Code links:

1. <https://colab.research.google.com/drive/1t2VtvOIk68r5GWWokB4VtNWz1qq5vIDV#scrollTo=lHINCCLBR383>
2. <https://colab.research.google.com/drive/10pv5uoskvwgHQGDAWwsxvTYGsL731FBl#scrollTo=ePZk0ufA82j9>
3. <https://colab.research.google.com/drive/1NQcuwUWME_PfqX_wQq137ml6OtNvu_tb#scrollTo=Zf0HuGIlyF0C>

Below are the plots for accuracy:

1. Simple Network - learning\_rate = 0.001
2. Pretrained = False - learning\_rate = 0.0001
3. Pretrained = True - learning\_rate = 0.0001

Chart, line chart

Description automatically generated

Fig: Simple Network - learning\_rate = 0.001

Chart, line chart

Description automatically generated

Fig: Pretrained = False - learning\_rate = 0.0001

Chart, line chart

Description automatically generated

Fig: Pretrained = False - learning\_rate = 0.0001

**Conclusion:**

1. **Pretrained= true** model works best amongst three, considering the accuracy of all three networks. Please note we are keeping same number of epochs.
2. The pre-trained model is much simpler to set up.
3. Using a pre-trained model requires less training and requires less effort in building the model’s architecture
4. Using a pre-trained model is significantly more accurate than uother two here.
5. Setting pretrained=False will tell PyTorch not to download pre-trained model.
6. When you set pretrained=False, PyTorch will download pretrained ResNet18 on ImageNet. And by default, it'll freeze first two blocks named conv1 and layer1

Accuracy table with epochs = 30 and different learning rate.

|  |  |
| --- | --- |
| **Learning\_rate = 0.0001** | |
| pretrained=True | 0.8252 |
| pretrained=false | 0.644 |
| simple network | 0.559 |
|  | |
| **Learning\_rate = 0.001** | |
| pretrained=True | 0.8301 |
| pretrained=false | 0.7794 |
| simple network | 0.6282 |
|  | |
| **Learning\_rate = 0.00001** | |
| pretrained=True | 0.7582 |
| pretrained=false | 0.488 |
| simple network | 0.4092 |
|  | |
| **Learning\_rate = 0.01** | |
| pretrained=True | 0.7913 |
| pretrained=false | 0.783 |
| simple network | 0.6036 |

With different learning rate, pretrained model ( i.e pretrained=True) performs best.