

Mini Project: Watermelon Price Prediction

Nicole Perez

July 27, 2024





Introduction

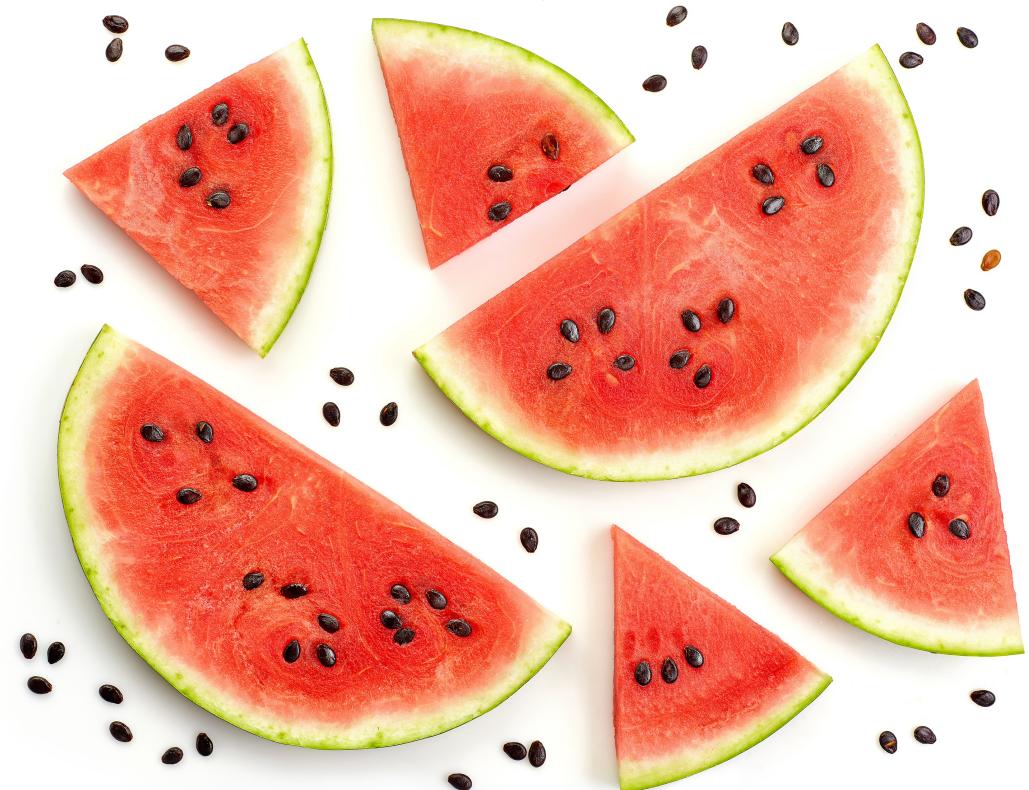
- ▶ Predicting the price of watermelons based on their **weight** that can help farmers or agriculture stakeholders and sellers optimise their pricing strategies. In this project, I will use linear regression to predict watermelon prices from their weights.

Dataset overview

- ▶ Each row: watermelon
- ▶ Each columns: weight and price

```
df = pd.read_csv('melon.csv')
df.head()

...    Weight  Price
0        1     50
1        2    100
2        3    140
3        4    180
4        5    220
```



Dataset Wrangling

```
#Importing modules  
(e.g.import pandas as pd )
```

```
# Load the dataset  
df = pd.read_csv(melon.csv')
```

```
# Check for missing values  
print(df.isnull().sum())
```

Exploratory Data Analysis (EDA)

- understand the structure of our data set
- identify missing or null values in our data set
- find the patterns and relationships between variables

Exploratory Data Analysis (EDA)

```
# Check summary  
df.info
```

```
#EDA  
df.info
```

			DataFrame.info of	Weight	Price
0	1	50			
1	2	100			
2	3	140			
3	4	180			
4	5	220			
5	6	250			
6	7	300			
7	8	340			
8	9	380			
9	10	390	>		

```
# Check null values  
df.isnull().sum()
```

```
df.isnull().sum()
```

```
...    Weight      0  
      Price      0  
      dtype: int64
```

Exploratory Data Analysis (EDA)

```
# Check dataset structure  
df.shape
```

```
df.shape
```

```
... (10, 2)
```

Exploratory Data Analysis (EDA)

