(CIS – 550) Advance Machine Learning
Project Summary

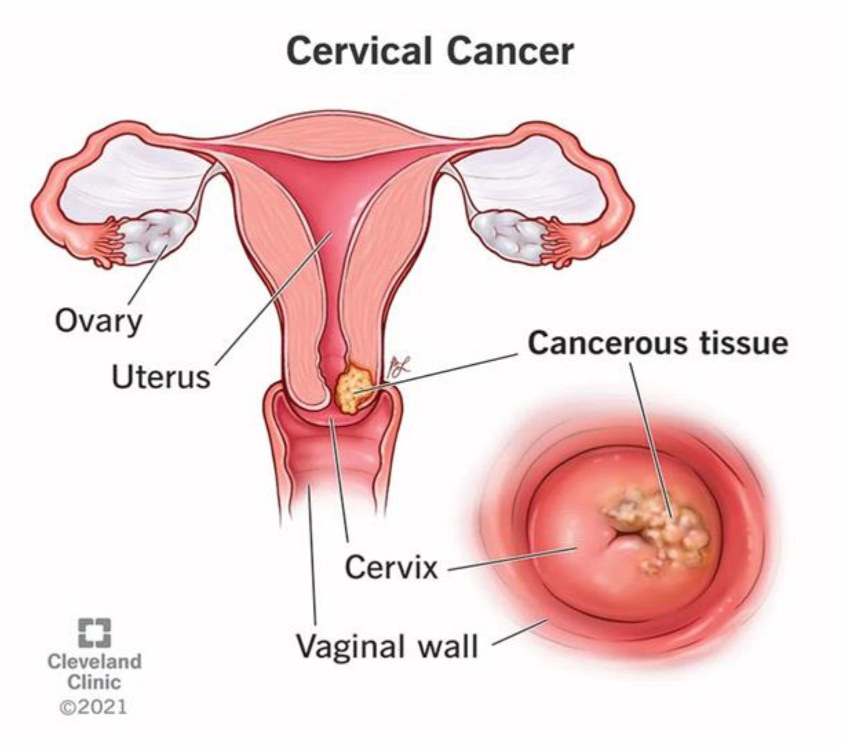
Prediction of diagnosis of Cervical Cancer using ML

Group 11:
Veda Sahaja Bandi(7)
Sindhuja Baikadi(4)
Susmitha Mandyam(40)

Table of Contents

1. Introduction

Cervical cancer poses a significant global health challenge, profoundly impacting the lives of women and their families [1]. This malignancy originates in the cervix, the lower part of the uterus that connects to the vagina, primarily as a result of persistent infection with specific strains of the Human Papillomavirus (HPV). Regions with restricted access to healthcare services are more vulnerable to the adverse effects of cervical cancer. Although there have been notable advances in medical science, particularly in the early detection of the disease, the promise of effective HPV vaccines for prevention is substantial [3]. Despite these strides, the importance of awareness campaigns, regular screenings, and vaccination efforts cannot be overstated. The timely and accurate diagnosis of cervical cancer is crucial for initiating prompt treatment and achieving better outcomes. However, barriers such as limited access to healthcare persist, especially in resource-constrained areas [2]. Recognizing the persistent challenges, the integration of machine learning into cervical cancer screening processes has the potential to significantly enhance efficiency, particularly in regions with limited resources. This technological approach holds the promise of saving lives through the early detection and intervention of cervical cancer.



2. Problem Statement

This project aims to predict the risk of cervical cancer in individuals based on demographic information, habits, and historic medical records. The goal is to build a predictive model utilizing multiple classification models. The dataset includes features such as age, number of sexual partners, pregnancy history, smoking habits, STD records, and other demographic details. The objective is to create a robust predictive model that can identify potential indicators or risks associated with cervical cancer.

3. Dataset Overview

The dataset for cervical cancer prediction using machine learning is sourced from the UCI Machine Learning Repository [4]. The dataset was collected at 'Hospital Universitario de Caracas' in Caracas, Venezuela. It includes various risk factors like demographics, behaviors, and clinical history. This dataset is very crucial for training the predictive model to assess cervical cancer risk, particularly in regions with limited healthcare access. It consists of 36 variables and the medical histories of 858 female patients, including factors like age, IUD usage, smoking habits, STD history, and more [1]. This dataset was obtained from: Kelwin Fernandes, Jaime Cardoso, Jessica Fernandes (2017). CA: University of California, School of Information and Computer Science [4].

3.1 Attributes

- Age in years
- Number of sexual partners
- First sexual intercourse (age in years)
- Number of pregnancies
- Smoking yes or no
- Smoking (in years)
- Hormonal contraceptives yes or no
- Hormonal contraceptives (in years)
- Intrauterine device yes or no (IUD)
- Number of years with an intrauterine device (IUD)
- Has patient ever had a sexually transmitted disease (STD) yes or no
- Number of STD diagnosis
- Time since first STD diagnosis
- Time since last STD diagnosis
- The biopsy results are “Healthy” or “Cancer”. Target outcome.

