Machine learning features of quiver moduli

Gianni Petrella - University of Luxembourg [1]

[1]Work supported by the Luxembourg National Research Fund (AFR-17953441)

UNIVERSITY OF LUXEMBOURG

Representations of quivers

A *quiver* Q is a finite directed multigraph with vertices Q_0 and arrows Q_1. A *representation* V of Q is a choice of vector space V_i per vertex i and a choice of linear application M_α for each arrow α [1].

The *dimension vector* of a representation V is

$$\mathrm{d} := \dim(V) := (\dim(V_i))_{i\in Q_0}.$$

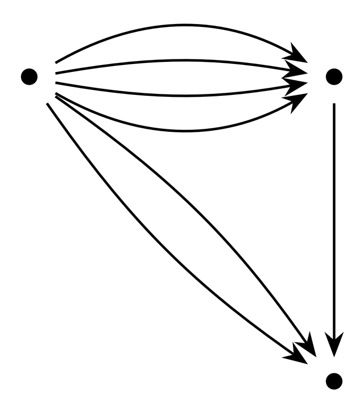

Figure 1. A quiver with three vertices.

A representation is, equivalently, a point in the *representation space*

$$\mathrm{R}(Q,\mathrm{d}) := {}_{\alpha\in Q_1}\mathrm{Mat}_{d_{t\alpha,s\alpha}}(\mathbb{C}) \qquad (1)$$

Two representations are *isomorphic* if they are equivalent up to a change of basis - that is, if they lie in the same orbit of the action of GL_d on $\mathrm{R}(Q,\boldsymbol{d})$.

If one fixes a *stability parameter* $\theta \in \mathbb{Z}^{Q_0}$ that satisfies the identity $\theta\cdot\mathrm{d}=0$, we say that a representation V is θ-stable, respectively θ-semistable, if every subrepresentation $W \subset V$ satisfies $\theta\cdot\dim(W) < 0$, respectively $\theta\cdot\dim(W)\leq 0$.

Moduli spaces of stable representations

The sets of θ-stable and semistable representations are GIT-stable opens in $\mathrm{R}(Q,\mathrm{d})$. We can thus consider their GIT quotients.

The relations between the affine, semistable and stable quotients are summarised in the following diagram:

$$\begin{array}{ccccc}
\mathrm{R}^{\theta-\mathrm{st}}(Q,\mathrm{d}) & \longrightarrow & \mathrm{R}^{\theta-\mathrm{sst}}(Q,\mathrm{d}) & \longleftarrow & \mathrm{R}(Q,\mathrm{d}) \\
\downarrow & & \downarrow & & \downarrow \\
\mathrm{R}^{\theta-\mathrm{st}}(Q,\mathrm{d})/\!/\mathrm{GL}_\mathrm{d} & \hookrightarrow & \mathrm{R}^{\theta-\mathrm{sst}}(Q,\mathrm{d})/\!/\mathrm{GL}_\mathrm{d} & \twoheadrightarrow & \mathrm{R}(Q,\mathrm{d})/\mathrm{GL}_\mathrm{d} \\
\| & & \| & & \| \\
\mathrm{M}^{\theta-\mathrm{st}}(Q,\mathrm{d}) & \longrightarrow & \mathrm{M}^{\theta-\mathrm{sst}}(Q,\mathrm{d}) & \twoheadrightarrow & \mathrm{M}^{-\mathrm{ssimp}}(Q,\mathrm{d}).
\end{array} \qquad (2)$$

It is known that horizontal inclusions are open and surjections are projective. Moreover, the moduli of stable representations is smooth, and $\mathcal{O}(\mathrm{M}^{-\mathrm{ssimp}}(Q,\boldsymbol{d}))$ is generated by traces of products over oriented cycles in Q [7]. If Q is acyclic, $\mathrm{M}^{-\mathrm{ssimp}}(Q,\boldsymbol{d})$ is thus a point and $\mathrm{M}^{\theta-\mathrm{sst}}(Q,\boldsymbol{d})$ is a projective variety.

The complement of $\mathrm{R}^{\theta-\mathrm{sst}}(Q,\mathrm{d})$ is the *unstable locus*. It admits a stratification into locally closed subsets, called *Harder–Narasimhan strata*.