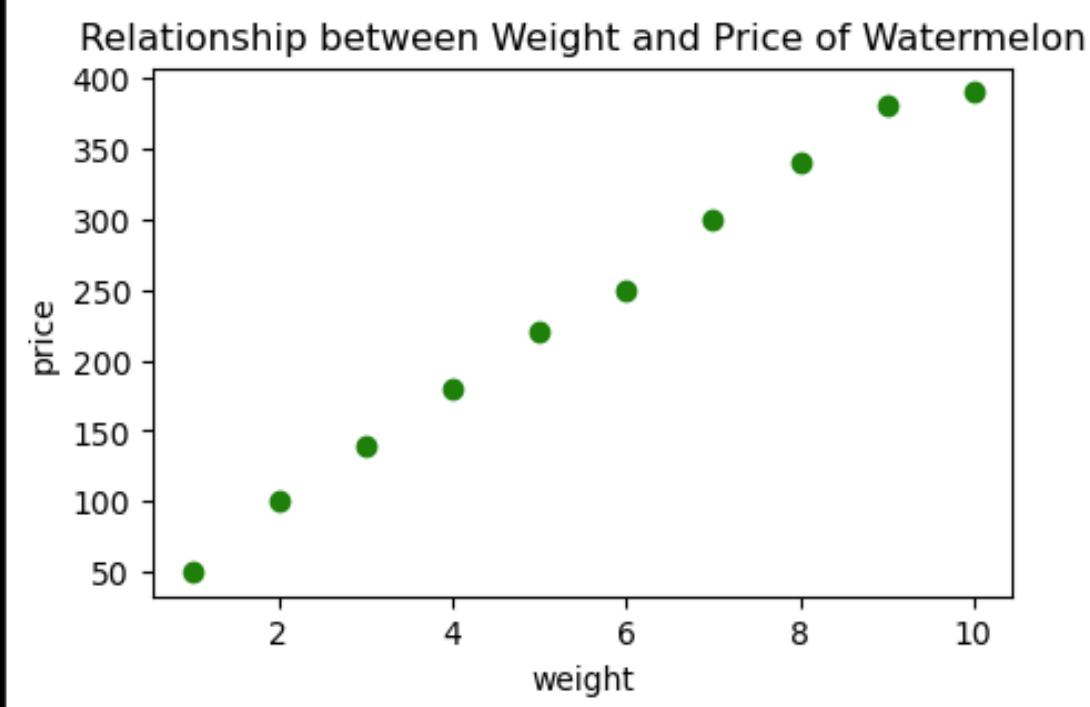
```
# Check data issues, for instance, outliers  
Df.describe()
```

▷ df.describe()

	Weight	Price
count	10.00000	10.000000
mean	5.50000	235.000000
std	3.02765	117.968922
min	1.00000	50.000000
25%	3.25000	150.000000
50%	5.50000	235.000000
75%	7.75000	330.000000
max	10.00000	390.000000

