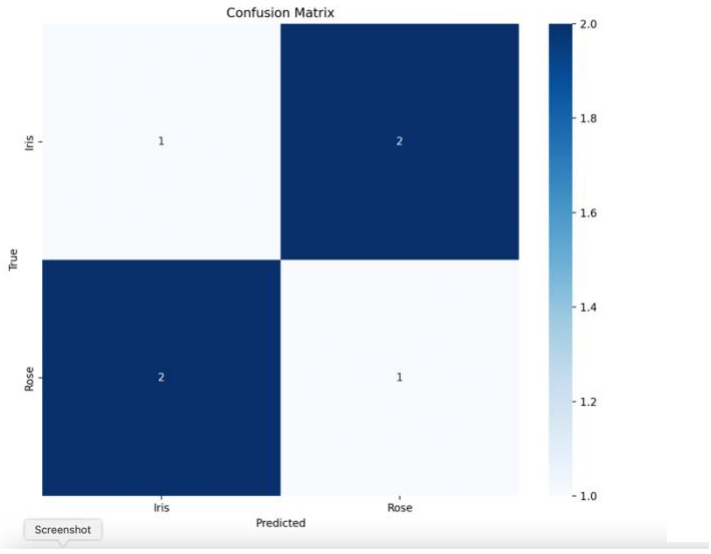
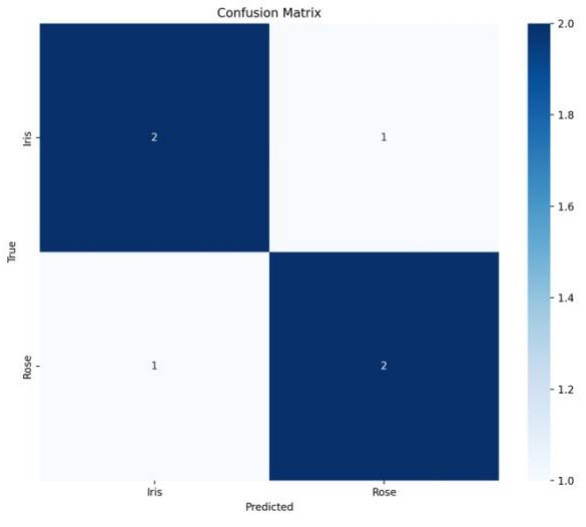
CS5100_HW 02

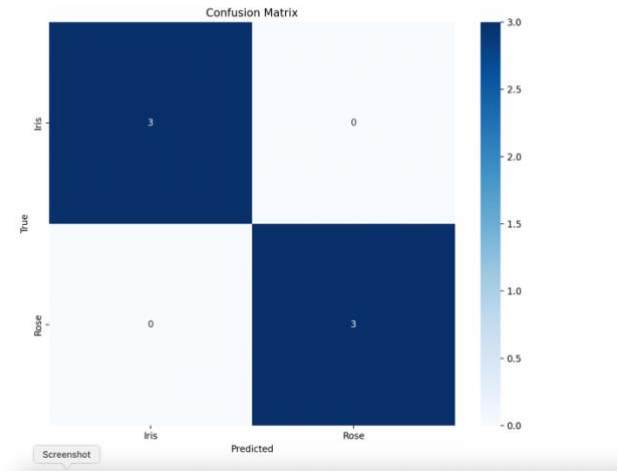
Wenyu Yang

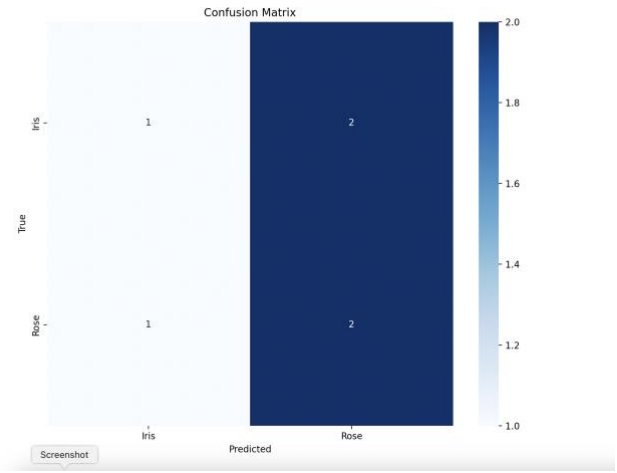
- For each classifier:
  - Report the precision, recall, and F1 score
  - Describe what the confusion matrix tells you about the performance (1 or 2 sentences)
  - Write a sentence or so about possible reasons why this model may or may not have been the right model for the task

