options

July 15, 2017

```
In [1]: import QuantLib as ql # version 1.5
        import matplotlib.pyplot as plt
        %matplotlib inline
```

```
In [2]: # option data
        maturity_date = ql.Date(15, 1, 2016)
        spot_price = 127.62
        strike_price = 130
        volatility = 0.20 # the historical vols for a year
        dividend_rate =  0.0163
        option_type = ql.Option.Call

        risk_free_rate = 0.001
        day_count = ql.Actual365Fixed()
        calendar = ql.UnitedStates()

        calculation_date = ql.Date(8, 5, 2015)
        ql.Settings.instance().evaluationDate = calculation_date
```

```
In [3]: # construct the European Option
        payoff = ql.PlainVanillaPayoff(option_type, strike_price)
        exercise = ql.EuropeanExercise(maturity_date)
        european_option = ql.VanillaOption(payoff, exercise)
```

```
In [4]: spot_handle = ql.QuoteHandle(
            ql.SimpleQuote(spot_price)
        )
        flat_ts = ql.YieldTermStructureHandle(
            ql.FlatForward(calculation_date, risk_free_rate, day_count)
        )
        dividend_yield = ql.YieldTermStructureHandle(
            ql.FlatForward(calculation_date, dividend_rate, day_count)
        )
        flat_vol_ts = ql.BlackVolTermStructureHandle(
            ql.BlackConstantVol(calculation_date, calendar, volatility, day_count)
        )
        bsm_process = ql.BlackScholesMertonProcess(spot_handle,
                                                   dividend_yield,
                                                   flat_ts,
                                                   flat_vol_ts)
```

```
In [5]: european_option.setPricingEngine(ql.AnalyticEuropeanEngine(bsm_process))
        bs_price = european_option.NPV()
        print("The theoretical price is ", bs_price)
```

```
The theoretical price is  6.749271812460607
```

```
In [ ]:
```