Options Payoff Calculator

October 30, 2020

1 Objective

In this exercise a simple options payoff calculator is coded and exercised for different positions on stocks, calls and puts. From the theory of options, we know the payoffs for individual calls and puts. Now, on the basis of individual positions on the options and assets, we compute the overall payoff from the strategy. We can also see from these charts the payoffs arising from various spreads and combinations.

Front-end Dashboard This dashboard is available at here. We input the parameters for creating the strategy, and display total payoffs from the strategy, as well as individual payoffs.

2 Theory

Calls are options that gives the owner a right to buy the stock (underlying asset) at a specific price (strike) on/by a specific date (expiration date). Call owners (call long) have a right and not an obligation to buy the underlying stock, however, the writer of a call (call short) are obliged to deliver if the counter-party wishes. Puts are options that gives the owner a right to sell the stock (underlying asset) at a specific price (strike) on/by a specific date (expiration date). Put owners (put long) have a right and not an obligation to sell the underlying stock, however, the writer of a put (put short) are obliged to deliver if the counter-party wishes.

The payoffs arising from puts and calls in both positions are enumerated below. c and p are premiums paid to the writer of the call and put respectively to earn the forestated rights.

- Call Long: payoff $= max(S_T - K - c, -c)$, unlimited gains when ITM: $S_T > K$
- Call Short: payoff $= min(K - S_T + c, c)$, limited gains when OTM: $S_T < K$
- Put Long: payoff $= max(K - S_T - p, -p)$, unlimited gains when ITM: $S_T < K$
- Put Short: payoff $= min(S_T - K + p, p)$, limited gains when OTM: $S_T > K$

3 Algorithm

In order to obtain the payoffs from individual calls and puts positions one can use the above payoffs given above. However in order to generate the payoffs from multiple spreads and combinations, one would have to use the respective payoff charts. For instance, in order to implement a butterfly spread, one would have to take two long calls at different strike prices K_1 and K_3 and short a call of a median strike price K_2. Here the payoff table must be created for S_T in different ranges:

$S_T < K_1$, $K_1 < S_T < K_2$, $K_2 < S_T < K_3$ and $S_T > K_3$, and for each case one must sum up the individual payoffs arising from individual positions.

Here, we seek a smarter solution in trying to develop a common algorithm for various possible spreads, combinations as well as covered and uncovered positions in individual options. The key trick is to realize that payoffs would be the sum of individual positions irrespective of the strategy.

1. Take 3 calls, 3 puts and an underlying stock. Specify their individual prices, number of units purchased and the strike price in case of options. A negative number in units purchased means that one is going short on the particular option or stock.

2. Compute the payoffs arising from these individual options/stock. These are stored in individual position variables. The sum of all these would be the cost associated with getting into a contract.

3. Create a range of stock prices varying from +/- 100percent deviation from initial stock price. This is stored in an array and for all stock prices, the corresponding payoffs have to be calculated for each option/stock and summed up to obtain the total payoff.

4. For a call the payoff is computed as $max[(N_c \times (S_T - K) + c), c]$. We have incorporated the position into the sign of c. Now if this payoff value is positive, we seek to maximize this. However, if this is negative, then we wish to minimize the loss: $min[(N_c \times (S_T - K) + c), c]$.

5. Likewise for a put, the postive payoffs are maximized: $max[(N_p \times (K - S_T) + p), p]$ and negative payoffs are minimized $min[(N_p \times (K - S_T) + p), p]$.

This implemented in the python code shown below. A separate front-end calculator implemented using Tableau has been deployed in this webpage.

4 Coding the calculator

Importing libraries

[1]:
```
import pandas as pd
import numpy as np
import numpy.random
import scipy.stats as stats
import plotly.express as px
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(rc={'figure.figsize':(12,8)}, style = 'whitegrid')
import warnings
warnings.simplefilter('ignore')

df = pd.DataFrame()
```

Defining the `payoff_calculator` function This function takes in a combination of inputs regarding the units, prices and strike prices of options and the underlying asset. The costs of the strategy is first computed and the individual components' payoff are also computed, for various stock price points (+/- 100% stdev from the current stock price). Using these, the total payoff of the strategy is calculated and plotted.

[200]:
```python
def payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price, c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price, p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price):

    # payoffs from individual option positions
    c1_cost = -c1_units*c1_price
    c2_cost = -c2_units*c2_price
    c3_cost = -c3_units*c3_price
    p1_cost = -p1_units*p1_price
    p2_cost = -p2_units*p2_price
    p3_cost = -p3_units*p3_price
    stock_cost = - stock_units*stock_price
    # cost to enter strategy
    cost = c1_cost + c2_cost + c3_cost + p1_cost + p2_cost + p3_cost + stock_cost

    print("--------------------------  Strategy -------------------------------")
    print("----------------------------------------------------------------------")
    print("Options\t|\tUnits\t|\tPremium\t|\tStrike\t| Setup Cost")
    print("----------------------------------------------------------------------")
    print("Call 1\t|\t",c1_units,"\t|\t",c1_price,"\t|\t",c1_strike,"\t|\t",c1_cost)
    print("Call 2\t|\t",c2_units,"\t|\t",c2_price,"\t|\t",c2_strike,"\t|\t",c2_cost)
    print("Call 3\t|\t",c3_units,"\t|\t",c3_price,"\t|\t",c3_strike,"\t|\t",c3_cost)
    print("Put 1\t|\t",p1_units,"\t|\t",p1_price,"\t|\t",p1_strike,"\t|\t",p1_cost)
    print("Put 2\t|\t",p2_units,"\t|\t",p2_price,"\t|\t",p2_strike,"\t|\t",p2_cost)
    print("Put 3\t|\t",p3_units,"\t|\t",p3_price,"\t|\t",p3_strike,"\t|\t",p3_cost)
    print("Stock\t|\t",stock_units,"\t|\t",stock_price,"\t|\t","--","\t|\t",
          stock_cost)
    print("----------------------------------------------------------------------")
    print("Initial Setup Cost\t\t\t\t\t",cost)
```

```
    # initializing the matrix, whose columns are option payoffs and row sum␣
↪gives the total payoff.
    df = pd.DataFrame()
    frac = stock_price/1000
    df['Index'] = list(range(1,2002))
    df['stock_prices'] =  np.arange(0,2*stock_price+0.0001,frac)
    #df['stock_prices'] =  list(range(0,202,1))
    df['c1_payoff']=0.0
    df['c2_payoff']=0.0
    df['c3_payoff']=0.0
    df['p1_payoff']=0.0
    df['p2_payoff']=0.0
    df['p3_payoff']=0.0
    df['stock_payoff']=0.0


    # computing the payoffs for calls
    for i in range(len(df)):
        if(c1_units>0):
            df['c1_payoff'][i] = max((␣
↪c1_units*(df['stock_prices'][i]-c1_strike) + c1_cost),c1_cost)
        else:
            df['c1_payoff'][i] = min((␣
↪c1_units*(df['stock_prices'][i]-c1_strike) + c1_cost),c1_cost)
    for i in range(len(df)):
        if(c2_units>0):
            df['c2_payoff'][i] = max((␣
↪c2_units*(df['stock_prices'][i]-c2_strike) + c2_cost),c2_cost)
        else:
            df['c2_payoff'][i] = min((␣
↪c2_units*(df['stock_prices'][i]-c2_strike) + c2_cost),c2_cost)
    for i in range(len(df)):
        if(c3_units>0):
            df['c3_payoff'][i] = max((␣
↪c3_units*(df['stock_prices'][i]-c3_strike) + c3_cost),c3_cost)
        else:
            df['c3_payoff'][i] = min((␣
↪c3_units*(df['stock_prices'][i]-c3_strike) + c3_cost),c3_cost)

    # computing payoffs for puts
    for i in range(len(df)):
        if(p1_units>0):
            df['p1_payoff'][i] = max((␣
↪p1_units*(p1_strike-df['stock_prices'][i]) + p1_cost),p1_cost)
        else:
```

```
            df['p1_payoff'][i] = min((␣
↪p1_units*(p1_strike-df['stock_prices'][i]) + p1_cost),p1_cost)
    for i in range(len(df)):
        if(p2_units>0):
            df['p2_payoff'][i] = max((␣
↪p2_units*(p2_strike-df['stock_prices'][i]) + p2_cost),p2_cost)
        else:
            df['p2_payoff'][i] = min((␣
↪p2_units*(p2_strike-df['stock_prices'][i]) + p2_cost),p2_cost)
    for i in range(len(df)):
        if(p3_units>0):
            df['p3_payoff'][i] = max((␣
↪p2_units*(p3_strike-df['stock_prices'][i]) + p3_cost),p3_cost)
        else:
            df['p3_payoff'][i] = min((␣
↪p2_units*(p3_strike-df['stock_prices'][i]) + p3_cost),p3_cost)

    # computing payoffs from underlying stock
    for i in range(len(df)):
        df['stock_payoff'][i] = stock_units*(df['stock_prices'][i] -␣
↪stock_price)

    df['total_payoff'] = (df['c1_payoff'] + df['c2_payoff'] + df['c3_payoff'] +
                          df['p1_payoff'] + df['p2_payoff'] + df['p3_payoff']+␣
↪df['stock_payoff'])

    # print('Cost to setup this strategy is ', cost)
    print('\n')
    print('Payoffs from this strategy are shown:')

    # plotting the payoffs
    sns.lineplot(x='stock_prices', y='total_payoff', data=df )
    #g.set(xlim=(120, 290))
    #g.set(xlim=(190, 230))
    #g.set(ylim=(-40, +40))
    #g.spines['bottom'].set_linewidth(5)

    # Interactive plots
    #fig = px.line(df, x='stock_prices', y='total_payoff', title='Payoff Chart')
    #fig.show()

    # payoff chart
    # print('Payoff chart for this strategy with a +/- 100% variation in stock␣
↪price from its price today:')
    # display(df)
```

5 Verifying the calculator payoffs

Consider the following strategy. The stock price today is $S_0 = 212$. We invest in the following options:

Option	Position	Units	Premium	Strike Price
Call 1	Long	1	5.35	220
Call 2	Short	-1	7.63	215
Put 1	Long	1	5.52	205
Put 2	Short	-1	7.2	210

We solve this by considering the payoffs of the component calls and puts in the individual components over different ranges of stock price. For now we first assume that the cost of setup is zero, and make this payoff table. Once the table is calculated, we can subtract the initial setup cost from the individual values.

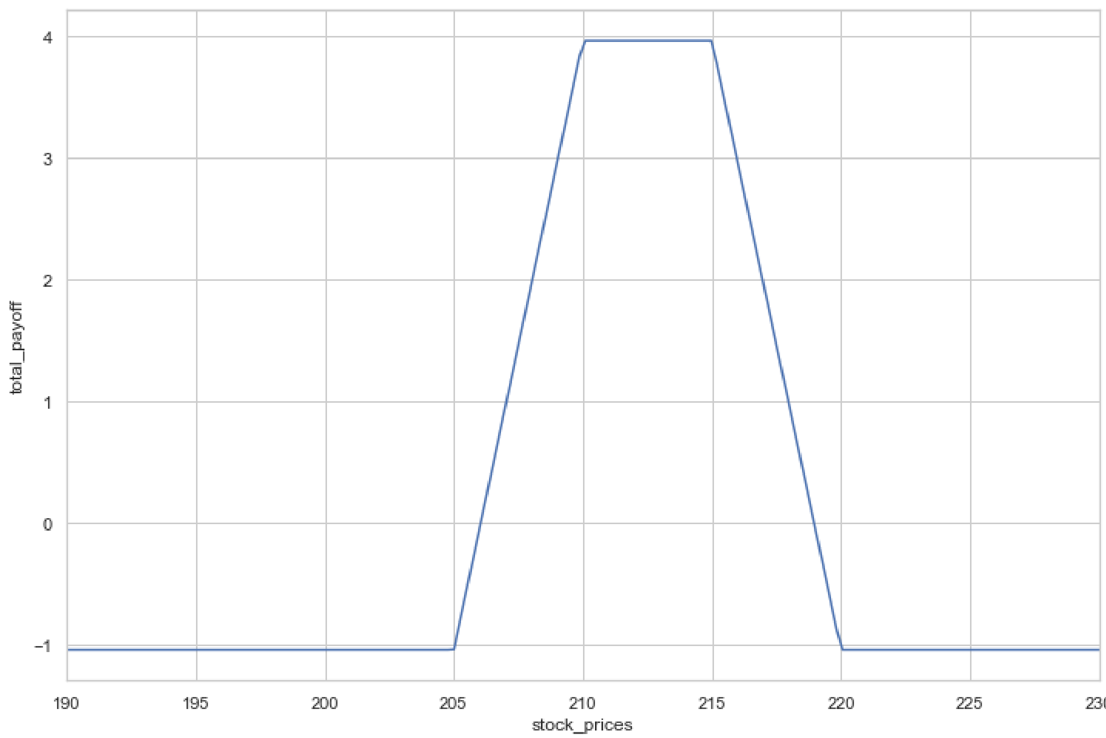
Stock price range	Exercised	Call 1 (L)	Call 2 (S)	Put 1 (L)	Put 2 (S)	Total Payoff
	put 1 & put 2	0	0			-5
$S_T \leq 205$				$205 - S_T$	$-(210 - S_T)$	
	put 2 alone	0	0	0		
$205 < S_T < 210$					$-(210 - S_T)$	$-(210 - S_T)$
	none	0	0	0	0	0
$210 < S_T < 215$						
	call 2 alone	0		0	0	
$215 < S_T < 220$			$-(S_T - 215)$			$-(S_T - 215)$
	call 1 & call 2			0	0	-5
$S_T \geq 220$		$(S_T - 220)$	$-(S_T - 215)$			

Let us take up particular values of S_T in these ranges, factor in the initial setup cost of $-c_1 + c_2 - p_1 + p_2 = -5.35 + 7.63 - 5.52 + 7.2 = 3.96$. The initial setup cost is positive. The maximum payoff from this strategy is 3.96.

