Convexity

BOND VALUATION AND ANALYSIS IN PYTHON

Joshua Mayhew
Options Trader

Plotting predicted vs. actual prices

```python
import numpy as np
import numpy_financial as npf
import pandas as pd
import matplotlib.pyplot as plt
```

```python
price = -npf.pv(rate=0.05, nper=20, pmt=5, fv=100)
price_up = -npf.pv(rate=0.06, nper=20, pmt=5, fv=100)
price_down = -npf.pv(rate=0.04, nper=20, pmt=5, fv=100)
duration = (price_down - price_up) / (2 * price * 0.01)
dollar_duration = duration * price * 0.01
print("Bond Price (USD): ", price)
print("Dollar Duration (USD): ", dollar_duration)
```

```
Bond Price (USD): 100.00
Dollar Duration (USD): 12.53
```

Plotting predicted vs. actual prices

```
bond_yields = np.arange(0, 10, 0.1)
bond = pd.DataFrame(bond_yields, columns=['bond_yield'])
bond['price'] = -npf.pv(rate=bond['bond_yield'] / 100, nper=20, pmt=5, fv=100)
```

```
    bond_yield       price
0          0.0  200.000000
1          0.1  196.978503
..         ...         ...
98         9.8   58.570780
99         9.9   57.997210

[100 rows x 2 columns]
```

Plotting predicted vs. actual prices

```
bond['yield_change'] = bond['bond_yield'] - 5
```

```
    bond_yield       price  yield_change
0          0.0  200.000000          -5.0
1          0.1  196.978503          -4.9
2          0.2  194.013231          -4.8
..         ...         ...           ...
97         9.7   59.153044           4.7
98         9.8   58.570780           4.8
99         9.9   57.997210           4.9

[100 rows x 3 columns]
```

Plotting predicted vs. actual prices

$$\text{Price Change} = -100 \times \text{Dollar Duration} \times \Delta y$$

```
bond['price_change'] = -100 * dollar_duration * bond['yield_change'] / 100
```

```
    bond_yield       price  yield_change  price_change
0          0.0  200.000000          -5.0     62.650619
1          0.1  196.978503          -4.9     61.397607
..         ...         ...           ...           ...
98         9.8   58.570780           4.8    -60.144594
99         9.9   57.997210           4.9    -61.397607

[100 rows x 4 columns]
```

Plotting predicted vs. actual prices

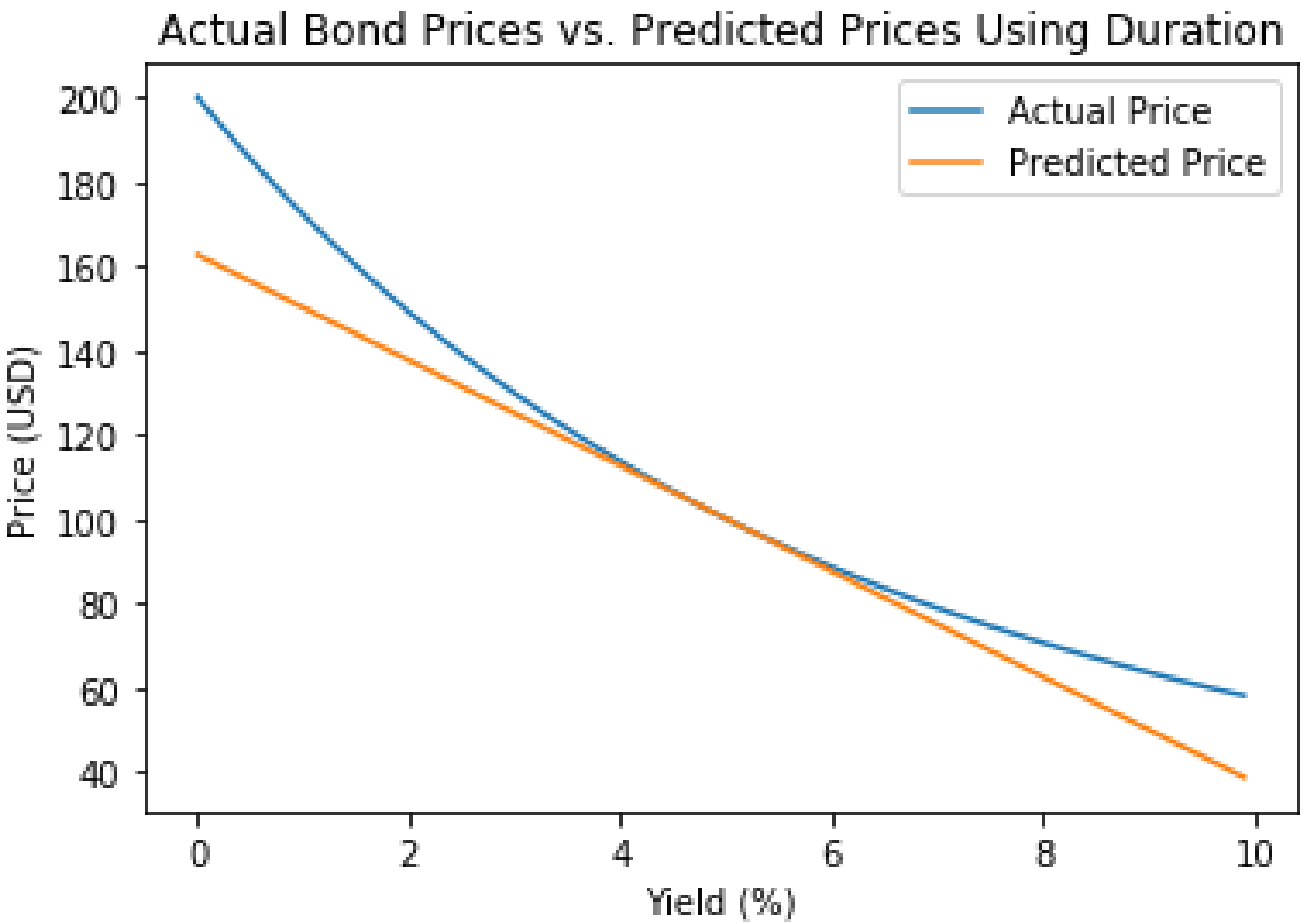
```
bond['predicted_price'] = price + bond['price_change']
```

```
   bond_yield       price  yield_change  price_change  predicted_price
0         0.0  200.000000          -5.0     62.650619       162.650619
1         0.1  196.978503          -4.9     61.397607       161.397607
2         0.2  194.013231          -4.8     60.144594       160.144594
..        ...         ...           ...           ...              ...
97        9.7   59.153044           4.7    -58.891582        41.108418
98        9.8   58.570780           4.8    -60.144594        39.855406
99        9.9   57.997210           4.9    -61.397607        38.602393

[100 rows x 5 columns]
```

