Problem 1

(A) Why certain letters (like O, A) survive mode collapse while others (Q, X, Z) disappear

O (a smooth closed loop) and A (few straight edges) form tight, compact clusters in feature space. A generator that "averages" shapes still lands near these clusters, so they keep showing up.

With Feature Matching, we align the mean of discriminator features. Means emphasize dominant, stable contours (circles/triangles) and down-weight small, class-specific details such as Q's tiny tail, Z's diagonals, and X's crossing.

(B) Quantitative comparison of mode coverage with and without your chosen fix

The fix I choose was feature matching, below are the results according to coverage.json:

- **Unique letters covered:** vanilla **17/26** vs. fixed **20/26.**
- **Coverage score:** vanilla **0.654** → fixed **0.769**.

Missing letters

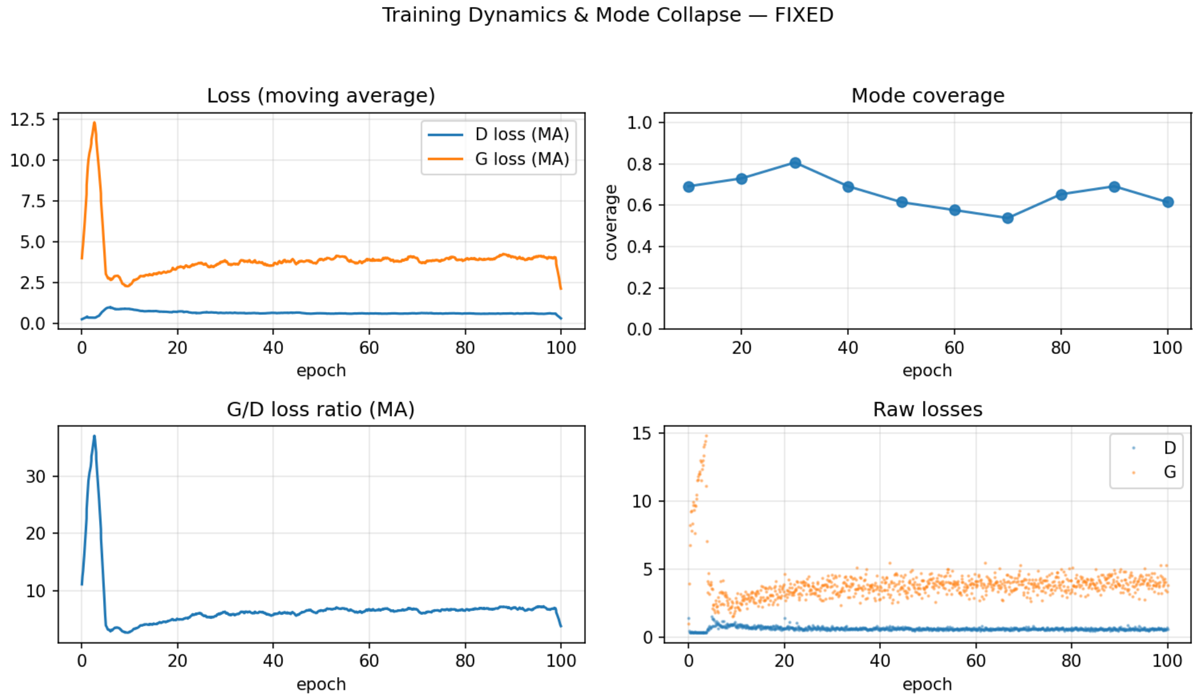
- Vanilla missing {A, B, G, M, N, Q, R, V, W}.
- Fixed missing {B, G, N, Q, W, Z}.
  → The fix **recovers A, M, R, V**, but **Z** is still absent.

Mode concentration (top class share): the model still heavily prefers **J**

- Vanilla: **41.0%** of samples are J.
- Fixed: **49.7%** of samples are J (even more concentrated).
  (Counts key 9 = J; ratios from the 1000-sample histograms.)

Overall, feature matching increases the number of distinct letters generated and overall coverage, but the **dominant "J" mode remains**, and even becomes more dominant in this run.

(C) Discussion of training dynamics: when does collapse begin?



