# Product Demand Prediction with Machine Learnings-Guidelines

**Name : Sathiyaprakasan.G**

**Regno : 912721104028**

**Email :** [**pulsarbalaji307@gmail.com**](mailto:pulsarbalaji307@gmail.com)

**PROJECT: Project Demand Prediction With Machine Learning-Guidelines\*\*\n",**

**"\*\*PHASE 4: Development Part2\*\*\n",**

**"The Product Demand Prediction model by:\n",**

**"\n",**

**"\n",**

**"\* Feature Engineering\n",**

**"\* Model Training\n",**

**"\* Evaluation.\n",**

**"\n",**

**"\n",**

**"\n",**

**"\*\*Introduction\*\*\n",**

**"\n",**

**"In today's fast-paced and competitive business landscape, accurately predicting product demand is crucial for optimizing inventory management, reducing costs, and ensuring customer satisfaction. Machine learning has emerged as a powerful tool to tackle this challenge. This introduction will provide an overview of product demand prediction using machine learning.\n",**

**"\n",**

**"\n",**

**"\n",**

**"\*\*1. The Importance of Demand Prediction:\*\*\n",**

**"Predicting product demand is essential for businesses across various industries. It helps in maintaining the right inventory levels, avoiding overstocking or stockouts, and making informed decisions about production and distribution.\n",**

**"\n",**

**"\*\*2. Traditional vs. Machine Learning Methods:\*\*\n",**

**"\n",**

**"Traditional demand forecasting methods often rely on historical data and simplistic models. Machine learning techniques, on the other hand, leverage advanced algorithms to analyze vast datasets, including various factors such as seasonality, economic indicators, and even social media sentiment.\n",**

**"\n",**

**"\*\*3. Key Steps in Demand Prediction:\*\*\n",**

**"\n",**

**"a. \*\*Data Collection\*\*: Gathering historical sales data, market trends, and relevant variables.\n",**

**"\n",**

**"b.\*\*Data Preprocessing\*\*: Cleaning and preparing the data for analysis, including handling missing values and outliers.\n",**

**"\n",**

**"c. \*\*Feature Engineering\*\*: Selecting and creating relevant features that can influence demand.\n",**

**"\n",**

**"d. \*\*Model Selection\*\*: Choosing the appropriate machine learning algorithm based on the data and the problem.\n",**

**"\n",**

**"e. \*\*Model Training\*\*: Training the chosen model using historical data.\n",**

**"\n",**

**"f. \*\*Evaluation\*\*: Assessing the model's performance using metrics like Mean Absolute Error (MAE) or Root Mean Square Error (RMSE).\n",**

**"\n",**

**"g. \*\*Deployment\*\*: Implementing the model to make real-time predictions.\n",**

**"\n",**

**"\*\*4. Data Sources:\*\*\n",**

**"\n",**

**"Machine learning models for demand prediction can be fed with data from various sources, including sales records, customer reviews, weather data, economic indicators, and more.\n",**

**"\n",**

**"\n",**

**"\*\*5. Challenges in Demand Prediction:\*\*\n",**

**"\n",**

**"Predicting demand accurately is not without its challenges. Factors like sudden market shifts, unforeseen events (e.g., pandemics), and the need for real-time predictions can complicate the process.\n",**

**"\n",**

**"\*\*6. Benefits of Machine Learning in Demand Prediction:\*\*\n",**

**"\n",**

**"Machine learning models can adapt to changing market conditions, handle complex patterns, and provide more accurate forecasts compared to traditional methods.\n",**

**"\n",**

**"\n",**

**"\n",**

**"\n",**

**"\n",**

**"\n",**

**"\n",**

**"\n",**

**"\n",**

**"\n",**

**"\n",**

**"\n",**

**"\n",**

**"\n",**

**"\n"**

**],**

**"metadata": {**

**"id": "M4VQaj2kCwCh"**

**}**

**},**

**{**

**"cell\_type": "markdown",**

**"source": [**

**"# \*\*Feature Engineering\*\*\n",**

**"\n",**

**"Certainly, I can provide you with a simple Python program for feature engineering with an output example. Feature engineering is a crucial step in machine learning and data analysis to create meaningful input features for your models. Here's a basic example:\n"**