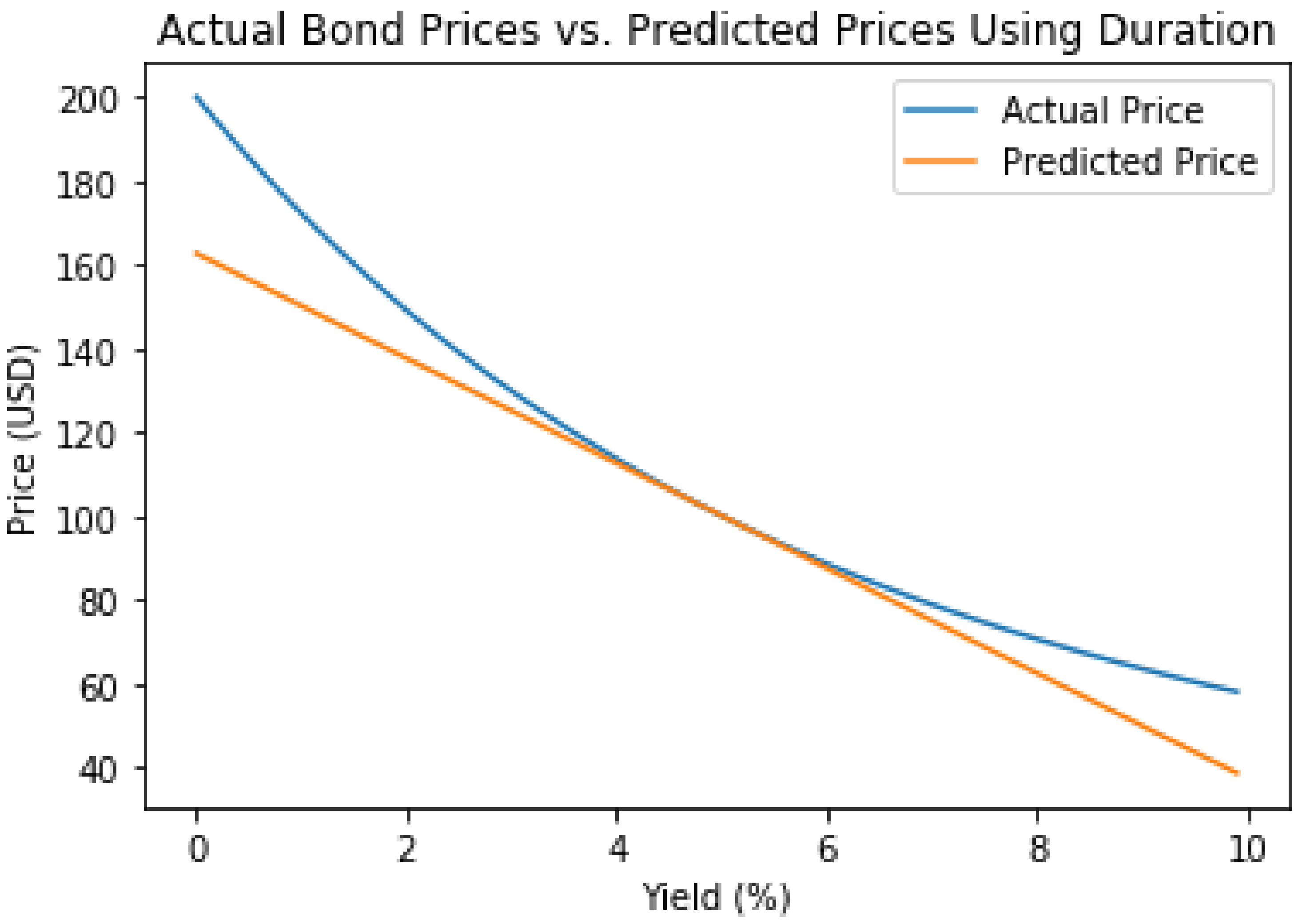
Plotting predicted vs. actual prices

```
plt.plot(bond['bond_yield'], bond['price'])
plt.plot(bond['bond_yield'], bond['predicted_price'])
plt.xlabel('Yield (%)')
plt.ylabel('Price (USD)')
plt.title("Actual Bond Prices vs.
    Predicted Prices Using Duration")
plt.legend(["Actual Price", "Predicted Price"])
plt.show()
```