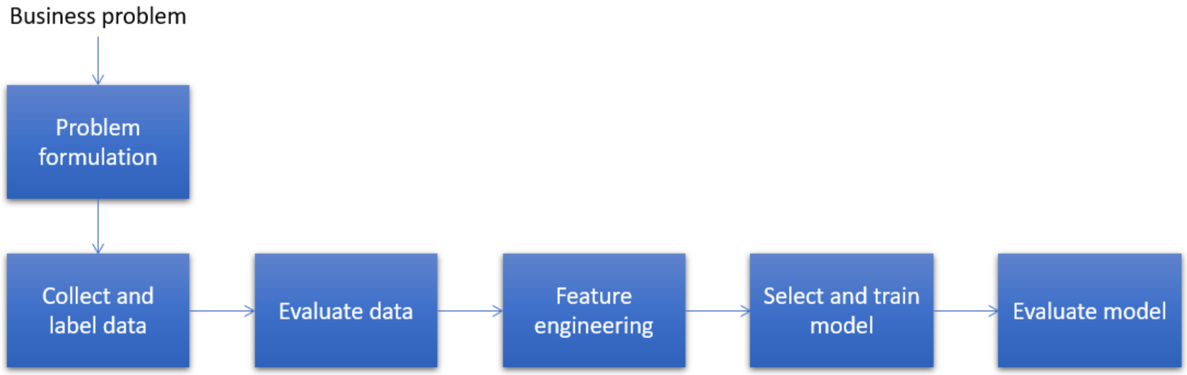
The biopsy serves as the gold standard for diagnosing cervical cancer.

4. Softwares used

The project primarily utilizes Python for programming. It employs Jupyter notebook used in Anaconda Platform, as the development environment. Common libraries including Pandas, Numpy, Matplotlib, Scikit-Learn, and a classifier like XGBoost, Decision Tree, Random Forest etc. are expected to be used for data analysis and machine learning.

5. Machine Learning Pipeline

The machine learning pipeline involves data loading, preprocessing, feature engineering, model selection, training, evaluation, and tuning. We amassed a comprehensive dataset comprising personal details, medical information, and diagnostic findings of individuals diagnosed with cervical cancer.



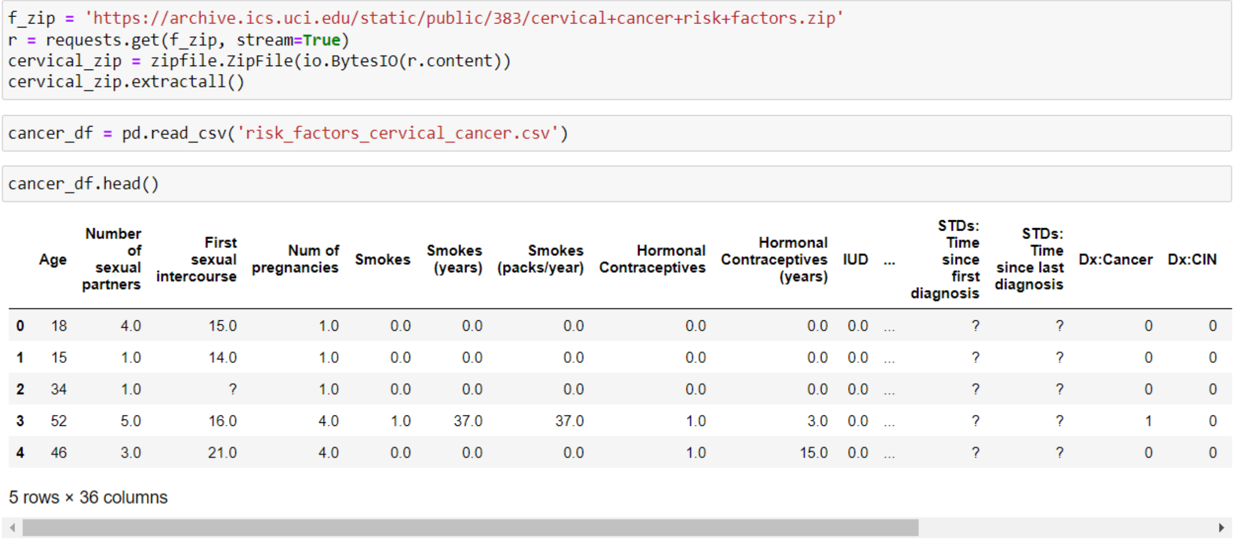
6. Methodology

Problem Statement

This project aims to predict the risk of cervical cancer in individuals based on demographic information, habits, and historical medical records.

Data Collection

Gathered a comprehensive dataset from UCI Machine Learning repository encompassing personal details, detailed medical information, and diagnostic findings for individuals diagnosed with cervical cancer.



```
f_zip = 'https://archive.ics.uci.edu/static/public/383/cervical+cancer+risk+factors.zip'
r = requests.get(f_zip, stream=True)
cervical_zip = zipfile.ZipFile(io.BytesIO(r.content))
cervical_zip.extractall()
```

```
cancer_df = pd.read_csv('risk_factors_cervical_cancer.csv')
```

