Polynomial Linear Regression

Import Libraries

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures
```

Importing Data and Data processing

```
df = pd.read_csv('Position_Salaries.csv')
x = df.iloc[:, 1:-1].values
y = df.iloc[:, -1].values
```

Training the Linear Regression model on the whole dataset

```
lr = LinearRegression()
lr.fit(x,y)
```

```
▾ LinearRegression
LinearRegression()
```

Training the Polynomial Regression model on the whole dataset

```
pr = PolynomialFeatures(degree = 4)
x_poly = pr.fit_transform(x)
lr2 = LinearRegression()
lr2.fit(x_poly, y)
```

```
▾ LinearRegression
LinearRegression()
```

Visualising the Linear Regression results

Suggested code may be subject to a license | abel-valle/ds-tools

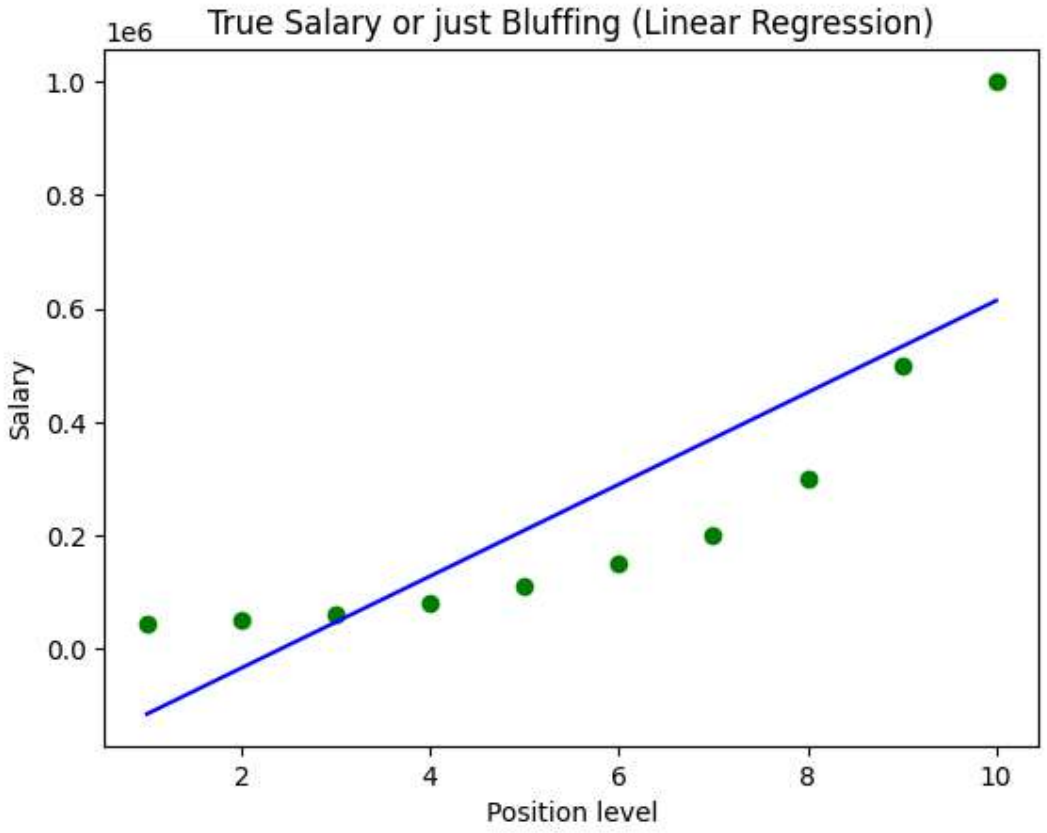
```
plt.scatter(x, y, color = 'green')
plt.plot(x, lr.predict(x), color = 'blue')
plt.title('True Salary or just Bluffing (Linear Regression)')
plt.xlabel('Position level')
```

