Vici 報告

胡祖望

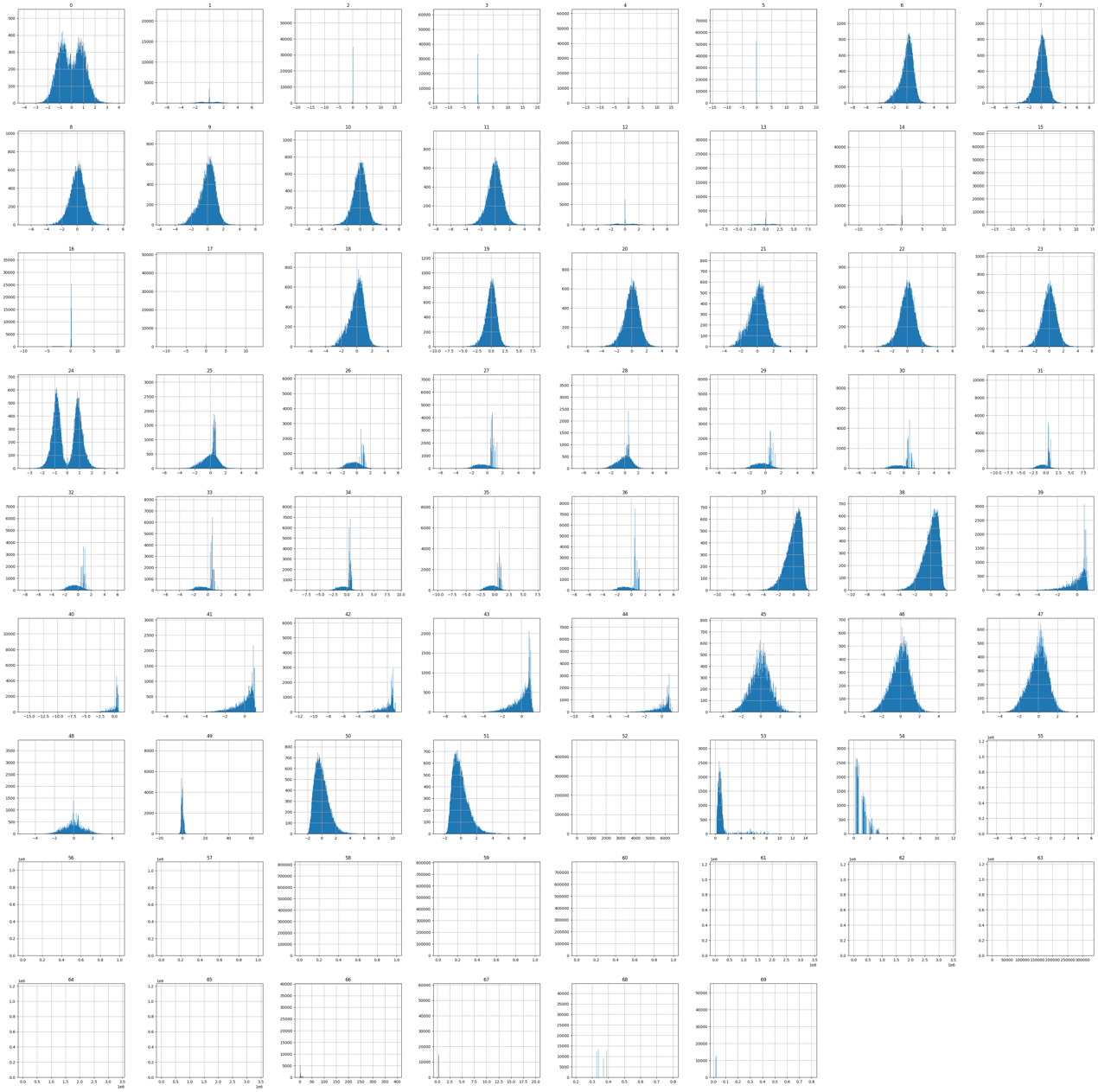
數據統計：

資料量：

training data : 1174461筆

eval data：1175302筆

由於數據量非常大，因此沒有把所有數據都畫上去



training data的feature描述：

feature 0 ~ 69：平均值為0、標準差為1

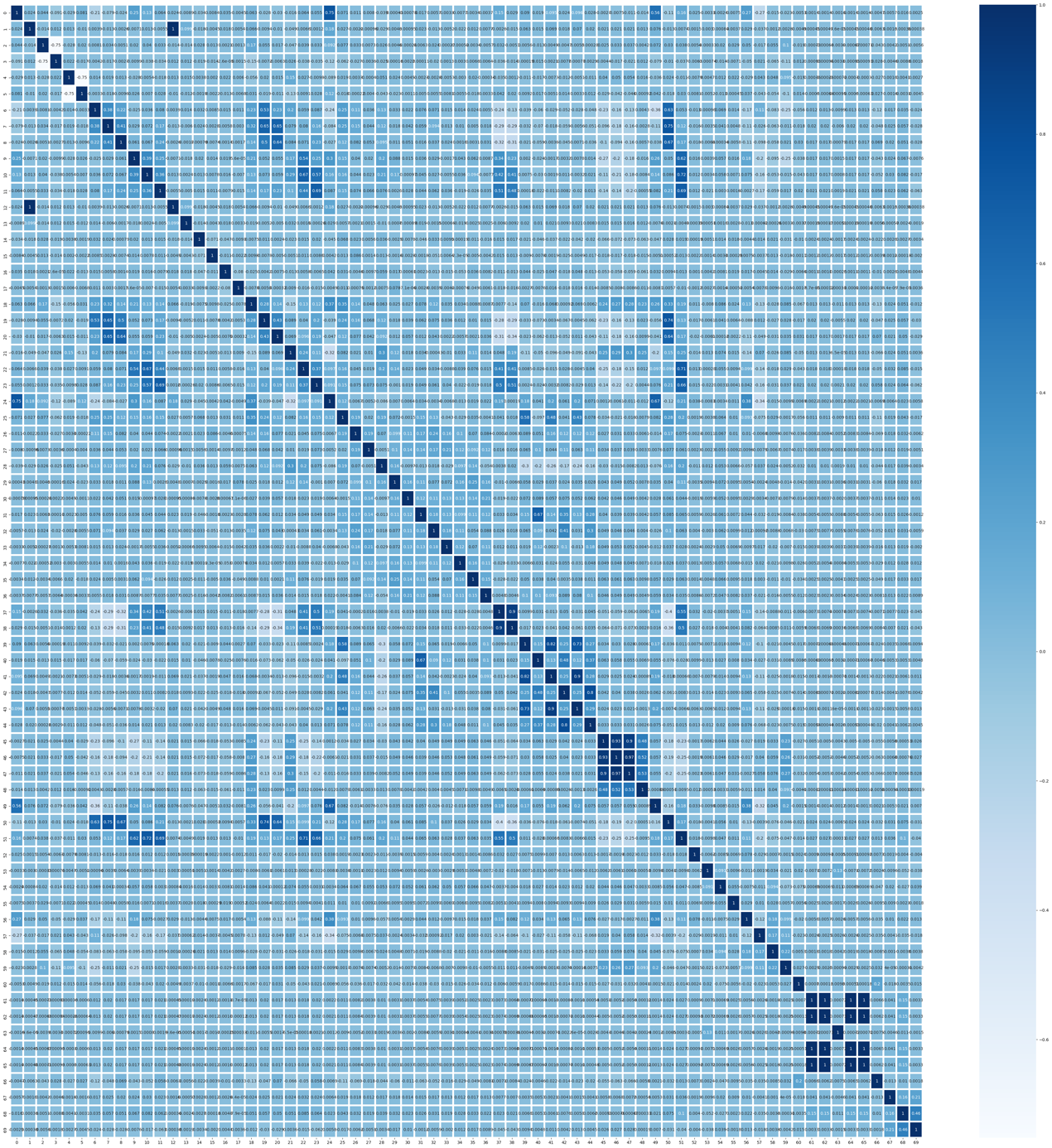
feature 56 ~ 60：數值為0 or 1的binary data

feature 61 ~ 67：大部份數據集中在0，

觀察發現與想法：

1. 感覺資料的建立是從特定的分配抽樣抽出，像是feature 0 是由兩個鐘型分配組成的，可以看出有些feature之間具有高度相關，發現feature 1 跟 feature 12完全相同，feature 41 42 43，feature 61, 62, 64, 65之間有趨近於1的相關性

之後在model 的訓練上會對高度相關的feature進行處理



模型：

我有做兩個模型來做預測，第一個是用傳統的NN model 去做預測，第二個是用Lightgbm 去做預測，會選擇Lightgbm是因爲之前專題使用過，他的優點有更快的訓練速度和更高的效率、低記憶體使用率、更好的準確度跟能夠處理大規模數據。

在數據上我會先隨機打散，因為有些數據是二元的，在切割成training 跟 validation 時怕都切到同一種資料。

Model 1: NN



可以看到傳統的NN model在訓練上表現並沒有表現的很好，因此打算改用

Lightgbm去做預測。

Model 2: Lightgbm