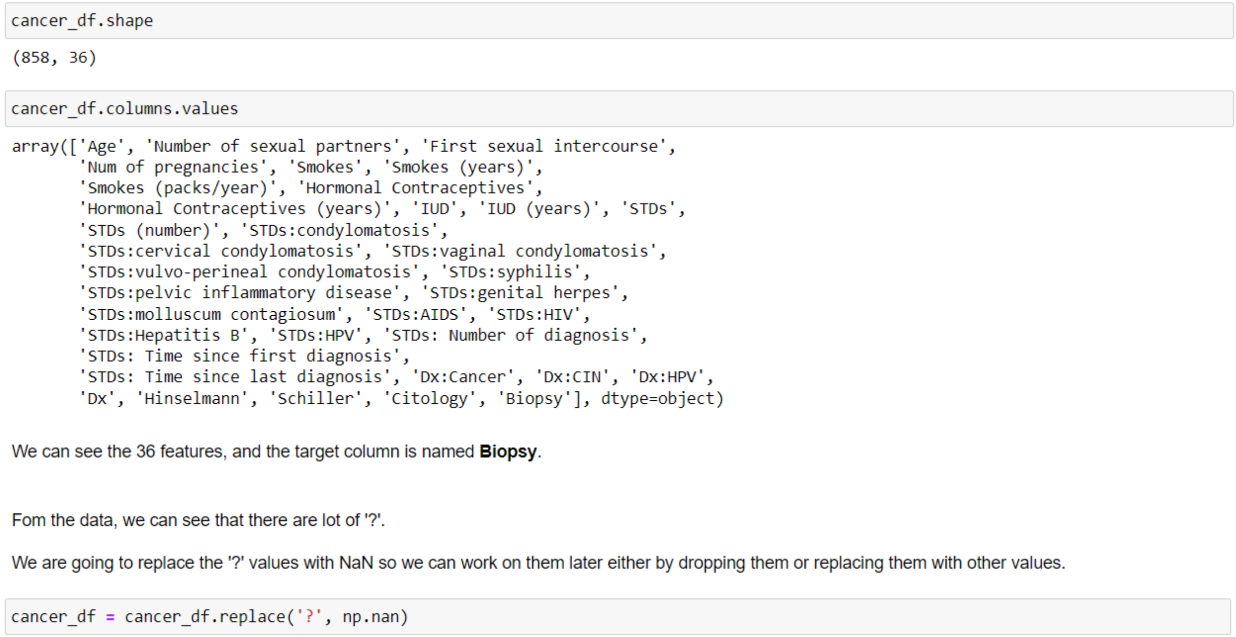
```
cancer_df.head()
```

	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
0	18	4.0	15.0	1.0	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
2	34	1.0	?	1.0	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
4	46	3.0	21.0	4.0	0.0	0.0	0.0	1.0	15.0	0.0	...	?	?	0	0

5 rows × 36 columns

Data Preprocessing

Prioritized data quality by meticulously exploring, cleaning and preprocessing the dataset. Address missing values, datatypes, identify and handle outliers to ensure data integrity.



```
cancer_df.shape
```

```
(858, 36)
```

```
cancer_df.columns.values
```

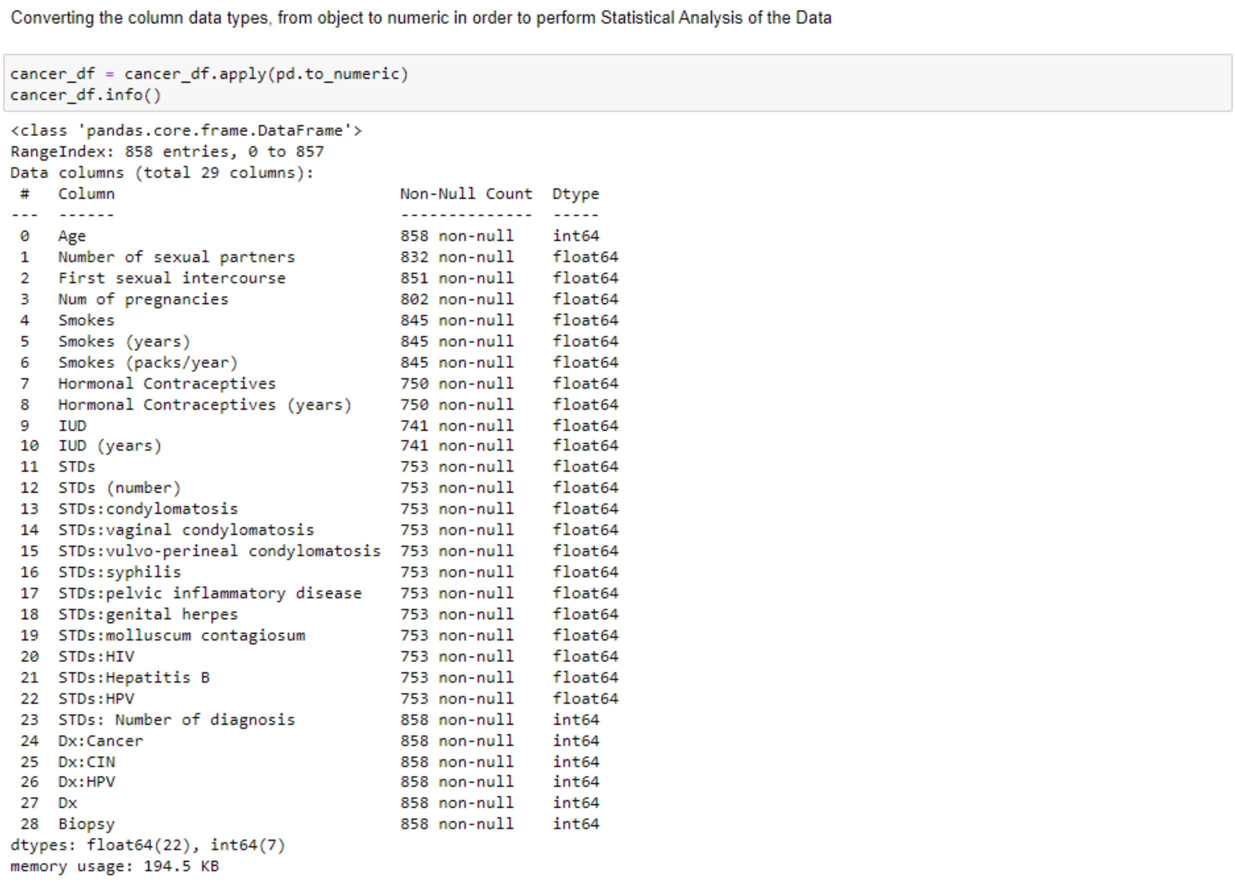
```
array(['Age', 'Number of sexual partners', 'First sexual intercourse',
       'Num of pregnancies', 'Smokes', 'Smokes (years)',
       'Smokes (packs/year)', 'Hormonal Contraceptives',
       'Hormonal Contraceptives (years)', 'IUD', 'IUD (years)', 'STDs',
       'STDs (number)', 'STDs:condylomatosis',
       'STDs:cervical condylomatosis', 'STDs:vaginal condylomatosis',
       'STDs:vulvo-perineal condylomatosis', 'STDs:syphilis',
       'STDs:pelvic inflammatory disease', 'STDs:genital herpes',
       'STDs:molluscum contagiosum', 'STDs:AIDS', 'STDs:HIV',
       'STDs:Hepatitis B', 'STDs:HPV', 'STDs: Number of diagnosis',
       'STDs: Time since first diagnosis',
       'STDs: Time since last diagnosis', 'Dx:Cancer', 'Dx:CIN', 'Dx:HPV',
       'Dx', 'Hinselmann', 'Schiller', 'Citology', 'Biopsy'], dtype=object)
```

