Stock_Kelly_Arithmetic_Fraction_Chart

September 29, 2021

1 Stock Kelly Arithmetic Chart

```python
[1]: # Library
     import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt

     import warnings
     warnings.filterwarnings("ignore")

     from pandas_datareader import data as pdr
     import yfinance as yf
     yf.pdr_override()
```

```python
[2]: start = '2019-01-01' #input
     end = '2020-07-01' #input
     symbol = 'AMD' #input
```

```python
[3]: stocks = yf.download(symbol, start=start, end=end)['Adj Close']
```

```
[*********************100%**********************]  1 of 1 completed
```

```python
[4]: stocks_returns = stocks.pct_change().dropna()
```

```python
[5]: def kelly_fraction(stock_returns):
         # returns = np.array(stock_returns)
         wins = stock_returns[stock_returns > 0]
         losses = stock_returns[stock_returns <= 0]
         W = len(wins) / len(stock_returns)
         R = np.mean(wins) / np.abs(np.mean(losses))
         kelly_f = W - ( (1 - W) / R )
         return kelly_f

     def expected_arith(stock_returns):
         expected_arith = np.mean(stock_returns)
         return expected_arith
```

```python
def kelly_arithmetic(stock_returns):
    bounded_rets = stock_returns / np.abs(np.min(stock_returns))
    kelly_f = kelly_fraction(bounded_rets) / np.abs(np.min(stock_returns))
    exp_arith_kelly = expected_arith(bounded_rets * kelly_f)
    return exp_arith_kelly
```

[6]:
```python
# Compute the running Kelly Arithmetic
running = [kelly_arithmetic(stocks_returns[i-90:i]) for i in range(90,
 ↪len(stocks_returns))]

# Plot running Kelly Arithmetic up to 100 days before the end of the data set
_, ax1 = plt.subplots(figsize=(12,8))
ax1.plot(range(90, len(stocks_returns)-100), running[:-100])
ticks = ax1.get_xticks()
ax1.set_xticklabels([stocks.index[int(i)].date() for i in ticks[:-1]]) # Label
 ↪x-axis with dates
plt.title(symbol + ' Kelly Arithmetic')
plt.xlabel('Date')
plt.ylabel('Kelly Arithmetic')
```

[6]: Text(0, 0.5, 'Kelly Arithmetic')
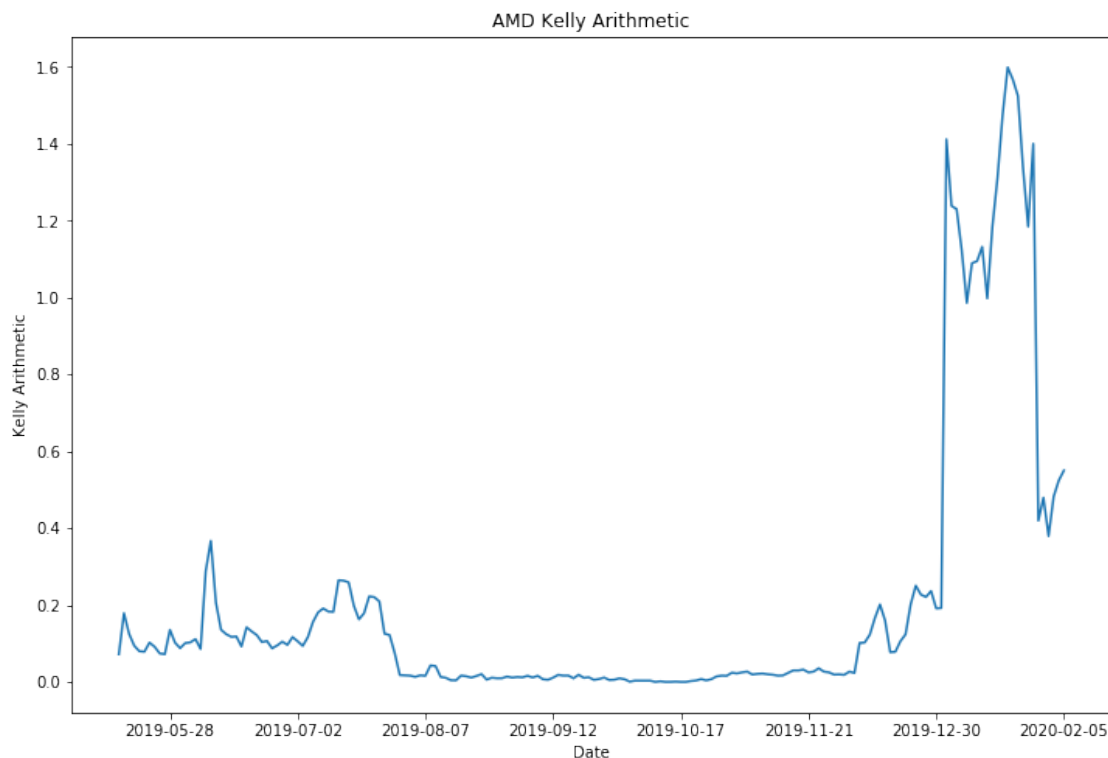
```
[7]: ka = kelly_arithmetic(stocks_returns)
     ka
```

