

# ML PROJECT

Presented By:

Group-03 of A1

Consisting IDs:

2018007, 2018008, 2018009

Date: 4th March of 2024



# MEET THE GROUP :

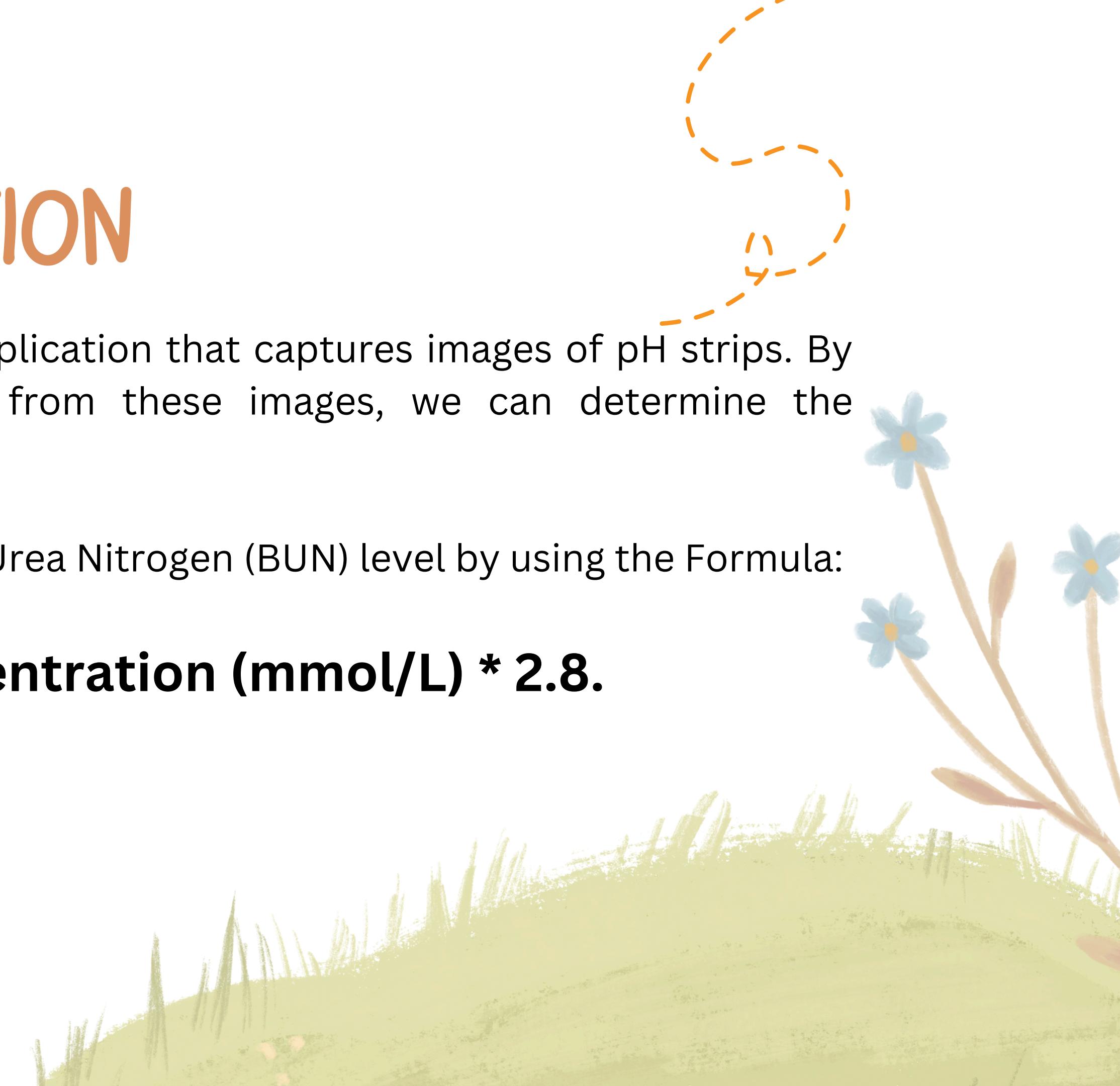
- 
- Mst. Tasnim Fariha Khanom
  - Nafish Ahanaf
  - Rudmila Nizam

# PROBLEM DEFINITION

We detect urine pH using a web application that captures images of pH strips. By analyzing the pH level obtained from these images, we can determine the concentration of urea in the urine.

Finally, we can calculate the Blood Urea Nitrogen (BUN) level by using the Formula:

**BUN (mg/dL)= Urea concentration (mmol/L) \* 2.8.**



# PROJECT GOALS



To optimize the utilization of the provided dataset



Create a web app using the dataset



Utilize the web app to assist patients in easily detecting  
CKD (Chronic Kidney Disease).