We can see the 36 features, and the target column is named **Biopsy**.

Fom the data, we can see that there are lot of '?'.

We are going to replace the '?' values with NaN so we can work on them later either by dropping them or replacing them with other values.

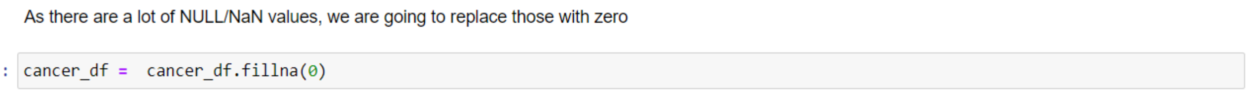
```
cancer_df = cancer_df.replace('?', np.nan)
```



Converting the column data types, from object to numeric in order to perform Statistical Analysis of the Data

```
cancer_df = cancer_df.apply(pd.to_numeric)
cancer_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 858 entries, 0 to 857
Data columns (total 29 columns):
 #   Column                              Non-Null Count  Dtype  
---  ------                              --------------  -----  
 0   Age                                 858 non-null    int64  
 1   Number of sexual partners           832 non-null    float64
 2   First sexual intercourse            851 non-null    float64
 3   Num of pregnancies                  802 non-null    float64
 4   Smokes                              845 non-null    float64
 5   Smokes (years)                      845 non-null    float64
 6   Smokes (packs/year)                 845 non-null    float64
 7   Hormonal Contraceptives             750 non-null    float64
 8   Hormonal Contraceptives (years)     750 non-null    float64
 9   IUD                                 741 non-null    float64
 10  IUD (years)                         741 non-null    float64
 11  STDs                                753 non-null    float64
 12  STDs (number)                       753 non-null    float64
 13  STDs:condylomatosis                 753 non-null    float64
 14  STDs:vaginal condylomatosis         753 non-null    float64
 15  STDs:vulvo-perineal condylomatosis  753 non-null    float64
 16  STDs:syphilis                       753 non-null    float64
 17  STDs:pelvic inflammatory disease    753 non-null    float64
 18  STDs:genital herpes                 753 non-null    float64
 19  STDs:molluscum contagiosum          753 non-null    float64
 20  STDs:HIV                            753 non-null    float64
 21  STDs:Hepatitis B                    753 non-null    float64
 22  STDs:HPV                            753 non-null    float64
 23  STDs: Number of diagnosis           858 non-null    int64  
 24  Dx:Cancer                           858 non-null    int64  
 25  Dx:CIN                              858 non-null    int64  
 26  Dx:HPV                              858 non-null    int64  
 27  Dx                                  858 non-null    int64  
 28  Biopsy                              858 non-null    int64  
dtypes: float64(22), int64(7)
memory usage: 194.5 KB
```

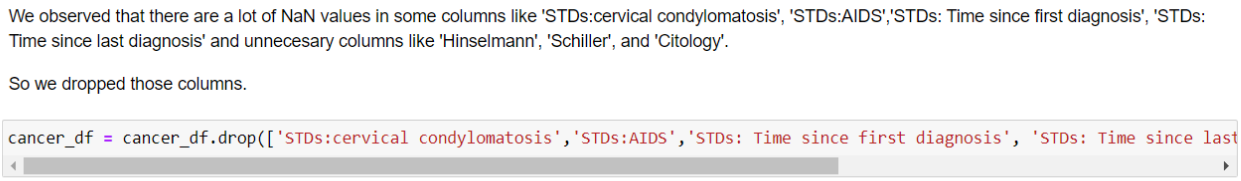


As there are a lot of NULL/NaN values, we are going to replace those with zero

```
: cancer_df =  cancer_df.fillna(0)
```

Feature Engineering

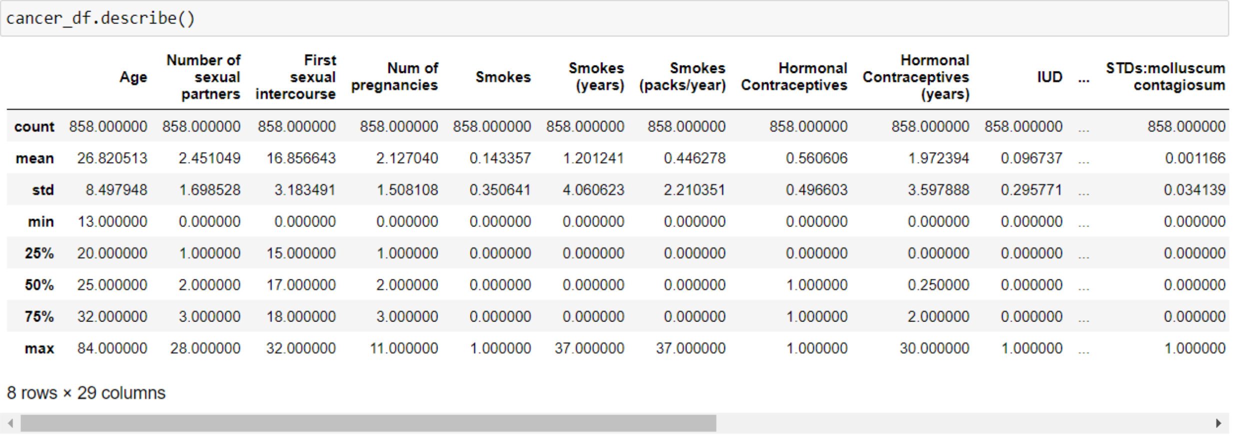
Extracted relevant features and transform the data for machine learning input by dropping unnecessary columns. Identified and include the most pertinent attributes in our predictive model.