Geometric invariants

Many geometric invariants of quiver moduli can be computed effectively using the software package `QuiverTools` [2, 3]. However, complexity is often a limiting factor, so for large scale applications we attempt to model several of these features using machine learning techniques.

- **Euler characteristic** is obtained via Betti numbers computations [8]. These are strong topological invariants of complex varieties, and appear in many different mathematical settings.
- **Teleman inequality ratios** are numbers between 0 and 1 defined on each Harder–Narasimhan stratum of $\mathrm{R}(Q,\mathrm{d})$. If all of them are strictly smaller than 1, *Teleman quantization* can be applied to show higher cohomology vanishings on $\mathrm{M}^{\theta-\mathrm{sst}}(Q,\mathrm{d})$. This is done in detail in [1].

Datasets

The training data is obtained by classical algorithms derived from the literature of quiver representations. These are implemented in `QuiverTools`, see [2] and the accompanying [3]. We harvested data using `QuiverTools.jl`, the Julia version of `QuiverTools`, and `Distributed.jl` to parallelize the computations across 128 cores, for a total CPU time of 1000h. For this work, we use a set of 50000 combinations of 4-vertex quivers and dimension vectors $0 \leq \mathrm{d} \leq (5,5,5,5)$, using the functions `betti_numbers()` and `teleman_weights()` from `QuiverTools.jl` to compute the ground truth values. For the Euler characteristic prediction, we ran experiments exclusively with dimension vector $(1,1,1,1)$. These are mathematically relevant because the resulting quiver moduli are toric varieties.

Methodology

We train feedforward neural networks to predict various numerical values associated to a quiver and a dimension vector. For all of our prediction problems we use multi-layer perceptron architectures, with one-dimensional layers. The input of our network is always the adjacency matrix of the quiver, flattened, joined to the dimension vector. The dataset of 50000 entries is always randomly split into training (80%) and testing data (20%), and batches are shuffled.

We use `Flux.jl` [5, 6], a pure Julia ML stack.

```
model = Chain(Dense(20 => 4096, relu),
  Dense(4096 => 4096, relu),
  Dense(4096 => 512, relu),
  Dense(512 => 1, sigmoid));

train_loader = DataLoader(
  (train_data, train_target), batchsize=1024);

for epoch in 0:1000
  for (x, y) in train_loader
    grad = gradient(m -> loss(m, x, y), model)
    Flux.update!(optimizer, model, grad[1])
  end
end
```

Figure 2. Model, data loader and training loop predicting Teleman ratios.

Current results are based on the following choices of hyperparameters. We use a decay scheme that halves the learning rate when the training loss plateaus for at least 30 epochs. Our models were let train for 1000 epochs.

Hyperparameter	Euler characteristic	Teleman ratio
Hidden layers	3	3
Layer sizes	4096, 4096, 1024	4096, 4096, 512
Activations	**relu**	**relu**, sigmoid
Batch size	1024	1024
Learning rate	10^{-4}	$5\cdot 10^{-5}$
Optimiser	Adam	Adam
Loss function	least squares	least squares

Table 1. Hyperparameters used in current setup.

Results

In all of our problems we were able to achieve high precision in predicting numerical invariants. Our model predicting Teleman ratios achieves an average loss of < 0.005 on training data and < 0.03 on test data.

