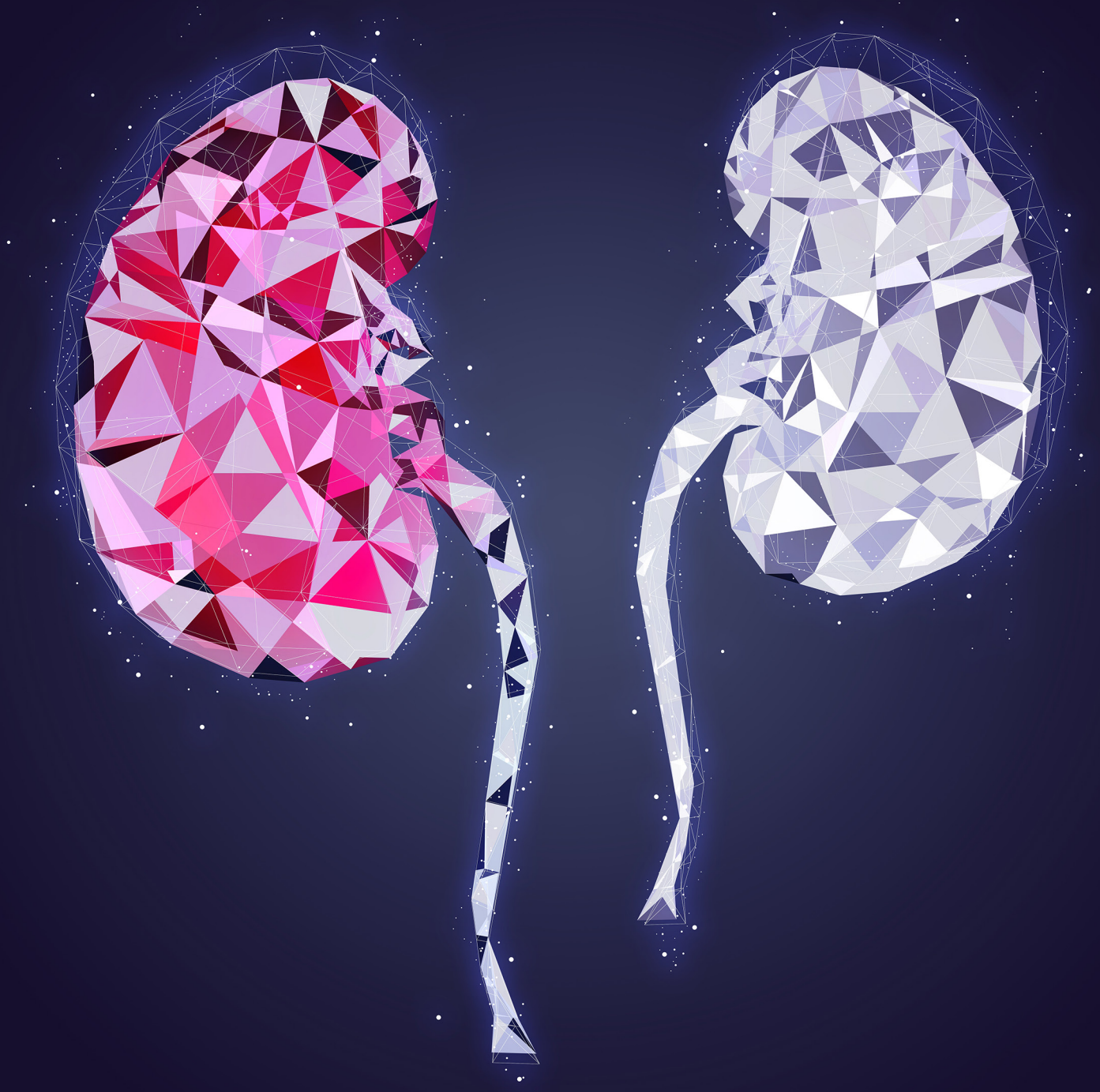
mlangles

mlangles Predictive AI

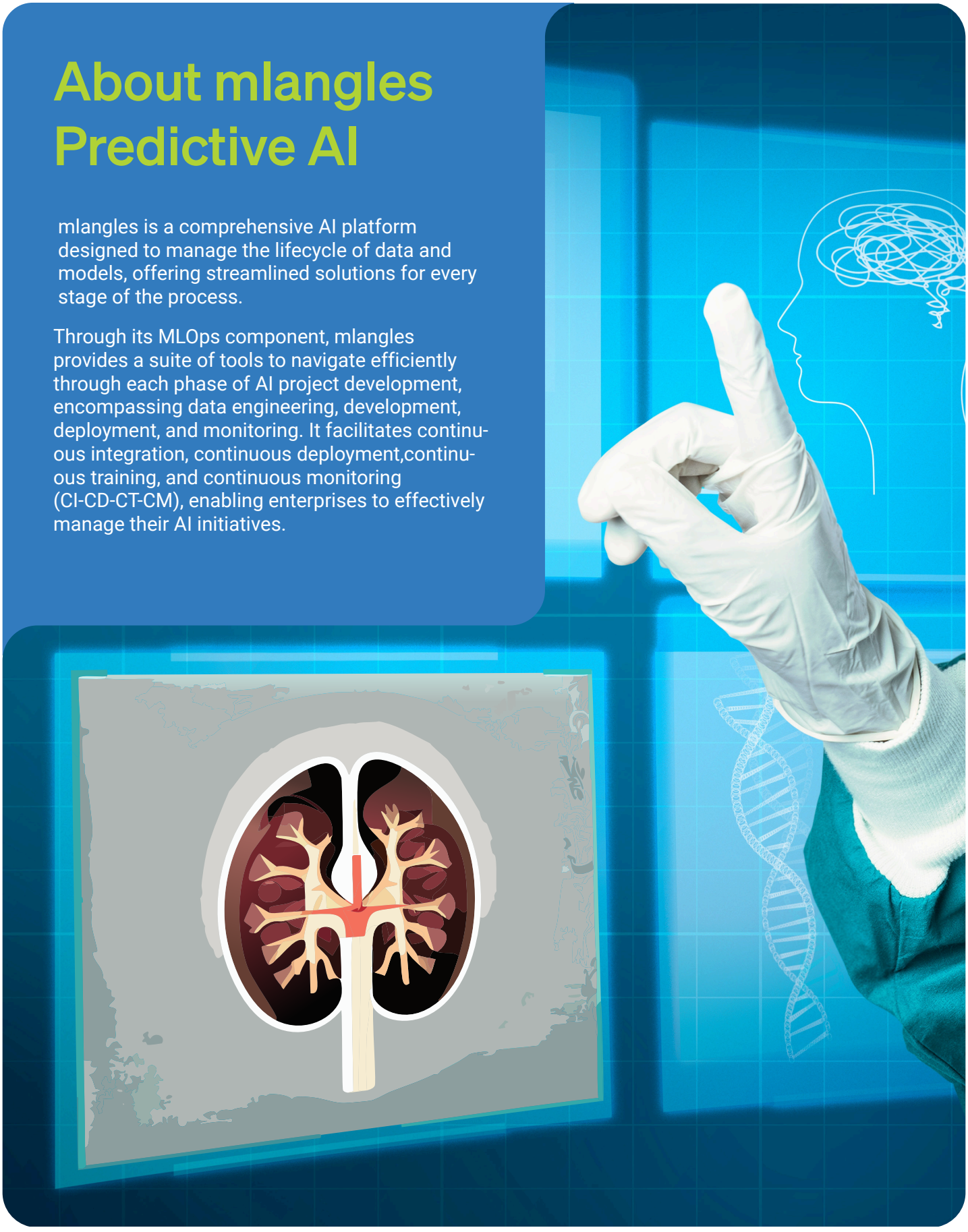
CHRONIC KIDNEY DISEASE

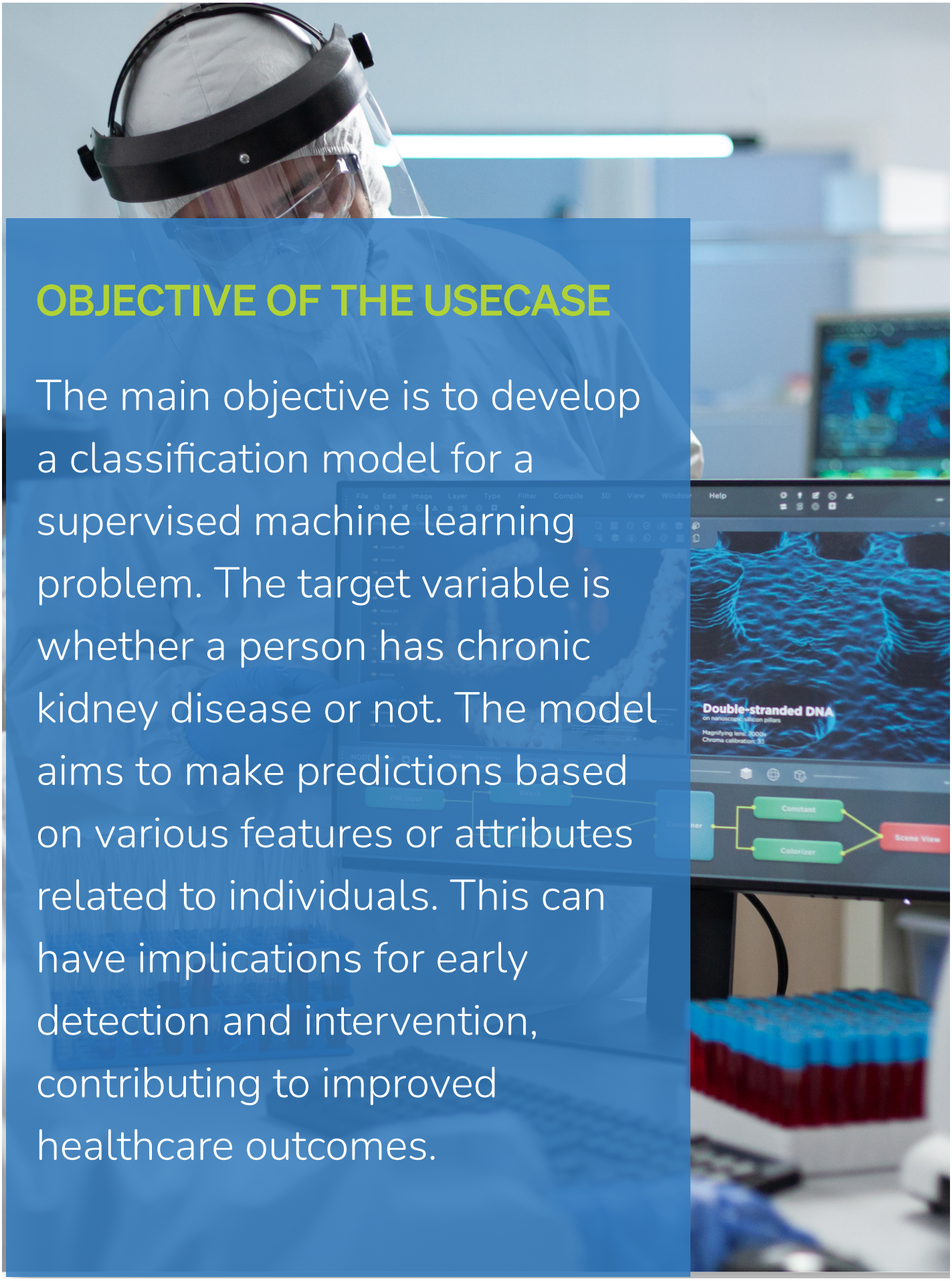


About mlangles Predictive AI

mlangles is a comprehensive AI platform designed to manage the lifecycle of data and models, offering streamlined solutions for every stage of the process.

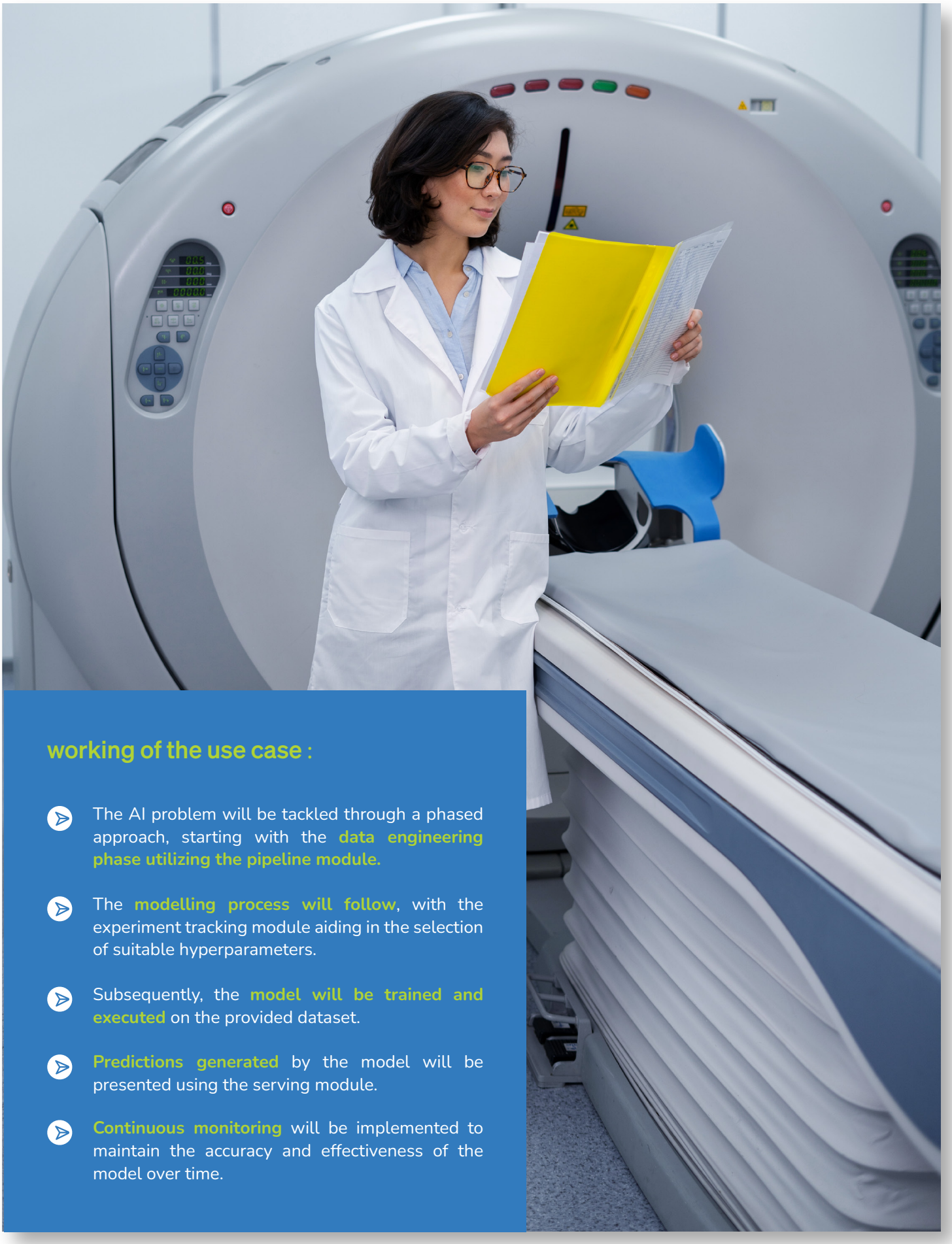
Through its MLOps component, mlangles provides a suite of tools to navigate efficiently through each phase of AI project development, encompassing data engineering, development, deployment, and monitoring. It facilitates continuous integration, continuous deployment,continuous training, and continuous monitoring (CI-CD-CT-CM), enabling enterprises to effectively manage their AI initiatives.





OBJECTIVE OF THE USECASE

The main objective is to develop a classification model for a supervised machine learning problem. The target variable is whether a person has chronic kidney disease or not. The model aims to make predictions based on various features or attributes related to individuals. This can have implications for early detection and intervention, contributing to improved healthcare outcomes.



working of the use case :

- The AI problem will be tackled through a phased approach, starting with the **data engineering phase utilizing the pipeline module.**
- The **modelling process will follow**, with the experiment tracking module aiding in the selection of suitable hyperparameters.
- Subsequently, the **model will be trained and executed** on the provided dataset.
- **Predictions generated** by the model will be presented using the serving module.
- **Continuous monitoring** will be implemented to maintain the accuracy and effectiveness of the model over time.

