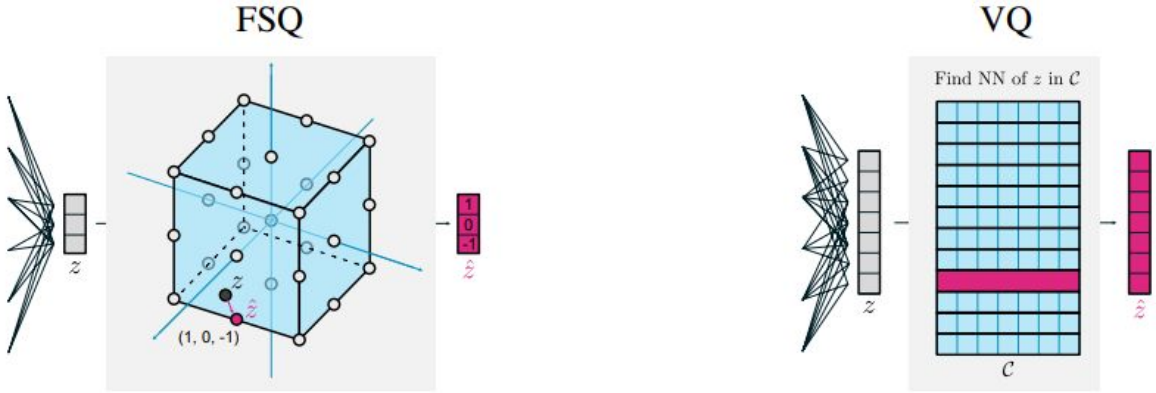
FSQ tokenizer study

Ronggian Qian,Binbin Dong,Reinhard Schwienhorst

Nov 4, 2026
TREASURE weekly

FSQ

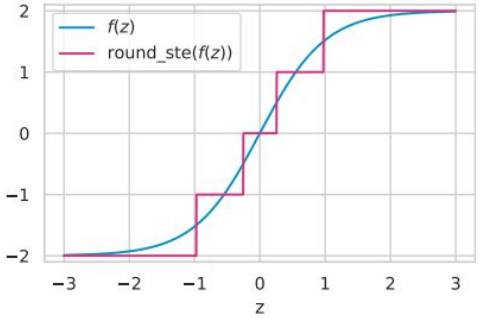


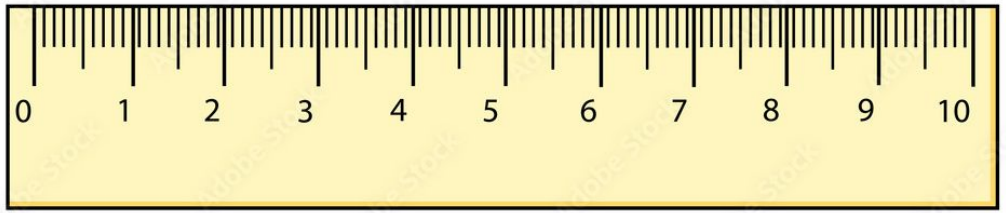
- From [FINITE SCALAR QUANTIZATION: VQ-VAE MADE SIMPLE]
  Designed to make a simple VQ by fixing code on the grid of a cubic codebook
- FSQ remove auxiliary losses from VQ-VAE and achieve high codebook utilization by design
- FSQ Cons: Limited expressiveness due to fixed implicit fixed codebook
- Each dimension is quantized independently while VQ shared a common codebook for latent variable z in multidimensional space

FSQ implementation

- One hyperparameter for FSQ: level, the prod(level) is the codebook size and len(level) is the latent variable z's dimension
  - E.g. [8,8,8,4], z is R^4, codebook size 2048 and each dimension has 8,8,8,4 code choices
- Use the same residual quantizer idea with VQ-VAE. Due to the fixed codebook design, each stage in FSQ is scaled by 1/L

	VQ	FSQ
Quantization	$\arg\min_{c \in \mathcal{C}} \|\|z - c\|\|$	$\text{round}(f(z))$
Gradients	STE	STE
Aux. Losses	Commitment, codebook, entropy loss	-
Tricks	EMA on codebook, codebook splitting projections, …	-
Parameters	Codebook	-





1st stage cm, 2rd stage mm

Training setup

Input variables

jet	pt,eta,phi, ntrk, QG_ntrk,QG_width,QG_C1, DL1d/GN2 for b/c/u jet
electron	pt,eta,phi,charge,LHMedium,LHTight,ptvarcone30,topoetcone20

Input statistics ATLAS opendata

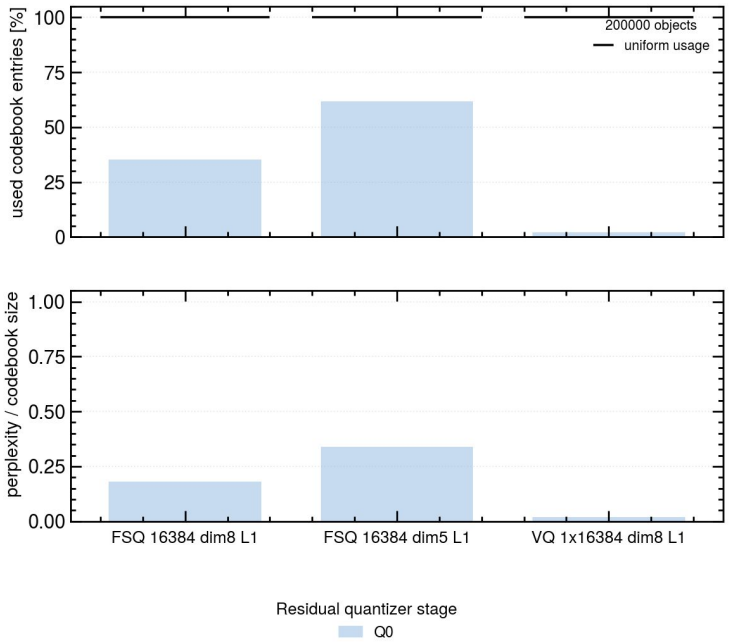
	Data	MC
event	43,853,327	16,648,176
jet	133,494,095	73,018,299
electron	3,627,650	5,732,654

Keep the other setup default: MLP encoder/decoder size, train/val/test split …

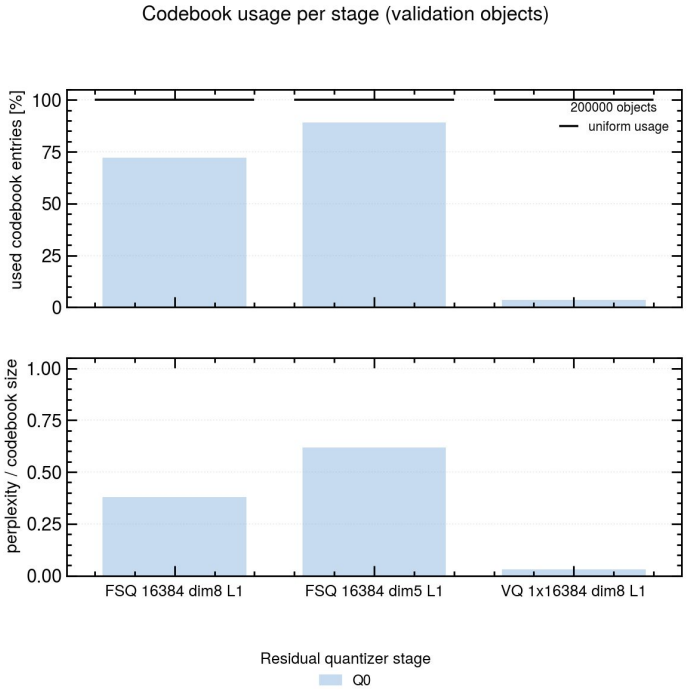
Q1 Training setup

- VQ-VAE locally trained only for MC samples using default setup
  - Q1-16384(14 bits)
  - FSQ comparison with the same size tokenizer
  - Levels Q1-16384: dim5 [8,8,8,8,4]
  - L1 reconstruction loss is used since the VQ-VAE also use L1, no other loss term compared with VQ-VAE
- Test with jet and electrons and MC only for R&D purpose

