ELEC 490/498 Final Report

Project Tile: Early Warning of Ovarian Cancer

Submitted By: Group 11

Date: 2024-04-08

Kieran Cosgrove - 20226841

Lucas Coster - 20223016

Matthew Mamelak - 20216737

Miodrag (Mile) Stosic - 20233349

Faculty Supervisor: Prof. Michael Korenberg

In association with: National Cancer Institute (NCI)

Table Of Contents

Table of Figures

Table of Tables

Table of Equations

Executive Summary

The "Early Warning of Ovarian Cancer" project is a ground-breaking initiative in medical diagnostics aimed at improving the early detection of ovarian cancer. It tackles the challenge of late diagnosis due to the asymptomatic nature of early-stage ovarian cancer. Using advanced machine learning techniques, the project develops a non-invasive predictive tool, significantly shifting the approach towards earlier and more effective treatment.

Leveraging the comprehensive Prostate, Lung, Colorectal, and Ovarian (PLCO) dataset, the project employs methods like class imbalance learning and utilizes models such as Random Forest, SVM, and Neural Networks. Special emphasis is placed on model transparency and comprehensibility. The most effective model identified was the Random Forest, enhanced with the Synthetic Minority Over-sampling Technique (SMOTE).

The project also includes the development of a user-friendly front-end application using Streamlit, focusing on ease of use, visual appeal, and accessibility. The design was iteratively refined based on user feedback to enhance usability.

Rigorous testing and evaluation of the machine learning model and the application revealed excellent performance in accuracy, recall, precision, and F1 score. Conducted within a planned budget and leveraging open-source resources, the project included extensive stakeholder considerations, covering ethical, legal, patient, and medical professional perspectives.

In summary, the "Early Warning of Ovarian Cancer" project represents a significant contribution to women's healthcare, merging sophisticated machine learning with practical application to advance early ovarian cancer detection.

1 Introduction

The "Early Warning of Ovarian Cancer" project represents a pivotal step in the field of medical diagnostics, specifically targeting the early detection of ovarian cancer. Ovarian cancer, often diagnosed in advanced stages due to its non-specific symptoms and the limitations of existing detection methods, presents a significant challenge in women's healthcare. Our project, driven by cutting-edge technological innovations, aims to disrupt this by introducing a more effective early detection system. The introduction of such a system has the potential to not only enhance diagnostic accuracy but also transform the overall approach to ovarian cancer treatment and management.

1.1 Problem Definition

The project focuses on a pressing issue in women's health: the lagging diagnosis of ovarian cancer due to current screening limitations. Ovarian cancer is often asymptomatic in its early stages and lacks effective early detection methods [1]. Typically, available diagnostic procedures, such as biopsies, are not only invasive but also fall short in sensitivity for early-stage cancer detection [1]. This disparity significantly hinders timely intervention and successful treatment outcomes.

In response, our project's ambition is to build a non-invasive, predictive tool designed to detect ovarian cancer's early indicators. By using machine learning algorithms, we aim to create a solution that enhances early detection, thus improving the likelihood of positive treatment results and increasing survival rates. Our tool aims to be a source of hope, potentially changing how we predict ovarian cancer.

Our project is designed to protect user safety and privacy, following strict data protection rules. Plus, the tool will carefully manage to identify true cancer cases accurately while minimizing the chances of mistakenly identifying someone as cancer-free when they are not.

In medical machine learning, a model's comprehensibility is crucial. Medical diagnostics extend beyond classification; they must offer insights that can guide timely and effective medical treatment. Our model avoids the 'black box' approach, opting for transparency that explicitly displays its decision-making process [2]. While symbolic learning methods like decision trees offer such clarity, they often do so at the expense of precision. Our solution strives to reconcile this trade-off, aiming for a high accuracy level and an intelligible model that medical professionals can trust.

Finally, the project highlights the importance of developing an algorithm that not only performs well but also conveys diagnostic insights understandably. It should enhance decision-making with reliable predictive probabilities and require minimal patient data, thereby reducing cost, time investment, and patient risk associated with data collection. This project, therefore, is situated at the intersection of technological innovation and medical advancement, aiming to address a critical need in ovarian cancer.

1.2 Motivation for the Project

Our endeavor engages the truth that early detection of ovarian cancer is essential to improving patient survival rates and quality of life. It is the linchpin that can significantly alter the trajectory from late-stage intervention to timely, less invasive, and more effective treatment options. The drive behind our project is multifaceted: to amplify women's health initiatives, spotlight an often-neglected health concern, reduce the high mortality rates of ovarian cancer, and fill a significant void in contemporary oncological practices.

Our dedication is intensified by the harsh truth that ovarian cancer is the most lethal of all the cancers affecting the female reproductive system. The survival statistics are shocking: less than 30% of women diagnosed with advanced-stage ovarian cancer survive in the long term [1]. In stark contrast, early-stage diagnosis—specifically stage 1—boasts up to a 90% cure rate post-surgery and chemotherapy [1]. Right now, only about one in four cases of ovarian cancer are caught early. This is often because the disease doesn't show clear signs until it's advanced, and there isn't a dependable way to screen for it early on [1].

Catching ovarian cancer early enough to save lives depends on a few key ideas: most ovarian cancers start in the ovaries and are made up of similar cells, the spread of the disease can be traced back to early-stage spots that we can find, and these cancers stay in one spot long enough for us to catch them with screening tests [1]. For a disease like this, our screening method must be good at picking it up in its early stages, but also specific enough so that when it indicates the disease is there, it's usually right.

1.3 Project Scope

The scope of this project extends across various domains, integrating data analysis, model development, and user interface design to address the challenge of early ovarian cancer detection. At the core of our approach is an in-depth exploration of the features most closely associated with ovarian cancer, such as ovulation rate. Leveraging the comprehensive PLCO dataset provided by the National Cancer Institute (NCI), supplemented by insights from interviews with a gynecologist and information from the Centers for Disease Control and Prevention, we have attempted to align our predictive model with the latest medical research on ovarian cancer. The project lays out how we built it and what it means to use different machine-learning models designed just for this task. Extra attention is given to models that make decisions and classify individually because they are good at dealing with complex datasets that do not have a balanced number of cases—something you often see in medical diagnosis, and even more so with ovarian cancer, which doesn't happen all that often.

A significant portion of the project is also dedicated to the development of a user-friendly front end. We have created a website that is accessible on all device formats, enabling individuals to assess their risk of ovarian cancer based on our refined model. This digital platform has been designed to adhere to the principles of public accessibility and non-commercialization, in line with the stipulations set forth by the NCI for the use of the PLCO dataset [3]. This initiative represents a bridge between sophisticated machine learning techniques and practical, user-centric applications, democratizing access to potentially life-saving information.

Furthermore, the project delves into a detailed analysis of the methodological considerations inherent in dealing with unbalanced datasets. Through a comprehensive review of sampling methods, algorithmic strategies, feature-level analysis, and the application of deep learning techniques, the paper discusses the challenges and potential solutions in classifying such datasets. Techniques like the SMOTE, Support Vector Machine (SVM), and K-Nearest Neighbor (KNN) are evaluated for their effectiveness in enhancing the model's sensitivity to rare positive cases without compromising overall accuracy.

Finally, we thoroughly checked our models using specific measures to see how well they work. This step is key to making sure they're good at predicting and can be trusted for medical or personal health checks. Our project not only tests the limits of machine learning in medical tests but also provides a strong new way to fight ovarian cancer.

1.4 Background & Model Terminology

Before discussing the specifics of our machine learning models and the intricacies of their development, it's important to understand the foundational terms and concepts that are pivotal to our project.

1.4.1 Class Imbalanced Learning

In the context of our project using the PLCO Cancer Screening Trial dataset, we encounter a significant challenge common in medical data analysis: class imbalance. The PLCO dataset, encompassing approximately 155,000 participants enrolled from November 1993 to July 2001, has proven its utility for machine learning models, particularly when analyzing biomarker data [3] [4]. Yet, when focusing on women's health records for ovarian cancer diagnosis, the dataset reveals a stark imbalance: only 1% of the records denote a positive diagnosis, illustrating a profound disparity between the majority (negative diagnoses) and minority (positive diagnoses) classes.

Class imbalance refers to the uneven distribution of classes within a dataset, where one class (the majority) significantly outnumbers the other (the minority). This imbalance is not just a numerical issue but a substantive one that can skew the performance of machine learning classifiers, leading them to favor the majority class and overlook the critical nuances of the minority class [4]. In the domain of ovarian cancer detection, where the minority class represents the positive diagnoses, such oversight could lead to dire consequences, underlining the importance of accurate and sensitive classification.

To address this challenge, the field of class imbalance learning has emerged, proposing strategies to improve classifier performance despite unrelated class sizes. These strategies fall into three broad categories: data-level methods, algorithm-level methods, and hybrid methods.

1. Data-Level Methods: These methods are employed during data preprocessing and aim to balance the class distribution by resampling. They include undersampling the majority class, oversampling the minority class, or applying a combination of both in hybrid sampling [4]. The goal is to mitigate the skew towards the majority class and provide a more balanced dataset for model training, thereby enhancing the classifier's ability to recognize and accurately classify minority class instances.
2. Algorithm-Level Methods: This approach addresses class imbalance by modifying the learning algorithms themselves, often by assigning greater weight to the minority class samples [4]. This increased emphasis helps to counteract the natural tendency of classifiers to favor the majority class, promoting greater sensitivity to the minority class and improving overall classification accuracy [4].
3. Hybrid Methods: Combining the strengths of data-level and algorithm-level methods, hybrid approaches use ensemble methods to leverage the advantages of both, further bolstering the model's capability to handle imbalanced datasets effectively [4].

In our project, understanding and applying class imbalance learning principles is crucial. It allows us to refine our machine learning models to be not only more equitable in their treatment of class data but also more effective in detecting early signs of ovarian cancer, a necessity given the life-saving potential of early diagnosis.

1.4.2 Introduction to Random Forest

Random Forest classification stands as a robust and versatile machine learning algorithm, particularly effective in addressing the complexities of unbalanced datasets, a common challenge in fields such as medical diagnosis. At its core, Random Forest is an ensemble learning method that constructs a multitude of decision trees during the training phase and outputs the mode of the classes (classification) of the individual trees to make a prediction [4]. This ensemble approach not only enhances the predictive accuracy but also mitigates the risk of overfitting associated with individual decision trees [4].

When dealing with unbalanced datasets, where the instances of one class significantly outnumber those of another, standard classification algorithms can become biased towards the majority class, often at the expense of the predictive performance of the minority class. In medical contexts, such as ovarian cancer detection, this imbalance can lead to a high rate of false negatives, where the model fails to identify true cases of the condition [4]. Random Forest addresses this challenge through its inherent structure and functionality.

One of the key strengths of Random Forest in managing unbalanced data is its ability to perform both bagging (bootstrap aggregating) and feature randomness when building each tree [4]. Bagging involves creating multiple subsets of the original dataset with replacement, ensuring a diverse set of data for training each tree. This diversity helps in reducing the variance and bias towards the majority class. Additionally, by selecting random subsets of features for splitting nodes, Random Forest ensures that the trees in the forest are uncorrelated, further enhancing the model's ability to generalize across unbalanced data.

1.4.3 Introduction to Support Vector Machine

The SVM represents a powerful and sophisticated approach in the realm of machine learning, particularly when dealing with binary classification problems. Its core principle involves mapping input vectors into a high-dimensional feature space where a linear decision boundary, or hyperplane, is constructed. This hyperplane is designed to maximize the margin between the two classes, thereby enhancing the model's generalization capabilities. The distinctiveness of SVM lies in its ability to construct a decision surface in a high-dimensional space, a feature that lends itself particularly well to complex classification tasks [5]. In the context of class-imbalanced datasets, such as those often encountered in medical diagnostics and specifically in ovarian cancer detection, traditional SVM techniques might face challenges due to the overwhelming influence of the majority class. Recognizing this, Zhang et al. proposed an innovative adaptation of SVM tailored for unbalanced data classification. Their methodology begins with a preprocessing step involving PCA (Principal Component Analysis) whitening and label binarization [5]. PCA whitening transforms the data such that the variance of each feature is normalized to 1, effectively decoupling the features and simplifying the model's training process. This transformation is mathematically represented as,

Equation 1: PCA Whitening

$$X_{pcawhite,i} = \frac{x_{rot,i}}{\sqrt{\lambda_i + \varepsilon}}$$

where $x_{rot,i}$ is the rotated data, λ_i is the eigenvalue associated with the i^{th} feature, and ϵ is a small constant added for numerical stability [5].