EXPLAINATION OF THE USECASE:

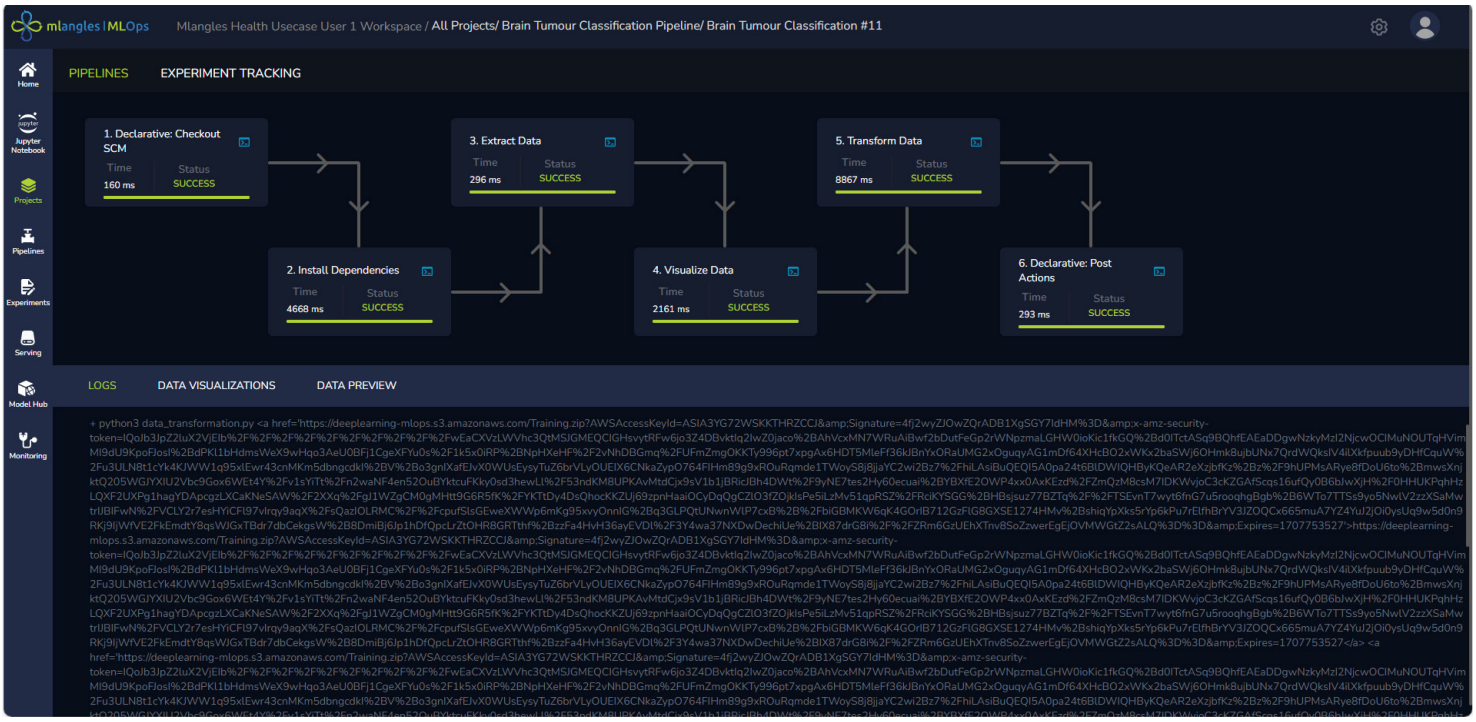
The data was taken over a 2-month period in India with 25 features (Example: 'Blood Pressure', 'Sugar', 'RedBlood-Cells', 'Pus Cells', 'Pus Cells Cumps', 'Bacteria', 'Blood Urea', 'WhiteBloodCellsCount', 'RedBloodCellsCount', 'Hypertension', 'coronary artery disease', 'Appetite', 'Pedal Edema', 'Anemia').

In a scenario where all these health indicators are within normal ranges, it might suggest that the patient is in good health and does not have chronic kidney disease. However, abnormal values or the presence of certain conditions might raise concerns indicating a need for further medical investigation. A comprehensive analysis of these features helps assess the patient's overall health and determine the likelihood of chronic kidney disease.

The target variable, or the outcome to be predicted, is a binary classification indicating whether an individual has chronic kidney disease ('ckd') or does not have chronic kidney disease ('notckd'). This implies that the machine learning model developed based on this dataset aims to predict the presence or absence of chronic kidney disease based on the provided health-related features.

The dataset has 400 rows with each of the 2 categories chronic kidney disease and not chronic kidney disease.

Step 1: Data Engineering & pipeline creation



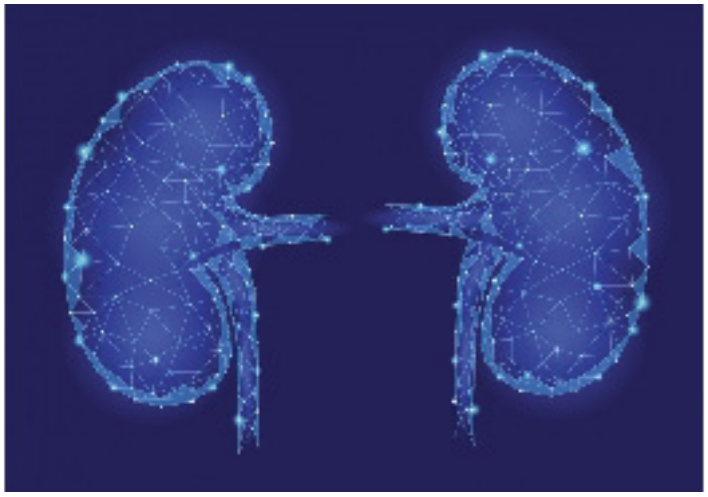
Install Dependencies: Essential packages and libraries have been installed.

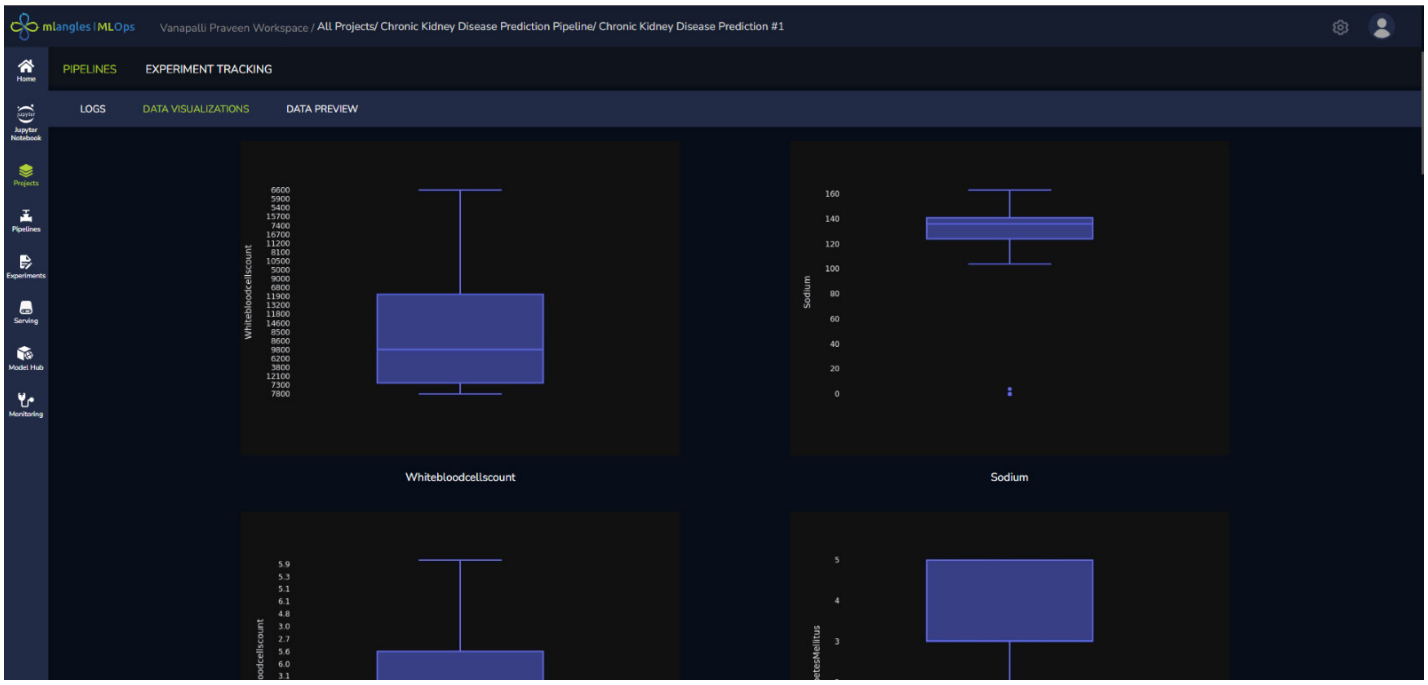
Data Extraction: Data extraction involves gathering raw data from the CSV file.

