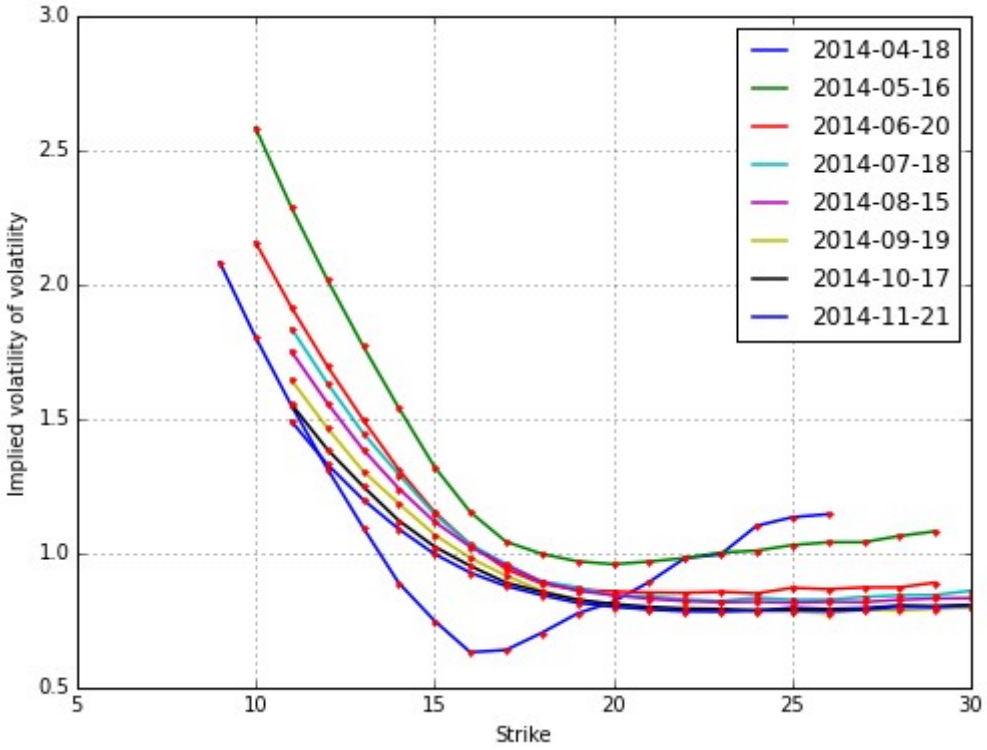
```
Python 2.7.10 |Anaconda 2.3.0 (64-bit)| (default, May 28 2015, 16:44:52) [MSC v.1500
64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 3.2.0 -- An enhanced Interactive Python.
Anaconda is brought to you by Continuum Analytics.
Please check out: http://continuum.io/thanks and https://anaconda.org
?         -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
help      -> Python's own help system.
object?   -> Details about 'object', use 'object??' for extra details.
%guiref   -> A brief reference about the graphical user interface.