Following the PCA whitening, the dataset undergoes a partitioning process using leave-one-out cross-validation. To address the imbalance, the majority class instances in the training set are randomly subsampled to achieve a more balanced ratio between the majority and minority classes. This rebalanced dataset is then utilized to train an ensemble of SVM models, where each model is weighted according to its performance in classifying the minority class correctly [5].

This ensemble approach not only mitigates the bias towards the majority class but also enhances the overall predictive performance of the minority class, which is critical in applications like ovarian cancer detection where the cost of misclassification is high [5]. By integrating SVM with ensemble techniques and strategic data preprocessing, this method offers a compelling solution to the challenge of class imbalance, providing a pathway to more accurate and equitable machine learning models in healthcare diagnostics.

1.4.4 Introduction to Neural Networks

Neural networks have guided a revolutionary era for medical diagnosis, fundamentally changing how complex data is analyzed and interpreted. These advanced computational models mimic the structure and function of the human brain's neural networks, enabling them to identify intricate patterns and relationships within vast datasets. Their ability to handle high-dimensional data, coupled with their proficiency in modeling nonlinear relationships, makes neural networks particularly unique in navigating the complexities of medical data [6].

The architecture of neural networks, comprising interconnected layers of nodes or neurons, is designed to perform simple processing on input data, with the capability to learn and adapt by adjusting the weights of connections based on the data it processes [6]. This feature is especially beneficial in the medical field, where the nature of data and disease patterns can be highly variable and complex.

Moreover, neural networks' adaptability is a key asset, allowing them to improve and refine their predictive performance continually. This learning capability is critical in the ever-evolving landscape of medical science, where new findings and data are constantly reshaping our understanding of various diseases. Beyond mere diagnosis, neural networks extend their utility to predictive analytics, forecasting disease progression, and patient outcomes, thus contributing to more proactive and preventive healthcare approaches [6].

However, the integration of neural networks in medical diagnosis is not without its challenges. The complexity and "black box" nature of these models often make their decision-making processes opaque, leading to concerns about their interpretability and reliability in clinical settings [7, 6]. Additionally, the substantial data and computational power required to train and run sophisticated neural networks pose significant barriers, especially in resource-limited environments.

2 Design

The design segment of our report encapsulates the detailed planning and development strategies of the "Early Warning of Ovarian Cancer" system. This crucial stage is divided into two main components: the development of a machine-learning model, and the design of the front-end application.

2.1 Machine Learning Model

The machine learning model at the heart of the "Early Warning of Ovarian Cancer" system is made with precision, drawing from a rich array of data sources and clinical expertise. It leverages cutting-edge algorithms to analyze key indicators of ovarian cancer risk with high accuracy. This model is the result of extensive data training, expert input, and iterative testing to ensure its effectiveness and reliability as a predictive tool in a clinical environment.

2.1.1 Data Analysis

The data analysis for our "Early Warning of Ovarian Cancer" system began with a thorough examination of a dataset provided by the PLCO study. With approximately 78,000 test subjects and 243 columns of comprehensive health-related questions, the dataset offered a foundation for our machine-learning model. It included around 800 positive cases of ovarian cancer, a crucial subset for our predictive analysis. Our exploratory data analysis focused on understanding the dataset's distribution, identifying patterns, and confirming assumptions. It was critical to understand the context and nuances of each variable, ensuring that the most relevant predictors were used in the model. This dataset was not only vast but also deeply detailed, requiring careful curation to distill the most significant features that could influence the risk of ovarian cancer.

2.1.2 Data Preprocessing

Setting the stage for accurate predictions, our data preprocessing approach was meticulously designed to maintain the quality and relevance of our dataset. The initial step was the feature selection process. In consultation with Dayna Freedman [8], a gynecologist from Scarborough Professional Centre, we identified the most significant features that are indicative of ovarian cancer. This approach leveraged expert clinical knowledge to guide our data-driven strategies, focusing on variables such as smoking status, family history of cancer, reproductive history, and hormone therapy.

Once the relevant features were selected, the next phase was normalization. We applied the z-score normalization method to variables including age, years of smoking, weight, and birth control usage. Normalization is an important process that ensures each feature contributes equally to the analysis by adjusting for scale and removing outliers, preventing any single variable from disproportionately influencing the model due to variance in measurement units or range.

In the data preprocessing phase, our code is designed to avoid the pitfalls of normalization when features have no variance, as this could lead to division by zero errors. This is crucial to ensure that normalization is only applied to features with variation. We also deal with missing values by assigning them a unique value (-1), which allows the model to process these entries without losing important information. For categorical data, such as smoking status, we've applied a specific type of encoding. This helps to clearly distinguish between non-smokers and cases where the data is missing, thus enhancing the interpretative accuracy of our features.

To evaluate our model, we split our dataset into distinct subsets for training, validation, and testing. Initially, we allocated 90% of the data to the training set. From this portion, we then reserved 25% for validation purposes. The validation set plays a critical role in fine-tuning the model's parameters, and we made sure to adjust these settings without utilizing the test set. This approach preserves the test set's integrity as a reliable indicator of the model’s real-world performance. The remaining 10% of the data was set aside as a test set, completely unseen by the model during its training or validation phases. It is this test set that allows us to assess how effectively the model can perform in practical scenarios, offering insights into its capability to detect ovarian cancer with real-world data.

2.1.3 Feature Importance

Understanding the most crucial factors in detecting ovarian cancer at an early stage is essential for the effectiveness of our tool. In our conversation with Dr. Freedman [8], a series of key predictors associated with ovarian cancer were identified, such as the prolonged use of birth control, tubal ligation (having one's tubes tied), childbearing history, and breastfeeding practices. Additionally, Mrs. Freedman highlighted that those who begin ovulating at an earlier age tend to have a higher risk of developing ovarian cancer. These insights have been carefully integrated into our model's design. Taking heed of Dr. Freedman's expertise, we've given special consideration to the age at which menstruation begins for individuals, recognizing its potential as a critical marker of risk. This level of detail in our model helps to sharpen its predictive accuracy and enhance its utility as a preventive measure.

Figure 1 displays when people typically started their first menstrual period, comparing those with and without ovarian cancer. The group without cancer started at more similar ages, shown by a tighter box in the plot. The ages vary more for those with cancer, shown by a wider box and more points outside it. Particularly, there are a lot of cases where patients have their first menstrual period at a very young age. This matches what Mrs. Freedman said about earlier menstruation being a risk factor for ovarian cancer.

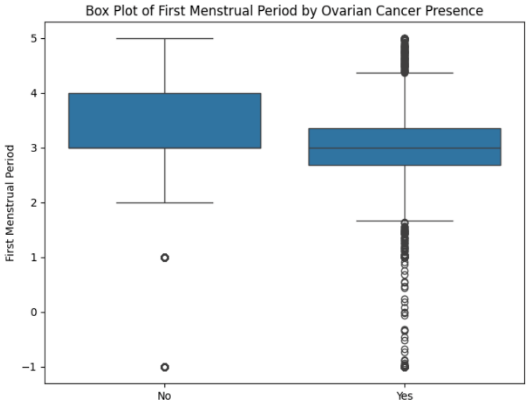


Figure 1: Box Plot Comparison for First Menstrual Period

In addition to the box plot, the heatmap, in Figure 2 provides a comprehensive view of the correlations between all considered features. Each square in the heatmap shows the strength of the relationship between two variables, with warmer colors indicating a stronger positive correlation and cooler colors representing a negative correlation. The heatmap is particularly useful in identifying features that are strongly related to ovarian cancer, as well as in spotting potential multicollinearity between variables, which could impact the model's performance.

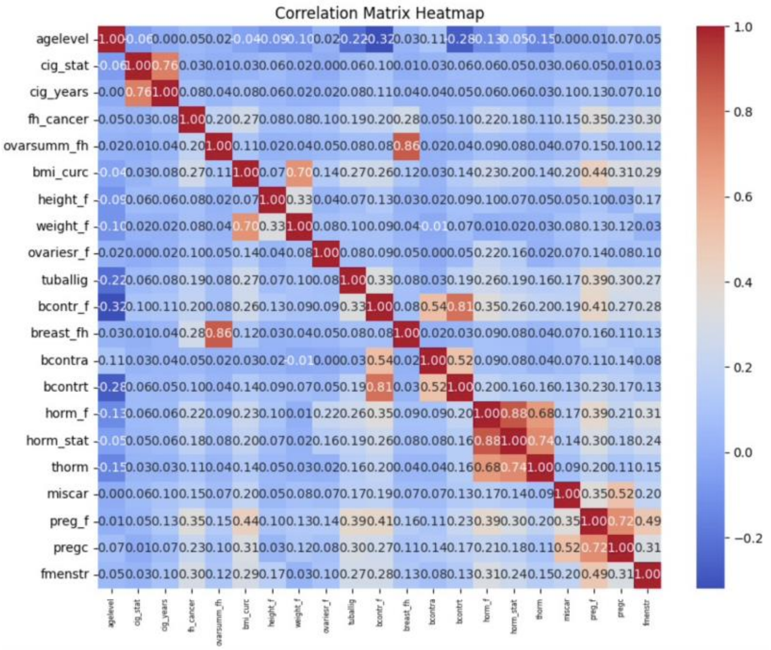


Figure 2: Correlation Matrix Heat Map for Random Forest

Features that show a stronger red shade about 'ovar_cancer' are the most important. These may include variables such as family history of cancer, body mass index, and reproductive history. Features with a high degree of correlation (close to 1 or -1) may exert a significant influence on the model's predictions and could be prime targets for intervention in a clinical setting. Conversely, features with little to no coloration (close to 0) have a minimal correlation and may be less critical in the prediction process. The bar chart in Figure 3 highlights the significance of each feature in predicting ovarian cancer according to our model. The length of the bars indicates how important each feature is. This chart complements the insights from the heatmap, reinforcing the value of certain features in our analysis. Together, these tools provide a clear visual of which factors our model finds most telling for ovarian cancer risk.

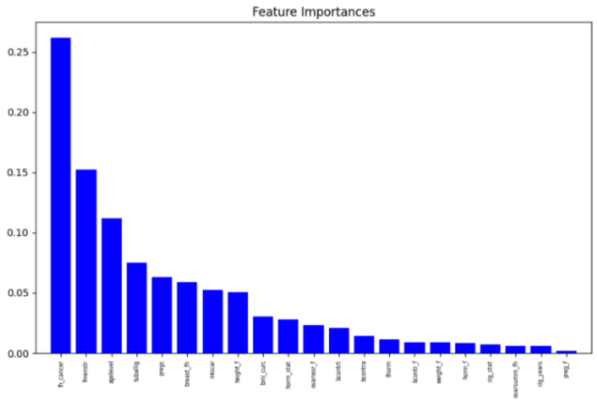


Figure 3: Feature Importance Graph

2.1.4 Model Development

We faced a significant challenge in the initial stages of our machine-learning model selection. Our dataset, which contains nearly 80,000 entries, presented a class imbalance explained in 1.4.1.

2.1.5 Synthetic Minority Over-Sampling Technique

To mitigate this issue, we employed the Synthetic Minority Over-Sampling Technique (SMOTE). Proposed by Chawla et al., SMOTE is an innovative approach that addresses the class imbalance problem by generating synthetic examples of the minority class [9]. This is achieved by algorithmically creating new samples that are interpolations of the minority class instances that lie together in the feature space [9]. It operates by randomly picking a point from the minority class and computing the k-nearest neighbors for this point. The synthetic points are then added between the chosen point and its neighbors [5]. This method not only augments the minority class but also helps to generalize the decision boundaries, avoiding overfitting to the minority class. Figure 4 illustrates the SMOTE process, showing how synthetic data points are added to balance the dataset before training.