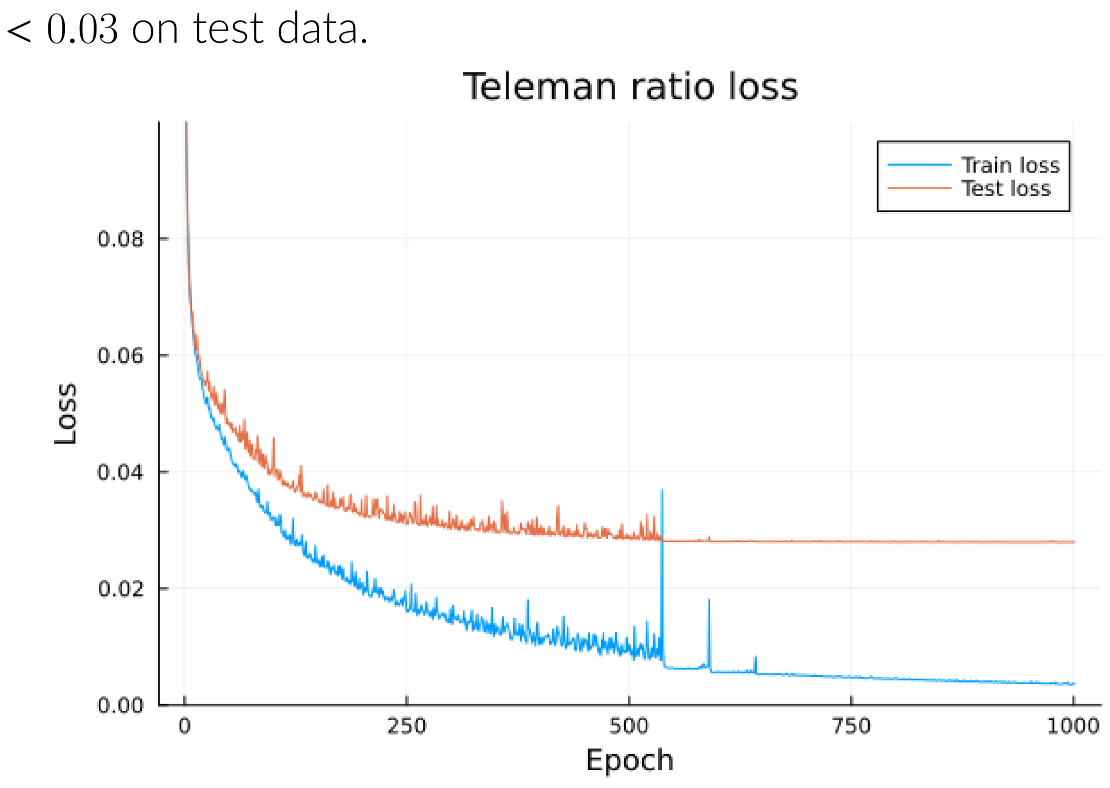

Figure 3. Loss of Teleman ratio predictor over training.

Our model predicting Euler characteristics scores $> 85\%$ on training data and 80% on test data, with both train and test losses being < 0.5.