In [1]: V0=17.6639

In [2]: r=0.01

In [3]: import pandas as pd

In [4]: h5=pd.HDFStore('./source/vstoxx_data_31032014.h5')
   ...: futures_data=h5['futures_data']
   ...: options_data=h5['options_data']
   ...: h5.close()
   ...:

In [5]: futures_data
Out[5]:
          DATE  EXP_YEAR  EXP_MONTH  PRICE   MATURITY    TTM
496 2014-03-31      2014          4  17.85 2014-04-18  0.049
497 2014-03-31      2014          5  19.55 2014-05-16  0.126
498 2014-03-31      2014          6  19.95 2014-06-20  0.222
499 2014-03-31      2014          7  20.40 2014-07-18  0.299
500 2014-03-31      2014          8  20.70 2014-08-15  0.375
501 2014-03-31      2014          9  20.95 2014-09-19  0.471
502 2014-03-31      2014         10  21.05 2014-10-17  0.548
503 2014-03-31      2014         11  21.25 2014-11-21  0.644

In [6]: options_data.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 395 entries, 46170 to 46564
Data columns (total 8 columns):
DATE         395 non-null datetime64[ns]
EXP_YEAR     395 non-null int64
EXP_MONTH    395 non-null int64
TYPE         395 non-null object
STRIKE       395 non-null float64
PRICE        395 non-null float64
MATURITY     395 non-null datetime64[ns]
TTM          395 non-null float64
dtypes: datetime64[ns](2), float64(3), int64(2), object(1)
memory usage: 27.8+ KB

In [7]: options_data[['DATE','MATURITY','TTM','STRIKE','PRICE']].head()
Out[7]:
            DATE   MATURITY    TTM  STRIKE  PRICE
46170 2014-03-31 2014-04-18  0.049       1  16.85
46171 2014-03-31 2014-04-18  0.049       2  15.85
46172 2014-03-31 2014-04-18  0.049       3  14.85
46173 2014-03-31 2014-04-18  0.049       4  13.85
46174 2014-03-31 2014-04-18  0.049       5  12.85

In [8]: options_data['IMP_VOL']=0.0
```

```
In [9]: tol=0.5
   ...: for option in options_data.index:
   ...:     forward=futures_data[futures_data['MATURIY']==options_data.loc[option]
['MATURITY']]['PRICE'].values[0]
   ...:     if(forward*(1-tol)<options_data.loc[option]['STRIKE']<forward*(1+tol)):
   ...:         imp_vol=bsm_call_imp_vol(
   ...:         V0,
   ...:         options_data.loc[option]['STRIKE'],
   ...:         options_data.loc[option]['TTM'],
   ...:         r,
   ...:         options_data.loc[option]['PRICE'],
   ...:         sigma_est=2.,
   ...:         it=100)
   ...:         options_data['IMP_VOL'].loc[option]=imp_vol
   ...:
Traceback (most recent call last):

  File "<ipython-input-9-7c193cec3c47>", line 3, in <module>
    forward=futures_data[futures_data['MATURIY']==options_data.loc[option]
['MATURITY']]['PRICE'].values[0]

  File "C:\Anaconda\lib\site-packages\pandas\core\frame.py", line 1797, in __getitem__
    return self._getitem_column(key)

  File "C:\Anaconda\lib\site-packages\pandas\core\frame.py", line 1804, in
_getitem_column
    return self._get_item_cache(key)

  File "C:\Anaconda\lib\site-packages\pandas\core\generic.py", line 1084, in
_get_item_cache
    values = self._data.get(item)

  File "C:\Anaconda\lib\site-packages\pandas\core\internals.py", line 2851, in get
    loc = self.items.get_loc(item)

  File "C:\Anaconda\lib\site-packages\pandas\core\index.py", line 1572, in get_loc