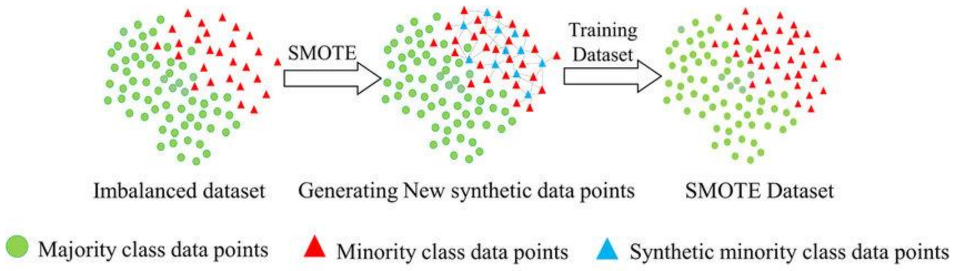

SMOTE
Training Dataset
Imbalanced dataset
Generating New synthetic data points
SMOTE Dataset
Majority class data points
Minority class data points
Synthetic minority class data points

Figure 4: SMOTE Graphical Representation

2.1.5.1 Baseline Classifiers using SMOTE:

After preprocessing our dataset with SMOTE to address the class imbalance issue, we proceeded to evaluate the performance of three baseline classifiers—each with distinct characteristics and learning approaches—to establish the most effective model for our application. The chosen classifiers were:

Random Forest Model: This model uses 190 decision trees (as determined by the best number of estimators during validation) with a maximum depth of 10 nodes. This depth helps to capture complex patterns in the data without overfitting, which is a common concern with medical datasets. The ‘min_samples_split’ parameter is set to 4 to ensure that nodes only split when at least four samples are present, further preventing overfitting. This model's ability to generalize is especially useful in medical diagnostics where the cost of misdiagnosis is high. Figure 5 below is a segment from our Random Forest Model’s decision trees.

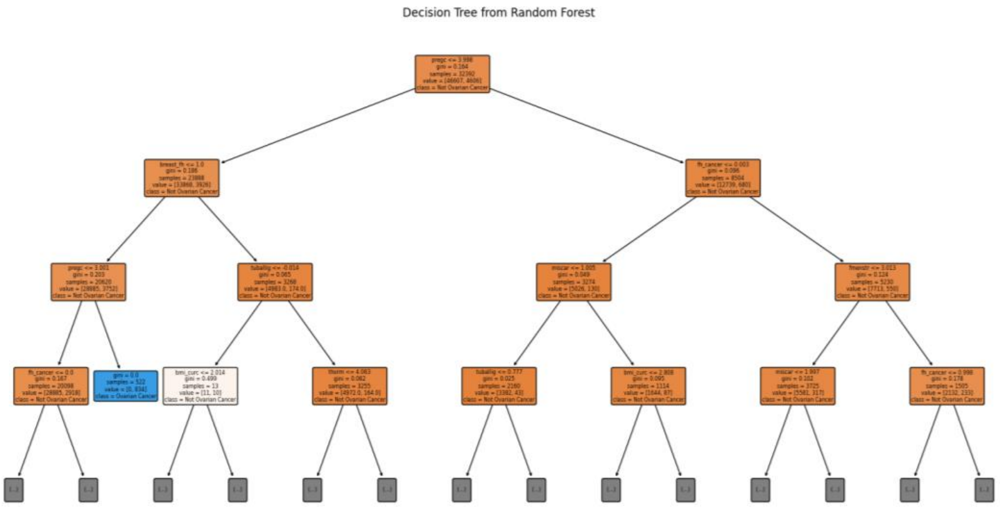


Figure 5: Random Forest Graphical Representation

Neural Network: The network consists of an input layer matched to the number of features in the dataset, followed by two hidden layers with 128 and 64 nodes, respectively. Each hidden layer uses the ReLU activation function, known for its efficiency in deep learning. The network also includes dropout layers set at a rate of 0.3 to reduce overfitting by randomly setting a fraction of input units to 0 during training.

This design, with a specific number of nodes and layers, allows the network to detect subtle and complex patterns in the data, which is critical for identifying early indicators of diseases like ovarian cancer.

Support Vector Machine: The SVM model is configured with a radial basis function (RBF) kernel, a standard choice for non-linear data, which is often the case in complex medical datasets. The regularization parameter C is set to 1, providing a balance between correctly classifying training data and maintaining a smooth decision boundary. A high C value could lead to a highly complex model that risks overfitting by trying to classify all training samples correctly. In contrast, a low C might be too forgiving and not capture the nuances necessary for accurate medical diagnosis. The choice of C=1 and the RBF kernel makes this SVM model a robust classifier capable of handling the intricate structures within medical data.

While crafting our models, we carefully selected hyperparameters to achieve a delicate balance: the models needed to learn from the training data with precision while also being able to apply that learning to new, unseen data—a vital requirement for tools used in medical diagnostics.

Upon reviewing performance metrics, the Random Forest model emerged as the top performer. It had the highest count of true positives, at 14,958, showing its strength in correctly pinpointing cases that are not cancerous. It did produce a moderate number of false positives (547), but in the medical context, these are less concerning as they would lead to further testing rather than direct treatment, minimizing the impact of such an error.

Crucially, the Random Forest model registered the least false negatives—only 137—showcasing its sensitivity in detecting instances where cancer is present. This aspect is vital; undetected cases can lead to life-threatening situations, making the minimization of false negatives a top priority.

Moreover, the model also reported a substantial number of true negatives (1,519), underlining its effectiveness in correctly identifying actual cases of ovarian cancer. This capability is of utmost importance for the early detection and subsequent treatment of the disease. Below, Table 1 provides a side-by-side comparison of these outcomes across the different models, illustrating the Random Forest's comparative advantage in our application.

Table 1: SMOTE Model Comparison

Model	True Positive	False Positive	False Negative	True Negative
Random Forest	14958	300	50	1549
Support Vector Machine	13998	1700	200	19
Neural Network	14840	950	150	1001

Figure 6 offers a side-by-side evaluation of essential performance metrics for the three models we tested. It clearly shows that the Random Forest model outshines the others, leading the way with its accuracy, precision, recall, and F1 score. This demonstrates the model's strong ability to identify both the presence and absence of ovarian cancer accurately.

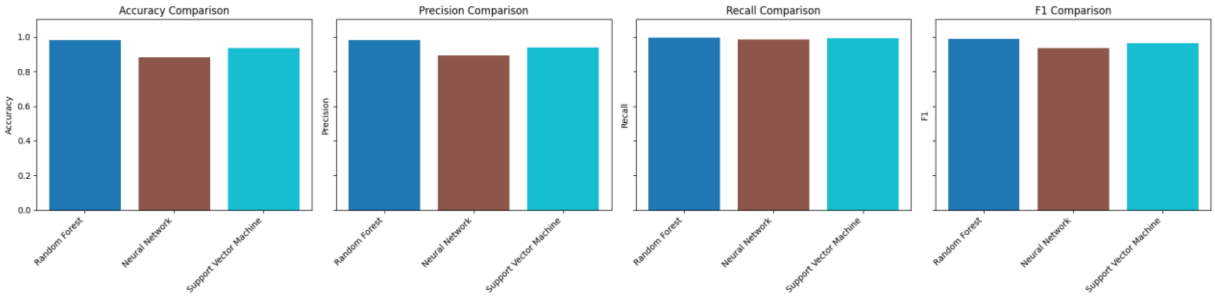


Figure 6: SMOTE Classifier Comparison Graphical

To enhance the performance of our Random Forest model, we employed a grid search algorithm. This approach explores a wide range of hyperparameter combinations, evaluating each to find the set that produces the best results for our specific dataset. By automating the search for the optimal parameters, grid search ensures that our model isn't just good, but as accurate and reliable as possible for detecting ovarian cancer.

2.1.6 One-Class Classification

In our initial approach, we used SMOTE to generate synthetic samples for the minority class in the dataset, aiming to create a balanced environment for our models. However, upon further analysis, we recognized that SMOTE operates within the limitations of the convex hull of the minority class. The convex hull is a concept from computational geometry that refers to the smallest convex set that encompasses all the points in a dataset. For example, imagine stretching a rubber band around the outermost points of the minority class; the area enclosed by the band represents the convex hull. Within this space, SMOTE interpolates new data points, which means that the diversity of the synthetic samples is inherently confined to the existing range of minority class features.

This confinement can unintentionally lead to a model that is still prone to overfitting because it hasn't truly learned to generalize beyond the observed minority class instances. It may perform well on the SMOTE-augmented dataset but struggles with new, real-world data that could present characteristics slightly outside of the convex hull defined by the minority class samples.

To address this concern, we shifted our focus to one-class classification (OCC). OCC, also known as anomaly detection, is a technique where the model is trained exclusively on the majority class—negative samples, in our case [2]. The model learns what "normal" looks like, based on the data from the majority class. Once trained, the model then attempts to identify all other points that do not fit this learned "normality" as anomalies, which, for our project, would be the positive cases of ovarian cancer [2]. The advantage of this approach is that it doesn't require a balanced dataset since it's only concerned with the characteristics of one class, and it can potentially detect anomalies that lie outside the minority class's convex hull. Figure 7 shows how OCC focuses on the majority class to identify outliers, which in our project represent the possible cases of Ovarian Cancer.

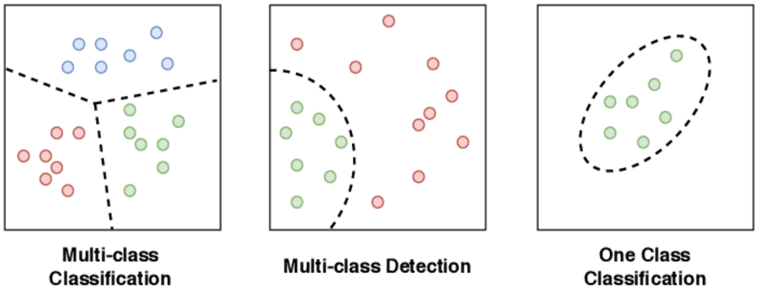


Figure 7: Classification Techniques

One-class classification works well for problems where the negative class (non-cases of ovarian cancer) is well-defined and substantially larger than the positive class (cases of ovarian cancer) [2]. By focusing on the larger class's structure, OCC creates a model that can more accurately identify data points that deviate from the norm.

2.1.6.1 Baseline Classifiers using One-Class Classification:

We selected three models for our analysis, each based on the assumption that the majority class in our dataset—representing the 'normal' condition—is well-defined and that any instances of ovarian cancer are outliers. The architecture and tuning of each model were specifically chosen to detect these outliers with as few false positives as possible. Below are the detailed descriptions of each OCC model:

Autoencoder Model: The autoencoder for anomaly detection is built with a symmetrical architecture: it has an input layer corresponding to the number of input features, followed by an encoding path down to a bottleneck of 16 neurons. This narrow layer captures the essence of the input data. The subsequent decoding path mirrors the encoder, reconstructing the input data. This model includes dropout layers at a rate of 0.1 to prevent overfitting and is trained with mean squared error loss, highlighting reconstruction errors that signal anomalies.

Isolation Forest: This model isolates anomalies instead of profiling normal data points. It's constructed with 100 base estimators, and its contamination parameter is set to 0.01, expecting that approximately 1% of the dataset contains anomalies. This parameter is crucial as it directly affects the model's threshold for anomaly detection. It's trained solely on the majority class, assuming anomalies are rare and primarily absent from the training set.

One-Class Support Vector Machine: The One-Class SVM is set with an RBF kernel to handle the non-linearity in the data. The nu parameter, representing the upper bound on the fraction of margin errors and the lower bound of the fraction of support vectors, is set to 0.01. This implies we expect no more than 1% of data to be outliers. A gamma value of 'auto' allows the model to automatically determine the best value for the kernel coefficient, catering to the complexity of the dataset.

Upon evaluating our one-class classification approach with three different models, the SVM stands out with a notably high precision, as it has the lowest number of false positives. However, the confusion matrix reveals a significant drawback: the SVM failed to identify any true negatives, indicating it could not correctly detect any instances of ovarian cancer, as can be seen in Table 2 below.

Table 2: One-Class Model Comparisons

Model	True Positive	False Positive	False Negative	True Negative
Isolation Forest	14738	781	121	2
Support Vector Machine	15296	224	0	0
Autoencoder	62074	0	493	0

Although the Isolation Forest model demonstrates a balanced performance across precision and recall, the extremely low number of true negatives across all models is concerning. This suggests that while the models are proficient at identifying non-cancerous cases, they struggle considerably in detecting the actual presence of cancer, which is critical for a successful early detection system. Thus, despite the precision seen in the SVM, the overall results from the confusion matrices are not satisfactory for a reliable diagnostic tool. Further investigation and refinement of the model are essential to improve its capability to identify true cancer cases effectively. Figure 8 illustrates the comparison between the Isolation Forest, Neural Network, and Support Vector Machine models, showcasing how each performs across the metrics of accuracy, precision, recall, and F1 score.