Codebook usage Q1



Electrons

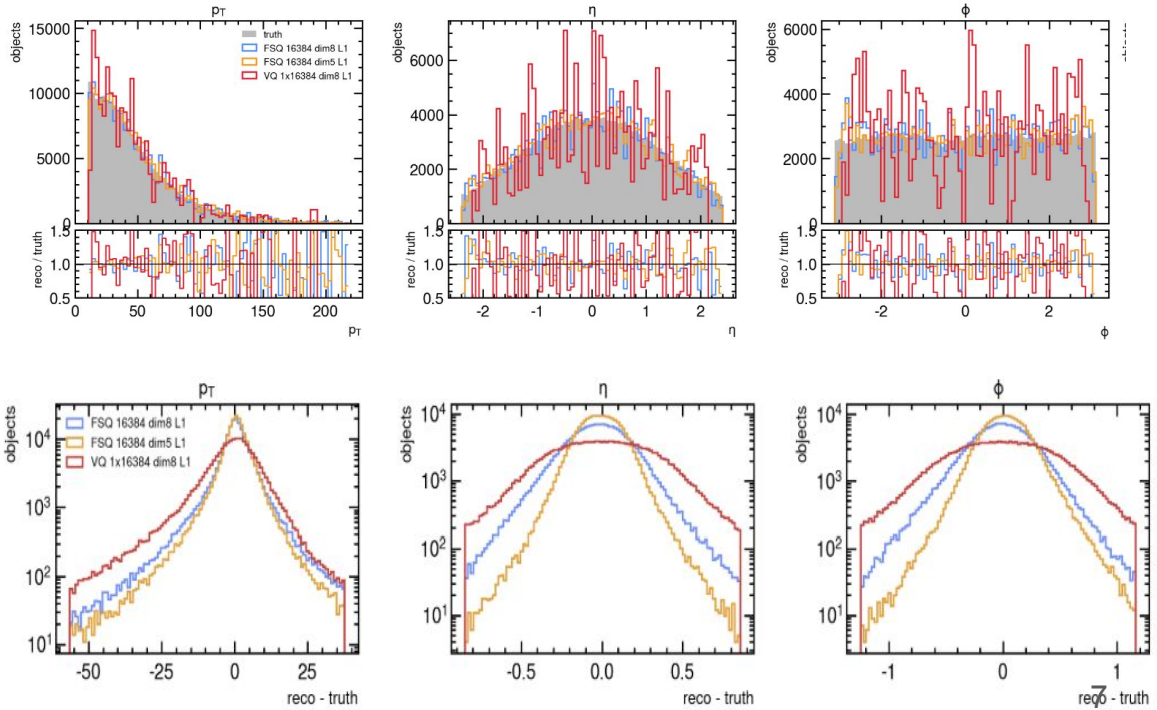


Jets

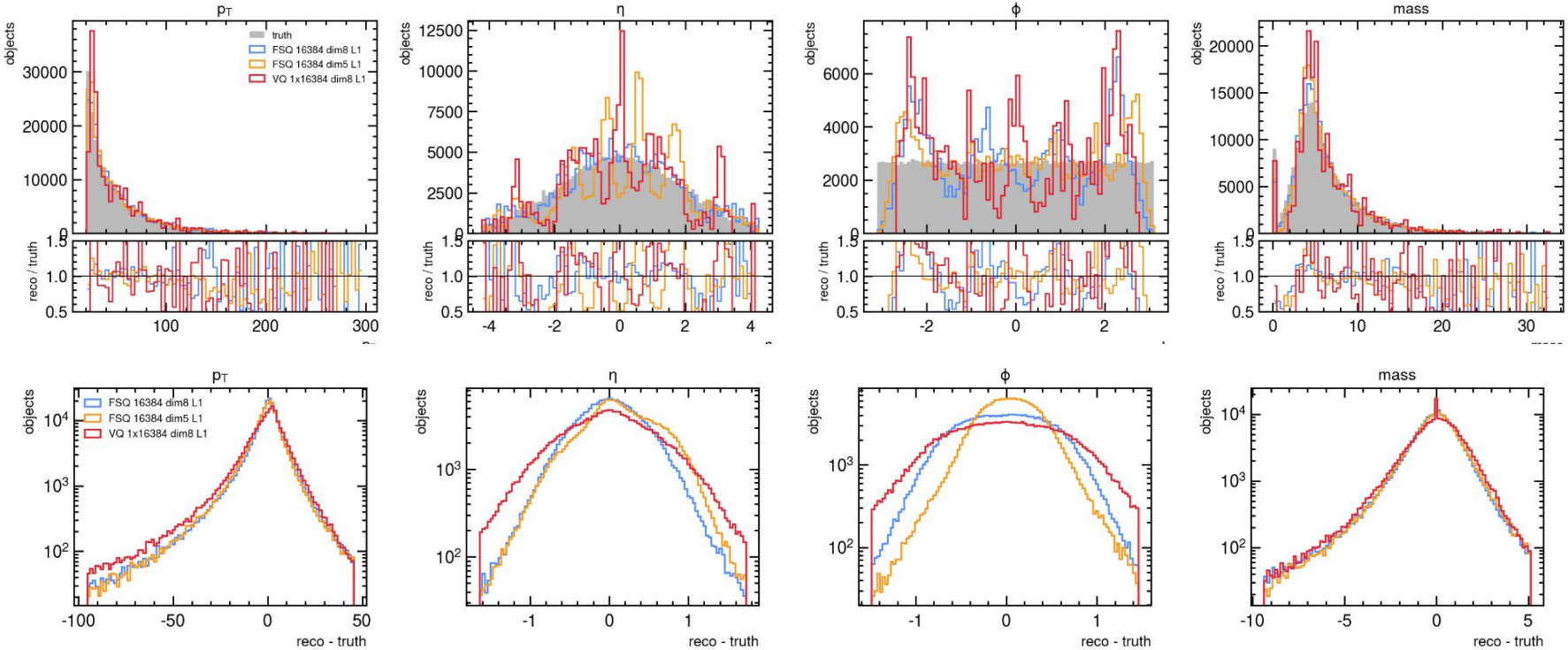
The codebook collapse problem is mitigated in FSQ

