

訊號與系統

期末報告

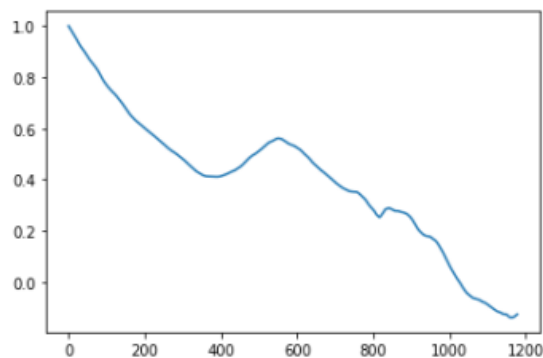
B0829024 葉季儒

一、課程內容

這堂課交會了我們許多訊號處理的技巧。包含各種波形、噪聲訊號，到訊號的處理，傅立葉轉換、捲積定理等實用的技術。並且了解到訊號除了用在多媒體處理外，對數據分析、股票市場分析都能有所運用。激起了我對訊號處理的熱情。

二、實作項目

在 Chapter 5 中教到自相關，並且用比特幣價格走勢做過練習。當初結果表示自相關係數相當低，且隨時間拉長，相關係數會逐步降低。



比特幣自相關圖 2013-2020

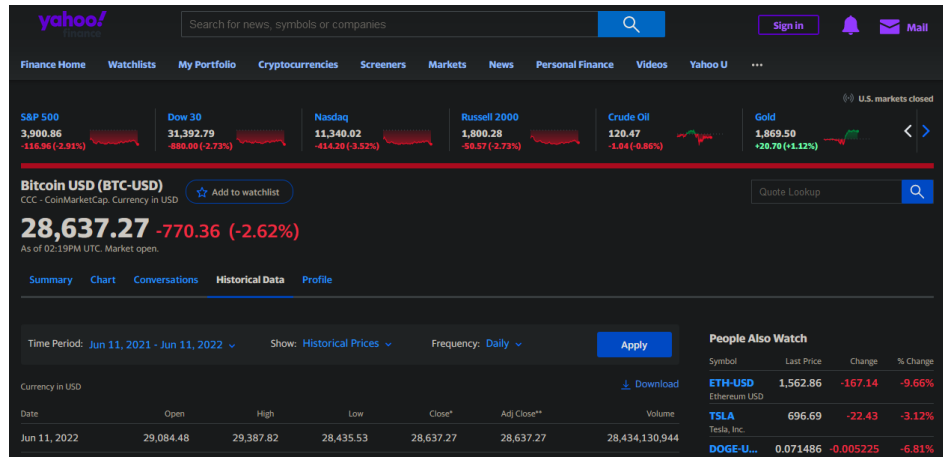
然而上述練習使用的幣價為 2013-2020 的資料，如今有了更多的資料可以驗證。再來上述的練習不能很好的反應自相關性對於價格預測的能力。因此有了這份實作項目。這份專案的目的是：

- (1) 用更多資料分析比特幣自相關圖
- (2) 使用自相關的數據預測未來趨勢
- (3) 使用自相關的結果，在一段時間內交易，統計最終的盈虧。
- (4) 探討其他影響要素

