MLOps and DLOps Assignment 1

Performance and Analysis of Deep Learning Models

Name: Sonam Sikarwar **Roll No:** B23CM1060

Introduction

This assignment evaluates deep learning and classical machine learning models under different training and computational conditions using MNIST and FashionMNIST datasets. ResNet-18, ResNet-50 and Support Vector Machines (SVMs) are compared across hyperparameters, optimizers and compute environments.

Here are the notebook links for all the experiments :

1. Resnet SVM training Notebook: colab

2. Performance differences with CPU and GPU: colab

GitHub Repository: MLOps Assignment–1 Repository

Datasets

MNIST (70,000) handwritten digit images and FashionMNIST (70,000) clothing images of size 28×28. Data split: 70% training, 10% validation, 20% testing.

I have performed various experiments by varying pin memory, and with epochs as 5 and 10, respectively (refer to the Colab notebook). Here, we present the results with the number of epochs fixed at 10 and pin memory as True.

Q1(a): MNIST Test Accuracy

Batch Size	Optimizer	Learning Rate	ResNet-18 Acc (%)	ResNet-50 Acc (%)	pin_memory
16	SGD	0.001	99.38	99.05	True
16	SGD	0.0001	98.97	99.11	True
16	Adam	0.001	99.18	99.20	True
16	Adam	0.0001	99.61	99.56	True
32	SGD	0.001	99.36	99.59	True
32	SGD	0.0001	99.02	99.36	True
32	Adam	0.001	99.74	99.61	True
32	Adam	0.0001	99.62	99.58	True

Analysis:

- Adam optimizer leads to faster convergence and higher final accuracy than SGD.
- ResNet-50 has the capacity to handle complex image patterns but takes more time than ResNet-18.

- A higher learning rate (0.001) performs better than a lower learning rate (0.0001), providing both speed and accuracy.
- Batch size variations (16 vs 32) have minimal effect on final accuracy, showing stability on MNIST dataset. Setting `pin_memory=True` in DataLoader boosted GPU performance by enabling faster transfer of data from the VM's CPU memory to the GPU memory.
- **Best Model:** ResNet-18, Adam optimizer, LR=0.001, Batch=32, pin_memory=True achieves 99.74% test accuracy.

Q1(a): FashionMNIST Test Accuracy

Batch Size	Optimizer	Learning Rate	ResNet-18 Acc (%)	ResNet-50 Acc (%)	pin_memory
16	SGD	0.001	92.08	90.16	True
16	SGD	0.0001	87.61	88.92	True
16	Adam	0.001	92.87	91.12	True
16	Adam	0.0001	92.44	91.08	True
32	SGD	0.001	91.89	92.88	True
32	SGD	0.0001	86.78	88.76	True
32	Adam	0.001	91.99	92.04	True
32	Adam	0.0001	92.36	91.62	True

Analysis:

- ResNet-50 has the capacity to handle complex image patterns but takes more time than ResNet-18.
- Adam optimizer gives higher test accuracy than SGD in almost all experiments.
- Higher learning rate (0.001) achieves better final accuracy compared to 0.0001.
- Increasing batch size to 32 slightly reduces performance in most cases, but results remain above 91%.
- **Best Model:** ResNet-16, Adam optimizer, LR=0.001, Batch=16, pin_memory=True achieves 92.87% test accuracy.

