Product demand forecasting :

Problem Description :

Product demand forecasting is a critical process for businesses to effectively manage their inventory levels, production schedules, and sales strategies. The objective of this project is to develop a machine learning model that can accurately forecast the demand for a given product over a certain time period.

Dataset:

The dataset for this project can be sourced from Kaggle: https://www.kaggle.com/felixzhao/productdemandforecasting. It contains historical demand data for a range of products across multiple locations, as well as information about the product and location attributes.

Framework and Steps:

1. **Data preprocessing:** This involves cleaning and transforming the raw data to make it suitable for machine learning algorithms. This step may include removing missing values, handling outliers, encoding categorical variables, and scaling numeric features.

2. **Feature engineering:** This involves creating new features from the existing ones to improve the performance of the machine learning model. This step may include lag features, rolling mean features, seasonality features, and trend features.

3. **Model selection:** This involves selecting the appropriate machine learning algorithm to forecast the demand for the product. The choice of algorithm may depend on the type of data, the forecasting horizon, and the required accuracy.

4. **Model training:** This involves training the selected machine learning model on the preprocessed and engineered data. The model is trained on a subset of the data and validated on another subset to avoid overfitting.

5. **Hyperparameter tuning:** This involves fine-tuning the hyperparameters of the machine learning model to optimize its performance. This step may involve using grid search or random search to search through a range of hyperparameters.

6. **Model evaluation:** This involves evaluating the performance of the machine learning model on a test set of data. The model is evaluated using various metrics such as mean absolute error (MAE), mean squared error (MSE), and root mean squared error (RMSE).

Deliverables:

The deliverables for this project include a machine learning model that can accurately forecast the demand for a given product over a certain time period. The model should be evaluated on a test set of data and the performance metrics should be reported. A report should also be generated that documents the data preprocessing steps, feature engineering techniques, machine learning algorithm selection, hyperparameter tuning, and model evaluation results.

Code Explanation :

Here is the simple explanation for the code which is provided in the code.py file.

1. **Feature Engineering:** One can explore different feature engineering techniques to create new features and improve model performance. Some examples include:

    - Create seasonal features to capture patterns in sales based on different seasons of the year.

    - Create lag features to capture the impact of past sales on current sales.

    - Create rolling mean and standard deviation features to capture trends and seasonality in sales.

2. **Ensemble Modeling**: One can explore different ensemble modeling techniques to combine the predictions of multiple models and improve overall accuracy. Some examples include:

    - Voting Classifier: Combine predictions of multiple models by taking the majority vote.

    - Stacking: Train a meta-model that learns to combine the predictions of multiple base models.

    - Bagging and Boosting: Train multiple models on different subsets of the data and combine their predictions to reduce variance and bias respectively.

3. **Time Series Modeling:** One can explore time series modeling techniques to capture the sequential nature of the data and improve model performance. Some examples include:

    - **ARIMA (AutoRegressive Integrated Moving Average):** A statistical model that uses past values and their differences to make predictions.

- **SARIMA (Seasonal ARIMA):** An extension of ARIMA that takes into account seasonality in the data.

- **LSTM (Long Short-Term Memory)**: A deep learning model that can capture long-term dependencies in time series data.

Step-by-Step Guide for Feature Engineering:

1. Explore the data and identify potential patterns and trends in sales.

2. Create new features based on these patterns and trends.

3. Test the new features with different models and evaluate their performance.

4. Repeat steps 2 and 3 until the desired level of accuracy is achieved.

Step-by-Step Guide for Ensemble Modeling:

1. Train multiple models on the same data.

2. Combine the predictions of these models using a predefined ensemble method.

3. Evaluate the performance of the ensemble model and compare it with individual models.

4. Experiment with different ensemble methods and hyperparameters to improve performance.

Step-by-Step Guide for Time Series Modeling:

1. Preprocess the data by transforming it into a stationary time series.

2. Split the data into training and validation sets.

3. Train the time series model on the training data.

4. Evaluate the performance of the model on the validation data.

5. Tune the hyperparameters of the model to improve performance.

6. Test the final model on a hold-out test set.

Exercise :

Try to answers the following questions by yourself to check your understanding for this project. If stuck, detailed answers for the questions are also provided.

1. **How can we improve the performance of the model for demand forecasting?** Answer: We can improve the performance of the model by fine-tuning hyperparameters and optimizing the model architecture. We can also try using different types of models like LSTM, GRU, or Transformer models.

2. **What are some possible features that can be added to the dataset to improve the accuracy of the model?**
Answer: Some possible features that can be added to the dataset to improve the accuracy of the model are holiday information, promotions or discounts, weather conditions, and product reviews.

3. **Can we use a different evaluation metric other than RMSE to evaluate the performance of the model?**
Answer: Yes, we can use different evaluation metrics like MAE, MAPE, or R-squared to evaluate the performance of the model. The choice of the evaluation metric will depend on the problem at hand and the business requirements.

4. **What would be the impact of removing outliers from the dataset on the performance of the model?**
Answer: Removing outliers from the dataset can improve the accuracy of the model as outliers can have a significant impact on the model's performance. However, it is Important to carefully analyze the outliers and determine whether they are true outliers or just extreme values.

5. **How can we use this model to make predictions for multiple products at once?**
Answer: To make predictions for multiple products at once, we can train separate models for each product and use a forecasting pipeline to make predictions for all products simultaneously. Alternatively, we can use a single model that can predict demand for multiple products by incorporating product-specific features into the model.