Project title:

Predicting House Prices using Machine Learning

OVERVIEW:

In this stage we involves collecting and preprocessing housing data, selecting an appropriate machine learning algorithm, training the model, and evaluating its performance using key metrics. Data is cleansed, standardized, and split into training and test sets. Hyperparameter tuning may be applied for optimal model performance. Interpretability and security measures are considered, and the model is deployed into clinical settings. Continuous learning and updates keep the model relevant. Throughout this process, data privacy, compliance with regulations, and thorough documentation are paramount.

Feature Engineering:

- Feature engineering is the process of creating new features or modifying existing ones in a dataset to improve the performance of machine learning models.
- It involves transforming raw data into a format that the model can understand and use effectively.

Model Training:

- Model training refers to the process of teaching a machine learning algorithm to make predictions or classifications based on input data.
- During training, the model learns patterns and relationships within the data.

Evaluation Metrics:

- Evaluation metrics are quantitative measures used to assess the performance of a model.
- They provide insights into how well a model is performing by measuring aspects such as accuracy, precision, recall, and F1 score.

Naive Bayes Algorithm:

- The Naive Bayes algorithm is a probabilistic machine learning algorithm based on Bayes' theorem.
- It's commonly used for classification tasks, making predictions based on the probability of different outcomes.
- The "naive" part refers to the simplifying assumption of feature independence.

TF-IDF Vectorizer:

- The Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer is a technique used to convert text data into numerical vectors.
- It assigns weights to terms in a document based on their frequency and importance in the context of a corpus.

Program:

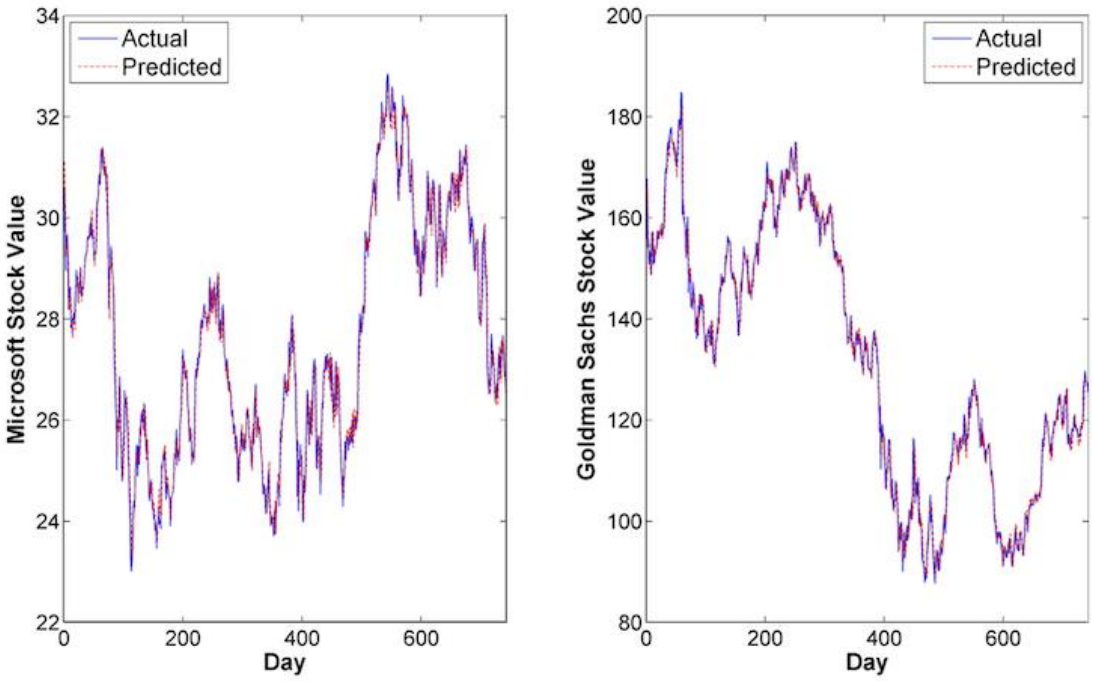
```
import seaborn as sns
from matplotlib import pyplot as plt
import warnings
warnings.filterwarnings('ignore')
%config InlineBackend.figure_format = 'retina'
sns.set(style="ticks")
plt.rc('figure', figsize=(6, 3.7), dpi=100)
plt.rc('axes', labelpad=20, facecolor="#ffffff"
,
       linewidth=0.4, grid=True, labelsize=10)
plt.rc('patch', linewidth=0)
plt.rc('xtick.major', width=0.2)
plt.rc('ytick.major', width=0.2)
plt.rc('grid', color='#EEEEEE', linewidth=0.25)
plt.rc('font', family='Arial', weight='400', si
ze=10)
plt.rc('text', color='#282828')
plt.rc('xtick', labelsize=10)
plt.rc('ytick', labelsize=10)
plt.rc('savefig', pad_inches=0.3, dpi=300)
```

