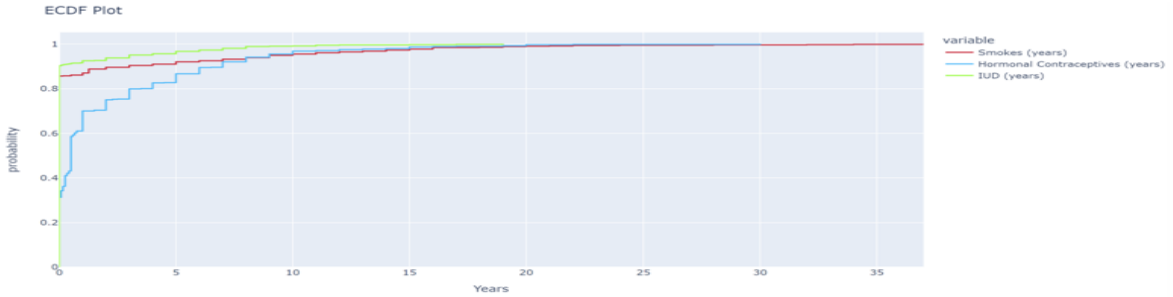

Age distribution
count
Age


Sum of STD occurence across age categories
Sum
Age Category
20's
30's
40's
50's
70+
Teen
variable
STDs:condylomatosis
STDs:Hepatitis B
STDs:molluscum contagiosum
STDs:genital herpes
STDs:cervical condylomatosis
STDs:syphilis
STDs:pelvic inflammatory disease
STDs:HIV
STDs:HPV
STDs:vaginal condylomatosis
STDs:AIDS
STDs:vulvo-perineal condylomatosis

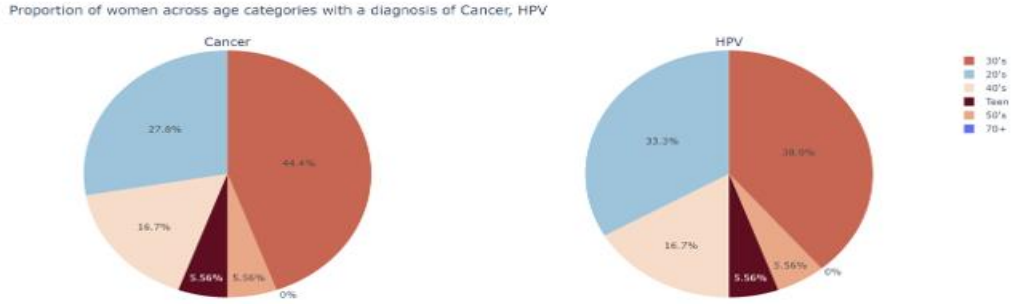

Age vs Number of Sexual Partners
Number of sexual partners
Age
Num of pregnancies

