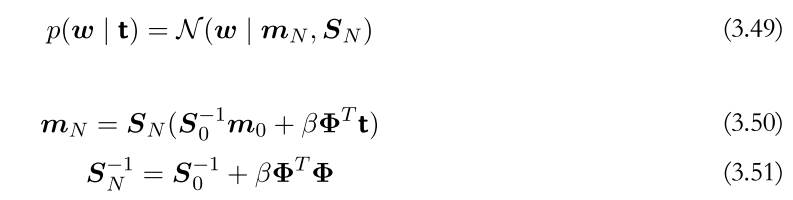
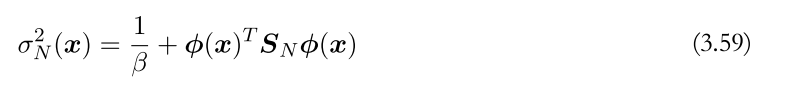
# ML HW2

1. **Sequential Bayesian Learning**

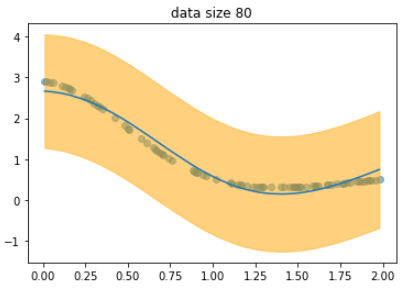
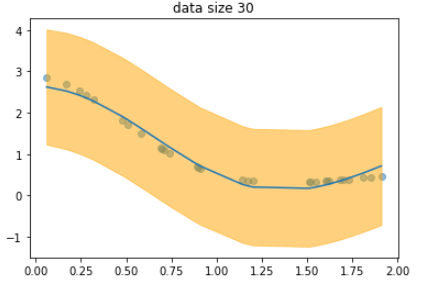
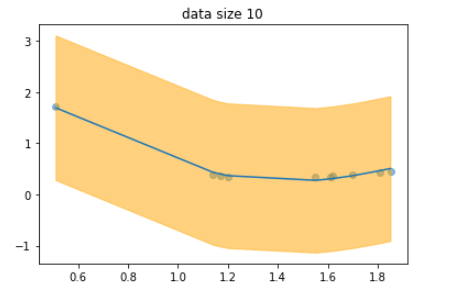
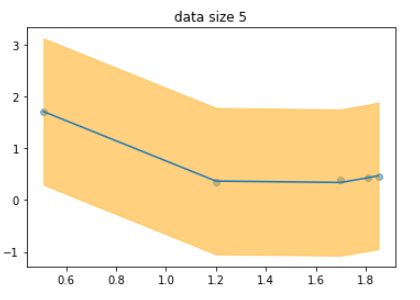
**Bayesian Learning會先利用x,y計算出posterior 分佈，我們會得到variance 跟 mean。根據公式如下，其中S0一開始設定爲identity matrix**



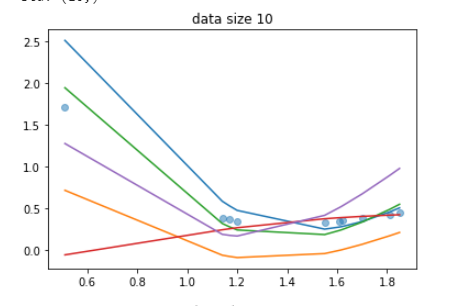
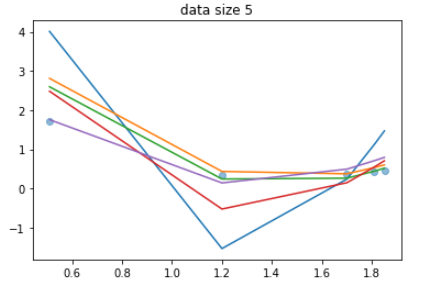
**接下來利用phi(x)，variance和mean來預測，先用multivariate 分佈來取得weight，將phi(x)\*weight得到預測答案。**

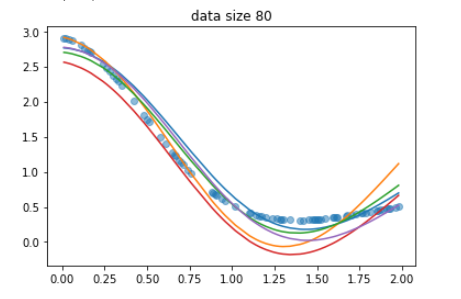
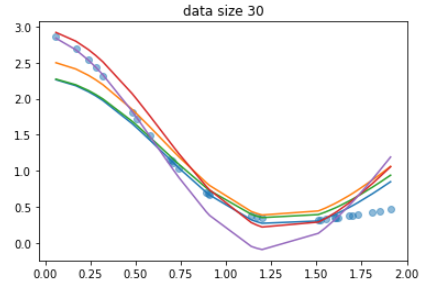
**Predictive distribution利用3.59公式計算來得到方差，我們將方差開根號可以得到標準差，用來繪製此圖。**

