

[首頁](#)[個人主頁](#)[活動資訊](#)[排名](#)[公告](#)[常見問題](#)[我要提問](#)[張](#)[登出](#)[100 道題目](#) > [D50：集成方法：堆疊泛化\(Stacking\)](#)

D50：集成方法：堆疊泛化(Stacking)

[PDF 下載](#)[全螢幕](#)

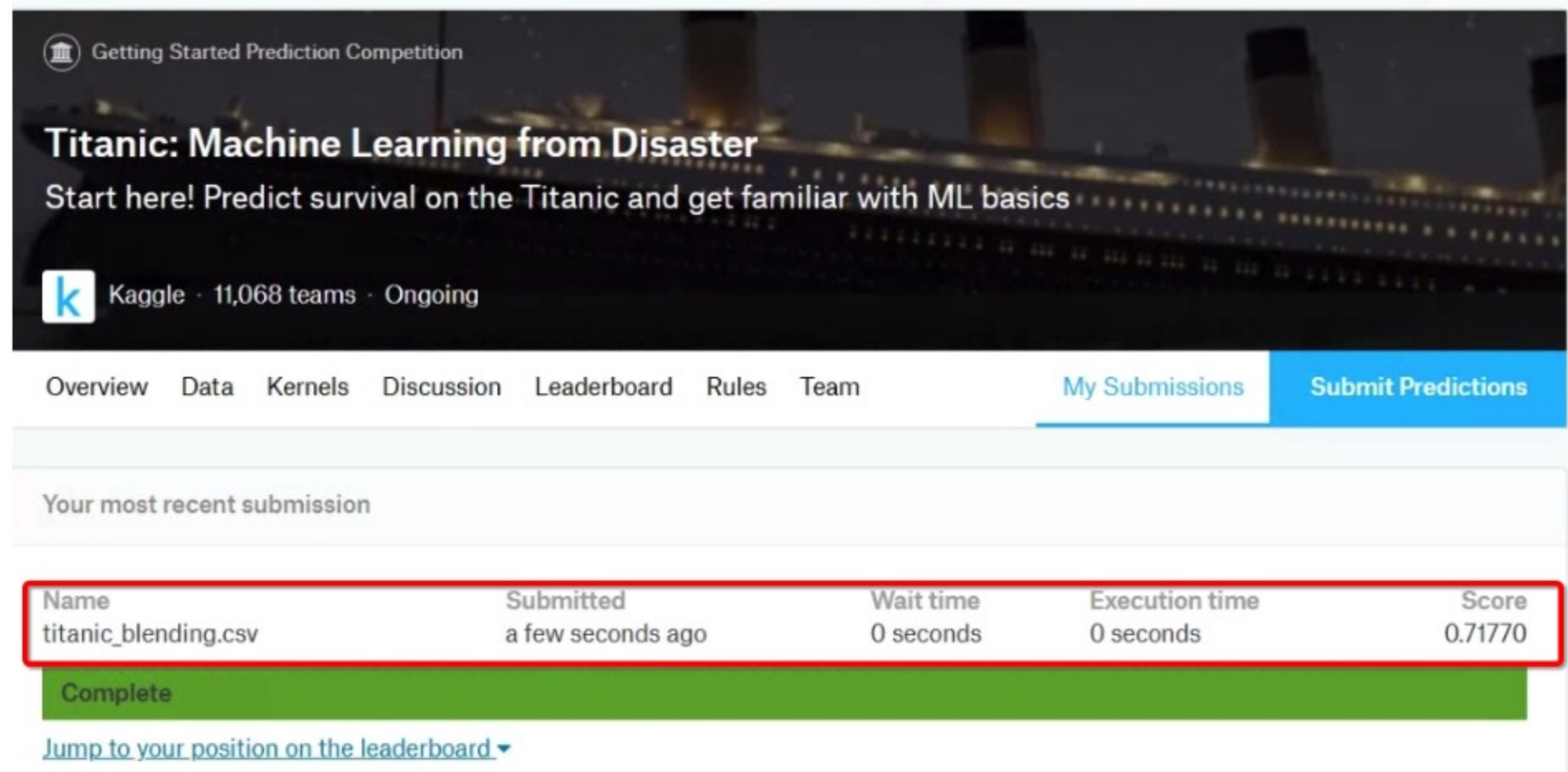
Sample Code & 作業內容

分類預測的集成泛化，也與回歸的很不一樣

既然分類的 Blending 要變成機率，才比較容易集成，那麼分類的 Stacking 要讓第一層的模型輸出機率當特徵，應該要怎麼寫呢？