**],**

**"metadata": {**

**"id": "xFOPD4q6Thk1"**

**}**

**},**

**{**

**"cell\_type": "code",**

**"source": [**

**"import pandas as pd\n",**

**"from sklearn.feature\_extraction.text import TfidfVectorizer\n",**

**"\n",**

**"# Sample dataset\n",**

**"data = {\n",**

**" 'text\_data': ['This is the first document.', 'This document is the second document.', 'And this is the third one.'],\n",**

**" 'numeric\_data': [10, 20, 30]\n",**

**"}\n",**

**"\n",**

**"df = pd.DataFrame(data)\n",**

**"\n",**

**"# Feature engineering: TF-IDF on text\_data\n",**

**"tfidf\_vectorizer = TfidfVectorizer()\n",**

**"tfidf\_matrix = tfidf\_vectorizer.fit\_transform(df['text\_data'])\n",**

**"\n",**

**"# Creating a DataFrame from the TF-IDF matrix\n",**

**"tfidf\_df = pd.DataFrame(data=tfidf\_matrix.toarray(), columns=tfidf\_vectorizer.get\_feature\_names\_out())\n",**

**"\n",**

**"# Output the original and engineered features\n",**

**"print(\"Original DataFrame:\")\n",**

**"print(df)\n",**

**"\n",**

**"print(\"\\nEngineered TF-IDF Features:\")\n",**

**"print(tfidf\_df)"**

**],**

**"metadata": {**

**"colab": {**

**"base\_uri": "https://localhost:8080/"**

**},**

**"id": "nGwDtQH8CeUw",**

**"outputId": "5c8f2215-88b7-4bab-9e77-1a8a7151049b"**

**},**

**"execution\_count": 11,**

**"outputs": [**

**{**

**"output\_type": "stream",**

**"name": "stdout",**

**"text": [**

**"Original DataFrame:\n",**

**" text\_data numeric\_data\n",**

**"0 This is the first document. 10\n",**

**"1 This document is the second document. 20\n",**

**"2 And this is the third one. 30\n",**

**"\n",**

**"Engineered TF-IDF Features:\n",**

**" and document first is one second the \\\n",**

**"0 0.00000 0.469417 0.617227 0.364544 0.00000 0.000000 0.364544 \n",**

**"1 0.00000 0.728445 0.000000 0.282851 0.00000 0.478909 0.282851 \n",**

**"2 0.49712 0.000000 0.000000 0.293607 0.49712 0.000000 0.293607 \n",**

**"\n",**

**" third this \n",**

**"0 0.00000 0.364544 \n",**

**"1 0.00000 0.282851 \n",**

**"2 0.49712 0.293607 \n"**

**]**

**}**

**]**

**},**

**{**

**"cell\_type": "code",**

**"source": [**

**"pip install pandas"**

**],**

**"metadata": {**

**"colab": {**

**"base\_uri": "https://localhost:8080/"**

**},**

**"id": "fo4pppjeVPR0",**

**"outputId": "c3b72832-3bc0-4576-add2-fe3b413db007"**

**},**

**"execution\_count": 2,**

**"outputs": [**

**{**

**"output\_type": "stream",**

**"name": "stdout",**

**"text": [**

**"Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (1.5.3)\n",**

**"Requirement already satisfied: python-dateutil>=2.8.1 in /usr/local/lib/python3.10/dist-packages (from pandas) (2.8.2)\n",**

