Not Over Thinking

Overnight Seasonality in Bitcoin

Algorithmic Trading Strategy with Full Code

Haixiang

2024.02 | Vol 71.

hxyan.2015@gmail.com | github.com/hxyan2020

STRATEGY & ECONOMIC RATIONALE

The investment universe consists of Bitcoin and the data are obtained from Gemini exchange. To exploit the seasonality, open a long position in the BTC at 22:00 (UTC +0) and hold it for two hours. The position is closed after the two hour holding period.

BUY	SELL
open a long position in the BTC at 22:00 (UTC +0) and ho ld it for two hours	The opposite

PARAMETER & VARIABLES

PARAMETER	VALUE
MARKETS TRADED	Crypto
FINANCIAL INSTRUMENTS	Cryptos
REGION	Global
PERIOD OF REBALANCING	Intraday
NO. OF TRADED INSTRUMENTS	1
WEIGHTING	Equal weighting
LOOKBACK PERIODS	N/A
LONG/SHORT	Long only

ALGORITHM

```
from AlgorithmImports import *# endregion
class OvernightSeasonalityinBitcoin(QCAlgorithm):

    def Initialize(self):
        self.SetStartDate(2016, 1, 1)
        self.SetCash(100000)

        # NOTE Coinbase Pro, CoinAPI, and Bitfinex data is all set in UTC Time. This means that
 when accessing data from this brokerage, all data will be time stamped in UTC Time.
        self.crypto = self.AddCrypto("BTCUSD", Resolution.Minute, Market.Bitfinex)
        self.crypto.SetLeverage(10)
        self.crypto.SetFeeModel(CustomFeeModel())
        self.crypto = self.crypto.Symbol

        self.open_trade_hour:int = 22
        self.close_trade_hour:int = 0

    def OnData(self, data):
        if self.crypto in data and data[self.crypto]:
            time:datetime.datetime = self.UtcTime

            # open long position
            if time.hour == self.open_trade_hour and time.minute == 0:
                self.SetHoldings(self.crypto, 1)

        # close position
        if time.hour == self.close_trade_hour and time.minute == 0:
            if self.Portfolio[self.crypto].Invested:
                self.Liquidate(self.crypto)
```

```python
class CustomFeeModel(FeeModel):
    def GetOrderFee(self, parameters):
        fee = parameters.Security.Price * parameters.Order.AbsoluteQuantity * 0.00005
        return OrderFee(CashAmount(fee, "USD"))
```