    return self._engine.get_loc(_values_from_object(key))

  File "pandas\index.pyx", line 134, in pandas.index.IndexEngine.get_loc
(pandas\index.c:3824)

  File "pandas\index.pyx", line 154, in pandas.index.IndexEngine.get_loc
(pandas\index.c:3704)

  File "pandas\hashtable.pyx", line 686, in
pandas.hashtable.PyObjectHashTable.get_item (pandas\hashtable.c:12280)

  File "pandas\hashtable.pyx", line 694, in
pandas.hashtable.PyObjectHashTable.get_item (pandas\hashtable.c:12231)

KeyError: 'MATURIY'


In [10]: tol=0.5
    ...: for option in options_data.index:
    ...:     forward=futures_data[futures_data['MATURIIY']==options_data.loc[option]
['MATURITY']]['PRICE'].values[0]
    ...:     if(forward*(1-tol)<options_data.loc[option]['STRIKE']<forward*(1+tol)):
    ...:         imp_vol=bsm_call_imp_vol(
    ...:         V0,
    ...:         options_data.loc[option]['STRIKE'],
```

```
    ...:         options_data.loc[option]['TTM'],
    ...:         r,
    ...:         options_data.loc[option]['PRICE'],
    ...:         sigma_est=2.,
    ...:         it=100)
    ...:         options_data['IMP_VOL'].loc[option]=imp_vol
    ...:
Traceback (most recent call last):

  File "<ipython-input-10-65c6e40226e6>", line 3, in <module>
    forward=futures_data[futures_data['MATURIIY']==options_data.loc[option]
['MATURITY']]['PRICE'].values[0]

  File "C:\Anaconda\lib\site-packages\pandas\core\frame.py", line 1797, in __getitem__
    return self._getitem_column(key)

  File "C:\Anaconda\lib\site-packages\pandas\core\frame.py", line 1804, in
_getitem_column
    return self._get_item_cache(key)

  File "C:\Anaconda\lib\site-packages\pandas\core\generic.py", line 1084, in
_get_item_cache
    values = self._data.get(item)

  File "C:\Anaconda\lib\site-packages\pandas\core\internals.py", line 2851, in get
    loc = self.items.get_loc(item)

  File "C:\Anaconda\lib\site-packages\pandas\core\index.py", line 1572, in get_loc
    return self._engine.get_loc(_values_from_object(key))

  File "pandas\index.pyx", line 134, in pandas.index.IndexEngine.get_loc
(pandas\index.c:3824)

  File "pandas\index.pyx", line 154, in pandas.index.IndexEngine.get_loc
(pandas\index.c:3704)

  File "pandas\hashtable.pyx", line 686, in
pandas.hashtable.PyObjectHashTable.get_item (pandas\hashtable.c:12280)

  File "pandas\hashtable.pyx", line 694, in
pandas.hashtable.PyObjectHashTable.get_item (pandas\hashtable.c:12231)

KeyError: 'MATURIIY'


In [11]: tol=0.5
    ...: for option in options_data.index:
    ...:     forward=futures_data[futures_data['MATURITY']==options_data.loc[option]
['MATURITY']]['PRICE'].values[0]
    ...:     if(forward*(1-tol)<options_data.loc[option]['STRIKE']<forward*(1+tol)):
    ...:         imp_vol=bsm_call_imp_vol(
    ...:         V0,
    ...:         options_data.loc[option]['STRIKE'],
    ...:         options_data.loc[option]['TTM'],
    ...:         r,
    ...:         options_data.loc[option]['PRICE'],
    ...:         sigma_est=2.,
    ...:         it=100)
    ...:         options_data['IMP_VOL'].loc[option]=imp_vol
    ...:
Traceback (most recent call last):
```

```
  File "<ipython-input-11-4cb558afb5db>", line 5, in <module>