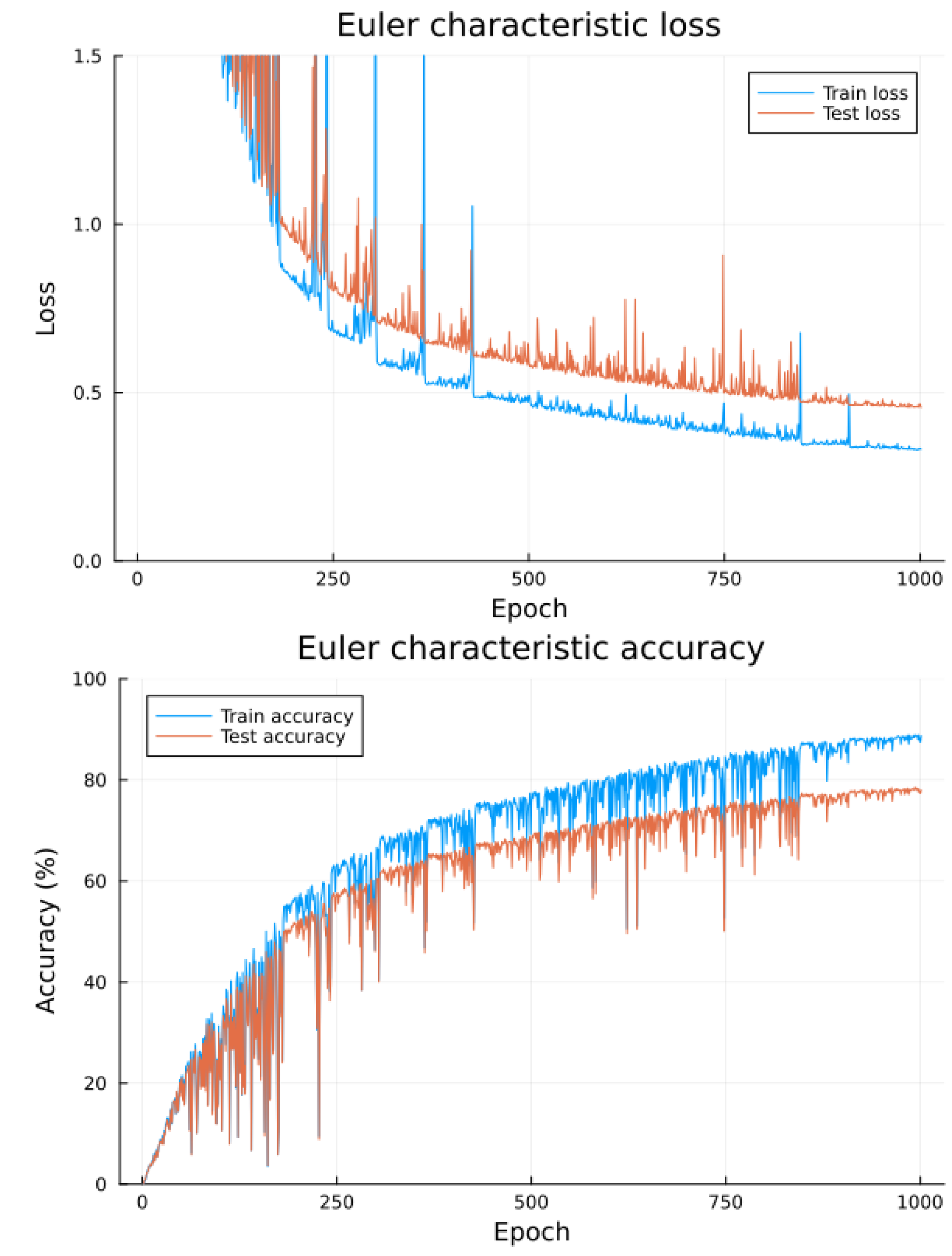

Figure 4. Loss and accuracy of Euler characteristic prediction over training.

Conclusions

Our work shows that relatively simple mathematical methods can reliably approximate invariants previously known to only be accessible via highly complex algorithms.

On one hand, this suggests that less complex algorithms may exist, and further research in this direction might be fruitful.

On the other hand, our results serve as evidence to the usefulness of ML-enabled workflows for mathematical research: as we have proved that our features are learnable, we plan to use more sophisticated workflows for generating mathematically meaningful examples, e.g. using transformer networks via PatternBoost [4].

References

[1] Pieter Belmans, Ana-Maria Brecan, Hans Franzen, Gianni Petrella, and Markus Reineke. *Rigidity and Schofield's partial tilting conjecture for quiver moduli*. 2023. arXiv: 2311.17003v2 [math.AG].

[2] Pieter Belmans, Hans Franzen, and Gianni Petrella. *QuiverTools*. DOI: 10.5281/zenodo.1268022. URL: https://quiver.tools.

[3] Pieter Belmans, Hans Franzen, and Gianni Petrella. *The QuiverTools package for SageMath and Julia*. 2025. arXiv: 2506.19432 [math.AG].

[4] François Charton, Jordan S. Ellenberg, Adam Zsolt Wagner, and Geordie Williamson. *PatternBoost: Constructions in Mathematics with a Little Help from AI*. 2024. arXiv: 2411.00566 [math.CO].

[5] Michael Innes, Elliot Saba, Keno Fischer, Dhairya Gandhi, Marco Concetto Rudilosso, Neethu Mariya Joy, Tejan Karmali, Avik Pal, and Viral Shah. *Fashionable Modelling with Flux*. 2018. arXiv: 1811.01457 [cs.PL].

[6] Mike Innes. "Flux: Elegant Machine Learning with Julia". In: *Journal of Open Source Software* (2018). DOI: 10.21105/joss.00602.

[7] Lieven Le Bruyn and Claudio Procesi. "Semisimple representations of quivers". In: *Trans. Amer. Math. Soc.* 317.2 (1990), pp. 585–598. DOI: 10.2307/2001477.

[8] Markus Reineke. "The Harder-Narasimhan system in quantum groups and cohomology of quiver moduli". In: *Invent. Math.* 152.2 (2003), pp. 349–368. DOI: 10.1007/s00222-002-0273-4.