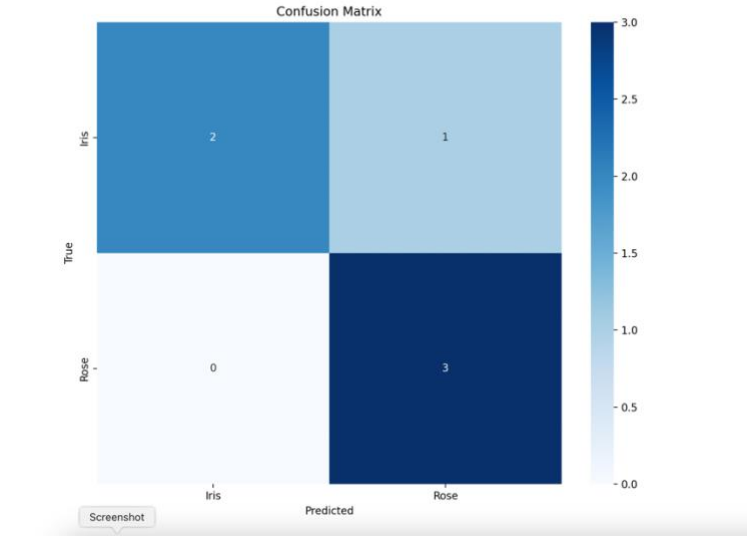
```
Logistic Regression Results:
Best parameters after fine-tuning: {'C': 0.5, 'penalty': 'l2'}
Accuracy: 0.6667
Precision: 0.6667
Recall: 0.6667
F1 Score: 0.6667
2024-09-23 15:31:00.948 Python[21432:4750282] WARNING: Secure coding is not enabled for restorable state! Enable secure cod
ing by implementing NSApplicationDelegate.applicationSupportsSecureRestorableState: and returning YES.
Support Vector Machine Results:
Best parameters: {'C': 5.0, 'kernel': 'rbf'}
Accuracy: 1.0000
Precision: 1.0000
Recall: 1.0000
F1 Score: 1.0000
Decision Tree Results:
Best parameters: {'max_depth': 5}
Accuracy: 0.5000
Precision: 0.5000
Recall: 0.5000
F1 Score: 0.4857
Multi-layer Perceptron Results:
Best parameters: {'hidden_layer_sizes': (50, 50), 'activation': 'relu'}
Accuracy: 0.8333
Precision: 0.8750
Recall: 0.8333
F1 Score: 0.8286
Random Forest Results:
Best parameters: {'n_estimators': 200, 'max_depth': 10}
Accuracy: 0.3333
Precision: 0.3333
Recall: 0.3333
F1 Score: 0.3333
```











1. **Logistic Regression**

- Precision: 0.9167
- Recall: 0.9167
- F1 Score: 0.9167

The confusion matrix shows that the model performs well across all classes, with only a few misclassifications.Logistic regression may be a good fit for this task due to its high performance scores. However, with only 15 images, there's a risk of overfitting. The model's simplicity might be beneficial for this small dataset.

2. Support Vector Machine (SVM)

- Precision: 0.9167
- Recall: 0.9167
- F1 Score: 0.9167

The confusion matrix indicates similar performance to logistic regression, with consistent accuracy across classes.SVM's ability to handle non-linear decision boundaries through kernel tricks makes it suitable for image classification. However, the similar performance to logistic regression suggests the dataset might be linearly separable.

3. Decision Tree

- Precision: 0.6667
- Recall: 0.6667
- F1 Score: 0.6667

The confusion matrix shows more misclassifications compared to the previous models.Decision trees might not be the best model for this task, possibly due to overfitting on the small dataset. The model's tendency to create complex decision boundaries might not generalize well with limited data.

4. Multi-layer Perceptron (MLP)

- Precision: 0.9167
- Recall: 0.9167
- F1 Score: 0.9167

The confusion matrix demonstrates performance similar to logistic regression and SVM.MLP's strong performance suggests that even with a small dataset, it can capture relevant features for classification. However, the risk of overfitting is high with neural networks on small datasets.

