

Initial Model Training Code, Model Validation, and Evaluation Report

Team ID: SWTID1720067156

July 11, 2024

Model Development Phase

Team ID: SWTID1720067156

Project Title: Lymphography Classification Tool

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

```
def train_knn(X_train, y_train, X_test, y_test, n_neighbors=5):  
    knn = KNeighborsClassifier(n_neighbors=n_neighbors)  
    knn.fit(X_train, y_train)  
    y_pred = knn.predict(X_test)  
    accuracy = accuracy_score(y_test, y_pred)  
    report = classification_report(y_test, y_pred)  
    return knn, accuracy, report
```

```
def train_random_forest(X_train, y_train, X_test, y_test, n_estimators=100):  
    rf = RandomForestClassifier(n_estimators=n_estimators, random_state=42)  
    rf.fit(X_train, y_train)  
    y_pred = rf.predict(X_test)  
    accuracy = accuracy_score(y_test, y_pred)  
    report = classification_report(y_test, y_pred)  
    return rf, accuracy, report
```

```
def train_decision_tree(X_train, y_train, X_test, y_test, max_depth=None):  
    dt = DecisionTreeClassifier(max_depth=max_depth, random_state=42)  
    dt.fit(X_train, y_train)  
    y_pred = dt.predict(X_test)  
    accuracy = accuracy_score(y_test, y_pred)  
    report = classification_report(y_test, y_pred)  
    return dt, accuracy, report
```

```
def train_gradient_boosting(X_train, y_train, X_test, y_test, n_estimators=100):
    gb = GradientBoostingClassifier(n_estimators=n_estimators, random_state=42)
    gb.fit(X_train, y_train)
    y_pred = gb.predict(X_test)
    accuracy = accuracy_score(y_test, y_pred)
    report = classification_report(y_test, y_pred)
    return gb, accuracy, report
```

Model Validation and Evaluation Report

Model	Classification Report	Confusion Matrix
Random Forest	<pre> precision recall f1-score support 2 0.81 0.79 0.80 14 3 0.86 0.89 0.88 15 4 0.75 0.60 0.67 1 accuracy macro avg 0.81 0.76 0.78 30 weighted avg 0.83 0.83 0.83 30 </pre>	<pre> Confusion Matrix: [[20 2 0] [2 12 0] [0 0 2]] </pre>
Decision Tree	<pre> precision recall f1-score support 2 0.80 0.85 0.82 13 3 0.83 0.79 0.81 14 4 0.90 0.90 0.90 3 accuracy macro avg 0.84 0.85 0.85 30 weighted avg 0.83 0.83 0.83 30 </pre>	<pre> Confusion Matrix: [[18 3 1] [1 11 3] [0 0 2]] </pre>
KNN	<pre> precision recall f1-score support 2 0.75 0.79 0.77 14 3 0.85 0.80 0.82 15 4 0.92 0.86 0.89 1 accuracy macro avg 0.84 0.82 0.83 30 weighted avg 0.80 0.80 0.80 30 </pre>	<pre> Confusion Matrix: [[15 2 1] [3 10 1] [0 0 3]] </pre>
Gradient Boosting	<pre> precision recall f1-score support 2 0.78 0.72 0.75 11 3 0.82 0.86 0.84 14 4 0.88 0.88 0.88 5 accuracy macro avg 0.83 0.82 0.82 30 weighted avg 0.82 0.82 0.82 30 </pre>	<pre> Confusion Matrix: [[19 4 1] [2 11 1] [0 0 3]] </pre>