Stock price	Payoff	Payoff value	Payoff with setup cost
203	-5	-5	-1.04
207	$-(210 - S_T)$	-3	0.96
212	0	0	3.96
217	$-(S_T - 215)$	-2	1.96
223	-5	-5	-1.04

[199]:
```
# Call parameters
c1_units, c1_price, c1_strike = 1,5.35,220
c2_units, c2_price, c2_strike = -1,7.63,215
c3_units, c3_price, c3_strike = 0,0,0

# Put parameter
p1_units, p1_price, p1_strike = 1,5.52,205
p2_units, p2_price, p2_strike = -1,7.2,210
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 0,212

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,␣
↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,␣
↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  ------------------------------
---------------------------------------------------------------------
Options |       Units    |       Premium |       Strike   | Setup Cost
---------------------------------------------------------------------
Call 1  |        1       |        5.35   |        220     |        -5.35
Call 2  |        -1      |        7.63   |        215     |        7.63
Call 3  |        0       |        0      |        0       |        0
Put 1   |        1       |        5.52   |        205     |        -5.52
Put 2   |        -1      |        7.2    |        210     |        7.2
Put 3   |        0       |        0      |        0       |        0
Stock   |        0       |        212    |        --      |        0
---------------------------------------------------------------------
Initial Setup Cost                                          3.960000000000001


Payoffs from this strategy are shown:
```



6 Collecting Stock and Option data

[201]:
```
from yahoo_fin import options
```

Collecting options data for a particular stock

[202]:
```
chain = options.get_options_chain("googl")
```

[203]:
```
calls = chain["calls"]
puts = chain["puts"]
```

[204]:
```
calls.head()
```

[204]:
```
          Contract Name          Last Trade Date  Strike  Last Price    Bid  \
0  GOOGL201030C00810000  2020-10-29 10:41AM EDT   810.0       856.2  800.1
1  GOOGL201030C00820000  2020-10-21 12:50PM EDT   820.0       770.4  794.2
2  GOOGL201030C00830000   2020-10-23 9:56AM EDT   830.0       795.9  781.0
3  GOOGL201030C00900000  2020-10-19 12:10AM EDT   900.0       663.7  713.8
4  GOOGL201030C00990000  2020-10-22 11:36AM EDT   990.0       605.6  623.0

     Ask  Change % Change Volume  Open Interest Implied Volatility
0  803.3   118.1  +16.00%      3              8              0.00%
1  796.9     0.0        -      -              2              0.00%
```

```
2  786.5      0.0          -       1              2              0.00%
3  716.7      0.0          -       -              1              0.00%
4  627.0      0.0          -       -              2              0.00%
```

[205]:
```
puts.tail()
```

[205]:
```
            Contract Name         Last Trade Date  Strike  Last Price    Bid  \
125  GOOGL201030P01940000  2020-10-23 12:08PM EDT  1940.0      320.42  325.3
126  GOOGL201030P01960000  2020-10-27 10:57AM EDT  1960.0      371.40  345.8
127  GOOGL201030P01980000   2020-10-22 9:55AM EDT  1980.0      383.20  365.8
128  GOOGL201030P02070000  2020-10-28 10:08AM EDT  2070.0      537.10  454.9
129  GOOGL201030P02090000   2020-10-27 9:45AM EDT  2090.0      506.70  471.2

       Ask  Change % Change Volume  Open Interest Implied Volatility
125  329.1     0.0        -      2              0            235.40%
126  348.3     0.0        -      1              1            244.12%
127  368.9     0.0        -      -              0            256.34%
128  459.3     0.0        -      2              1            295.87%
129  477.8     0.0        -      1              1            277.12%
```

While one can obtain all the contracts data using the above pulls, it also makes sense to search for expiration dates and collect contracts expiring on/by that particular date. This is performed next.

7 Strategy using real data

It is useful to query real options and stock data in order to develop strategies. Here we take up the `GOOGL` stock (Google) and option data, query the expiration dates and collect calls and puts that expire on that day.

[206]:
```
expDates = options.get_expiration_dates("googl")
expDates
```

[206]:
```
['October 30, 2020',
 'November 6, 2020',
 'November 13, 2020',
 'November 20, 2020',
 'November 27, 2020',
 'December 4, 2020',
 'December 18, 2020',
 'January 15, 2021',
 'February 19, 2021',
 'March 19, 2021',
 'June 18, 2021',
 'July 16, 2021',
 'August 20, 2021',
 'September 17, 2021',
 'October 15, 2021',
```

```
 'January 21, 2022',
 'June 17, 2022',
 'September 16, 2022',
 'January 20, 2023']
```

[207]:
```python
chains = options.get_options_chain("googl", "December 18, 2020")
calls = chains['calls']
puts = chains['puts']
# Alternatively use:
# options.get_options_chain("aapl", "MM/DD/YY")
# options.get_options_chain("aapl", "MM/DD/2021")
```

[208]:
```python
calls[(calls['Strike']>1500)&(calls['Strike']<1900)].head()
```

[208]:
```
            Contract Name          Last Trade Date  Strike  Last Price    Bid  \
98   GOOGL201218C01510000   2020-10-29 3:56PM EDT  1510.0      123.76  152.4
99   GOOGL201218C01520000   2020-10-29 3:56PM EDT  1520.0      189.14  147.6
100  GOOGL201218C01530000   2020-10-29 1:00PM EDT  1530.0      180.00  140.8
101  GOOGL201218C01540000   2020-10-30 9:57AM EDT  1540.0      143.93  134.6
102  GOOGL201218C01550000  2020-10-30 11:36AM EDT  1550.0      135.00  128.3

       Ask  Change % Change Volume  Open Interest Implied Volatility
98   158.9    0.00        -      1              2             40.74%
99   152.0   70.32  +59.18%      3             89             40.52%
100  145.6   61.10  +51.39%      1             13             40.47%
101  140.0   27.87  +24.01%      5            175             40.70%
102  131.2   28.32  +26.55%     23            623             39.42%
```

[209]:
```python
puts[(puts['Strike']>1400)&(puts['Strike']<2000)].tail()
```

[209]:
```
            Contract Name          Last Trade Date  Strike  Last Price    Bid  \
132  GOOGL201218P01800000   2020-09-30 9:36AM EDT  1800.0      247.00  167.1
133  GOOGL201218P01820000  2020-09-14 12:14PM EDT  1820.0      297.74  266.9
134  GOOGL201218P01850000  2020-10-30 11:33AM EDT  1850.0      250.70  251.0
135  GOOGL201218P01860000  2020-08-26 10:04AM EDT  1860.0      286.55  451.3
136  GOOGL201218P01880000  2020-08-26 10:04AM EDT  1880.0      302.77  467.5

       Ask  Change % Change Volume  Open Interest Implied Volatility
132  176.6    0.00        -      1              3              0.00%
133  272.5    0.00        -      4              2             58.70%
134  254.3 -138.38  -35.57%      1              0             38.97%
135  457.2    0.00        -      4              4            125.84%
136  473.4    0.00        -      4              4            126.72%
```

Let consider the above calls and puts with strike price in the range between 800 and 1400. Consider the option with a strike price of 1550. The corresponding call & puts have an implied volatility of about 42%. Implied volatility refers to the expected volatility of a stock over the lifetime of the option. These options are have an expiration date of December 18, 2020.

Since, we expect high volatility in stock prices in either direction, and we are unsure about the direction of this large movement, we use a **straddle combination** here. As we know this involves going long on both a call and put of the same strike price and expiration date. Here $K = 1550$, $c = 116.2$, $p = 83.8$. Current stock price for Google is 1556 USD.

7.0.1 Straddle

[210]:
```
print(calls[(calls['Strike']==1550)])
print(puts[(puts['Strike']==1550)])
```

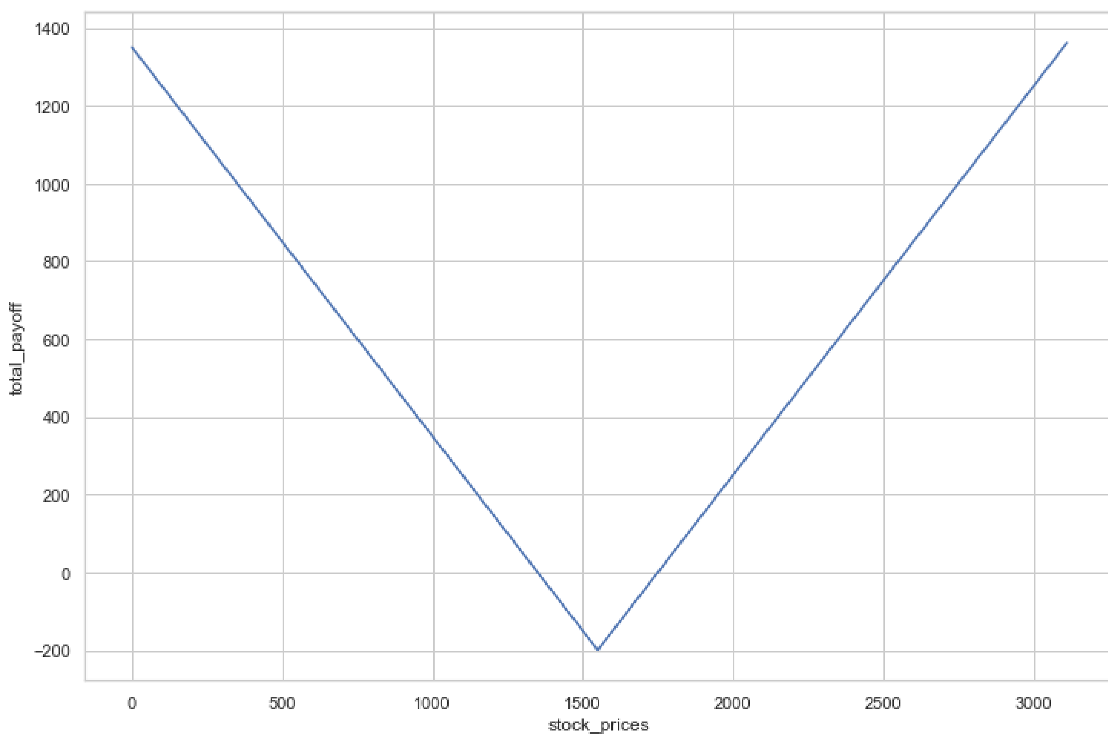
```
            Contract Name         Last Trade Date  Strike  Last Price    Bid  \
102  GOOGL201218C01550000  2020-10-30 11:36AM EDT  1550.0       135.0  128.3

       Ask  Change % Change Volume  Open Interest Implied Volatility
102  131.2   28.32  +26.55%     23            623             39.42%
            Contract Name        Last Trade Date  Strike  Last Price   Bid  \
118  GOOGL201218P01550000  2020-10-30 9:57AM EDT  1550.0        60.0  63.2

      Ask  Change % Change Volume  Open Interest Implied Volatility
118  66.2   -21.9  -26.74%      7            178             41.55%
```

[211]:
```
# Call parameters
c1_units, c1_price, c1_strike = 1,116.2,1550
c2_units, c2_price, c2_strike = 0,0,0
c3_units, c3_price, c3_strike = 0,0,0

# Put parameters
p1_units, p1_price, p1_strike = 1,83.8,1550
p2_units, p2_price, p2_strike = 0,0,0
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 0,1556

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,␣
 ↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,␣
 ↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  ------------------------------
--------------------------------------------------------------------
Options |        Units     |        Premium |        Strike    | Setup Cost
--------------------------------------------------------------------
Call 1  |        1         |         116.2  |         1550     |         -116.2
Call 2  |        0         |        0       |        0         |        0
```

```
Call 3   |        0       |        0       |        0       |        0
Put 1    |        1       |        83.8    |        1550    |        -83.8
Put 2    |        0       |        0       |        0       |        0
Put 3    |        0       |        0       |        0       |        0
Stock    |        0       |        1556    |        --      |        0
---------------------------------------------------------------------
Initial Setup Cost                                       -200.0
```

```
Payoffs from this strategy are shown:
```



Thus, we would make profits if the stock price goes below 1350 or beyond 1750. We are aware that a few years ago, Google started investing in driverless car technology. Say, we have information that Google might release their Driverless Cars to the market. There has been some uncertainty regarding the public perception of effective implementation of these cars. One could either perceive this as a revolutionary technology that would disrupt the transportation industry, or as a mere marketing stunt for an unattainable technology.