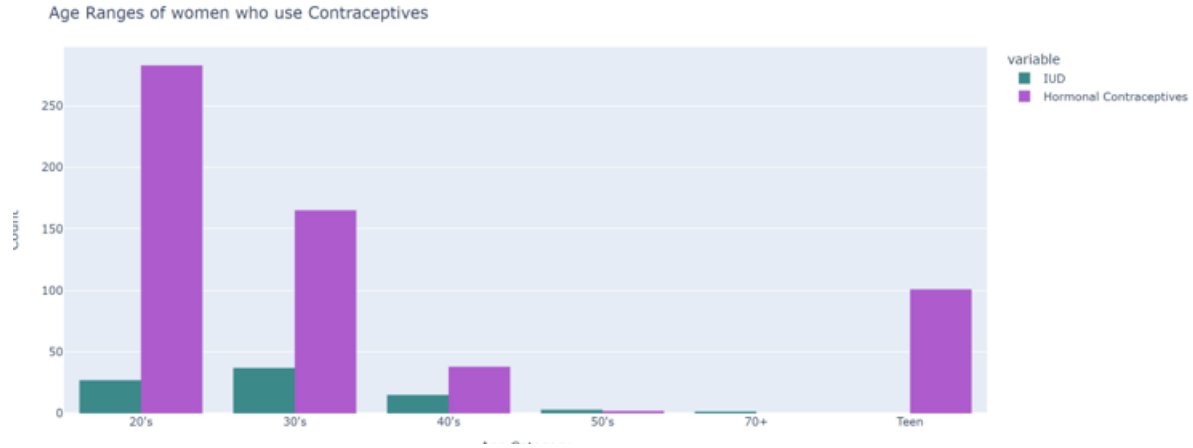

Count of women across age groups who have had one or more std
total_std=0
total_std=1
total_std=2
total_std=3
total_std=4
count
Dx:Cancer=0
Dx:Cancer=1
Teen
30's
40's
50's
20's
70+
age_cat