According to the mode_collapse_analysis_fixed.png above:

- **Coverage over time:** early training peaks around **epoch ~30 (≈0.8)**, then **declines through ~60–70 (≈0.55)** before partly recovering to **~0.65–0.7** near **80–90** and ending ~0.6.
- **Loss traces:**

  D loss hovers very low (~0.3–0.5) after the warm-up,

  G loss climbs to ~4 and stays elevated,

  G/D ratio rises to **~6–7** and stays high.

These are signs that collapse starts **mid-training**

(D) Evaluation of your chosen stabilization technique's effectiveness

What it helped:

- +3 modes and +11.5 percentage points coverage vs. vanilla—clear, measurable diversity gains.
- Training is more stable early (coverage reaches ~0.8 by epoch ~30), indicating FM is encouraging the generator to match feature statistics rather than only the discriminator's decision boundary.

What it didn't fully solve:

- Single-mode dominance persists (J takes ~50% of samples under the fixed model), and coverage still falls mid-training. The fix reduces but does not eliminate mode collapse in this setup.

Problem 2

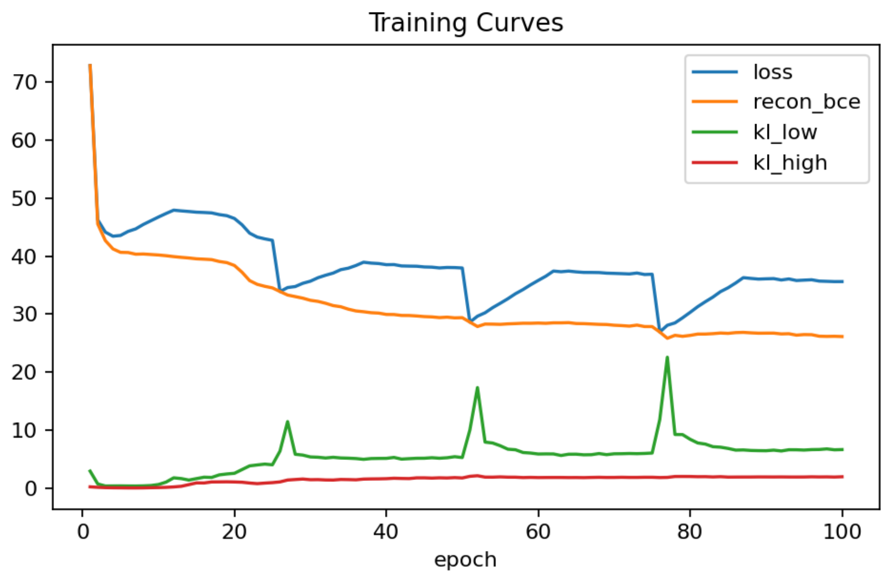
(A) Evidence of posterior collapse and how annealing prevented it

We use data from cyclical strategy to demonstrate.

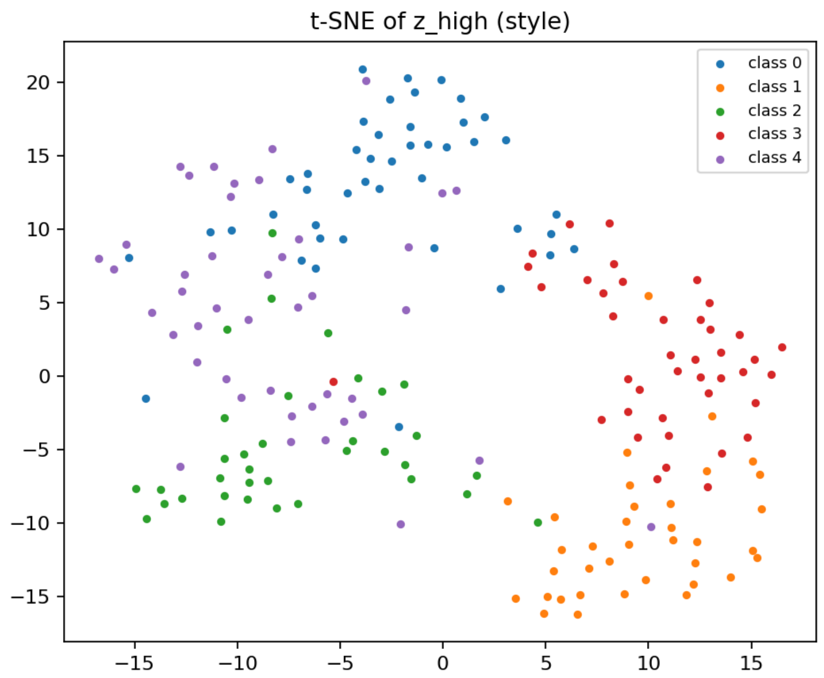
According to posterior_collapse.json:

- Threshold = **0.02**; **collapsed_high_indices = []**, **collapsed_low_indices = []** (over **200** samples).
- Per-dim mean KL: **z_high = [0.469, 0.463, 0.472, 0.475]**; **z_low = 0.516–1.272**.
  → All dimensions carry non-trivial KL (>0.02), so there's **no posterior collapse**.

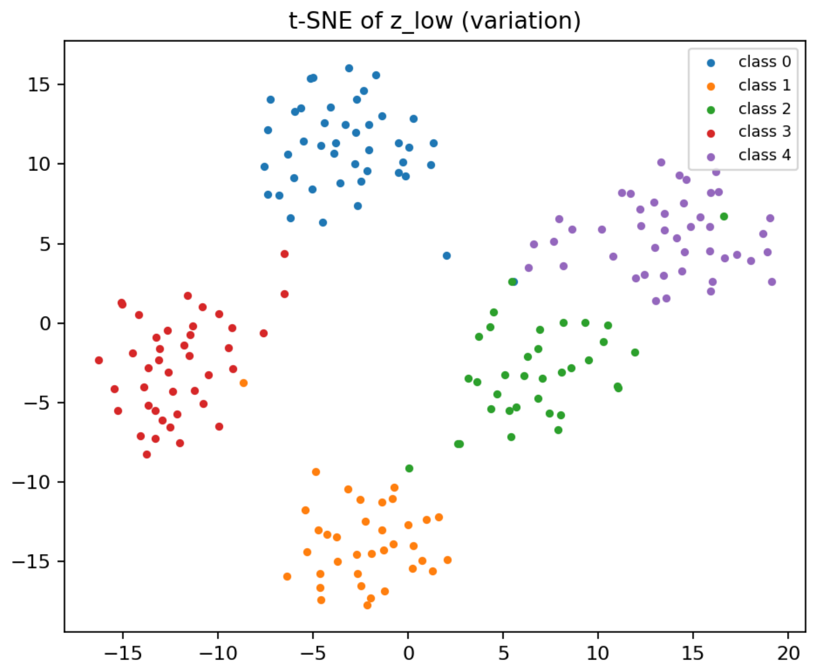
Evidence that annealing prevented posterior collapse:



- **Training Curves** – `kl_low` shows periodic spikes then stabilizes at a non-zero plateau; `kl_high` stays non-zero while `recon_bce` keeps improving. → KLs don't vanish ⇒ no collapse, and the cyclical schedule is lifting KL each cycle.



- **t-SNE of z_high** – clear, separated clusters by style. → The high-level latent carries style information rather than degenerating to a single blob.

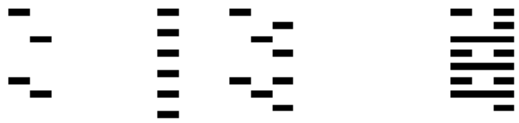


- **t-SNE of z_low** – tight, coherent clusters reflecting within-style variation. → The low-level latent encodes meaningful variation, not collapse.

