**Data selection:**

[https://archive.ics.uci.edu/ml/datasets/Electrical+Grid+Stability+Simulated+Data+#](https://archive.ics.uci.edu/ml/datasets/Electrical+Grid+Stability+Simulated+Data+)

Data columns (total 14 columns):

tau1 10000 non-null float64

tau2 10000 non-null float64

tau3 10000 non-null float64

tau4 10000 non-null float64

p1 10000 non-null float64

p2 10000 non-null float64

p3 10000 non-null float64

p4 10000 non-null float64

g1 10000 non-null float64

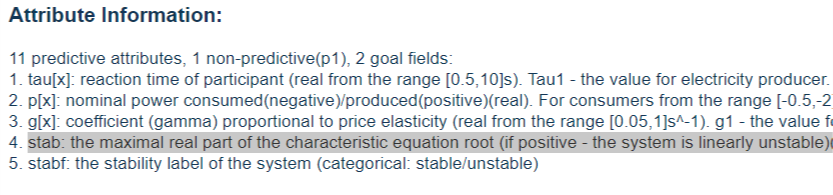
g2 10000 non-null float64

g3 10000 non-null float64

g4 10000 non-null float64

stab 10000 non-null float64

stabf 10000 non-null object



**Problem define 1:**

預測系統是否穩定(即stabf欄位) - classification

**Observation:**

根據資料說明stab是穩定度的數值(幾乎和stabf相關)故不考慮做特徵，剩下的又以tau1, tau2, tau3, tau4, g1, g2, g3, g4 欄位和 stab,stabf 相關度較高，故考慮作為特徵。

**Steps:**

先用RandomForestClassifier起手，發現居然完全overfitting train set，再逐步透過組合(加入不同觀點看是否有幫助)LogisticRegression、svc、還有加上max\_depth限制後，將精確度調整到train 0.9265 ,test 0.89的相近水準，解決了overfitting的問題也維持了精確度。

**心得:**

我覺得此份資料的難易度和其他份的相差頗大，因為我在其他資料集中都各試了1-2小時所得到的預測精確度都只有6-7成多而已，實在是有點誇張。但是我怕這樣寫太少所以下面還繼續再做第二個problem。

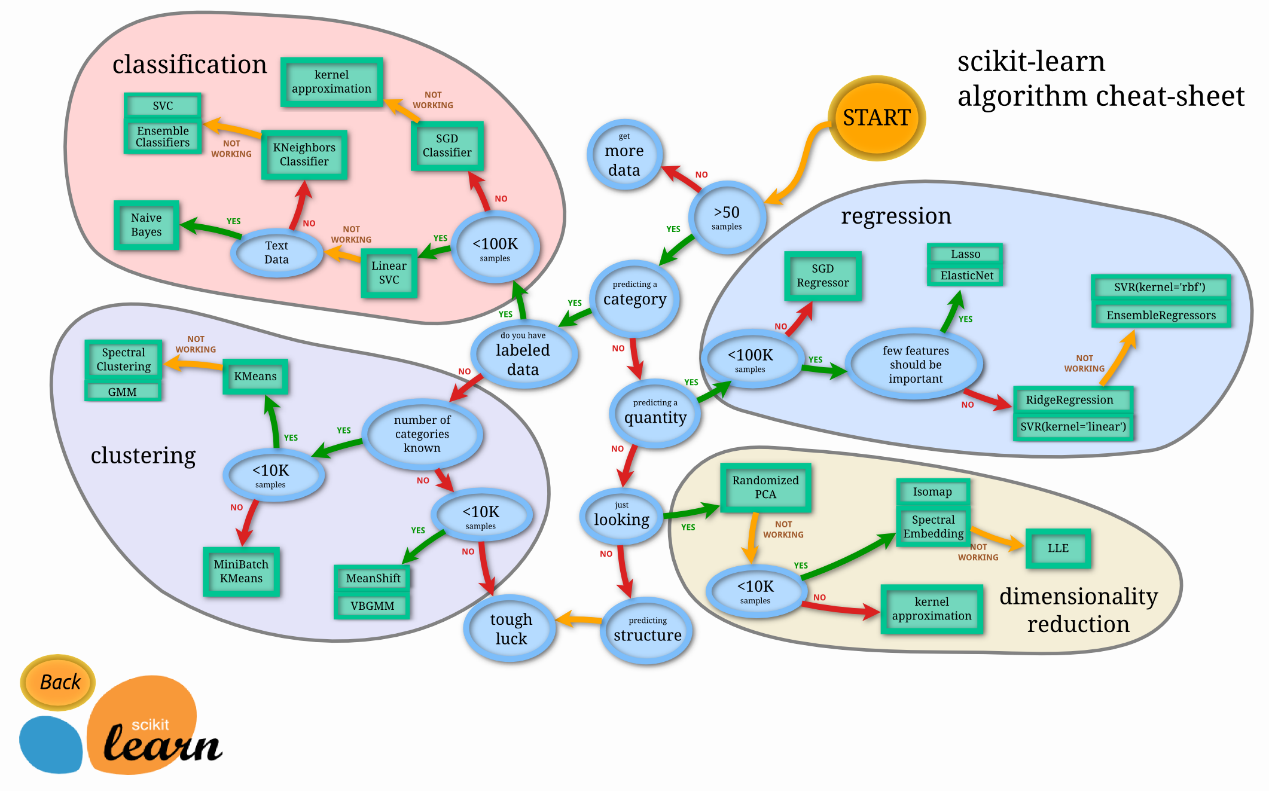
**Problem define 2:**

預測系統穩定度的值(即stab欄位) - regression

**Observation 2:**

同上，因為stab, stabf欄位本來就高度相關，故採用相同的欄位當特徵。

**Steps:**



根據上圖，分別依序嘗試了SGDRegressor、LinearRegression(其中分別測試了1-6次式逼近的結果)、svr、lasso，透過r2\_score(越接近1越好)、mse(越小越好)來衡量訓練效果。

最後發現是使用LinearRegression的準確度最好(4或5次式逼近)，r2\_score不管是在訓練跟測試集都可以到0.95以上，但是6次式表現就會下降了。

**心得:**

Regression的預測精確度一開始並不是很高，因為預設只會用1次式去訓練逼近，後來改用2-6次方才發現4-5次式是最好的，不管是在訓練跟測試集都表現得很好。