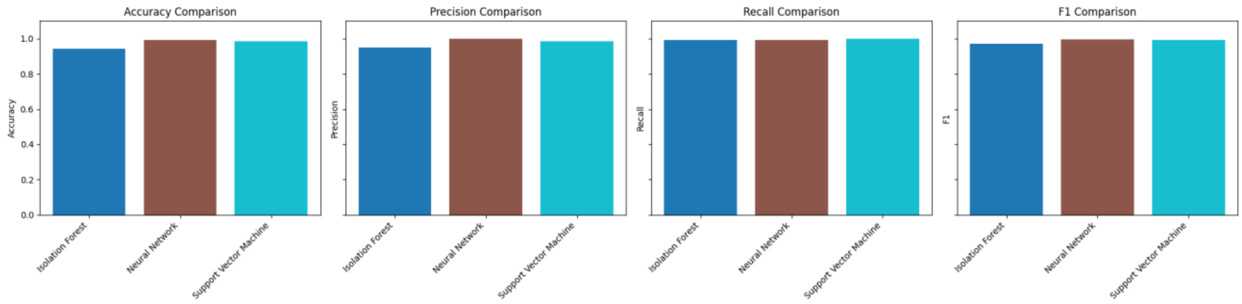


Figure 8: One-Class Classification Graphical Representation

2.1.7 Model Conclusions

The SMOTE-enhanced Random Forest model has proven to be the most effective approach for the early detection of ovarian cancer in our study. This model showed a strong capability to balance sensitivity and specificity, outperforming other models in precision and recall—a crucial factor when dealing with life-threatening conditions where early detection is paramount. We encountered challenges with the one-class classification approach. Initially, it was considered to counteract overfitting—a common concern with imbalanced datasets. However, this method did not perform as expected in recognizing the minority class, which, in this context, represents the actual cancer cases. The model's inability to adequately identify true positives underlines the necessity for a balanced training approach, as provided by SMOTE, to ensure that both classes are appropriately represented and learned by the model.

Our exploration highlights the complexities of medical diagnostic model development and the importance of choosing the right data preprocessing techniques. While one-class classification may offer benefits in certain scenarios, our project reaffirms that for highly imbalanced data, like that in ovarian cancer detection, methods like SMOTE combined with a Random Forest classifier can provide a more robust solution by enhancing the model's ability to learn from the minority class without compromising the overall accuracy.

2.2 Front-End Application

The user-facing component of the "Early Warning of Ovarian Cancer" project is shown in a modernized and accessible web application. Developed with Streamlit, GitHub Codespaces, Python, and CSS, the front end is designed to prioritize user experience while ensuring functionality and responsiveness.

2.2.1 Design and Evolution of Technology Stack

In the iterative journey of our project, the front-end application experienced a transformative shift from the initial proposal's concept. The initial proposal envisioned a React-based front-end; however, upon further exploration and prototyping, Streamlit emerged as a more fitting choice. This pivot was driven by

Streamlit's ability to integrate tightly with our Python-based data processing and machine learning models. While React offers broad capabilities for web development, Streamlit's simplicity and direct support for Python scripts allowed for a more efficient development process and a smoother integration of the machine learning backend.

Streamlit's interactive widgets and state management significantly reduced the development time, enabling us to focus on the application's functionality and user experience without compromising on interactivity. Furthermore, Streamlit made it easier to move our web application from development to production and to share it with others.

2.2.2 User-Centric Interface Design

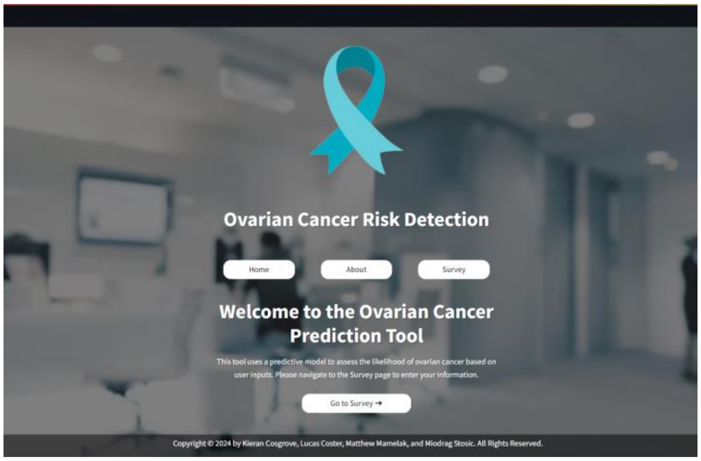
Our design philosophy maintained a loyal commitment to user-centric principles. The interface design is marked by a minimalist aesthetic, ensuring users are greeted with an uncluttered and simple-to-read layout. Our application provides a welcoming introduction, clear navigation, and an intuitive survey interface that invites user engagement.

2.2.2.1 Home Page

The home page of the Ovarian Cancer Risk Detection application sets the tone for the user experience. It had a simple, contemporary look with soft colours that created a feeling a sense of calm and trustworthiness. At the forefront is the application's title, "Ovarian Cancer Risk Detection," prominently displayed alongside the universally recognized teal ribbon that symbolizes ovarian cancer awareness.

The navigation is straightforward, with clearly labeled tabs for 'Home,' 'About,' and 'Survey,' facilitating easy transitions between sections of the application seen in Figure 9. The message, "Welcome to the Ovarian Cancer Prediction Tool," is a friendly and clear statement that describes the tool’s purpose. Underneath, the prompt ‘Go to Survey’ encourages users to start using the tool, indicating a straightforward next step.

The homepage's background has a picture that blurs gently, keeping attention on the information provided. This design choice minimizes distractions and highlights the app's practicality and the user's engagement with it.



Ovarian Cancer Risk Detection

Welcome to the Ovarian Cancer Prediction Tool

This tool uses a predictive model to assess the likelihood of ovarian cancer based on user inputs. Please navigate to the Survey page to enter your information.

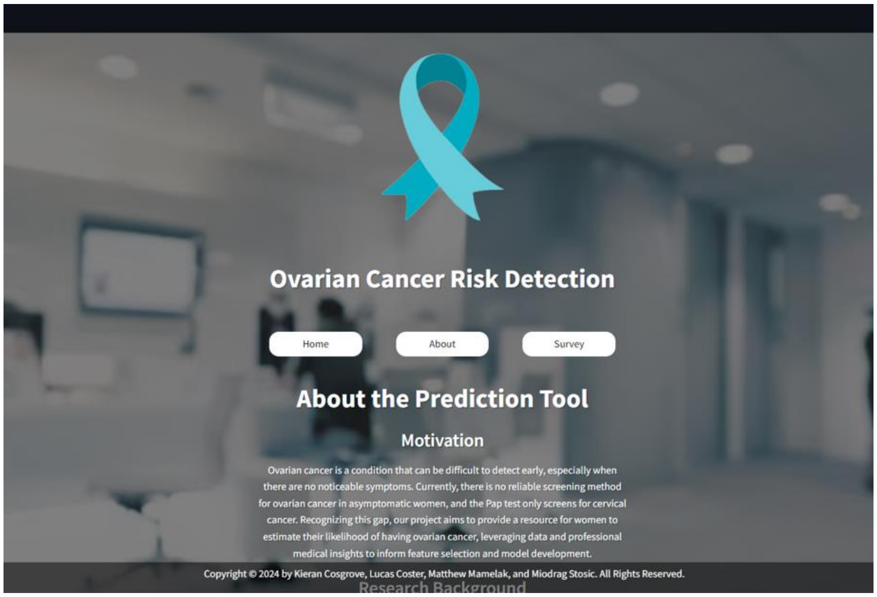
Copyright © 2024 by Kieran Cosgrove, Lucas Coster, Matthew Mamelak, and Miodrag Stosic. All Rights Reserved.

Figure 9: Main Application Home Page

2.2.2.2 About Page

The 'About' page is an essential part of the app, providing insight into the purpose and research behind the Ovarian Cancer Risk Detection tool. The page starts by explaining the reason for developing the app—to improve early detection of ovarian cancer, which is hard due to subtle symptoms and limited screening options. It then explains the app's scientific basis, including collaboration with medical experts and the use of the PLCO dataset, ensuring the tool's credibility and data-focused methodology, seen in Figure 10 below.

Key performance metrics, specifically the model's accuracy and recall rates, are presented, establishing trust, and reinforcing the application's reliability. Instructions on how to navigate and use the website are provided, ensuring users have a clear guide to follow. The ‘About’ page concludes with a disclaimer, noting that while the tool is a helpful resource, it is not a substitute for professional medical advice, thereby positioning the application as a complement to, rather than a replacement for, traditional medical consultation.



Ovarian Cancer Risk Detection

About the Prediction Tool

Motivation

Ovarian cancer is a condition that can be difficult to detect early, especially when there are no noticeable symptoms. Currently, there is no reliable screening method for ovarian cancer in asymptomatic women, and the Pap test only screens for cervical cancer. Recognizing this gap, our project aims to provide a resource for women to estimate their likelihood of having ovarian cancer, leveraging data and professional medical insights to inform feature selection and model development.

Copyright © 2024 by Kieran Cosgrove, Lucas Coster, Matthew Mamelak, and Miodrag Stosic. All Rights Reserved.

Research Background

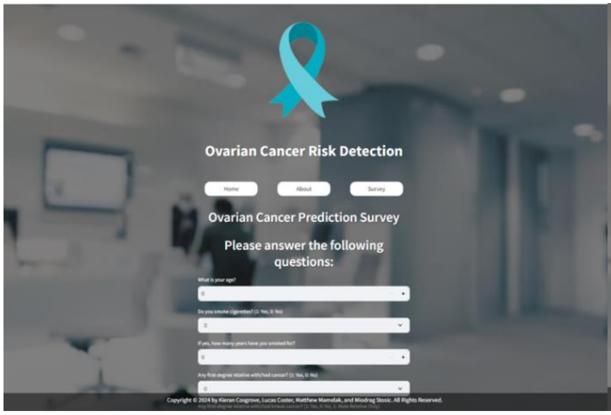
Figure 10: About Application Page

2.2.2.3 Survey Page

The 'Survey' page is the heart of the user interface, where data collection occurs. The layout is intentionally uncomplicated, with the user prompted to provide information via a series of straightforward questions. This design decision mitigates the complexity and potential anxiety associated with medical data provision. Following the completion of the survey, the user is invited to submit their responses through the 'Predict' button—a feature that stands out clearly on the page, guiding the user towards obtaining their risk assessment seen in Figure 11 and

Figure 12.

Once the prediction is made, the result is displayed with both clarity and sensitivity—a visual representation of risk is provided alongside a percentage, ensuring comprehensibility. Importantly, the presentation of results is carefully crafted to be informative while avoiding any alarmism. The application consistently emphasizes the importance of professional medical interpretation of the results, reflecting a commitment to ethical user engagement.



Ovarian Cancer Risk Detection

Ovarian Cancer Prediction Survey

Please answer the following questions:

What is your age?
0

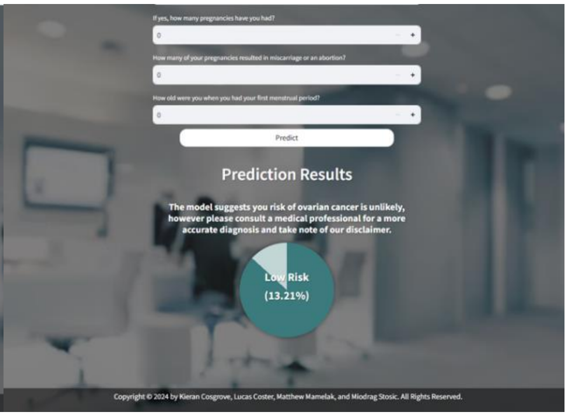
Do you smoke cigarettes? (1: Yes, 0: No)
0

If yes, how many years have you smoked for?
0

Any first-degree relative with/had cancer? (1: Yes, 0: No)
0

Copyright © 2024 by Kieran Cosgrove, Lucas Coster, Matthew Mamelak, and Miodrag Stosic. All Rights Reserved.

Any first-degree relative with/had breast cancer? (1: Yes, 0: No, 2: Male Relative Only)



If yes, how many pregnancies have you had?
0

How many of your pregnancies resulted in miscarriage or an abortion?
0

How old were you when you had your first menstrual period?
0

Prediction Results

The model suggests you risk of ovarian cancer is unlikely, however please consult a medical professional for a more accurate diagnosis and take note of our disclaimer.

Copyright © 2024 by Kieran Cosgrove, Lucas Coster, Matthew Mamelak, and Miodrag Stosic. All Rights Reserved.

Figure 11: Survey Page

Figure 12: Prediction Wheel Example.

