FUTURE SALES PREDICTION

IBM NAAN MUDHALVAN

PROJECT SUBMISSION PHASE-3

Future sales prediction is the process of predicting what the demand for certain products will be in the future. This helps manufracturers to decide what they should produce and guides retailers toward what they should stock.

Sales forecasting is the process of estimating future revenue by predicting how much of a product or service will sell in the next week, month, quarter, or year. At its simplest, a sales forecast is a projected measure of how a market will respond to a company's go-to-market efforts.

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.

The model can take into account factors such as past sales, marketing campaigns, and economic indicators to make its predictions.

DATA GATHERING:

Dataset is taken from kaggle competition and it can be downloaded from here:

https://www.kaggle.com/datasets/chakradharmattapalli/future-sales-prediction

Machine Learning Algorithms:

Decision Trees and Random Forests: Useful for capturing complex relationships in the data.

Gradient Boosting Models (e.g., XGBoost, LightGBM): Excellent for predictive modeling and handling non-linear relationships.

Predicting future sales involves using data science techniques. A simple algorithmic approach could involve:

1. Data Collection:

Historical sales data are gathered from the kaggle competition and included a relevant features like promotions, holidays, and other factors affecting sales.

2. Data Preprocessing:

Processing performed on raw data to prepare it for another data processing procedure.

5. Training the Model:

The datas were split into training and validation sets.

8. Prediction:

Use the trained model to predict future sales based on new data.

9. Monitoring and Updating:

Regularly monitor the model's performance and update it as needed with new data.

Feature Engineering:

Feature engineering includes remodeling raw data into a format that successfully represents the underlying patterns within the data. It involves selecting, combining, and crafting attributes that capture the relationships between variables, enhancing the predictive power of machine learning models. These engineered features acts as the input for algorithms, using progressed performance and robustness.

Feature Tools:

A python library centered on automated feature engineering, particularly for time-series and relational data. It automates producing new features by leveraging domain-specific knowledge and entity relationships.

Applications: Creating time-based features, aggregating data over different time intervals, and handling a couple of associated data tables.

Scikit-Research:

A broadly used python library that gives various feature selection, extraction, and preprocessing tools. It provides a steady API, making enforcing numerous feature engineering strategies easy. Its wide adoption guarentees tremendous community suppoprt and resources.

Applications: Handling missing values, transforming categorical variables using one-hot encoding, and standardizing features with scaling strategies.

Scikit-Learn Models:

from sklearn.preprocessing import MinMaxScaler

from sklearn.linear_model import

LinearRegression

from sklearn.metrics importmean_squared_error,mean_absolute_error,
r2_score

from sklearn.ensemble import

RandomForestRegress

from xgboost.sklearn import XGBRegressor

from sklearn.model_selection import KFold, cross_val_score, train_test_split