1. 請從 Network Pruning/Quantization/Knowledge Distillation/Low Rank Approximation 選擇兩個方法(並詳述)，將同一個大 model 壓縮至同等數量級，並討論其 accuracy 的變化。 (2%)

Quantization： Acc Val：0.8801 Size：21.36M

Knowledge Distillation： Acc Val：0.8376 Size：49M

在這題中我選擇Quantization以及Knowledge Distillation將助教所提供的teacher\_resnet18.bin做壓縮並且將正確率做比較。Quantization的部分我則是用助教所寫的function encode16進行壓縮，Knowledge Distillation我則是用我作業三的架構向助教所提供的teacher\_resnet18.bin進行學習。

由數據可以看到做Quantization的效果其實非常的好，正確率跟原本(0.8841)幾乎沒有改變。而Knowledge Distillation效果則差上許多。由此可知壓縮model的效果Quantization > Knowledge Distillation。

我想Knowledge Distillation之所以會那麼差可能是因為在學習的過程中包含了許多的訓練技巧，包括alpha的設置，optimizer的選擇，learning rate的大小等等，有種種原因可以使我們的學習成效不好，所以乍看之下Knowledge Distillation可能效果比較不好，但經過適當的調整，正確率可以再提升也說不定。

以下三題只需要選擇兩者即可，分數取最高的兩個。

1. [Knowledge Distillation] 請嘗試比較以下 validation accuracy (兩個 Teacher Net 由助教提供)以及 student 的總參數量以及架構，並嘗試解釋為甚麼有這樣的結果。你的 Student Net 的參數量必須要小於 Teacher Net 的參數量。(2%)  
   x. Teacher net architecture and # of parameters: torchvision’s ResNet18, with 11,182,155 parameters.  
   y. Student net architecture and # of parameters: 275,111  
   a. Teacher net (ResNet18) from scratch: 80.09%

b. Teacher net (ResNet18) ImageNet pretrained & fine-tune: 88.41%

c. Your student net from scratch:

最後的Val Acc為0.6459

d. Your student net KD from (a.):

最後的Val Acc為0.7898

e. Your student net KD from (b.):

最後的Val Acc為0.8301

討論：

我們所使用的student net大小差不多為1M，我們從實驗結果可以看到有越好的老師，學生的Acc會越好，並且有老師勝過沒有老師(e > d > c)。我想這個結果是符合直覺的，因為有越好的老師，如果Knowledge Distillation做得好的話，學生的正確率可以越來越逼近老師，而如果老師自己本身的正確率就不是很好的話(80.09%)，它所訓練出來的學生d自然就很難超越學生e。

而學生c沒有使用KD，而且自己本身的模型架構又不大，造成難以收斂到一個正確的點形成好結果，所以它的正確率自然會最小。

1. [Network Pruning] 請使用兩種以上的 pruning rate 畫出 X 軸為參數量，Y 軸為 validation accuracy 的折線圖。你的圖上應該會有兩條以上的折線。(2%)
2. [Low Rank Approx / Model Architecture] 請嘗試比較以下 validation accuracy，並且模型大小須接近 1 MB。 (2%)

a. 原始 CNN model (用一般的 Convolution Layer) 的 accuracy

b. 將 CNN model 的 Convolution Layer 換成參數量接近的 Depthwise & Pointwise 後的 accuracy

c. 將 CNN model 的 Convolution Layer 換成參數量接近的 Group Convolution Layer  (Group 數量自訂，但不要設為 1 或 in\_filters)

以下為三種模型的正確率比較(統一跑100個epoch)：

a. 0.5206

b. 0.6924

c. 0.6430

使用原始CNN模型架構因為大小的限制，所以沒辦法做很多層，以至於正確率較低。

而使用Depthwise & Pointwise 後的 accuracy有明顯上升，因為他有為了將模型變小而特化過，所以可以疊出比較多層，效果也比較好。

在同樣大小下Group Convolution Layer可以疊出最多層的model(我疊了21層)，但是我好像有點疊太多層了以至於run 100 個epoch還沒有收斂，有點underfitting，這裡為了討論公平性，所以只討論在100當下的正確率，但是如果想要真的train好一個model可能需要在增加epoch才可以達到比較好的結果。