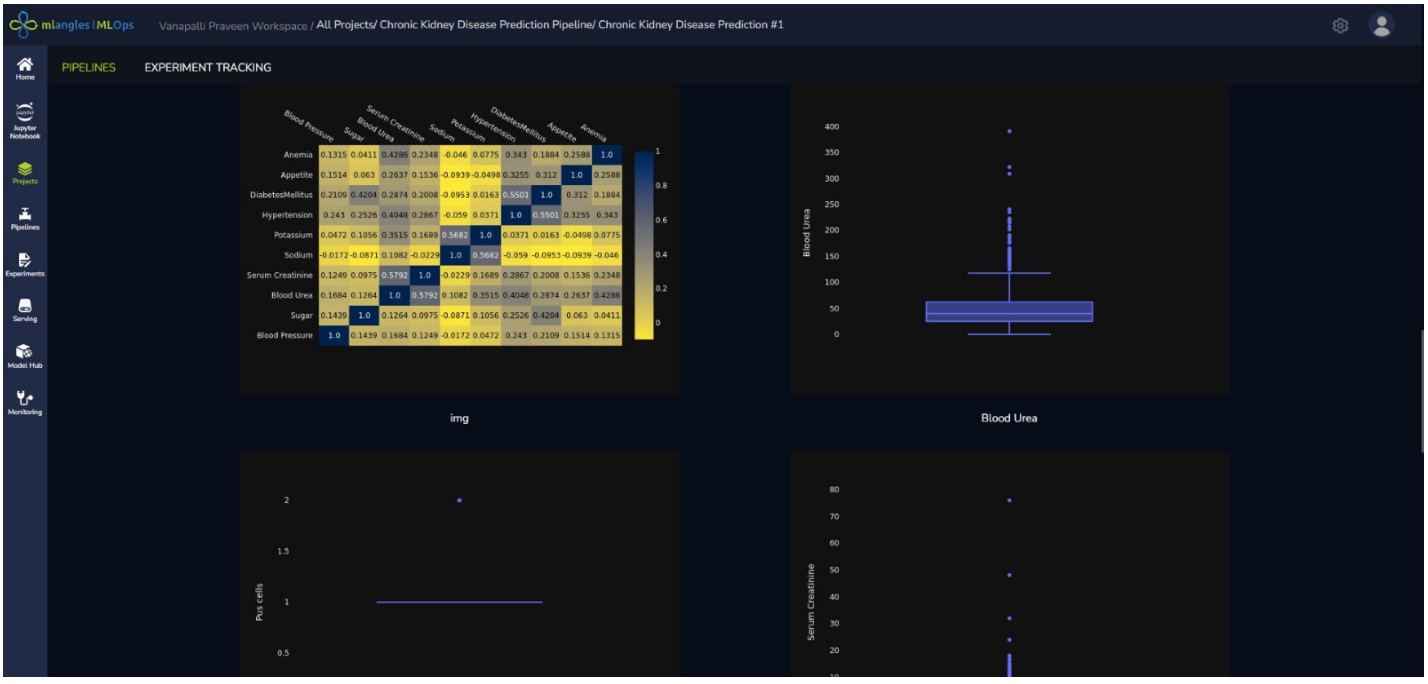
Data Analysis: Once the raw data is collected, the next step is to analyse it to gain insights and understand its characteristics. Data analysis involves examining the structure, patterns, and relationships within the data. This step helps in identifying trends, outliers, or any anomalies present in the dataset. Techniques such as descriptive statistics, correlation analysis, and data profiling are commonly used during this stage.

Data Preprocessing: After analyzing the data, it's common to encounter inconsistencies, missing values, or errors that need to be addressed. Data cleaning involves preprocessing the data to ensure its quality and reliability. This may include tasks such as inputting missing values, removing duplicates, standardizing formats, and handling outliers. The goal is to prepare the data for further analysis and modeling.

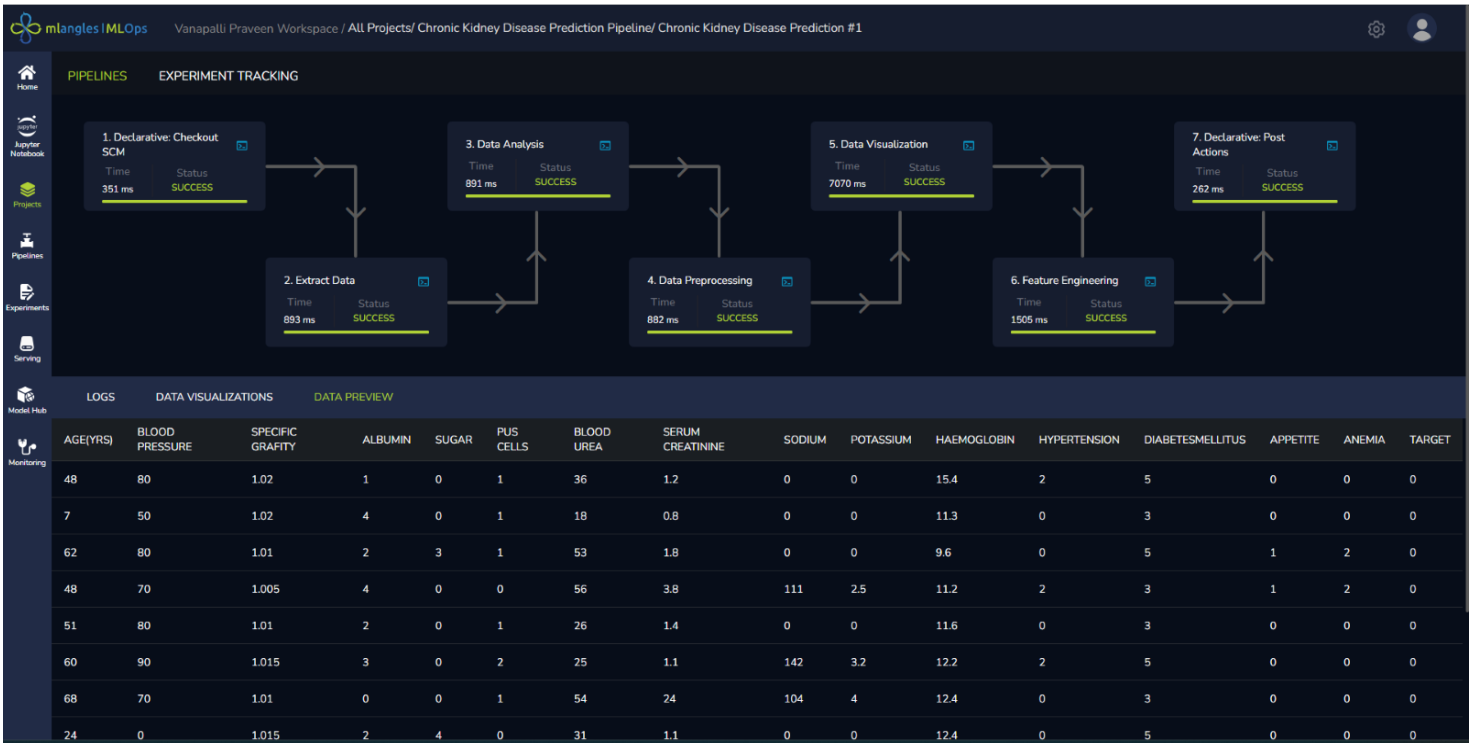
Data Visualization: Data visualization is a powerful tool for exploring and communicating insights from the data. Visualizations such as box plots, histograms, and heat maps are used to represent different aspects of the data distribution, relationships, and trends. Box plots are useful for visualizing the distribution of numerical data and detecting outliers. Histograms provide a graphical representation of the frequency distribution of continuous variables. Heat maps are effective for visualizing the correlation between variables in a tabular dataset.







Feature Engineering: This step aims to extract relevant information from the data and represent it in a format that is suitable for modelling. Feature engineering techniques include encoding categorical variables, scaling numerical features, creating interaction terms, and extracting domain-specific features. The goal is to enhance the predictive power of the model by providing it with informative and discriminative features.



AGE(YRS)	BLOOD PRESSURE	SPECIFIC GRAFITY	ALBUMIN	SUGAR	PUS CELLS	BLOOD UREA	SERUM CREATININE	SODIUM	POTASSIUM	HAEMOGLOBIN	HYPERTENSION	DIABETESMELLITUS	APPETITE	ANEMIA	TARGET
48	80	1.02	1	0	1	36	1.2	0	0	15.4	2	5	0	0	0
7	50	1.02	4	0	1	18	0.8	0	0	11.3	0	3	0	0	0
62	80	1.01	2	3	1	53	1.8	0	0	9.6	0	5	1	2	0
48	70	1.005	4	0	0	56	3.8	111	2.5	11.2	2	3	1	2	0
51	80	1.01	2	0	1	26	1.4	0	0	11.6	0	3	0	0	0
60	90	1.015	3	0	2	25	1.1	142	3.2	12.2	2	5	0	0	0
68	70	1.01	0	0	1	54	24	104	4	12.4	0	3	0	0	0
24	0	1.015	2	4	0	31	1.1	0	0	12.4	0	5	0	0	0

Step 2: Experiment Tracking - Modelling with Hyper-Parameter Optimizationation

After the data has been prepared and cleaned, the subsequent step involves training a model using this refined dataset. Since the problem at hand is a classification task, there are several models suitable for this purpose. Common options include the Random Forest Classifier, Decision Tree Classifier, Gradient Boosting Classifier, AdaBoost Classifier, Extra Trees Classifier.