Reconstruction Q1 - Electrons

- Better performance observed from FSQ with high codebook usage



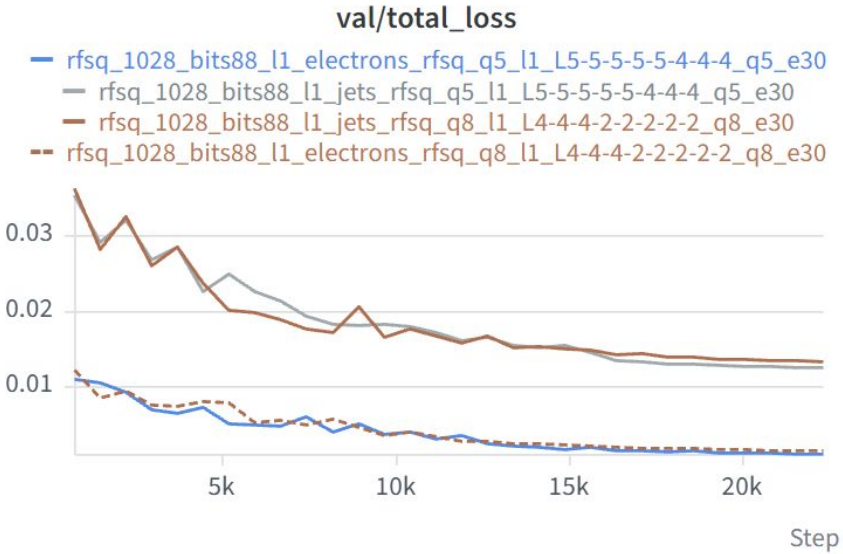
Reconstruction Q1 - Jets



- FSQ Slightly better than VQ-VAE

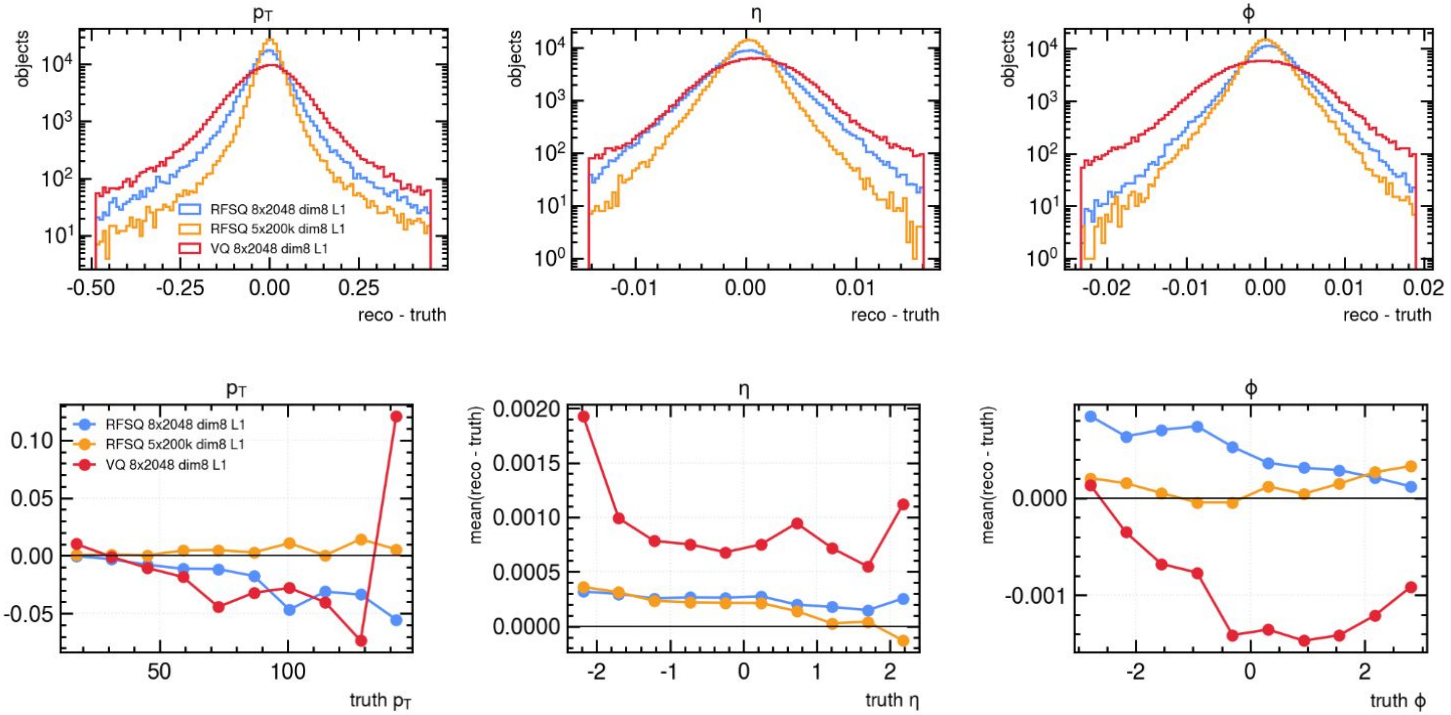
Q8 Training setup

- VQ-VAE locally trained only for using default setup
  - 8 quantizer with 2048 codebook for each: Q8-2048 (88 bits)
- FSQ comparison with the same size tokenizer
  - 88 bits Q8-2048 and 88 bits Q5-200k
- L1 reconstruction loss
- Full Data + MC samples