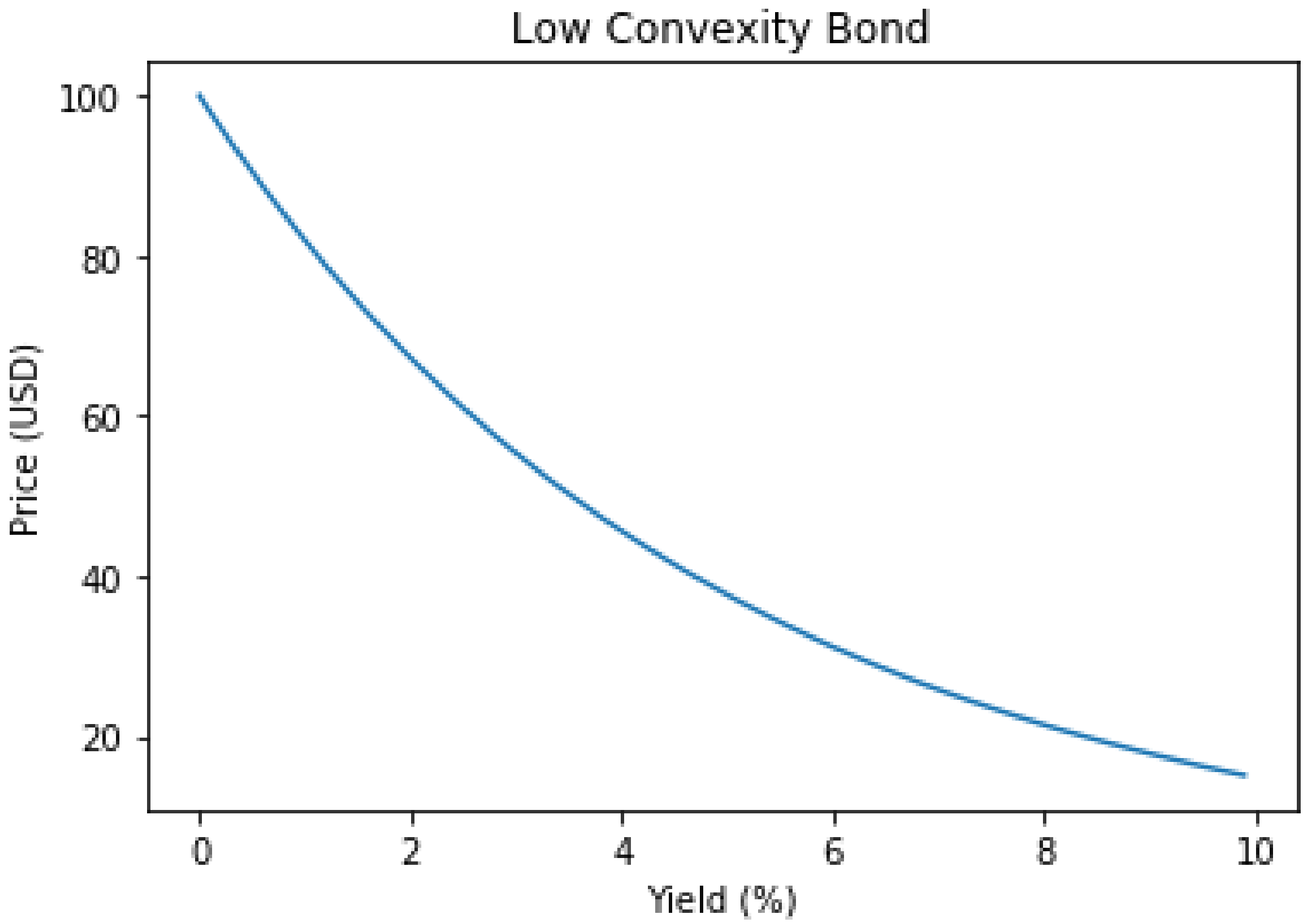
Limitations of duration

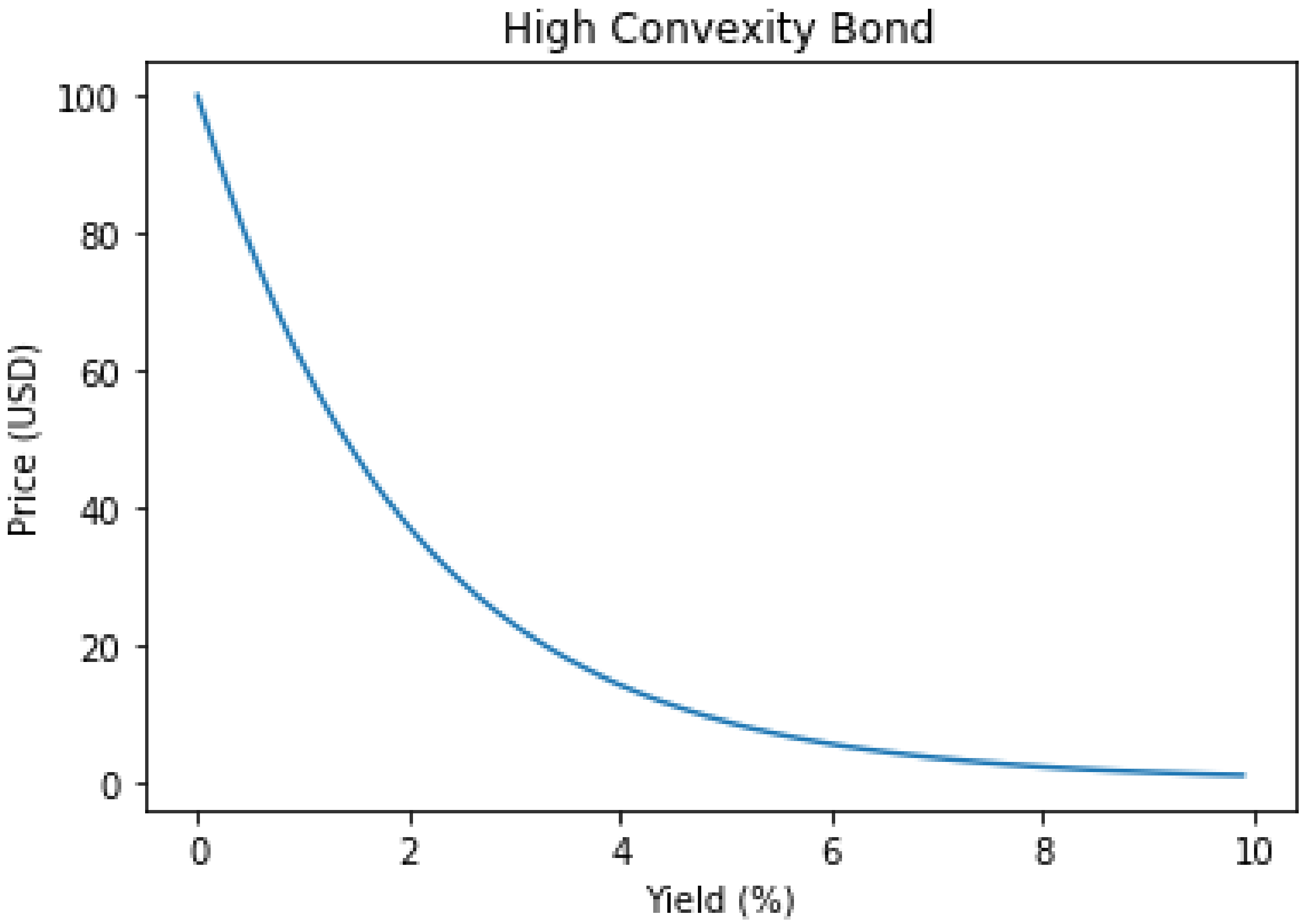
- Duration is a linear measure
- Bond prices are convex
- Duration is accurate for small yield changes only



What is convexity?

- Measures the curvature of bond prices
- Used to improve bond price prediction and risk measurement
- Higher convexity = more curved price/yield relationship





Convexity formula

We will use a simplified formula for convexity:

$$Convexity = \frac{P_{down} + P_{up} - 2{\times}P}{P \times (\Delta y)^2}$$

- P_{down} = Bond price at 1% lower yield
- P_{up} = Bond price at 1% higher yield
- $2 \times P$ = Double the bond price at current yield
- $(\Delta y)^2$ = Change in yield squared (we will use 1% ˆ 2)

Convexity example

10 year bond, 5% annual coupon, 4% yield to maturity, what is its convexity?

$$Convexity = \frac{P_{down} + P_{up} - 2 \times P}{P \times (\Delta y)^2}$$

```
price = -npf.pv(rate=0.04, nper=10, pmt=5, fv=100)
price_up = -npf.pv(rate=0.05, nper=10, pmt=5, fv=100)
price_down = -npf.pv(rate=0.03, nper=10, pmt=5, fv=100)
convexity = (price_down + price_up - 2 * price) / (price * 0.01 ** 2)
print("Convexity: ", convexity)
```

```
Convexity:  77.56
```

Summary

- Duration is a linear measure
- Bond prices are curved
- Duration is accurate for small yield changes only
- Convexity measures this curvature

$$Convexity = \frac{P_{down} + P_{up} - 2{\times}P}{P \times (\Delta y)^2}$$

Let's practice!