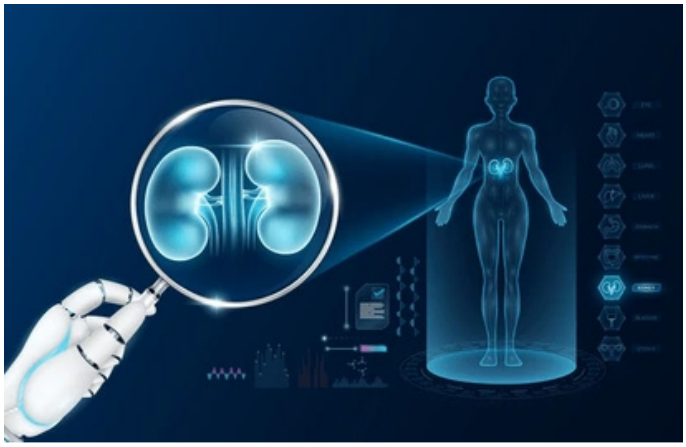
RandomForest Classifier: This ensemble learning technique constructs multiple decision trees during training and provides the mode of the classes (classification) or the mean prediction (regression) of the individual trees as output.

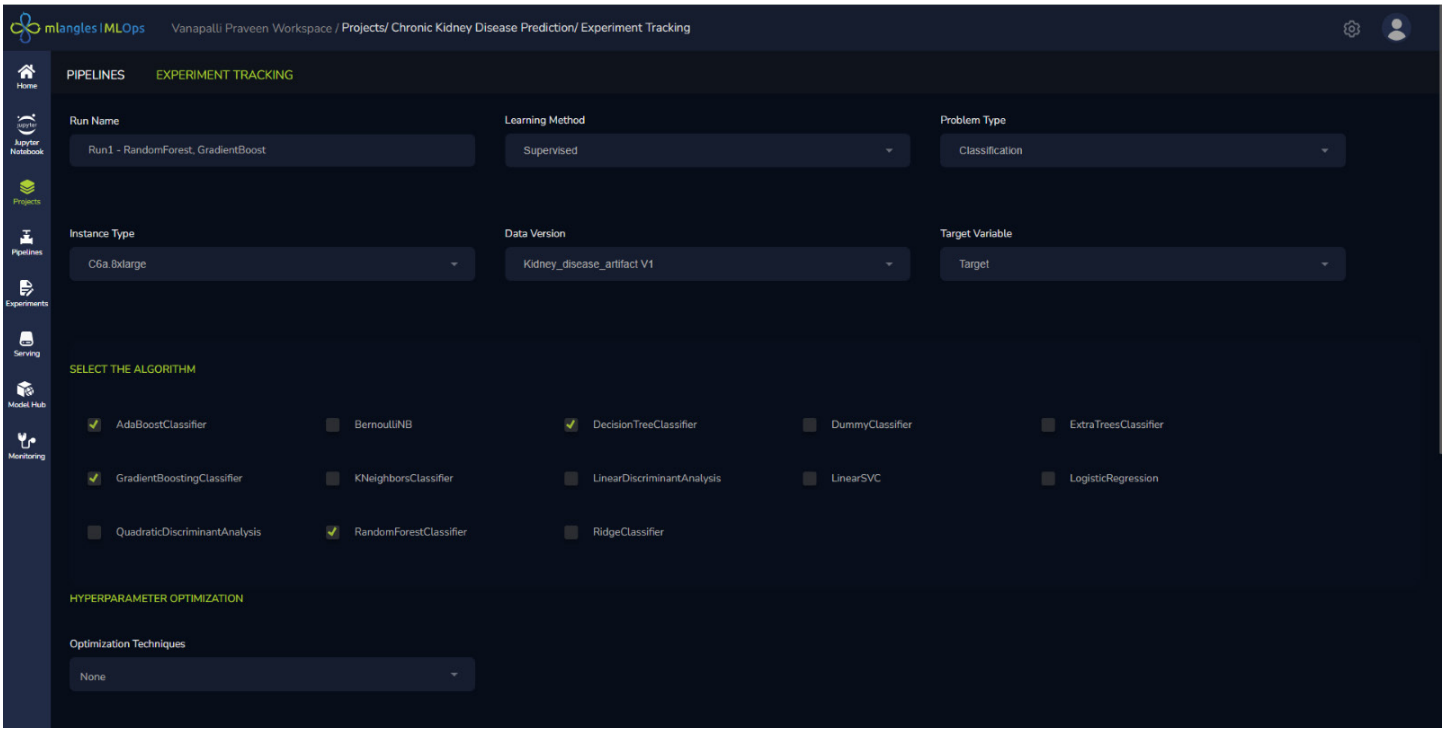
Decision Tree Classifier: Decision trees partition the data into subsets based on the values of features and make decisions at each node. They are straightforward yet effective for classification tasks.

Gradient Boosting Classifier: Sequentially builds a series of weak learners (usually decision trees) by correcting the errors of the previous models, producing a strong predictive model with high accuracy.

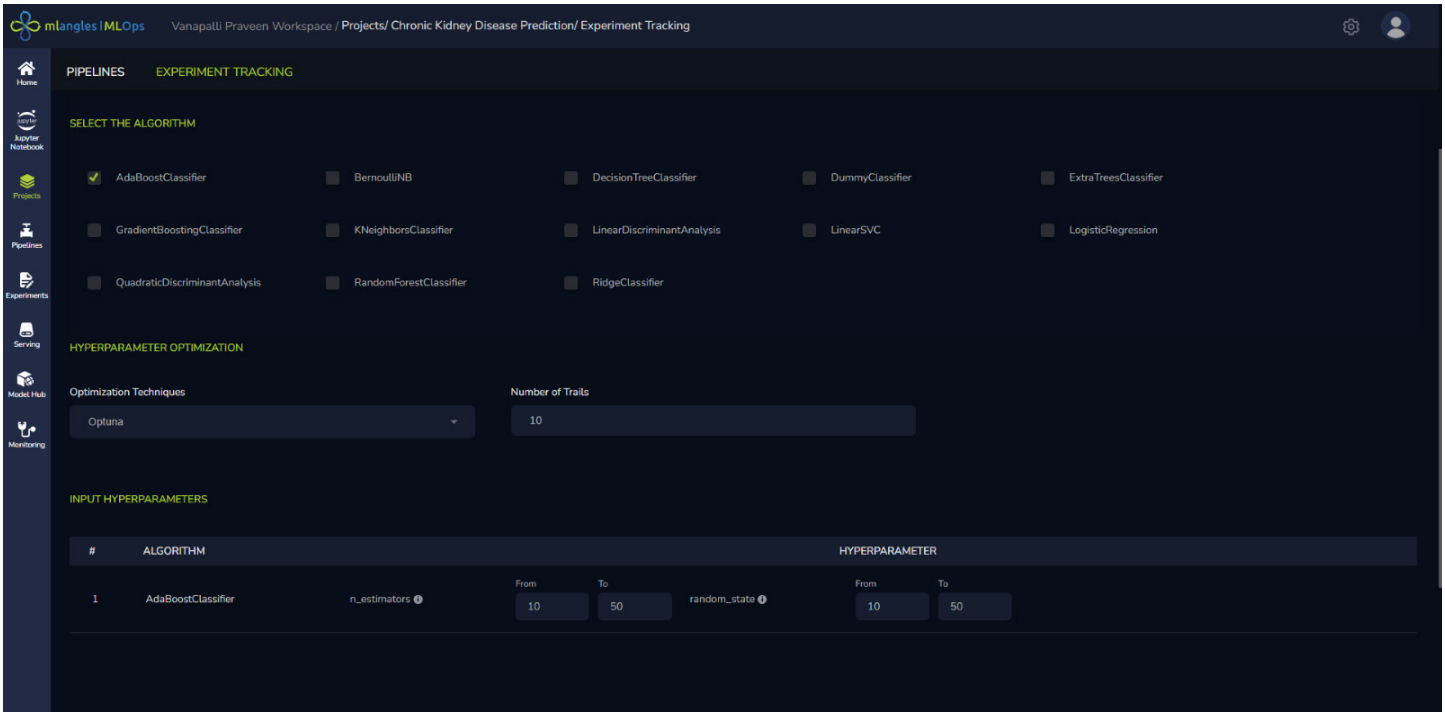
AdaBoost Classifier: Trains a series of weak learners sequentially, adjusting the weights of misclassified instances at each iteration, and combines their predictions to form a strong ensemble model with improved performance.