**可以從中觀察到資料增加時，標準差越來越小。**



**依序從data抽取5, 10, 30 ,80筆資料做訓練，可以從結果圖看到一開始只有5跟10筆資料很難去fit。Model 預測的五條線難以收斂，但到data有30到80筆時候可以看到資料開始慢慢收斂**

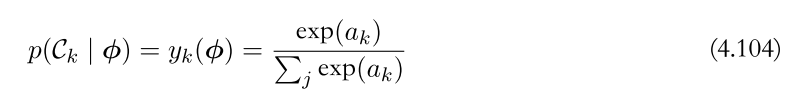




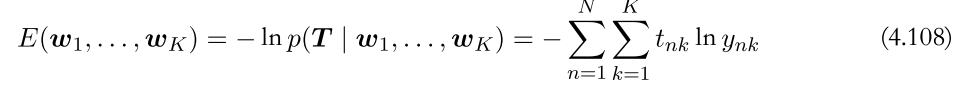
1. **Logistic Regression**

**我們使用圖片作爲分類任務訓練資料，資料匯入後會先將其坦平((28, 28) -> 784)後做一般化並將dataset隨機選取各類32張作爲測試資料其餘作爲訓練資料。**

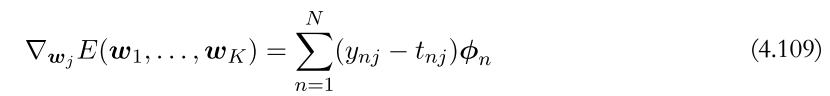
**我們可以利用 softmax來取得預測結果。**



**計算loss利用cross entropy來計算**

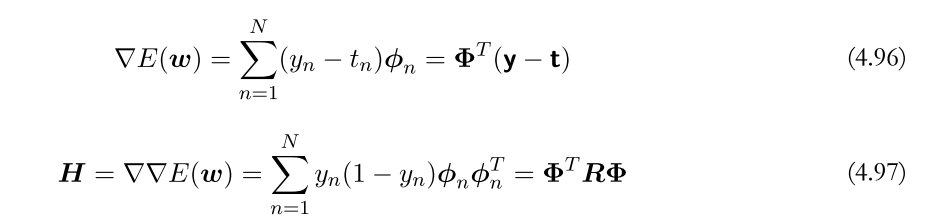


**將此公式求導，得到公式4.109，我們將用此公式來更新weight。**



**Newton-Raphson利用求導一次的E以及求導兩次的海珊矩陣來更新weight**

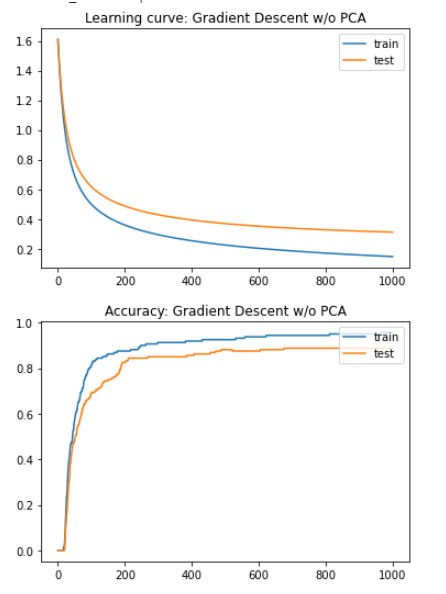
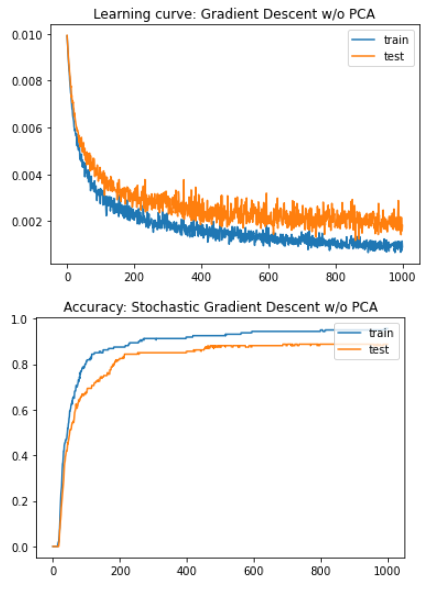
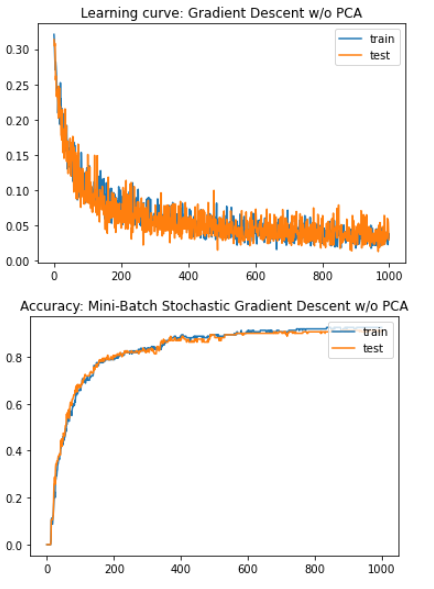
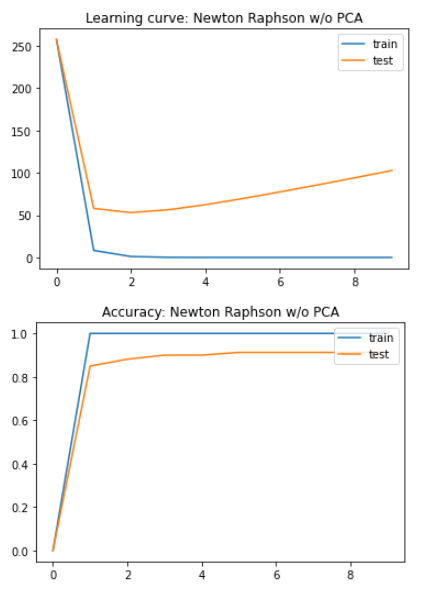