(B) Interpretation of what each latent dimension learned to control

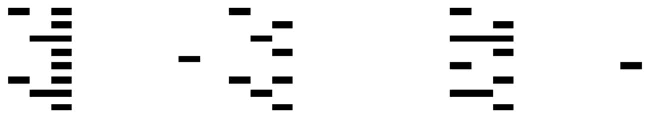
We use data from cyclical strategy to demonstrate.

z_high[0] sweep



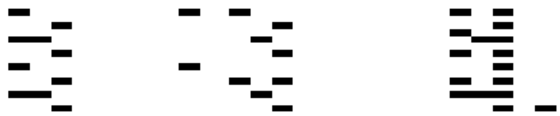
- **z_high[0]**: sparse with crash accents → tighter, denser groove → strong kick + closed-hi-hat drive.

z_high[1] sweep



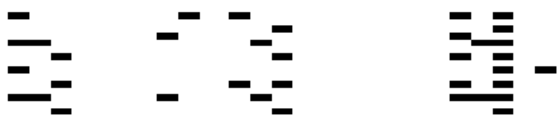
- **z_high[1]**: shifts density/accents: hat-heavy at left → balanced mid-groove → sparser, few accent hits at the extreme.

z_high[2] sweep



- **z_high[2]**: adds closed-hat drive and tom-fill flavor as it increases; splashy accents fade

z_high[3] sweep



- **z_high[3]**: moves from crash-accented patterns → clear kick+hat groove with light snare/toms.

(C) Quality assessment: Do generated patterns sound musical?

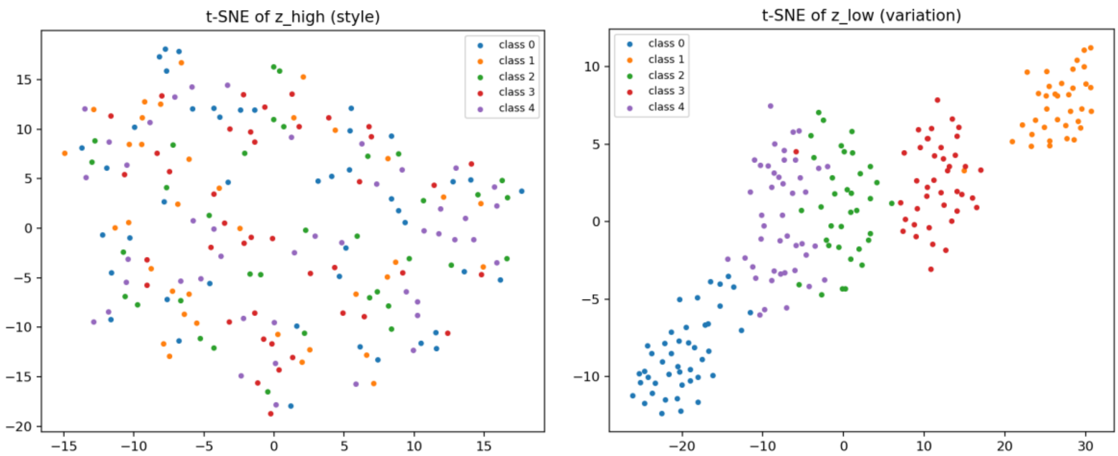
According to plugin_metrics.json:

- Mean validity across 50 samples: **0.996**; per-style means **[1.00, 1.00, 1.00, 1.00, 0.98]** → patterns almost always satisfy basic rhythmic plausibility (kick presence, backbeat, reasonable density).
- Diversity: **0.051** (mean pairwise Hamming distance) → variety is modest; many grooves share a similar kick/hat skeleton.

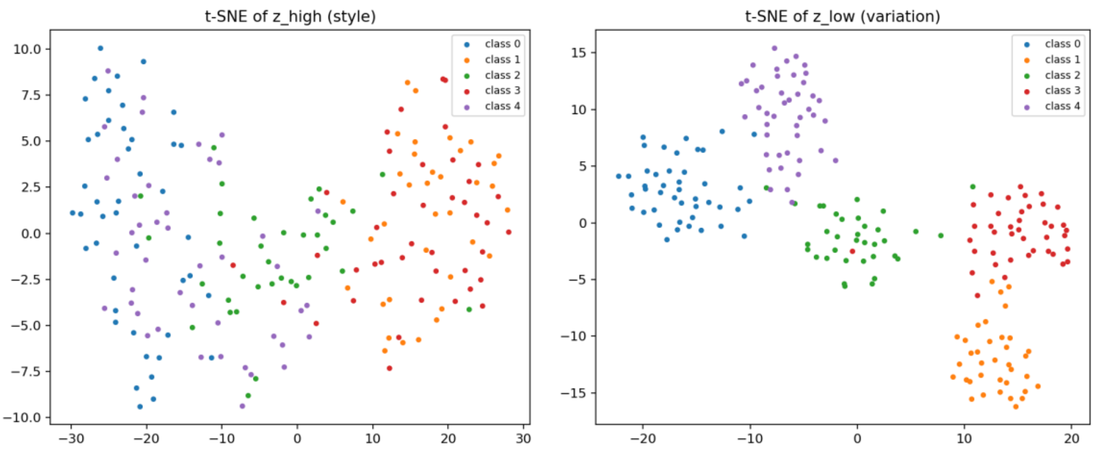
From the metrics alone, the generated patterns sound **musical and usable**, though somewhat conservative in variety.