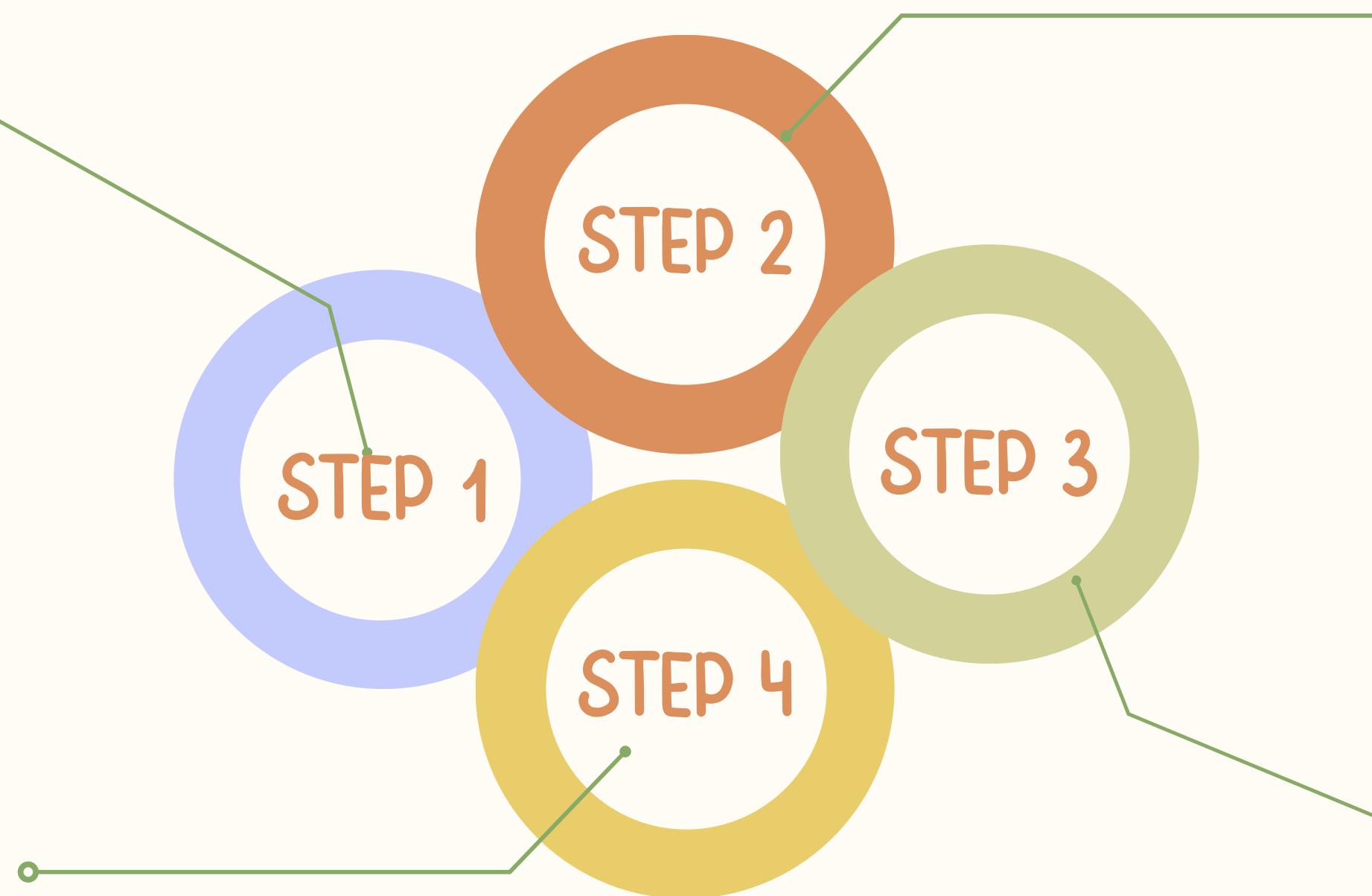


# LITERATURE REVIEW

## "MACHINE LEARNING-BASED pH COLOR RECOGNITION FOR MONITORING CHRONIC KIDNEY DISEASE" EXPLORES

- Predicting pH values using RGB profile data obtained from pH test strips.
- The motivation behind this approach is to offer a less burdensome and invasive method for monitoring chronic kidney disease compared to traditional methods.
- This study faces challenges due to limited availability of datasets, including the presence of outliers and unclear boundaries between pH RGB profiles.

# PROCEDURE



- Imported pH dataset mapped to RGB values
- Conducted exploratory data analysis (EDA) and performed train-test split.

Developed a web app integrating all components for user-friendly access.

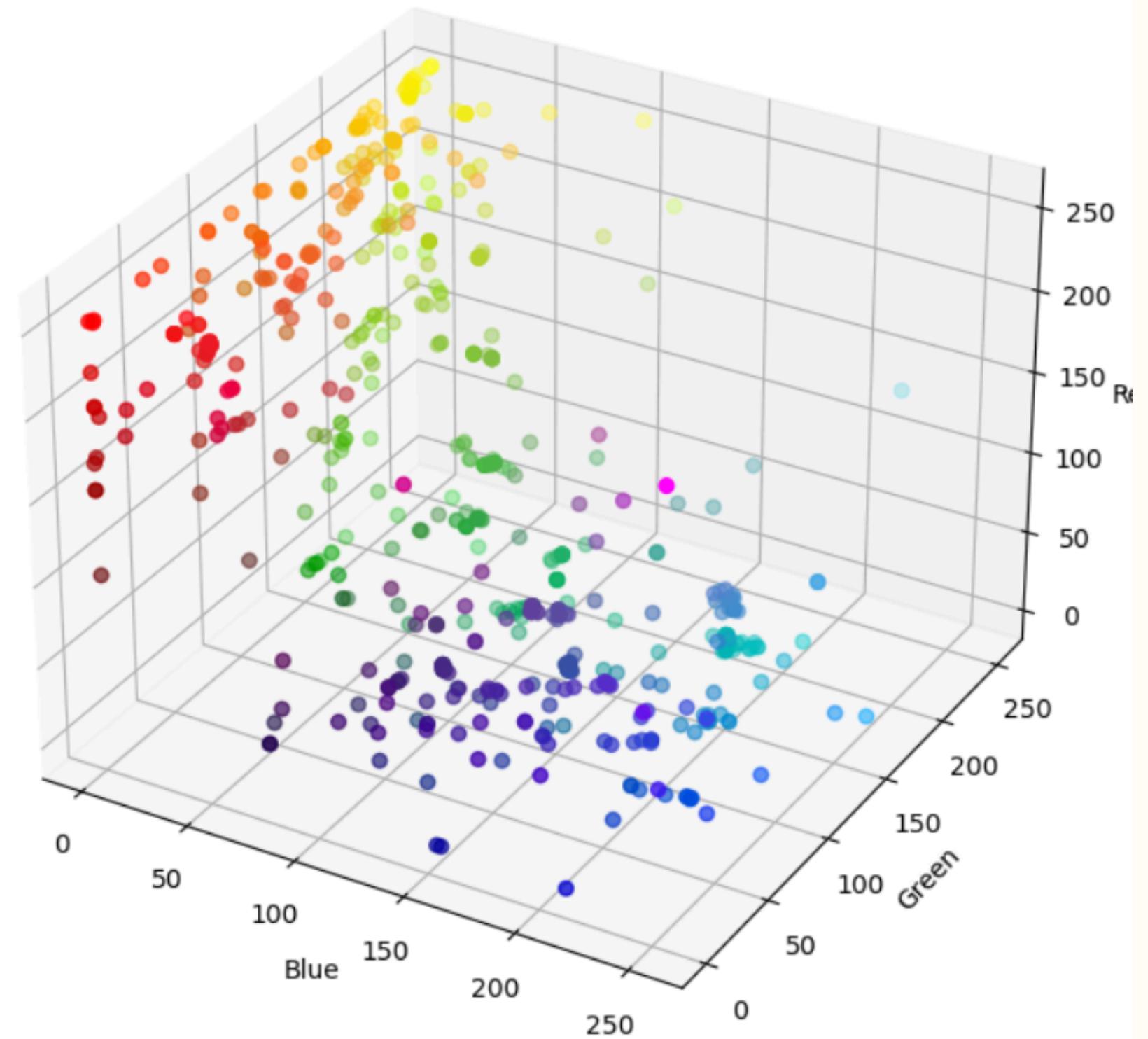
- Utilized KNN approach for pH prediction.
- Experimented with different kernels for SVM model.
- Implemented Neural Network (NN) for pH prediction.

