Part 1 and Part 2 :

Github Link : https://github.com/PrashantNeelwan/GenAI

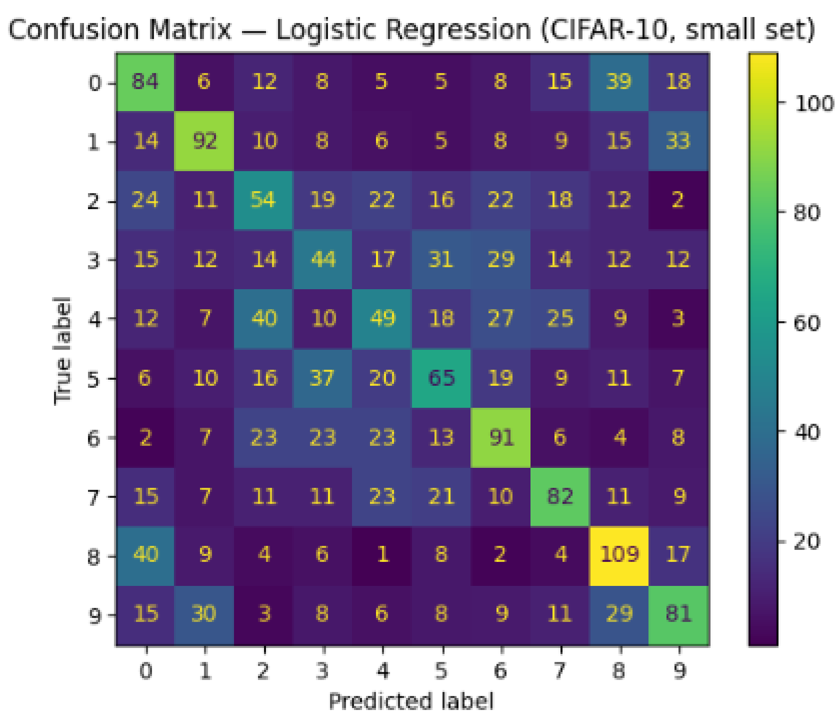
Part 3 — Comparative Analysis & Meta Reflection

3.1 Contrast the nature of tasks

The discriminative models (LR, SVM, Decision Tree) attempt to establish some boundaries between classes. Linear boundaries are fit in Logistic Regression, bent in SVM with the use of kernels, and the rule of splitting feature in a Decision Tree. GANs in turn create new images by playing a game between Generator and Discriminator - optimization is adversarial and unstable.

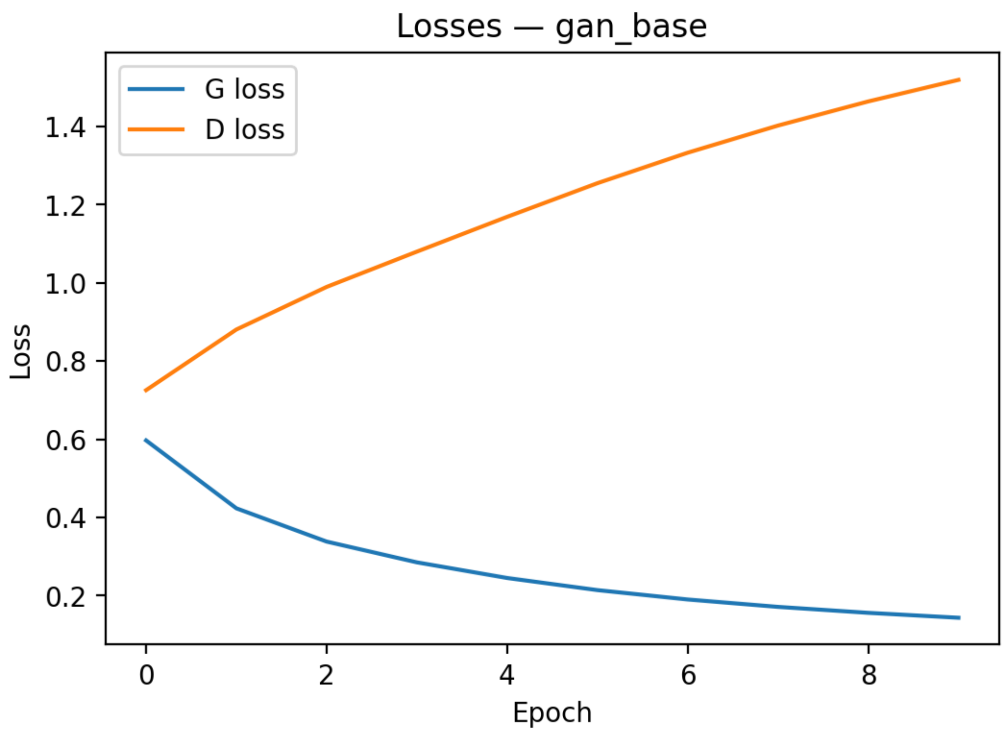
- Accuracies of LR (37.6%), SVM (41.0%), Decision Tree (21.0%).