On the other hand, if we possess information that Google is not going to make any changes in their products or portfolio, in the upcoming 2 months, we feel that there might not be much change in the stock prices. In such a scenario, we adopt a butterfly strategy, with strike price pivoted around the current value of 1550.

7.0.2 Butterfly spread

Here we go short on 2 calls with $K_2 = 1550$. Simulataneously we go long on two calls with $K_1 = 1510$ and $K_3 = 1590$, such that $K_2 = 0.5(K_1 + K_3)$. We find that, $c_1 = 138.4$, $c_2 = 116.2$ and $c_3 = 93.5$.

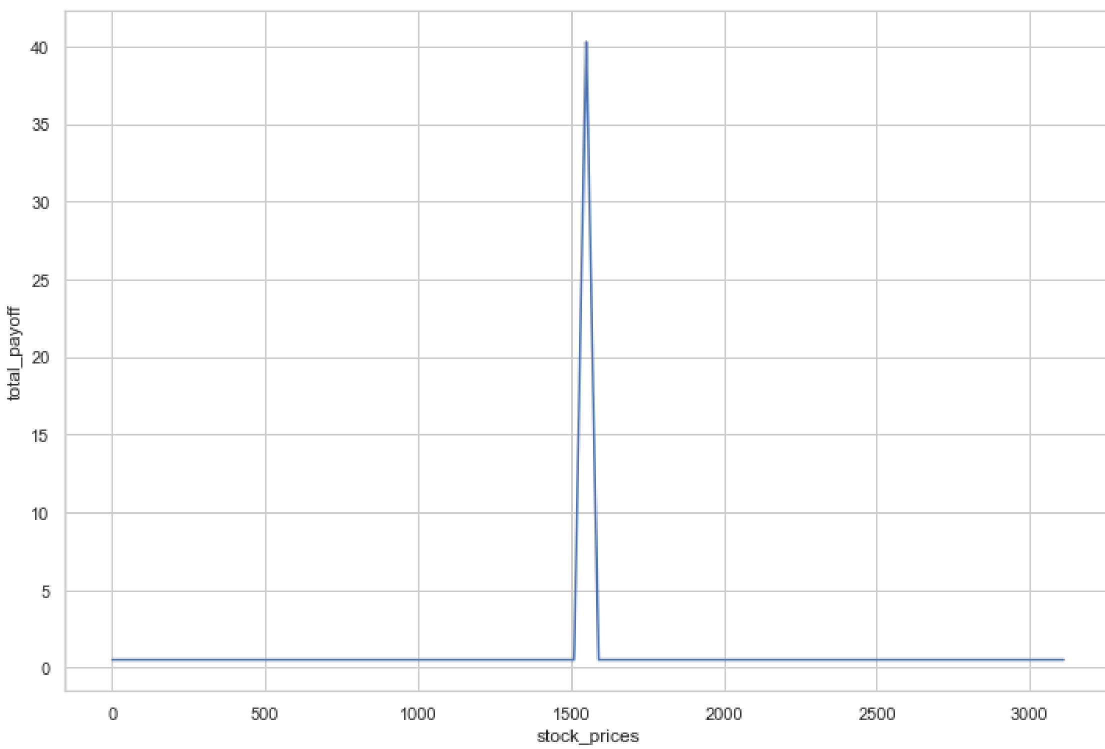
[212]:
```
calls[(calls['Strike']>1500)&(calls['Strike']<1600)]
```

[212]:
```
            Contract Name         Last Trade Date  Strike  Last Price    Bid  \
98   GOOGL201218C01510000   2020-10-29 3:56PM EDT  1510.0      123.76  152.4
99   GOOGL201218C01520000   2020-10-29 3:56PM EDT  1520.0      189.14  147.6
100  GOOGL201218C01530000   2020-10-29 1:00PM EDT  1530.0      180.00  140.8
101  GOOGL201218C01540000   2020-10-30 9:57AM EDT  1540.0      143.93  134.6
102  GOOGL201218C01550000  2020-10-30 11:36AM EDT  1550.0      135.00  128.3
103  GOOGL201218C01560000   2020-10-30 9:32AM EDT  1560.0      159.77  122.2
104  GOOGL201218C01570000   2020-10-29 3:53PM EDT  1570.0      128.03  116.2
105  GOOGL201218C01580000   2020-10-29 3:56PM EDT  1580.0      115.20  110.4
106  GOOGL201218C01590000   2020-10-27 3:57PM EDT  1590.0      121.30  104.7

       Ask  Change % Change Volume  Open Interest Implied Volatility
98   158.9    0.00        -      1              2             40.74%
99   152.0   70.32  +59.18%      3             89             40.52%
100  145.6   61.10  +51.39%      1             13             40.47%
101  140.0   27.87  +24.01%      5            175             40.70%
102  131.2   28.32  +26.55%     23            623             39.42%
103  128.2   63.53  +66.01%      2            135             40.67%
104  120.6   31.71  +32.92%      2              9             39.80%
105  114.5   29.01  +33.66%      4           1342             39.52%
106  108.7   21.60  +21.67%      2             21             39.31%
```

[213]:
```
# Call parameters
c1_units, c1_price, c1_strike = 1,138.4, 1510
c2_units, c2_price, c2_strike = -2,116.2, 1550
c3_units, c3_price, c3_strike = 1,93.5, 1590

# Put parameters
p1_units, p1_price, p1_strike = 0,4,50
p2_units, p2_price, p2_strike = 0,0,0
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 0,1556

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,␣
 ↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,␣
 ↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  ------------------------------
--------------------------------------------------------------------
Options |        Units     |        Premium |        Strike   | Setup Cost
--------------------------------------------------------------------
Call 1  |         1        |         138.4  |         1510    |         -138.4
Call 2  |         -2       |         116.2  |         1550    |         232.4
Call 3  |         1        |         93.5   |         1590    |         -93.5
Put 1   |         0        |         4      |         50      |         0
Put 2   |         0        |         0      |         0       |         0
Put 3   |         0        |         0      |         0       |         0
Stock   |         0        |         1556   |         --      |         0
--------------------------------------------------------------------
Initial Setup Cost                                        0.5
```

```
Payoffs from this strategy are shown:
```



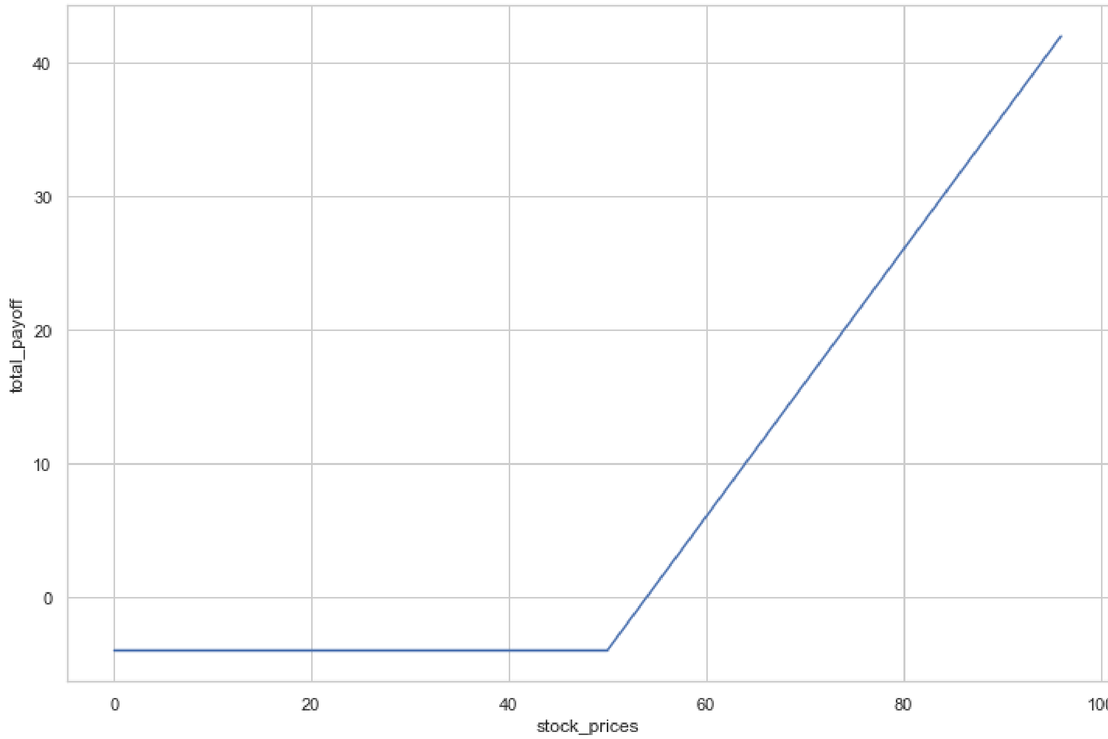
Here we see that the butterfly spread would be profitable in case the Google stock price lies within the range between 1510 and 1590.

8 Long Call

Consider a call option with the following parameters: $S_0 = 48$, $K = 50$ and $c = 4$. Our position is to go long on this call which means that we have the right to buy the underlying stock at the expiration date. In taking this position we are betting that the stock price will go up in the future. That is, if the stock price increases tomorrow by a value greater than 50, say 55, then we will exercise the option and buy the stock at strike price 50. Now, on selling it in the market for 55 we make a profit of 1 which is (5-4). In such a scenario, we say that the call is *in the money* (ITM).

[214]:
```
# Call parameters
c1_units, c1_price, c1_strike = 1,4,50
c2_units, c2_price, c2_strike = 0,0,0
c3_units, c3_price, c3_strike = 0,0,0

# Put parameters
p1_units, p1_price, p1_strike = 0,0,0
p2_units, p2_price, p2_strike = 0,0,0
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 0,48

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,␣
↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,␣
↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  ------------------------------
---------------------------------------------------------------------
Options |       Units    |       Premium |       Strike   | Setup Cost
---------------------------------------------------------------------
Call 1  |         1      |         4      |       50      |         -4
Call 2  |         0      |         0      |       0       |         0
Call 3  |         0      |         0      |       0       |         0
Put 1   |         0      |         0      |       0       |         0
Put 2   |         0      |         0      |       0       |         0
Put 3   |         0      |         0      |       0       |         0
Stock   |         0      |         48     |       --      |         0
---------------------------------------------------------------------
Initial Setup Cost                                         -4


Payoffs from this strategy are shown:
```



From the graph, we see that for all values of stock price less than $K = 50$, we do not exercise the call option and incur a loss of $c = 4$, (which is also the initial cost to set up this strategy). The break even point occurs at 54 which is $(K + c)$. In the region between 50 and 54, we minimze the loss. Beyond this point, we make profits.