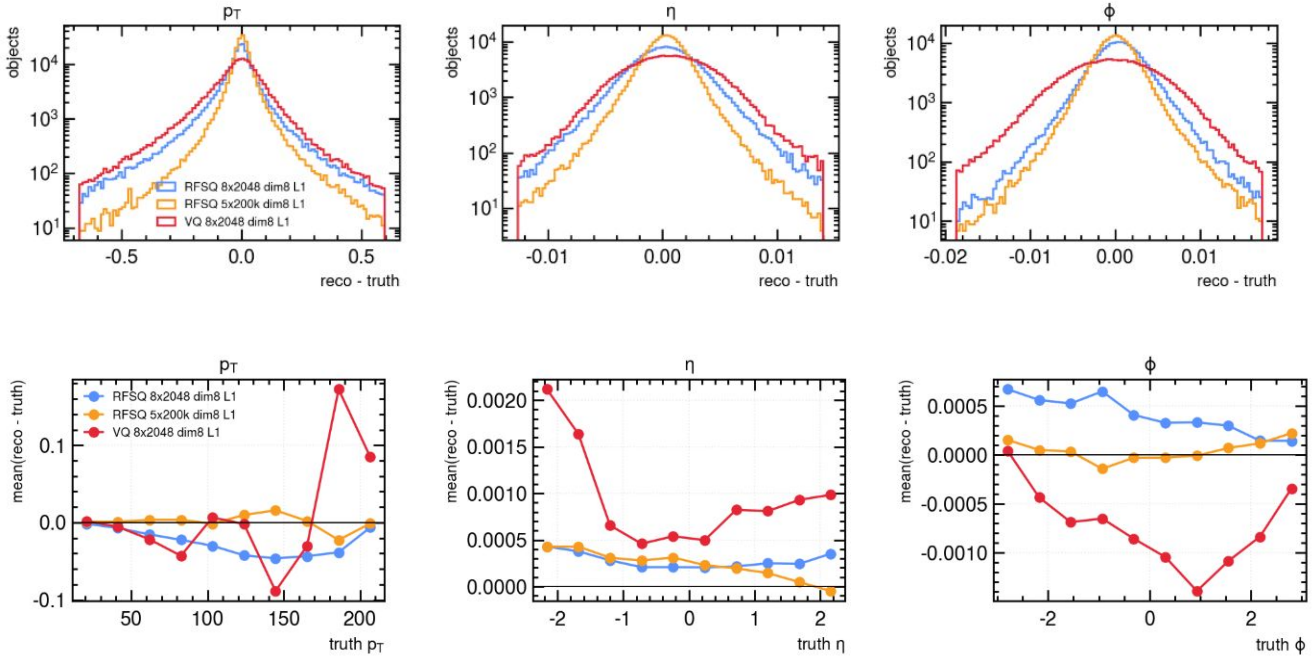


Reconstruction Q8 - Electrons - Data

Q5 FSQ has the best performance for electron reconstruction the Q8 RFSQ and VQ

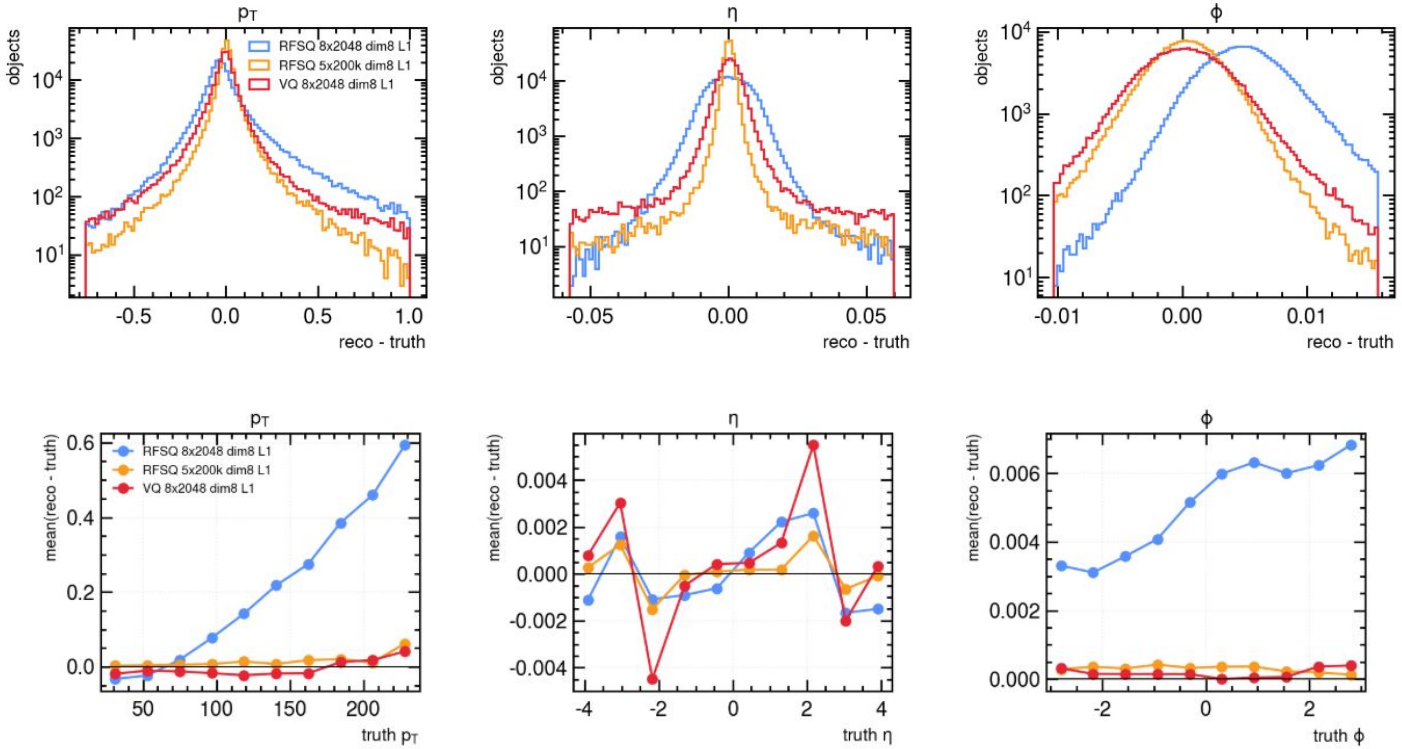


Reconstruction Q8 - Electrons - MC



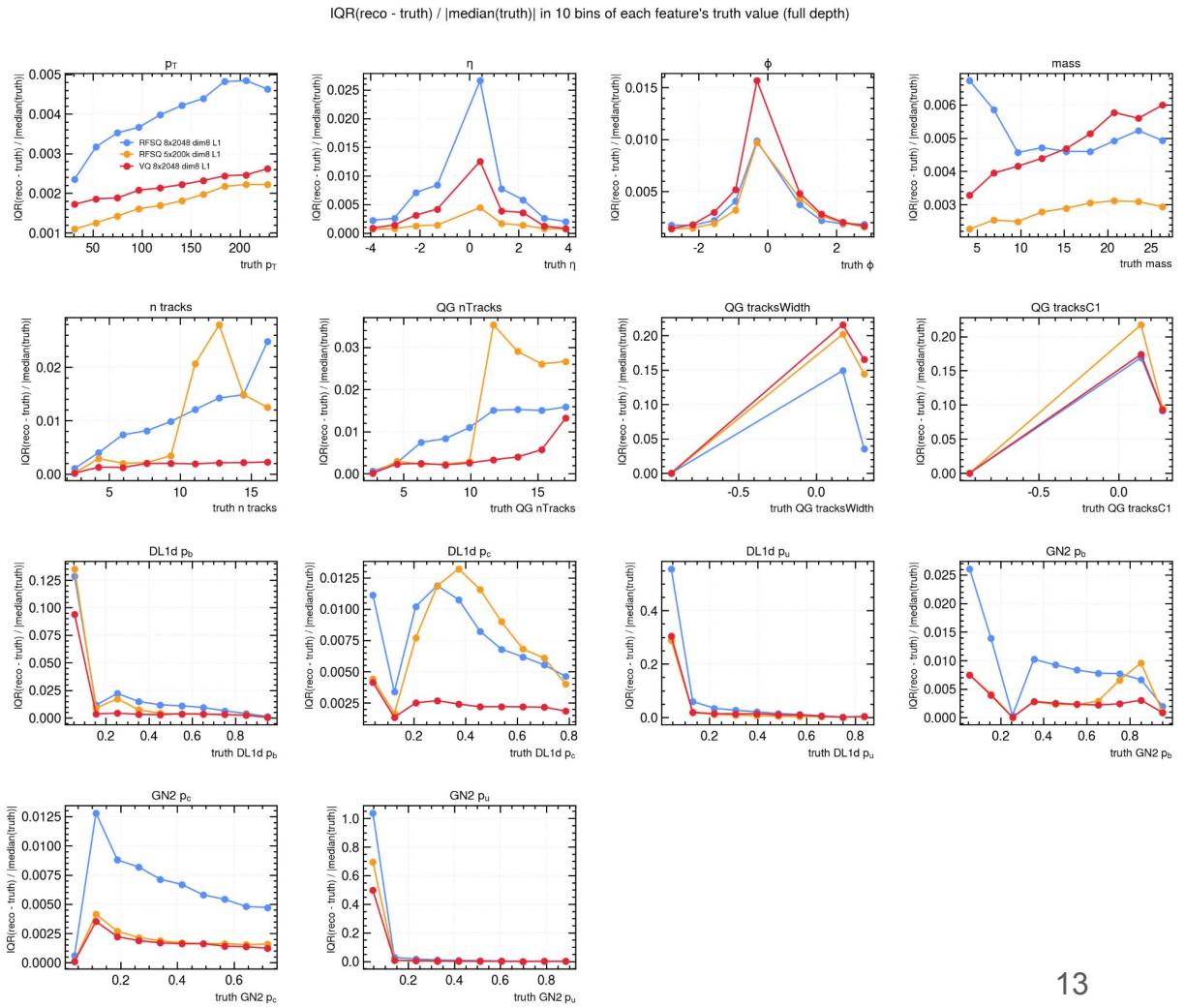
Reconstruction Q8 - Jets - Data

Similar performance between VQ-VAE and FSQ



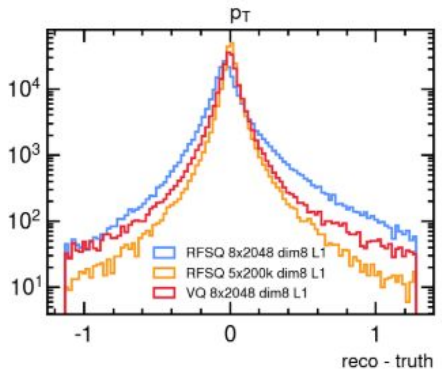
IQR

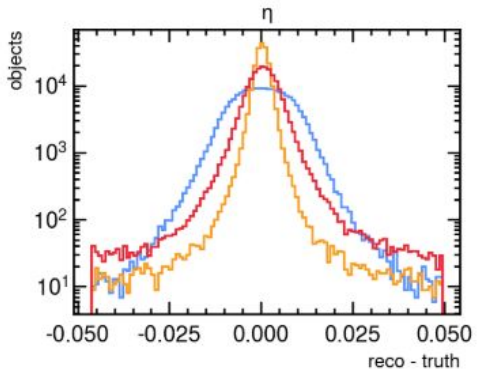
- VQ-VAE has better reconstruction on GN2 variables but worse in others

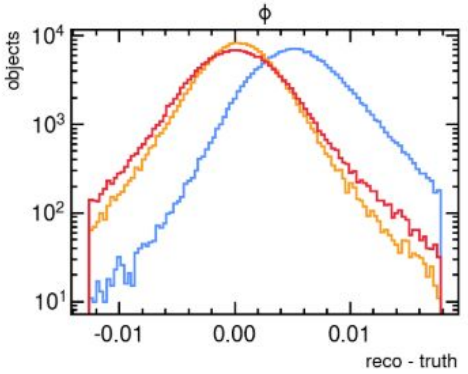


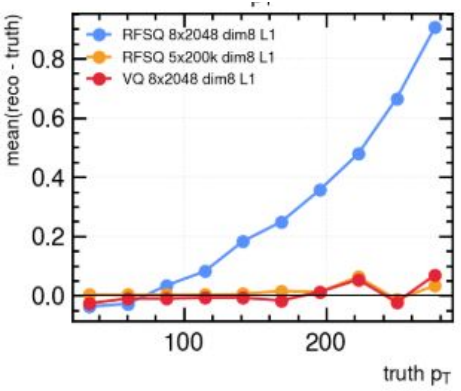
Reconstruction Q8 - Jets - MC

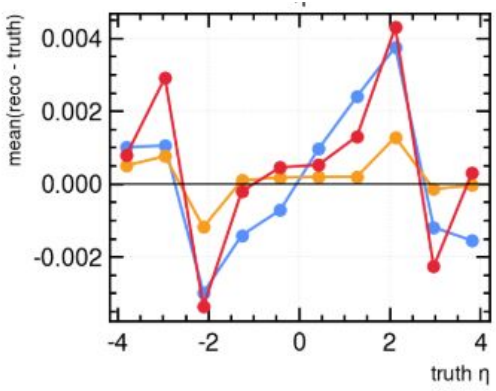
Similar performance between VQ-VAE and FSQ

