A machine learning algorithm for the nonlinear Fokker-Planck-Landau collision operator in XGC

Marco Andrés Miller, Michael Churchill

Background

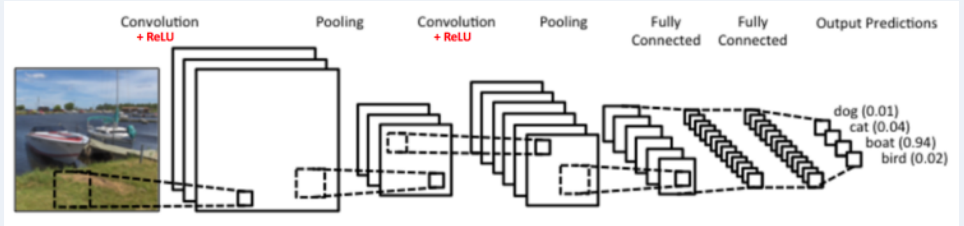
- XGC1 is a gyrokinetic particle-in-cell code solving Boltzmann equation
- Includes 2D (velocity space) solver of nonlinear Fokker-Planck-Landau collision operator

$$\frac{df_a}{dt} = \sum_b C_{ab}(f_a; f_b') = -\sum_b \frac{e_a^2 e_b^2 \ln\Lambda}{8\pi\epsilon_0^2 m_a}\frac{\partial}{\partial \boldsymbol{v}} \cdot \int \boldsymbol{U} \cdot \left(\frac{f_a}{m_b}\frac{\partial f_b'}{\partial \boldsymbol{v}'} - \frac{f_b'}{m_a}\frac{\partial f_a}{\partial \boldsymbol{v}'}\right) d^3v'$$

- Current runtime for operator scales as O(n^2), where n is number of species
- Propose use of machine learning to learn nonlinear transformation of operator
  - Investigate training deep convolutional neural network on XGC distribution function data

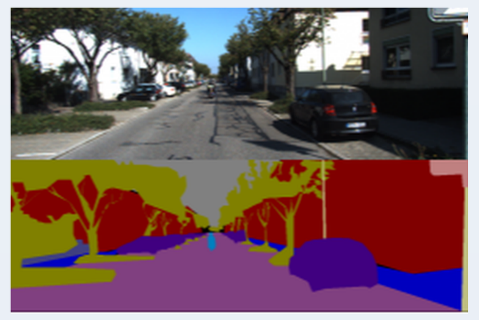
Machine Learning for Computer Vision

Image Recognition



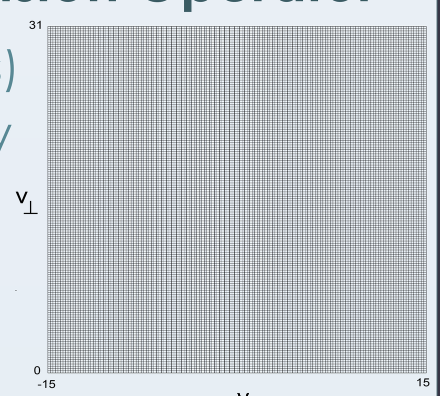
- Input image: H x W x 3 (3 color channels)
- Network ends with fully-connected layers → C-dim vector (C scores predicted)

