Pattern Recognition & Machine Learning Techniques for Automated Classification of Signals in Swarm Time Series

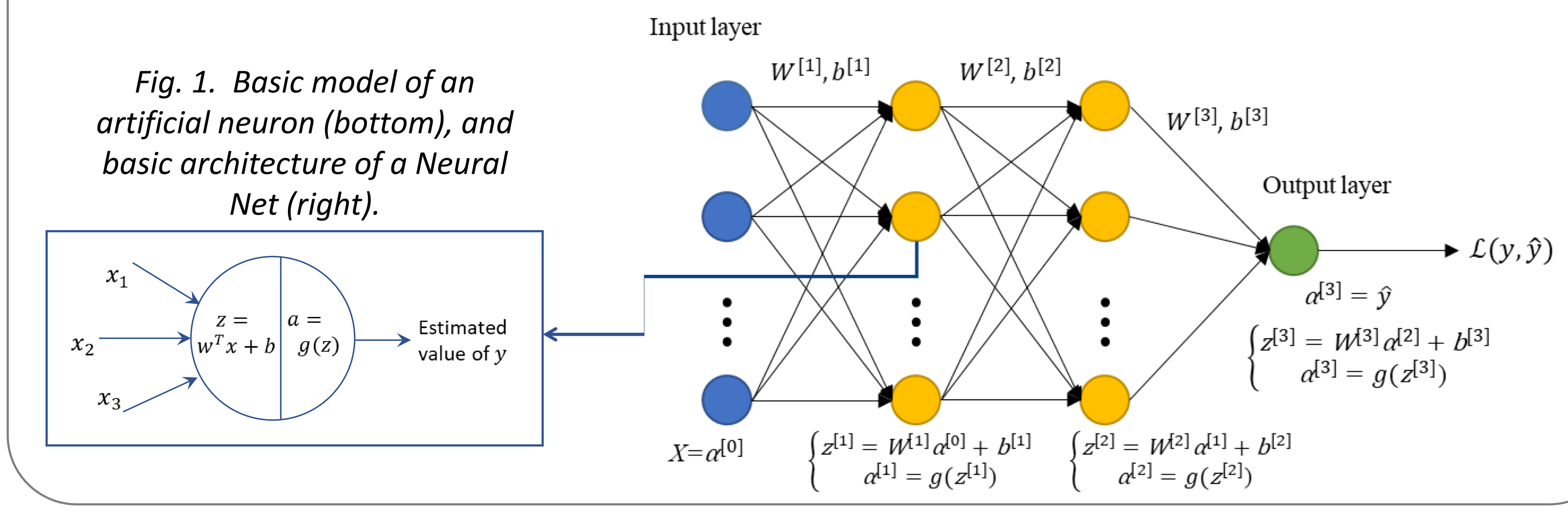
A. Antonopoulou[1,2], G. Balasis[1], C. Papadimitriou[1,2], A. Z. Boutsi[1,2], O. Giannakis[1], I. A. Daglis[1,2,3]

1. Institute for Astronomy, Astrophysics, Space Applications and Remote Sensing, National Observatory of Athens, Greece;
2. Department of Physics, National and Kapodistrian University of Athens, Greece; 3. Hellenic Space Center, Greece

Neural Nets & main steps of algorithm

- Composition of l parametric functions referred to as *layers*.
- Each layer consists of a different number of units (*neurons*) with trainable weights and biases (w, b).
- All neurons are *fully connected* to the neurons in previous and post layers.
  1) Define the model architecture
  2) Initialize the parameters $W^{[0]}, b^{[0]}$
  3) Forward prop (left-to-right pass)
     - Compute the loss function $\mathcal{L}(y, \hat{y})$ where $\hat{y}$ the predicted value and y the ground truth
     - The Cost is then computed by summing over all training examples, $J = \frac{1}{m}\sum_{i=1}^{m}\mathcal{L}(y, \hat{y})$
  4) Backward prop (right-to-left pass)
     - Compute the derivatives of $\mathcal{L}$ wrt each variable
     - Based on Calculus chain rule, e.g. $\frac{d\mathcal{L}}{dz} = \frac{d\mathcal{L}}{da}\frac{da}{dz}$
  5) Optimization method (e.g. Gradient Descent): $\min_{\Theta} J$
     - Learn the model's parameters $\Theta = (W, b)$ by minimizing the Cost
  6) Use the learned parameters to make predictions (on the test set)

