

SPRINT 2

Project Deliverables (Model Building Code & Evaluation)

Team ID	PNT2022TMID50914
Project Name	Efficient Water Quality Analysis & Prediction using Machine Learning

LOGISTIC REGRESSION EVALUATION:

IBM-Project-12969-16595 x IBM-Project-34997-16602 x IBM-Project-18872-16596 x Water Quality x IBM x Project Deliverables Subm x notebook078b9b7ff6 | Kag x water-quality-prediction-7 x

colab.research.google.com/drive/1c7IKyTqseuOfVt19gM0tt84Tlaazcw4#scrollTo=567b7f03

Gmail YouTube Maps perfect Plan B # 20... My Courses | Perfec... Python for Everybo... Home | Codewars Certifications | InfyTQ AICTE Internship En... Online Courses - Le... Striver's SDE Sheet... TCS NQT 2023 Batc... Last 20 days of TCS... Bubble Sort Algorit...

water-quality-prediction-7-model.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

pred_lg = model_lg.predict(x_test)

```
[ ] # Calculating Accuracy Score
lg = accuracy_score(y_test, pred_lg)
print(lg)
```

0.6284658040665434

```
[ ] print(classification_report(y_test, pred_lg))
```

	precision	recall	f1-score	support
0	0.63	1.00	0.77	680
1	0.00	0.00	0.00	402
accuracy			0.63	1082
macro avg	0.31	0.50	0.39	1082
weighted avg	0.39	0.63	0.49	1082

```
[ ] # confusion Matrix
cm1 = confusion_matrix(y_test, pred_lg)
sns.heatmap(cm1/np.sum(cm1), annot = True, fmt= '0.2%', cmap = 'Reds')
```

<AxesSubplot:>

	0	1
0	62.85%	0.00%
1	0.00%	0.00%

Activate Windows
Go to Settings to activate Windows.

Type here to search

04:19 PM
11-11-2022

DECISION TREE EVALUATION:

water-quality-prediction-7-model.ipynb

```
[ ] # Calculating accuracy score
dt = accuracy_score(y_test, pred_dt)
print(dt)

0.6451016635859519

[ ] print(classification_report(y_test, pred_dt))
```

	precision	recall	f1-score	support
0	0.66	0.90	0.76	680
1	0.56	0.22	0.32	402
accuracy			0.65	1082
macro avg	0.61	0.56	0.54	1082
weighted avg	0.62	0.65	0.60	1082

```
[ ] # confusion Maxtrix
cm2 = confusion_matrix(y_test, pred_dt)
sns.heatmap(cm2/np.sum(cm2), annot = True, fmt= '0.2%', cmap = 'Reds')
```

<AxesSubplot:>

	Actual 0	Actual 1
Predicted 0	56.28%	6.56%
Predicted 1	28.93%	8.23%

Using Random Forest

Activate Windows
Go to Settings to activate Windows.

Type here to search

04:20 PM
11-11-2022

RANDOM FOREST EVALUATION:

IBM-Project-12969-16595 x IBM-Project-34997-16602 x IBM-Project-18872-16596 x Water Quality x IBM x Project Deliverables Subm x notebook078b9b7ff6 | Kag x water-quality-prediction-7 x

colab.research.google.com/drive/1c7IKyTqseuOfVt19gM0tt84Tlaazcw4#scrollTo=967ca8e9

Gmail YouTube Maps perfect Plan B # 20... My Courses | Perfec... Python for Everybo... Home | Codewars Certifications | InfyTQ AICTE Internship En... Online Courses - Le... Striver's SDE Sheet... TCS NQT 2023 Batc... Last 20 days of TCS... Bubble Sort Algorit...

water-quality-prediction-7-model.ipynb ☆

File Edit View Insert Runtime Tools Help [All changes saved](#)

+ Code + Text

Connect Editing

```
[ ] # Calculating Accuracy Score
rf = accuracy_score(y_test, pred_rf)
print(rf)

0.6284658040665434

[ ] print(classification_report(y_test,pred_rf))
```

	precision	recall	f1-score	support
0	0.63	1.00	0.77	680
1	0.00	0.00	0.00	402
accuracy			0.63	1082
macro avg	0.31	0.50	0.39	1082
weighted avg	0.39	0.63	0.49	1082

```
# confusion Maxtrix
cm3 = confusion_matrix(y_test, pred_rf)
sns.heatmap(cm3/np.sum(cm3), annot = True, fmt= '0.2%', cmap = 'Reds')
```

<AxesSubplot:>

	0	1
0	62.85%	0.00%
1	37.15%	0.00%

Activate Windows
Go to Settings to activate Windows.