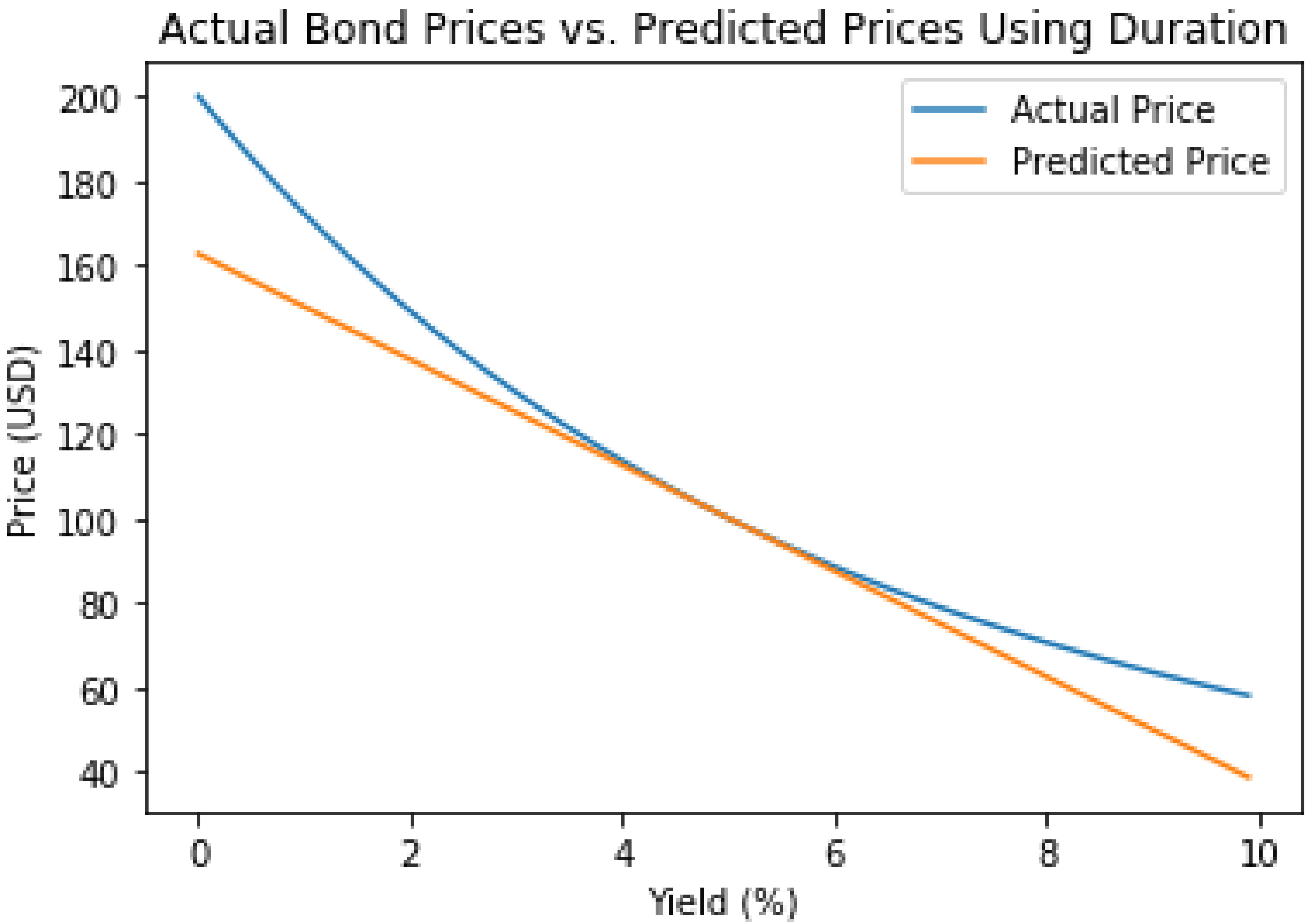
Factors affecting convexity

BOND VALUATION AND ANALYSIS IN PYTHON

Joshua Mayhew
Options Trader

Convexity is a benefit

- Bond prices rise more when yields decrease, fall less when yields increase



Investigating convexity

- Compare the convexity of two bonds directly
- Plot a graph of factor against convexity
- Directly examine the curvature of the price/yield relationship