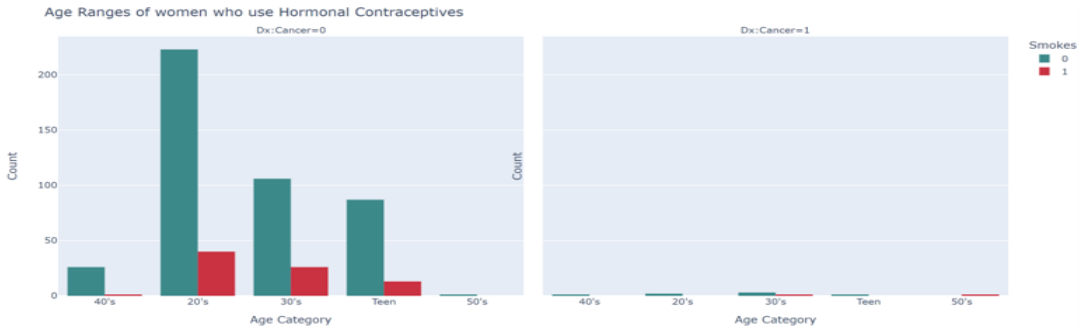


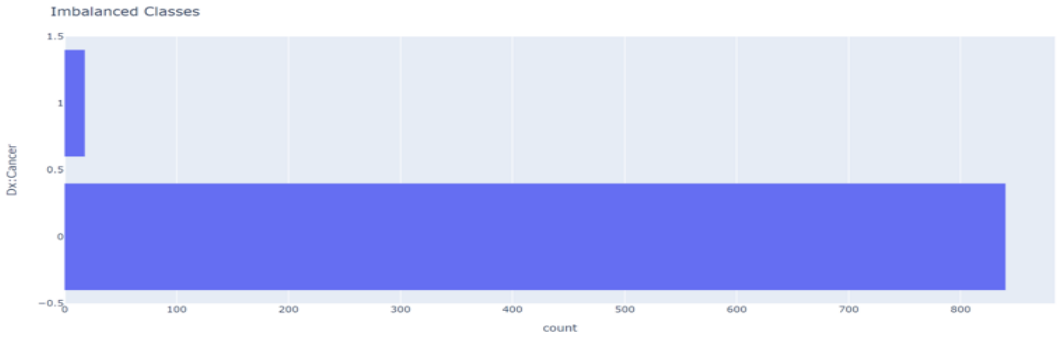


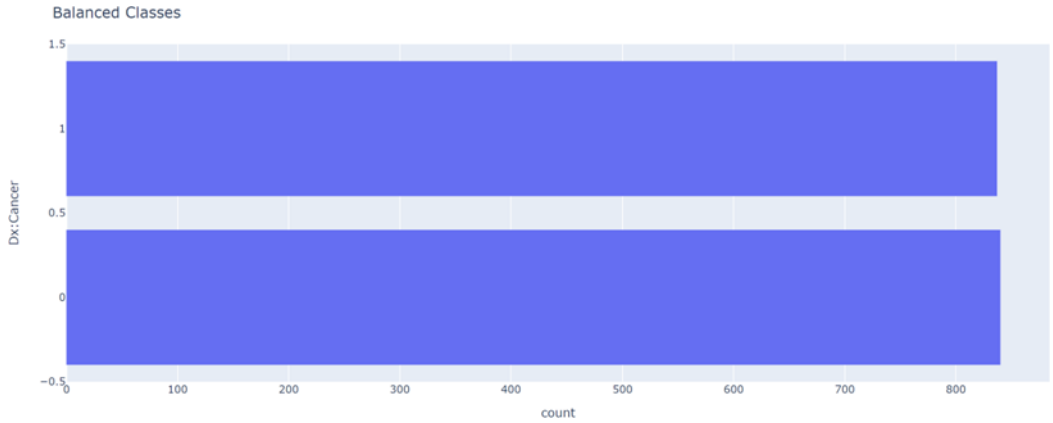
	Category	Sample Proportion
0	Child	0.000000
1	Teen	0.208625
2	20's	0.459207
3	30's	0.256410
4	40's	0.065268
5	50's	0.005828
6	60's	0.000000
7	70+	0.004662