9 Long Put

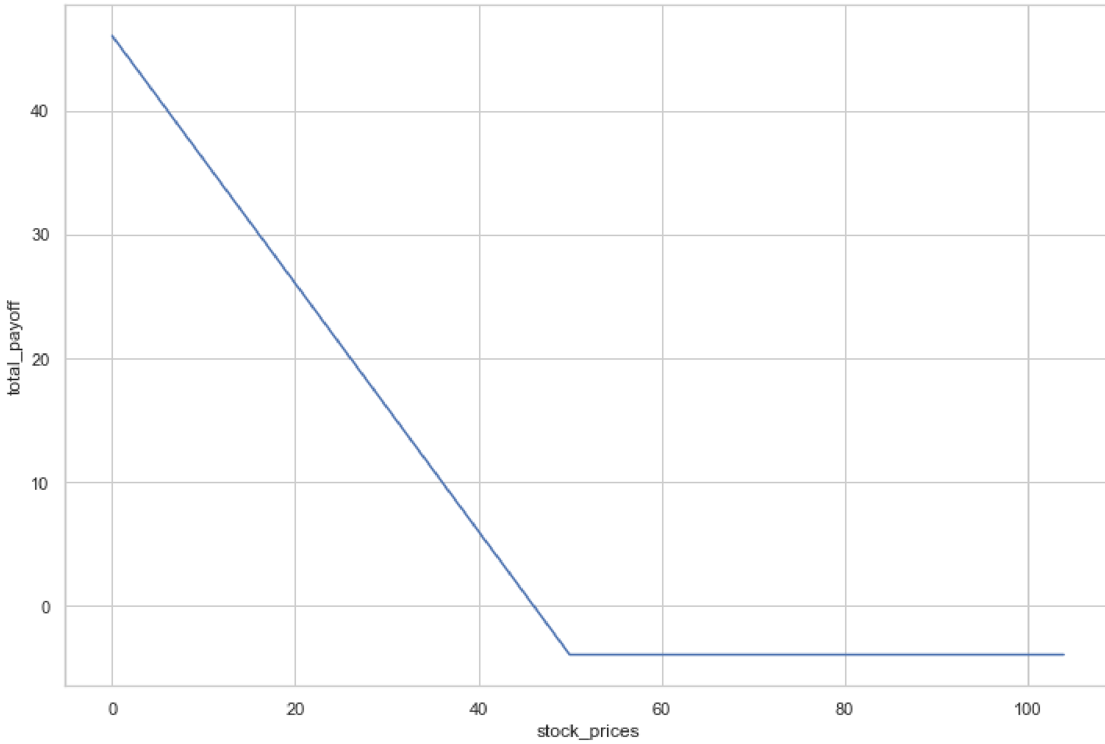
Now, consider a put option with the following parameters: $S_0 = 52$, $K = 50$ and $p = 4$. Our position is to go long on this put which means that we have the right to sell the underlying stock at the expiration date. In taking this position we are betting that the stock price will go down in the future. That is, if the stock price decreases tomorrow by a value lesser than 50, say 42, then we will exercise the option. Firstly, I buy the stock from the market at 42 and sell the stock at strike price 50, thus making a profit of 4 which is (50-42-4). In such a scenario, we say that the put is *in the money* (ITM).

[215]:
```
# Call parameters
c1_units, c1_price, c1_strike = 0,0,0
c2_units, c2_price, c2_strike = 0,0,0
c3_units, c3_price, c3_strike = 0,0,0

# Put parameters
p1_units, p1_price, p1_strike = 1,4,50
p2_units, p2_price, p2_strike = 0,0,0
```

```python
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 0,52

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,␣
 ↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,␣
 ↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  -------------------------------
---------------------------------------------------------------------
Options |        Units     |        Premium |        Strike   | Setup Cost
---------------------------------------------------------------------
Call 1  |         0        |         0      |         0       |         0
Call 2  |         0        |         0      |         0       |         0
Call 3  |         0        |         0      |         0       |         0
Put 1   |         1        |         4      |         50      |         -4
Put 2   |         0        |         0      |         0       |         0
Put 3   |         0        |         0      |         0       |         0
Stock   |         0        |         52     |         --      |         0
---------------------------------------------------------------------
Initial Setup Cost                                       -4
```

Payoffs from this strategy are shown:



10 Short Call

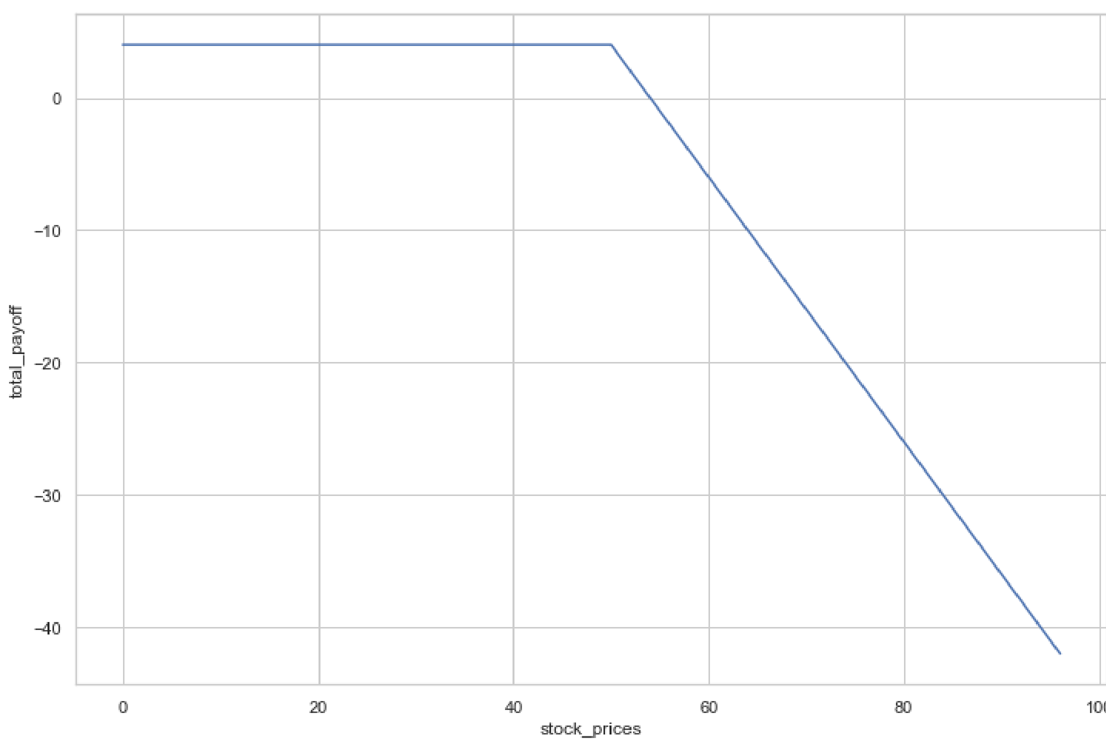
Consider the same call option as before with $S_0 = 48$, $K = 50$ and $c = 4$. Our position now, is to go short on this call which means that we are obligated to buy the underlying stock at the expiration date. In taking this position we make limited profits (of $c = 4$) if the stock price goes down in the future. On the other hand if the stock price goes up, we potentially make unlimited losses.

[216]:

```
# Call parameters
c1_units, c1_price, c1_strike = -1,4,50
c2_units, c2_price, c2_strike = 0,0,0
c3_units, c3_price, c3_strike = 0,0,0

# Put parameters
p1_units, p1_price, p1_strike = 0,0,0
p2_units, p2_price, p2_strike = 0,0,0
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 0,48

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,␣
↪c2_strike,c3_units, c3_price, c3_strike,
```

```
                       p1_units, p1_price, p1_strike, p2_units, p2_price,␣
↪p2_strike,p3_units, p3_price, p3_strike,
                       stock_units, stock_price)
```

```
--------------------------  Strategy  -------------------------------
---------------------------------------------------------------------
Options |        Units    |       Premium |        Strike   | Setup Cost
---------------------------------------------------------------------
Call 1  |         -1      |       4       |        50       |       4
Call 2  |        0        |       0       |        0        |       0
Call 3  |        0        |       0       |        0        |       0
Put 1   |        0        |       0       |        0        |       0
Put 2   |        0        |       0       |        0        |       0
Put 3   |        0        |       0       |        0        |       0
Stock   |        0        |       48      |        --       |       0
---------------------------------------------------------------------
Initial Setup Cost                                        4


Payoffs from this strategy are shown:
```



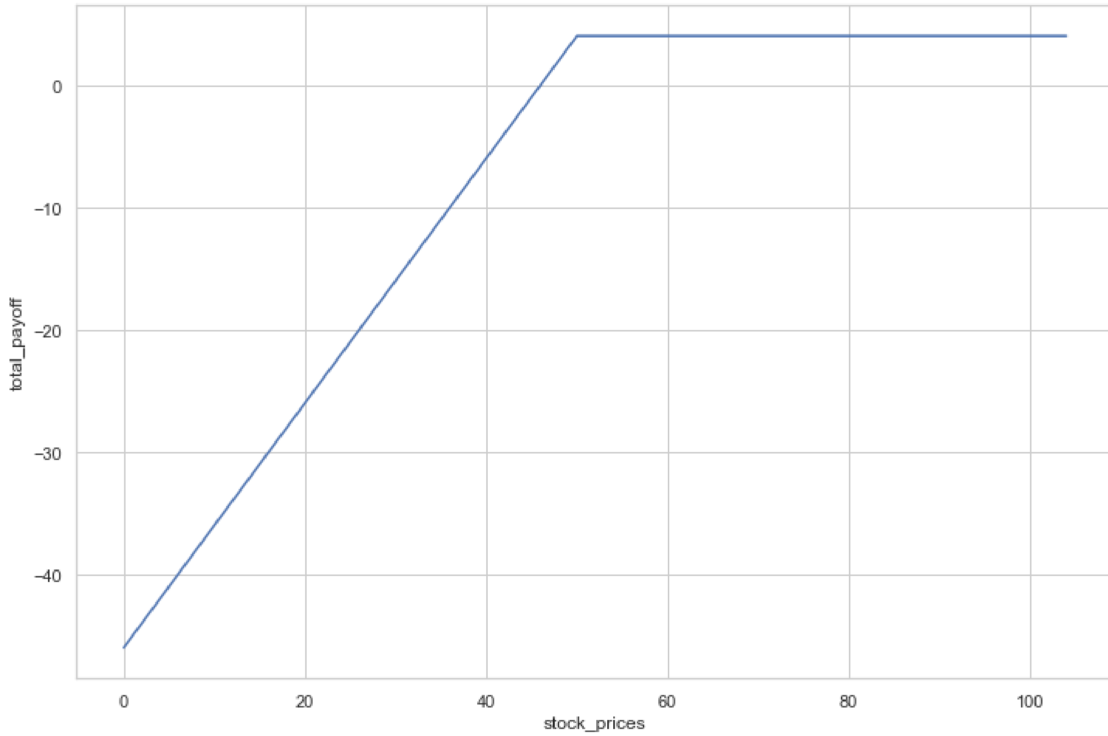
11 Short Put

Now, consider the same put option as before: $S_0 = 52$, $K = 50$ and $p = 4$. Our position is to go short on this put which means that we are obligated to sell the underlying stock on the expiration date. In taking this position we make limited profits if prices go up, but unlimeted losses if prices go down (bound by zero).

[217]:

```
# Call parameters
c1_units, c1_price, c1_strike = 0,0,0
c2_units, c2_price, c2_strike = 0,0,0
c3_units, c3_price, c3_strike = 0,0,0

# Put parameters
p1_units, p1_price, p1_strike = -1,4,50
p2_units, p2_price, p2_strike = 0,0,0
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 0,52

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,␣
↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,␣
↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  -------------------------------
---------------------------------------------------------------------
Options |        Units    |        Premium |        Strike   | Setup Cost
---------------------------------------------------------------------
Call 1  |         0       |        0       |        0        |        0
Call 2  |         0       |        0       |        0        |        0
Call 3  |         0       |        0       |        0        |        0
Put 1   |        -1       |        4       |        50       |        4
Put 2   |         0       |        0       |        0        |        0
Put 3   |         0       |        0       |        0        |        0
Stock   |         0       |        52      |        --       |        0
---------------------------------------------------------------------
Initial Setup Cost                                          4


Payoffs from this strategy are shown:
```



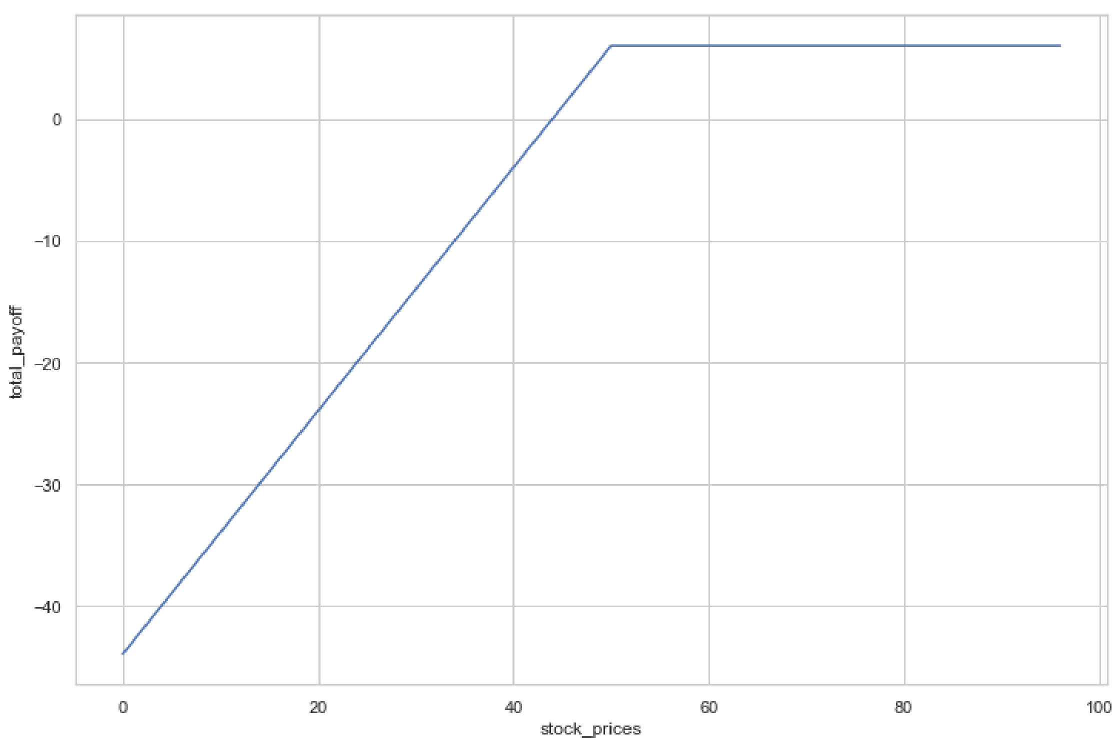
12 Covered Call

- In a covered call we take a long position in a stock and a short position in a call option.
- Here, we observe that the payoff is similar to that of a short put. This is because, if $S_T > K$, the call makes a loss, which is compensated by profits in the stock, just as seen in a short put.
- We know from put-call parity that $S_0 - c = Ke^{-rT} + D - p$. This means that a short position in call and a long position in stock, is equivalent to a short put and a certain amount of cash $(Ke^{-rT} + D)$.