Coupon vs. convexity

- 10 year bonds with 5% yield, first pays no coupon, second pays 10% coupon

```
price_1 = -npf.pv(rate=0.05, nper=10, pmt=0, fv=100)
price_up_1 = -npf.pv(rate=0.06, nper=10, pmt=0, fv=100)
price_down_1 = -npf.pv(rate=0.04, nper=10, pmt=0, fv=100)
convexity_1 = (price_down_1 + price_up_1 - 2 * price_1) / (price_1 * 0.01 ** 2)
```

```
price_2 = -npf.pv(rate=0.05, nper=10, pmt=10, fv=100)
price_up_2 = -npf.pv(rate=0.06, nper=10, pmt=10, fv=100)
price_down_2 = -npf.pv(rate=0.04, nper=10, pmt=10, fv=100)
convexity_2 = (price_down_2 + price_up_2 - 2 * price_2) / (price_2 * 0.01 ** 2)
```

```
print("Low Coupon Bond Convexity: ", convexity_1)
print("High Coupon Bond Convexity: ", convexity_2)
```

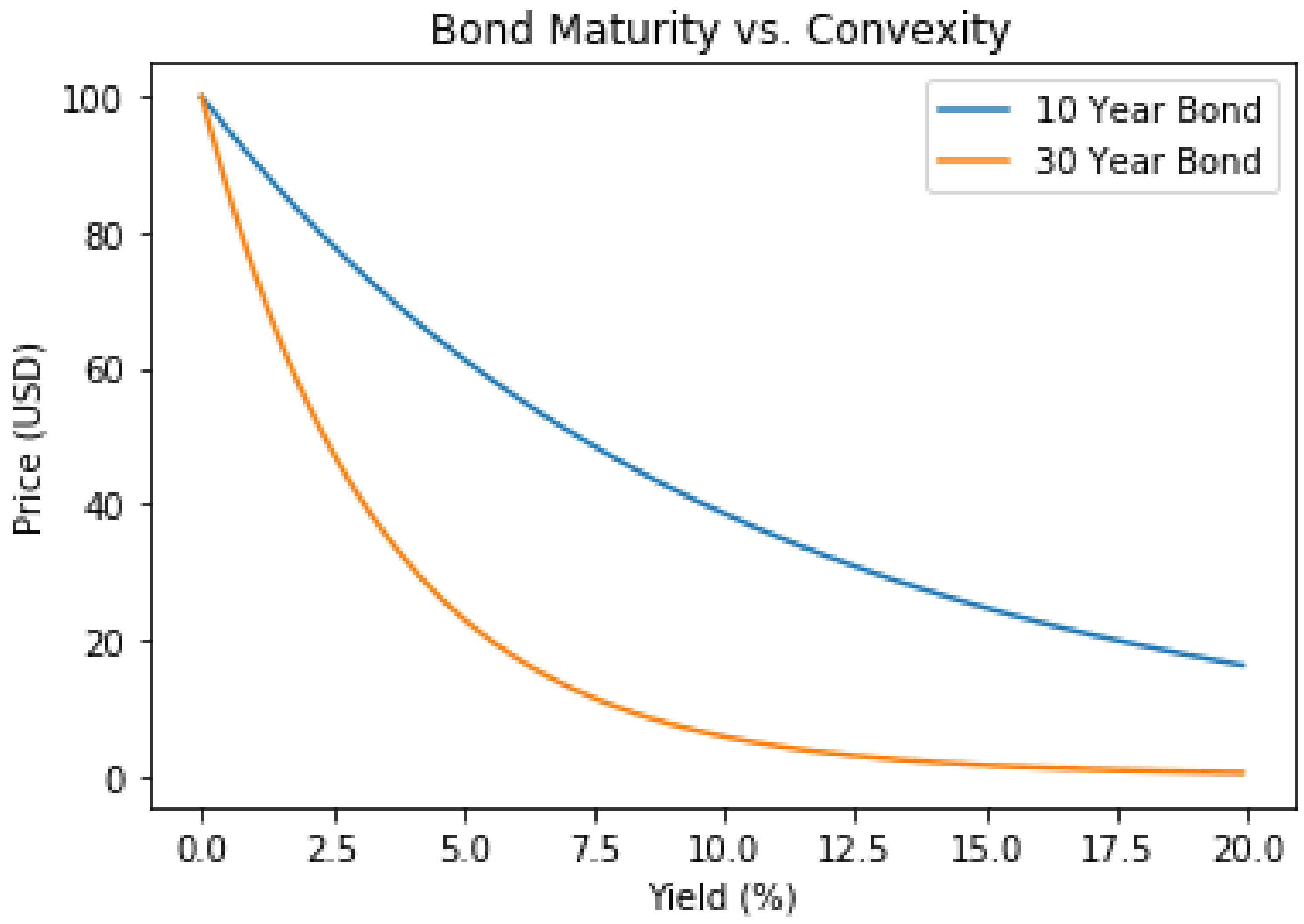
```
Low Coupon Bond Convexity:  99.89
High Coupon Bond Convexity:  64.09
```