+ Code + Text

Type here to search

04:21 PM
11-11-2022

XGBOOST CLASSIFIER EVALUATION:

IBM-Project-12969-16595 x IBM-Project-34997-16602 x IBM-Project-18872-16596 x Water Quality x IBM x Project Deliverables Subm x notebook078b9b7ff6 | Kag x water-quality-prediction-7 x

colab.research.google.com/drive/1c7IKyTqseuOfVt19gM0tt84Tlaazcw4#scrollTo=ad6a27a0

water-quality-prediction-7-model.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

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

```
[ ] # Calculating Accuracy Score
xgb = accuracy_score(y_test, pred_xgb)
print(xgb)
```

0.6709796672828097

```
[ ] print(classification_report(y_test, pred_xgb))
```

	precision	recall	f1-score	support
0	0.68	0.89	0.77	680
1	0.61	0.31	0.41	402
accuracy			0.67	1082
macro avg	0.65	0.60	0.59	1082
weighted avg	0.66	0.67	0.64	1082

```
# confusion Matrix
cm4 = confusion_matrix(y_test, pred_xgb)
sns.heatmap(cm4/np.sum(cm4), annot = True, fmt= '%0.2%', cmap = 'Reds')
```

<AxesSubplot:>

	0	1
0	55.73%	7.12%
1	25.79%	11.37%

Activate Windows
Go to Settings to activate Windows.

Type here to search

04:22 PM
11-11-2022

KNN EVALUATION:

IBM-Project-12969-16595 x IBM-Project-34997-16602 x IBM-Project-18872-16596 x Water Quality x IBM x Project Deliverables Subm x notebook078b9b7ff6 | Kag x water-quality-prediction-7 x

colab.research.google.com/drive/1c7iKyTqseuOfvVt19gM0tt84Tlaazcw4#scrollTo=c8a68fdd

Gmail YouTube Maps perfect Plan B # 20... My Courses | Perfec... Python for Everybo... Home | Codewars Certifications | InfyTQ AICTE Internship En... Online Courses - Le... Striver's SDE Sheet... TCS NQT 2023 Batc... Last 20 days of TCS... Bubble Sort Algori...

water-quality-prediction-7-model.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Connect Editing

```
[ ] # Calculating Accuracy Score
kn = accuracy_score(y_test, pred_kn)
print(kn)

0.6534195933456562

[ ] print(classification_report(y_test, pred_kn))
```

	precision	recall	f1-score	support
0	0.69	0.82	0.75	680
1	0.55	0.37	0.44	402
accuracy			0.65	1082
macro avg	0.62	0.60	0.59	1082
weighted avg	0.64	0.65	0.63	1082

```
# confusion Maxtrix
cm5 = confusion_matrix(y_test, pred_kn)
sns.heatmap(cm5/np.sum(cm5), annot = True, fmt= '0.2%', cmap = 'Reds')
```

<AxesSubplot:>

	0	1
0	51.66%	11.18%
1	23.48%	13.68%

Activate Windows
Go to Settings to activate Windows.

Type here to search

04:23 PM
11-11-2022

SVM EVALUATION:

colab.research.google.com/drive/1c7IKyTqseuOfVt19gM0tt84Tlaazcw4#scrollTo=c7ee370d

water-quality-prediction-7-model.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
pred_svm = model_svm.predict(X_test)
```

```
[ ] # Calculating Accuracy Score
sv = accuracy_score(y_test, pred_svm)
print(sv)
```

```
0.6885397412199631
```

```
[ ] print(classification_report(y_test, pred_kn))
```

	precision	recall	f1-score	support
0	0.69	0.82	0.75	680
1	0.55	0.37	0.44	402
accuracy			0.65	1082
macro avg	0.62	0.60	0.59	1082
weighted avg	0.64	0.65	0.63	1082

```
# confusion Maxtrix
cm6 = confusion_matrix(y_test, pred_svm)
sns.heatmap(cm6/np.sum(cm6), annot = True, fmt= '0.2%', cmap = 'Reds')
```

<AxesSubplot:>

	0	1
0	57.58%	5.27%
1	25.88%	11.28%

Activate Windows
Go to Settings to activate Windows.

Type here to search

04:24 PM
11-11-2022

COMPARISSON:

water-quality-prediction-7-model.ipynb

```
models = pd.DataFrame({
    'Model': ['Logistic Regression', 'Decision Tree', 'Random Forest', 'XGBoost', 'KNeighbours', 'SVM', 'AdaBoost'],
    'Accuracy_score': [lg, dt, rf, xgb, kn, sv, ada]
})
models
sns.barplot(x='Accuracy_score', y='Model', data=models)

models.sort_values(by='Accuracy_score', ascending=False)
```

	Model	Accuracy_score
5	SVM	0.688540
3	XGBoost	0.670980
4	KNeighbours	0.653420
1	Decision Tree	0.645102
6	AdaBoost	0.634011
0	Logistic Regression	0.628466
2	Random Forest	0.628466

Conclusion :- Here SVM classifier has achieved highest accuracy.

CONCLUSION:

Out of all we found that *SVM Classifier* has the highest accuracy among all , so we are going to use *SVM Classifier* to train **our Water Prediction machine learning model**