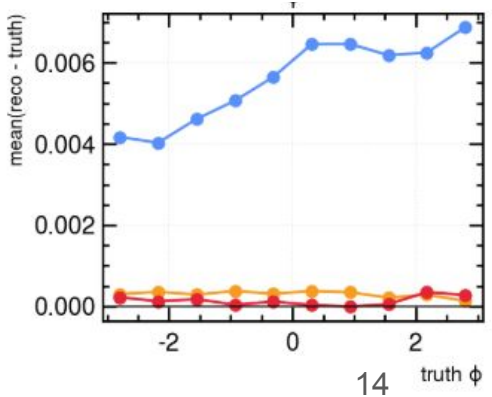




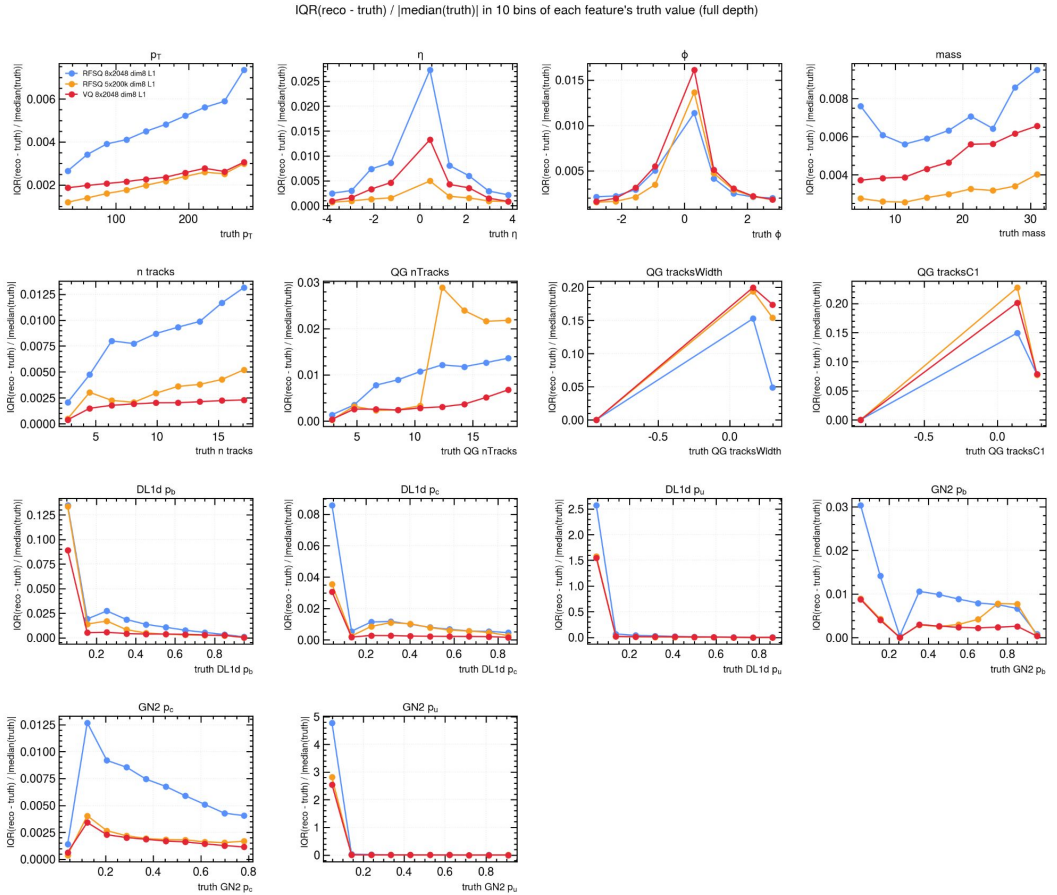








Reconstruction Q8 - Jets - MC



Similar performance between VQ-VAE and FSQ

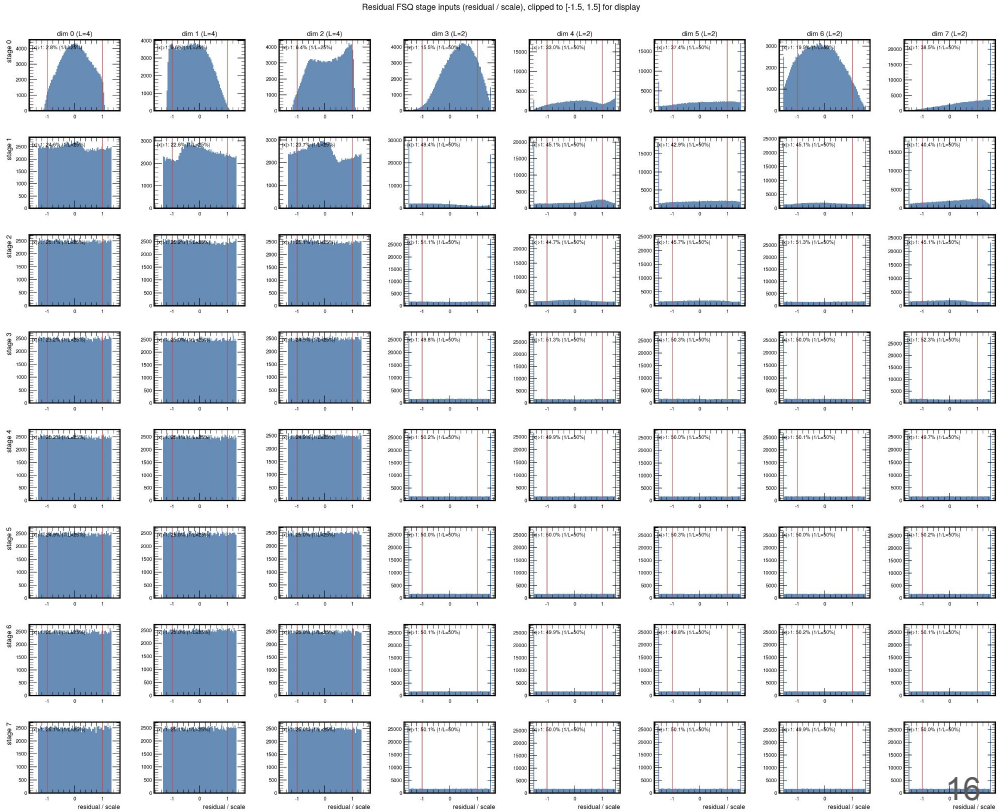
Fine tuning FSQ quantizer number

- Study why Q5 works better than Q8 for FSQ
- Shows latent z distribution for each dimension and for each stage
- The Q8 RFSQ shows few dead dimension for electrons
- Deep quantizer just learn from the noise

Electron

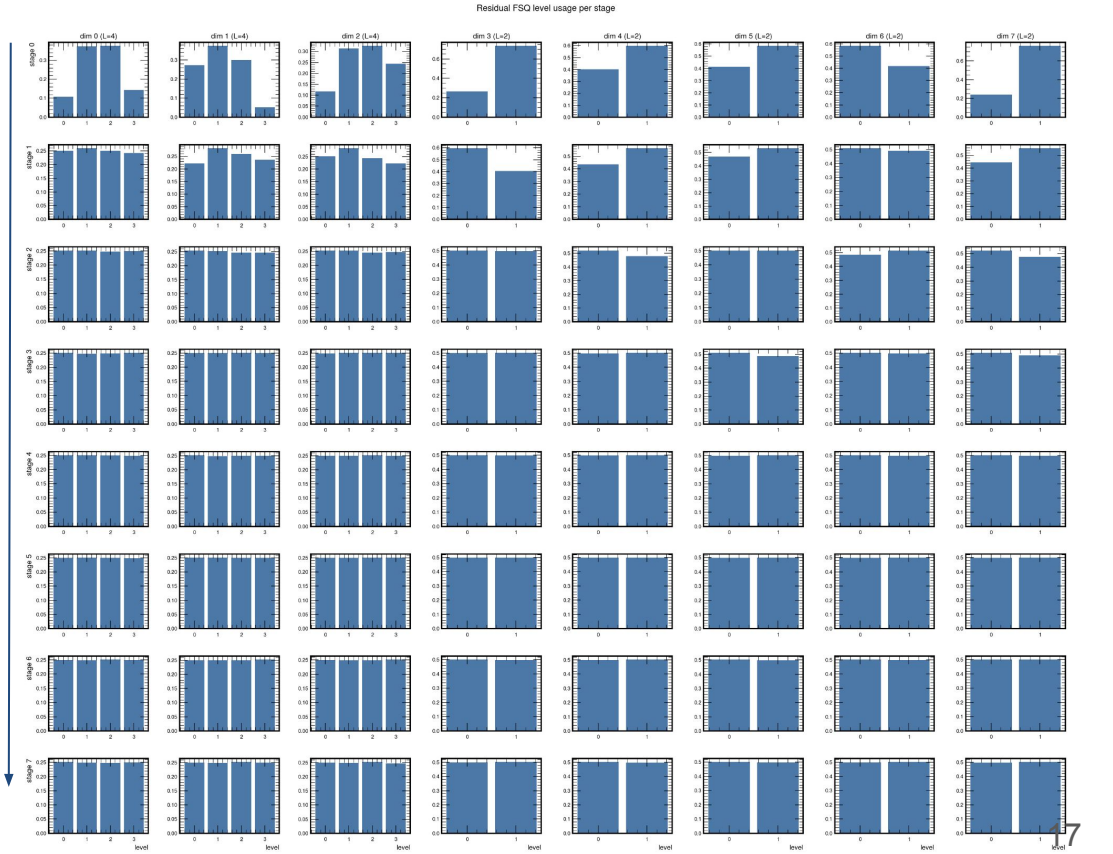
Quantizer stage

Dimension



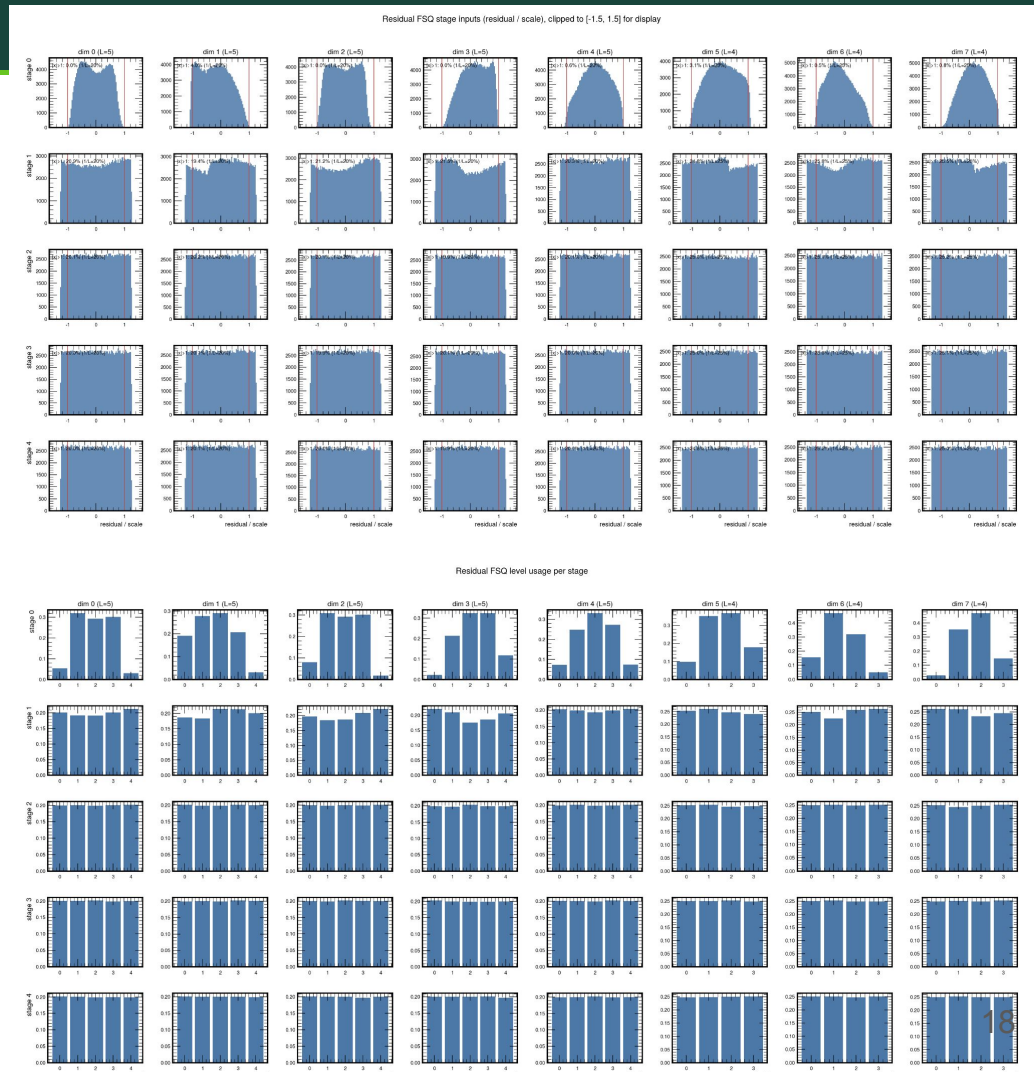
FSQ codebook usage

- Quantizing in FSQ is like filling binned histo for continuous distribution
- Limited codebook size limit the resolution