Performed image processing using OpenCV, averaging RGB values.

blue	green	red	label
36	27	231	0
36	84	250	1
37	164	255	2
22	205	255	3
38	223	221	4
29	214	148	5
0	181	76	6
13	156	0	7
92	166	0	8
184	191	0	9
200	136	1	10
198	76	0	11
180	38	51	12
181	22	72	13
140	17	57	14
38	27	237	0
50	100	244	1
30	143	247	2
36	195	255	3
0	242	254	4
65	195	132	5
73	183	77	6

Table of dataset

Color distribution



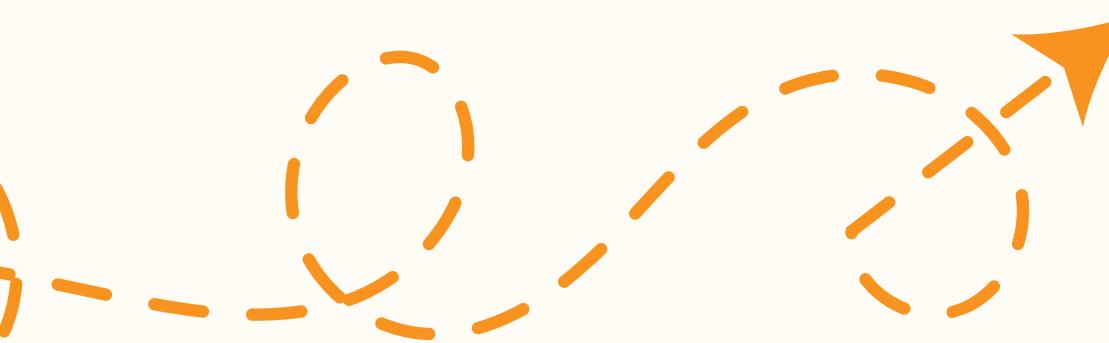
3D view of our dataset

# Methodology

The paper we followed evaluated **KNN**, **SVM**, and **NN** models on the dataset.

- For KNN, four test cases are conducted with varying configurations, including different **numbers of neighbors** and **data normalization** methods.
- SVM models are evaluated using different kernels with **rbf** or **poly** kernels.
- For NN models, three test cases are conducted, varying the **number of hidden layers** and **activation functions**.

We have evaluated the dataset using KNN, SVM, and NN models, as described in the paper. Additionally, we attempted to improve accuracy by employing a **Random Forest model**.



# PROJECT TIMELINE

## WEEK 1

Explored the datasets and thoroughly reviewed the paper.

## WEEK 2

Gathered information about other datasets connecting to this paper.

## WEEK 3

Initiated data processing and model training

## WEEK 4

Improved the model and attempted to enhance its accuracy.

## WEEK 5

Developed the web app and conducted beta testing using web-based images.

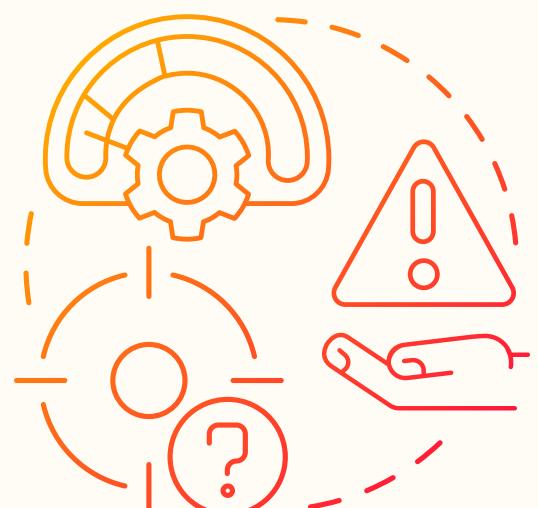
# COMPARISON OF RESULTS

In the paper, the **KNN** model achieved an accuracy of about 72% in the first iteration, and it increased to 83% in the third iteration. For our model, the initial KNN accuracy was 71.75%. After adjusting the test-train percentage, the accuracy increased to **82%**.