請參考範例程式碼Day_050_Stacking，完成Day_050_Stacking_HW(作業檔)之後輸出成csv檔上傳到kaggle，並將結果提交到 Kaggle 網站看看結果。(Kaggle鐵達尼競賽連結作業檔內提供)

作業提交請截圖kaggle競賽頁面提交畫面上傳至github，並回到官網提交github連結。(以下為Kaggle競賽頁面截圖範例)

[檢視範例](#)

參考資料

StackingCVClassifier

mixtrend 官方網站 [網頁連結](#)

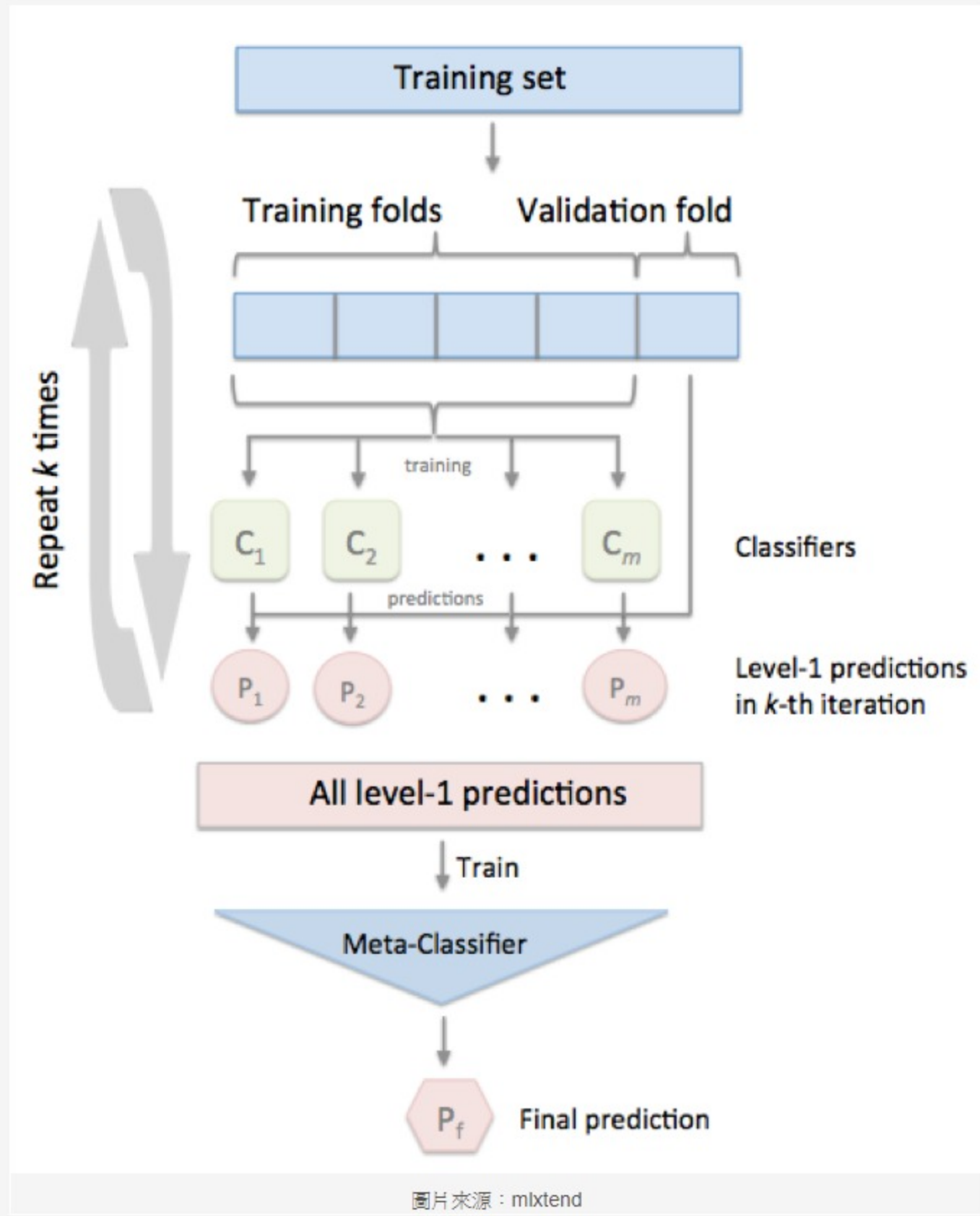
這個連結在課程中有給過，但是這網頁說明的不只有參數調整的寫法，還包含 Stacking 的 Classifier 寫作細節，在看我們的範例之餘，如果還有其他疑問，建議也可以直接來官方網站挖挖寶