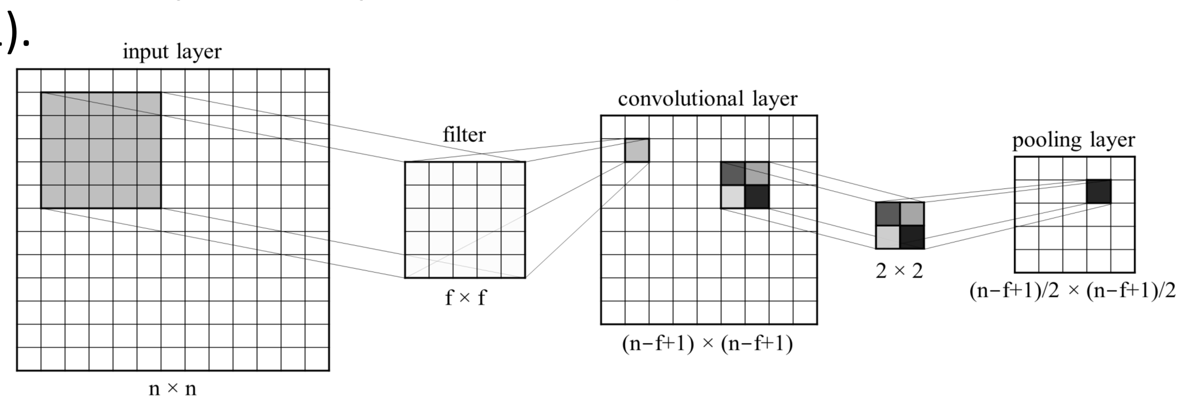
Fig. 1. Basic model of an artificial neuron (bottom), and basic architecture of a Neural Net (right).



Convolutional Neural Nets

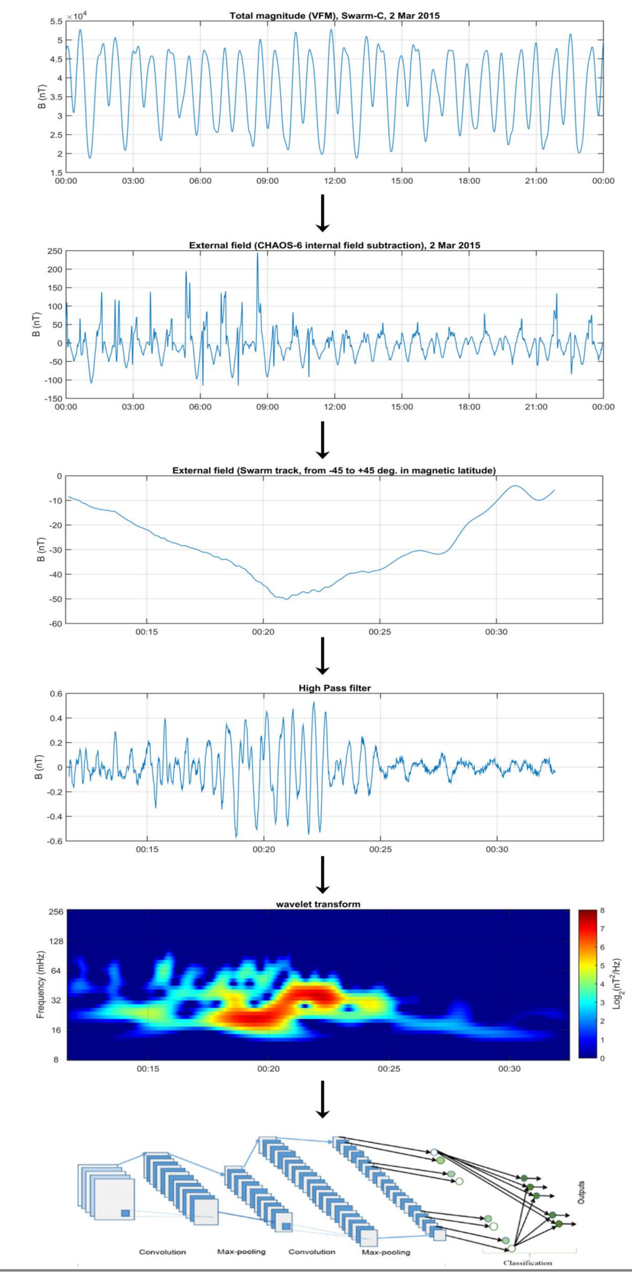
- Based on regular neural nets. They are trainable architectures of multiple stages. Each stage is composed of two major layers: convolutional & pooling.
  - ***Convolutional Layer (CONV):*** a number of trainable filters perform discrete convolution operations on *local patches* of the input wrt its dimensions. Layer's hyper-parameters: **filter size *f***, **stride *s***.
  - ***Pooling Layer (POOL):*** merges semantically similar features into one (LeCun et al., *Nature*, 2015). It takes small rectangular blocks from the CONV layer and *subsamples* it.
  - ***Fully Connected Layer (FC):*** the output of the topmost CONV layer is converted to a 1D feature vector (*flattening*). The top layer is fully connected, with one output unit per class label (Ciresan et al., IJCAI-11, AAAI Press, 2011).