[218]:
```
# Call parameters
c1_units, c1_price, c1_strike = -1,4,50
c2_units, c2_price, c2_strike = 0,0,0
c3_units, c3_price, c3_strike = 0,0,0

# Put parameters
p1_units, p1_price, p1_strike = 0,0,0
p2_units, p2_price, p2_strike = 0,0,0
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 1,48
```

```
payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,␣
↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,␣
↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  ------------------------------
---------------------------------------------------------------------
Options |        Units    |       Premium |       Strike   | Setup Cost
---------------------------------------------------------------------
Call 1  |        -1       |       4       |       50       |       4
Call 2  |        0        |       0       |       0        |       0
Call 3  |        0        |       0       |       0        |       0
Put 1   |        0        |       0       |       0        |       0
Put 2   |        0        |       0       |       0        |       0
Put 3   |        0        |       0       |       0        |       0
Stock   |        1        |       48      |       --       |       -48
---------------------------------------------------------------------
Initial Setup Cost                                      -44


Payoffs from this strategy are shown:
```

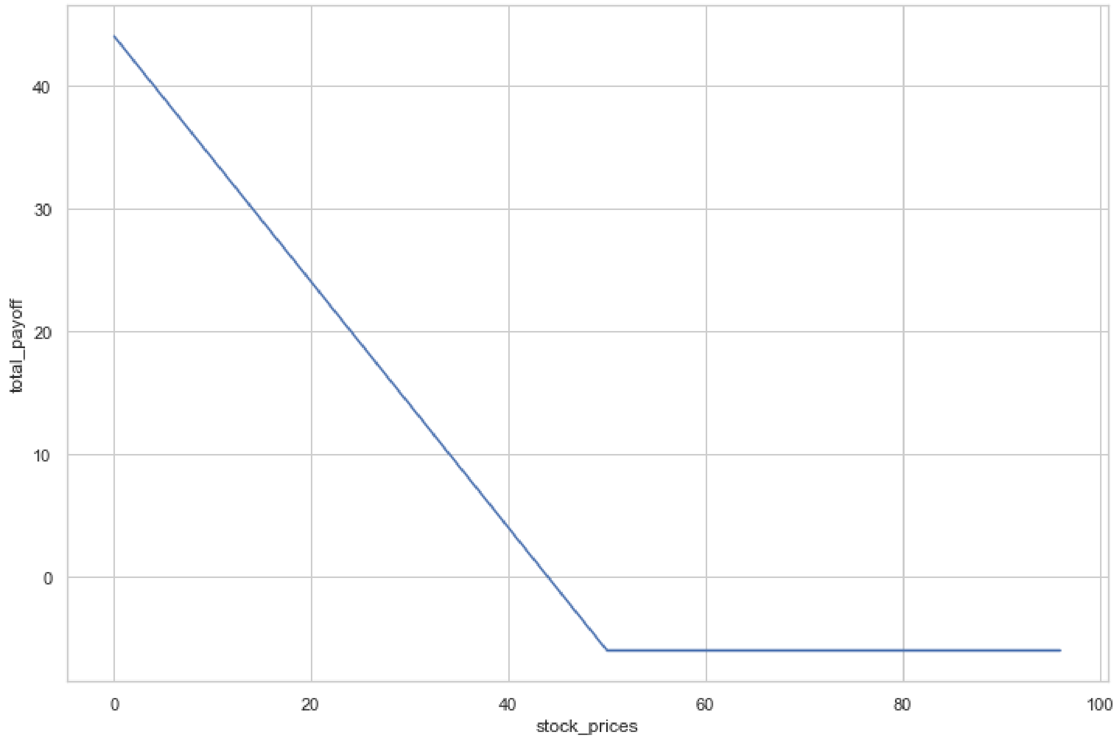


13 Reverse Covered Call

- In a reverse covered call we take a short position in a stock and a long position in a call option.
- Here, we observe that the payoff is similar to that of a long put. This is because, if $S_T > K$, the stock makes a loss, which is compensated by profits in the long call, just as seen in a long put.
- We know from put-call parity that $-S_0 + c = -(Ke^{-rT} + D) + p$. This means that a long position in call and a short position in stock, is equivalent to a long put and loss of a certain amount of cash $(Ke^{-rT} + D)$.

[219]:
```
# Call parameters
c1_units, c1_price, c1_strike = 1,4,50
c2_units, c2_price, c2_strike = 0,0,0
c3_units, c3_price, c3_strike = 0,0,0

# Put parameters
p1_units, p1_price, p1_strike = 0,0,0
p2_units, p2_price, p2_strike = 0,0,0
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = -1,48

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,␣
↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,␣
↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  -------------------------------
---------------------------------------------------------------------
Options |        Units    |        Premium |        Strike   | Setup Cost
---------------------------------------------------------------------
Call 1  |        1        |        4       |        50       |        -4
Call 2  |        0        |        0       |        0        |        0
Call 3  |        0        |        0       |        0        |        0
Put 1   |        0        |        0       |        0        |        0
Put 2   |        0        |        0       |        0        |        0
Put 3   |        0        |        0       |        0        |        0
Stock   |        -1       |        48      |        --       |        48
---------------------------------------------------------------------
Initial Setup Cost                                        44


Payoffs from this strategy are shown:
```



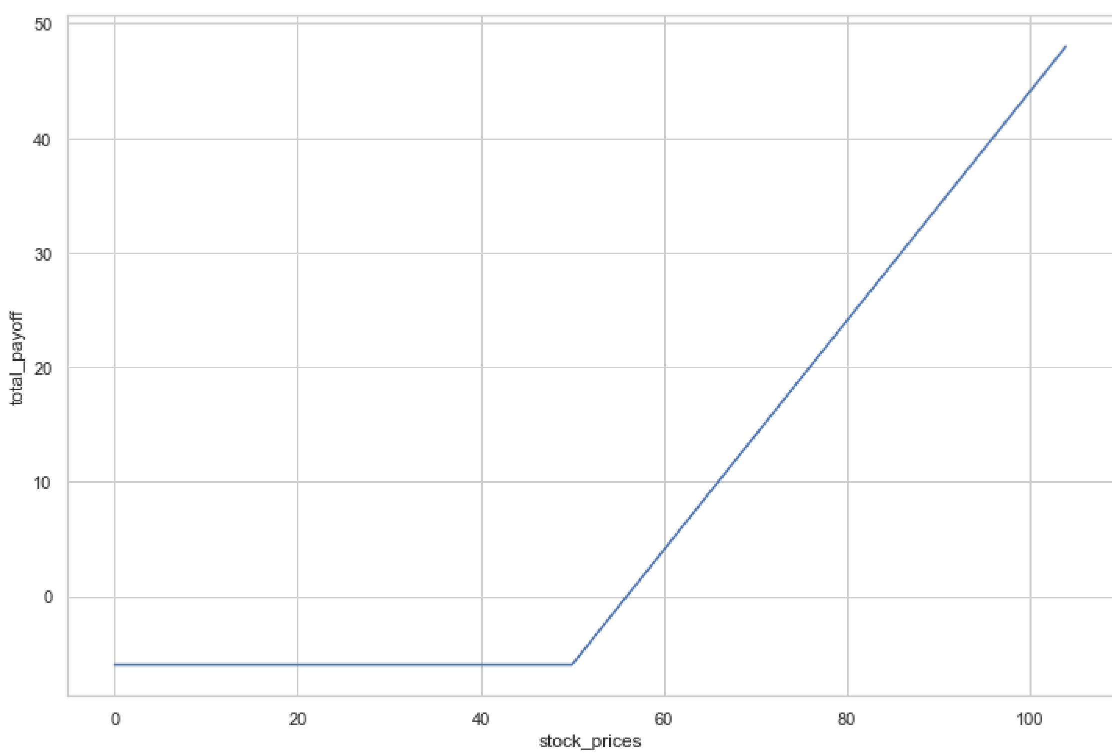
14 Protective Put

- In a protective put we take a long position in a stock and a long position in a put option.
- Here, we observe that the payoff is similar to that of a long call. This is because, if $S_T < K$, the stock makes a loss, which is compensated by profits in the long put, just as seen in a long call.
- We know from put-call parity that $p + S_0 = Ke^{-rT} + D + c$. This means that a long position in put and a long position in stock, is equivalent to a long call and a certain amount of cash $Ke^{-rT} + D$.

[220]:
```
# Call parameters
c1_units, c1_price, c1_strike = 0,0,0
c2_units, c2_price, c2_strike = 0,0,0
c3_units, c3_price, c3_strike = 0,0,0

# Put parameters
p1_units, p1_price, p1_strike = 1,4,50
p2_units, p2_price, p2_strike = 0,0,0
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 1,52
```

```
payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,␣
 ↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,␣
 ↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  -------------------------------
---------------------------------------------------------------------
Options |        Units    |       Premium |       Strike   | Setup Cost
---------------------------------------------------------------------
Call 1  |         0       |         0     |         0      |         0
Call 2  |         0       |         0     |         0      |         0
Call 3  |         0       |         0     |         0      |         0
Put 1   |         1       |         4     |         50     |         -4
Put 2   |         0       |         0     |         0      |         0
Put 3   |         0       |         0     |         0      |         0
Stock   |         1       |         52    |         --     |         -52
---------------------------------------------------------------------
Initial Setup Cost                                         -56


Payoffs from this strategy are shown:
```



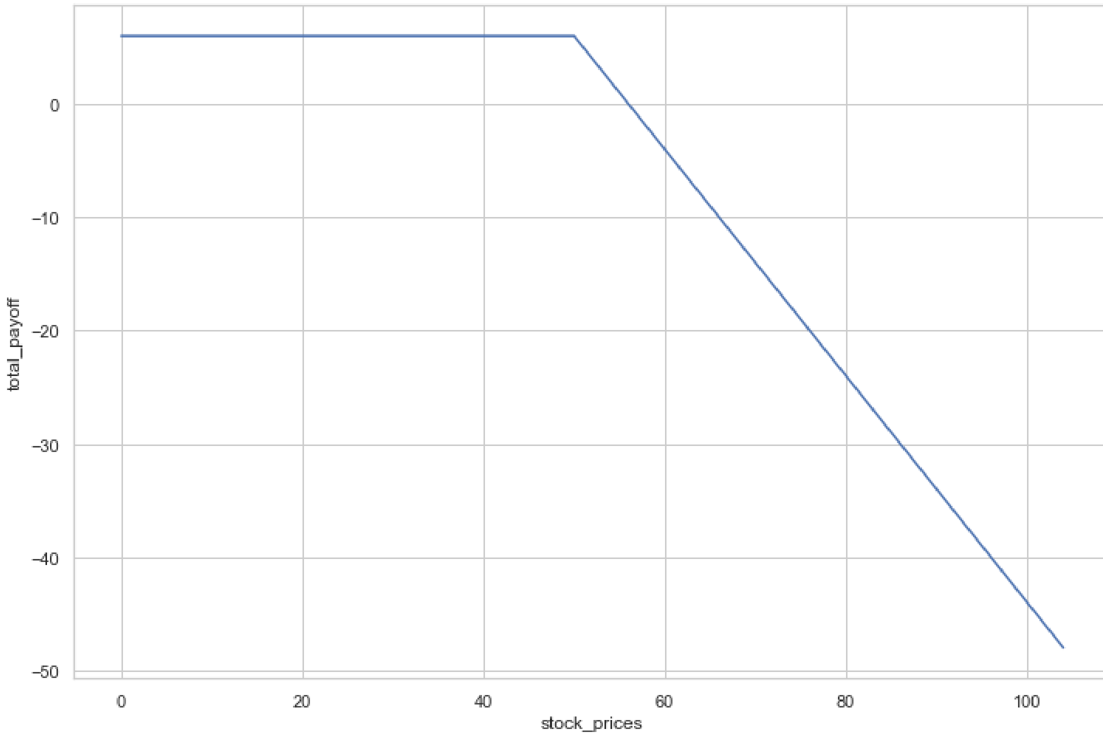
15 Reverse Protective Put

- In a reverse protective put we take a short position in a stock and a short position in a put option.
- Here, we observe that the payoff is similar to that of a short call. This is because, if $S_T < K$, the short put makes a loss, which is compensated by profits in the short stock, just as seen in a short call.
- We know from put-call parity that $-(p + S_0) = -(Ke^{-rT} + D) - c$. This means that a long position in put and a long position in stock, is equivalent to a short call and loss of a certain amount of cash $-(Ke^{-rT} + D)$.