The **SVC** model's accuracy remained the same as in the paper, at **69.4%**. However, in the SVC model, the accuracy became saturated after certain hyperparameter tuning.

Additionally, the accuracy of the **NN** model decreased with increasing epochs.

	Score	Parameters
LogisticRegression	0.807603	{'fit_intercept': True, 'penalty': 'l2', 'solv...
KNeighborsClassifier	0.909655	{'algorithm': 'auto'}
RandomForestClassifier	0.948808	{'max_depth': 15, 'n_estimators': 7}
GradientBoostingClassifier	0.910748	{'max_depth': 3, 'n_estimators': 6}



# Confusion Matrix

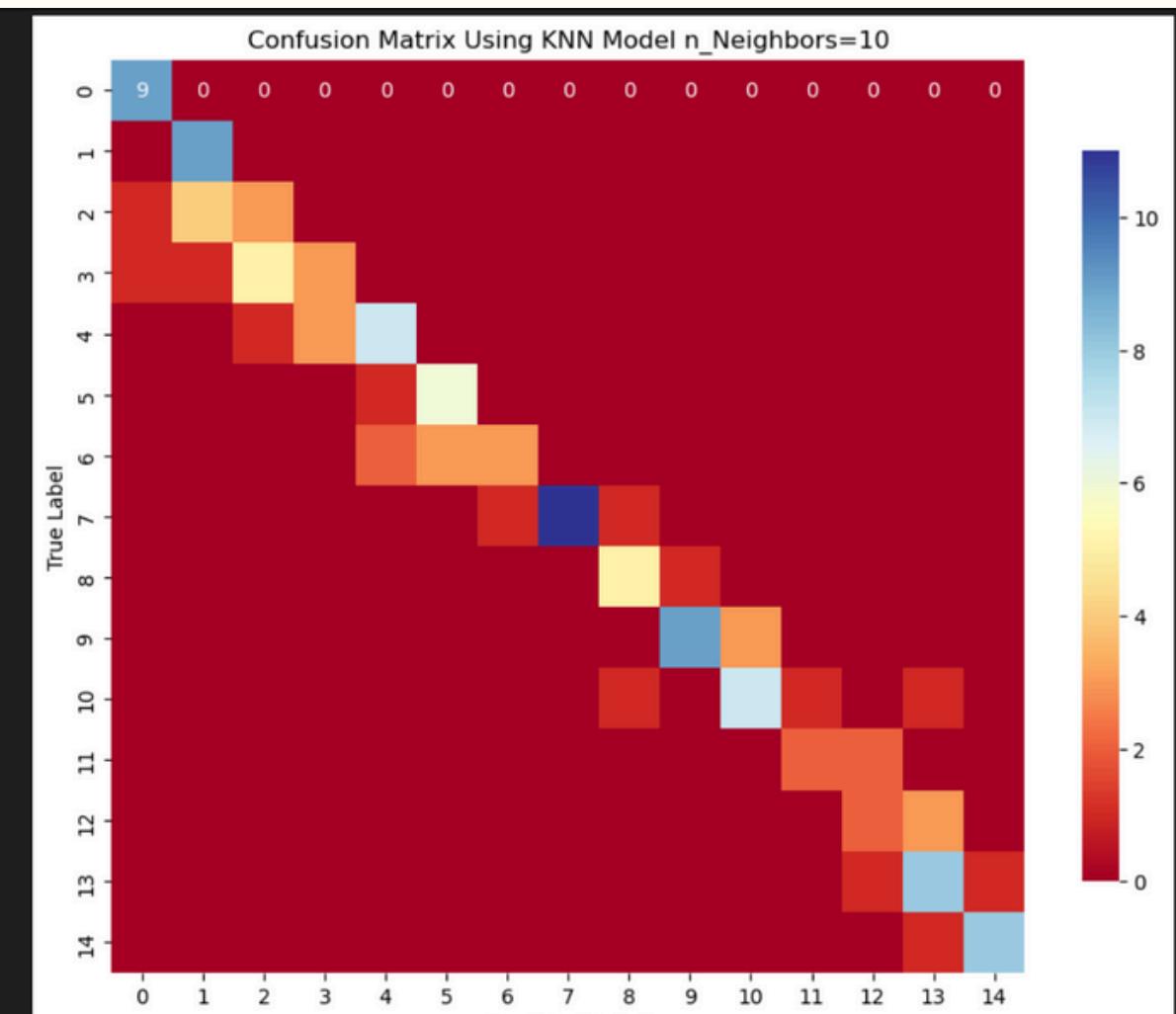


Fig:Confusion Matrix of KNN  
(n\_neighbors=10)