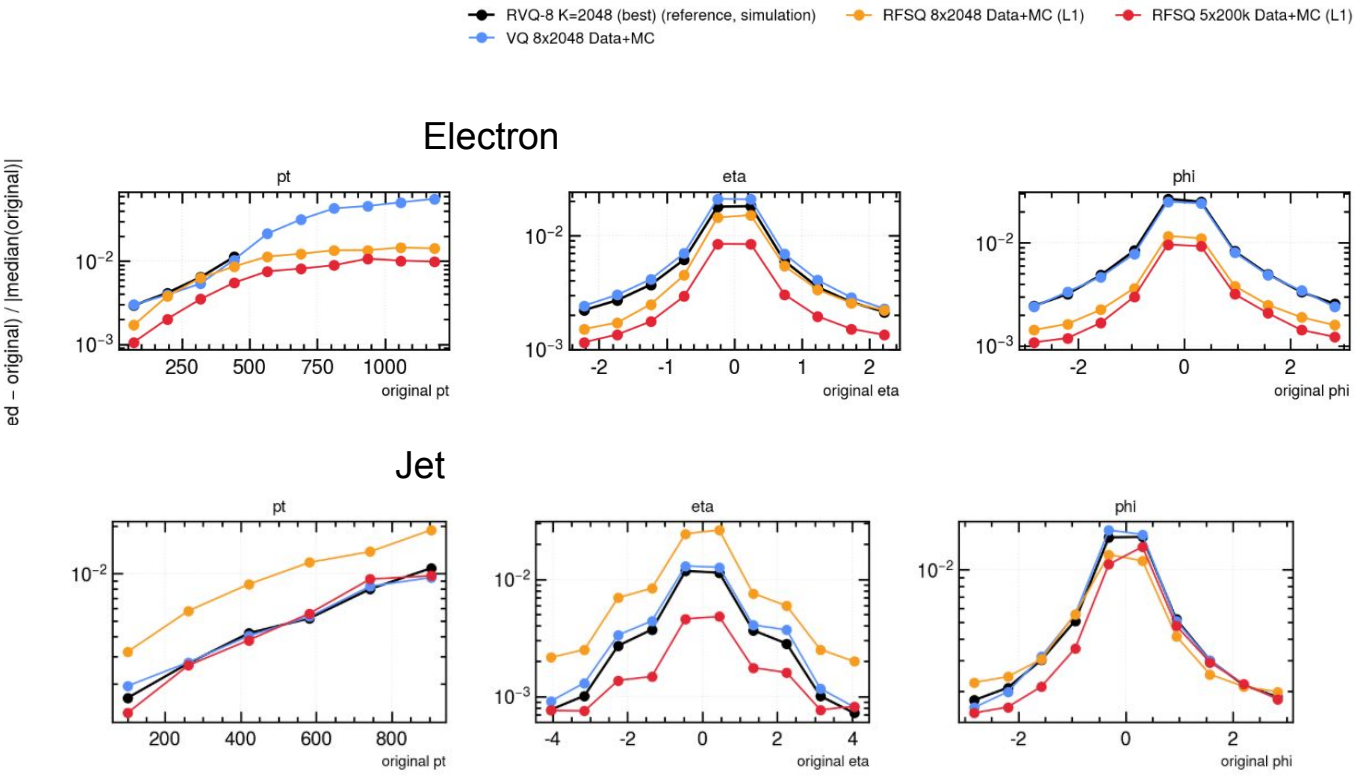
FSQ Fine tuning

- To solve those problems, less quantizer is used and each codebook size is enlarged



Baseline comparison

Directly comparison with Nazlim Q8 result in black shows the reproducibility of the current workflow



Conclusion

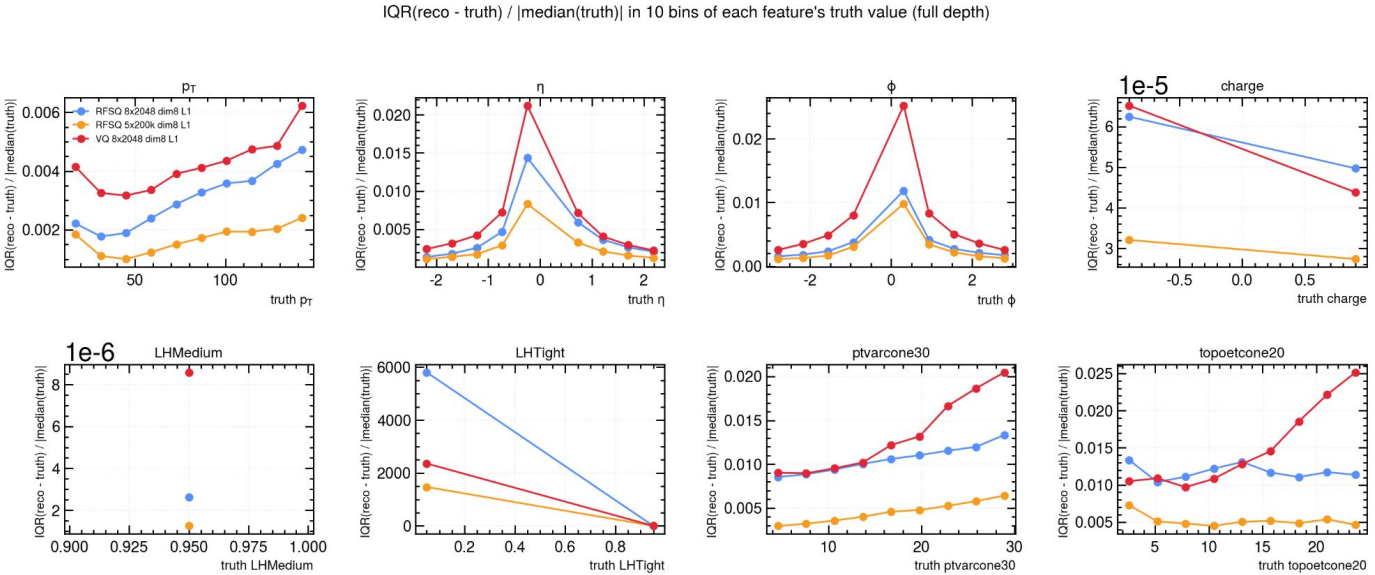
- FSQ with residual quantizer is implemented
- Q1 shows FSQ solve the codebook collapse problem for VQ-VAE
- Using large size token, Q8 FSQ shows similar performance with VQ-VAE
- Fine tuning is needed to avoid dead dimension and improve expressiveness