其中 Stacking 內部參數的寫法，我們在這邊說明一下，例如下列參數：

`randomforestclassifier__n_estimators': [10, 50]`

對應到的是 Stacking 裡面 RandomForestClassifier 單模的 n_estimators 參數，可以看出不只有中間連結的底線要兩個，單模模型的呼叫，也需要把原本模型的大寫全部換成小寫，同學如果還有疑問，不妨可以自己試跑看看

不過時間 Stacking 執行會稍微有點久喔，建議先估計一下執行時間。



圖片來源：mixtrend

如何在 Kaggle 首戰中進入前 10%

Wille 個人心得 [網頁連結](#)

雖然是較早期的文章了，照著做大概也已經很難進入前 10% 了，但是機器學習該講解的部分都有略略提到，在此獻給即將中考的同學準備複習之用。

而其中有詳細講解了 Stacking 的用法與他自己的寫法，同學可以參考程式本身，理解一下我們 Stacking 講解的未盡之處，雖然現在工具是方便多了，但是有一些想不通的細節，有程式碼可以推敲，我想會方便很多。

```
1 class Ensemble(object):
2     def __init__(self, n_folds, stacker, base_models):
3         self.n_folds = n_folds
4         self.stacker = stacker
5         self.base_models = base_models
6
7     def fit_predict(self, X, y, T):
8         X = np.array(X)
9         y = np.array(y)
10        T = np.array(T)
11
12        folds = list(KFold(len(y), n_folds=self.n_folds, shuffle=True, random_state=2016))
13
14        S_train = np.zeros((X.shape[0], len(self.base_models)))
15        S_test = np.zeros((T.shape[0], len(self.base_models)))
16
17        for i, clf in enumerate(self.base_models):
18            S_test_i = np.zeros((T.shape[0], len(folds)))
19
20            for j, (train_idx, test_idx) in enumerate(folds):
21                X_train = X[train_idx]
22                y_train = y[train_idx]
23                X_holdout = X[test_idx]
24                # y_holdout = y[test_idx]
25                clf.fit(X_train, y_train)
26                y_pred = clf.predict(X_holdout)[: ]
27                S_train[test_idx, i] = y_pred
28                S_test_i[:, j] = clf.predict(T)[: ]
29
30            S_test[:, i] = S_test_i.mean(1)
31
32        self.stacker.fit(S_train, y)
33        y_pred = self.stacker.predict(S_test)[: ]
34        return y_pred
```

圖片來源：Wille

提交作業

請將你的作業上傳至 Github，並貼上該網網址，完成作業提交

[確定提交](#)[如何提交](#)

熱門問答

孫博慶 · 2019 06 09

投影片第八頁圖例問題

在我去搜尋了其他關於Stacking的介紹後，發現對每一個Model都會做一k-fold，這樣才會讓Training Set的大小保持相同到遠下一層。但第八頁的圖很容易誤導成每個不同的Model只會用到其中一個Training Se...

到 Cupoy 問答社區提問，讓教練群回答你的疑難雜症

[向專家提問](#)[如何提問](#)