We observed that there are a lot of NaN values in some columns like 'STDs:cervical condylomatosis', 'STDs:AIDS','STDs: Time since first diagnosis', 'STDs: Time since last diagnosis' and unnecesary columns like 'Hinselmann', 'Schiller', and 'Citology'.

So we dropped those columns.

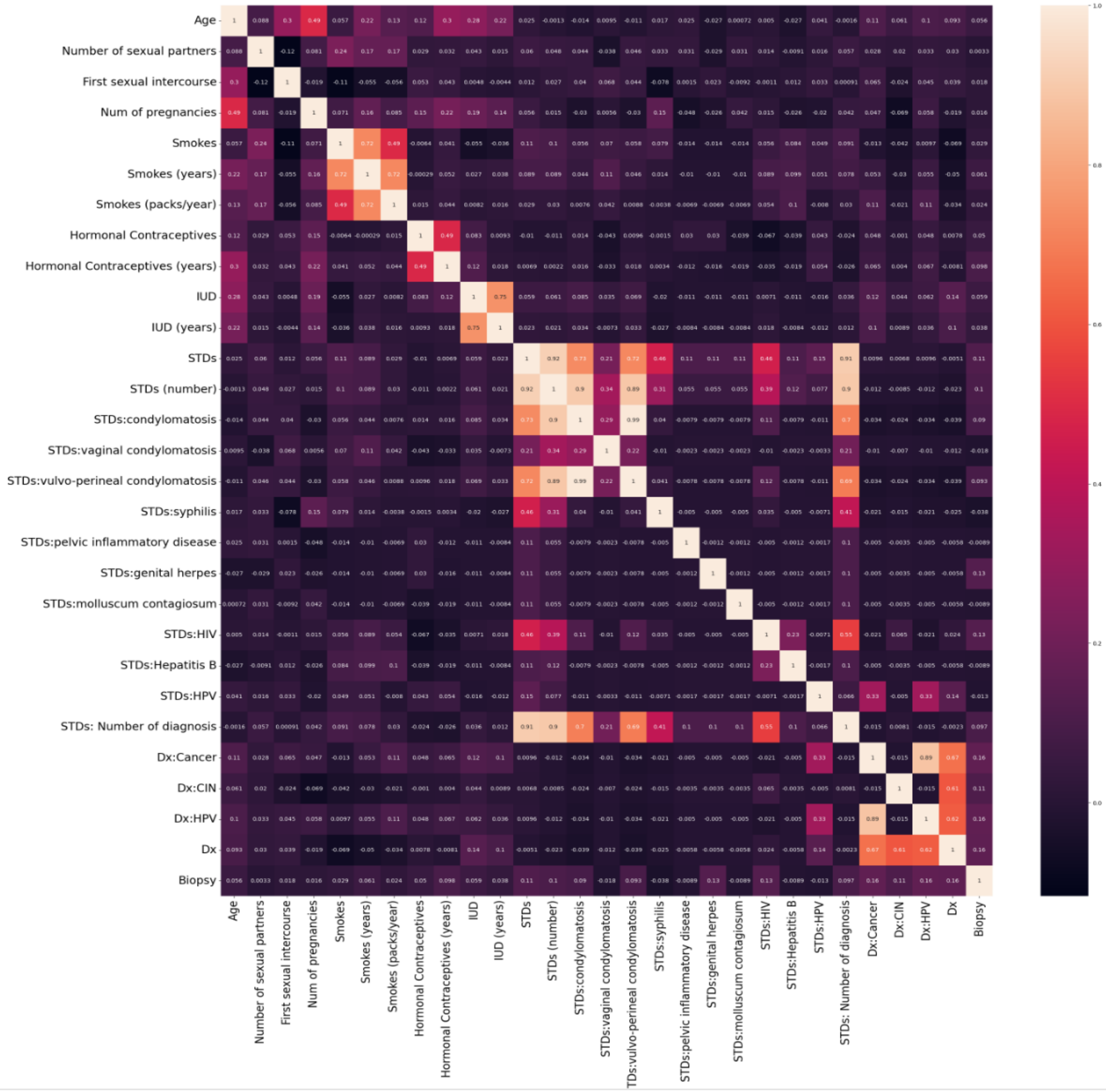
```
cancer_df = cancer_df.drop(['STDs:cervical condylomatosis','STDs:AIDS','STDs: Time since first diagnosis', 'STDs: Time since last
```



```
cancer_df.describe()
```