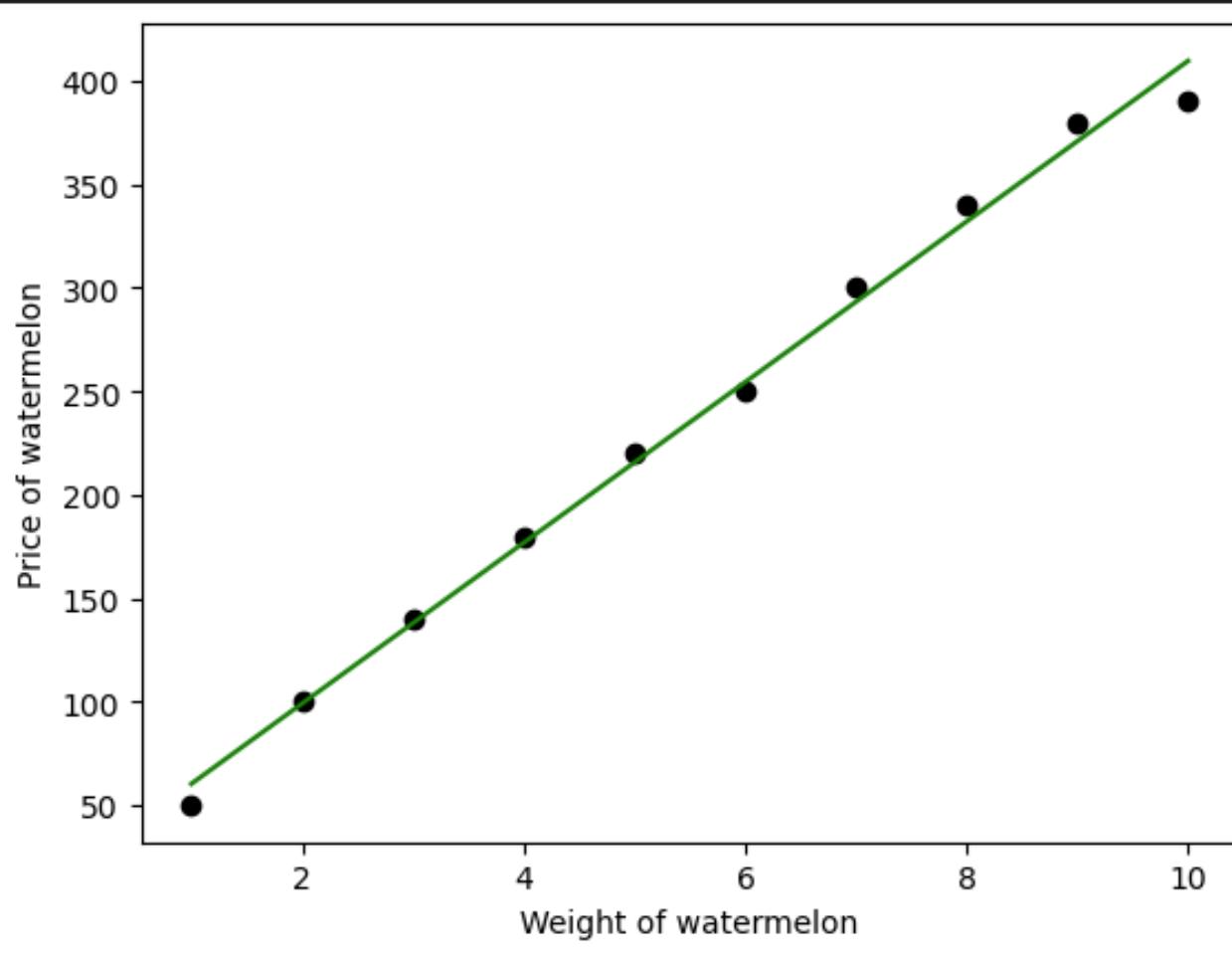
Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA)



- ▶ `plt.figure(figsize=(5,3))`
- ▶ `plt.scatter(df['Weight'],df['Price'],c='green')`
- ▶ `plt.title("Relationship between Weight and Price of Watermelon")`
- ▶ `plt.xlabel('weight')`
- ▶ `plt.ylabel('price')`
- ▶ `plt.show()`

Single linear regression



- ▶

```
#import scikit learn  
from sklearn import linear_model
```
- ▶

```
# Linear regressor  
regr = linear_model.LinearRegression()  
regr.fit(weight, price)
```
- ▶

```
a = regr.predict(weight)
```

Model Evaluation

- Mean Squared Error (MSE)
- R-squared (R^2)

```
from sklearn.metrics import mean_squared_error, r2_score  
  
print("Mean Squared Error ", mean_squared_error(price, z))  
print("r2 coefficient", r2_score(price,z))
```

```
Mean Squared Error  74.06060606060605  
r2 coefficient 0.9940869775600315
```

These metrics suggest that the model is performing very well, with high accuracy in its predictions and relatively low error.

Project Summary: Watermelon Price Prediction Model

- ▶ In this project, developed a **linear regression model** to predict watermelon prices based on their weight. The model's performance metrics indicate a high level of accuracy and reliability, making it a valuable tool for future price predictions.



Project Summary: Watermelon Price Prediction Model

Key Highlights:

- ▶ **Model Accuracy:** The regression model demonstrates high accuracy, evidenced by a low Mean Squared Error (MSE) and a high R-squared (R^2) value.
- ▶ **Reliability:** The strong performance metrics suggest that the model can reliably predict watermelon prices based on weight, even for unseen data.
- ▶ **Future Enhancements:** Potential improvements include incorporating additional features and exploring alternative regression techniques to further enhance model performance.