Maturity vs. convexity

```
bond_yields = np.arange(0, 20, 0.1)
bond = pd.DataFrame(bond_yields, columns=['yield'])
```

```
bond['price_10y'] = -npf.pv(rate=bond['yield'] / 100,
    nper=10, pmt=0, fv=100)
bond['price_30y'] = -npf.pv(rate=bond['yield'] / 100,
    nper=30, pmt=0, fv=100)
```

```
plt.plot(bond['yield'], bond['price_10y'])
plt.plot(bond['yield'], bond['price_30y'])
plt.xlabel('Yield (%)')
plt.ylabel('Price (USD)')
plt.title('Bond Maturity vs. Convexity')
plt.legend(["10 Year Bond", "30 Year Bond"])
plt.show()
```



Yield vs. convexity

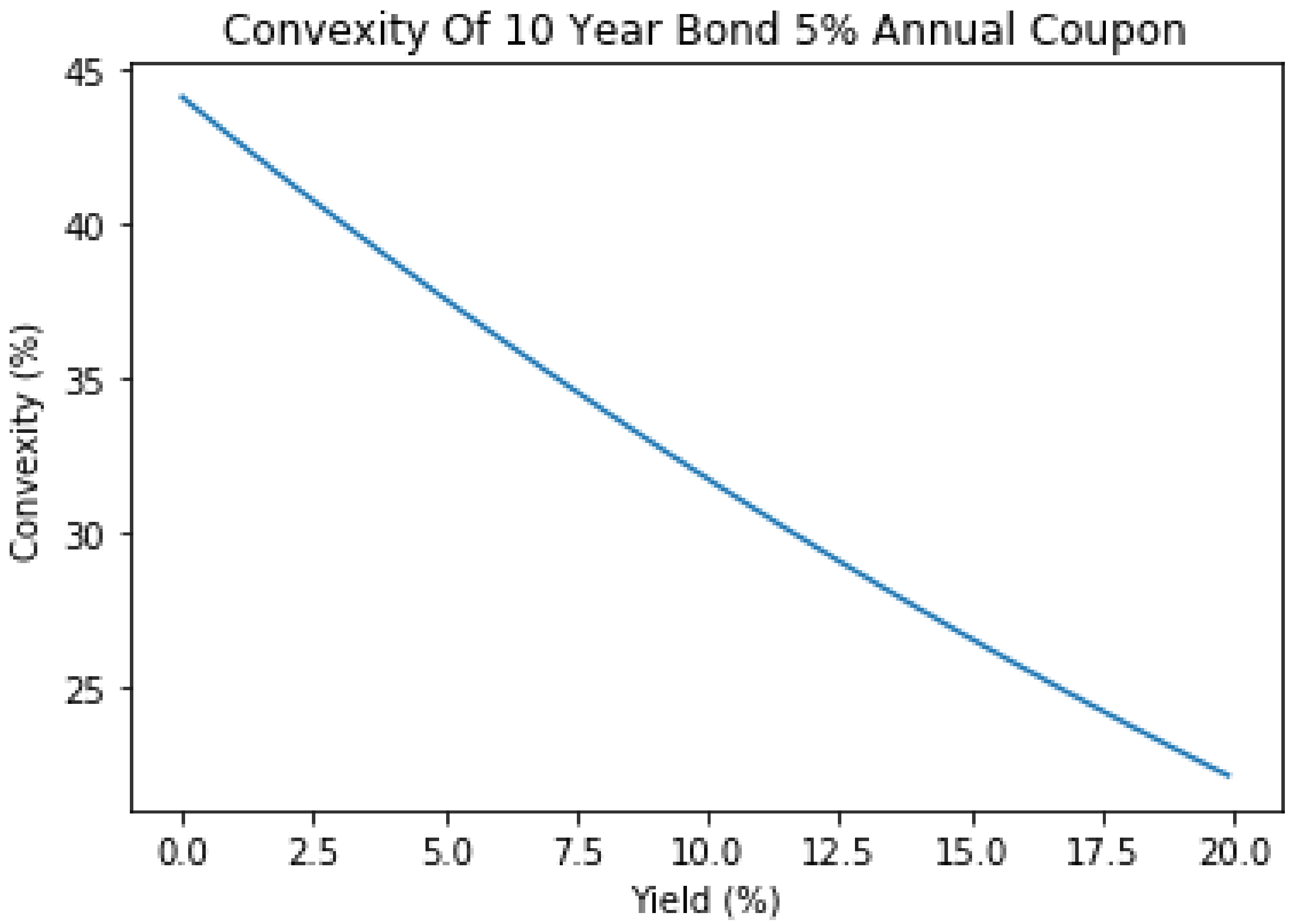
```
bond_yields = np.arange(0, 20, 0.1)
bond = pd.DataFrame(bond_yields, columns=['bond_yield'])
bond['price'] = -npf.pv(rate=bond['bond_yield'] / 100,
    nper=10, pmt=5, fv=100)
```

```
bond['price_up'] = -npf.pv(rate=bond['bond_yield'] / 100
    + 0.01, nper=10, pmt=5, fv=100)
```

```
bond['price_down'] = -npf.pv(rate=bond['bond_yield'] / 100
    - 0.01, nper=10, pmt=5, fv=100)
```

```
bond['convexity'] = (bond['price_down'] + bond['price_up']
- 2 * bond['price']) / (bond['price'] * 0.01 ** 2)
```

```
plt.plot(bond['bond_yield'], bond['convexity'])
plt.xlabel('Yield (%)')
plt.ylabel('Convexity (%)')
plt.title("Convexity Of 10 Year Bond 5% Annual Coupon")
plt.show()
```



Summary

- Positive convexity is a benefit:
  - Lose less when yields rise, make more when yields fall

- Convexity increases when a bond has a:
  - Higher maturity
  - Lower coupon
  - Lower yield

Let's practice!