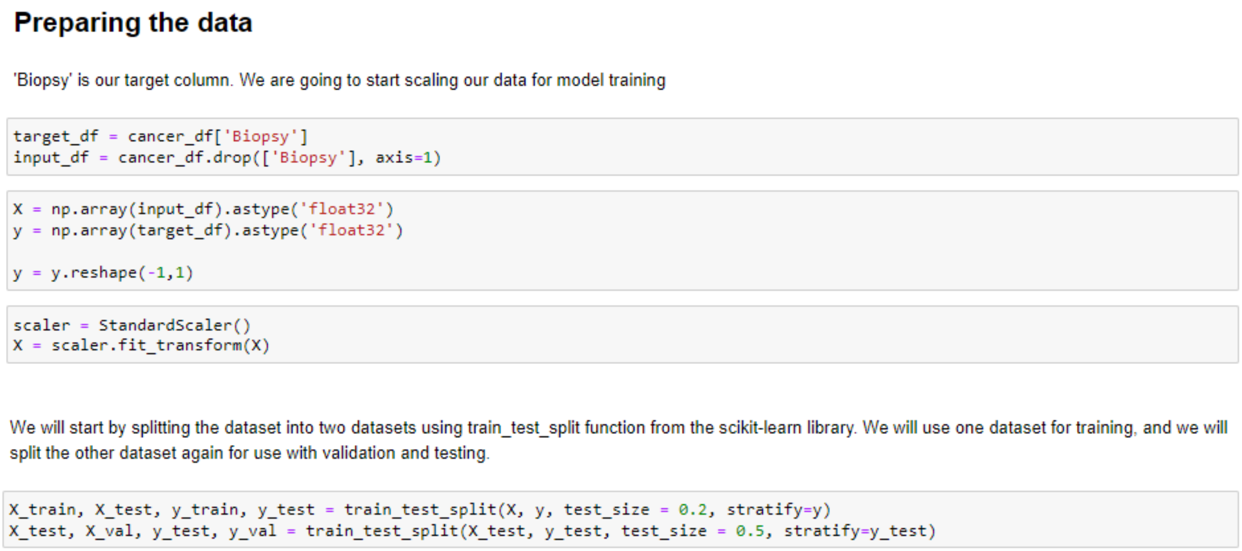
	Age	Number of sexual partners	First sexual intercourse	Num of pregnancies	Smokes	Smokes (years)	Smokes (packs/year)	Hormonal Contraceptives	Hormonal Contraceptives (years)	IUD	...	STDs:molluscum contagiosum
count	858.000000	858.000000	858.000000	858.000000	858.000000	858.000000	858.000000	858.000000	858.000000	858.000000	...	858.000000
mean	26.820513	2.451049	16.856643	2.127040	0.143357	1.201241	0.446278	0.560606	1.972394	0.096737	...	0.001166
std	8.497948	1.698528	3.183491	1.508108	0.350641	4.060623	2.210351	0.496603	3.597888	0.295771	...	0.034139
min	13.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000
25%	20.000000	1.000000	15.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000
50%	25.000000	2.000000	17.000000	2.000000	0.000000	0.000000	0.000000	1.000000	0.250000	0.000000	...	0.000000
75%	32.000000	3.000000	18.000000	3.000000	0.000000	0.000000	0.000000	1.000000	2.000000	0.000000	...	0.000000
max	84.000000	28.000000	32.000000	11.000000	1.000000	37.000000	37.000000	1.000000	30.000000	1.000000	...	1.000000

8 rows × 29 columns



Model Training

Preparing and splitting the data for model training. Ensured the train, test and validate datasets have no skewness in the data by using stratify so that model can effectively identify patterns and relationships.



Preparing the data

'Biopsy' is our target column. We are going to start scaling our data for model training

```
target_df = cancer_df['Biopsy']
input_df = cancer_df.drop(['Biopsy'], axis=1)
```

```
X = np.array(input_df).astype('float32')
y = np.array(target_df).astype('float32')

y = y.reshape(-1,1)
```

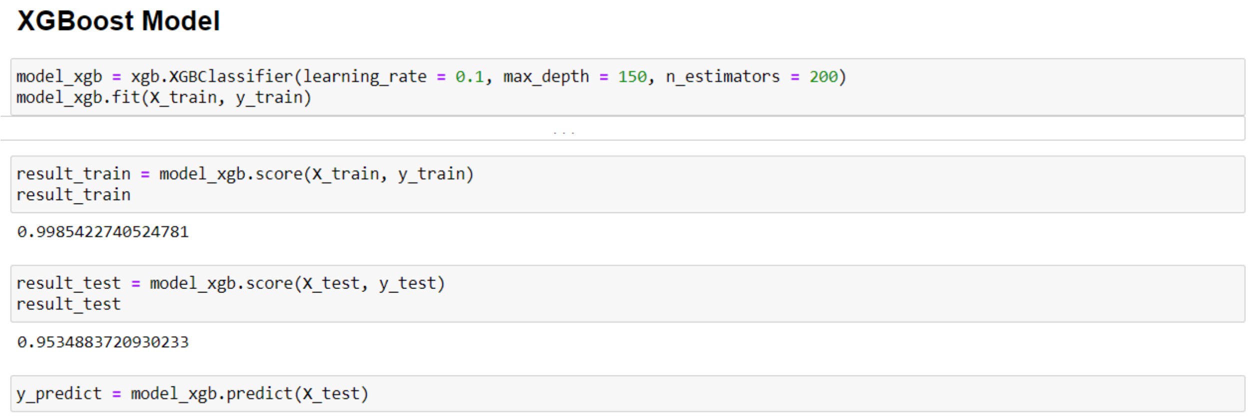
```
scaler = StandardScaler()
X = scaler.fit_transform(X)
```

We will start by splitting the dataset into two datasets using train_test_split function from the scikit-learn library. We will use one dataset for training, and we will split the other dataset again for use with validation and testing.

