E:\sales.py

```python
import pandas as pd
import numpy as np
import matplotlib.colors as col
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
import datetime
from pathlib import Path
import random
from sklearn.preprocessing import MinMaxScaler
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
from sklearn.ensemble import RandomForestRegressor
from xgboost.sklearn import XGBRegressor
from sklearn.model_selection import KFold, cross_val_score, train_test_split
import keras
from keras.layers import Dense
from keras.models import Sequential
from keras.callbacks import EarlyStopping
from keras.utils import np_utils
from keras.layers import LSTM
import statsmodels.tsa.api as smt
import statsmodels.api as sm
from statsmodels.tools.eval_measures import rmse
import pickle
import warnings
warnings.filterwarnings("ignore", category=FutureWarning)
dataset = pd.read_csv('../input/demand-forecasting-kernels-only/sample_submission.csv')
df = dataset.copy()
df.head()
def load_data(file_name):
return pd.read_csv(file_name)
df_s.tail()
df_s['sales'].describe()
df_s['sales'].plot()
layout = (1, 2)
raw = plt.subplot2grid(layout, (0 ,0))
law = plt.subplot2grid(layout, (0 ,1))
years = y_df['sales'].plot(kind = "bar",color = 'mediumblue', label="Sales",ax=raw, figsize=(12, 5))
months = m_df['sales'].plot(marker = 'o',color = 'darkorange', label="Sales", ax=law)
years.set(xlabel = "Years",title = "Distribution of Sales Per Year")
months.set(xlabel = "Months", title = "Distribution of Sales Per Mounth")
sns.despine()
plt.tight_layout()
years.legend()
months.legend()
def sales_per_store(data): sales_by_store = data.groupby('store')['sales'].sum().reset_index()
fig, ax = plt.subplots(figsize=(8,6))
sns.barplot(sales_by_store.store, sales_by_store.sales, color='darkred')ax.set(xlabel = "Store Id", ylabel = "Sum of Sales", title = "Total Sales Per Store")
return sales_by_store
```