```
plt.ylabel('Salary')
plt.show
```

matplotlib.pyplot.show
`def show(*args, **kwargs)`

/usr/local/lib/python3.10/dist-packages/matplotlib/pyplot.py
Display all open figures.

Parameters
----
block : bool, optional

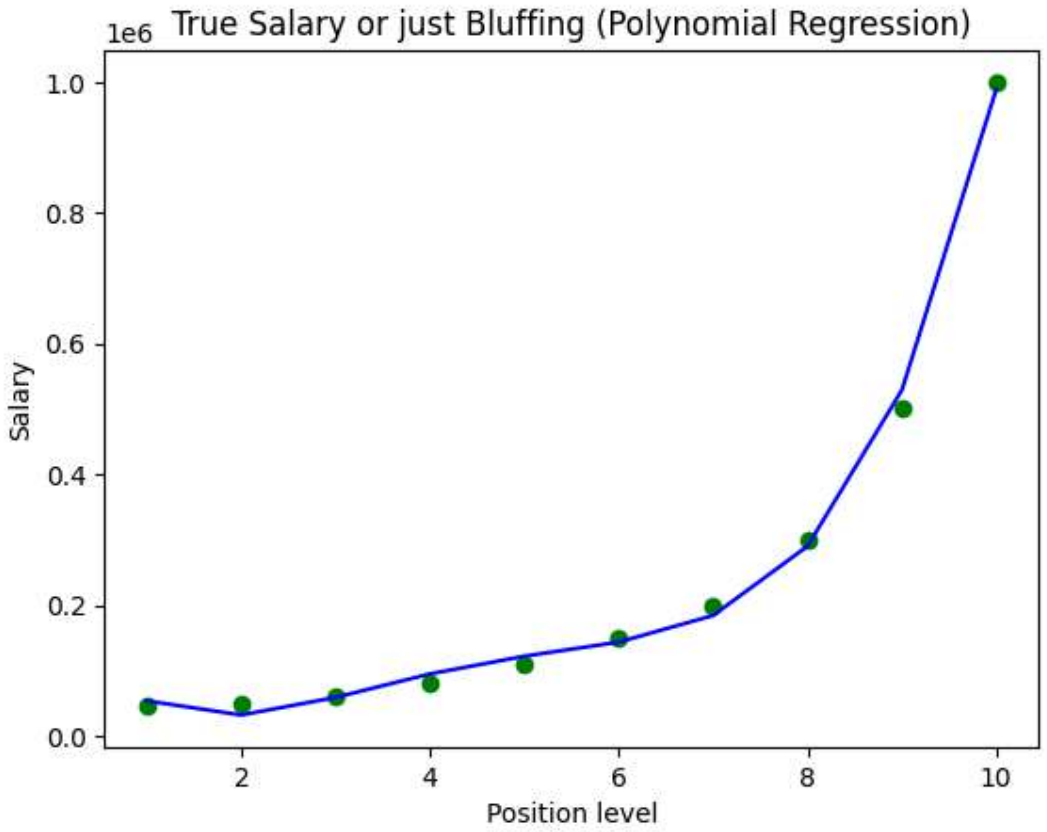


Visualising the Polynomial Regression results

```
plt.scatter(x, y, color = 'green')
plt.plot(x, lr2.predict(pr.fit_transform(x)), color = 'blue')
plt.title('True Salary or just Bluffing (Polynomial Regression)')
plt.xlabel('Position level')
plt.ylabel('Salary')
plt.show
```

```
matplotlib.pyplot.show
def show(*args, **kwargs)
```

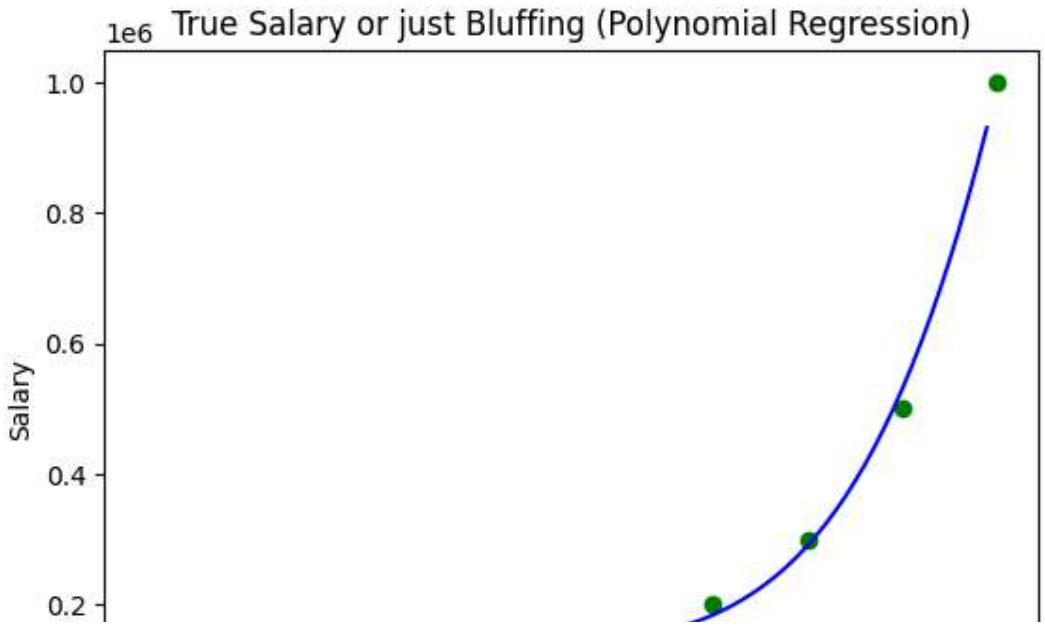
```
/usr/local/lib/python3.10/dist-packages/matplotlib/pyplot.py
Display all open figures.

Parameters
----------
block : bool, optional
```



Visualising The Polynomial Regression result with smoother curve

```
x_grid = np.arange(min(x), max(x), 0.1)
x_grid = x_grid.reshape((len(x_grid), 1))
plt.scatter(x, y, color = 'green')
plt.plot(x_grid, lr2.predict(pr.fit_transform(x_grid)), color = 'blue')
plt.title('True Salary or just Bluffing (Polynomial Regression)')
plt.xlabel('Position level')
plt.ylabel('Salary')
plt.show()
```

```
<ipython-input-14-5775815f47f6>:1: DeprecationWarning: Conversion of an array with ndim > 0 to
  x_grid = np.arange(min(x), max(x), 0.1)
```



Predicting a new result with Linear Regression

```
lr.predict([[6.5]])
```

```
array([330378.78787879])
```

Predicting a new result with Polynomial Regression

```
lr2.predict(pr.fit_transform([[6.5]]))
```

```
array([158862.45265155])
```