2.2.3 Incorporating User Feedback and Accessibility

Throughout its development, we actively incorporated user feedback, which led to several iterative improvements to the interface design. These enhancements focused on creating an accessible and inclusive experience, ensuring that users of all backgrounds and technical proficiencies could use the tool with confidence.

Accessibility considerations were factored into design decisions, ensuring that the web application was responsive and adaptable across various devices and screen sizes. The application's color scheme, typography, and interactive elements were selected to accommodate a diverse user base, including those with visual or motor impairments.

2.2.4 Performance Optimization

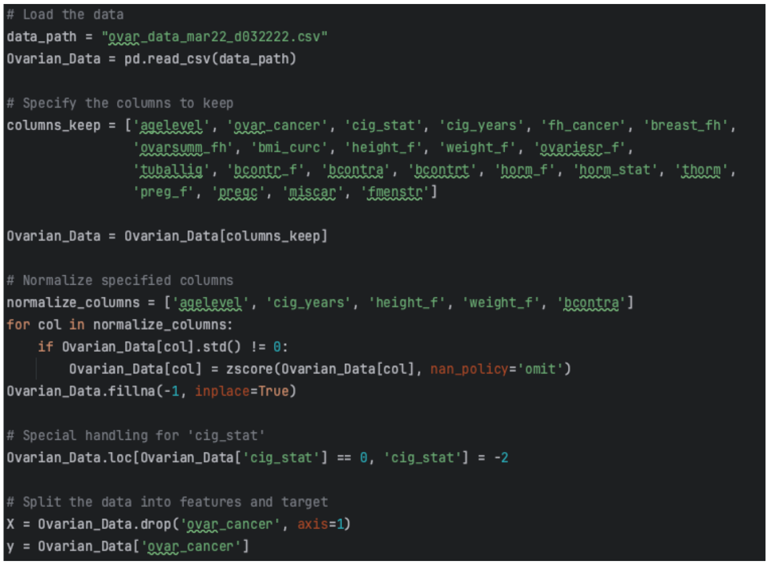
Performance optimization remained a high priority, with the application's loading times and responsiveness being carefully monitored and refined. Streamlit's native handling of caching and efficient data processing ensured that users experienced minimal latency, a critical factor given the data-intensive nature of the tool.

3 Implementation

The implementation of the "Early Warning of Ovarian Cancer" system was a structured and strategic process. It comprised the development and integration of the machine learning model with the front-end application, forming a properly functional tool.

3.1 Machine Learning Model

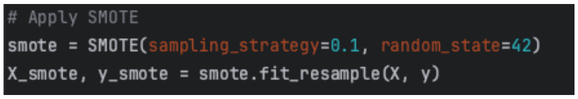
We initiated our machine learning model's implementation by loading the dataset, ovar_data_mar22_d032222.csv, into a Pandas Data Frame. We then employed statistical normalization to ensure data uniformity. This normalization accounted for variations in scale among different features and aided in mitigating the influence of outliers. The code block in Figure 13 details the initial steps in our machine learning model's implementation, focusing on data loading, feature selection, and normalization.



```
# Load the data
data_path = "ovar_data_mar22_d032222.csv"
Ovarian_Data = pd.read_csv(data_path)

# Specify the columns to keep
columns_keep = ['agelevel', 'ovar_cancer', 'cig_stat', 'cig_years', 'fh_cancer', 'breast_fh',
                'ovarsumm_fh', 'bmi_curc', 'height_f', 'weight_f', 'ovariesr_f',
                'tuballig', 'bcontr_f', 'bcontra', 'bcontrt', 'horm_f', 'horm_stat', 'thorm',
                'preg_f', 'pregc', 'miscar', 'fmenstr']

Ovarian_Data = Ovarian_Data[columns_keep]

# Normalize specified columns
normalize_columns = ['agelevel', 'cig_years', 'height_f', 'weight_f', 'bcontra']
for col in normalize_columns:
    if Ovarian_Data[col].std() != 0:
        Ovarian_Data[col] = zscore(Ovarian_Data[col], nan_policy='omit')
Ovarian_Data.fillna(-1, inplace=True)

# Special handling for 'cig_stat'
Ovarian_Data.loc[Ovarian_Data['cig_stat'] == 0, 'cig_stat'] = -2

# Split the data into features and target
X = Ovarian_Data.drop('ovar_cancer', axis=1)
y = Ovarian_Data['ovar_cancer']
```

Figure 13: Pre-Processing Code Block

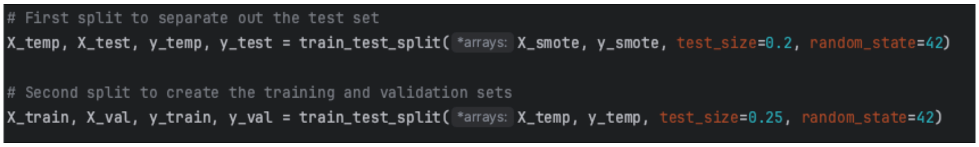
Next, we use SMOTE to address the imbalanced nature of our dataset. The code block, shown in Figure 14, shows the implementation of SMOTE using the imblearn.over_sampling library, which provides a practical method to balance our dataset. By calling SMOTE (sampling_strategy=0.1, random_state=42), we instruct the algorithm to augment the minority class until it constitutes 10% of the majority class, thus addressing the dataset's imbalance. The fit_resample function is then applied to the feature and target sets, effectively creating synthetic samples that are statistically clear with the original minority class.



```python
# Apply SMOTE
smote = SMOTE(sampling_strategy=0.1, random_state=42)
X_smote, y_smote = smote.fit_resample(X, y)
```

Figure 14: SMOTE Implementation

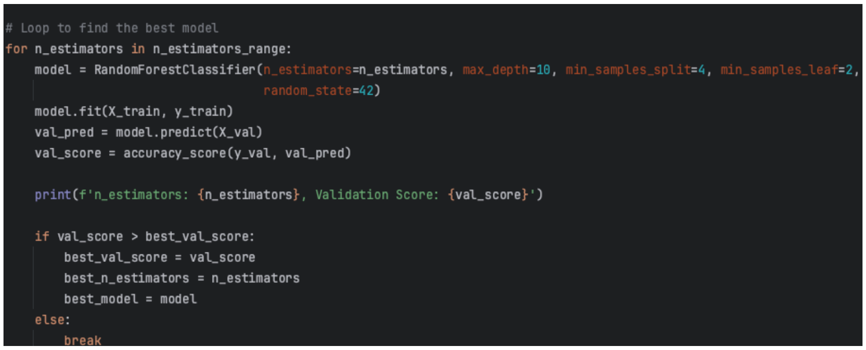
The dataset was then split, using the train_test_split function from sklearn.model_selection. In the code snippet below, we illustrate how our data was systematically partitioned into training, validation, and testing sets.



```python
# First split to separate out the test set
X_temp, X_test, y_temp, y_test = train_test_split( *arrays: X_smote, y_smote, test_size=0.2, random_state=42)

# Second split to create the training and validation sets
X_train, X_val, y_train, y_val = train_test_split( *arrays: X_temp, y_temp, test_size=0.25, random_state=42)
```

Figure 15: Data Splitting

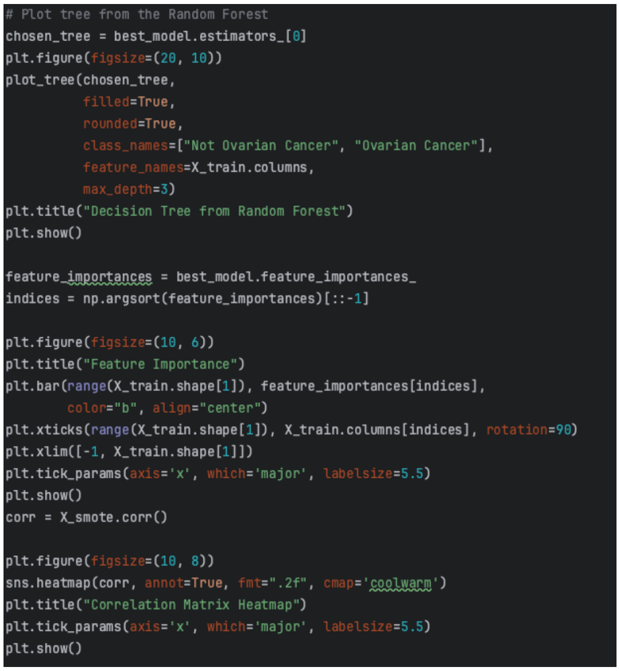
For our model implementation, we calibrated the RandomForestClassifier by tuning its hyperparameters directly within a for-loop. This loop incrementally adjusted the n_estimators parameter—essentially the count of decision trees in the forest—between 10 and 200 in increments of 10. For each iteration, we trained the model on the training subset and then gauged its accuracy against the validation subset. By tracking the highest accuracy score, we pinpointed the most optimal n_estimator value for our final model configuration.



```python
# Loop to find the best model
for n_estimators in n_estimators_range:
    model = RandomForestClassifier(n_estimators=n_estimators, max_depth=10, min_samples_split=4, min_samples_leaf=2,
                                   random_state=42)
    model.fit(X_train, y_train)
    val_pred = model.predict(X_val)
    val_score = accuracy_score(y_val, val_pred)

    print(f'n_estimators: {n_estimators}, Validation Score: {val_score}')

    if val_score > best_val_score:
        best_val_score = val_score
        best_n_estimators = n_estimators
        best_model = model
    else:
        break
```

Figure 16: Model Creation and Testing Loop

After setting our hyperparameters, we evaluated our model against the test set. In this phase, we implemented a specific decision threshold, chosen to be 0.1, to fine-tune our model's sensitivity toward identifying ovarian cancer instances. Then, to interpret the model's decision-making process, we visualized a single decision tree from the Random Forest using the plot_tree function seen in Figure 17, and we plotted feature importance to understand which variables most significantly influenced the model's predictions.



```
# Plot tree from the Random Forest
chosen_tree = best_model.estimators_[0]
plt.figure(figsize=(20, 10))
plot_tree(chosen_tree,
          filled=True,
          rounded=True,
          class_names=["Not Ovarian Cancer", "Ovarian Cancer"],
          feature_names=X_train.columns,
          max_depth=3)
plt.title("Decision Tree from Random Forest")
plt.show()

feature_importances = best_model.feature_importances_
indices = np.argsort(feature_importances)[::-1]

plt.figure(figsize=(10, 6))
plt.title("Feature Importance")
plt.bar(range(X_train.shape[1]), feature_importances[indices],
        color="b", align="center")
plt.xticks(range(X_train.shape[1]), X_train.columns[indices], rotation=90)
plt.xlim([-1, X_train.shape[1]])
plt.tick_params(axis='x', which='major', labelsize=5.5)
plt.show()
corr = X_smote.corr()

plt.figure(figsize=(10, 8))
sns.heatmap(corr, annot=True, fmt=".2f", cmap='coolwarm')
plt.title("Correlation Matrix Heatmap")
plt.tick_params(axis='x', which='major', labelsize=5.5)
plt.show()
```

Figure 17: Plotting Model Results

Lastly, we saved our fine-tuned model with joblib, making it easy to load and use for predictions in our application seen in Figure 17.

3.2 Front-End Application

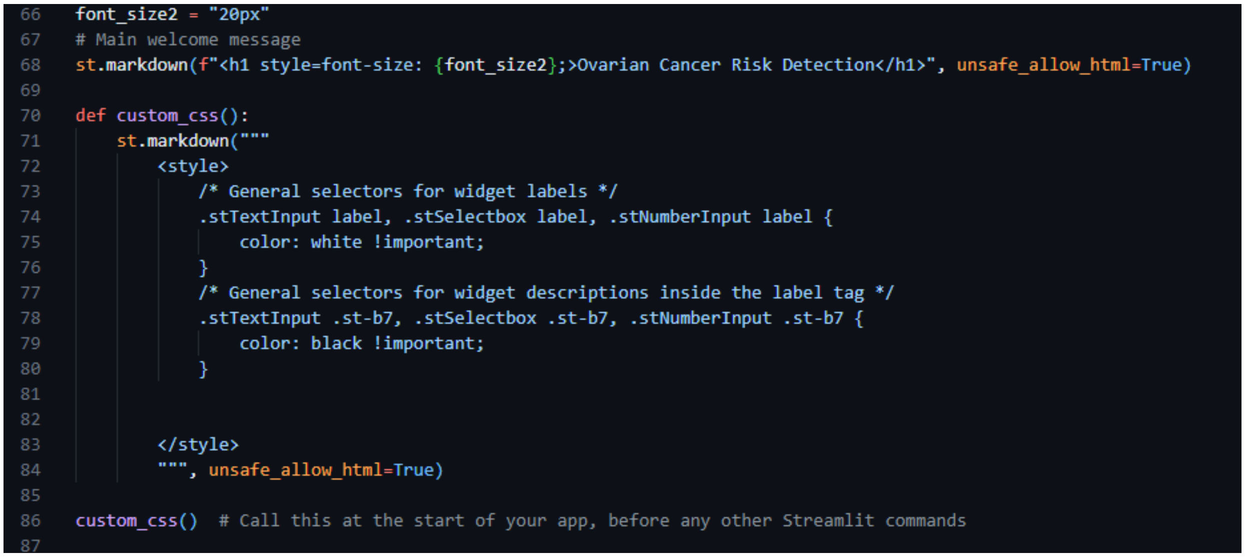
In the practical realization of our Ovarian Cancer Risk Detection tool, the front-end application was constructed using an integration of several software technologies, each contributing to the robustness and user-friendliness of the interface.