    imp_vol=bsm_call_imp_vol(

NameError: name 'bsm_call_imp_vol' is not defined


In [12]: from bsm_functions import *

In [13]: tol=0.5
    ...: for option in options_data.index:
    ...:     forward=futures_data[futures_data['MATURITY']==options_data.loc[option]
['MATURITY']]['PRICE'].values[0]
    ...:     if(forward*(1-tol)<options_data.loc[option]['STRIKE']<forward*(1+tol)):
    ...:         imp_vol=bsm_call_imp_vol(
    ...:         V0,
    ...:         options_data.loc[option]['STRIKE'],
    ...:         options_data.loc[option]['TTM'],
    ...:         r,
    ...:         options_data.loc[option]['PRICE'],
    ...:         sigma_est=2.,
    ...:         it=100)
    ...:         options_data['IMP_VOL'].loc[option]=imp_vol
    ...:
C:\Anaconda\lib\site-packages\pandas\core\indexing.py:115: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  self._setitem_with_indexer(indexer, value)

In [14]: options_data.loc[46170]
Out[14]:
DATE         2014-03-31 00:00:00
EXP_YEAR                    2014
EXP_MONTH                      4
TYPE                           C
STRIKE                         1
PRICE                      16.85
MATURITY     2014-04-18 00:00:00
TTM                       0.049
IMP_VOL                        0
Name: 46170, dtype: object

In [15]: plot_data=options_data[options_data['IMP_VOL']>0]

In [16]: maturities=sorted(set(options_data['MATURITY']))

In [17]: maturities
Out[17]:
[Timestamp('2014-04-18 00:00:00'),
 Timestamp('2014-05-16 00:00:00'),
 Timestamp('2014-06-20 00:00:00'),
 Timestamp('2014-07-18 00:00:00'),
 Timestamp('2014-08-15 00:00:00'),
 Timestamp('2014-09-19 00:00:00'),
 Timestamp('2014-10-17 00:00:00'),
 Timestamp('2014-11-21 00:00:00')]

In [18]: import matplotlib.pyplot as plt
    ...: %matplotlib inline
    ...: plt.figure(figsize=(8,6))
    ...: for maturity in maturities:
```

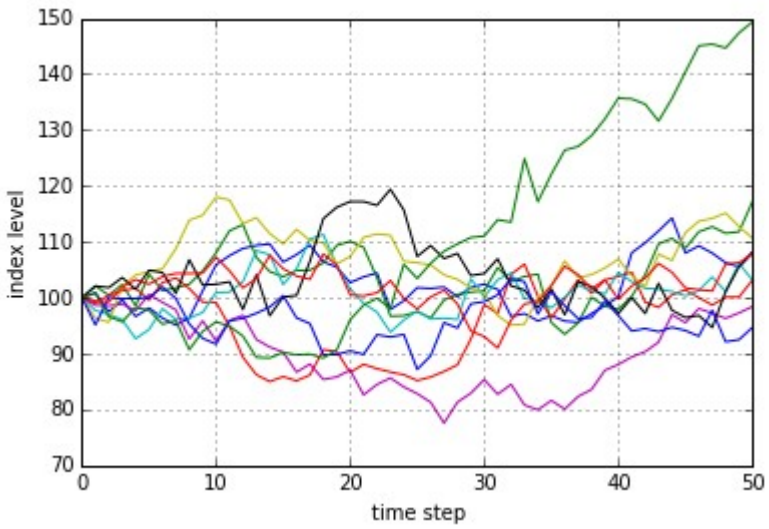
```
    ...:     data=plot_data[option_data.MATURITY==maturity]
    ...:     plt.plot(data['STRIKE'],data['IMP_VOL'],
    ...:           label=maturity.date(),lw=1.5)
    ...:     plt.plot(data['STRIKE'],data['IMP_VOL'],'r.')
    ...: plt.grid(True)
    ...: plt.xlabel('Strike')
    ...: plt.ylabel('Implied volatility of volatility')
    ...: plt.legend()
    ...: plt.show()
    ...:
<matplotlib.figure.Figure at 0x156a1fd0>
Traceback (most recent call last):

  File "<ipython-input-18-63c5a8f9ef0c>", line 5, in <module>
    data=plot_data[option_data.MATURITY==maturity]

NameError: name 'option_data' is not defined


In [19]: import matplotlib.pyplot as plt
    ...: %matplotlib inline
    ...: plt.figure(figsize=(8,6))
    ...: for maturity in maturities:
    ...:     data=plot_data[options_data.MATURITY==maturity]
    ...:     plt.plot(data['STRIKE'],data['IMP_VOL'],
    ...:           label=maturity.date(),lw=1.5)
    ...:     plt.plot(data['STRIKE'],data['IMP_VOL'],'r.')
    ...: plt.grid(True)
    ...: plt.xlabel('Strike')
    ...: plt.ylabel('Implied volatility of volatility')
    ...: plt.legend()
    ...: plt.show()
    ...:
C:\Anaconda\lib\site-packages\pandas\core\frame.py:1825: UserWarning: Boolean Series
key will be reindexed to match DataFrame index.