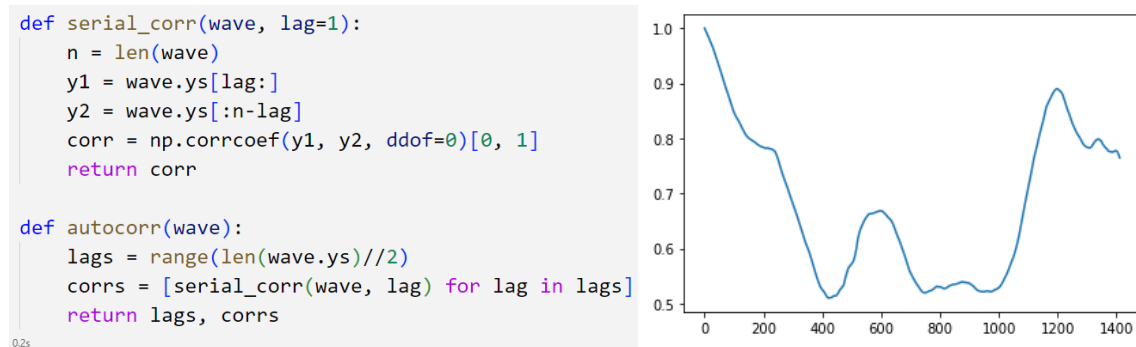


三、實作結果

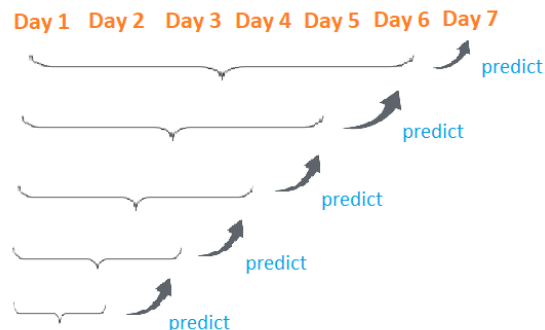
接著就開始製作。首先在 yahoo finance 找到 BTC-USD 的歷史價格。並且取得 2014-2022 的歷史價格文件(csv)。



然後，套用 Chapter5. 介紹的方法讀取與處理，計算當下的自相關圖。可以發現對於 2020/06/10 的數據，在 1199 天前有最好的相關係數=0.88。這個結果證明，對於比特幣價格可能可以找到歷史趨勢，為接下來的實驗確立基礎。



接下來我做了一些運算，從 1000~1400 的時間中，對每一個時間點切段，得到 0~切段點的多筆數據，分別對每筆數據計算自相關。如此就能找到與每個時刻最相關的歷史時刻。



此時，我碰到一個問題，該選擇什麼策略來判斷趨勢呢？

首先，我實作的是單日的價格判斷。也就是 $X+1 \text{ day's price} > X \text{ day's price}$ 則認為有漲的趨勢。計算的結果果然相當的差，我忽略了單日價格波動的可能，造成很多地方誤認為有上漲的趨勢，所以虧損嚴重。



於是我改成了多日的價格判斷。也就是 $\text{mean}(X+1 \text{ day}, X+2 \text{ day} \dots X+w \text{ day}) > X \text{ day}$ 則認為有漲的趨勢(這裡的 w 可以視為短期趨勢或長期趨勢， w 越大，會參考到長遠的數據，判斷的目標會較為長遠)。這樣判斷確實相對準確，虧損略為減少，但仍然沒有達到正向損益。



最後我想可能是因為相關係數不夠高，所以會參考到代表性不足的歷史趨勢。所以我加入了一個閾值來判斷。當相關係數 $>$ 閾值才列入考量(閾值越大，相關性也越大，但資料量會顯著減少)。如此一來，在高相關度的情況，終於實現了正收益，驗證了這個策略的可能性。



```
def predict_trend(wave, current_time, window_size): # 計算趨勢
    new_wave = wave.copy()
    new_wave.ys = new_wave.ys[:current_time] # 剪取到當前時間點
    lags, corrs = autocorr(new_wave) # 計算自相關曲線
    max_corr = max(corrs[100:]) # lag->0, corr always->1, so remove it
    max_corr_index = corrs.index(max_corr) # 計算相關度最高的時間點
    # print('max_corr_index',max_corr_index,'; max_corr',max_corr)

    next_average = np.mean(np.array(wave.ys[max_corr_index+1:max_corr_index+1+window_size]))
    # 計算 x day ~ x+window_size day的平均值，回傳平均-x day的值作為趨勢
    return (max_corr,next_average-wave.ys[max_corr_index])
```