**"Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas) (2023.3.post1)\n",**

**"Requirement already satisfied: numpy>=1.21.0 in /usr/local/lib/python3.10/dist-packages (from pandas) (1.23.5)\n",**

**"Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.1->pandas) (1.16.0)\n"**

**]**

**}**

**]**

**},**

**{**

**"cell\_type": "code",**

**"source": [**

**"pip install standard install"**

**],**

**"metadata": {**

**"colab": {**

**"base\_uri": "https://localhost:8080/"**

**},**

**"id": "jGRPlU-qVgKk",**

**"outputId": "5ca808e2-0ec1-4c86-d14c-a85a68bde8df"**

**},**

**"execution\_count": 3,**

**"outputs": [**

**{**

**"output\_type": "stream",**

**"name": "stdout",**

**"text": [**

**"Collecting standard\n",**

**" Downloading standard-1.0.3.tar.gz (3.1 kB)\n",**

**" Preparing metadata (setup.py) ... \u001b[?25l\u001b[?25hdone\n",**

**"Requirement already satisfied: install in /usr/local/lib/python3.10/dist-packages (1.3.5)\n",**

**"Requirement already satisfied: termcolor in /usr/local/lib/python3.10/dist-packages (from standard) (2.3.0)\n",**

**"Building wheels for collected packages: standard\n",**

**" Building wheel for standard (setup.py) ... \u001b[?25l\u001b[?25hdone\n",**

**" Created wheel for standard: filename=standard-1.0.3-py3-none-any.whl size=3373 sha256=7471c58f7f2671a64097c18fdd19e503c90ae9a6e669f9dcf89a1073268507f3\n",**

**" Stored in directory: /root/.cache/pip/wheels/59/d3/94/c65dde6f25d6279ca86794e9600cb54f2203ac0b5b15562e50\n",**

**"Successfully built standard\n",**

**"Installing collected packages: standard\n",**

**"Successfully installed standard-1.0.3\n"**

**]**

**}**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"source": [**

**"# \*\*Modal Training\*\*\n",**

**"Creating a model to predict product demand involves various steps and would typically require coding in a programming language such as Python. Here's a high-level overview of how you can create such a program:\n",**

**"\n",**

**"Data Collection: Gather historical data on product demand. This data should include information like time, product attributes, pricing, marketing efforts, and actual demand.\n",**

**"\n",**

**"Data Preprocessing: Clean and prepare the data. This may involve handling missing values, normalizing data, and encoding categorical variables.\n",**

**"\n",**

**"Feature Engineering: Create relevant features that can help predict demand. For example, you might calculate moving averages, seasonality, or lag features.\n",**

**"\n",**

**"Split Data: Divide your data into training and testing datasets to evaluate the model's performance.\n",**

**"\n",**

**"Choose a Model: Select an appropriate machine learning model for demand prediction. Time series forecasting models like ARIMA, Prophet, or machine learning models like linear regression, decision trees, or neural networks can be used.\n",**

**"\n",**

**"Train the Model: Train the selected model using the training data.\n",**

**"\n",**

**"Evaluate the Model: Use the testing data to assess the model's performance using metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), or Root Mean Squared Error (RMSE).\n",**

**"\n",**

**"Tune Hyperparameters: Optimize the model's hyperparameters to improve its performance.\n",**

**"\n",**

**"Make Predictions: Once the model is trained and tuned, you can make predictions on future demand based on new data.\n",**

**"\n",**

**"Visualize Results: Visualize the model's predictions and compare them to actual demand to assess accuracy.\n",**

**"\n",**

**"Here's a simple Python code snippet that demonstrates a basic linear regression model for product demand prediction:bold text"**

