Model Development Phase

Date	03 August 2025
Project Title	Anemia Sense-Machine Learning for Precise Anemia Recognition.
Maximum Marks	4 Marks

Initial Model Training Code, Model Validation and Evaluation Report

The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include classification reports, accuracy, and confusion matrices for multiple models, presented through respective screenshots.

Initial Model Training Code:

```
#Logistic Regression
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
from sklearn.metrics import classification_report
logistic_regression = LogisticRegression()
logistic_regression.fit(x_train, y_train)
y_pred = logistic_regression.predict(x_test)
acc_lr = accuracy_score(y_test,y_pred)
c_lr = classification_report (y_test,y_pred)
print('Accuracy Score: ',acc_lr)
print(c_lr)
```

```
#RANDOM FOREST
from sklearn.ensemble import RandomForestClassifier
random_forest = RandomForestClassifier()
random_forest.fit(x_train, y_train)
y_pred = random_forest.predict(x_test)
acc_rf = accuracy_score(y_test,y_pred)
c_rf = classification_report(y_test,y_pred)
print('Accuracy Score: ',acc_rf)
print(c_rf)
```

```
#Decision Tree
from sklearn.tree import DecisionTreeClassifier
decision_tree_model = DecisionTreeClassifier()
decision_tree_model.fit(x_train, y_train)
y_pred = decision_tree_model.predict(x_test)
acc_dt = accuracy_score(y_test,y_pred)
c_dt = classification_report (y_test,y_pred)
print('Accuracy Score: ',acc_dt)
print(c_dt)
```

```
#Gaussion Naive Bayes
from sklearn.naive_bayes import GaussianNB
NB = GaussianNB()
NB.fit(x_train, y_train)
y_pred = NB.predict(x_test)
acc_nb = accuracy_score(y_test,y_pred)
c_nb = classification_report(y_test,y_pred)
print('Accuracy Score: ',acc_nb)
print(c_nb)
```

```
#Support Vector Classifier
from sklearn.svm import SVC
support_vector = SVC()
support_vector.fit(x_train, y_train)
y_pred = support_vector.predict(x_test)
acc_svc = accuracy_score(y_test,y_pred)
c_svc = classification_report(y_test,y_pred)
print('Accuracy Score: ',acc_svc)
print(c_svc)
```

```
#Gradient Boost Classifier
from sklearn.ensemble import GradientBoostingClassifier
GBC=GradientBoostingClassifier()
GBC.fit(x_train, y_train)
y_pred = GBC.predict(x_test)
acc_gbc = accuracy_score(y_test,y_pred)
c_gbc = classification_report(y_test,y_pred)
print('Accuracy Score: ',acc_gbc)
print(c_gbc)
```

Model Validation and Evaluation Report:

Model	Classification Report	F1 Score
Logistic Regression	Accuracy Score: 1.0 precision recall f1-score support 0 1.00 1.00 1.00 167 1 1.00 1.00 1.00 118 accuracy 1.00 285 macro avg 1.00 1.00 1.00 285 weighted avg 1.00 1.00 1.00 285	100%
Random Forest	Accuracy Score: 1.0 precision recall f1-score support 0 1.00 1.00 1.00 167 1 1.00 1.00 1.00 118 accuracy 1.00 285 macro avg 1.00 1.00 1.00 285 weighted avg 1.00 1.00 1.00 285	100%
Decision Tree	Accuracy Score: 1.0 precision recall f1-score support 0 1.00 1.00 1.00 167 1 1.00 1.00 1.00 118 accuracy 1.00 285 macro avg 1.00 1.00 1.00 285 weighted avg 1.00 1.00 1.00 285	100%

Gaussian Naïve Bayes	Accuracy Score: 0.9403508771929825 precision recall f1-score support 0 0.97 0.93 0.95 167 1 0.90 0.96 0.93 118 accuracy 0.94 285 macro avg 0.94 0.94 0.94 285 weighted avg 0.94 0.94 0.94 285	94%
Support Vector Classifier	Accuracy Score: 0.9017543859649123 precision recall f1-score support 0 0.95 0.87 0.91 167 1 0.84 0.94 0.89 118 accuracy 0.90 285 macro avg 0.90 0.91 0.90 285 weighted avg 0.91 0.90 0.90 285	90%
Gradient Boost Classifier	Accuracy Score: 1.0 precision recall f1-score support 0 1.00 1.00 1.00 167 1 1.00 1.00 1.00 118 accuracy 1.00 285 macro avg 1.00 1.00 1.00 285 weighted avg 1.00 1.00 1.00 285	100%