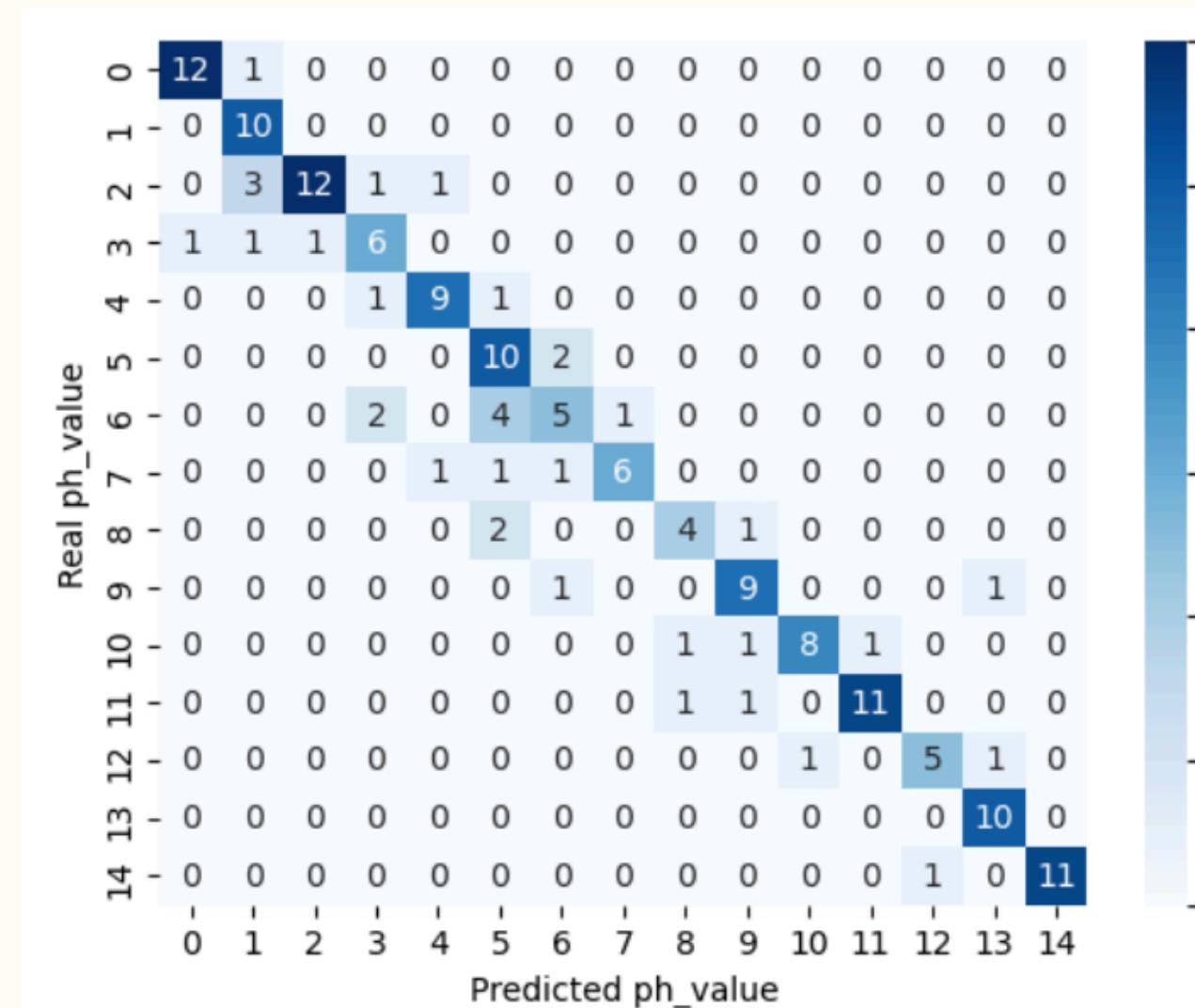


Fig:Confusion Matrix of RF  
(n\_estimators=16, max\_depth=22)