3.2.1 Implementation Overview

The application's front end is designed in Python, leveraging the Streamlit framework. In parallel, we harnessed the power of various libraries, including Pandas for data manipulation, NumPy for numerical operations, and SciPy for more complex statistical functions, such as z-score normalization.

3.2.2 Major Code Blocks

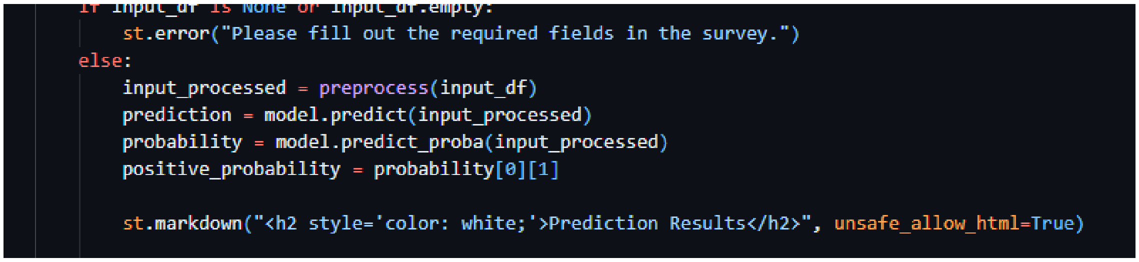
Custom CSS Integration: To achieve the desired aesthetic and user experience, custom CSS was employed seen in Figure 18. This included hiding the hyperlink icon that Streamlit typically displays next to headers and fine-tuning the styling of widget labels, buttons, and the footer.



```
font_size2 = "20px"
# Main welcome message
st.markdown(f"<h1 style=font-size: {font_size2};>Ovarian Cancer Risk Detection</h1>", unsafe_allow_html=True)

def custom_css():
    st.markdown("""
        <style>
            /* General selectors for widget labels */
            .stTextInput label, .stSelectbox label, .stNumberInput label {
                color: white !important;
            }
            /* General selectors for widget descriptions inside the label tag */
            .stTextInput .st-b7, .stSelectbox .st-b7, .stNumberInput .st-b7 {
                color: black !important;
            }


        </style>
        """, unsafe_allow_html=True)

custom_css()  # Call this at the start of your app, before any other Streamlit commands
```

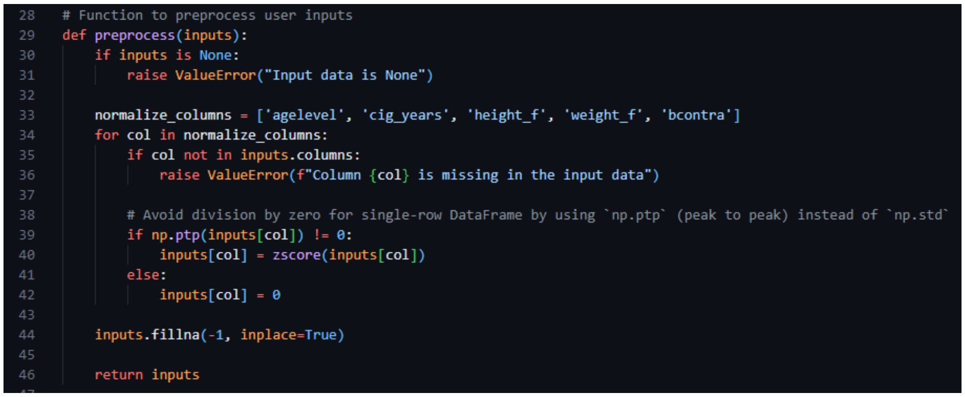
Figure 18: Custom CSS Code

Model Integration and Preprocessing: A significant portion of the code is dedicated to loading the pre-trained machine learning model and defining a function for preprocessing the user inputs which can be seen in Figure 19. This ensures that the data provided by users through the web interface is in the correct format for the model to process accurately.



```python
if input_df is None or input_df.empty:
    st.error("Please fill out the required fields in the survey.")
else:
    input_processed = preprocess(input_df)
    prediction = model.predict(input_processed)
    probability = model.predict_proba(input_processed)
    positive_probability = probability[0][1]

    st.markdown("<h2 style='color: white;'>Prediction Results</h2>", unsafe_allow_html=True)
```

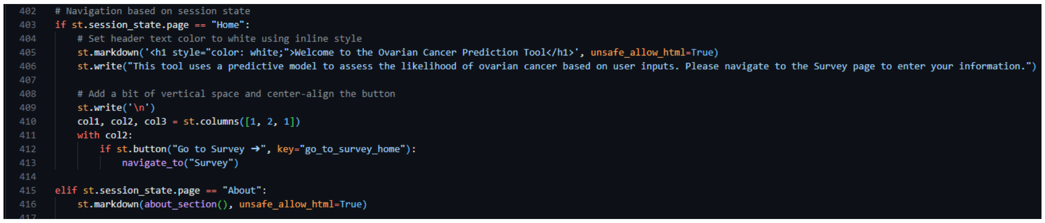
Figure 19: Model Integration



```python
# Function to preprocess user inputs
def preprocess(inputs):
    if inputs is None:
        raise ValueError("Input data is None")

    normalize_columns = ['agelevel', 'cig_years', 'height_f', 'weight_f', 'bcontra']
    for col in normalize_columns:
        if col not in inputs.columns:
            raise ValueError(f"Column {col} is missing in the input data")

        # Avoid division by zero for single-row DataFrame by using `np.ptp` (peak to peak) instead of `np.std`
        if np.ptp(inputs[col]) != 0:
            inputs[col] = zscore(inputs[col])
        else:
            inputs[col] = 0

    inputs.fillna(-1, inplace=True)

    return inputs
```

Figure 20: Pre-Processing Code Integration

Interactive Widgets and Base64 Encoding: Interactive elements, such as buttons and input fields, were implemented to collect user data. Additionally, we used base64 encoding to embed images, such as the project's logo, directly within the application displayed in Figure 21.



```python
# Navigation based on session state
if st.session_state.page == "Home":
    # Set header text color to white using inline style
    st.markdown('<h1 style="color: white;">Welcome to the Ovarian Cancer Prediction Tool</h1>', unsafe_allow_html=True)
    st.write("This tool uses a predictive model to assess the likelihood of ovarian cancer based on user inputs. Please navigate to the Survey page to enter your information.")

    # Add a bit of vertical space and center-align the button
    st.write('\n')
    col1, col2, col3 = st.columns([1, 2, 1])
    with col2:
        if st.button("Go to Survey ➔", key="go_to_survey_home"):
            navigate_to("Survey")

elif st.session_state.page == "About":
    st.markdown(about_section(), unsafe_allow_html=True)
```

Figure 21: Interactive and Base64

3.2.3 Application Design and Flow

The application flow for the Ovarian Cancer Risk Detection tool is designed to be intuitive and linear, as depicted in the provided flow diagram. Upon entering the application, the user lands on the 'Home Page,' which serves as the starting point. From here, they encounter the 'Navigation Process,' a critical juncture that channels the user experience into three primary pathways: proceeding to the 'About Page,' where the user can read more about the application and its purpose; moving to the 'Survey Page,' where they input personal and medical data; or returning to the 'Home Page.'

The 'Survey Page' leads users through a series of questions designed to gather necessary information for risk assessment. Once the user completes the survey, their responses are processed ('Prediction'), and the machine learning model generates a risk prediction. The results of this prediction are then displayed to the user in a clear and accessible manner ('Displayed Results').

The flow diagram seen in Figure 22 illustrates the application's user-centric design, emphasizing an efficient and logical progression through the tool's various components.

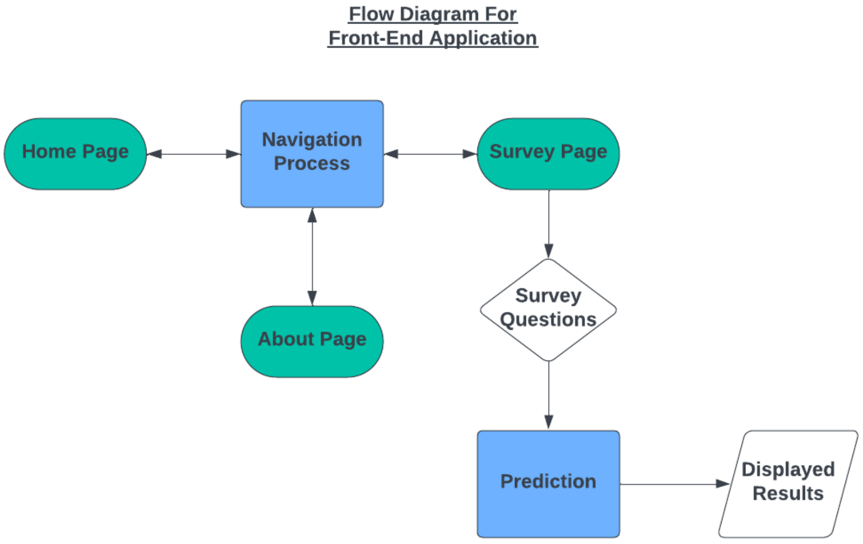
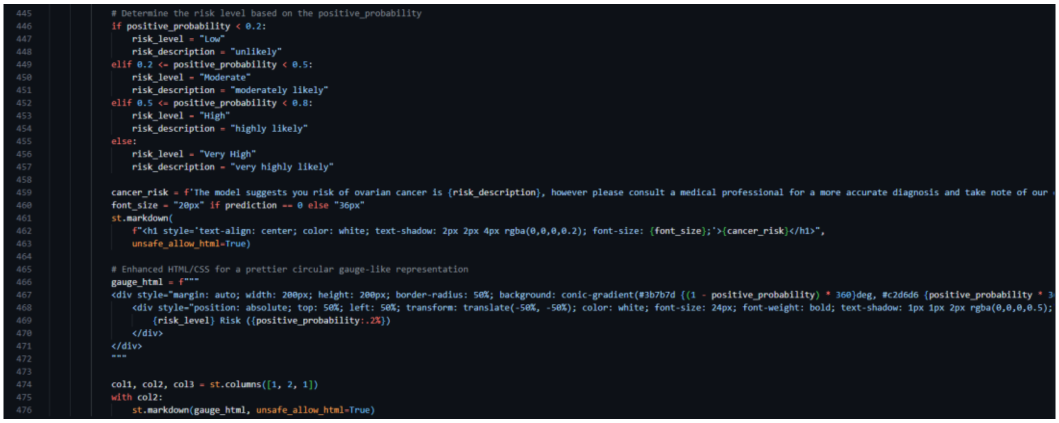


Figure 22: Application Flow Diagram

3.2.4 Visualization and Results Display

Upon submission of their data, users are presented with a results page that visually communicates their risk assessment. The implementation of this feature involved CSS and HTML to render a circular gauge-like graphic that represents the user's risk level.



```
# Determine the risk level based on the positive_probability
if positive_probability < 0.2:
    risk_level = "Low"
    risk_description = "unlikely"
elif 0.2 <= positive_probability < 0.5:
    risk_level = "Moderate"
    risk_description = "moderately likely"
elif 0.5 <= positive_probability < 0.8:
    risk_level = "High"
    risk_description = "highly likely"
else:
    risk_level = "Very High"
    risk_description = "very highly likely"

cancer_risk = f'The model suggests you risk of ovarian cancer is {risk_description}, however please consult a medical professional for a more accurate diagnosis and take note of our
font_size = "20px" if prediction == 0 else "36px"
st.markdown(
    f"<h1 style='text-align: center; color: white; text-shadow: 2px 2px 4px rgba(0,0,0,0.2); font-size: {font_size};'>{cancer_risk}</h1>",
    unsafe_allow_html=True)

# Enhanced HTML/CSS for a prettier circular gauge-like representation
gauge_html = f"""
<div style="margin: auto; width: 200px; height: 200px; border-radius: 50%; background: conic-gradient(#3b7b7d {(1 - positive_probability) * 360}deg, #c2d6d6 {positive_probability * 3
    <div style="position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%); color: white; font-size: 24px; font-weight: bold; text-shadow: 1px 1px 2px rgba(0,0,0,0.5);
        {risk_level} Risk ({positive_probability:.2%})
    </div>
</div>
"""

col1, col2, col3 = st.columns([1, 2, 1])
with col2:
    st.markdown(gauge_html, unsafe_allow_html=True)
```

Figure 23: Prediction Wheel Code Block

3.2.5 Bill of Materials and Budget for the Front-End Application

The software development aspect of the project primarily involved GitHub Codespaces for collaborative coding and version control. Python's Joblib library was employed for model loading, and Matplotlib was used for creating visual representations of the risk assessment. All the software used was open-sourced, which significantly minimized costs. The detailed list of software components is as follows:

Table 3: List of Software used.