Fig. 2. The two major processes in a ConvNet (in 2-D): convolution & max pooling



Proposed Methodology

Swarm Time Series, raw data
↓
External residual (Subtract CHAOS-6 model (internal part) from Swarm raw data)
↓
Smaller windows in time:
- cut each time series per satellite track;
- keep only low & mid-latitudes (±45°)
↓
Apply high-pass filter
- cut-off = 22 mHz
↓
Wavelet-transformed series
↓
ConvNet



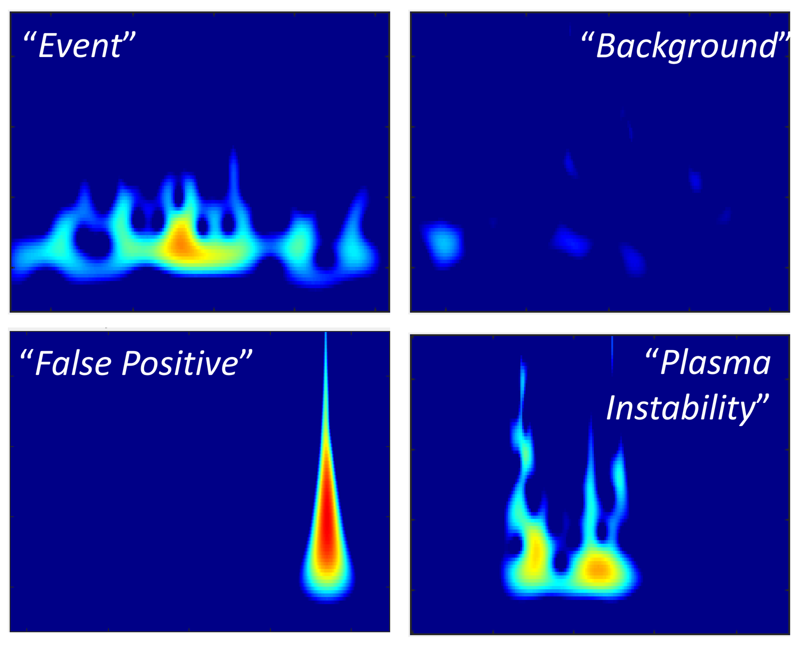


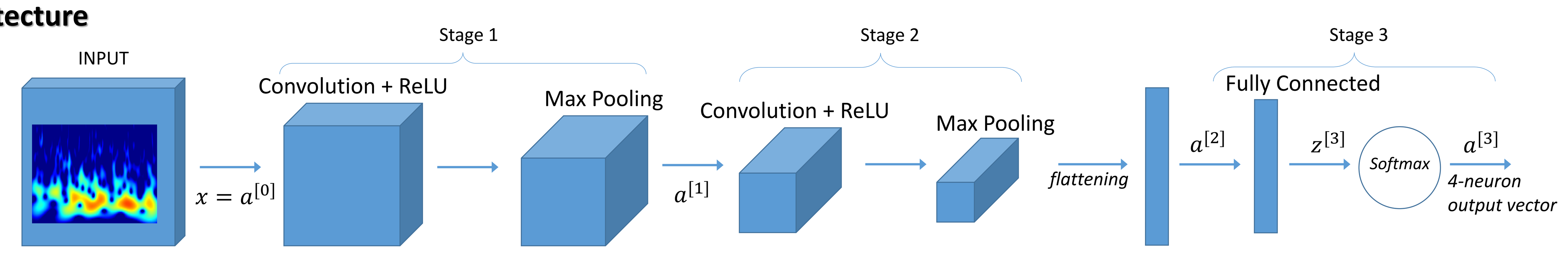
Fig. 3. The 4-class classification problem (Balasis et al., 2019):
1st class ("Events")*: existence of Pc3 (22-100 mHz) ULF wave events;*
2nd class ("Background noise")*: background noise without significant wave activity;*
3rd class ("False Positives")*: artificial signals that exhibit wave power in the Pc3 range;*
4th class ("Plasma Instabilities")*: or "plasma bubbles", attributed primarily to Equatorial Spread-F events (ESF).*

Training of the Network (Antonopoulou et al., 2022)

- *Data used: **total magnitude**, **Swarm-C**, **VFM**, NEC frame, 1s sampling rate (MAGX_LR_1B Product), for February, March & April of the year 2015.*
- **Number of total samples:** 2620 samples, *manually annotated* in 4 classes
- **Input:** pairs of wavelet images & their annotation (class label)
- **Training / Test set split:** 80% / 20% of total samples
- **Parameter initializer:** Xavier Initialization
- **Activation functions:** ReLU, Softmax
- **Cost function:** Cross-entropy (Log Loss)
- **Optimizer:** Adam Optimization (*training for 100 epochs*)

Network's Architecture

Fig. 4. A simplified view of our ConvNet architecture: two alternating CONV and POOL layers, and one FC layer.