BACKTESTING PERFORMANCE

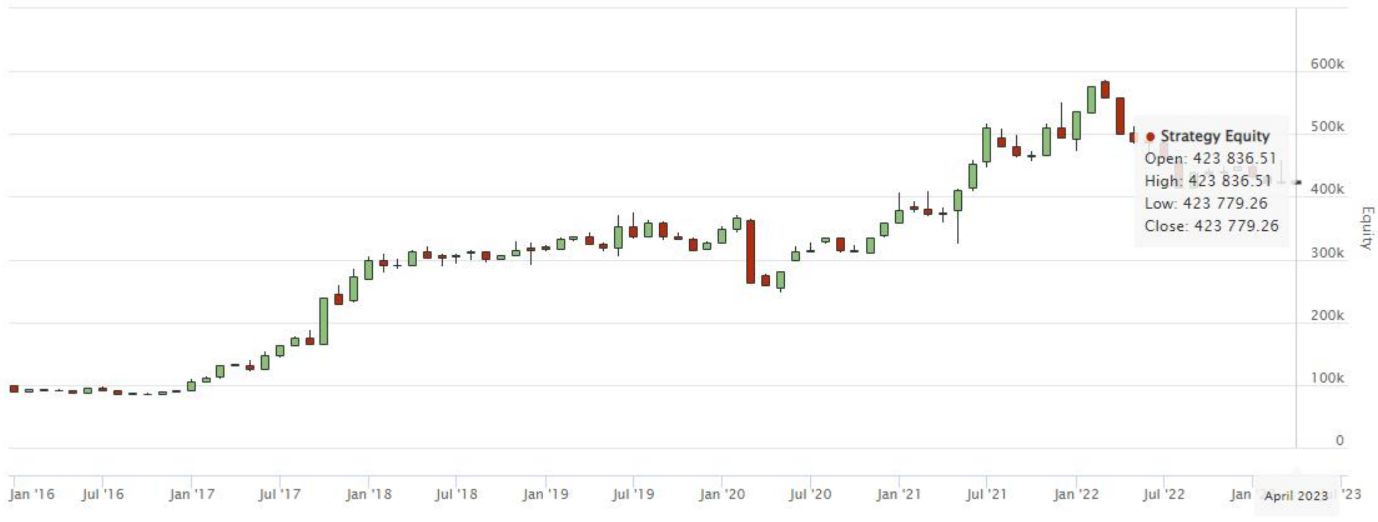
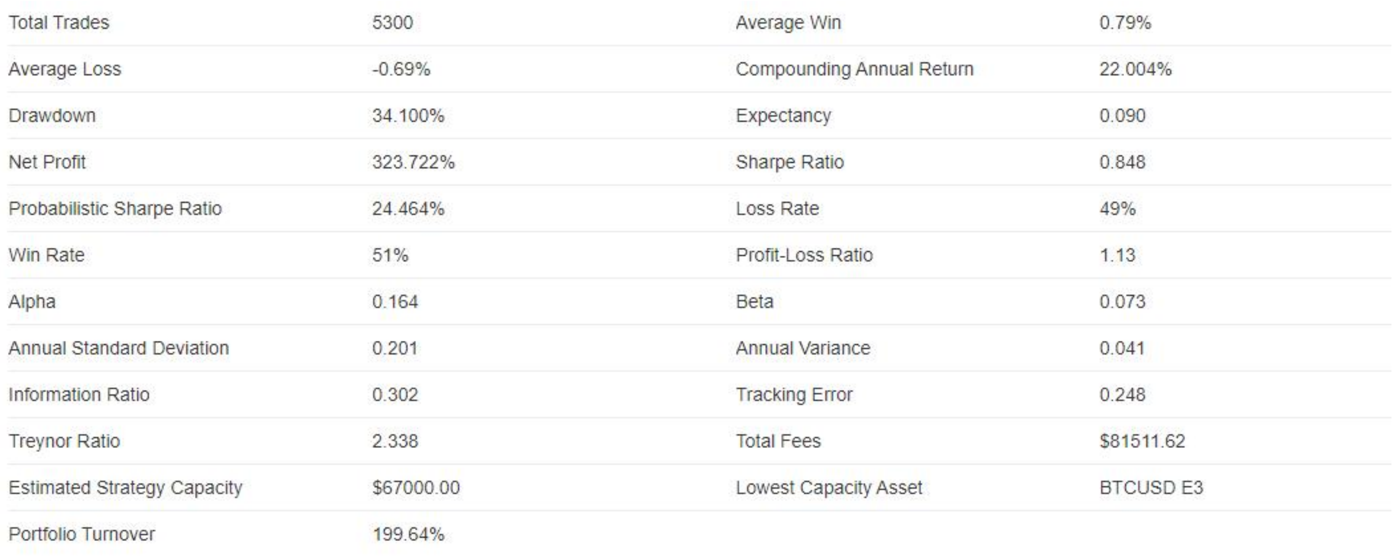


Fig 1. Overall Performance



Total Trades	5300	Average Win	0.79%
Average Loss	-0.69%	Compounding Annual Return	22.004%
Drawdown	34.100%	Expectancy	0.090
Net Profit	323.722%	Sharpe Ratio	0.848
Probabilistic Sharpe Ratio	24.464%	Loss Rate	49%
Win Rate	51%	Profit-Loss Ratio	1.13
Alpha	0.164	Beta	0.073
Annual Standard Deviation	0.201	Annual Variance	0.041
Information Ratio	0.302	Tracking Error	0.248
Treynor Ratio	2.338	Total Fees	$81511.62
Estimated Strategy Capacity	$67000.00	Lowest Capacity Asset	BTCUSD E3
Portfolio Turnover	199.64%		

Fig 2. Performance Metrics