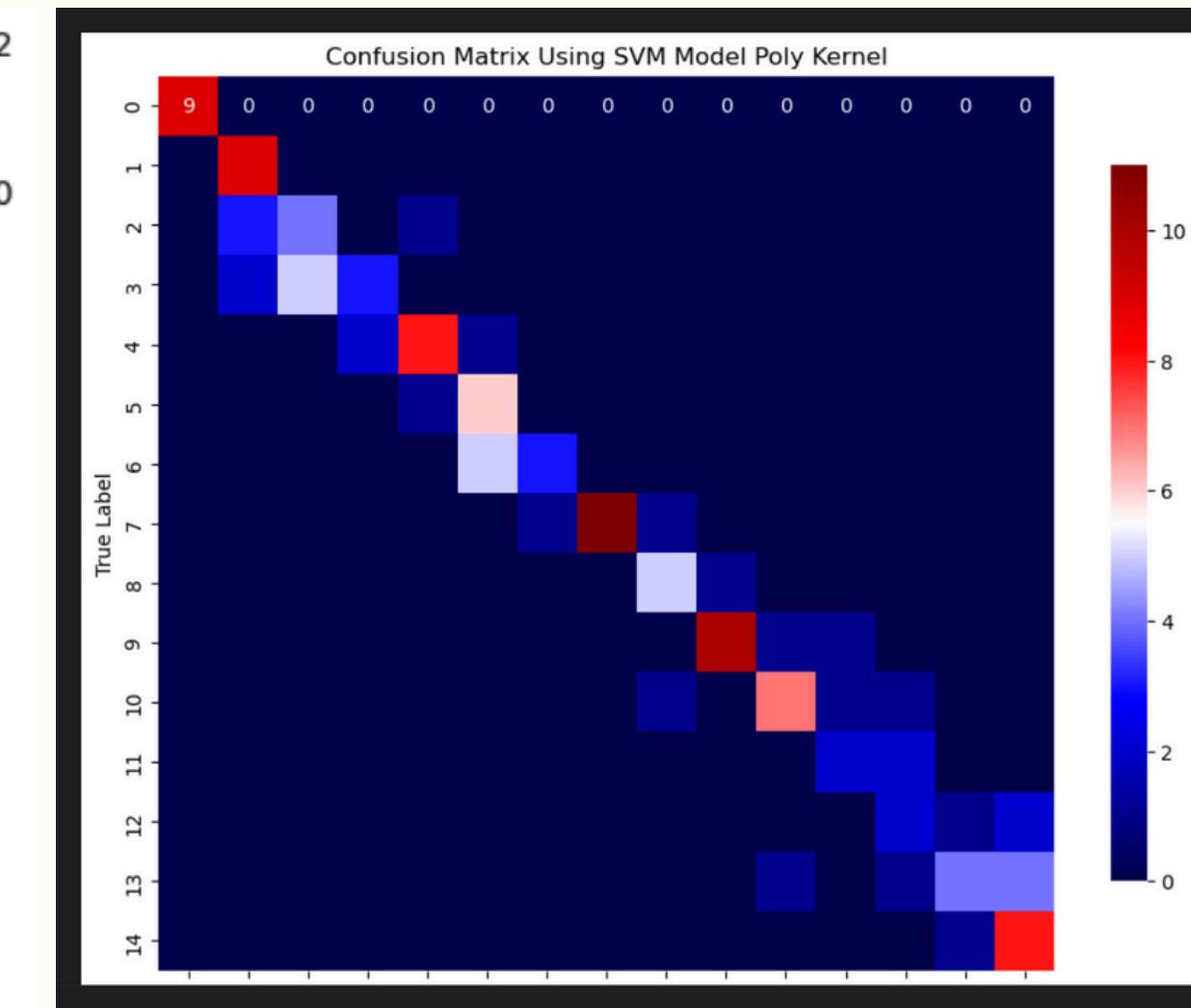
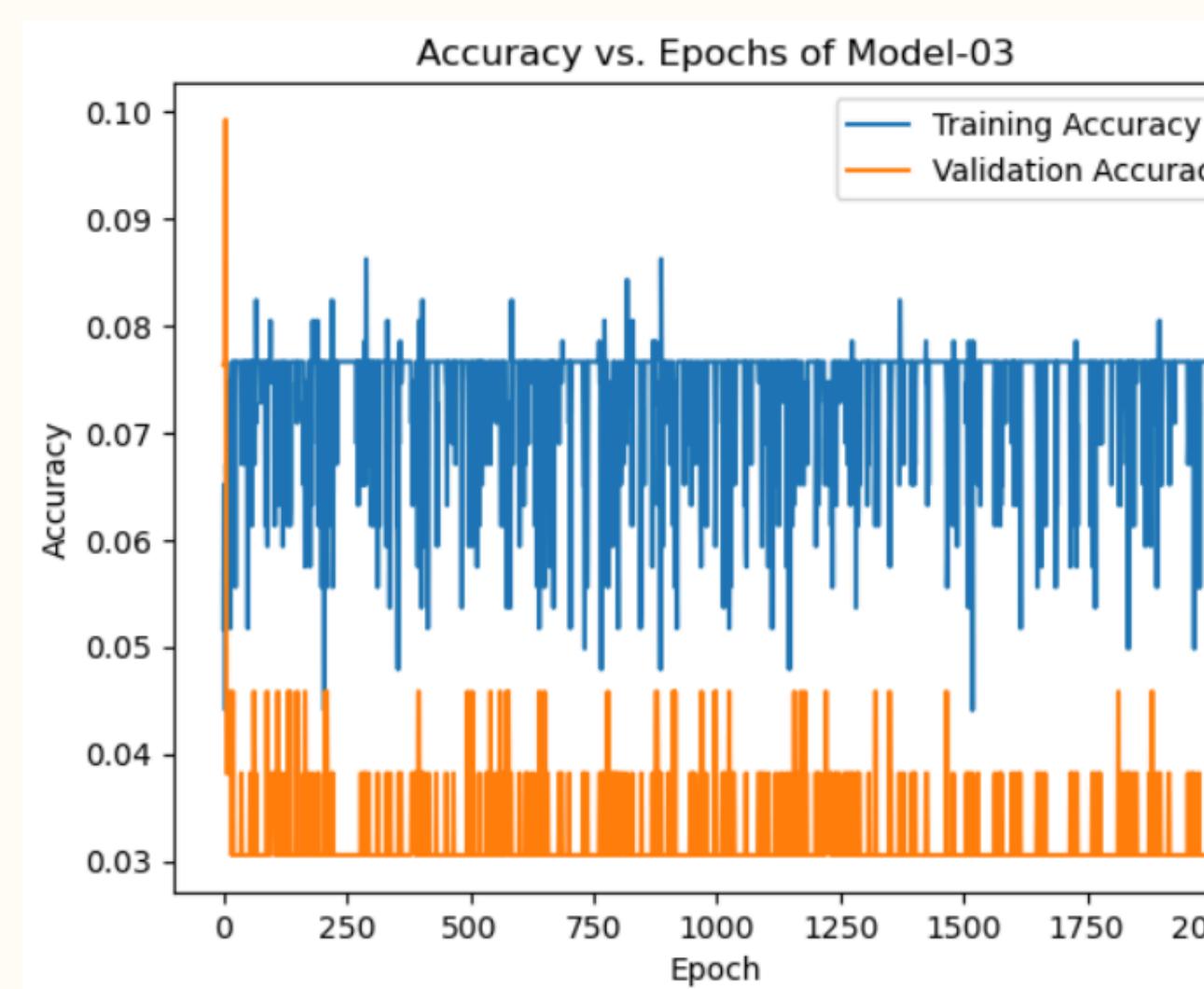