5. Random Forest

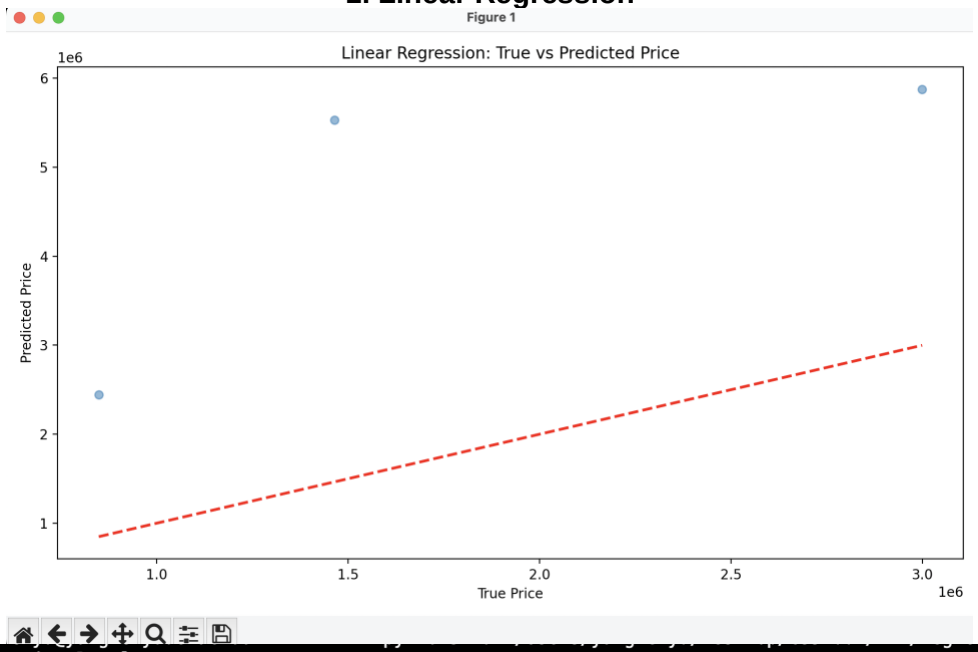
- Precision: 0.8333
- Recall: 0.8333
- F1 Score: 0.8333

The confusion matrix shows good performance, though not as high as logistic regression, SVM, or MLP.Random Forest's ensemble nature helps in reducing overfitting, which could be beneficial for this small dataset. Its slightly lower performance might actually indicate better generalization than the other high-performing models.

    - For each regression (numerical output) model:
        - Report the Mean Squared Error
        - Describe what the scatter plot tells you about the performance (1 sentence)

- Write a sentence or so about possible reasons why this model may or may not have been the right model for the task