Semantic Segmentation



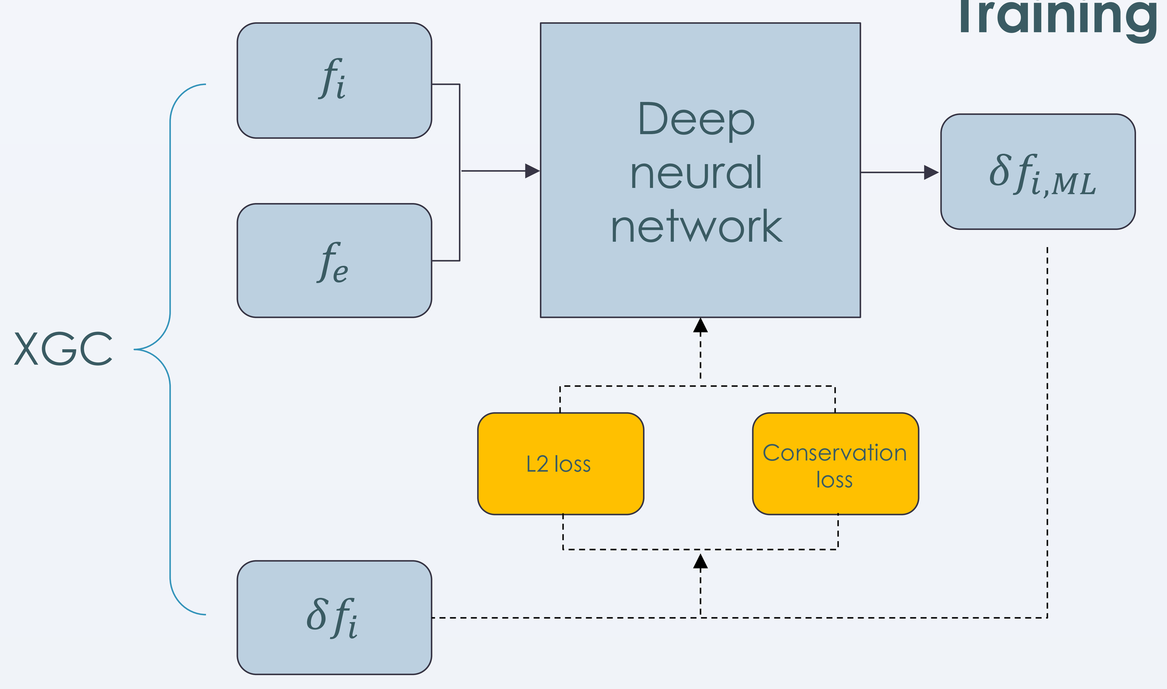
- H x W x 3 → H x W x C
  - Upsample + 2D fully connected layers
- Score each pixel and classify

Machine Learning for Collision Operator



- Input f: 32 x 31 x n (n species) → output df: 32 x 31 x 1 (only one species for now)
- Preserve spatial structure of input to make decision at "pixel" level
- Turn binary classification (yes/no) into regression (how far off) → no fully connected layers at end
- Loss function now composed two comparisons between output and target df:
  1. Overall L2 loss
  2. Conservation of density/momentum/energy losses

Training Schematic



$$L_{L2} = \frac{1}{n}\sum_{i=1}^{n}(\delta f_{XGC} - \delta f_{ML})^2 \qquad L_{cons} = \sum_{i=1}^{3}\lambda_i(x^i\delta f_{XGC} - x^i\delta f_{ML})$$

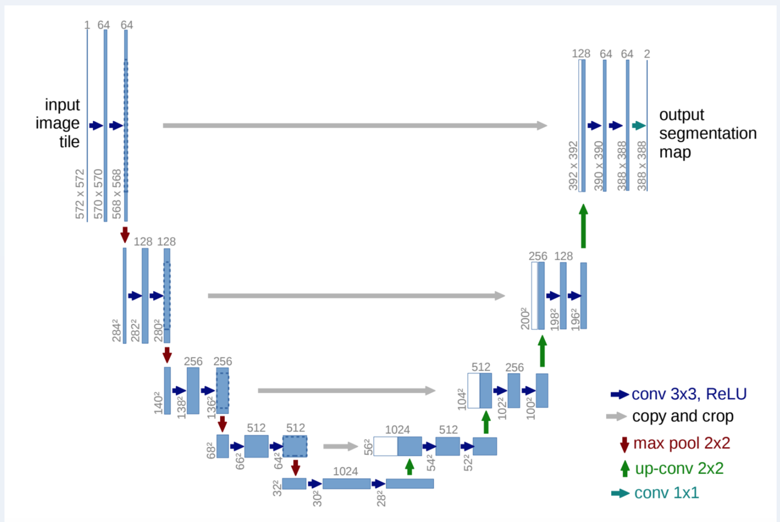
- Temporal discretization of derivative in XGC gives df

$$\delta f_{i,col} = dt \cdot C_i(f_i, f_e) = dt \cdot (C_{ii}(f_i, f_i) + C_{ie}(f_i, f_e))$$

