Binomial Model

March 21, 2017

Alexander Lerma

Zhiwei Zhang

Anoopa Sundararajan

Landi Gu

1 1. Programming Binomial Model

```
In [14]: from model.binomial import EuropeanVanillaModel, AmericanModel
         import pandas as pd

         #for filling column headers
         columns = lambda x: ['t = ' + str(i) for i in range(x + 1)]
```

1.0.1 Vanilla European Call Example

```
In [15]: european_call = EuropeanVanillaModel(20, 22, 3, 1, 1.2, .8, .08, 3, 'call'
```

Tree of stock prices **Note:** stock trees and option trees are computed row by row in our model.binomial module and appear to grow downward. To make the DataFrame output look similar to the way we draw trees by hand, we transpose the tree matrices so they appear in order by column instead.

```
In [16]: st_df = pd.DataFrame(european_call.s_tree.T, columns=columns(european_call
         st_df
```

Out[16]:

	t = 0	t = 1	t = 2	t = 3
0	20.0	24.0	28.8	34.56
1	NaN	16.0	19.2	23.04
2	NaN	NaN	12.8	15.36
3	NaN	NaN	NaN	10.24

Tree of derivative prices

```
In [17]: ot_df = pd.DataFrame(european_call.o_tree.T, columns=columns(european_call
         ot_df
```

Out[17]:

	t = 0	t = 1	t = 2	t = 3
0	3.868797	5.734561	8.491440	12.56
1	NaN	0.444508	0.679918	1.04
2	NaN	NaN	0.000000	0.00
3	NaN	NaN	NaN	0.00

Optimal price at t=0

```
In [18]: european_call.price()
```

```
Out[18]: 3.8687965915200624
```

1.0.2 Vanilla European Put Example

```
In [28]: european_put = EuropeanVanillaModel(80, 100, 2, 1, 1.2, .8, 0.04, 3, optio
```

Tree of stock prices

```
In [29]: st_df = pd.DataFrame(european_put.s_tree.T, columns=columns(european_put.
         st_df
```

Out[29]:

	t = 0	t = 1	t = 2	t = 3
0	80.0	96.0	115.2	138.24
1	NaN	64.0	76.8	92.16
2	NaN	NaN	51.2	61.44
3	NaN	NaN	NaN	40.96

Tree of derivative prices

```
In [31]: ot_df = pd.DataFrame(european_put.o_tree.T, columns=columns(european_put.
         ot_df
```

Out[31]:

	t = 0	t = 1	t = 2	t = 3
0	16.092361	9.105629	2.997768	0.00
1	NaN	28.311635	19.278944	7.84
2	NaN	NaN	44.878944	38.56
3	NaN	NaN	NaN	59.04

Optimal price at t=0

```
In [32]: european_put.price()
```

```
Out[32]: 16.092360501520012
```

1.0.3 American Put Example

```
In [33]: american_put = AmericanModel(80, 100, 2, 1, 1.2, .8, 0.04, 3, option='put'
```

Tree of stock prices

```
In [34]: st_df = pd.DataFrame(american_put.s_tree.T, columns=columns(american_put.r
         st_df
```

Out[34]:

	t = 0	t = 1	t = 2	t = 3
0	80.0	96.0	115.2	138.24
1	NaN	64.0	76.8	92.16
2	NaN	NaN	51.2	61.44
3	NaN	NaN	NaN	40.96

Tree of derivative prices

```
In [35]: ot_df = pd.DataFrame(american_put.o_tree.T, columns=columns(american_put.r
         ot_df
```

Out[35]:

	t = 0	t = 1	t = 2	t = 3
0	20.0	10.604917	2.997768	0.00
1	NaN	36.000000	23.200000	7.84
2	NaN	NaN	48.800000	38.56
3	NaN	NaN	NaN	59.04

Optimal price at t=0

```
In [26]: american_put.price()
```

```
Out[26]: 20.0
```

```
In [27]: american_call = AmericanModel(80, 100, 2, 1, 1.2, .8, 0.05, 3, 'call')
         american_call.price()
```

```
Out[27]: 8.1587070888752518
```