A&A, 687, A45 (2024)
https://doi.org/10.1051/0004-6361/202449532

Astronomy & Astrophysics

Determining the time before or after a galaxy merger event

W. J. Pearson[1], V. Rodriguez-Gomez[2], S. Kruk[3], and B. Margalef-Bentabol[4]

[1] National Centre for Nuclear Research, Pasteura 7, 02-093 Warszawa, Poland
e-mail: william.pearson@ncbj.gov.pl
[2] Instituto de Radioastronomía y Astrofísica, Universidad Nacional Autónoma de México, Apdo. Postal 72-3, 58089 Morelia, Mexico
[3] European Space Agency (ESA), European Space Astronomy Centre (ESAC), Camino Bajo del Castillo s/n, 28692 Villanueva de la Cañada, Madrid, Spain
[4] SRON Netherlands Institute for Space Research, Landleven 12, 9747 AD Groningen, The Netherlands

Received 7 February 2024 / Accepted 16 April 2024

Narodowe Centrum Badań Jądrowych
National Centre for Nuclear Research

Overview

- Introduction
- Data
- Neural Networks
- Latent Space

Introduction

- Galaxy mergers have time dependant properties (e.g. SFR)
- Take a long time (Gyr)
- Observations don’t give us the time
- Simulations do give us the time

Introduction

- From simulations to reality?

william.pearson@ncbj.gov.pl

Introduction

- This paper looks at the first step:
- Can we train a deep learning method to estimate galaxy merger times with idealised images from simulations?

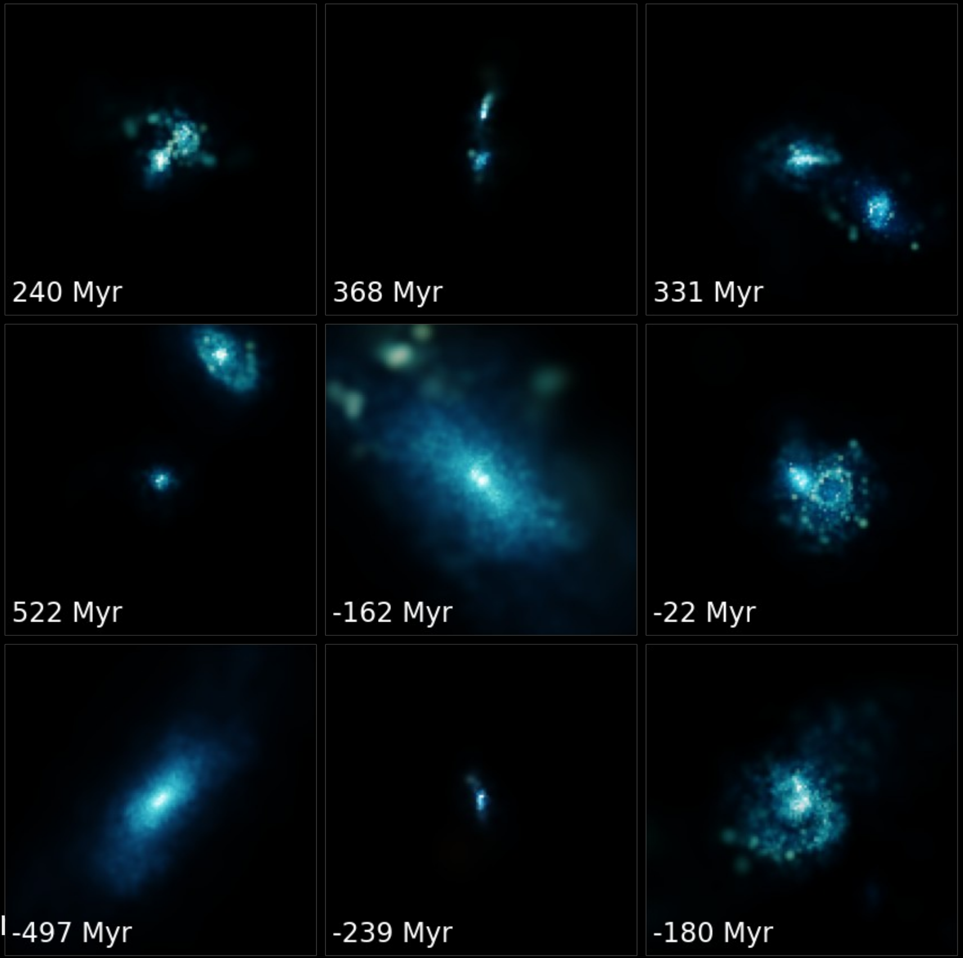
william.pearson@ncbj.gov.pl

Data – IllustrisTNG

- Large cosmological simulation
  - TNG100 (Nelson et al. 2019)
  - Time resolution of ~162 Myr
- Mergers
  - last 500 Myr
  - next 1000 Myr
  - Major - mass ratio 1:4
  - $z < 0.15$ ($0.07 < z < 0.15$)