Extra Trees Classifier: Like Random Forests but introduces additional randomness by selecting random thresholds for feature splits, resulting in reduced overfitting and faster training times at the expense of slightly lower predictive performance.

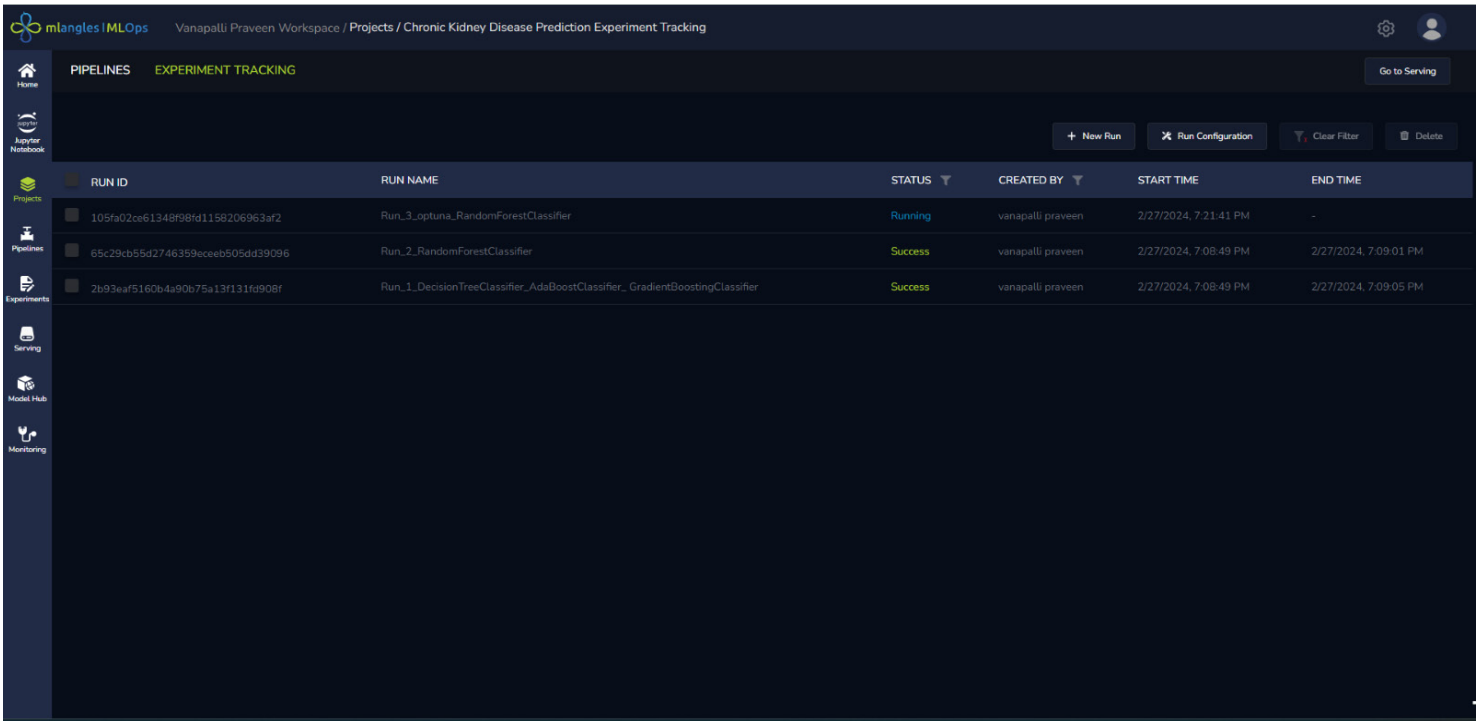




Additionally, **to enhance model performance, a hyperparameter optimization technique called Optuna is employed.** Optuna automates the process of tuning hyperparameters



After generating the run, a range of information can be extracted, such as visualizations of hyperparameters for the optimal algorithm, parameters employed during training, and metrics along with artifacts. These observations offer crucial insights into the model's performance and characteristics, facilitating comprehension of its efficacy and areas where enhancements can be made.



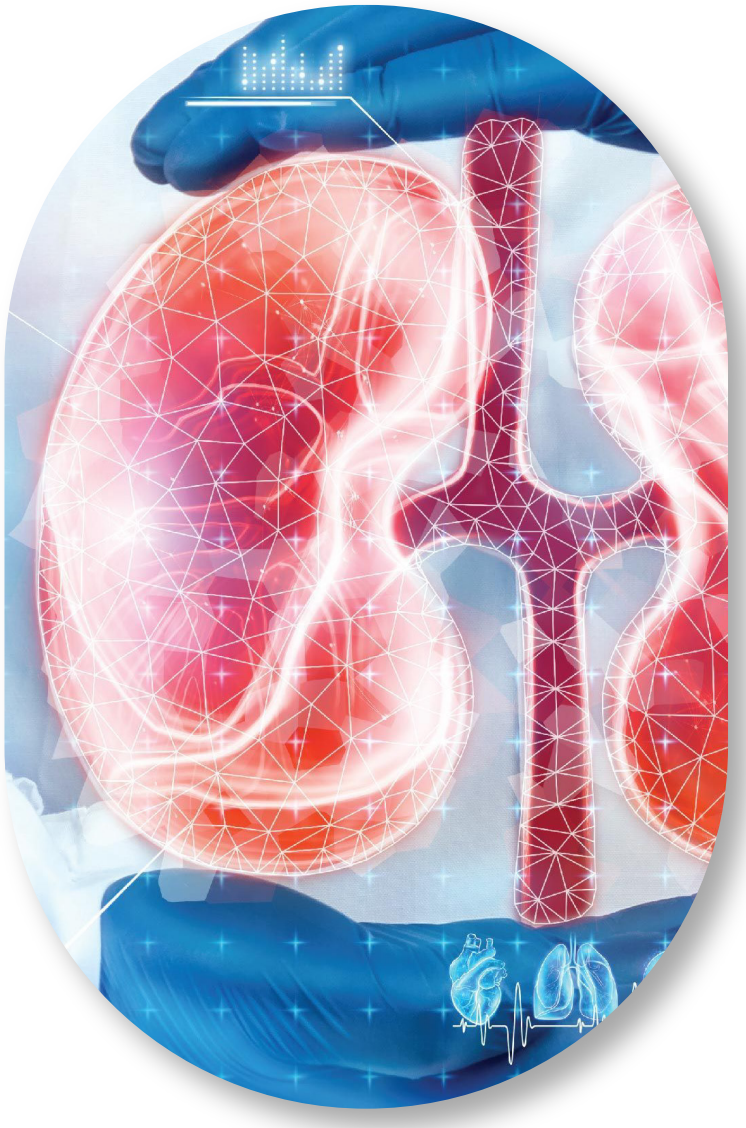
Visual representations of hyperparameters illustrate the influence of various parameter settings on model performance, aiding in the identification of optimal configurations.