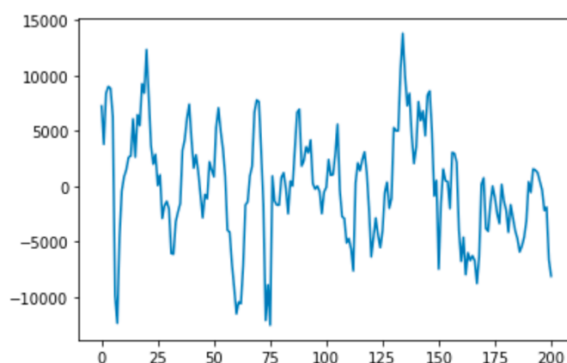
Python

```
def cal_profit(wave, start, end, amount, window_size, trend_threshold, corr_threshold):
    profit = []
    for day in range(start,end):
        (corr,trend) = predict_trend(wave, day, window_size) # 計算趨勢
        if trend>trend_threshold and corr>corr_threshold:
            # 如果趨勢向上(>trend_threshold)，且相關度較大(>corr_threshold)，則開始交易
            profit.append((wave.ys[day+window_size] - wave.ys[day]) * amount)
            # 交易結果為 (x+window_size day的價格 - x day的價格) * 數量
    return profit
```

Python

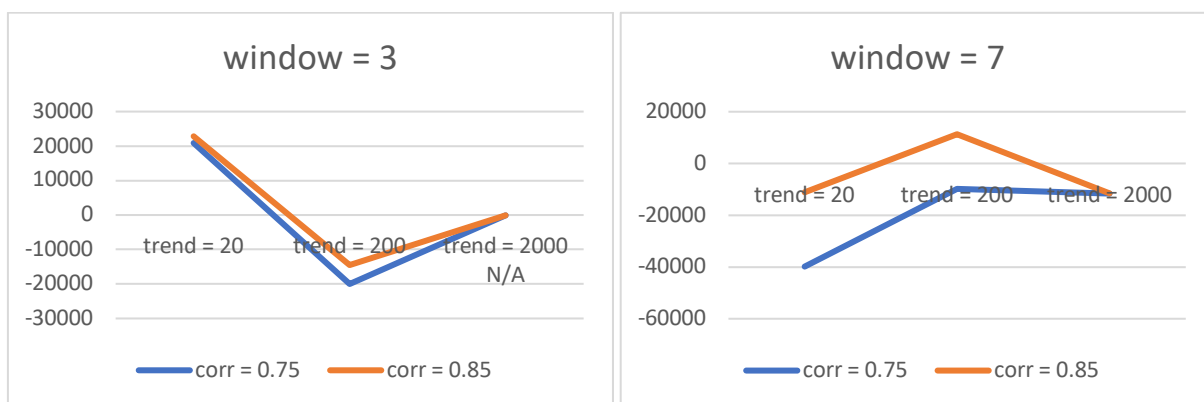
四、結論

這是在 $w=7$ (預測目的為一週)， $\text{trend_threshold} = 200$ (歷史上下週會漲 >100 鎊)， $\text{corr_threshold} = 0.85$ (相關性要求較嚴格)，每日最多交易 1 美元的狀態執行的交易結果。最終損益為+11295 美元。



可以看出縱使已經參考了高度相關的歷史資料，仍然會出現某日大幅虧損的狀況，說明只依靠歷史價格做交易不是很好的策略。然而總體而言，賺錢 $>$ 賠錢，仍有達到預期的目的。

