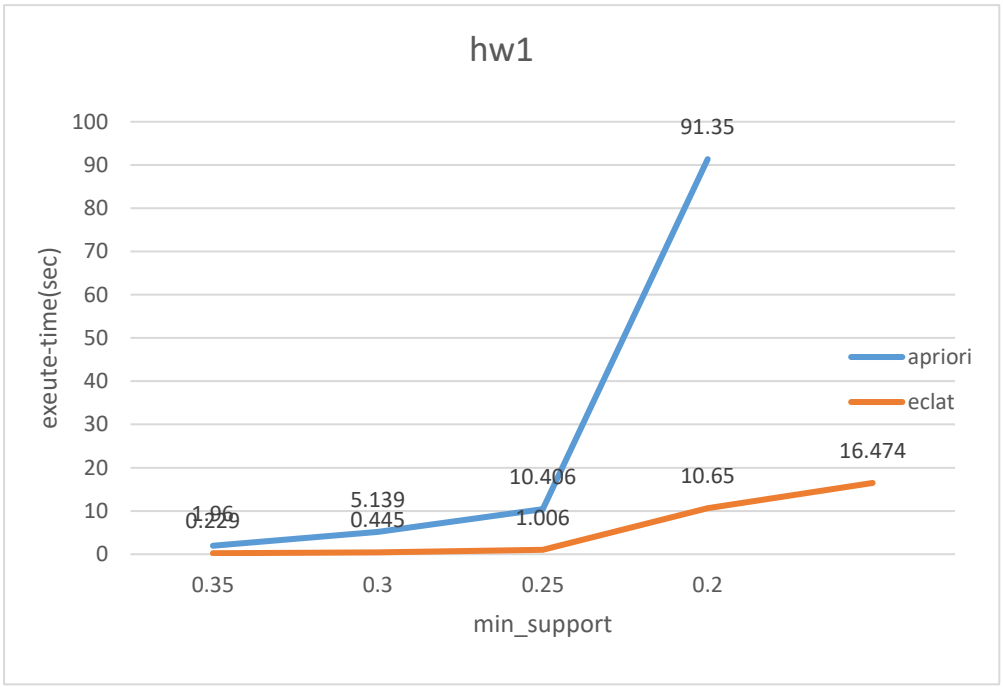
資訊勘測 HW1 R06521705 陳思愷

1. 一開始是嘗試使用 list 去實作 apriori，最後出來的結果須花費太長的時間 (min_sup = 0.2 , time >4000 sec)， 後來開始思考利用其他資料結構以及 apriori 演算法中各細部流程的優化:
   a. 資料量變大時，在 list 上進行查找和更動的動作需花費較 dict 結構比起來相對多很多的時間，故後來改使用 dict 結構。
   b. 在 apriori 算法尋找 candidate 的過程中，需要去 database 中尋找是否有符合的人選 ex: 1234 ,1235 ➔ 若 L 中 2345 , 1345 , 1245 則可證明 12345 是可被考慮的 candidate，此一步驟可利用 dict hash 結構的特性進行快速的對比和查找，有助於整體運算時間的降低。
   c. Python 中有許多已經寫好的 function，ex: issubset(), intersection()...，可以善用這些已經存在的 function 而非自己重些寫一個，會有較好的運行效率和簡潔呈現

2.



關於 eclat 最後 support = 0.01 and 0.05 的部分，由於記憶體有限以及自己的程式碼寫法無法有效節省記憶體，無法達成要求。

會再詢問及探討是否有更好的做法。