**],**

**"metadata": {**

**"id": "aLxQDADbW9xy"**

**}**

**},**

**{**

**"cell\_type": "code",**

**"source": [**

**"import numpy as np\n",**

**"from sklearn.model\_selection import train\_test\_split\n",**

**"from sklearn.linear\_model import LinearRegression\n",**

**"\n",**

**"# Sample data (replace with your dataset)\n",**

**"X = np.array([1, 2, 3, 4, 5]).reshape(-1, 1) # Features\n",**

**"y = np.array([2, 4, 5, 4, 5]) # Target variable\n",**

**"\n",**

**"# Split the data into training and testing sets\n",**

**"X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)\n",**

**"\n",**

**"# Create and train a linear regression model\n",**

**"model = LinearRegression()\n",**

**"model.fit(X\_train, y\_train)\n",**

**"\n",**

**"# Make predictions on the test data\n",**

**"y\_pred = model.predict(X\_test)\n",**

**"\n",**

**"# Display the model's coefficients and intercept\n",**

**"print(\"Model Coefficients:\", model.coef\_)\n",**

**"print(\"Model Intercept:\", model.intercept\_)\n",**

**"\n",**

**"# Display the predicted output\n",**

**"print(\"Predicted Output:\", y\_pred)"**

**],**

**"metadata": {**

**"colab": {**

**"base\_uri": "https://localhost:8080/"**

**},**

**"id": "mqANVlb5Wson",**

**"outputId": "6743098d-8f9d-458d-90c3-91e9a6385f5c"**

**},**

**"execution\_count": 8,**

**"outputs": [**

**{**

**"output\_type": "stream",**

**"name": "stdout",**

**"text": [**

**"Model Coefficients: [0.68571429]\n",**

**"Model Intercept: 1.7714285714285714\n",**

**"Predicted Output: [3.14285714]\n"**

**]**

**}**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"source": [**

**"# \*\*Evaluation\*\*\n",**

**"\n",**

**"Evaluation:To evaluate the demand for a product, you can create a program that takes in data related to the product and provides an output indicating the level of demand. Here's a simplified Python program that you can use as a starting point. Please note that this is a basic example, and a real-world application would require more sophisticated data analysis and modeling."**

**],**

**"metadata": {**

**"id": "2GaRUgTZYa1q"**

**}**

**},**

**{**

**"cell\_type": "code",**

**"source": [**

**"# Import necessary libraries\n",**

**"import pandas as pd\n",**

**"\n",**

**"# Sample data for product demand evaluation\n",**

**"data = {\n",**

**" 'Price': [10, 15, 20, 25, 30],\n",**

**" 'Advertising': [100, 80, 60, 40, 20],\n",**

**" 'Competitor\_Price': [12, 14, 18, 22, 28],\n",**

**" 'Demand': [500, 450, 350, 300, 200]\n",**

**"}\n",**

**"\n",**

**"# Create a DataFrame from the data\n",**

**"df = pd.DataFrame(data)\n",**

**"\n",**

**"# Function to evaluate product demand\n",**

**"def evaluate\_demand(product\_data):\n",**

**" price = product\_data['Price']\n",**

**" advertising = product\_data['Advertising']\n",**

**" competitor\_price = product\_data['Competitor\_Price']\n",**

**"\n",**

**" # You can customize this formula based on your specific needs\n",**

**" demand = 1000 - (price \* 20) + (advertising \* 5) - (competitor\_price \* 10)\n",**

**"\n",**

**" return demand\n",**

**"\n",**

**"# Apply the evaluation function to each row in the DataFrame\n",**

**"df['Estimated\_Demand'] = df.apply(evaluate\_demand, axis=1)\n",**

**"\n",**

**"# Display the DataFrame with estimated demand\n",**

**"print(df)"**

**],**

**"metadata": {**

**"colab": {**

**"base\_uri": "https://localhost:8080/"**

**},**

**"id": "WiKPasogYFBa",**

**"outputId": "02c3dee9-f577-410f-924e-6439cac3b785"**

**},**

**"execution\_count": 9,**

**"outputs": [**

**{**

**"output\_type": "stream",**

**"name": "stdout",**

**"text": [**

**" Price Advertising Competitor\_Price Demand Estimated\_Demand\n",**

**"0 10 100 12 500 1180\n",**

**"1 15 80 14 450 960\n",**

**"2 20 60 18 350 720\n",**

**"3 25 40 22 300 480\n",**

**"4 30 20 28 200 220\n"**

**]**

**}**

**]**

**},**

**{**

**"cell\_type": "markdown",**

**"source": [**

**"This program uses a basic formula to estimate demand based on price, advertising, and competitor pricing. You can customize the formula and add more complex factors as needed for your specific product. The output will be a DataFrame with the estimated demand for each set of product data.\n",**

**"\n",**

**"Keep in mind that in a real-world scenario, you'd likely have more extensive data and might use machine learning or statistical models for more accurate demand forecasting."**

**],**

**"metadata": {**

**"id": "W7C4s4l3ZmRN"**

**}**

**},**

**{**

**"cell\_type": "markdown",**

**"source": [**

**"# \*\*7.Conclusion\*\*\n",**

**" Product demand prediction using machine learning is a powerful tool that can enhance a company's ability to manage inventory effectively, reduce costs, and meet customer expectations. It leverages the capabilities of advanced algorithms and vast datasets to improve forecasting accuracy.\n",**

**"\n",**

**"In the following discussion, we can delve deeper into the techniques, algorithms, and best practices for product demand prediction with machine learning."**

**],**

**"metadata": {**

**"id": "INFl5Y03aCUq"**

**}**

**}**

**]**

**}**