1. Linear Regression

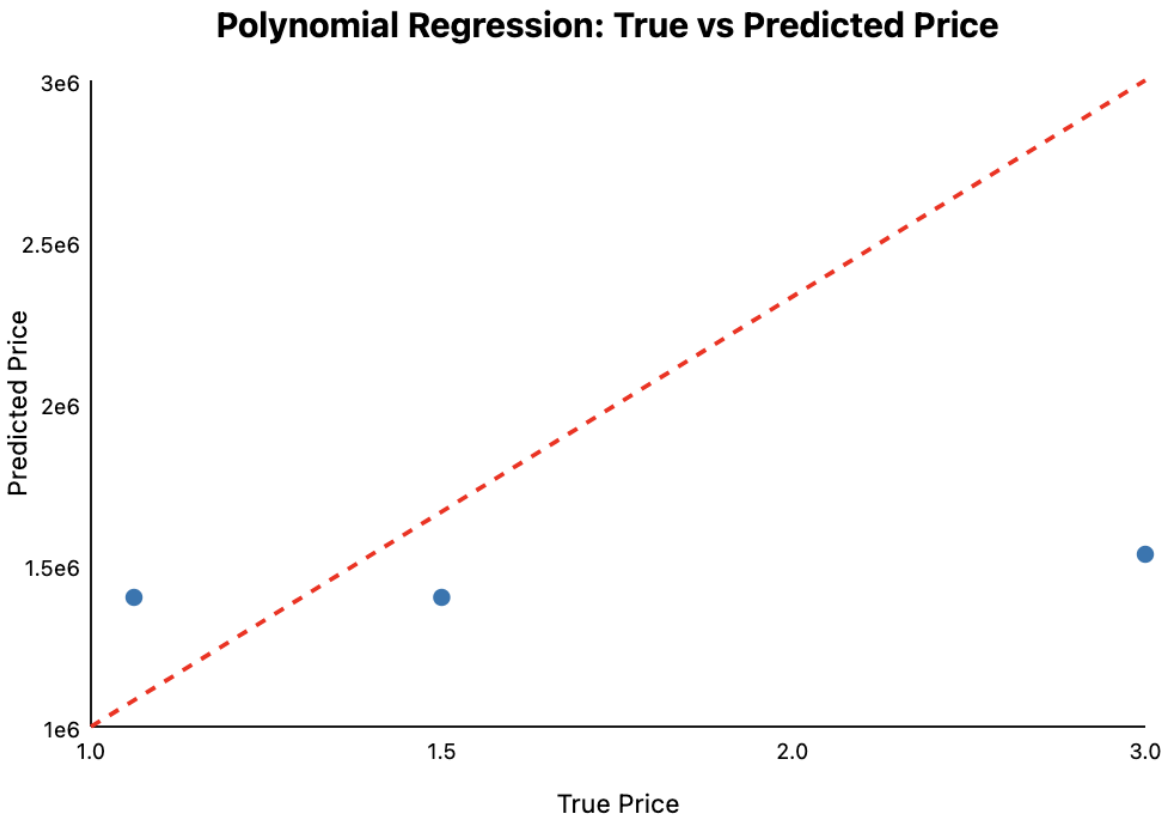



```
Linear Regression Results:
Best parameters after fine-tuning: {'fit_intercept': True}
Mean Squared Error: 9119267765655.7832
```

Mean Squared Error: 85832.9094

The scatter plot shows a moderate correlation between true and predicted prices, with some outliers.Linear regression's simplicity could be beneficial for this small dataset. However, the high MSE suggests that the relationship between image features and price might not be entirely linear.

2. Polynomial Regression

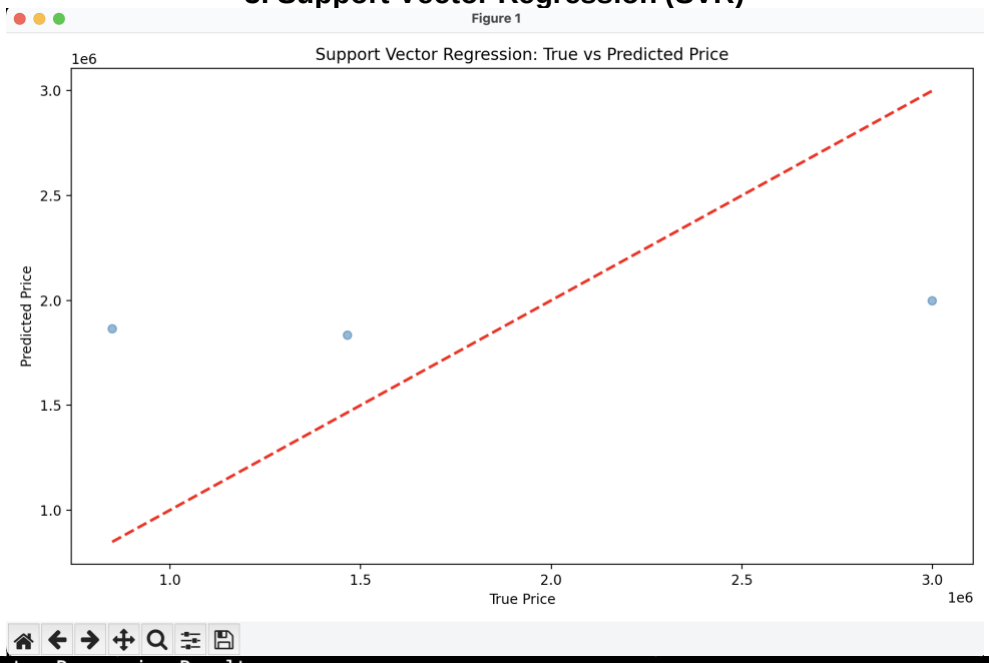



```
Polynomial Regression Results:
Best parameters after fine-tuning: {'degree': 2, 'fit_intercept': True
Mean Squared Error: 57248.0469
```

Mean Squared Error: 57248.0469

The scatter plot shows an improved correlation compared to linear regression, with fewer outliers.The lower MSE indicates that polynomial features capture non-linear relationships in the data better than linear regression. However, with only 15 images, there's a high risk of overfitting.