Dollar convexity and bond price prediction

BOND VALUATION AND ANALYSIS IN PYTHON

Joshua Mayhew
Options Trader

Dollar convexity

- Convexity = % change in duration for 1% change in yields
- Dollar convexity = $ change in duration for 1% change in yields:

$$\text{Dollar Convexity} = \text{Convexity} \times \text{Bond Price} \times 0.01^2$$

Dollar convexity example

- 10 year bond with 3% coupon, 5% yield and USD 100 face value, what is its dollar convexity?

```
price = -npf.pv(rate=0.05, nper=10, pmt=3, fv=100)
price_up = -npf.pv(rate=0.06, nper=10, pmt=3, fv=100)
price_down = -npf.pv(rate=0.04, nper=10, pmt=3, fv=100)
convexity = (price_down + price_up - 2 * price) / (price * 0.01 ** 2)
dollar_convexity = convexity * price * 0.01 ** 2
print("Dollar Convexity: ", dollar_convexity)
```

```
Dollar Convexity:  0.69
```

The convexity adjustment

- Convexity can be used to improve bond price prediction
- Convexity adjustment = how much bond prices change due to convexity

$$\text{Convexity Adjustment} = 0.5 \times \text{Dollar Convexity} \times 100^2 \times (\Delta y)^2$$

Convexity adjustment example

- 10 year bond with 3% coupon, 5% yield and USD 100 face value
- What is its convexity adjustment?

```
price = -npf.pv(rate=0.05, nper=10, pmt=3, fv=100)
price_up = -npf.pv(rate=0.06, nper=10, pmt=3, fv=100)
price_down = -npf.pv(rate=0.04, nper=10, pmt=3, fv=100)
convexity = (price_down + price_up - 2 * price) / (price * 0.01 ** 2)
dollar_convexity = convexity * price * 0.01 ** 2
convexity_adjustment = 0.5 * dollar_convexity * 100 ** 2 * 0.01 ** 2
print("Convexity Adjustment: ", convexity_adjustment)
```

```
Convexity Adjustment:  0.35
```

Combining duration and convexity

- Predicting price changes from duration alone:

$$\text{Price Change} = -100 \times \text{Dollar Duration} \times \Delta y$$

- Predicting price changes from both duration and convexity:

$$\text{Price Change} = -100 \times \text{Dollar Duration} \times \Delta y \; + \; \text{Convexity Adjustment}$$

$$= -100 \times \text{Dollar Duration} \times \Delta y \; + \; 0.5 \times \text{Dollar Convexity} \times 100^2 \times (\Delta y)^2$$

- Combining duration and convexity improves price estimation

Duration and convexity example

- 10 year bond, 3% coupon, 5% yield, USD 100 face value:

```
price = -npf.pv(rate=0.05, nper=10, pmt=3, fv=100)
price_up = -npf.pv(rate=0.06, nper=10, pmt=3, fv=100)
price_down = -npf.pv(rate=0.04, nper=10, pmt=3, fv=100)
```

```
duration = (price_down - price_up) / (2 * price * 0.01)
dollar_duration = duration * price * 0.01
```

```
convexity = (price_down + price_up - 2 * price) / (price * 0.01 ** 2)
dollar_convexity = convexity * price * 0.01
convexity_adjustment = dollar_convexity * 100 ** 2 * 0.01 ** 2
```

```
combined_prediction = -100 * dollar_duration * 0.01 + convexity_adjustment
print("Predicted Price Change: ", combined_prediction)
```

```
Predicted Price Change:  -6.64
```

Let's practice!

BOND VALUATION AND ANALYSIS IN PYTHON

Congratulations!

BOND VALUATION AND ANALYSIS IN PYTHON

Joshua Mayhew
Options Trader

Chapter 1 summary - time value of money

- Simple and compound interest
- Future value and `fv()` function
- Compounding frequencies
- Other functions: `nper()` , `rate()` , `pmt()`

Chapter 2 summary - bond prices and yields

- Present value and `pv()` function
- Zero coupon bonds
- Coupon paying bonds
- Bond prices vs. bond yields

Chapter 3 summary - duration

- Duration introduction
- Factors affecting duration
- Dollar duration and DV01
- Creating a duration neutral portfolio
- Predicting bond prices using duration

Chapter 4 summary - convexity

- Convexity introduction
- Factors affecting convexity
- Dollar convexity and convexity adjustment
- Combining duration and convexity

Congratulations!

BOND VALUATION AND ANALYSIS IN PYTHON