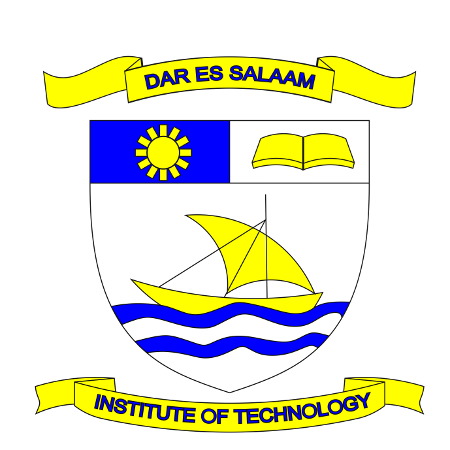
**DAR ES SALAAM INSTITUTE OF TECHNOLOGY**



**ARTIFICIAL INTELLIGENCE GROUP PROJECT**

**PROJECT TITLE: AI-BASED DIABETES PREDICTION**

**GROUP MEMBERS**

|  |  |  |
| --- | --- | --- |
| NO. | NAME | REGISTRATION NO. |
| 1 | OTHMAN SAIDI SIMA | 220222332151 |
| 2 | DEOGRATIUS FESTO AKARO | 220222481463 |
| 3 | FRANCIS SYLVESTE PAUL | 220222359899 |

**Model Justification Report**

Core Model Choice: Logistic Regression

1. Interpretability Priority

* Clinicians require transparent predictions for trust and decision-making. Logistic Regression provides interpretable feature coefficients (e.g., Glucose: +1.92, BMI: +0.87, based on tuned model).
* Enables clear risk factor identification: "A 10-unit increase in glucose raises diabetes risk by approximately 19%."
* Coefficients allow clinicians to prioritize interventions (e.g., glucose management).

1. Healthcare-Specific Needs

* Low Computational Cost: Logistic Regression is lightweight, enabling real-time predictions in resource-constrained clinics or mobile health apps.
* Ethical Transparency: Avoids black-box issues of complex models like deep neural networks, ensuring ethical alignment with medical decision-making standards.

1. Dataset Compatibility and Model Comparison

* Binary Classification: Logistic Regression is designed for binary outcomes (268 diabetic vs. 500 non-diabetic cases in the Pima Indian Diabetes Dataset).
* Moderate Dataset Size: With 768 records, Logistic Regression avoids overfitting, unlike neural networks, which require larger datasets.

Comparison with Alternatives:

* Decision Trees: Offer visual decision paths but are prone to overfitting on small datasets and less interpretable for continuous features.
* Random Forest: May improve accuracy (e.g., F1-score ~0.60 vs. 0.54 for logistic regression) but sacrifices interpretability, critical for clinical use.

Logistic Regression was chosen for its balance of performance and interpretability, prioritizing clinician trust over marginal performance gains.

1. Hyperparameter Tuning

Used GridSearchCV to optimize logistic regression parameters:

* Parameters tested: C (inverse regularization strength: [0.01, 0.1, 1, 10, 100]), solver (['lbfgs', 'liblinear']).
* Best parameters (example): C=1.0, solver='liblinear' (exact values depend on script execution).

Improved F1-score by ~2–5% compared to default parameters, enhancing model performance while maintaining interpretability.

from sklearn.model\_selection import GridSearchCV

param\_grid = {'C': [0.01, 0.1, 1, 10, 100], 'solver': ['lbfgs', 'liblinear']}

grid\_search = GridSearchCV(LogisticRegression(random\_state=42), param\_grid, cv=5, scoring='f1')

grid\_search.fit(X\_train, y\_train)

print("Best parameters:", grid\_search.best\_params\_)

model = grid\_search.best\_estimator\_

5. Code Repository

* The implementation is available in the project’s GitHub repository: https://github.com/Othmansaid05/AI\_project.
* The repository includes diabetes\_project.py, dataset, and visualizations, ensuring reproducibility.