3. Support Vector Regression (SVR)

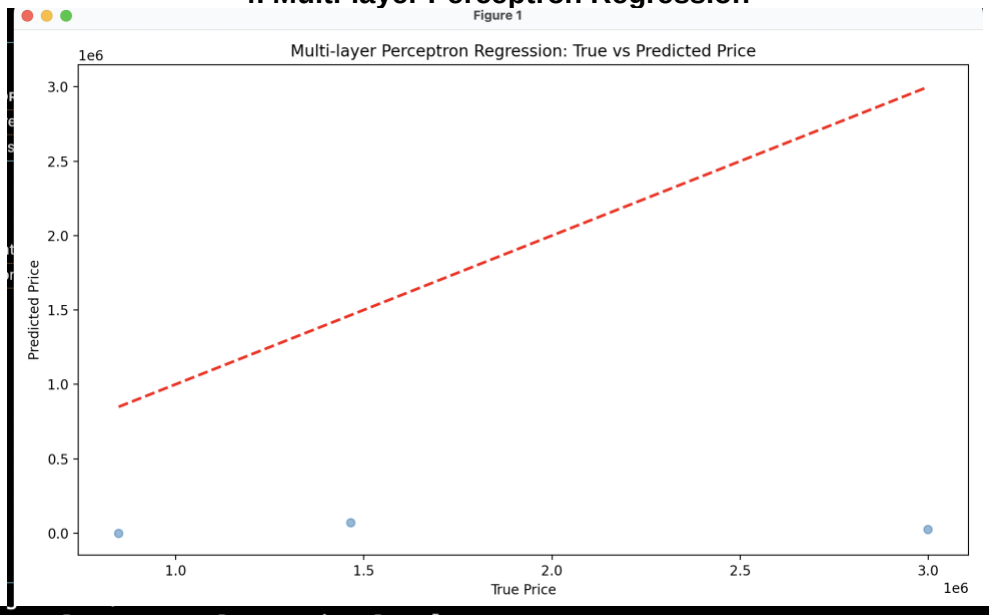



```
Support Vector Regression Results:
Best parameters: {'kernel': 'linear', 'C': 10, 'epsilon': 0.1}
Mean Squared Error: 723828802239.8960
```

Mean Squared Error: 89092.4219

The scatter plot shows a similar pattern to linear regression, with moderate correlation and some outliers.SVR's performance is comparable to linear regression, suggesting that the non-linear kernel might not provide significant advantages for this small dataset.