  "DataFrame index.", UserWarning)
```



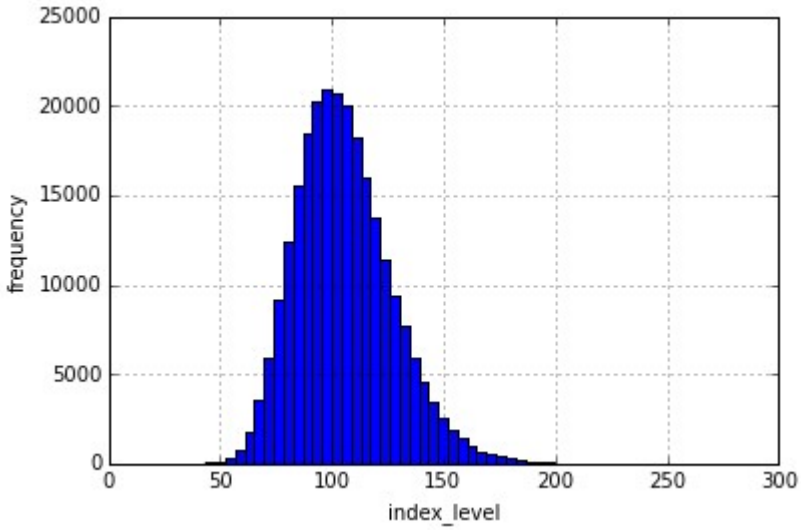

```
In [20]: keep=['PRICE','IMP_VOL']

In [21]: group_data=plot_data.groupby(['MATURITY','STRIKE'])[keep]

In [22]: group_data
Out[22]: <pandas.core.groupby.DataFrameGroupBy object at 0x0000000015C379E8>

In [23]: group_data=group_data.sum()

In [24]: group_data
Out[24]:
                      PRICE   IMP_VOL
MATURITY   STRIKE
2014-04-18 9           8.85  2.083388
           10          7.85  1.804193
           11          6.85  1.550283
           12          5.85  1.316103
           13          4.85  1.097184
           14          3.85  0.889581
           15          2.90  0.748630
           16          2.00  0.630958
           17          1.35  0.639297
           18          0.95  0.703208
           19          0.70  0.775629
           20          0.50  0.818813
           21          0.40  0.892494
           22          0.35  0.981551
           23          0.25  0.995124
           24          0.25  1.101613
           25          0.20  1.133795
           26          0.15  1.144977
2014-05-16 10          9.55  2.583783
           11          8.55  2.287971
           12          7.55  2.019846
           13          6.55  1.772845
```

```
            14       5.55  1.541737
            15       4.55  1.321948
            16       3.65  1.153127
            17       2.90  1.042663
            18       2.35  0.997287
            19       1.90  0.969408
            20       1.55  0.958881
            21       1.30  0.968536
...                   ...       ...
2014-10-17 21       3.05  0.799966
            22       2.75  0.793566
            23       2.50  0.791992
            24       2.25  0.785962
            25       2.10  0.795640
            26       1.90  0.791735
            27       1.75  0.794615
            28       1.65  0.805125
            29       1.50  0.802227
            30       1.40  0.807714
2014-11-21 11      10.25  1.491546
            12       9.25  1.331759
            13       8.30  1.200063
            14       7.40  1.090057
            15       6.55  0.997041
            16       5.80  0.927519
            17       5.15  0.877302
            18       4.60  0.843453
            19       4.10  0.815015
            20       3.70  0.800333
            21       3.35  0.790023
            22       3.05  0.784194
            23       2.80  0.783116
            24       2.60  0.787200
            25       2.40  0.787899
            26       2.20  0.785411
            27       2.05  0.789241
            28       1.95  0.800016