[7]: 0.020292436157276394

```
[8]: running
```

[8]: [0.07195786408206359,
      0.1786418583151776,
      0.12511125614176008,
      0.0942875772241262,
      0.07981019184978605,
      0.07849674685689145,
      0.10219547968968515,
      0.09131369909193927,
      0.07392377390398866,
      0.07219793396727854,
      0.13536508734634795,
      0.10192384057688521,
      0.08775978061030573,
      0.10096534243710682,
      0.10280690416834098,
      0.11137888757100876,
      0.08561530637159809,
      0.28773286955295724,
      0.3669577569980651,
      0.20646342310957383,
      0.1356927974964329,
      0.1242092299690515,
      0.1172136037952894,
      0.11810675406879179,
      0.09212721725860534,
      0.14197667434172248,
      0.1313127687812458,
      0.12169099077332221,
      0.10361455937133414,
      0.10625771959037278,
      0.08725317016835128,
      0.09485962254910087,
      0.10458545120738069,
      0.09626829738301355,
      0.1165909155851589,
      0.10565962802076696,
      0.09356992839239327,
      0.11646834778755791,
      0.15592467948134217,
      0.18079819518793272,
```

0.19102954742452974,
0.1830432398891349,
0.1822391820305413,
0.2644420217113663,
0.2633563759144365,
0.2586965554979142,
0.19826517330004204,
0.1626190235243182,
0.17856990143877766,
0.22233151581951122,
0.22054208905366446,
0.20916014350478807,
0.1254813345612118,
0.12198110385663757,
0.07485950097818594,
0.017718061510089055,
0.01691805383429516,
0.01630643699988727,
0.013135520938724454,
0.017079217279639965,
0.015909688516819294,
0.04283414184415004,
0.04116382520575343,
0.012753444761267215,
0.01067804370165647,
0.004320150465862859,
0.004023898085548764,
0.016489519035093093,
0.014592257704963148,
0.011068751919988733,
0.015395389185295453,
0.02021336959793276,
0.005845379375924283,
0.010888416911007675,
0.009419253564233787,
0.00956099160391903,
0.013873669141720531,
0.01113427063082097,
0.012701017224818329,
0.011815480731464031,
0.015825179295183612,
0.011444105642227333,
0.015832050510810027,
0.00693412883895434,
0.005516254111657559,
0.010952227181814081,
0.01831426974664598,

0.016362412089584392,
0.01648783723938569,
0.00930652777863472,
0.018613090199841857,
0.010589246729721065,
0.0119961165235835,
0.005289606948004838,
0.007537330406485489,
0.011269535625780999,
0.0049780239377710984,
0.005683325008373149,
0.00935593665223794,
0.006852757042815747,
0.0005838358141982173,
0.003220899467026027,
0.003169700560002556,
0.0030016121730244323,
0.0031529140542647673,
2.6726178030195897e-06,
0.001513433918966624,
9.42027326313403e-08,
8.224713707392671e-05,
0.0006978139743514496,
7.304841368498665e-07,
4.506458499594422e-06,
0.002237054262031917,
0.003605249953892572,
0.007354808027349529,
0.003931498480177026,
0.0071375779633896606,
0.014238858605199363,
0.01626810414116634,
0.015828471851381157,
0.02388211831565298,
0.022112288275087662,
0.024801537691015035,
0.026808253619407887,
0.01922838479311343,
0.020889766075568965,
0.021743553188750035,
0.019848161374678746,
0.018649059997228445,
0.01628349850129296,
0.016801473216924146,
0.023064689027586446,
0.02980075533170627,
0.029759019780511053,

0.031990404418544914,
0.024640061826202875,
0.02675021837652115,
0.03537433657842346,
0.02675438088753492,
0.02481087163935758,
0.019020758573712757,