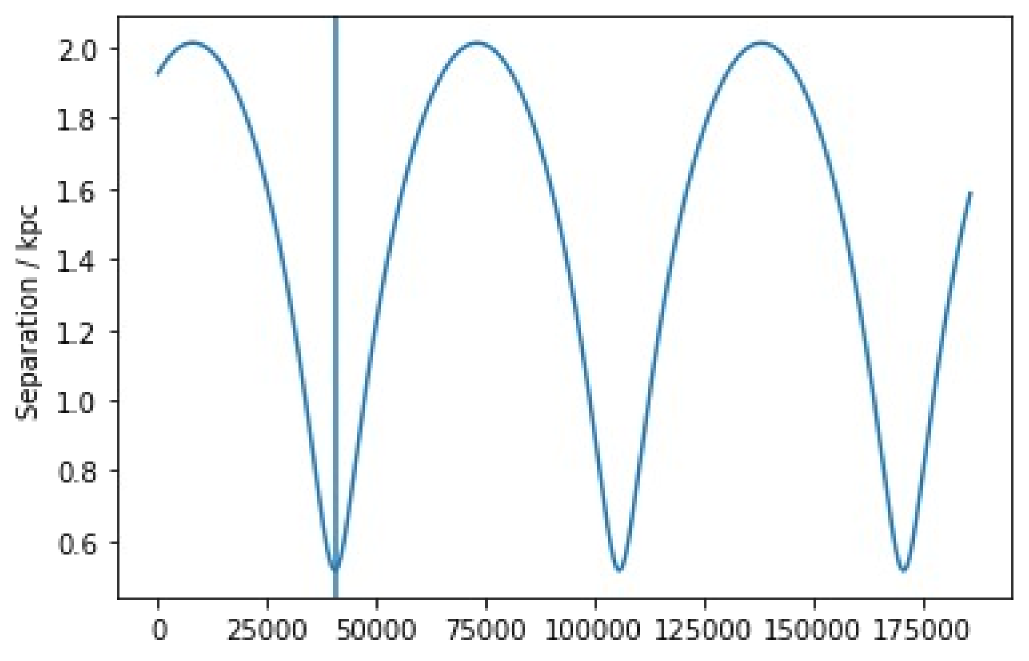
Data – IllustrisTNG

- ugri band, ~0.2 arcec
  - Match KiDS



Data – IllustrisTNG

- Particle simulation to increase time resolution

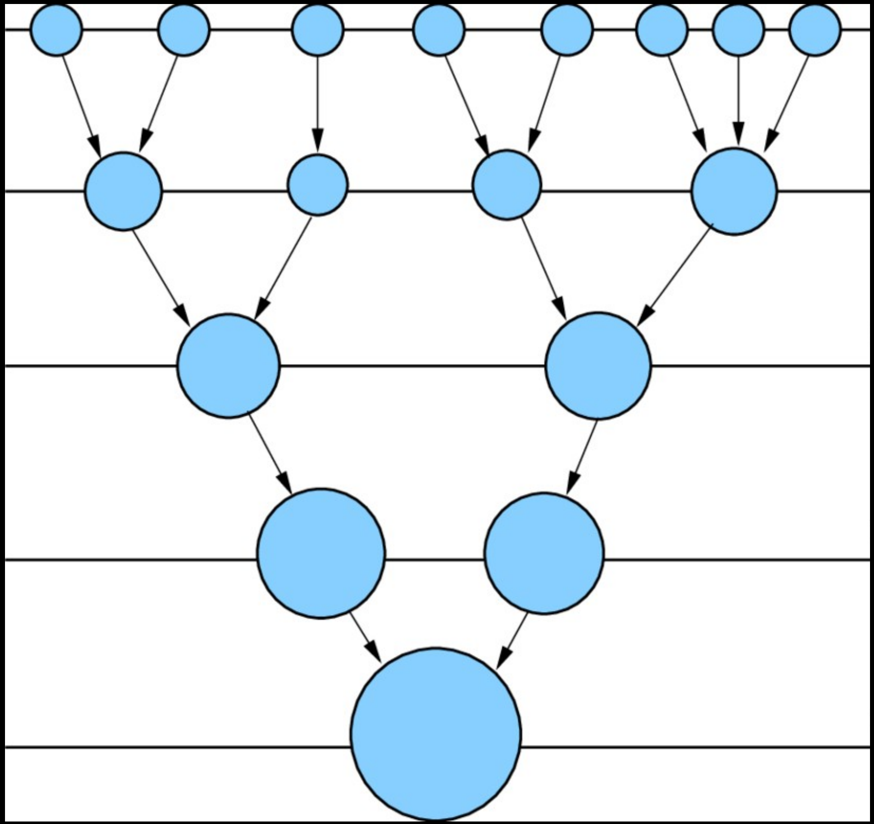


william.pearson@ncbj.gov.pl

Data – IllustrisTNG

- 3321 galaxies
  - Project along 3 axes to give 9963 images
- Split into training (80%), validation (10%), test (10%) sets
  - Ensure merger trees are only in one set