Software	**Cost**
Streamlit	Free (Open source)
Pandas, NumPy, SciPy	Free (Open source)
Matplotlib	Free (Open source)
GitHub Codespaces	Based on usage
Joblib	Free (Open source)
Python	Free (Open source)

Through careful selection of software tools and an emphasis on user-centric design, the front-end application of the Ovarian Cancer Risk Detection tool was implemented. The application provides a streamlined experience from data input to risk prediction, made possible by the strategic use of Streamlit, custom styling, and thoughtful navigation layout. The result is an application that effectively balances functionality with usability, all while maintaining a minimal project budget.

4 Testing and Evaluation

Testing and evaluation form a critical part of the development cycle of the "Early Warning of Ovarian Cancer" system, ensuring that both the machine learning model and the front-end application meet the highest standards of quality, reliability, and user satisfaction.

4.1 Machine Learning Model Testing and Evaluation

The general goal for the machine learning model was to build up positive and negative classifiers and increase the accuracy overall. However, due to the issue of the imbalanced dataset, that assumption was not satisfactory for testing as accuracy alone produces misleading conclusions. This is due to the fact it would only consider the classifier's general performance and not the specific performance for each class. Therefore, we must consider more metrics when evaluating the imbalanced class problem. The most common way to evaluate a machine learning model is a confusion matrix which allows the comparison of predictions to calculate multiple evaluation metrics.

A binary classification problem such as this one has 4 different sections, True Negatives, True Positives, False Negatives, and False Positives. From these values, we can produce the recall, precision, and F1 metrics which allow us to better understand our model class-specific performance. Precision is the measurement of correctly predicted positive class values, recall compares correctly predicted negative class values, and the F1 score is the comparison between the two. The recall metric was chosen as the main testing measurement because the group felt it was better to falsely classify a woman as having ovarian cancer when she does not then miss a woman who does have it. Therefore, all models and classifiers were generally measured based on their recall score rather than their overall accuracy. All testing was done on 10% of the dataset, as mentioned in 2.1.2, which was removed at the beginning of the model training process. Throughout the testing process results were used to cycle back and make improvements to existing models. These metrics were a benchmark showing model performance and giving the team insight into where mistakes could have been made. From there, the model ran through numerous iterations until finding the best-performing classifier.

Table 4: Final Model Results

Model	True Positive	False Positive	False Negative	True Negative
Random Forest	14958	300	50	1549

Table 4 shows the testing results of the top-performing model, even though other models had slightly higher accuracy scores. This model had the strongest recall score, which is what was most important. Along with the typical model prediction, the group considered including a custom threshold value within the model. Most binary classifiers use a threshold of 0.5. That is if a value is above a 50% prediction it is the one class and if it is below, it is the other. We can lower this threshold value for the negative cases which in turn raises our recall score at the cost of precision which was the group's goal at the start.

The final model, with a threshold of 0.1, had an accuracy of 94%, a recall of 90%, a precision value of 70%, and a total F1 score of 78%. When evaluating the final model results and comparing them to the goals the team set at the start of the project, we can see the final product met our goals. The team had an expectation the model would have a minimum of 80% recall with an 85% accuracy. Based on evaluation the model can effectively predict the risk of ovarian cancer as was the goal.

4.2 Application Testing and Evaluation

At the start of the development process, the team set several objectives for the front-end application including functionality, user experience, and usability. To ensure that the developed application met these objectives, it went through testing and evaluation in a variety of scenarios. The main source of evaluation was from the expected user group, 10 females age ranging from twenty to fifty years old. These users were able to provide the most relevant feedback as they were the individuals who would be using the finished application. Moreover, the development team was all male and therefore many details were glossed over and noticed later by the female testing group. The testing group was a collection of friends and family of the team. They volunteered to apply their unbiased eyes and critique the design of the application.

The two main sources of testing done on the application were visual testing on the site as a whole and questionnaire testing focusing specifically on the wording and delivery of questions within the survey. The testing was all qualitative focusing on the metrics that could not be numbered such as user understanding and visual appearance.

The first aspect that the application was tested for was the visual design. Each member of the testing group was given the web application and asked to provide feedback on the visual appearance. Tests were done in a semi-structured interview style with members of the group getting real-time feedback. The design went through several iterations, each one based on feedback the group received from users. The group received three main points of feedback which went into the creation of the final design.

The first point the group observed was that the main logo, consistently used in our reports, did not fit well aesthetically on the website's front page. This led us to replace it with a simpler logo more representative of ovarian cancer awareness for better visual harmony and relevance. The second visual change was to the information page in the application. Users described this page as being too cluttered and felt that the information was just thrown onto the page. To improve this the group reformatted the ‘About tab’, adding section headers, changing the font color and size, and centering the text on the page. The final visual feedback was related to the survey tab, users felt the outputted prediction message along with the risk probability was not appealing. To combat this the team implemented a percentage wheel indicating the risk in a pie chart-like fashion. All user input played vital roles in making the application more visually appealing for the designated user group.

The second aspect that the team tested the application on was understandability and eligibility ensuring that patients working through the survey understood the questions they were asked and could answer them without confusion. This testing was vital in the process especially due to the lack of gender diversity within the development team. The test group was asked to go through the survey questions and remark on any questions they felt were potentially ambiguous or did not make sense as a reader. An example of this feedback was related to the “Family History of Cancer” question which asks if the individual family has a history of cancer. Users were unsure whether to consider only their immediate family or extend it to more distant relatives like 2nd and 3rd cousins, indicating a need for greater specificity in the wording. To fix this problem, the group changed the wording to “Does any of your extended family have a history of Cancer” giving the user a clearer image of what is being asked. Another example of feedback was for the “Birth Control” question along with the “Hormone” question. It was stated that the hormone question was confusing to users due to the fact birth control is a hormone. They had no idea whether they should answer yes to both or if birth control was excluded from the second question.

This kind of feedback was vital, as the all-male development team has no direct experience with aspects such as birth control or female hormones. The hormone question was updated to ask specifically about hormone-altering drugs, excluding birth control. Thanks to input from the intended users, survey questions were revised for clarity, resulting in a final version that was user-friendly and well-suited to the needs of those who would be using the app.

4.3 Solution Performance Comparison

In our project, we undertook a unique approach by focusing exclusively on survey data from the PLCO dataset to predict ovarian cancer risk. This perspective distinguishes our work from several other studies that have predominantly concentrated on biomarker data. Our final model achieved an impressive accuracy of 94%, with a recall of 90%, precision of 70%, and an F1 score of 78%. The high recall rate is particularly noteworthy in the medical field, as it reflects the model's ability to correctly identify most positive cases, a crucial factor in early cancer detection.

Comparatively, a notable study presented at the 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), titled "Biomarker CA125 Feature Engineering and Class Imbalance Learning Improves Ovarian Cancer Prediction," uses a decision tree enhanced by SVMSMOTE on the

PLCO dataset [4]. This study reported a positive predictive value (PPV) of 0.9041, an AUC of 0.9532, a sensitivity of 0.7792, and an exceptionally high specificity of 0.9982 [4]. While their specificity and AUC are commendable, our model demonstrates a superior balance between sensitivity (recall) and precision, ensuring fewer false negatives—a critical aspect in cancer screening where missing a positive case can have bad consequences.

Another broader study, "Symbolic One-Class Learning from Imbalanced Datasets: Application in Medical Diagnosis," explored a one-class learning approach [2]. This study faced challenges in applying machine learning strategies to medical diagnosis, as indicated by its moderate performance across various metrics, with the best performances not surpassing an 80% threshold in AUC, geometric mean, and ranker, except for a specific dataset (Hyperthyroid) [2]. The study highlights the inherent difficulties in medical dataset classification, exacerbated by the need to finely tune additional parameters for class imbalance strategies like oversampling. In contrast, our approach, by focusing on survey data and employing Random Forest classification, managed to navigate these challenges more effectively, achieving a high level of accuracy and recall without the need for extensive parameter tuning.

5 Project Planning and Budgeting

To ensure the group met all project goals the team created a project plan near the start of the term to outline work that needed to be done, rough estimates on due dates, and which members would be focused on what sections. In the table below we have outlined the original project plan along with a new column containing information on when goals and milestones were met.

Table 5: Project Plan and Work Distribution

No.	Milestone	Due Date	Responsible	Actual Completion
1	**Data Pre-Processing**			**December 1, 2023**
1.1	Initial Assessment	November 10, 2023	Matthew, Lucas	November 13, 2023
1.2	Feature Selection & Analysis	November 17, 2023	All group members	November 20, 2023
1.3	Data Cleaning &Normalization	November 24, 2023	Kieran, Mile	November 26, 2023
1.4	Review of Pre-Processed Data	November 30, 2023	All group members	November 30, 2023
2	**Model Development**			January 12, 2024

2.1	Initial Model Design & Algorithm Selection	December 5, 2023	Lucas, Mile	December 13, 2023
2.2	Prototype Development	December 10, 2023	Matthew, Kieran	January 4, 2024
2.3	Model Training &Optimization	December 15, 2023	All group members	January 10, 2024
2.4	Model Validation &Refinement	December 20, 2023	Lucas, Kieran	January 12, 2024
3	**UI Development**			January 20, 2024
3.1	UI Mock-up & Design	January 15, 2024	Mile, Lucas	January 14, 2024
3.2	UI Implementation	January 25, 2024	Matthew, Kieran	January 17, 2024
3.3	UI Testing & Feedback	January 29, 2024	All group members	January 19, 2024
4	**Integration, Test & Deployment**			May 7, 2024
4.1	System Integration	February 2, 2024	All group members	February 1, 2024
4.2	Testing & Debugging	February 5, 2024	Kieran, Lucas	February 29, 2024
4.3	Deployment Preparation	February 10, 2024	Mile, Matthew	March 2, 2024
4.4	Final Deployment & Go-Live	February 20, 2024	All group members	March 6, 2024

When looking back at the project plan in Table 5 a few notable differences stand out. The first is that the completion of the project was well past the initial deadline set. This is due to the fact initially the team had been under the impression the project had to be completed by the end of February but due to changes the final poster presentation did not end up being until March. This allowed the team to spend extra time improving and finalizing factors, improving the final product.

Another noticeable difference was the UI development took much less time than expected. The transition from React Native to Streamlit allowed our team to simplify the development process. By using Streamlit's straightforward library, we could dedicate more time and resources to refine other crucial aspects of the project. One aspect we didn't foresee was the ongoing need for model refinement. Since finalizing our initial model, we've been continuously working to improve it, which has consumed a significant portion of our time. Model improvement continued right up until the writing of this report ensuring the final product was the best possible version.

Another aspect that the group predicted at the start of the project was the final budget, which was combined with the timeline to create a cost estimation table.

Table 6: Cost Estimation and Final Result

Item	Purchase Date	Current Cost	Future Cost
Google Colab	December 1st, 2023	$0	$14
Google Colab	January 1st, 2023	$0	$14
Google Colab	February 1st, 2023	$0	$14
Google Colab	March 1st, 2023	$0	$14
Total	$56		

Table 6 was created with the assumption the group would require Google Colab for model training. The project's budget turned out to be zero. We managed to build a model that didn't need extra computing power, so we didn't have to pay for services like Colab. We used free open-source libraries and received the NCI PLCO dataset at no cost, which covered all our data needs. This allowed us to finish the project without any cost, which was a significant achievement for the team.

