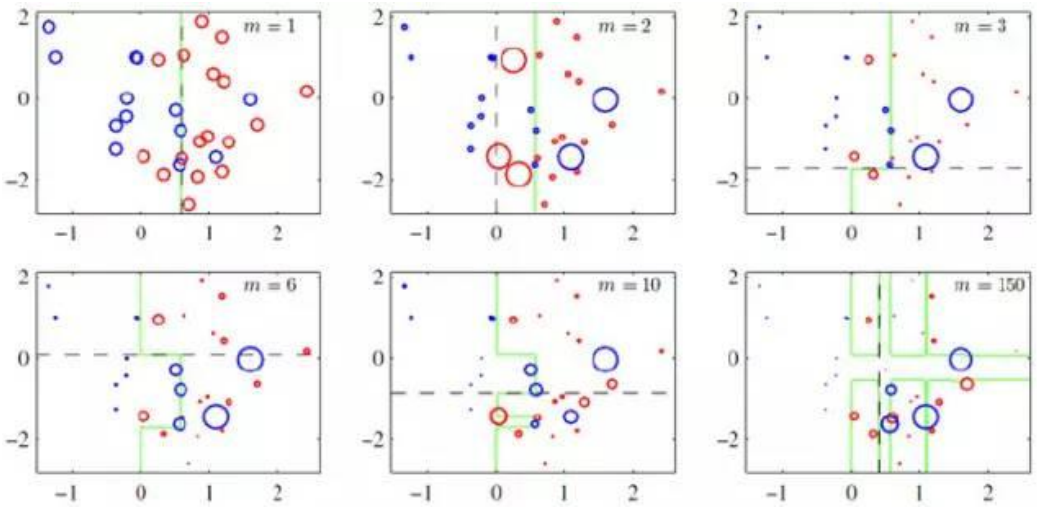
# XGBoost

- 什麼是 XGBoost ?

1. 是由 Tianqi Chen http://homes.cs.washington.edu/~tqchen/ 最初開發的實現可擴展，便攜，分佈式 gradient boosting (GBDT, GBRT or GBM) 算法的一個庫，可以下載安裝並應用於 C++，Python，R，Julia，Java，Scala，Hadoop，現在有很多協作者共同開發維護。
2. XGBoost 所應用的算法就是 gradient boosting decision tree，既可以用於分類也可以用於迴歸問題中。
3. Boosting
    a. 基本思想：不同的訓練集是通過調整每個樣本對應的權重實現的，不同的權重對應不同的樣本分布，而這個權重為分類器不斷增加對錯分樣本的重視程度。
    b. 步驟：
        i. 1. 首先賦予每個訓練樣本相同的初始化權重，在此訓練樣本分布下訓練出一個弱分類器；
        ii. 2. 利用該弱分類器更新每個樣本的權重，分類錯誤的樣本認為是分類困難樣本，權重增加，反之權重降低，得到一個新的樣本分布；
        iii. 3. 在新的樣本分布下，在訓練一個新的弱分類器，並且更新樣本權重，重複以上過程 T 次，得到 T 個弱分類器。
    c. 通過改變樣本分布，使得分類器聚集在那些很難分的樣本上，對那些容易錯分的數據加強學習，增加錯分數據的權重。這樣錯分的數據再下一輪的疊代就有更大的作用（對錯分數據進行懲罰）。對於這些權重，一方面可以使用它們作為抽樣分布，進行對數據的抽樣；另一方面，可以使用權值學習有利於高權重樣本的分類器，把一個弱分類器提升為一個強分類器。
    決策樹。
    d. 對於 Boosting 來說，有兩個問題需要回答：一是在每一輪如何如何改變訓練數據的機率分布；二是如何將多個弱分類器組合成一個強分類器。




e. 上圖（圖片來自 prml p660）就是一個 Boosting 的過程，綠色的線表示目前取得的模型（模型是由前 m 次得到的模型合併得到的），虛線表示當前這次模型。每次分類的時候，會更關注分錯的資料，上圖中，紅色和藍色的點就是資料，點越大表示權重越高，看看右下角的圖片，當 m=150 的時候，獲取的模型已經幾乎能夠將紅色和藍色的點區分開了。

f. 參考網址：https://www.readhouse.net/articles/132582732/

4. Gradient Boosting
   a. Gradient boosting 是 boosting 的其中一種方法
   b. Gradient boosting 就是通過加入新的弱學習器，來努力糾正前面所有弱學習器的殘差，最終這樣多個學習器相加在一起用來進行最終預測，準確率就會比單獨的一個要高。之所以稱為 Gradient，是因為在添加新模型時使用了梯度下降算法來最小化的損失。
5. 機器學習算法中 GBDT 和 XGBOOST 的區別有哪些
   a. 基分類器的選擇：傳統 GBDT 以 CART 作為基分類器，XGBoost 還支持線性分類器，這個時候 XGBoost 相當於帶 L1 和 L2 正則化項的邏輯斯蒂回歸（分類問題）或者線性回歸（回歸問題）。
   b. 二階泰勒展開：傳統 GBDT 在優化時只用到一階導數信息，XGBoost 則對代價函數進行了二階泰勒展開，同時用到了一階和二階導數。順便提一下，XGBoost 工具支持自定義損失函數，只要函數可一階和二階求導。
   c. 方差-方差權衡：XGBoost 在目標函數里加入了正則項，用於控制模型的複雜度。正則項里包含了樹的葉子節點個數 TT、每個葉子節點

上輸出分數的 L2 模的平方和。從 Bias-variance tradeoff 角度來講，正則項降低了模型的 variance，使學習出來的模型更加簡單，防止過擬合，這也是 XGBoost 優於傳統 GBDT 的一個特性。

d. 其他特點參考 http://blog.csdn.net/chengfulukou/article/details/76906710