Intro to FinTech – Final Project report

資工二 B07902046 高長聖

1. Introduction:

首先，我將詳述在本次作業，我使用的策略及流程:先建立策略模型，然後分析2011/1/3 ~ 2019/12/16之間的台股開盤價，藉此定義模型中參數範圍。最後，利用GA(遺傳演算法)來尋找最佳參數。

而在文末，我將探討為何我使用的策略無法達到更佳的獲利，並找出未來能夠繼續讓模型更加優化的方法。

1. Implementation:
2. 模型設計與資料分析:

我使用的模型，是混和傳統數據，包括MA、RSI、KD值，以及較直覺的股票漲跌策略，包括已經連續漲、跌幾天。並在買與賣分別設定界線，若全部符合，即執行買賣，反之則不動作。

MA: 設定短期天數、長期天數，及短期MA-長期MA所要超過界線。經由分析，可知隔日開盤價的漲跌幅度大致落在[-300, 300]之間。因此，短期天數的範圍為[1, 20]，長期為[21, 50]，差異程度範圍[-300, 300]。

RSI: 設定短期天數、長期天數，即短期RSI-長期RSI所要超過的界線。而RSI的範圍為[0, 1]，因此，設定短期天數範圍[1, 20], 長期天數範圍[1, 50]，差異程度範圍[0, 1]。

KD: 設定K值小於D值時，D值的上界；以及K值大於D值時，D值的下界。由於K和D值都位在[0, 100]，因此上界和下界範圍皆為[0, 100]。

連續漲跌天數: 設定連續漲跌天數的界線，以及目前是漲或是跌的的狀態。經由分析，所有連續漲、連續跌的天數，都是在[0, 8]。因此，設定連續漲跌天數界線範圍[0, 8]。

1. GA(遺傳演算法)設計:

定義染色體: 模型本身，而基因即為模型參數，共14個參數。

Fitness function: 採用多點之間的比較。由於有可能有時間長短所造成的資料差異，因此不採用單純獲利率，而是將每一點的最佳可能獲利算出，並檢查當前模型在那天的獲利，算出兩者之間的比例。

Encoding: 利用Python中的list，將所有參數，以數值的方式，按照固定的順序存於其中。

初始族群: 完全由隨機產生。族群大小依照所設定的值進行調整。

Selection: 依照fitness function的大小，挑選出所需族群規模的模型個體。並採用菁英主義(elitism)，將母代的模型也加入挑選名單中。

Crossover: 全體互相交配，也就是交配率100%。子代從父代和母代那邊，隨機繼承各個模型參數。這樣的邏輯，是假設個體會有所突出的原因，在於某幾個圖的的基因(參數)互相搭配。因此採用隨機繼承的方式，尋找其他可能的參數搭配方式。

Mutation: 每個參數的突變機率皆為1/10。突變的方式，即是從新再從參數的定義域中，隨機選取一個值。

終止條件: 依照一開始設定的回合數，迭代此回合後即可終止。

以下，為最終寫出來的程式:

daily\_data: 2011/1/3~2019/12/16的股市ohlcv\_daily資料。

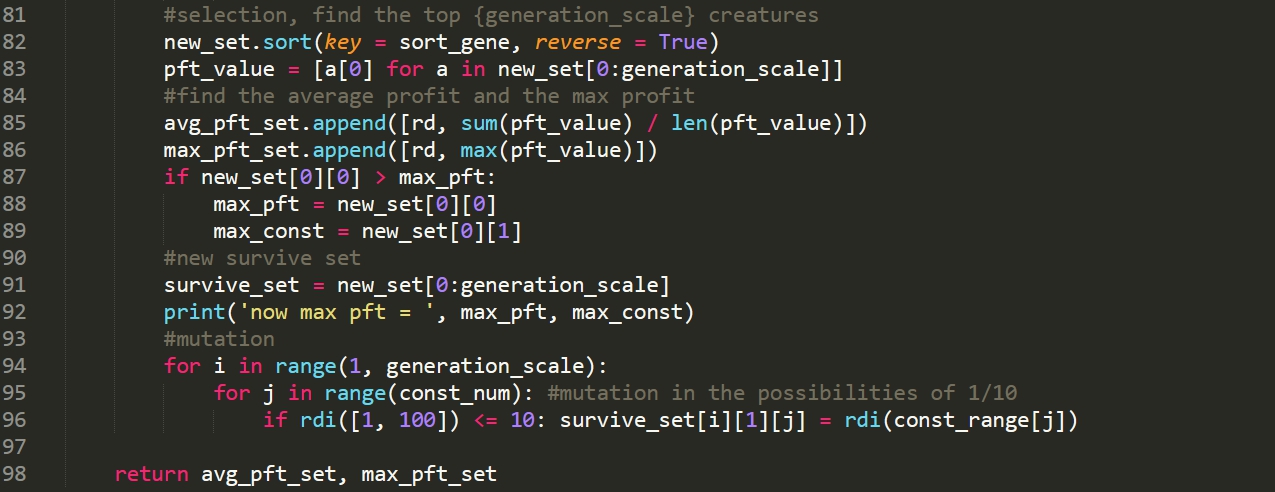
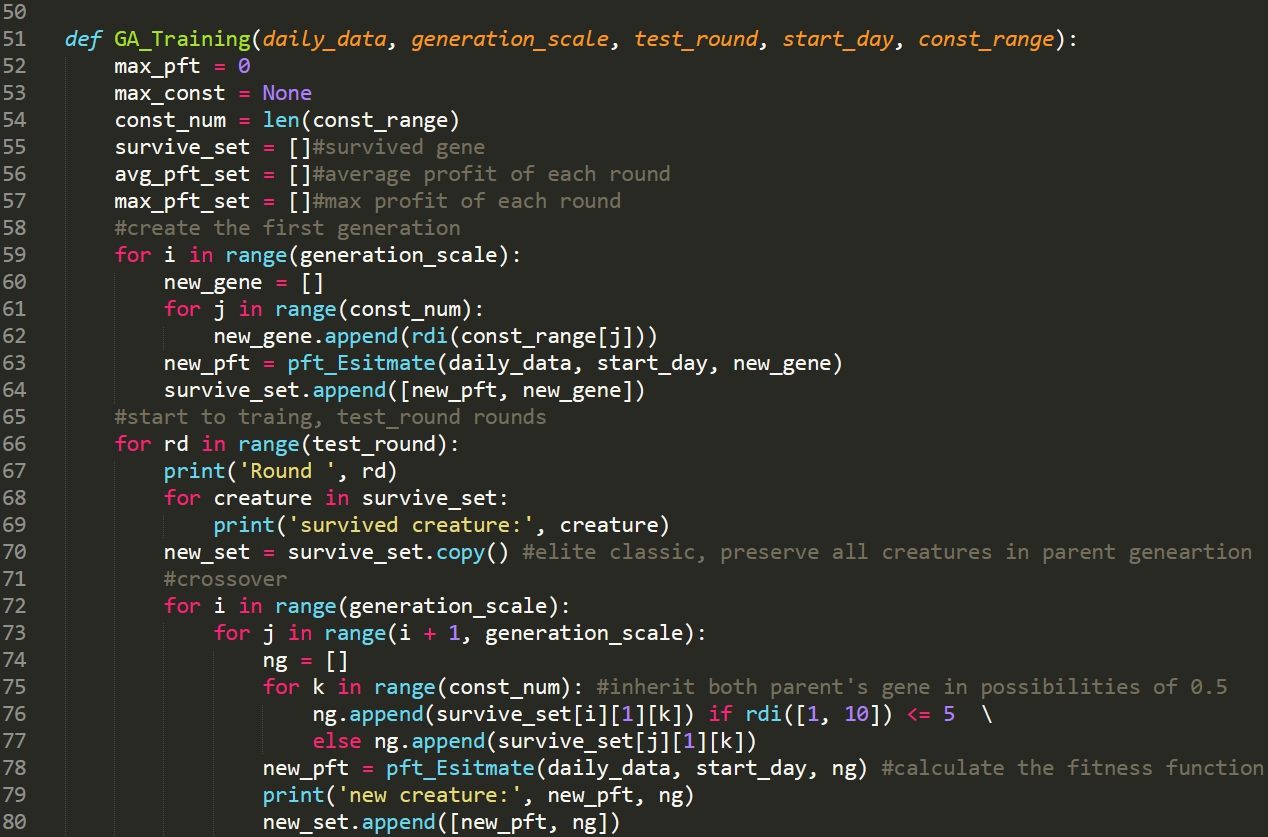
generation\_scale: 設定族群大小。