Next step

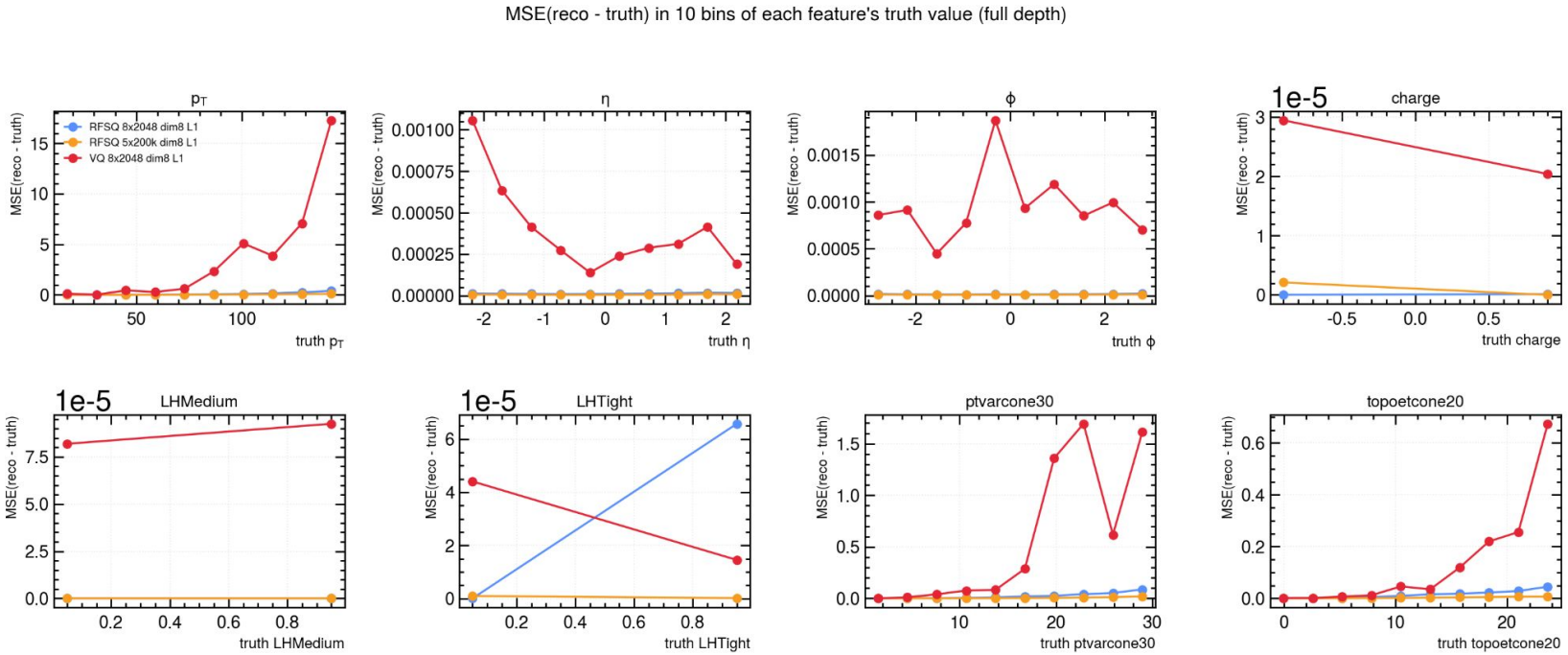
- Finish other objects training and comparison
- Try downstream tasks