0.019488066321921203,
0.018321003474698633,
0.0267237258797974,
0.02322396703242857,
0.10121846687682101,
0.1023851780779156,
0.12259863930959602,
0.16605268436836126,
0.20137956193467352,
0.15941460436617266,
0.07721187709348806,
0.0782729075518857,
0.10583258650176036,
0.1235662049543891,
0.20148586894798462,
0.25022624320832965,
0.22718236826349666,
0.22075821508157015,
0.23647062606728073,
0.19122360378314582,
0.19214138949347365,
1.4113662117862105,
1.2380811406135563,
1.2296660160333845,
1.122832702501741,
0.9856623413125415,
1.0889162804509545,
1.0943420225737324,
1.1315837797834527,
0.9974950059959717,
1.183576309497582,
1.310772938377506,
1.473997340807877,
1.5983879512531327,
1.5669128213516967,
1.524282348476612,
1.3370600880522354,
1.1842820469885547,
1.399770442558231,
0.41930519298806374,

0.47913152557080424,
0.37934913916674107,
0.48219018873818265,
0.5239635866807029,
0.549888052792173,
0.5804051934231285,
0.5787607857958956,
0.6857433788132754,
0.7820090524853133,
0.7631301974018473,
0.7637414491074622,
0.807639633746803,
0.931955339244867,
0.9803595491739066,
0.9166375189925988,
0.48524579537316576,
0.26705783814164874,
0.2259272396321951,
0.2181788289917429,
0.14497162613759593,
0.1741633676383874,
0.1805326605742465,
0.1816120087807724,
0.24174788651164186,
0.19344890307682086,
0.17891107246719862,
0.042077369214241586,
0.06253066927800857,
0.06371840714245643,
0.010153554730208075,
0.02106808063432896,
0.004466862118290361,
0.011513587991510086,
0.006087447269069204,
0.006684861375025854,
0.0061370234695032125,
0.010190915334256435,
0.020418410412027285,
0.013640045520358633,
0.018001192425159292,
0.014888243166615779,
0.014223889578972857,
0.0066054681686107545,
0.00444261316536 2736,
0.008375746717497828,
0.005848367669394201,
0.013788636632487784,

0.01631990202375722,
0.017891655489421603,
0.01768034665472254,
0.025520962207239418,
0.03525782540034844,
0.03261594182886865,
0.03856667358625577,
0.03682232388330941,
0.041466329546262615,
0.027811751664829312,
0.035160960496065256,
0.024457237280151537,
0.030347454673528926,
0.027309620975813943,
0.02327187652028909,
0.020450271772439764,
0.016078735133007647,
0.008708111275315846,
0.010563691222162955,
0.008020112382934635,
0.0076847425264348805,
0.008101014214772955,
0.011278717287966237,
0.014792972940968636,
0.006191559341017553,
0.0049942462472061236,
0.00824344626539967,
0.008033786035599608,
0.00940765326937066,
0.008831594226035801,
0.011733589747718834,
0.008039519149430973,
0.009797096296743446,
0.006436406228210946,
0.004354844027187759,
0.002320164142964793,
0.003977308181851787,
0.003389878479492584,
0.003005320856015095,
0.0035223685637735145,
0.004536910996002425,
0.003701179624752453,
0.0070884058401930496,
0.009696918589923989,
0.014707607222399284,
0.006348059929451986,
0.005352050220652841,

```
    0.006172197220146772,
    0.00658533343270086,
    0.006181084834704824,
    0.0029736870895162113,
    0.001966609608925586,
    0.00226033521733872,
    0.001345318770526138,
    0.0003507683869932361,
    1.3479037005301112e-05,
    0.0006100488393304183,
    0.00015027058332057054]
```

[9]:
```python
print('Expected Value (Arithmetic): {}%'.format(np.
 ↪round(kelly_arithmetic(stocks_returns) * 100, 5)))
```

```
Expected Value (Arithmetic): 2.02924%
```