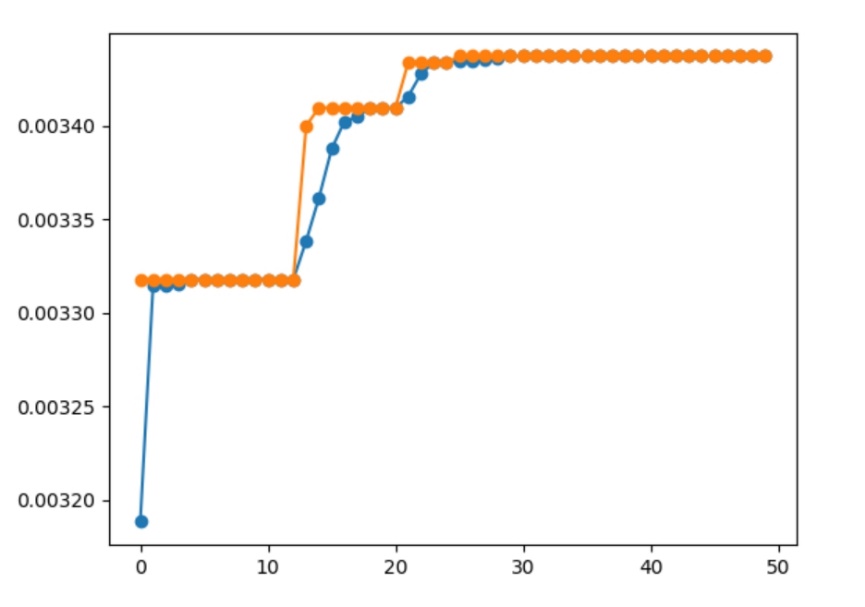
test\_round: 設定迭代次數。

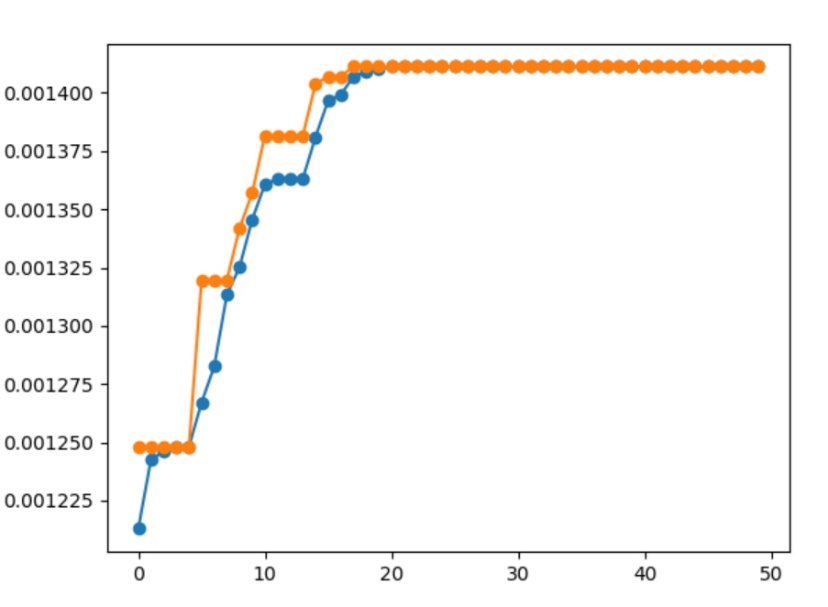
start\_day: 訓練資料的起始位置。藉此控制訓練資料大小。

const\_range: 各個參數的定義域。

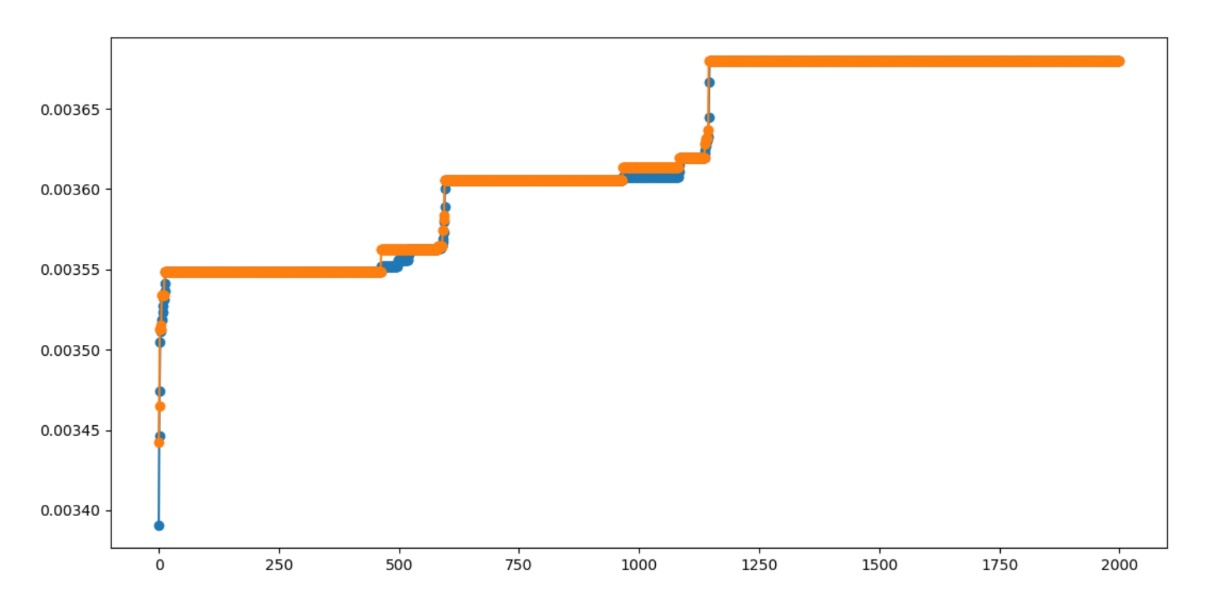
回傳值: 在每一回合的平均fitness以及最大fitness



1. 結果:

橫軸:迭代次數；縱軸:fitness；橘線:最大fitness；藍線:平均fitness

(a) Data size: 14, Test round: 50 (b) Data size: 140, Test round: 50



(c) Data size: 14, Test round: 2000

1. 分析:

首先，上面三張圖都顯示，當我的GA迭代夠多次之後，最大fitness會達到某個值而穩定下來。這表示，我所設計的遺傳演算法，具有尋找區域最佳解的能力。

再來，上面三張圖也都顯示，當最大fitness達到平穩狀態後，平均fitness也會跟進，最後趨近於最大fitness。這表示，在菁英主義(elitism)的搭配下，族群可以迅速往好的方向演進。然而，這也增加演算法跳脫出區域最佳解的難度。

最後，比較圖(a)和圖(c)，同樣的Testing data size都為14，而一個迭代50次，另一個迭代2000次。可以發現，當不理會平穩狀態而繼續迭代下去的話，每隔一段時間，都有可能讓最大值fitness繼續進步。推測，這是因為mutation的關係，創造出一個或者是多個特別強大的個體，使得整體族群跳脫出區域最佳解之中，往更好的地方邁進。這也顯示出，GA中的mutation，實際上有利於族群的發展。