6 Stakeholders and Other Factors

The successful implementation of the "Early Warning of Ovarian Cancer" system includes a spread of considerations that extend beyond the technical domain, encompassing social safety, environmental impact, and economic implications. Each of these aspects is crucial in aligning the project with the broader interests of stakeholders.

To identify the main stakeholders the team considered groups of people who would have an interest in the completion and progress of the project. Considering those who would get the most benefit or receive the most harm from the final product. The group decided to focus on four main stakeholder groups. Ethical and legal experts, this group surrounds all things related to the product's legality. Anything to do with saving or storing data, especially in the medical sense is always a legal issue and the team wants to ensure all practice is up to industry standard. Another group is the patients and advocacy groups themselves. These people will be directly affected by completion, they are the end users and promoters, and their needs must be considered. The third group is the medical professionals, these are the doctors or gynecologists who will be suggesting or promoting the application. The final group is the sponsors of this project, specifically Queen’s University, focusing on how their support would be affected by the project direction and more importantly, cost.

6.1 Ethical and Legal Experts

For any project relating to user information, especially medical information, the legal implications are going to be significant and must be adhered to. From the beginning of the design process, the group understood these requirements and held ethical and legal implications as a key consideration throughout the process. The group was given a dataset from the NCI containing patient medical data. The group was required to sign an NDA ensuring that the data would not be shared or used to profit in any way. Adhering to the legal requirements the group made sure the data was never in a location that had public access and the dataset was not shared with any outside users. For code organization and work the team used a GitHub Repository but made sure the dataset was never shared within it, the excel sheet was kept on local computers reducing any risk of data leak in consideration of legal implications.

When the group had completed the front-end of the application and the product was ready to begin testing the results the group agreed no user data would be saved. The process of storing data would have improved the model's accuracy and performance, but the team could not guarantee data privacy or safety and therefore felt it was better to not save any user data at all. This is further backed up by StreamLit’s agreement to data minimization, ensuring any applications built using the platform will only save the data necessary and will not hold unseen caches of information. It was assumed woman would be more open to using the application if they were not worried about their data being saved and sold off or used. The survey contains a couple of questions that are quite personal, and the team wants women to feel completely comfortable using the application.

As the group looks into the potential future of this product, the legal and ethical aspects remain an important consideration. The group has considered asking women to allow the application to store their data and build a form of profile for each user. To do this the team would need to invest in top-level security software ensuring data is secure and will only be used for model improvement. The group would also make a promise to never share this data for monetary gain, focusing on the betterment of the prediction app and women's health in general.

6.2 Patients and Advocacy Groups

In the case of this application, the primary stakeholders are those who will be in direct use of the product. This would be any patients, likely females between the ages of 20-60 who are interested in learning about their risk of ovarian cancer. Included in this category are advocacy groups that would promote the product to a wider spread of users trying to share the positives of the finished product. There were several key design and implementation decisions made throughout the process with this specific stakeholder in mind. The first design decision was focused on patient usability and ease of access. The group was given two large datasets, one containing simple survey answers and the other containing biomarkers that are less common for patients to have access to. The group had the option to base the model on a combination of the two sets but decided to focus on the survey data. This way there would be less of a barrier for women to use the application, they would not require any prior testing and could get immediate results which would then lead them to the next steps. Throughout development, we consistently emphasized that our team isn't medically trained and that our app's results are not medical diagnoses but rather predictions and assessments of risk. To emphasize this point, we've included disclaimers and reminders at every stage within the app, from the About page to the final prediction output.

The final step our team took was to engage with the intended users—women—in testing the survey. Their feedback helped us shape the app to meet the specific needs and expectations of our patient stakeholders.

6.3 Medical Professionals

Our team engaged with gynecologists, who are experts in ovarian cancer, early in our development process. Although these medical professionals wouldn't use the app themselves, their role was crucial in endorsing it to patients and using it as an early detection aid. We ensured the app's credibility by incorporating their expertise to determine the most predictive factors of ovarian cancer risk. This collaboration helped build a trustworthy product that doctors can confidently recommend to patients. Furthermore, throughout the project, adherence to industry standards for medical products was paramount. Our model was carefully designed to exclude factors like race or ethnicity to prevent biases, ensuring it aligns with ethical medical care practices. The result is a tool that medical professionals can confidently recommend, providing them with a supplementary resource in their diagnostic process.

6.4 Queen’s University

The final stakeholder that the team considered throughout the design process was the Queen’s support team. Specifically, the budget and the feasibility of creating the product up to a certain standard. In designing our app, we viewed Queen's support team as investors, aiming to create a cost-effective product. With zero material costs due to our access to necessary devices and utilizing free software like Visual Studio Code and StreamLit we managed to keep our budget at zero. We chose a simpler machine-learning model that didn't need extra resources like Google Colab's paid service. This approach met our efficiency goals and allowed us to deliver the project with no additional funding needed from Queen's. Moving forward, any new costs will be covered by the team or new sponsors.

7 Compliance with Specifications

Table 7: Specifications from Blueprint

1	**Functional Requirements**	**Specification Met**
1.1	Machine learning model for assessing user inputs and predicting ovarian cancer likelihood.	Yes
1.2	Data preprocessing to clean, format, and extract relevant data from user input, ensuring data suitability and vital information inclusion.	Yes
2	**Interface Requirements**	
2.1	Inputs: User inputs from a designated questionnaire covering essential data and the NCI-provided dataset with approximately 78,000 entries.	Yes
2.2	Outputs: Model's risk analysis of ovarian cancer likelihood, including disclaimers, suggestions for next steps, and direct result explanations on the user interface.	Yes

2.3	User Interface: Tabs for data entry, result display, contact information, and user-friendly elements like a progress bar for data visualization.	Yes
2.4	Development Languages: Python for the model (with support from libraries like PyTorch and TensorFlow); JavaScript using React Native for the front-end interface.	No
3	**Performance Requirements**	
3.1	Fast response times and stable connection to the back end.	Yes
3.2	User-friendly design for enhanced customer experience.	Yes
3.3	Speed and accuracy in the machine learning model, with a focus on minimizing inference time.	Yes
3.4	High precision in model accuracy, prioritizing reducing the risk of missing actual cases, even at the expense of a lower recall rate.	Yes

When reflecting on the specifications in Table 7 outlined in the Blueprint document, we can see that the project has met 90% of the outlined requirements. These requirements were a collection of functional, interface, and performance-based goals the team outlined as landmarks to reach. Functional requirements outlined what the program would do, these specifications were vital to the completion of the project and included goals such as a machine learning prediction model. The interface requirements were related to the process of completing the project, the inputs and outputs, and what sort of tools would be used.

The project met all specifications for the inputs, outputs, and user interface section but fell short in the development languages section. The team set the goal of using JavaScript with React Native to develop the front end but ended up using a Python library called StreamLit. This change allowed the group to focus more on developing a working model while also not sacrificing a user-friendly application.

The final specification section was the performance section that focused on giving the model specific benchmarks and goals the team aimed for it to hit. This included factors such as accuracy and recall rate on the model and usability of the application. All specification goals within performance were met so the group considers the project a success.

8 Conclusion and Recommendations

The "Early Warning of Ovarian Cancer" project has reached a successful conclusion, achieving the critical objective of developing an early detection system for ovarian cancer. The team's dedication and expertise led to the creation of a predictive tool driven by a robust machine learning model, which addresses the urgent need for improved diagnostic methods in women's health. Using SMOTE to combat class imbalance within the data set, coupled with the Random Forest algorithm for classification, has resulted in a powerful model with high predictive accuracy. The front-end application, integrated with the machine learning model and crafted using the user-friendly Streamlit framework, delivers an accessible and intuitive interface for users to assess their risk of ovarian cancer. The team has consistently upheld the highest standards of ethical responsibility and legal compliance, particularly in the realm of data privacy, by choosing not to retain any personal data after interaction with the tool. The incorporation of medical expertise and the iterative refinement based on user feedback have ensured that the tool remains accurate, trustworthy, and aligned with the user's needs.

The project's achievements lay the foundation for several recommendations that can extend its impact and effectiveness. Firstly, ongoing improvement of the machine learning model is advised, with the integration of evolving medical research and the exploration of advanced data analysis techniques to refine the tool's accuracy further. Expanding the scope of user testing, particularly across diverse demographics, will enhance the reliability of the model's predictive power and ensure broader applicability. In the realm of user education, the development of supplementary materials to inform users about ovarian cancer and interpret the tool's findings is recommended to reinforce its role as an adjunct to professional healthcare. As the tool potentially evolves to include user data retention for model refinement, it is imperative to implement rigorous data protection and security measures to maintain user trust and comply with privacy regulations. The formation of partnerships with healthcare providers and advocacy groups can provide validation and credibility, fostering wider acceptance and use of the tool. Improving the tool's accessibility features and considering localization for non-English speaking users will make it more inclusive and global in reach. Finally, establishing structured feedback mechanisms within the application will be vital for continuous performance enhancement and ensuring that the tool evolves in line with user needs and expectations.

In essence, the "Early Warning of Ovarian Cancer" project is a pivotal step toward revolutionizing the early detection of a disease that affects women worldwide. The team's ability to merge technical innovation with practical application has resulted in a tool that not only meets the current healthcare challenges but also sets a precedent for future advancements in the field.

9 References

[1] D. &. B. R. C. J. (. Badgwell, "Early detection of ovarian cancer. Disease markers, .," *https://doi.org/10.1155/2007/309382,* pp. 23(5-6), 397–410, 2017.

[2] J. G. Luis Mena, "Symbolic One-Class Learning from Imbalanced Datasets: Application in Medical Diagnosis," *International Journal of Artificial Intelligence Tools,* vol. 18, no. 10.1142/S0218213009000135, pp. 200-287, 2009.

[3] National Cancer Institute, "Trial Summary," 2023. [Online]. Available: https://cdas.cancer.gov/learn/plco/trial-summary/.

[4] M. K. a. K. S. X. Yang, "Biomarker CA125 Feature Engineering and Class Imbalance Learning Improves Ovarian Cancer Prediction," *2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE),* vol. 1, no. 10.1109/CSDE50874.2020.9411607, pp. 1-6,, 2020.

[5] M. H. X. L. N. Z. a. H. C. L. Wang, "Review of Classification Methods on Unbalanced Data Sets," *IEEE Access ,* vol. 9, no. 10.1109/ACCESS.2021.3074243, pp. 64606 - 64628, 2021.

[6] G.-B. B. S. L. K. S. A. K. S. W. W. ohamed Elhoseny, "Effective features to classify ovarian cancer data in internet of medical things," *Computer Networks,* vol. 159, no. doi.org/10.1016/j.comnet.2019.04.016., pp. 147-156, 2019.

[7] R. C. J. B. M. Z. C. H. M. A. D. M. O. R. L. K. L. Z. B. D. M. G. B. S. S. Z. Z. C. D. L. A. &. Y. Y. Bast, "Prevention and early detection of ovarian cancer: mission impossible?," *Recent results in cancer research.,* vol. 174, no. 0.1007/978-3-540-37696-5_9, p. 91–100, 2007.

[8] D. Freidman, Interviewee, *Ovarian Cancer Discussion with gynecologist.* [Interview]. 18 01 2024.

[9] K. W. B. L. O. H. W. P. K. N. V. Chawla, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research,* vol. 16, no. 10.1613/jair.953, pp. 321–357,, 2002.

[10] NCI, "Cancer.gov," 2023. [Online]. Available: https://cdas.cancer.gov/plco/.

[11] C. S. Z. e. al, "The Prostate, Lung, Colorectal, and Ovarian Cancer Screening Trial and Its Associated Research Resource," *Journal of the National Cancer Institute,* pp. 1684-1693, 2013.

[12] M. S. K. C. L. C. M. M. Hossein Khonsari, *T.A Meeting 1,* Kingston, Ontario, 2023.

[13] M. S. K. C. L. C. M. M. Hossein Khonsari, *T.A Meeting 2,* Kingston, Ontario, 2023.

[14] M. S. L. C. M. M. Kieran Cosgrove, "ELEC 490-498-Project Proposal," Kingston, ON, 2023.

[15] C. C. &. V. Vapnik, "Support-Vector Networks," *Springer Link,* no. 20, p. 25, 15 05 1995.

Appendices

Table 8: Group Participation

Name	Overall Effort Expended
Lucas Coster	100%
Matthew Mamelak	100%
Kieran Cosgrove	100%
Miodrag Stosic	100%