[221]:

```
# Call parameters
c1_units, c1_price, c1_strike = 0,0,0
c2_units, c2_price, c2_strike = 0,0,0
c3_units, c3_price, c3_strike = 0,0,0

# Put parameters
p1_units, p1_price, p1_strike = -1,4,50
p2_units, p2_price, p2_strike = 0,0,0
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = -1,52

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,
 ↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,
 ↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  ------------------------------
--------------------------------------------------------------------
Options |        Units    |        Premium |        Strike   | Setup Cost
--------------------------------------------------------------------
Call 1  |        0        |        0       |        0        |        0
Call 2  |        0        |        0       |        0        |        0
Call 3  |        0        |        0       |        0        |        0
Put 1   |        -1       |        4       |        50       |        4
Put 2   |        0        |        0       |        0        |        0
Put 3   |        0        |        0       |        0        |        0
Stock   |        -1       |        52      |        --       |        52
--------------------------------------------------------------------
Initial Setup Cost                                        56


Payoffs from this strategy are shown:
```



16 Bull Spread (call)

- Buy call at K_1 and sell call at K_2 such that $K_1 < K_2$, and the expiration dates and underlying assets are the same.
- As K increases, the price of the call always decreases. Thus, $K_1 < K_2$ implies $c_1 > c_2$. That is, the value of the call sold is less than that of the call purchased. Hence, bull spreads created from calls require an initial investment. This can be seen from the following example.
- Bull strategy, while protecting the investor's downside risk, also limits the upside risk. When the investor sells the second call at K_2, they are essentially limiting the upside potential.
- One invests in the bull strategy when they believe that the underlying asset prices will rise.

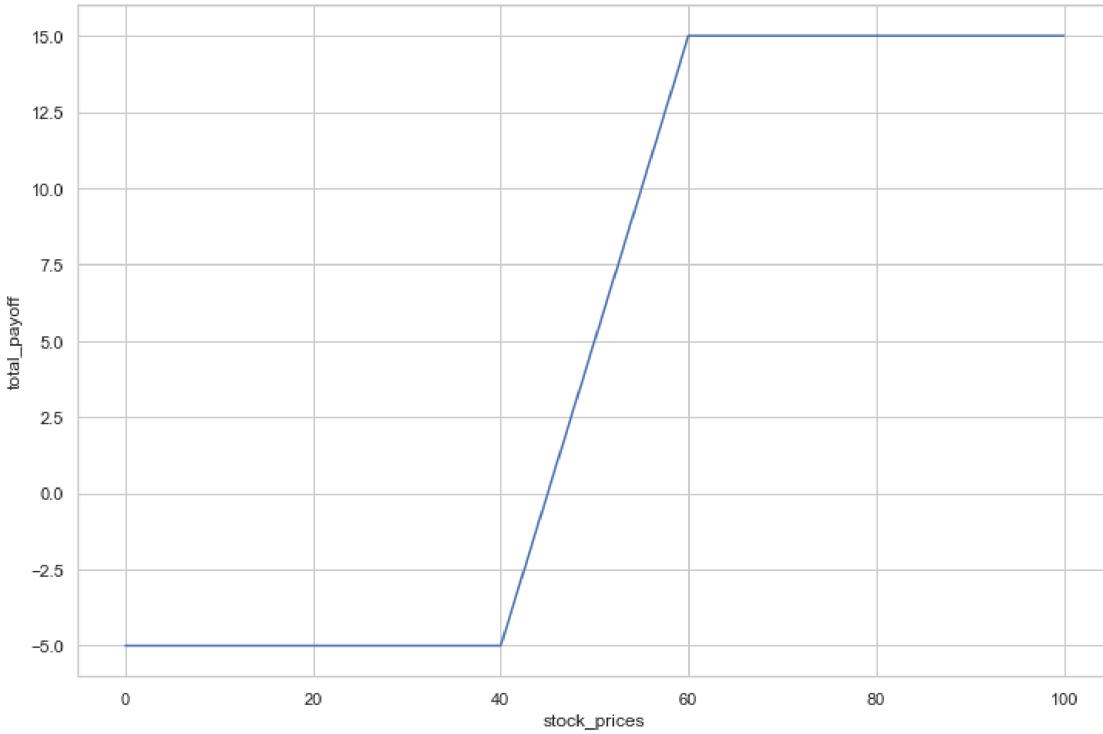
S_T range	long call payoff	short call payoff	total payoff
$S_T \leq K_1$	0	0	0
$K_1 < S_T < K_2$	$S_T - K_1$	0	$S_T - K_1$
$S_T \geq K_2$	$S_T - K_1$	$-(S_T - K_2)$	$K_2 - K_1$

Payoffs from bull spreads created using calls

[222]:
```
# Call parameters
c1_units, c1_price, c1_strike = 1,15,40
c2_units, c2_price, c2_strike = -1,10,60
c3_units, c3_price, c3_strike = 0,0,0
```

```python
# Put parameters
p1_units, p1_price, p1_strike = 0,0,0
p2_units, p2_price, p2_strike = 0,0,0
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 0,50

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,
 ↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,
 ↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  -------------------------------
---------------------------------------------------------------------
Options |       Units    |       Premium |       Strike   | Setup Cost
---------------------------------------------------------------------
Call 1  |        1       |        15     |        40      |        -15
Call 2  |        -1      |        10     |        60      |        10
Call 3  |        0       |        0      |        0       |        0
Put 1   |        0       |        0      |        0       |        0
Put 2   |        0       |        0      |        0       |        0
Put 3   |        0       |        0      |        0       |        0
Stock   |        0       |        50     |        --      |        0
---------------------------------------------------------------------
Initial Setup Cost                                       -5


Payoffs from this strategy are shown:
```



In this example, the investor buys the call with $K_1 = 40$ at $c_1 = 15$ and sells the call with $K_2 = 60$ at $c_2 = 10$. As noted earlier, the value of European calls decreases as strike price increases. Initially the stock is priced at $S_0 = 50$. The cost of setting up the bull strategy is $15 - 10 = 5$. If S_T falls below $K_1 = 40$, the payoff from the bull strategy is -5. If S_T is between 40 and 60, the payoff is $S_T - K_1 - (c_1 - c_2) = S_T - 40 - 5$. If S_T rises above 60, the payoff is the $K_2 - K_1 - (c_1 - c_2)$. Note that $(c_1 - c_2)$ is the cost to set up this strategy.

S_T	payoff
$S_T \leq 40$	$cost = -5$
$40 < S_T < 60$	$S_T - K_1 - cost = S_T - 45$
$S_T \geq 60$	$K_2 - K_1 - cost = 15$

17 Bear Spread (put)

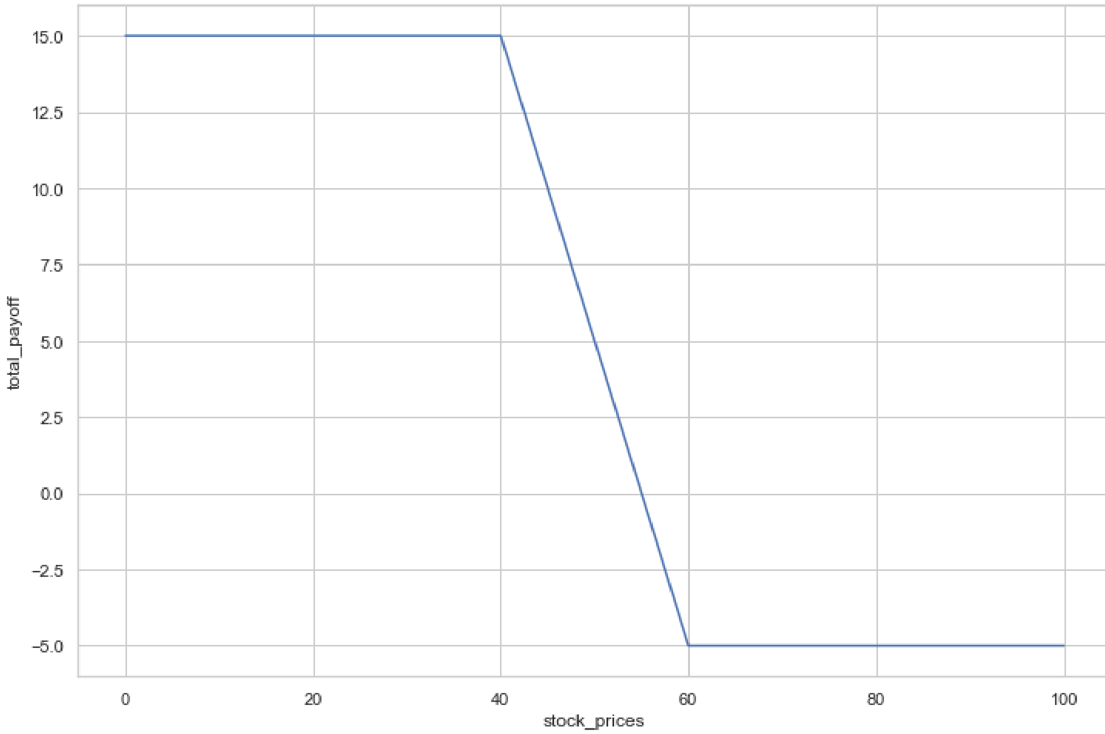
- Buy put at K_2 and sell put at K_1 such that $K_1 < K_2$, and the expiration dates and underlying assets are the same.
- We invest in the bear spread strategy if we expect the stock price to go down in the future.
- Since $K_2 > K_1$, we expect that $p_2 > p_1$, for puts with same expiration date and underlying stock.
- Since we sell the second put at p_1 which is a lower price than that we bought (at p_2), there is a initial cost to setup this strategy. This cost equals $p_2 - p_1$.

S_T range	long call payoff	short call payoff	total payoff
$S_T \leq K_1$	$K_2 - S_T$	$-(K_1 - S_T)$	$K_2 - K_1$
$K_1 \leq S_T \leq K_2$	$K_2 - S_T$	0	$K_2 - S_T$
$S_T \geq K_2$	0	0	0

Payoffs from bear spreads created using puts

[223]:
```
# Call parameters
c1_units, c1_price, c1_strike = 0,0,0
c2_units, c2_price, c2_strike = 0,0,0
c3_units, c3_price, c3_strike = 0,0,0

# Put parameters
p1_units, p1_price, p1_strike = 1,15,60
p2_units, p2_price, p2_strike = -1,10,40
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 0,50

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,␣
↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,␣
↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  ------------------------------
--------------------------------------------------------------------
Options |        Units    |        Premium |        Strike   | Setup Cost
--------------------------------------------------------------------
Call 1  |        0        |        0       |        0        |        0
Call 2  |        0        |        0       |        0        |        0
Call 3  |        0        |        0       |        0        |        0
Put 1   |        1        |        15      |        60       |        -15
Put 2   |        -1       |        10      |        40       |        10
Put 3   |        0        |        0       |        0        |        0
Stock   |        0        |        50      |        --       |        0
--------------------------------------------------------------------
Initial Setup Cost                                        -5


Payoffs from this strategy are shown:
```



The investor, buys the put with $K_2 = 60$ at $p_2 = 15$ and sells a put with $K_1 = 40$ at $p_2 = 10$. The cost to setup this bear spread is $p_2 - p_1 = 15 - 10 = 5$. We see the following payoff:

S_T	payoff
$S_T \leq 40$	$K_2 - K_1 - cost = 15$
$40 < S_T < 60$	$K_2 - S_T - cost = 55 - S_T$
$S_T \geq 60$	cost = -5