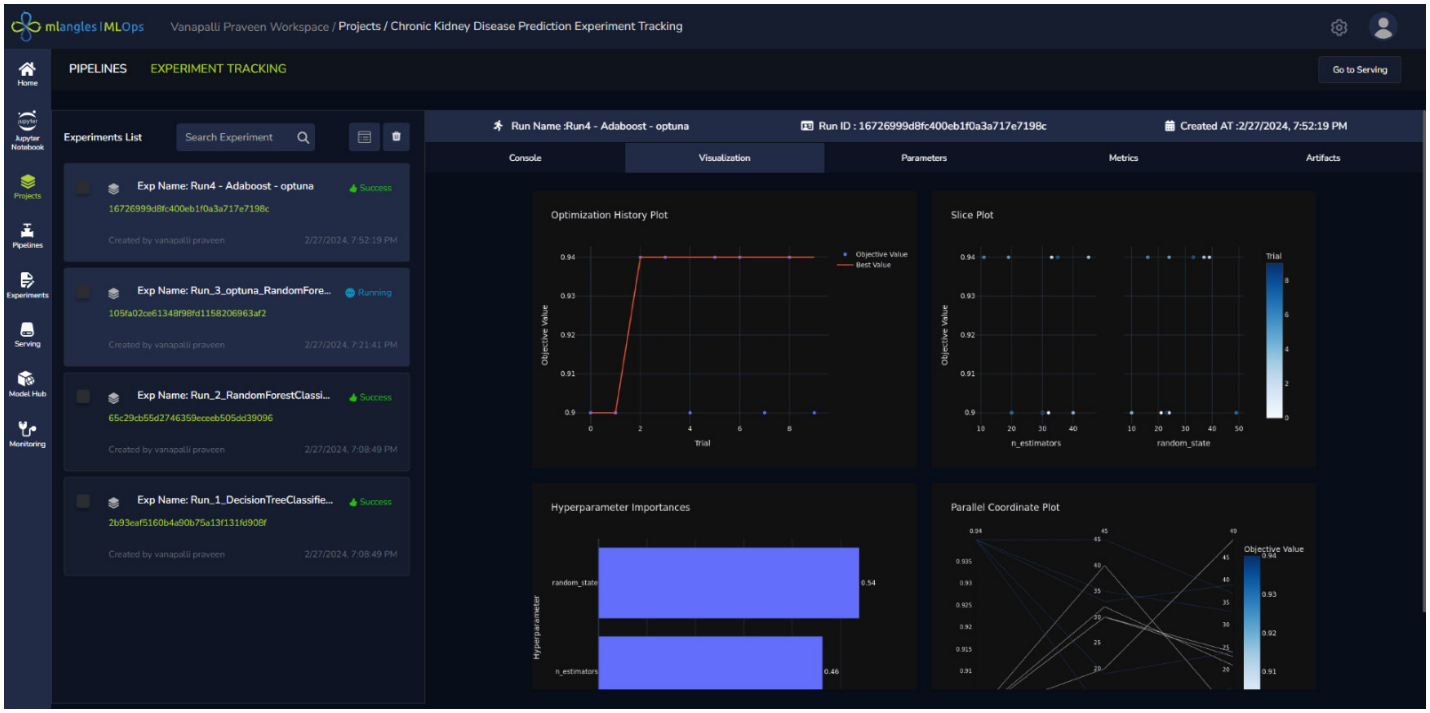
Optimization History Plot: The Optimization History Plot illustrates the evolution of the objective function (e.g., accuracy or loss) throughout the hyperparameter search iterations, providing insights into convergence patterns and the efficacy of the optimization algorithm.

Slice Plot: A Slice Plot depicts the correlation between two hyperparameters while keeping the values of other hyperparameters constant. This visualization enables the exploration of interactions among hyperparameters and their impact on model performance, aiding in the discovery of optimal parameter combinations.

Hyperparameter Importances Plot: The Hyperparameter Importances Plot ranks the significance of hyperparameters according to their impact on model performance. This visualization assists in identifying the most influential hyperparameters, informing subsequent optimization endeavors or strategies for feature selection.

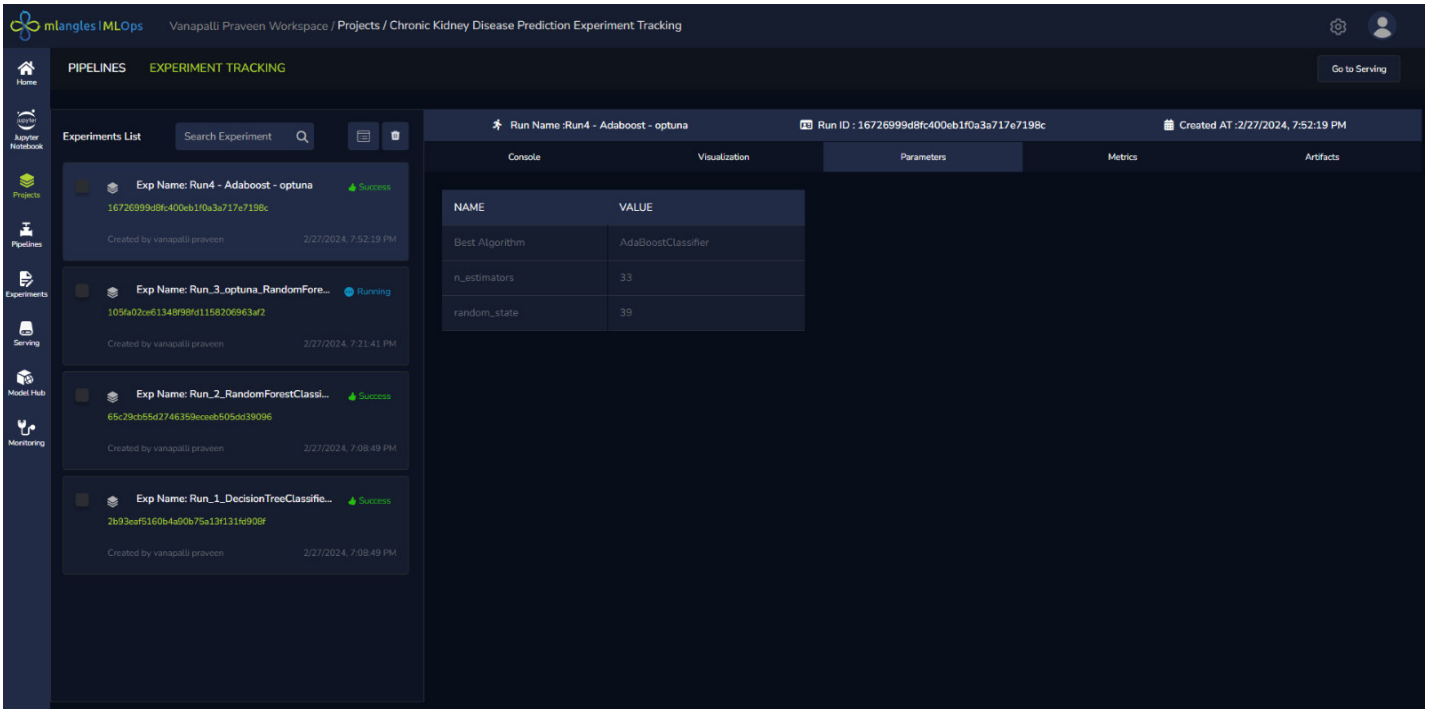
Parallel Coordinate Plot: The Parallel Coordinate Plot represents high-dimensional hyperparameter spaces by depicting each hyperparameter as a vertical axis and each point in the plot as a hyperparameter configuration. Lines connecting points illustrate.