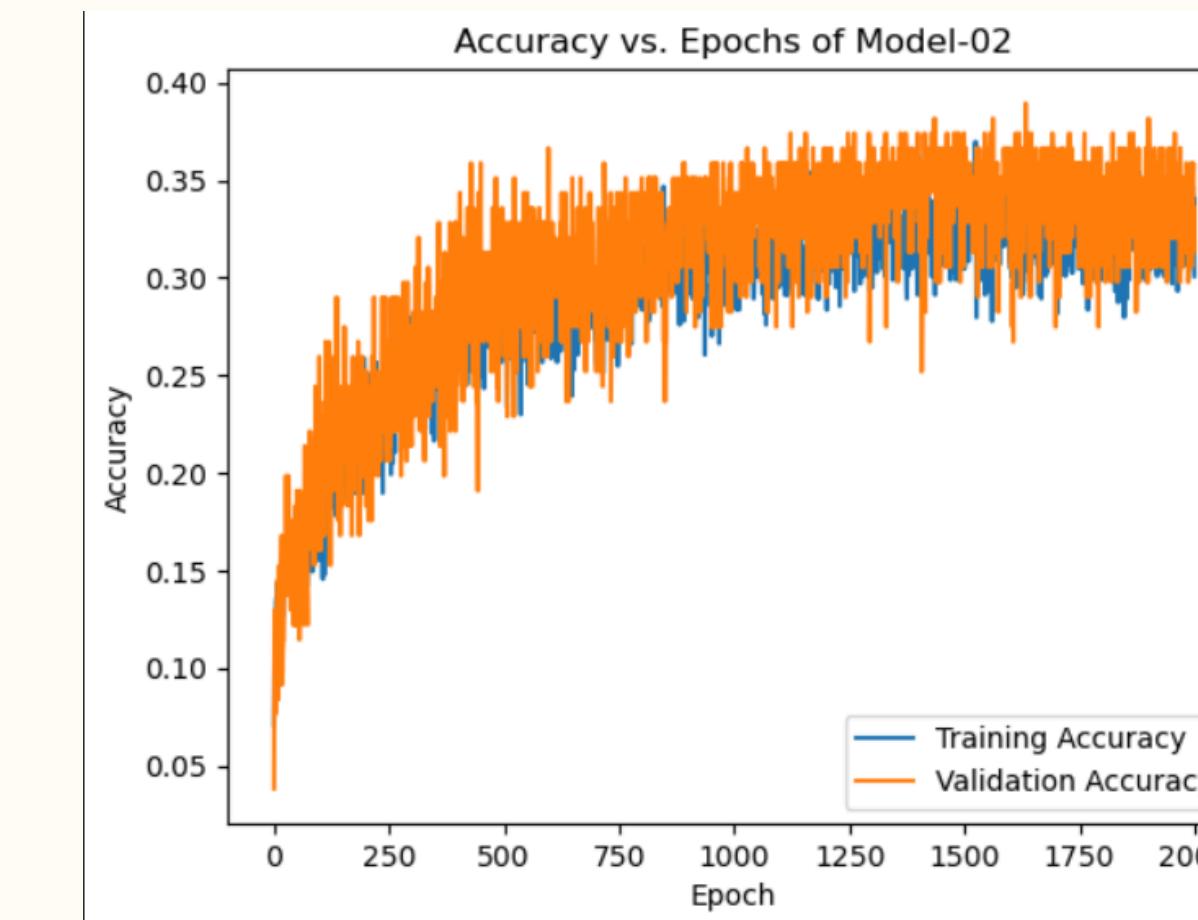
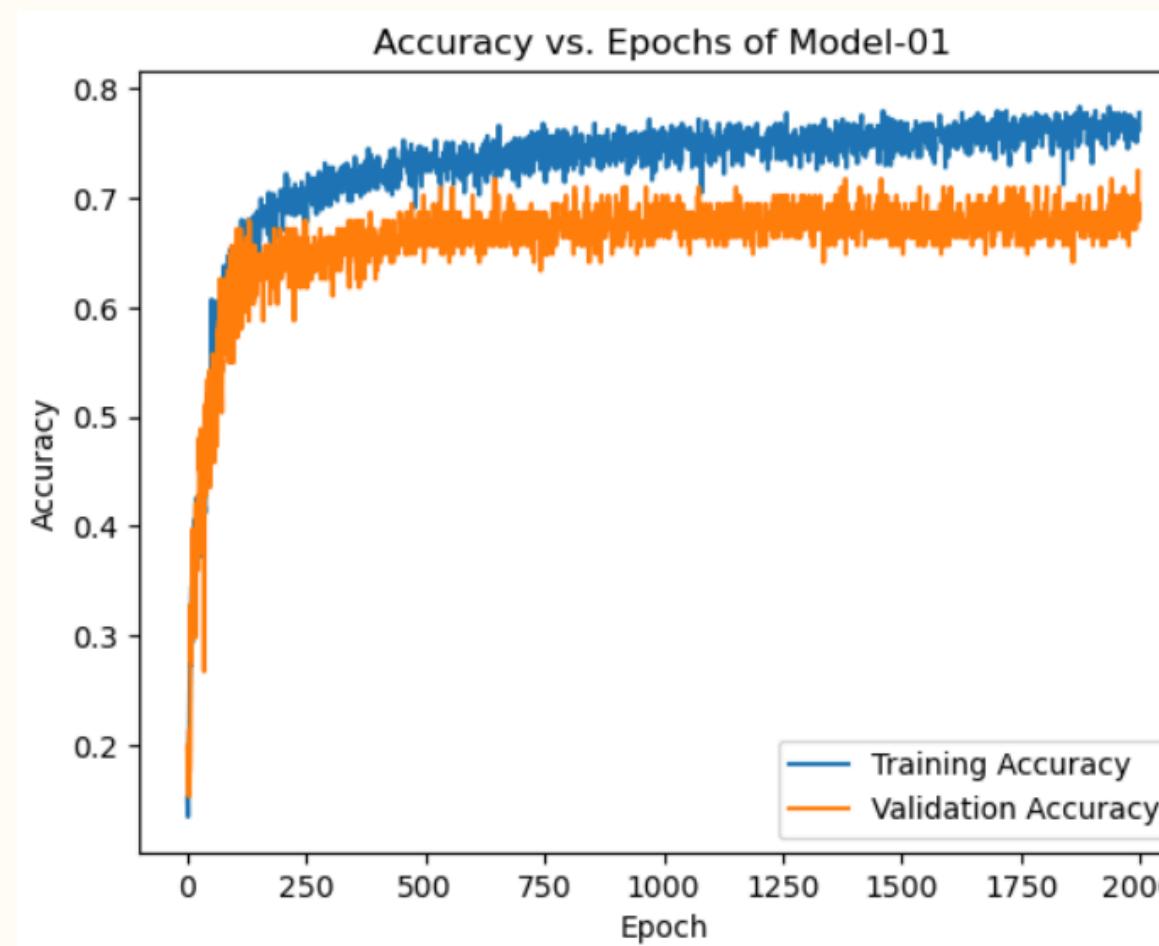