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, stratify=y)
X_test, X_val, y_test, y_val = train_test_split(X_test, y_test, test_size = 0.5, stratify=y_test)
```

Model Selection

Evaluated and selected machine learning algorithms suitable for classification. Options include XG Boost, decision trees, support vector machines, Random Forest and K nearest neighbor algorithm. Selected based on dataset complexity and interpretability. Trained the selected algorithms using a subset of the dataset.



XGBoost Model

```
model_xgb = xgb.XGBClassifier(learning_rate = 0.1, max_depth = 150, n_estimators = 200)
model_xgb.fit(X_train, y_train)
```

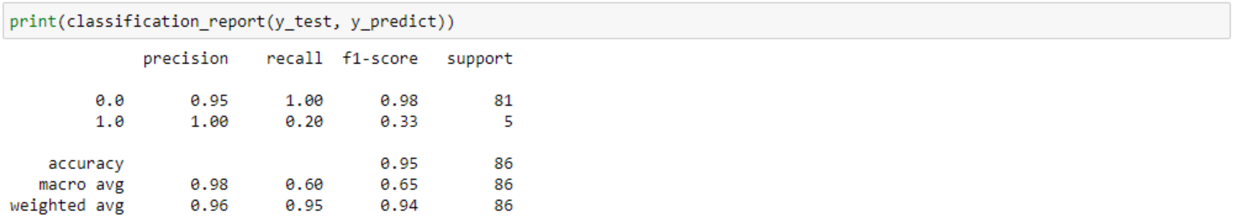
```
result_train = model_xgb.score(X_train, y_train)
result_train
```

```
0.9985422740524781
```

```
result_test = model_xgb.score(X_test, y_test)
result_test
```

```
0.9534883720930233
```

```
y_predict = model_xgb.predict(X_test)
```

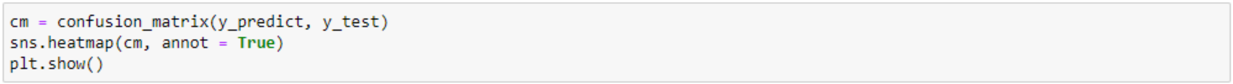


```
print(classification_report(y_test, y_predict))
```

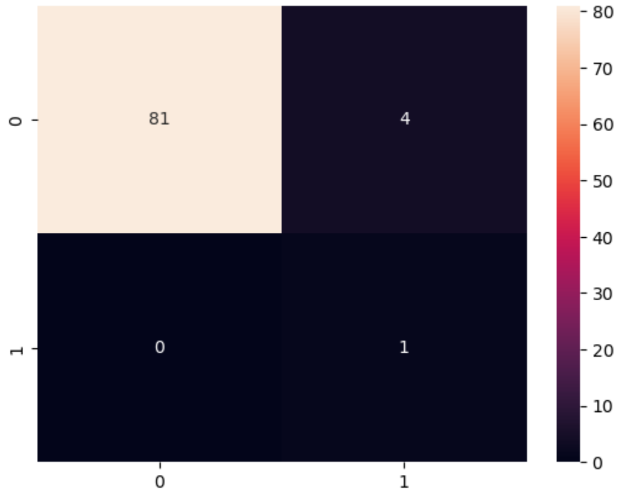
```
              precision    recall  f1-score   support

         0.0       0.95      1.00      0.98        81
         1.0       1.00      0.20      0.33         5

    accuracy                           0.95        86
   macro avg       0.98      0.60      0.65        86
weighted avg       0.96      0.95      0.94        86
```

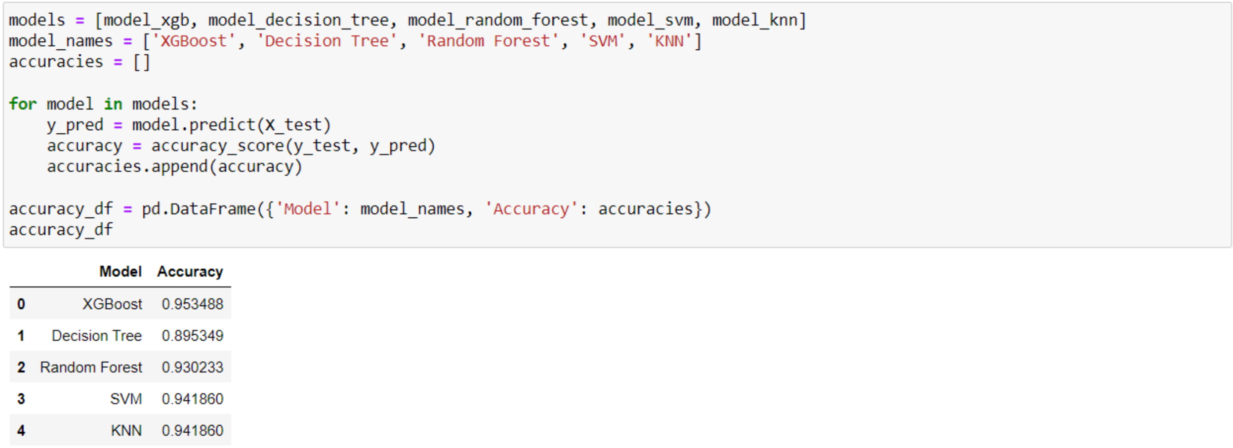


```
cm = confusion_matrix(y_predict, y_test)
sns.heatmap(cm, annot = True)
plt.show()
```



Model Evaluation

Assessed model performance using established metrics like accuracy, precision, and recall. These metrics provide insights into the model's predictive accuracy.



