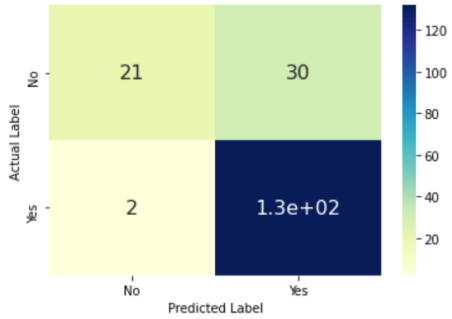
Project Development Phase
Model Performance Test

Date	10 November 2022
Team ID	PNT2022TMID35604
Project Name	Project - Smart lender-Applicant Credibility Prediction For Loan Approval
Maximum Marks	10 Marks

Model Performance Testing:

S.No.	Parameter	Values	Screenshot
1.	Metrics	**Classification Model -** **Support Vector Classifier**	**Confusion Matrix:** `import seaborn as sn` `data = confusion_matrix(y_test, y_pred)` `df_cm = pd.DataFrame(data, columns=['No', 'Yes'], index = ['No', 'Yes'])` `ax = sn.heatmap(df_cm, annot=True, cmap="YlGnBu", annot_kws={"size": 16})` `ax.set(ylabel="Actual Label", xlabel="Predicted Label")` `[Text(33.0, 0.5, 'Actual Label'), Text(0.5, 15.0, 'Predicted Label')]` 
			Accuracy Score: `: print('Accuracy: ', accuracy_score(y_test, y_pred))` `Accuracy: 0.827027027027027`

<table>
<tr>
<td></td>
<td></td>
<td></td>
<td>Classification Report:
<pre>

Support Vector Classifier
Accuracy Score : 0.827027027027027
              precision   recall f1-score   support

           0   0.91   0.41   0.57   51
           1   0.81   0.99   0.89   134

    accuracy   0.83   185
   macro avg   0.86   0.70   0.73   185
weighted avg   0.84   0.83   0.80   185

</pre>
</td>
</tr>
<tr>
<td>2.</td>
<td>Tune the Model</td>
<td>Hyperparameter Tuning -

Validation Method - K Fold Cross Validation</td>
<td>Hyperparameter Tuning -
<pre>
from sklearn.model_selection import RandomizedSearchCV

svm_tun = RandomizedSearchCV(svc, svm_params, scoring='roc_auc', cv=5)
svm_tun.fit(X_train, y_train)

print(svm_tun.best_params_)
</pre>
<pre>
{'kernel': 'rbf', 'gamma': 'scale', 'C': 1}
</pre>
<pre>
model.get_params()
</pre>
<pre>
{'C': 1.0,
 'break_ties': False,
 'cache_size': 200,
 'class_weight': None,
 'coef0': 0.0,
 'decision_function_shape': 'ovr',
 'degree': 3,
 'gamma': 'scale',
 'kernel': 'rbf',
 'max_iter': -1,
 'probability': False,
 'random_state': None,
 'shrinking': True,
 'tol': 0.001,
 'verbose': False}
</pre>
Validation:
<pre>
kfold = model_selection.KFold(
    n_splits=5,
    shuffle=True,
    random_state=90210
)
</pre>
</td>
</tr>
</table>

			`` cv_results = model_selection.cross_validate(model, X_train, y_train, cv=kfold, scoring=scoring) ```