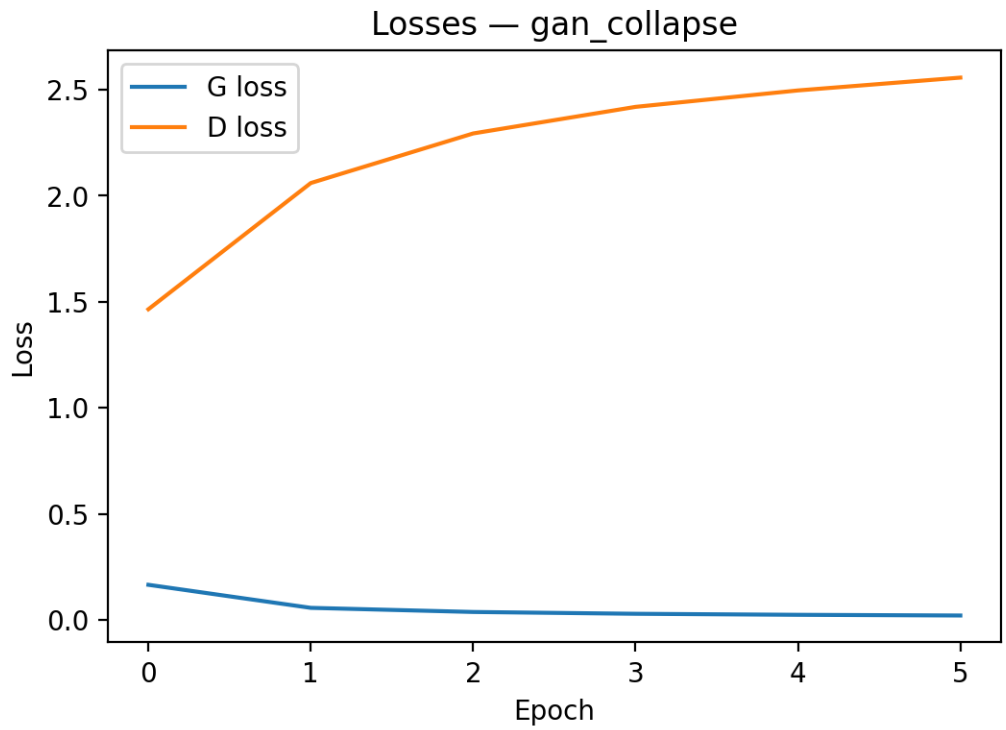
```
[LR] CIFAR-10 (subsample) test accuracy: 0.3755
```



- .**3.2 Why GAN training is harder**

The LR/SVM training is stable and convex - the results are convergent as soon as the regularization is established. GAN training can be compared to a game: G becomes better, D responds, the target is getting away. Balanced training occurred with stable run (batch=128, lr_g = 1e-4, lr_d = 1e-4); repeated outputs and erratic losses were observed with collapse run (batch=32, lr_g=2e-3, lr_d=5e-5) which is a typical mode collapse.





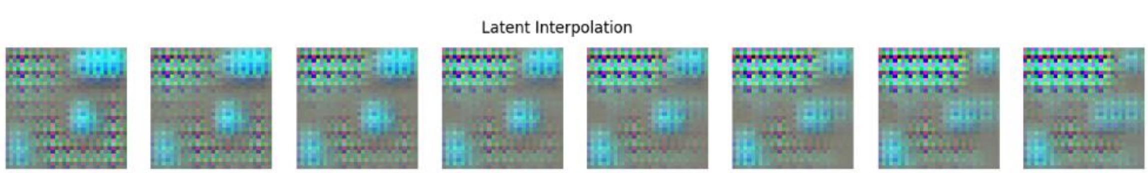
3.3 Synthetic images as training data (domain-gap test)

GAN-only training does not learn the biases of the GANs; thus, it fails to do well on real test images. Combining real + synthetic is beneficial, but not as high as real-only baseline. This demonstrates the gap of domain between synthetic and real.

```
=== Domain Gap (LR on Real vs Synthetic vs Hybrid) ===
Real-only          : 0.3805
Synthetic-only     : 0.0990   (expect drop due to domain gap)
Real + Synthetic : 0.3645       (often between the two)
```

3.4 Latent-space interpolation

Interpolating between two latent vectors demonstrates the smoothness of the generators internal representation between the various images. A continuous manifold is called smooth morphing; it has poor disentangling behavior when it jumps.