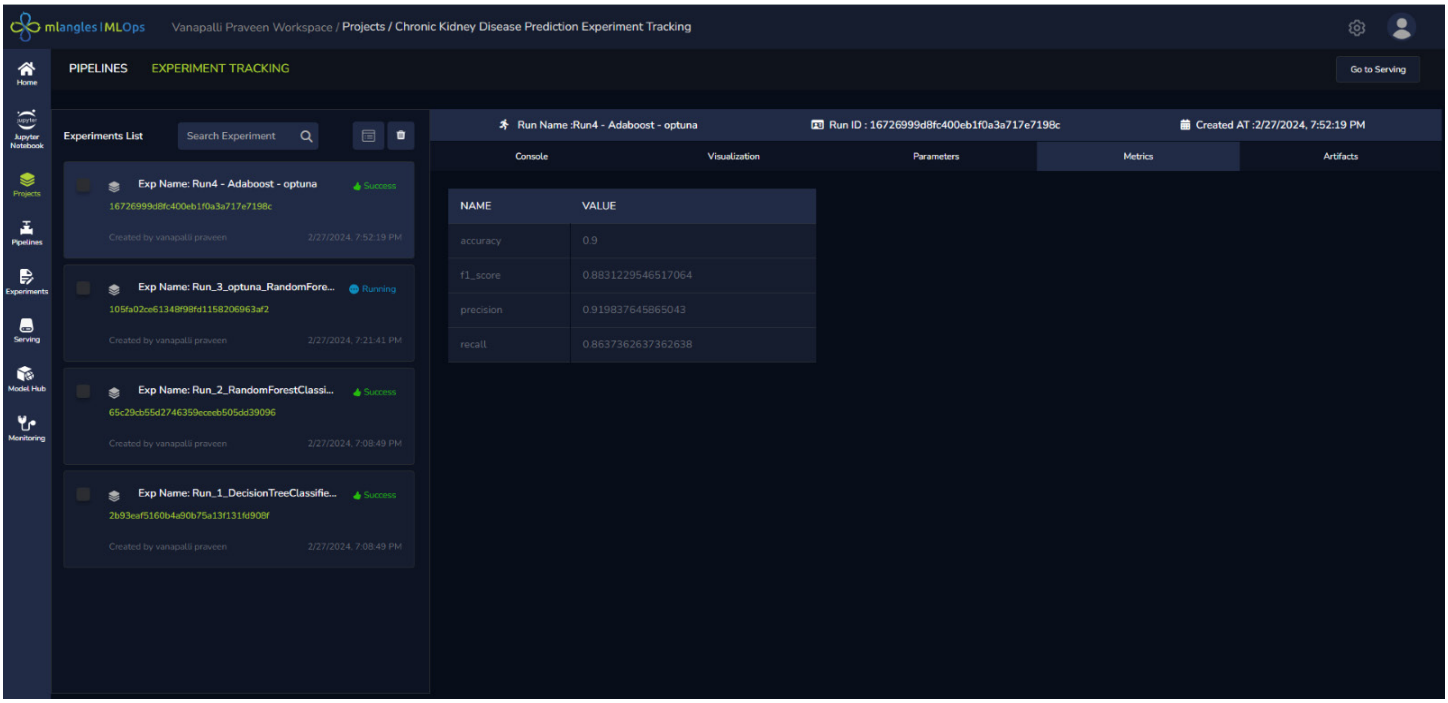


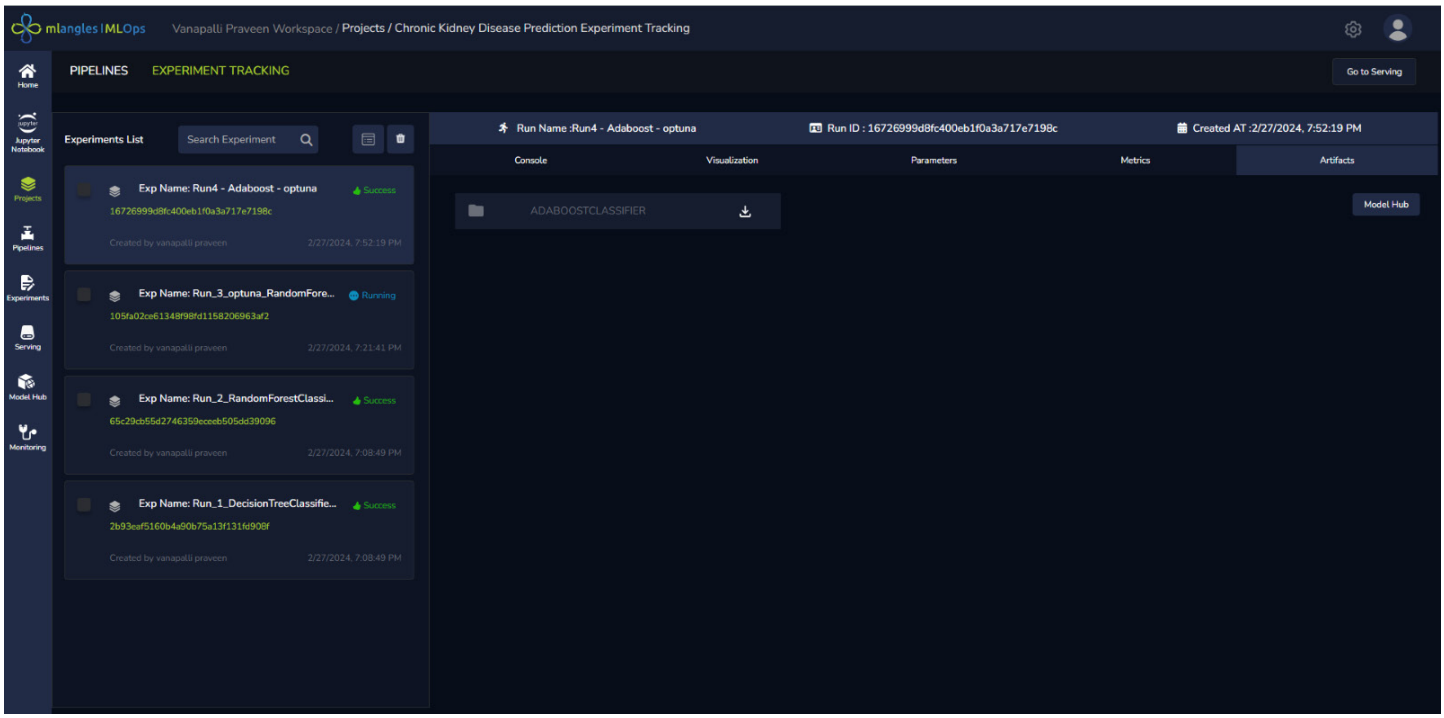
hyperparameter configurations, facilitating the examination of relationships and patterns across multiple hyperparameters simultaneously.

Furthermore, extracting parameters utilized during training enhances reproducibility and transparency, guaranteeing that the model's configurations are documented and readily available for future reference.



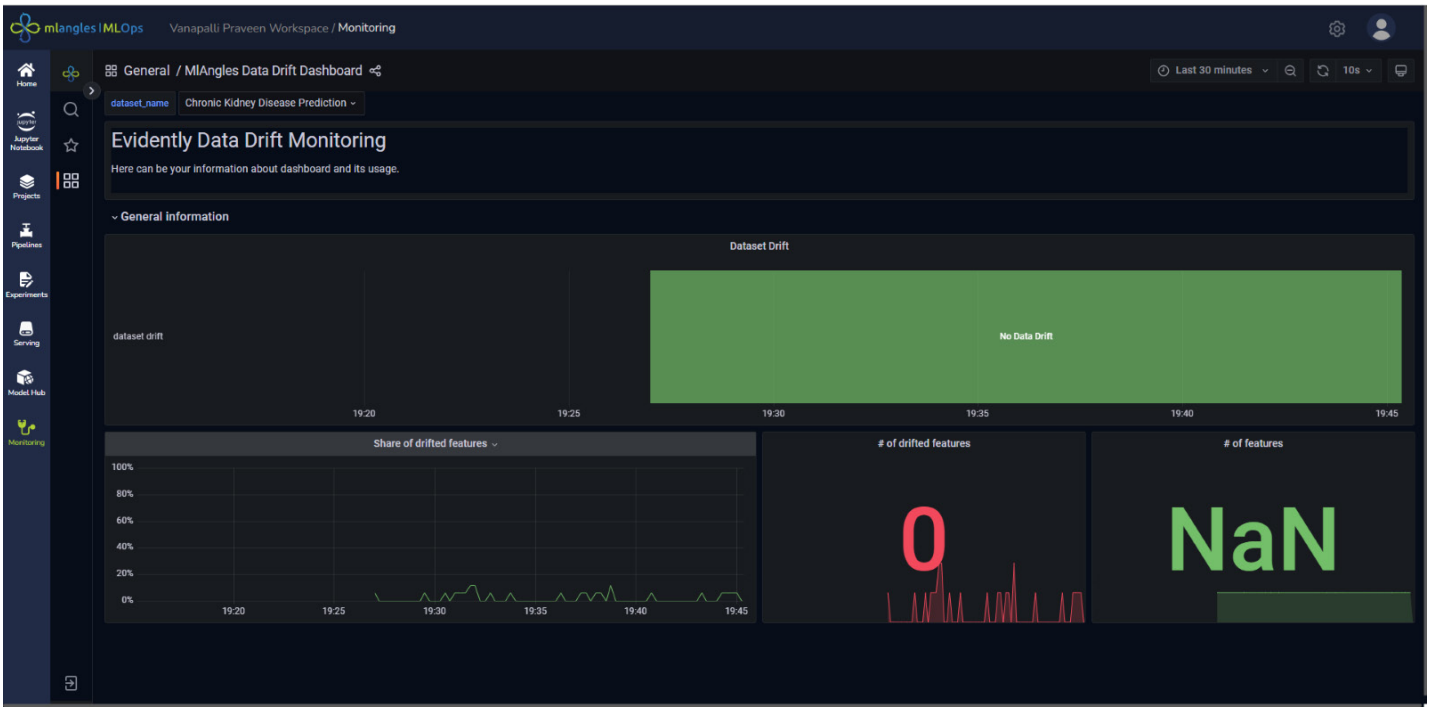
Metrics such as accuracy offer insights into the model's performance. Accuracy measures overall correctness, precision assesses the accuracy of positive predictions, recall emphasizes capturing all positive instances, and the F1 score balances precision and recall. These metrics collectively aid in evaluating the effectiveness and robustness of classification models.





Monitoring :

Monitoring for data drift in CKD involves regularly assessing the distribution of patient data over time and updating machine learning models accordingly. This ensures that the models remain accurate and reliable in predicting CKD progression, guiding treatment decisions, and improving patient outcomes. Effective monitoring and adaptation strategies are essential for maintaining the relevance and effectiveness of machine learning applications in the management of chronic kidney disease.



Conclusion

This project underscores the effectiveness of classification algorithms in medical data analysis, showcasing their capability for accurate and efficient disease prediction based on diverse health parameters. Our findings emphasize the significance of integrating advanced machine learning techniques into healthcare practices, facilitating improved diagnosis and treatment planning and ultimately leading to better patient outcomes in healthcare.

To setup Demo

Info.mlangles@cloudangles.com

Visit: www.mlangles.ai