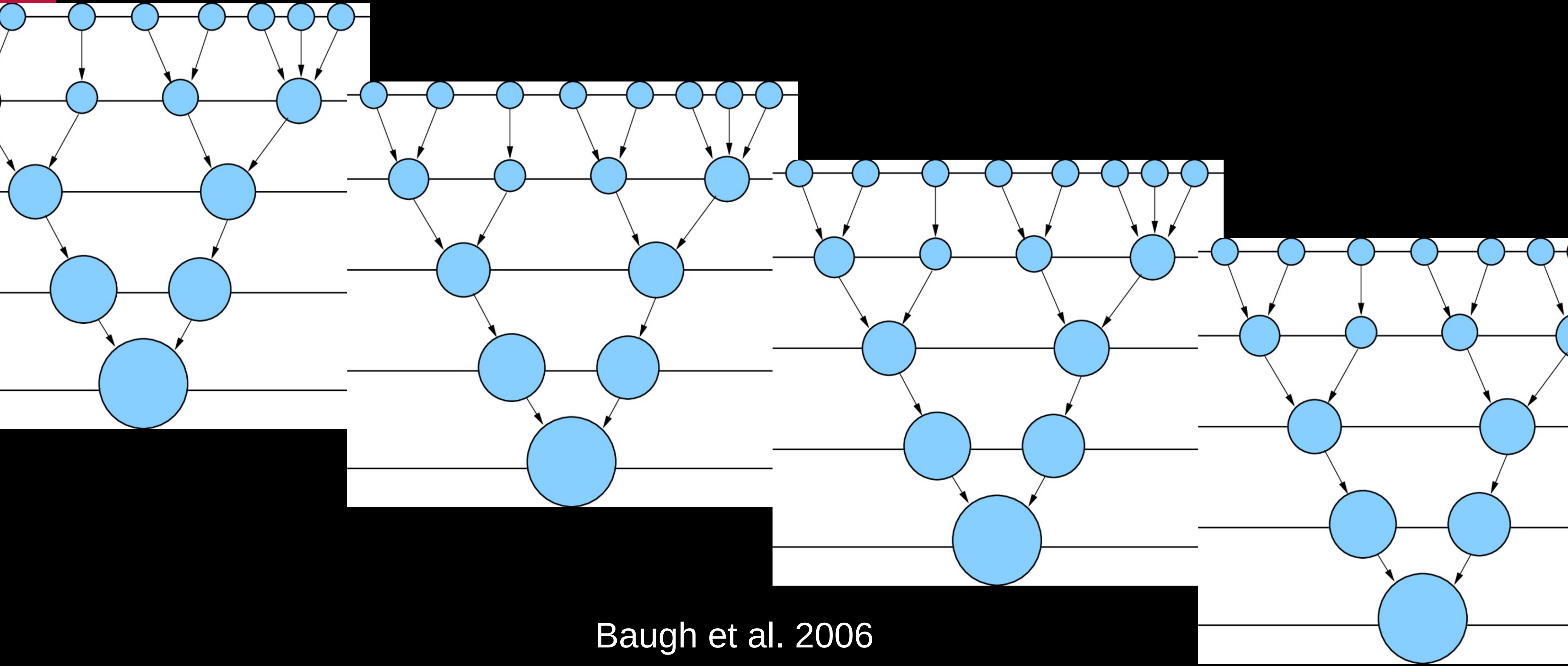
Data – IllustrisTNG



Baugh et al. 2006

william.pearson@ncbj.gov.pl

Data – IllustrisTNG



Baugh et al. 2006

william.pearson@ncbj.gov.pl

Data - Preprocessing

- Images
  - Arcsinh scale twice
  - Linearly normalise between 0 and 1
  - (Training) Randomly rotate the image by multiples of 90º
  - (Training) Randomly flip horizontally and vertically
- Times
  - Linearly normalise between 0 and 1