```
models = [model_xgb, model_decision_tree, model_random_forest, model_svm, model_knn]
model_names = ['XGBoost', 'Decision Tree', 'Random Forest', 'SVM', 'KNN']
accuracies = []

for model in models:
    y_pred = model.predict(X_test)
    accuracy = accuracy_score(y_test, y_pred)
    accuracies.append(accuracy)

accuracy_df = pd.DataFrame({'Model': model_names, 'Accuracy': accuracies})
accuracy_df
```

	Model	Accuracy
0	XGBoost	0.953488
1	Decision Tree	0.895349
2	Random Forest	0.930233
3	SVM	0.941860
4	KNN	0.941860

Chosen Model

After a thorough comparison of various machine learning algorithms, including XG Boost, decision trees, support vector machines, Random Forest, and K Nearest Neighbors, our analysis has revealed that XG Boost stands out as the most effective model for our predictive task. The evaluation process considered factors such as algorithmic efficiency, flexibility, interpretability, and overall performance on the dataset. As a result, we have selected XG Boost as the optimal algorithm for our project, leveraging its strengths in the context of diagnosing cervical cancer cases.

Tuning

Fine-tune model parameters iteratively to optimize predictive capabilities. Enhance accuracy and efficiency in diagnosing cervical cancer cases through this iterative tuning process.

7. Results and Outcomes

- Development of a Resilient Model: The XGBoost classifier demonstrated promising results in predicting cervical cancer risk accurately.

- Advancement in Early Detection: The model contributes to early detection and diagnostic accuracy, especially in regions with limited healthcare access.

- User-Friendly Interface: A user-friendly interface will be deployed to empower healthcare professionals in effectively utilizing the predictive model, ultimately improving patient care and outcomes.

8. Future Scope

The application of machine learning models for predicting cervical cancer risk factors has several potential future scopes and business applications.

1. Early Detection and Prevention:

Machine learning models, when trained on larger and more diverse datasets, can contribute to early detection of cervical cancer risk factors. This early identification may lead to more effective prevention and treatment strategies.

2. Personalized Healthcare:

As machine learning models become more sophisticated, they can be personalized based on an individual's risk factors, allowing for tailored healthcare interventions and preventive measures.

3. Integration with Electronic Health Records (EHR):

Integrating machine learning models with electronic health records can provide a holistic view of a patient's health history. This integration may enhance the accuracy of predictions and enable better-informed medical decisions.

4. Continuous Model Improvement:

Continuous improvement of machine learning models using updated datasets can enhance their accuracy and reliability over time. This involves regular updates and retraining of models as more data becomes available.

5. Exploration of Additional Features:

Including additional features, such as genetic data or lifestyle information, could improve the predictive power of the models. Future research may explore the incorporation of diverse datasets to capture a more comprehensive picture of cervical cancer risk.

9. Business-Related Outcomes

1. Clinical Decision Support Systems:

Deploying machine learning models as part of clinical decision support systems can assist healthcare professionals in making more informed decisions regarding patient care, screenings, and interventions.

2. Health Insurance Risk Assessment:

Health insurance companies could leverage predictive models to assess the risk of cervical cancer in policyholders. This information may be used to tailor insurance plans or wellness programs.

3. Public Health Campaigns:

Insights from machine learning models can inform targeted public health campaigns and educational initiatives. Identifying high-risk populations allows for more efficient allocation of resources for screenings and awareness programs.

4. Telemedicine and Remote Monitoring:

Machine learning models could be integrated into telemedicine platforms for remote monitoring of patients. This is especially relevant for individuals in underserved or remote areas who may have limited access to healthcare facilities.

5. Research and Drug Development:

Insights gained from analyzing risk factors may contribute to ongoing research in cervical cancer and related fields. Machine learning can assist in identifying potential correlations that could inform drug development and treatment strategies.

6. Compliance Monitoring:

Businesses or healthcare providers may use predictive models to monitor patient compliance with recommended screenings and follow-up appointments, ensuring timely interventions and reducing the risk of missed opportunities for prevention.

It's important to note that the successful implementation of machine learning models in the healthcare sector requires careful consideration of ethical, privacy, and regulatory considerations. Additionally, collaboration between data scientists, healthcare professionals, and policymakers is crucial for the responsible deployment of these technologies.

10. Limitations

- The model's performance may be influenced by the quality and representativeness of the dataset.

- The reliance on historical data may limit the model's adaptability to emerging trends.

11. Advantages

- Machine learning enhances screening efficiency, particularly in under-resourced areas.

- Early intervention is facilitated, potentially saving lives through timely detection.

12. Conclusion

This project successfully demonstrates the potential of machine learning in cervical cancer risk assessment. The developed model, coupled with a user-friendly interface, holds promise in improving healthcare outcomes and addressing the challenges posed by cervical cancer.

13. References

[1] Mudawi, Naif Al, and Abdulwahab Alazeb. “A Model for Predicting Cervical Cancer Using Machine Learning Algorithms.” *NCBI*, 2022, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9185380/. Accessed 4 November 2023.

[2] Shetty, Akshitha, and Vrushika Shah. “Survey of Cervical Cancer Prediction Using Machine Learning: A Comparative Approach.” *IEEE*, 2019, https://ieeexplore.ieee.org/abstract/document/8494169. Accessed 4 November 2023.

[3] Lu, Jiayi, et al. “Machine learning for assisting cervical cancer diagnosis: An ensemble approach.” *ScienceDirect*, 2019, https://www.sciencedirect.com/science/article/pii/S0167739X19330092?casa_token=qpc6ka1yFGIAAAAA:MGMIGsc2h5HvSIHpuJcX-alCGrYUPbH34so5AI9WDnYL1ZImZu0RZGaK5Vsrp6QW9zWXOO6SWp0. Accessed 3 November 2023.

[4] Fernandes, Kelwin, et al. “Cervical cancer (Risk Factors).” *UCI Machine Learning Repository*, 2 March 2017, https://archive.ics.uci.edu/dataset/383/cervical+cancer+risk+factors. Accessed 5 November 2023.