            29       1.80  0.798958
            30       1.70  0.805459

[158 rows x 2 columns]

In [25]: from bsm_functions import bam_call_value
Traceback (most recent call last):

  File "<ipython-input-25-a29524b20ef1>", line 1, in <module>
    from bsm_functions import bam_call_value

ImportError: cannot import name bam_call_value


In [26]: from bsm_functions import bsm_call_value

In [27]: S0=100
    ...: K=105
    ...: T=1.0
    ...: R=0.05
    ...: sigma=0.2
    ...: bsm_call_value(S0,K,T,r,sigma)
    ...:
Out[27]: 6.2972545390860333
```

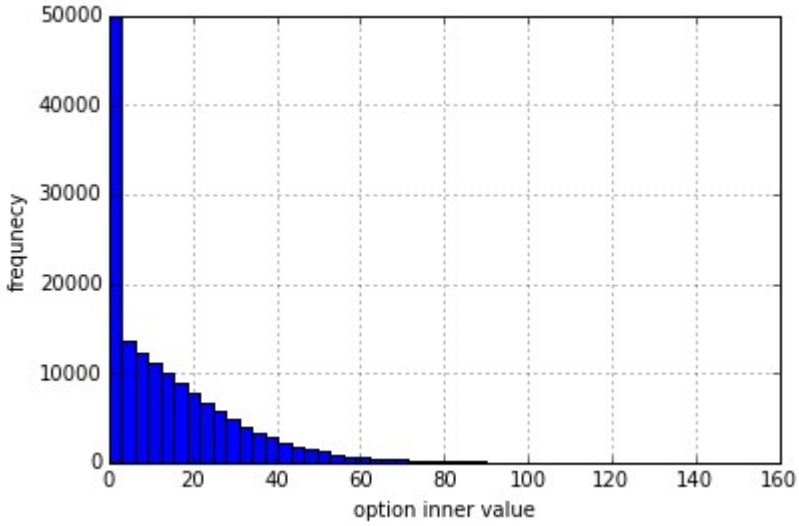
```
In [28]: %run mcs_pure_python.py
Traceback (most recent call last):

  File "C:\Users\Ďocuments\Python Scripts\mcs_pure_python.py", line 32, in <module>
    print "European Option Value &7.3f" % C0

TypeError: not all arguments converted during string formatting


In [29]: %run mcs_pure_python.py
10.420999262
27.9830000401

In [30]: import numpy as np

In [31]: v=np.arange(1,6)

In [32]: v
Out[32]: array([1, 2, 3, 4, 5])

In [33]: 2*v
Out[33]: array([ 2,  4,  6,  8, 10])

In [34]: %run mcs_vector_numpy.py
8.03650296251
Traceback (most recent call last):

  File "C:\Users\Ďocuments\Python Scripts\mcs_vector_numpy.py", line 23, in <module>
    print tnp1

NameError: name 'tnp1' is not defined


In [35]: %run mcs_vector_numpy.py
8.03650296251
0.797999858856

In [36]: %run mcs_full_vector_numpy.py
8.16580796626
0.851999998093

In [37]: import matplotlib.pyplot as plt

In [38]: plt.plot(S[:,:10])\
    ...: plt.grid(True)
  File "<ipython-input-38-ddb81aa3585f>", line 1
    plt.plot(S[:,:10])plt.grid(True)
                      ^
SyntaxError: invalid syntax


In [39]: plt.plot(S[:,:10])
    ...: plt.grid(True)
    ...: plt.xlabel('time step')
    ...: plt.ylabel('index level')
    ...:
Out[39]: <matplotlib.text.Text at 0x15d51b00>
```




```
In [40]: plt.hist(S[-1],bins=50)
    ...: plt.grid(True)
    ...: plt.xlabel('index_level')
    ...: plt.ylabel('frequency')
    ...:
Out[40]: <matplotlib.text.Text at 0x151d3da0>
```




```
In [41]: plt.hist(np.maximum(S[-1]-K,0),bins=50)\
    ...: plt.grid(True)
  File "<ipython-input-41-50305ff4a9d5>", line 1
    plt.hist(np.maximum(S[-1]-K,0),bins=50)plt.grid(True)
                                           ^
SyntaxError: invalid syntax


In [42]: plt.hist(np.maximum(S[-1]-K,0),bins=50)
    ...: plt.grid(True)
    ...: plt.xlabel('option inner value')
    ...: plt.ylabel('frequnecy')
    ...: plt.ylim(0,50000)
    ...:
Out[42]: (0, 50000)
```




```
In [43]: sum(S[-1]<K)
Out[43]: 133533

In [44]:
```