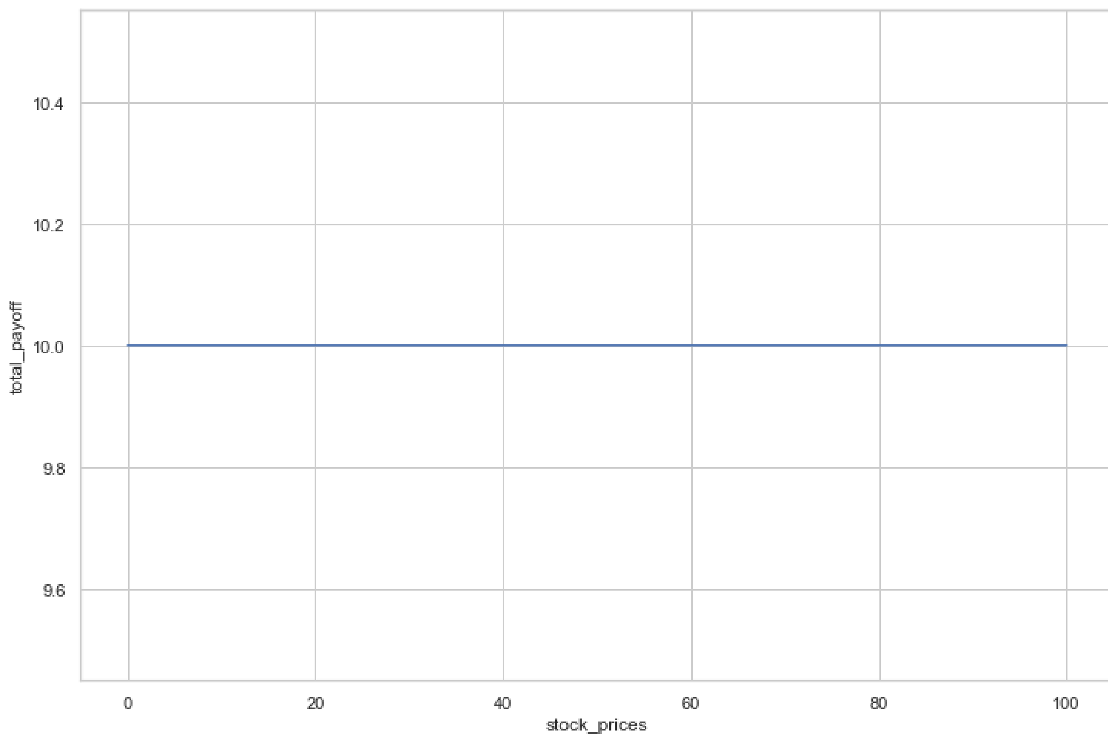
18 Box Spread

A box spread is a combination of a bull call spread with strike prices K1 and K2 and a bear put spread with the same two strike prices. The payoff from a box spread is always $K_2 - K_1$ (without factoring in the cost). The value of a box spread is therefore always the present value of this payoff that is $(K_2 - K_1)e^{-rT}$.

[224]:
```
# Call parameters
c1_units, c1_price, c1_strike = 1,15,40
c2_units, c2_price, c2_strike = -1,10,60
c3_units, c3_price, c3_strike = 0,0,0

# Put parameters
p1_units, p1_price, p1_strike = 1,15,60
```

```
p2_units, p2_price, p2_strike = -1,10,40
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 0,50

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,␣
↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,␣
↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  -------------------------------
---------------------------------------------------------------------
Options |        Units    |        Premium |        Strike   | Setup Cost
---------------------------------------------------------------------
Call 1  |         1       |         15     |         40      |         -15
Call 2  |         -1      |         10     |         60      |         10
Call 3  |         0       |         0      |         0       |         0
Put 1   |         1       |         15     |         60      |         -15
Put 2   |         -1      |         10     |         40      |         10
Put 3   |         0       |         0      |         0       |         0
Stock   |         0       |         50     |         --      |         0
---------------------------------------------------------------------
Initial Setup Cost                                        -10


Payoffs from this strategy are shown:
```



19 Butterfly Spread

- We construct a butterfly spread using positions in options with 3 different 3 strike prices.
- When implemented using calls it involves buying a call with low strike price K_1 and a call with high strike price K_2. In addition we also sell two calls with strike price halfway between the two $K_2 = 0.5(K_1 + K_3)$.
- If the stock price has a limited movement around K_2, this strategy is profitable.

S_T range	1^{st} long call payoff	2^{nd} long call payoff	short calls payoffs	total payoff
$S_T \leq K_1$	0	0	0	0
$K_1 < S_T < K_2$	$S_T - K_1$	0	0	$S_T - K_1$

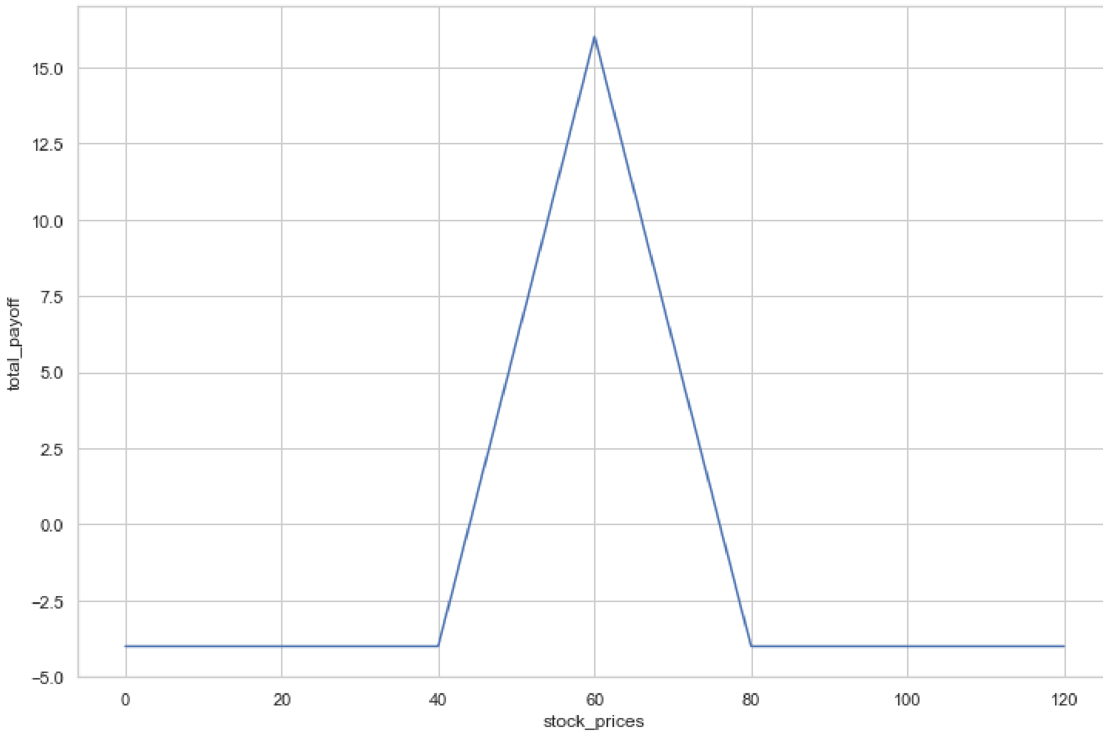
S_T range	1^{st} long call payoff	2^{nd} long call payoff	short calls payoffs	total payoff
$K_2 < S_T < K_3$	$S_T - K_1$	0	$-2(S_T - K_2)$	$K_3 - S_T$
$S_T \geq K_3$	$S_T - K_1$	$S_T - K_3$	$-2(S_T - K_2)$	0

Payoffs from a butterfly spread

[225]:
```
# Call parameters
c1_units, c1_price, c1_strike = 1,25,40
c2_units, c2_price, c2_strike = -2,18,60
c3_units, c3_price, c3_strike = 1,15,80

# Put parameters
p1_units, p1_price, p1_strike = 0,4,50
p2_units, p2_price, p2_strike = 0,0,0
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 0,60

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,␣
↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,␣
↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  -------------------------------
---------------------------------------------------------------------
Options |        Units    |        Premium |        Strike   | Setup Cost
---------------------------------------------------------------------
Call 1  |         1       |         25     |         40      |         -25
Call 2  |         -2      |         18     |         60      |         36
Call 3  |         1       |         15     |         80      |         -15
Put 1   |         0       |         4      |         50      |         0
Put 2   |         0       |         0      |         0       |         0
Put 3   |         0       |         0      |         0       |         0
Stock   |         0       |         60     |         --      |         0
---------------------------------------------------------------------
Initial Setup Cost                                         -4
```

```
Payoffs from this strategy are shown:
```



We create this butterfly spread by buying two calls with $K_1 = 40$ at $c_1 = 25$ and $K_3 = 80$ at $c_3 = 15$; and selling two calls with $K_2 = (40 + 120)/2 = 60$ at 18. The total cost to setup this strategy is $25 + 15 - (2 \times 18) = 40 - 36 = 4$.

If the stock price in the future becomes greater than 80 or less than 40, then the total payoff is zero, and the initial cost of 4 is the loss incurred. On the other hand, as predicted, if the stock price is between 40 and 80, then a profit is made. The maximum profit of made is when $S_T = 60$. This profit corresponds to $S_T - K_1 - cost = 60 - 40 - 4 = 16$ or equivalently, $K_3 - S_T - cost = 80 - 60 - 4 = 16$.

20 Straddle

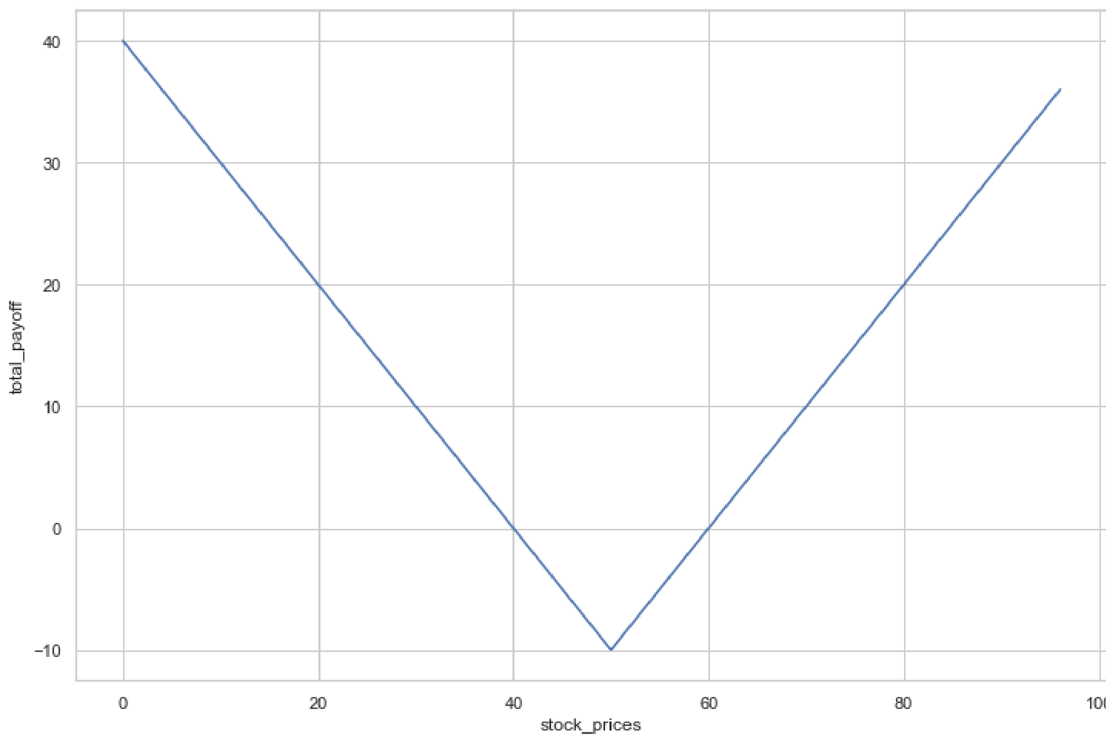
- Straddle is setup by buying a call and a put with same K and expiration date.
- Straddles are profitable if there is a high movement in the stock prices from the strike price. If the stock price tomorrow is close to K, the investor would incur a loss.

S_T range	long call payoff	long put payoff	total payoff
$S_T \leq K$	0	$K - S_T$	$K - S_T$
$S_T > K$	$S_T - K$	0	$S_T - K$

Payoffs for a straddle Construct a straddle by going long on a call with price $c_1 = 4$ and a put with $p_1 = 6$. Both have the same strike price $K = 50$ and expiration date. $S_0 = 48$ today.