1. 結論:

最終，我使用了由Testing data size為14，Test round為50的測試資料，選擇了:

短期MA: 4，長期MA: 48，買進長期MA-短期MA: 203，賣出短期MA-長期MA: 213

短期RSI: 18，長期RSI: 46，買進長期RSI-短期RSI: 0.81，賣出短期RSI-長期RSI:0.77

買進D值上界: 71，賣出D值下界: 14

買進時的趨勢: 較前一天價格低，連續漲跌界線: 跌8天

賣出時的趨勢: 較前一天價格高，連續漲跌界線: 漲6天

作為最終買賣策略模型。

1. 缺失:

首先，在資料分析方面，我僅以每日的開盤價格進行分析，錯失許多微小細節，模型的設計方面也較不全面。再來，由於我GA採用elitism的關係，會加深受限於區域最佳解的影響，導致跨區域的演化速度較慢。最後，我所使用的crossover方式、mutation rate、generation scale，以及族群初始化的方式，都僅僅是證明能找到區域最佳，無法說明擁有更好，找到全域最佳解的能力。

1. Improvement:
2. 從模型下手:

可以增加模型的強度，例如，將分析資料從僅僅是每日的開盤價，到對OHLC都進行分析，或者是納入每分鐘的交易趨勢。

1. 從GA下手:

GA最大的缺點，就是容易墮入區域最佳解之中，而不易找到全域最佳解。因此，或許在crossover的方式，或者mutation rate，都可以進行調整。也可以搭配災變(catastrophe)的運用，當最大fitness和平均fitness趨於平穩狀態時便摧毀掉當前優秀染色體，以加速區域最佳位置的轉變。

1. 配合其他演算法:

配合蟻群演算法、模擬退火法、梯度遞增法等演算法的運用，彼此互相彌補對方的缺點，也可以找到更好的參數解。