Fig:Confusion Matrix of SVM  
(Poly Kernel)



# Demonstration

**Urine pH Detector**

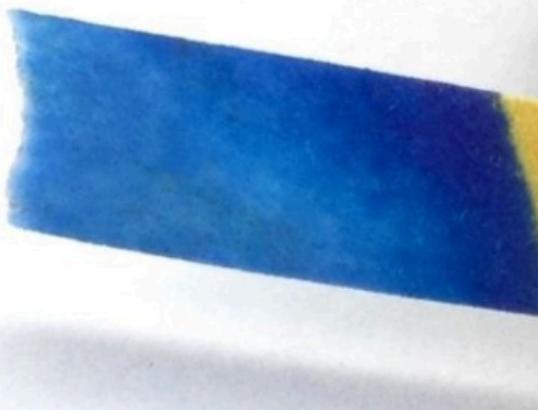
image 

Model Choice  
 KNN  SVM

Clear

Submit

Use via API  · Built with Gradio 

image 

output

Model: KNN  
Predicted pH: 8  
Acidity/Base Level: Moderately Alkaline

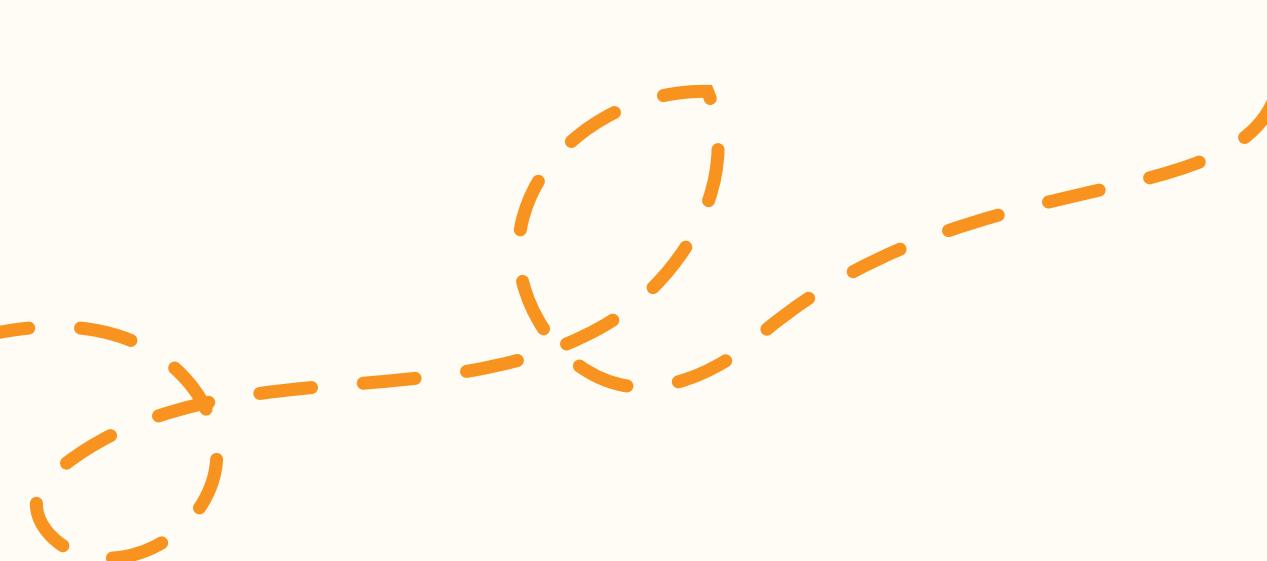
Flag

Clear

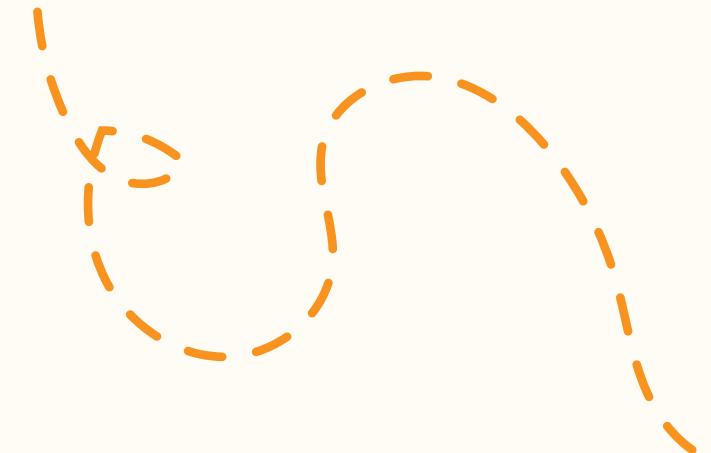
output

Model: SVM  
Predicted pH: 8  
Acidity/Base Level: Moderately Alkaline

Flag



**DOES ANYONE HAVE A  
QUESTION ?**



# THANK YOU