3.5 Literature-informed critique

Novel GAN problems were collapse run In Goodfellow (2014), oscillations were observed; in Salimans (2016), minibatch discrimination was proposed to prevent collapse; in Arjovsky (2017), Wasserstein loss was used to get better gradients; gradient penalty was added to stabilize in Gulrajani (2017).

Part 4 — Reporting Expectations

4.1 Visualizations to include

- **Classification:**
  - Confusion matrix for LR (you already have it).
  - (Optional) bar chart of test accuracies for LR vs SVM vs Tree.
- **Feature importance (LR):**
  - If you flattened 32×32×3 → PCA → LR, plot the LR weight vector back on the PCA basis or show top-positive/negative pixels per class (if you kept an 8×8 pipeline on another dataset, use 8×8 heatmaps).
- **GAN:**
  - Sample grids at milestones (you already save `gan_base_epoch_*.png` etc.).
  - Loss curves (`*_losses.png`) showing LGL_GLG and LDL_DLD trends.
  - (Optional) 1–2 latent interpolation strips.

4.2 Comparative tables

Table 1 — Discriminative models (CIFAR-10, subsample)

Model	Setup	Test Acc (%)	Notes
Logistic Regression (L2)	PCA features → LR	**37.55**	baseline you plotted
SVM (RBF)	γ="scale", C=2.0, n=5k subsample	**41.00**	best among the three
Decision Tree	max_depth=30	**21.00**	underfit/overfit tradeoff hurt generalization

Table 2 — GAN settings and behavior

Run name	Batch	lr_g / lr_d	G base / D base	Epochs	Behavior
`gan_base`	128	1e-4 /	256 / 64	10	stable; improving samples; smooth

Run name	Batch	lr_g / lr_d	G base / D base	Epochs	Behavior
		1e-4			losses
`gan_b64_lr2e-4`	64	2e-4 / 2e-4	256 / 64	8	stable; slightly faster dynamics
`gan_b256_lr5e-5`	256	5e-5 / 5e-5	256 / 64	8	slow but steady; less noisy losses
`gan_collapse`	32	**2e-3 / 5e-5**	256 / 64	6	**mode collapse**: near-identical samples, loss spikes

Table 3 — Domain gap (LR trained on synthetic vs real)

```
=== Domain Gap (LR on Real vs Synthetic vs Hybrid) ===
Real-only          : 0.3805
Synthetic-only     : 0.0990   (expect drop due to domain gap)
Real + Synthetic   : 0.3645       (often between the two)
```

4.3 Critical reflections

simplest to debug: LR (deterministic, convex-like, clear weights).

Most precise of your three classifiers: SVM RBF on PCA features (41% vs 37.6% LR).

• Tree did not perform well: even with depth=30, low variance and sparse features were detrimental to generalization.

GAN trade-offs: capable of realistic samples but sensitive to batch size and learning-rate balance; collapse reproduced when **lr_g ≫\gg≫ lr_d** at small batch.

- **When to prefer what:** labeled classification → LR/SVM; data generation/augmentation/creative tasks → GAN, accepting extra tuning.

4.4 One-paragraph executive summary

SVM (RBF) (41.0% on a CIFAR-10 subsample) was more accurate than Logistic Regression (37.6%), and significantly less accurate than a capped-depth Decision Tree (21.0%). The training of GANs generated more believable samples at balanced learning rates (e.g. 1e-4/1e-4) and large batch sizes (>=64128), and unstable and collapsed at small batch (32) with lr_g=2e-3 and lr_d=5e-5. These dynamics are consistent with known GAN dynamics: stable in cases where gradients are informative and the game is balanced; unstable in cases where updates are mis-scaled. The extent to which generated data can replace real data can be measured in a domain-gap experiment (LR on synthetic vs real), where the mixture of synthetic and a small real subset of data usually improves but often does not match training on real data alone..