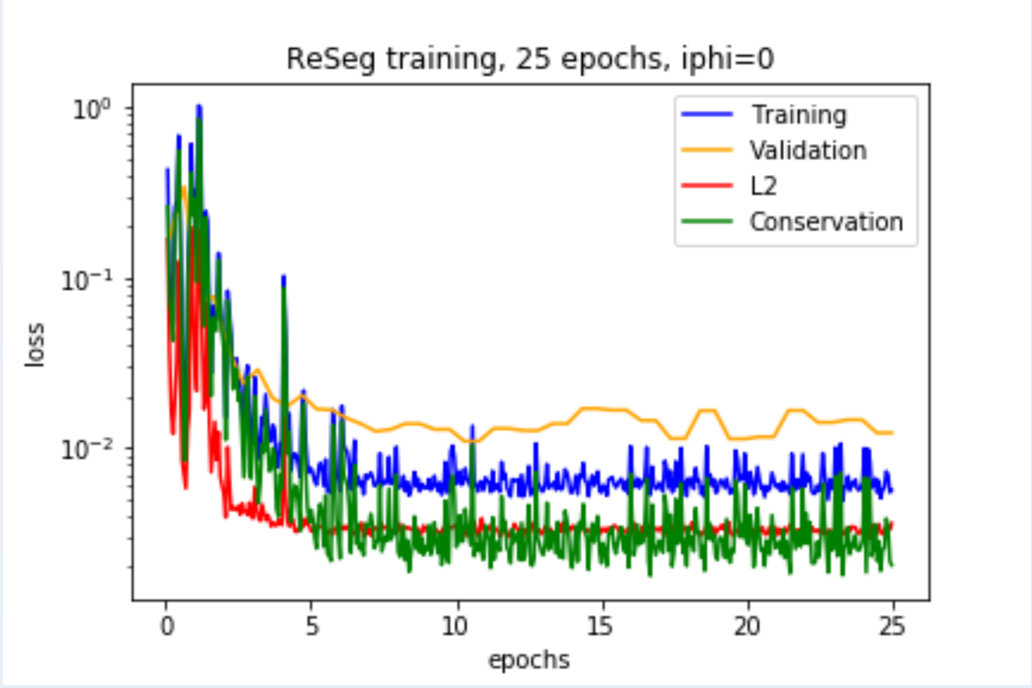
- Initial f and change in f (df) obtained from XGC collision kernel
- Phase-space volume of velocity bins and temperature also extracted from XGC
  - Can compare actual vs. predicted density/momentum/energy and compute conservation loss (as well as L2 loss)

NN Architecture

- U-Net (pictured) – one of fundamental architectures for semantic segmentation techniques - can add residual connections for better learning
- Downsampling blocks: composed of convolutions followed by max-pooling layers – interspersed with ReLU activations
- Upsampling blocks: composed of convolutions followed by up-convolution layers - also ReLU activations
- End with 1x1 convolutions to decrease depth to 1
- Skip connections preserve "original resolution" for learning