雖然上述的交易為賺錢，但其實參數的調配相當的重要。在進行多次實驗後，得到了如下的關係表：



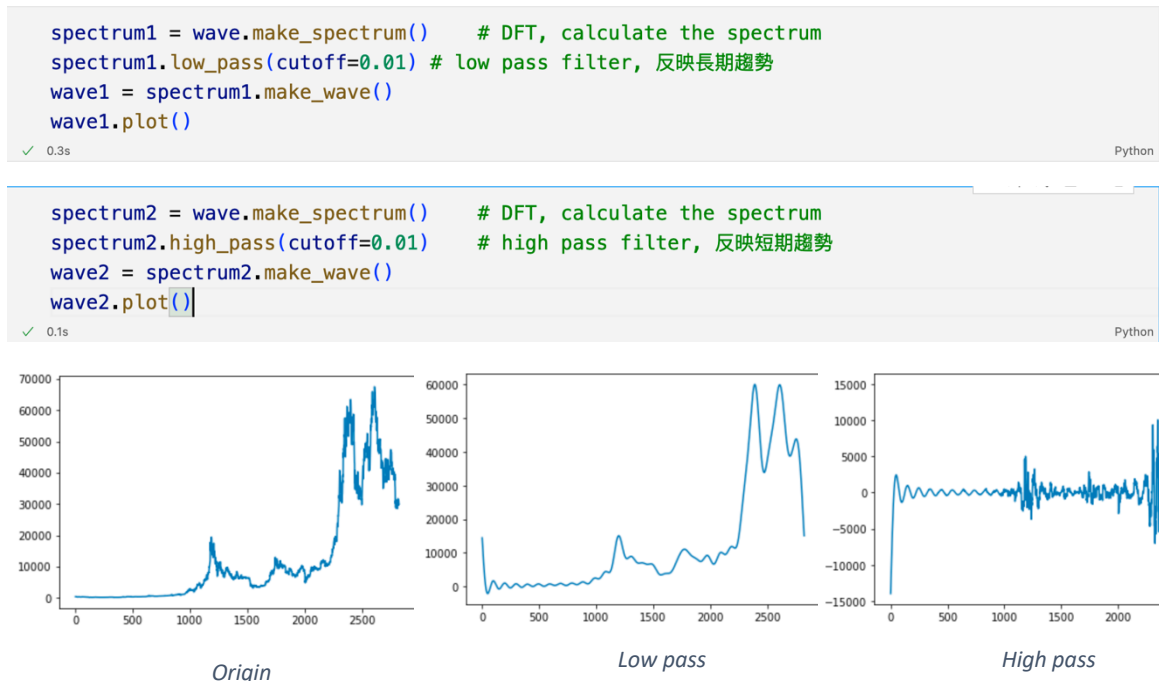
從這張表可以觀察到許多細節。首先自相關閾值扮演一個重要的角色，在資料量足夠的情況相關度越大，取得的趨勢越是吻合，可以更好的預測未來價格。其實這點相當合理，只是閾值仍然不可以太大，否則會沒有資料量。

再來可以看到預測長短和趨勢閾值的組合會顯著影響結果。當預測短時($\text{window}=3$)，波動會較小，此時趨勢閾值需要縮小，才能吻合實際的波動。當預測長時($\text{window}=10$)，波動會較大，此時趨勢閾值需要拉大，才能濾掉短期波動，更好的貼合長期趨勢。

最終我們在 $\text{window}=3, \text{trend_threshold}=20$ 和 $\text{window}=7, \text{trend_threshold}=200$ 時，實現了正向收益。

通過上述結論，可以發現單靠自相關性做幣價分析有些不足，如果整合更多資訊，應該可以有更好的預測。

在訊號領域，我們可以通過對頻譜圖套用低通濾波器與高通濾波器，分別反應短期價格變化和長期價格變化：



有許多方法可以應用這些資料，包含再拿濾波後的數據計算自相關，或是直接當成策略等。礙於篇幅，這邊就不繼續延伸了。

經過這些計算，我們對訊號處理在虛擬貨幣價格分析的應用有了初步的了解，初步應證了可行性，也為進階的處理確立了基礎。

五、加分項目

在 github 項目中，我附上了這份專案的程式碼和一些分析結果。此外，我還附上訊號與系統課程中的 HomeWork 練習與電子筆記（簡單的章節沒有筆記）。

<https://github.com/fdsf53451001/signal-crypto-analyze>