From my personal point of view, the .wav files the model creates do sound familiar and referable to some pop music.

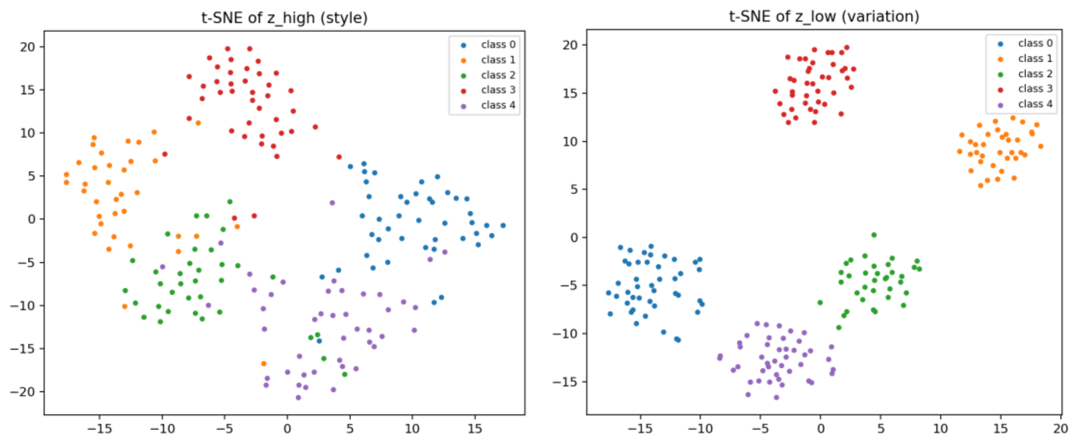
(D) Comparison of different annealing strategies



Constant β: Above are results of constant strategy. KL stays ~0 (no lift); `z_high` t-SNE is color-mixed with no clear clusters, indicates collapse/under-utilized latents.



Cyclical: Above are results of cyclical strategy. KL shows periodic spikes and non-zero plateaus; `z_high` clusters emerge, `z_low` is structured. This strategy avoids collapse with good hierarchy.



Linear: Above are results of linear strategy. KL ramps up smoothly to a steady non-zero level; `z_high` shows the clearest, tightest clusters; `z_low` well separated. Like cyclical strategy, also avoids collapse, with clean separation.

To conclude, Constant ≪ Linear ≈ Cyclical; cyclical gives a robust trade-off, linear shows the sharpest cluster separability, constant underperforms.

(E) Success rate of style transfer while preserving rhythm

Style-transfer evaluation criteria:

- **Style correctness:** Encode the transferred sample to **z_high** and run **1-NN (cosine)** against the validation gallery of z_high. If the predicted style equals the target style → pass.
- **Rhythm preservation:** Compare transferred vs source; both must hold:
  - **Kick Jaccard ≥ 0.80** (overlap of kick hits)
  - **Step-energy correlation ≥ 0.90** (correlation of per-step total activity)
- **Success:** Must satisfy **both** style correctness and rhythm preservation.

According to the **style_transfer_eval.json** generated in the **generated_pattern** folder,

- **Constant:** style **0%**, rhythm **0%**, both **0%**.
- **Cyclical:** style **0%**, rhythm **20%** (1/5), both **0%**.
- **Linear:** style **0%**, rhythm **20%** (1/5), both **0%**.

None of the runs hit “style + rhythm” simultaneously under the current thresholds. Cyclical / linear preserved rhythm on 1 case each, while constant preserved none.