**Result: 這四張圖分別代表batch gradient, stochastic gradient, min-batch stochastic gradient, newton Raphson。**

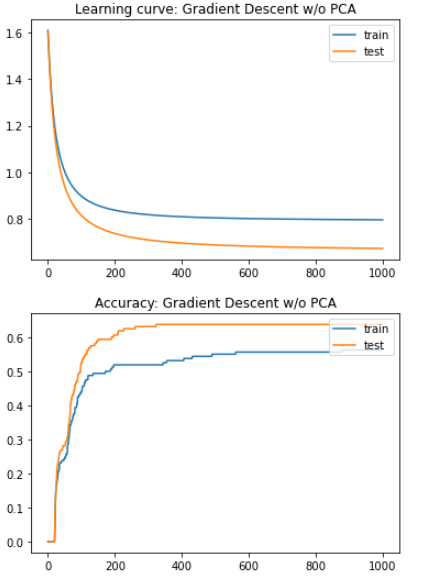
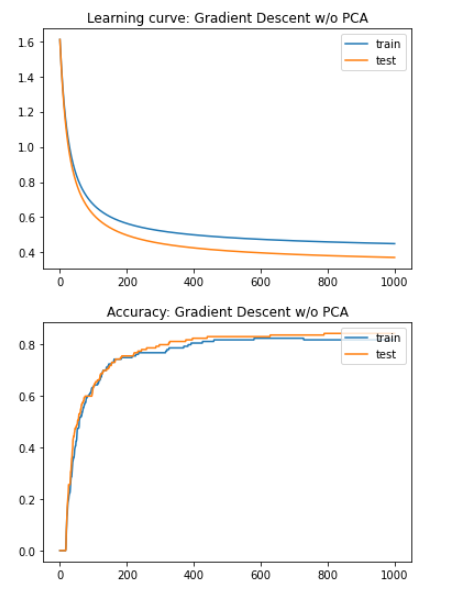
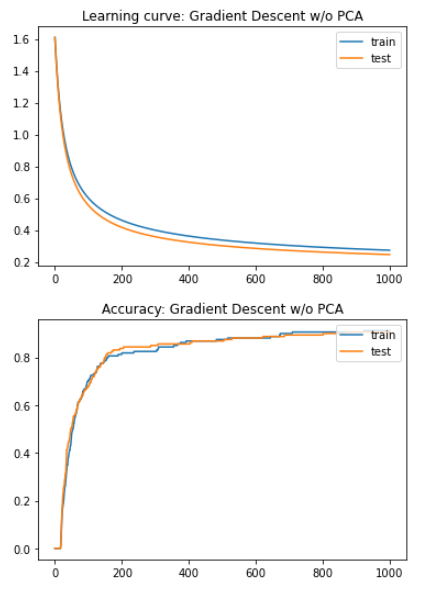
**有一點比較奇怪的是batch gradient跟stochastic gradient我得出來的accuracy是一樣的，我認爲可能是資料量過小。**

**我有發現做過一般化後訓練資料準確度會從0.8提升到0.9。**

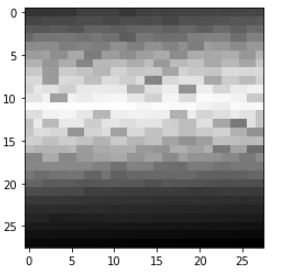
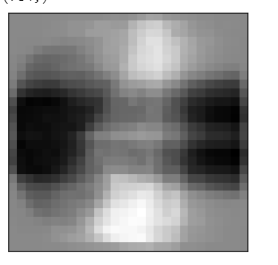
   

**PCA：利用training data計算出eignvectors，**並利用 top d eigenvalue 對應的 eigenvectors ，將 training data 及 test data 投影上去以降至 d 維。

**Result: 這三張圖是使用batch gradient分別做pca2, 5, 10的結果，可以發現如果將維度降爲2維準確度只有0.6，大概維度爲10準確度比較接近原始資料而且訓練速度快很多。**

**在plot** eigenvectors corresponding to top d eigenvalues 中我有拿我的方法跟sklearn比較，如下。這邊可能還要研究一下到底是哪裏出問題，因爲準確度是差不多的可能是方法不同。

1. **Decision regions and data points**