Q1(b): SVM Performance

Q1(b): SVM Results on MNIST and FashionMNIST

Dataset	Kernel	C	Degree	Accuracy (%)	Train Time (ms)
MNIST	RBF	0.1	–	91.02	67,697
MNIST	RBF	1.0	–	95.18	40,595
MNIST	RBF	10.0	–	96.08	38,397
MNIST	Poly	0.1	2	90.54	112,352
MNIST	Poly	0.1	3	73.10	165,372
MNIST	Poly	0.1	4	35.90	189,232
MNIST	Poly	1.0	2	95.89	48,666
MNIST	Poly	1.0	3	93.46	83,336
MNIST	Poly	1.0	4	81.40	128,708
MNIST	Poly	10.0	2	**96.45**	33,757
MNIST	Poly	10.0	3	96.41	55,124
MNIST	Poly	10.0	4	93.43	89,727
FashionMNIST	RBF	0.1	–	81.82	54,466
FashionMNIST	RBF	1.0	–	86.55	35,037
FashionMNIST	RBF	10.0	–	**87.86**	34,137
FashionMNIST	Poly	0.1	2	79.44	72,853
FashionMNIST	Poly	0.1	3	75.38	82,871
FashionMNIST	Poly	0.1	4	68.06	112,170
FashionMNIST	Poly	1.0	2	86.09	35,820
FashionMNIST	Poly	1.0	3	84.83	41,173
FashionMNIST	Poly	1.0	4	80.11	63,275
FashionMNIST	Poly	10.0	2	87.54	29,386
FashionMNIST	Poly	10.0	3	87.32	35,277
FashionMNIST	Poly	10.0	4	85.75	44,664

Q1(b) Analysis

- **Best Performance:**
  MNIST: Polynomial kernel with $C = 10$ and degree 2 achieves **96.45%** accuracy.
  FashionMNIST: RBF kernel with $C = 10$ achieves **87.86%** accuracy.

- **Effect of Regularization Parameter C:**
  Increasing C improves accuracy. Small C (e.g., 0.1) causes underfitting. Large C allows learning more expressive decision boundaries.

- **Polynomial Kernel Behavior:**
  Higher polynomial degrees increase training time. Overfitting occurs at higher degrees, reducing accuracy. Degree 2 provides the best trade-off between accuracy and computational cost.

- **RBF vs Polynomial Kernels:**
  RBF performs better on FashionMNIST due to complex nonlinear patterns.
  Polynomial performs better on MNIST because of simpler digit structure.

- **Training Time:**
  Polynomial kernels take significantly longer to train than RBF, especially for higher degrees. RBF scales more efficiently as C increases.

- **Comparison with Deep Learning:**
  SVMs achieve strong accuracy (up to 96.45% on MNIST), but deep learning models outperform them. Deep models exceed 99% accuracy on MNIST and 92% on FashionMNIST. GPU acceleration further enhances deep learning performance.

Q2: CPU vs GPU Performance on FashionMNIST

Performance and Training Time

Compute	Batch	Optimizer	LR	ResNet-18 Acc (%)	ResNet-32 Acc (%)	ResNet-50 Acc (%)	ResNet-18 Time (ms)	ResNet-32 Time (ms)	ResNet-50 Time (ms)
CPU	16	SGD	0.001	83.90	20.56	80.13	1,103,327	4,418,103	4,065,350
CPU	16	Adam	0.001	75.27	26.42	79.46	1,036,824	4,508,394	3,985,297
GPU	16	SGD	0.001	83.75	22.19	82.26	59,704	163,156	132,047
GPU	16	Adam	0.001	80.77	18.73	74.06	52,771	164,316	136,962

FLOPs Analysis

Model	FLOPs (G)
ResNet-18	1.751
ResNet-32	3.311
ResNet-50	3.967

Analysis:

- Training on GPU is drastically faster, achieving ˜18–35x speedup depending on the model, while maintaining similar accuracy for ResNet-18 and ResNet-50.
- ResNet-18 achieves the highest stable accuracy on both CPU (83.90%) and GPU (83.75%). ResNet-50 performs comparably, while ResNet-32 shows poor performance (18–26%), likely due to instability with this architecture on FashionMNIST.
- SGD slightly outperforms Adam in terms of final accuracy for ResNet-18 and ResNet-50. Adam occasionally leads to lower accuracy despite similar or slightly faster training times on GPU.
- ResNet-18 has the lowest FLOPs (1.751 G), making it the most efficient choice, while ResNet-50 offers slightly better accuracy at a higher computational cost (3.967 G FLOPs). ResNet-32 has high FLOPs (3.311 G) but performs poorly, making it inefficient for this task.