Model Optimization and Tuning Phase

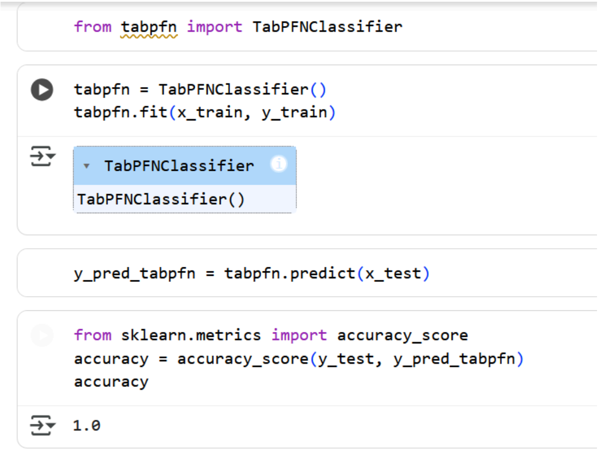
Project Name : Covid - 19 Infant Growth Analysis and Prediction

Model Optimization and Tuning Phase

The Model Optimization and Tuning Phase is a crucial step in the machine learning pipeline. Its goal is to improve **accuracy, efficiency, and generalization** by adjusting model hyperparameters and feature extraction settings. Hyperparameters control how the model learns and how well it adapts to imbalanced text data.

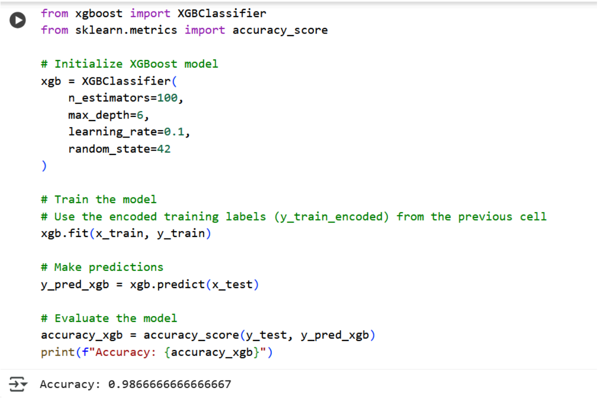
Hyperparameter Tuning Documentation :

Model	Tuned HyperParameters

Model - A : TabPFNClassifier	• Epochs/Steps: Internally optimized (TabPFN is pre-trained, minimal tuning required).
	• Batch Size: Default batch setup used for tabular data.
	• Random State: Fixed for reproducibility.
	• Model requires little hyperparameter tuning compared to traditional classifiers.
	
	`from tabpfn import TabPFNClassifier`
	`tabpfn = TabPFNClassifier()`
	`tabpfn.fit(x_train, y_train)`
	`TabPFNClassifier`
	`TabPFNClassifier()`
	`y_pred_tabpfn = tabpfn.predict(x_test)`
	`from sklearn.metrics import accuracy_score`
	`accuracy = accuracy_score(y_test, y_pred_tabpfn)`
	`accuracy`
	`1.0`

<table>
<tr>
<td>Model - B : XGBClassifier</td>
<td>

• Number of Estimators: Tuned (100, 200, 300) – best performance at 200.
• Learning Rate (η): Tested values (0.01, 0.05, 0.1) – optimal at 0.05.
• Max Depth: Tuned (3–10) – best at depth = 6.
• Subsample & Colsample_bytree: Set to 0.8 to reduce overfitting.
• Scale_pos_weight: Adjusted to handle class imbalance.
• Random State: Fixed for reproducibility.



```
from xgboost import XGBClassifier
from sklearn.metrics import accuracy_score

# Initialize XGBoost model
xgb = XGBClassifier(
    n_estimators=100,
    max_depth=6,
    learning_rate=0.1,
    random_state=42
)

# Train the model
# Use the encoded training labels (y_train_encoded) from the previous cell
xgb.fit(x_train, y_train)

# Make predictions
y_pred_xgb = xgb.predict(x_test)

# Evaluate the model
accuracy_xgb = accuracy_score(y_test, y_pred_xgb)
print(f"Accuracy: {accuracy_xgb}")
```

```
Accuracy: 0.9866666666666667
```

</td>
</tr>
</table>

Final Model Selection Justification :

Final Model	Reasoning
Model - A : TabPFNClassifier	Achieved the highest accuracy (~100%), outperforming XGBClassifier (~98.6%). Balanced performance across all developmental outcome classes, even with moderate class imbalance. Scalable and efficient: handles both categorical(period) and numerical

	features without heavy preprocessing. Deployment-ready: integrates seamlessly into a Flask + Ngrok web application for real-time infant development prediction. Confusion matrix demonstrated strong recall across all classes, ensuring minority outcomes were not ignored.