[226]:
```
# Call parameters
c1_units, c1_price, c1_strike = 1,4,50
c2_units, c2_price, c2_strike = 0,0,0
c3_units, c3_price, c3_strike = 0,0,0

# Put parameters
p1_units, p1_price, p1_strike = 1,6,50
p2_units, p2_price, p2_strike = 0,0,0
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 0,48

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,␣
↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,␣
↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
-------------------------- Strategy -------------------------------
---------------------------------------------------------------------
Options |       Units     |       Premium |       Strike   | Setup Cost
---------------------------------------------------------------------
Call 1  |        1        |        4      |        50      |        -4
Call 2  |        0        |        0      |        0       |        0
Call 3  |        0        |        0      |        0       |        0
Put 1   |        1        |        6      |        50      |        -6
Put 2   |        0        |        0      |        0       |        0
Put 3   |        0        |        0      |        0       |        0
Stock   |        0        |        48     |        --      |        0
---------------------------------------------------------------------
Initial Setup Cost                                       -10


Payoffs from this strategy are shown:
```



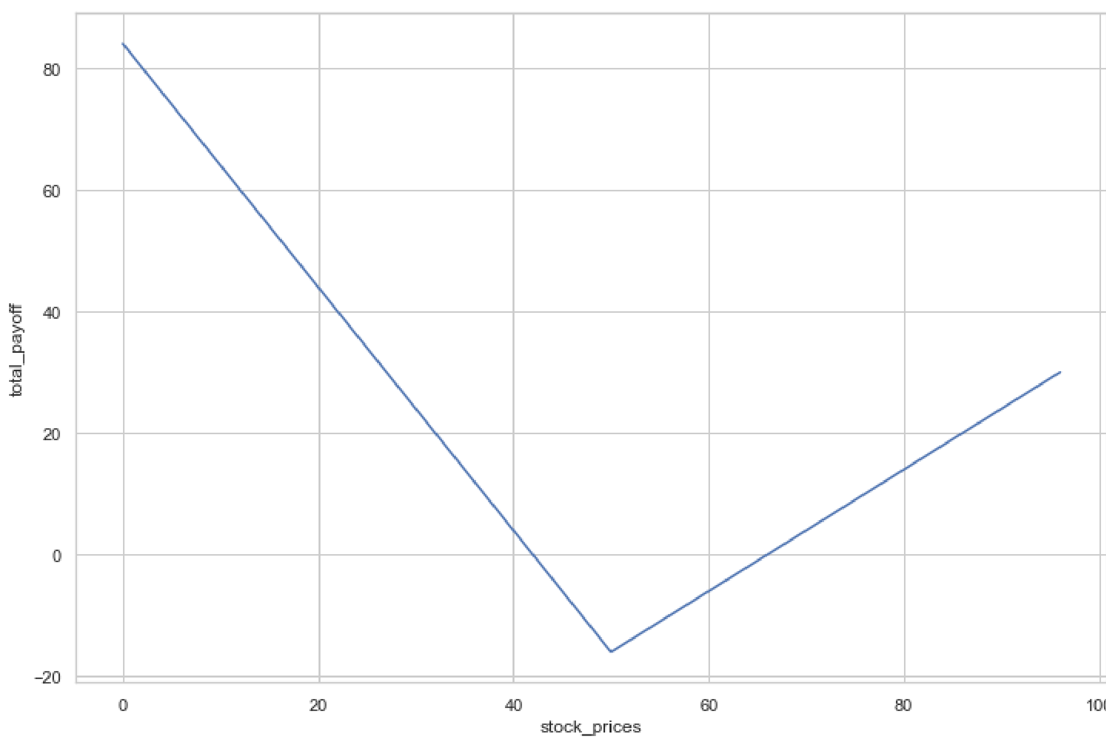
21 Strips and Straps

- Strips comprise a long position in one call and 2 puts, with the same expiration date and strike price.
- Straps comprise a long position in one put and 2 calls, with the same expiration date and strike price.
- These are essentially modifications on the straddle. One invests in strips if one considers that a decrease in stock prices are more likely. This also explains a higher weight on puts.
- The opposite is true for straps. The investor considers that there would be a large movement in stock prices in the upside, rather than the downside. Hence the higher weight on calls.

[227]:
```
# strips
# Call parameters
c1_units, c1_price, c1_strike = 1,4,50
c2_units, c2_price, c2_strike = 0,0,0
c3_units, c3_price, c3_strike = 0,0,0

# Put parameters
p1_units, p1_price, p1_strike = 2,6,50
p2_units, p2_price, p2_strike = 0,0,0
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
```

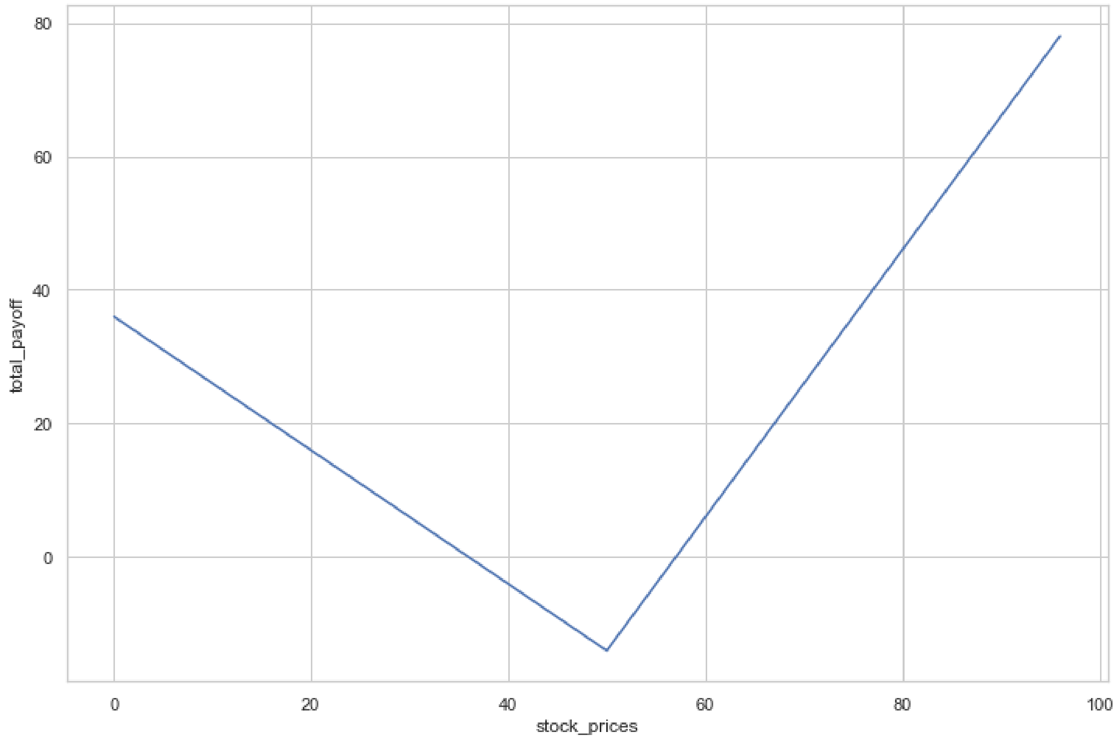
```
stock_units, stock_price = 0,48

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,
 ↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,
 ↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  -------------------------------
---------------------------------------------------------------------
Options |        Units     |       Premium |        Strike   | Setup Cost
---------------------------------------------------------------------
Call 1  |         1        |        4      |         50      |         -4
Call 2  |         0        |        0      |         0       |         0
Call 3  |         0        |        0      |         0       |         0
Put 1   |         2        |        6      |         50      |         -12
Put 2   |         0        |        0      |         0       |         0
Put 3   |         0        |        0      |         0       |         0
Stock   |         0        |        48     |         --      |         0
---------------------------------------------------------------------
Initial Setup Cost                                              -16


Payoffs from this strategy are shown:
```



[228]:
```python
# Call parameters
c1_units, c1_price, c1_strike = 2,4,50
c2_units, c2_price, c2_strike = 0,0,0
c3_units, c3_price, c3_strike = 0,0,0

# Put parameters
p1_units, p1_price, p1_strike = 1,6,50
p2_units, p2_price, p2_strike = 0,0,0
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 0,48

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,␣
 ↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,␣
 ↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  -------------------------------
---------------------------------------------------------------------
Options |       Units    |       Premium |       Strike   | Setup Cost
---------------------------------------------------------------------
Call 1  |        2       |        4      |        50      |        -8
Call 2  |        0       |        0      |        0       |        0
Call 3  |        0       |        0      |        0       |        0
Put 1   |        1       |        6      |        50      |        -6
Put 2   |        0       |        0      |        0       |        0
Put 3   |        0       |        0      |        0       |        0
Stock   |        0       |        48     |        --      |        0
---------------------------------------------------------------------
Initial Setup Cost                                         -14


Payoffs from this strategy are shown:
```



22 Strangle

- Here, one buys a put and a call with same expiration date, but with different strike prices.
- The call strike price is K_2 and the put strike price is K_1. The payoffs are shown below.
- Much like the case with a straddle, an investor takes up this strategy if they expect a large movement in stock prices but are unsure of the direction of this change.
- The key difference here is that the downside risk is lesser if the stock price takes a central value. Rather than a singular central value K about which loss is incurred in a straddle, here in a strangle we find that the downside occurs over a range of central values between K_1 and K_2. This is also called a bottom vertical combination.

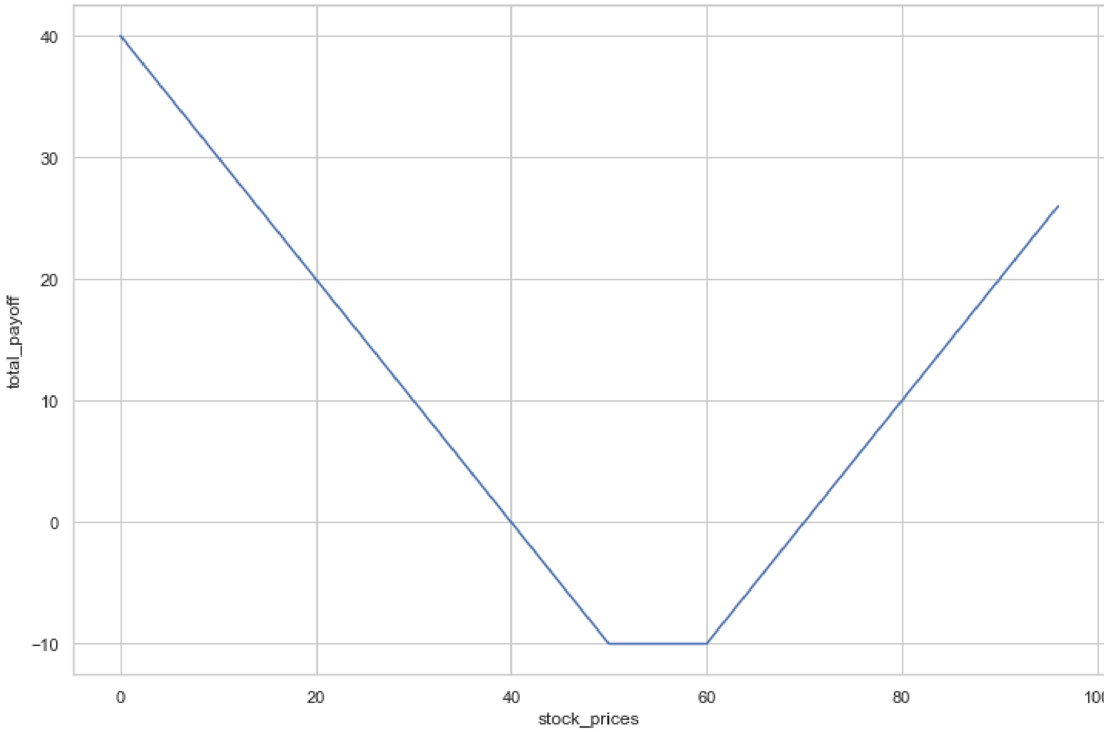
S_T range	long call payoff	long put payoff	total payoff
$S_T \leq K_1$	0	$K_1 - S_T$	$K_1 - S_T$
$K_1 < S_T < K_2$	0	0	0
$S_T \geq K_2$	$S_T - K_2$	0	$S_T - K_2$

Payoffs for a strangle

[229]:
```
# Call parameters
c1_units, c1_price, c1_strike = 1,4,60
c2_units, c2_price, c2_strike = 0,0,0
c3_units, c3_price, c3_strike = 0,0,0
```

```python
# Put parameters
p1_units, p1_price, p1_strike = 1,6,50
p2_units, p2_price, p2_strike = 0,0,0
p3_units, p3_price, p3_strike = 0,0,0

# Stock parameters
stock_units, stock_price = 0,48

payoff_calculator(c1_units, c1_price, c1_strike, c2_units, c2_price,
 ↪c2_strike,c3_units, c3_price, c3_strike,
                      p1_units, p1_price, p1_strike, p2_units, p2_price,
 ↪p2_strike,p3_units, p3_price, p3_strike,
                      stock_units, stock_price)
```

```
--------------------------  Strategy  -------------------------------
---------------------------------------------------------------------
Options |       Units     |       Premium |       Strike   | Setup Cost
---------------------------------------------------------------------
Call 1  |        1        |        4      |        60      |        -4
Call 2  |        0        |        0      |        0       |        0
Call 3  |        0        |        0      |        0       |        0
Put 1   |        1        |        6      |        50      |        -6
Put 2   |        0        |        0      |        0       |        0
Put 3   |        0        |        0      |        0       |        0
Stock   |        0        |        48     |        --      |        0
---------------------------------------------------------------------
Initial Setup Cost                                         -10


Payoffs from this strategy are shown:
```


total_payoff
stock_prices
40
30
20
10
0
−10
0
20
40
60
80
100