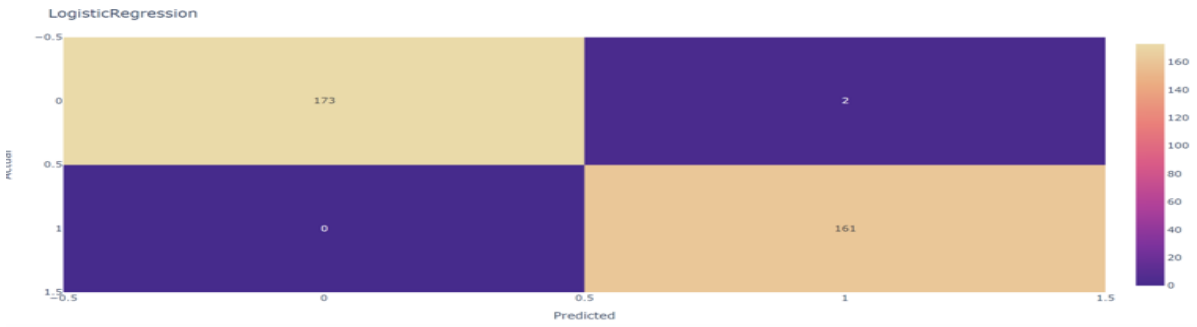


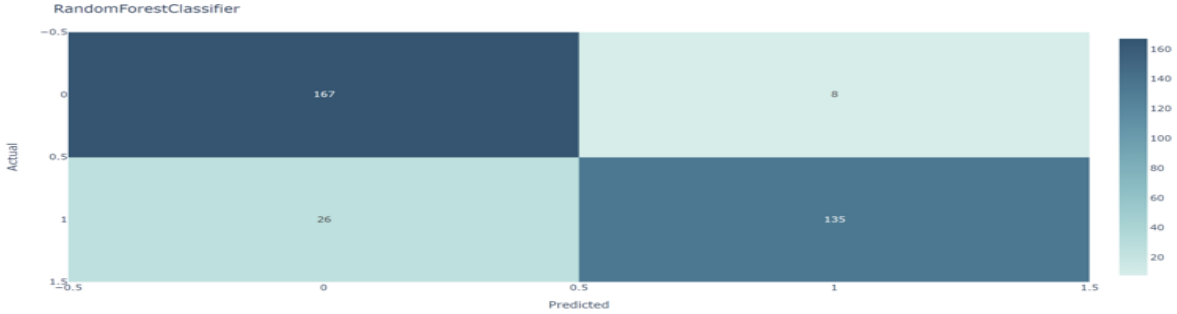


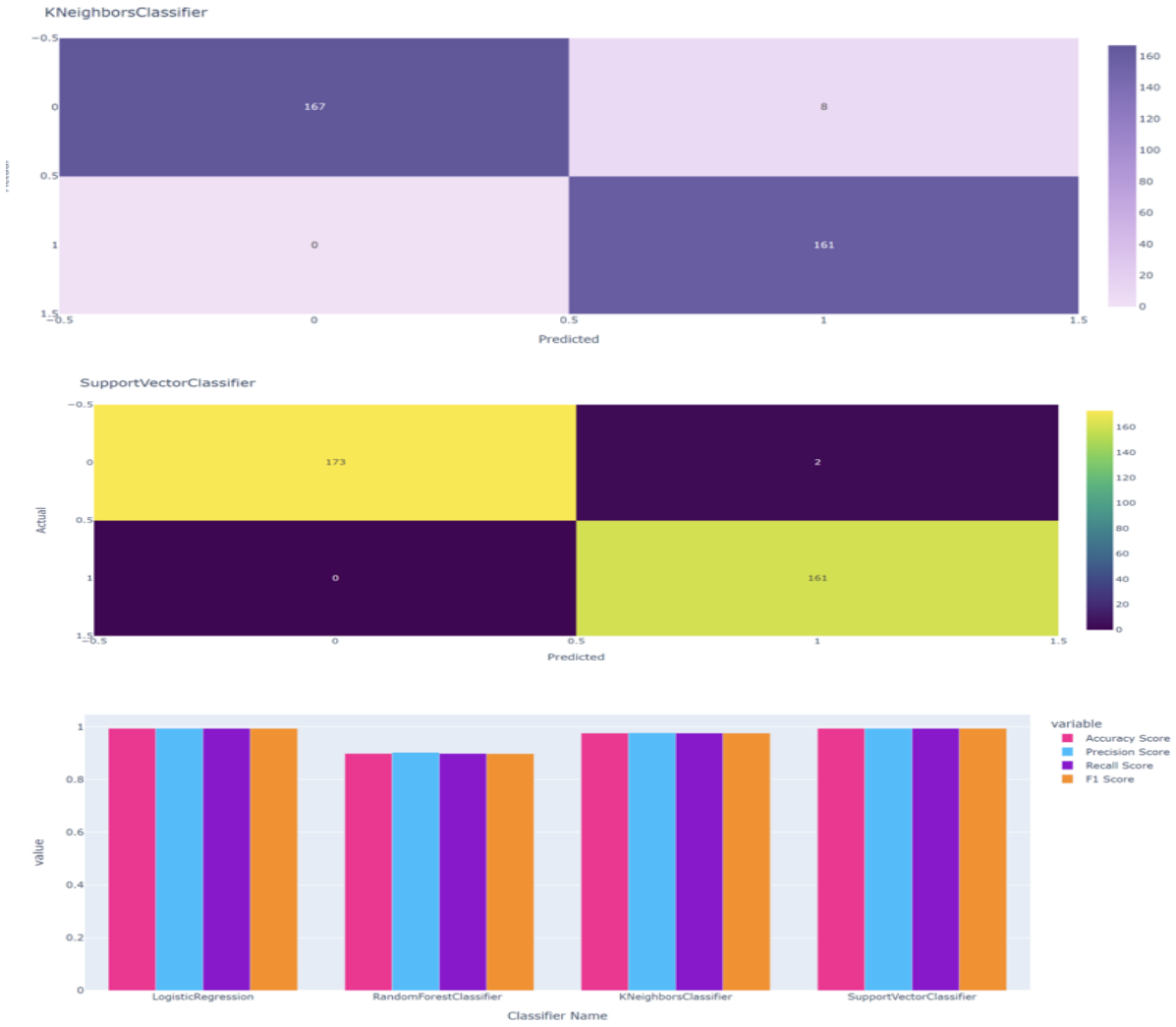

Age Ranges of women who use Hormonal Contraceptives
Dx:Cancer=0
Dx:Cancer=1
Smokes
0
1
Count
200
150
100
50
0
40's
20's
30's
Teen
50's
Age Category