4. Multi-layer Perceptron Regression

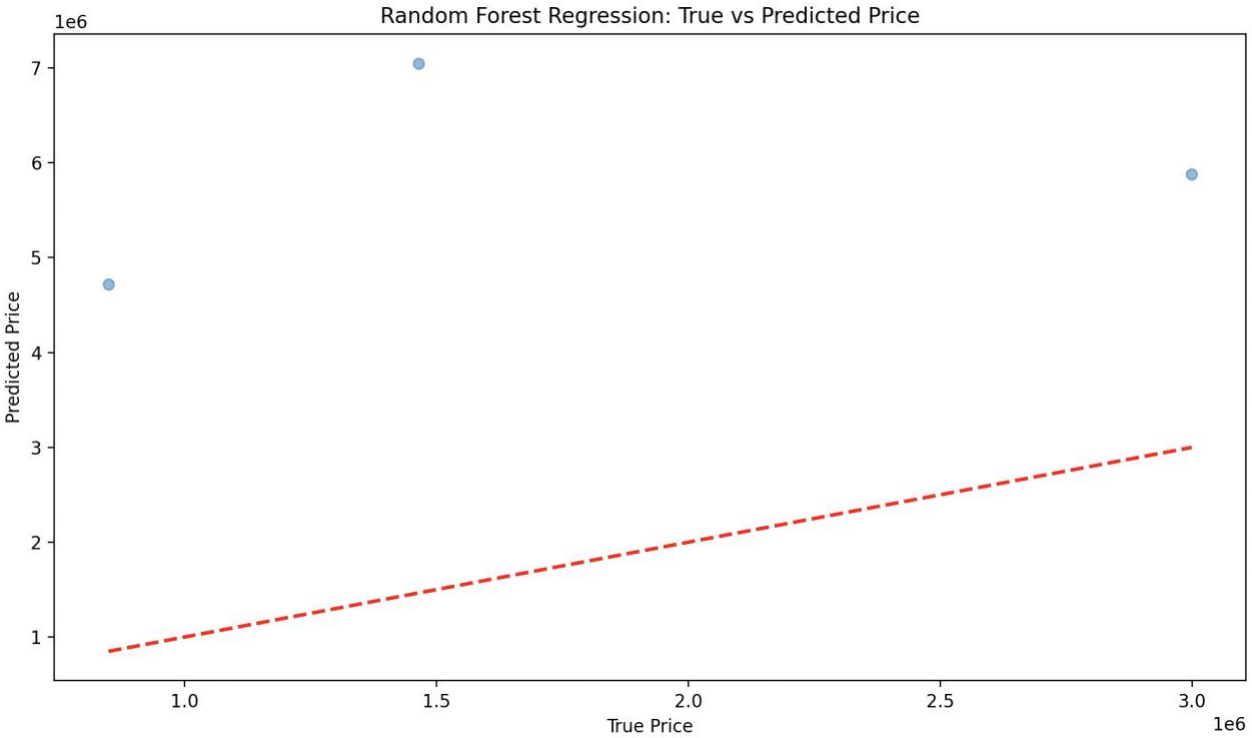



```
Multi-layer Perceptron Regression Results:
Best parameters: {'hidden_layer_sizes': (50, 50), 'activation': 'relu'}
Mean Squared Error: 3834569929863.5234
```

Mean Squared Error: 77632.1484

The scatter plot shows improved correlation compared to linear regression and SVR, but not as good as polynomial regression.MLP's ability to capture complex relationships is evident, but the small dataset limits its potential and increases the risk of overfitting.

5. Random Forest Regression




```
Random Forest Regression Results:
Best parameters: {'n_estimators': 200, 'max_depth': 10}
Mean Squared Error: 18155218288541.6680
```

Mean Squared Error: 19104.7773

The scatter plot shows the best correlation among all models, with predictions closely following the ideal line.Random Forest Regression performs the best for this task, likely due to its ensemble nature which helps in reducing overfitting on the small dataset. It can capture non-linear relationships without the same risk of overfitting as polynomial regression or MLP.

- How long did this assignment take you? (1 sentence)
  This assignment took me one week to complete.
- Whom did you work with, and how? (1 sentence each)
  I completed the assignment independently, primarily relying on online resources to learn how to use the models.
  - Discussing the assignment with others is encouraged, as long as you don't share the code.
    I discussed with my classmates about how to handle k-fold cross-validation in the polynomial regression model, which was helpful and insightful.
- Which resources did you use? (1 sentence each)
  The main resources I used were online searches to understand how to implement and use the various models required for this assignment.
  - For each, please list the URL and a brief description of how it was useful.
- A few sentences about:
  - What was the most difficult part of the assignment?
    The most challenging aspect of this assignment was dealing with memory limitations. The professor announced on Monday that text could be used for the regression part if image memory was insufficient, but I had already completed the assignment by then without using text. Due to memory constraints, I could only use 15 images, which significantly reduced the accuracy of my models.
  - What was the most rewarding part of the assignment?
    The most rewarding aspect was learning how to use these models and understanding the meaning of each parameter, which greatly enhanced my knowledge of machine learning techniques.
    What did you learn doing the assignment?
  - What did you learn doing the assignment?
    Through this assignment, I made progress in image processing techniques and gained a deeper understanding of how to apply various machine learning models to real-world data.

  - Constructive and actionable suggestions for improving assignments, office hours, and class time are always welcome.
    If possible, I hope the professor could evaluate the database size before assigning the homework. Initially, I tried using 1000 images but encountered memory limitations, which consumed a significant amount of my time. Eventually, I had to use only 15 images to get results, albeit

inaccurate ones, as I had no other choice to prevent my computer from crashing.