Layers	Details
Conv1	8 filters, f = 4, s = 1
Pool1	f = 8, s = 4
Conv2	16 filters, f = 2, s = 1
Pool2	f = 4, s = 4
FC	4-neuron output

Results

Fig. 5. Confusion matrix and Precision & Recall for each class of the test set. The confusion matrix is almost diagonal.

Predicted class \ Actual class	Background	Events	PI's	FP's
Background	167	2	3	0
Events	1	194	7	0
PI's	0	1	76	0
FP's	0	0	0	73

Classes	Recall (%)	Precision (%)
Background Noise	99.4	97.1
Pc3 ULF wawe Events	98.5	96.0
Plasma Instabilities (PI's)	88.4	98.7
False Positives (FP's)	100	100

Conclusions

✓ *Accuracy on the training set (2096 samples) = 98.3%*
✓ *Accuracy on the test set (524 samples) = 97.3%*
✓ *Heidke Skill Score (HSS) = 96.2%*
✓ *Comparing with the popular **k-Nearest Neighbors (kNN)** and the very competitive **Support Vector Machines (SVM)** classifiers: kNN (k=5, p=1) = 57.5%, SVM = 88.1% → ConvNet gives the best results with the highest accuracy.*
✓ *The methodology could be applied to investigate:*
  - *other frequency ranges (Pc1/EMIC, Pc2, Pc4, Pc5)*
  - *observations from other satellite missions*
  - *ground-based observations*

1. Antonopoulou, A.; Balasis, G.; Papadimitriou, C.; Boutsi, A.Z.; Rontogiannis, A.; Koutroumbas, K.; Daglis, I.A.; Giannakis, O. Convolutional Neural Networks for Automated ULF Wave Classification in Swarm Time Series. *Atmosphere* **2022**, *13*, 1488. https://doi.org/10.3390/atmos13091488.
2. Balasis, G., S. Aminalragia-Giamini, C. Papadimitriou, I. A. Daglis, A. Anastasiadis, and R. Haagmans, A machine learning approach for automated ULF wave recognition, *J. Space Weather Space Clim.* **2019**, 9, A13, https://doi.org/10.1051/swsc/2019010.