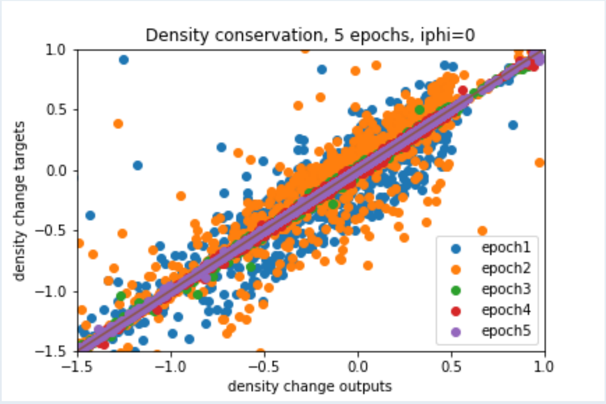
Training and Validation Results

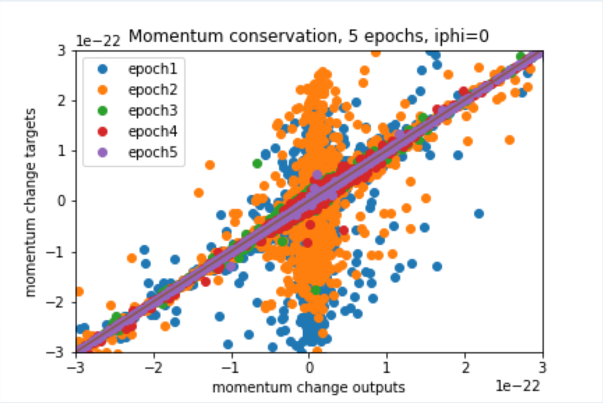


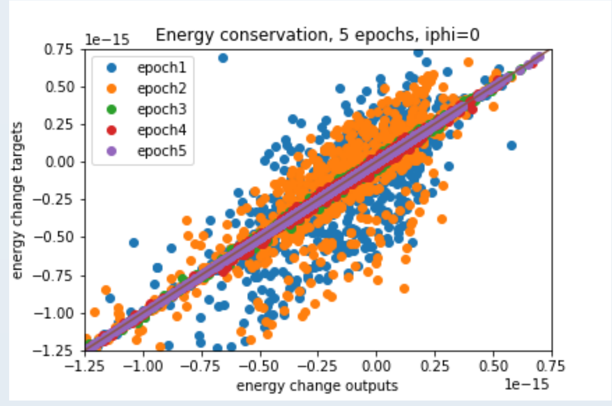
- Initial training done for JET XGC1 data – ti272_JET_heat_load
- Adam optimization algorithm – converges faster
  - Combines "momentum" and "RMS Prop"
- 25 total epochs
  - L2 and conservation loss both decrease
- Validation done every 2000 iterations (~twice per epoch)
- Final weights used for testing

Preliminary Testing Results

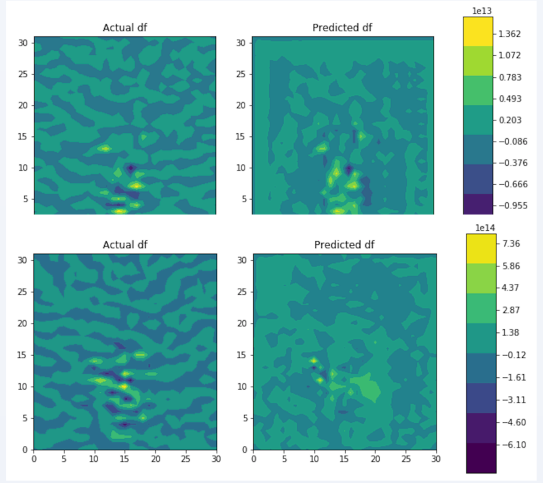
- Maximum L2 error out of (16668 testing points): 0.0389
  - Percentage of points with error below:
    - 1e-1: all
    - 1e-2: 13248
    - 1e-3: 160
- Maximum conservation error:
  - Density: 1.976e-02 (mean: 1.610e-04)
  - Momentum: 1.261e-01 (mean: 1.217e-03)
  - Energy: 1.040e-03 (mean: 1.977e-05)







δf Prediction



$\psi_n = 0.5$

$\psi_n = 0.98$

XGC Reimplementation (future)

- Export PyTorch modules to C++
- Can be implemented as option, with current Picard iteration scheme as backup
- Can use both algorithms in tandem
  - ML provides initial guess, Picard iteration completes calculation

Further Work

- Test new NN architectures – add residual connections – improves learning
- Investigate broader collisionality regime
  - Physics of collision depend on collisionality of species – may need new NN for different regimes
  - Perhaps one NN still sufficient with broad enough range of training data
- Investigate validity of NN between species
  - Same NN for electrons and ions?
  - Does mass difference imply different enough collision mechanism such that one NN cannot be trained on both (even with one training set for both species)
  - Multiple ion species?
  - Even larger mass difference – also charge difference

References

Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab, N., Hornegger, J., Wells, William M., Frangi, Alejandro F. (eds.) MICCAI 2015. LNCS, vol. 9351. Springer, Cham (2015).

Visin, F., Ciccone, M., Romero, A., Kastner, K., Cho, K., Bengio, Y., Matteucci, M., Courville, A.: ReSeg: A recurrent neural network-based model for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. (2016)

Yoon, E.S.; Chang, C.S. A Fokker-Planck-Landau collision equation solver on two-dimensional velocity grid and its application to particle-in-cell simulation. Phys. Plasmas 2014, 21, 032503.

Acknowledgements

This work was made possible by funding from the Department of Energy for the Summer Undergraduate Laboratory Internship (SULI) program. This work is supported by the US DOE Contract No.DE-AC02- 09CH11466.