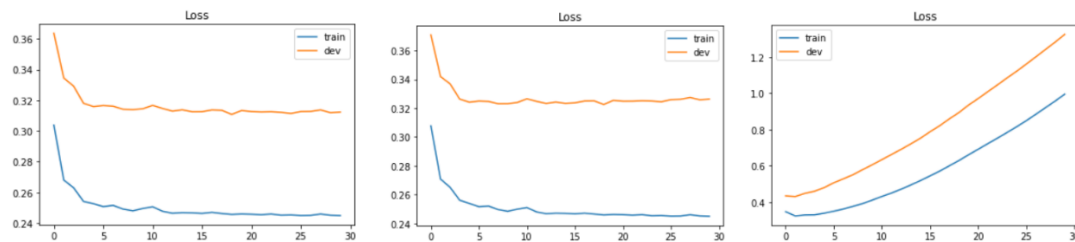


1. (2%) 請比較實作的 **generative model** 及 **logistic regression** 的準確率，何者較佳？請解釋為何有這種情況？

Generative model 的 training accuracy: 0.8732746262543518，而 **logistic regression** 的 training accuracy: 0.8844357976653696。可以看到 **logistic regression** 的準確率明顯高於 **generative model**，可能的原因是因為 **generative model** 有預先假設資料是什麼機率分布，但說不定這資料根本就不是這個機率分布，造成在計算時有過多的「腦補」以至於遠離實際狀況，這種方法在資料小或是少的時候可能會比較好，因為「腦補」可以彌補資料少的缺陷。而 **logistic regression** 則是建一份數據說一句話，所以當資料大的時候他相較於 **generative model** 會是一個比較好的方法。

2. (2%) 請實作 **logistic regression** 的正規化 (regularization)，並討論其對於你的模型準確率的影響。接著嘗試對正規項使用不同的權重 (λ)，並討論其影響。(有關 regularization 請參考 <https://goo.gl/SSWGhf> p.35)

下圖由左至右分別是用 $\lambda = 0.01, 0.1, 0.5$ 做出來的圖，其中我測出來他們的 loss 隨著 λ 的增加而上升，而正確率則是跟著下降。使用正規化的目的是希望在作 training 時我們的 training 曲線可以光滑一點，這樣就可以避免 overfitting 但是若 λ 的比重太大則會出現最右邊那種狀況，training 曲線過於平滑使得 loss 一直增加，理論上我們在做正規化的時候會希望試出一個 λ 讓 loss 變最少，意思就是說找到曲線適合的平滑度，但是從我的 model 中我找不到那個適合的 λ ，正確度一直隨著減少，可能是我的 model 並不是和使用這種方法來優化的關係。



3. (1%) 請說明你實作的 **best model**，其訓練方式和準確率為何？

我分數最高的 model: 0.89139。我拿每個標準化後的 feature 去跟要預測的東西作

covariance，如果 covariance 取絕對值之後小於 0.01 我就把那個 feature 刪掉，然後 $\text{max_iter} = 100$ ， $\text{batch_size} = 50$ ， $\text{learning_rate} = 0.05$ (這些數字好像沒什麼技巧就是一直試試出來的)

4. (1%) 請實作輸入特徵標準化 (feature normalization)，並比較是否應用此技巧，會對於你的模型有何影響。

左圖是有 **normalize** 的 **features** 對 **loss** 的作圖右邊的則是沒有 **normalize** 過的 **features** 作的圖，我們可以看出左邊的數據明顯晃動的劇烈很多，原因是因為有些 **feature** 本身量值就比較大變化也比較劇烈，這樣的數據就不行跟那些數值小變化小的數值用同樣的 **learning rate** 計算，不然會影響過大，所以 **normalize** 的用意就是為了避免這樣的事情發生，把所有的數據的平均以及標準差都變成一樣，這樣每個 **feature** 所佔的權重也會一樣跑出來的數據也會比較可信合理。