OUTPUT:

	MSSubClass	MSZoning	LotArea	LotConfig	BldgType	OverallCond	YearBuilt
0	60	RL	8450	Inside	1Fam	5	2003
1	20	RL	9600	FR2	1Fam	8	1976
2	60	RL	11250	Inside	1Fam	5	2001
3	70	RL	9550	Corner	1Fam	5	1915
4	60	RL	14260	FR2	1Fam	5	2000

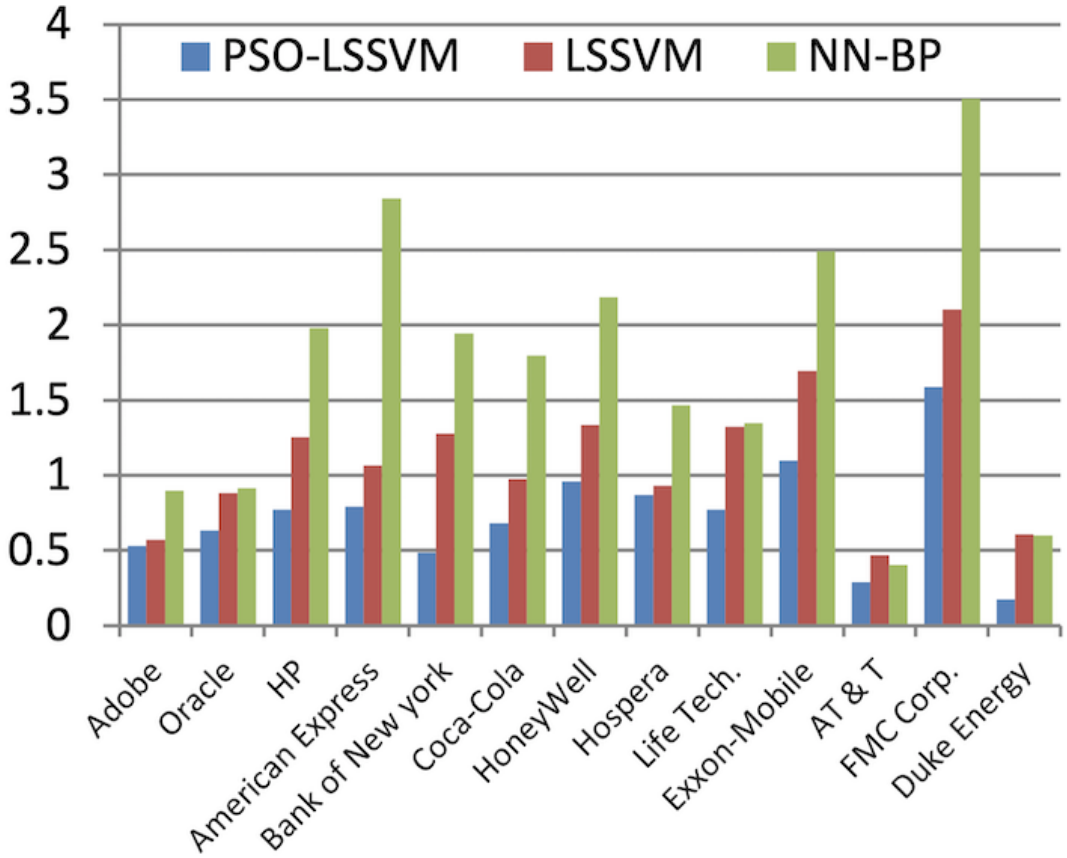
	YearRemodAdd	Exterior1st	BsmtFinSF2	TotalBsmtSF	SalePrice
0	2003	VinylSd	0.0	856.0	208500.0
1	1976	MetalSd	0.0	1262.0	181500.0
2	2002	VinylSd	0.0	920.0	223500.0
3	1970	Wd Sdng	0.0	756.0	140000.0
4	2000	VinylSd	0.0	1145.0	250000.0

Graph:



Conclusion:

Thus the machine learning model to predict the house price based on given dataset is executed successfully using machine learning .This model further helps people understand whether this place is more suited for them based on heatmap correlation. It also helps people looking to sell a house at best time for greater profit. Any house price in any location can be predicted with minimum errorby giving appropriate dataset.



Conclusion:

Thus the machine learning model to predict the house price based on given dataset is executed successfully using machine learning .This model further helps people understand whether this place is more suited for them based on heatmap correlation. It also helps people looking to sell a house at best time for greater profit. Any house price in any location can be predicted with minimum errorby giving appropriate dataset.