Backup

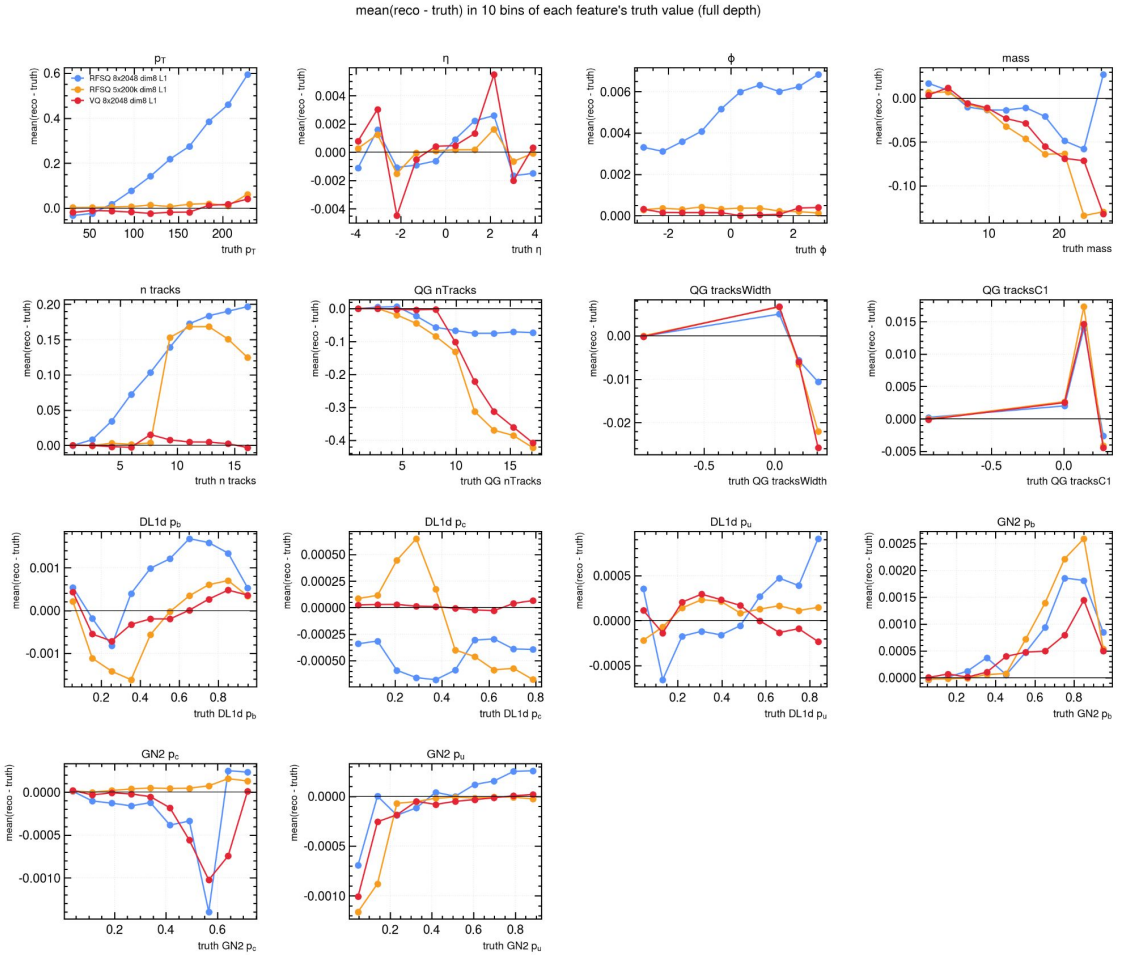
Reconstruction Q8 - Electrons - Data



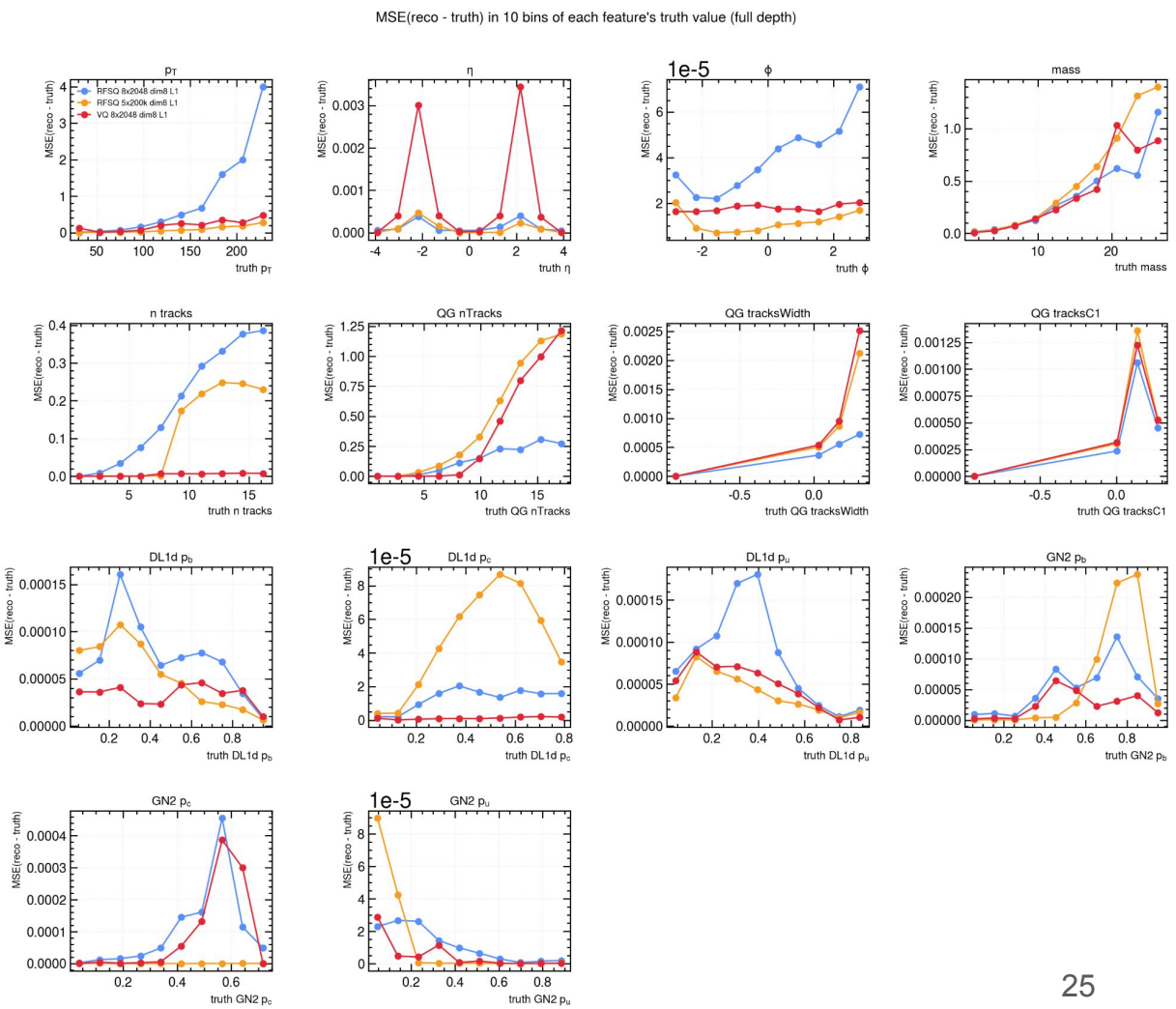
Reconstruction Q8 - Electrons - Data



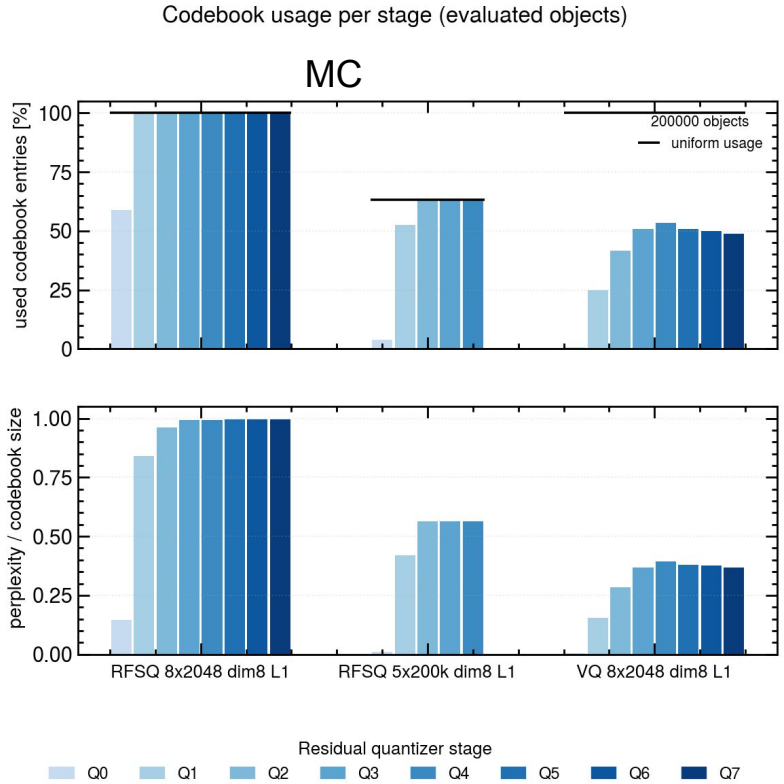
Reconstruction Q8 - Jets - Data

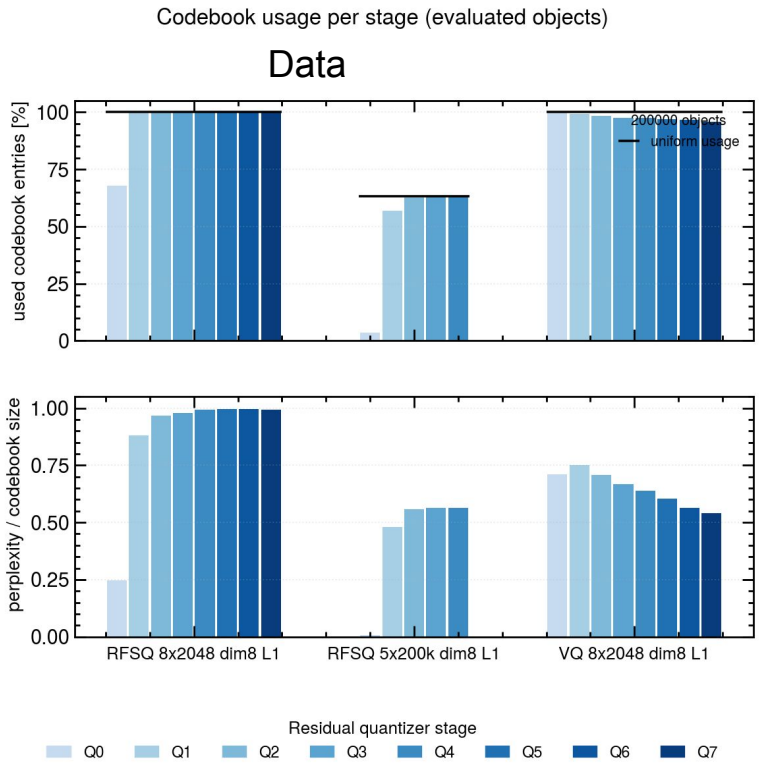


Reconstruction Q8 - Jets - Data

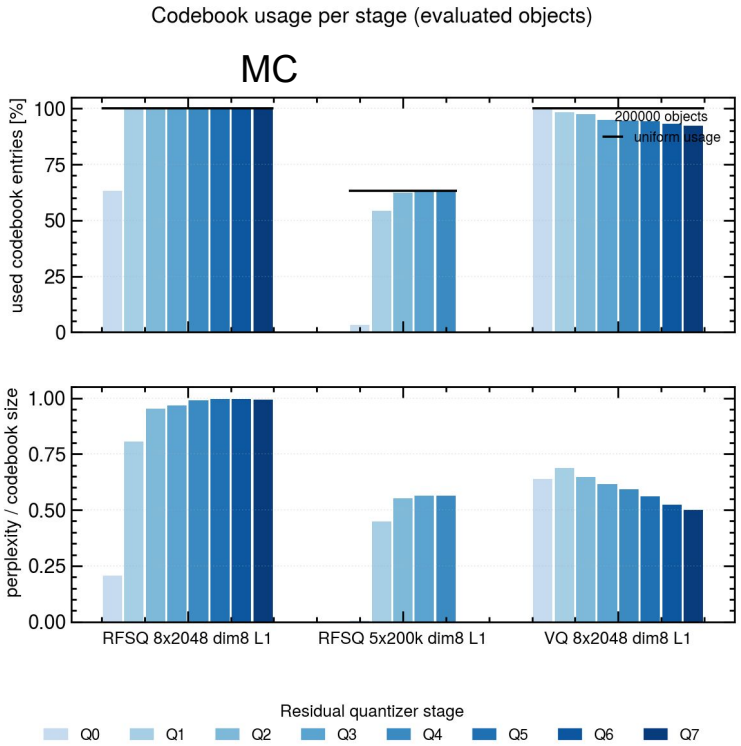


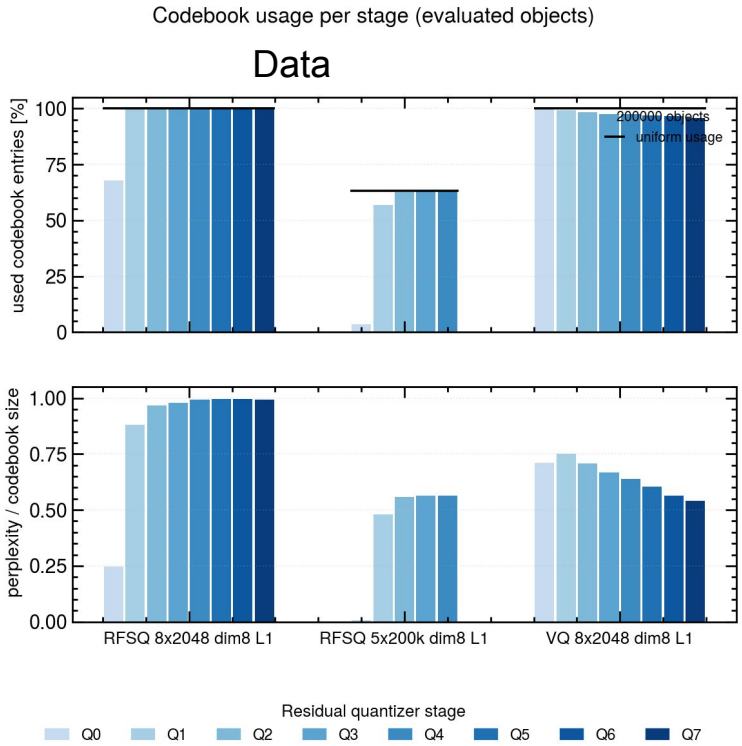
88 bit electron





88 bit Jet


Codebook usage per stage (evaluated objects)
MC
used codebook entries [%]
100
75
50
25
0
200000 objects
uniform usage
perplexity / codebook size
1.00
0.75
0.50
0.25
0.00
RFSQ 8x2048 dim8 L1
RFSQ 5x200k dim8 L1
VQ 8x2048 dim8 L1
Residual quantizer stage
Q0
Q1
Q2
Q3
Q4
Q5
Q6
Q7


Codebook usage per stage (evaluated objects)
Data
used codebook entries [%]
100
75
50
25
0
200000 objects
uniform usage
perplexity / codebook size
1.00
0.75
0.50
0.25
0.00
RFSQ 8x2048 dim8 L1
RFSQ 5x200k dim8 L1
VQ 8x2048 dim8 L1
Residual quantizer stage
Q0
Q1
Q2
Q3
Q4
Q5
Q6
Q7

IQR comparison

