THE FUTURE SALES PREDICTION

PROGRESS USING PYTHON

Phase 4 submission Documents

Project Title : The future sales prediction

User Authentication

Name: Timpul Dharshini.B

Reg No:712221205041

College: park college of engineering and technology

User Authentication :

- Step 1 : Import libraries

  A library is a collection of functions that can be added to your Python code and called as necessary, just like any other function.

- Step 2 : load and explore data

  Data exploration is a key aspect of data analysis and model building. Without spending significant time on understanding the data.

- Step 3 : Data Processing

Gathering and manipulating data elements to return useful, potentially valuable information.

- Step 4 : Choose and train the model

  One of the most common methods used to predict sales is regression analysis. This method involves using historical sales data to train a model that can predict future sales.

- Step 5 : Make predictions

  Sales forecasting, decision trees can be used to make predictions about future sales by considering multiple factors that impact sales.
- Step 6 : Evaluate the model

  The main models are trend analysis, regression analysis, and causal analysis.

- Step 7 : Make future sales prediction

  1)Sales cycle length forecasting. This forecasting method ranks opportunities based on how long a potential customer has been communicating with 2)the company. …
  3)Intuitive forecasting. …
  4)Historical forecasting. …
  5)Opportunity stage forecasting. …
  6)Pipeline forecasting. …
  7)Multivariable forecasting.

```
import pandas as pd
from sklearn.model_selection import
    train_test_split
from sklearn.linear_model import
    LinearRegression
from sklearn.metrics import mean_squared_error
# Load your sales data into a Pandas DataFrame
sales_data = pd.read_csv('sales_data.csv')

# Assuming your data has columns like
    'feature1', 'feature2', 'sales'
# Separate features and target variable
X = sales_data[['feature1', 'feature2']]
y = sales_data['sales']

# Split data into training and testing sets
X_train, X_test, y_train, y_test =
    train_test_split(X, y, test_size=0.2,
    random_state=42)
# Initialize the Linear Regression model
model = LinearRegression()

# Train the model using the training data
model.fit(X_train, y_train)
# Make predictions using the test data
predictions = model.predict(X_test)
```