params = {

'learning\_rate': 0.1,

'lambda\_l1': 0.1,

'lambda\_l2': 0.2,

'max\_depth': 50,

'objective': 'multiclass',

'num\_class': 3,

'num\_leaves':50,

"boosting":'dart' or ‘gbdt’

}

Boosting 我有使用 dart（Dropouts meet Multiple Additive Regression Trees）跟 gbdt（ Gradient Boosting Decision Tree）兩種，gbdt的表現比較好。

1. Full data：
   1. 100萬筆資料當training剩下的資料當 validation



* 1. 10萬筆資料當training data 剩下的資料當 validation



在前面的數據分析當中我們能看到有很多資料中的 feature是高度相關甚至是一模一樣的 ，因此我將這些高度相關的feature移除掉再去做training，結果如下。

1. Cleaned data：

將資料相同的feature移除，也將高度相關的feature移除。

* 1. 100萬筆資料當training剩下的資料當 validation



* 1. 10萬筆資料當training data 剩下的資料當 validation



我們可以看到其實10萬筆資料train 出來的結果其實與100萬筆的差不多，我還有做5000筆與1萬筆資料當training的模型，他們的accuracy也有50幾％將近6成。

最終可以看到Lightgbm訓練的結果比傳統的NN還好很多，在訓練時間上也快很多，