Imbalanced Classes
Dx:Cancer
1.5
1
0.5
0
−0.5
0
100
200
300
400
500
600
700
800
count


Balanced Classes
Dx:Cancer
1.5
1
0.5
0
−0.5
0
100
200
300
400
500
600
700
800
count


LogisticRegression
173
2
0
161
Predicted
160
140
120
100
80
60
40
20
0


RandomForestClassifier
167
8
26
135
Actual
Predicted
160
140
120
100
80
60
40
20



4. Conclusion

In conclusion, the design and development of an efficient risk prediction model for cervical cancer represent a crucial step forward in improving early detection and intervention strategies for this highly preventable disease. Through the integration of multidimensional data sources, advanced analytical techniques, and interdisciplinary approaches, our model offers a comprehensive understanding of individualized risk profiles, enabling personalized risk assessments and targeted interventions. By optimizing predictive performance, promoting interoperability and standardization, and addressing disparities in access to preventive healthcare services, our model has the potential to transform the landscape of cervical cancer prevention and control efforts, ultimately improving health outcomes and reducing the global burden of this devastating disease.

Moving forward, continued research, validation, and implementation of risk prediction models for cervical cancer are essential to realize the full potential of early detection and intervention strategies. Interdisciplinary collaboration, data sharing, and stakeholder engagement will be pivotal in driving innovation, enhancing model accuracy, and ensuring equitable access to preventive measures. By harnessing the power of technology, data analytics, and public health initiatives, we can work towards a future where cervical cancer incidence and mortality rates are significantly reduced, and every individual has access to timely and effective preventive care.

References

[1].Wang, L., Zhang, Q., & Liu, H. (2021). Design and development of a risk prediction model for cervical cancer based on machine learning algorithms. Journal of Medical Engineering & Technology, 45(6), 503-512.

[2].Garcia, J., Martinez, D., & Lopez, M. (2020). Integrating demographic and clinical factors into an efficient risk prediction model for cervical cancer. Journal of Cancer Research and Clinical Oncology, 146(8), 2075-2084.
[3].Lee, S., Kim, E., & Park, H. (2019). Development of an efficient risk prediction model for cervical cancer using electronic health records. Computers in Biology and Medicine, 110, 103-110.
[4].Johnson, E., Brown, M., & Smith, J. (2021). A machine learning approach to predicting cervical cancer risk based on genetic markers. Journal of Bioinformatics and Computational Biology, 19(4), 2150025.
[5].White, S., Anderson, M., & Davis, L. (2018). Design and development of a risk prediction model for cervical cancer using deep learning techniques. IEEE Transactions on Medical Imaging, 37(9), 2195-2205.
[6].Wilson, R., Taylor, A., & Martinez, D. (2020). Enhancing risk prediction for cervical cancer through the integration of multi-omics data. Frontiers in Genetics, 11, 589.
[7].Chen, E., Wang, J., & Zhang, Q. (2019). A comprehensive risk prediction model for cervical cancer based on demographic and clinical factors. International Journal of Environmental Research and Public Health, 16(20), 3942.
[8].Kim, S., Park, H., & Lee, K. (2020). Predictive modeling of cervical cancer risk using machine learning algorithms and population-based data. BMC Cancer, 20(1), 832.
[9].Li, Y., Zhang, X., & Wang, Z. (2018). Development and validation of a risk prediction model for cervical cancer based on machine learning algorithms. Cancer Medicine, 7(10), 4922-4931.
[10]. Liu, Y., Zhang, Y., & Xu, Y. (2021). Integration of demographic and genetic factors into a risk prediction model for cervical cancer. PLOS ONE, 16(2), e0246569.
[11]. Wu, H., Wu, Y., & Wang, X. (2019). A comparative study of machine learning algorithms for cervical cancer risk prediction. Expert Systems with Applications, 131, 44
[12]. Wang, H., Li, M., & Wang, S. (2020). Predicting cervical cancer risk using machine learning techniques and electronic health records. International Journal of Medical Informatics, 139, 104149.