william.pearson@ncbj.gov.pl

Neural Networks

- Basically a series of matrix multiplications

  $\underline{\mathbf{wx}}+b$

- Wrapped in activation functions
  - ReLU: max(0, $\underline{\mathbf{wx}}+b$)
  - Sigmoid: $\frac{1}{1+e^{\vec{w}\vec{x}+b}}$

william.pearson@ncbj.gov.pl

Neural Networks

- Image classification typically uses convolution layers

Some maths → Sloth

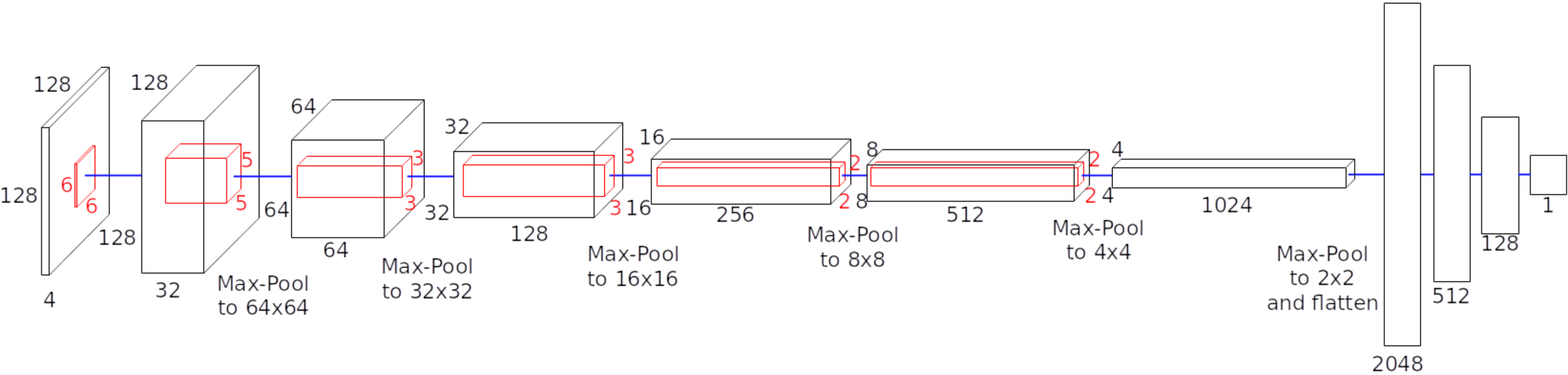
- Then pool: take average or max of some group of neurons

Neural Networks

- Convolutional Neural Network (CNN)
- Autoencoder
- Residual Network
- Swin Transformer

william.pearson@ncbj.gov.pl

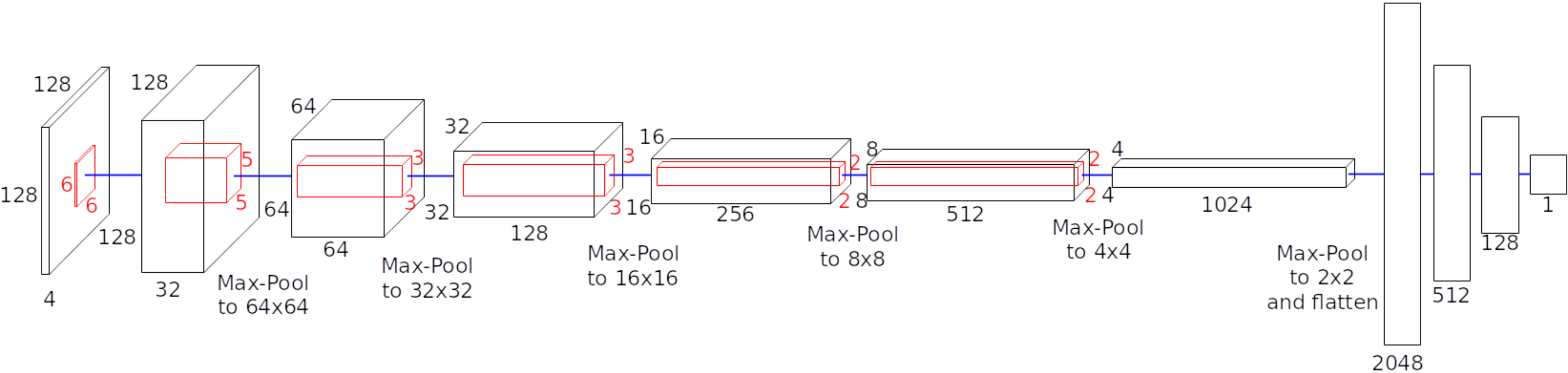
CNN



- Input: 4 channel 128x128 image
- Output: 1 sigmoid neuron – merger time

william.pearson@ncbj.gov.pl

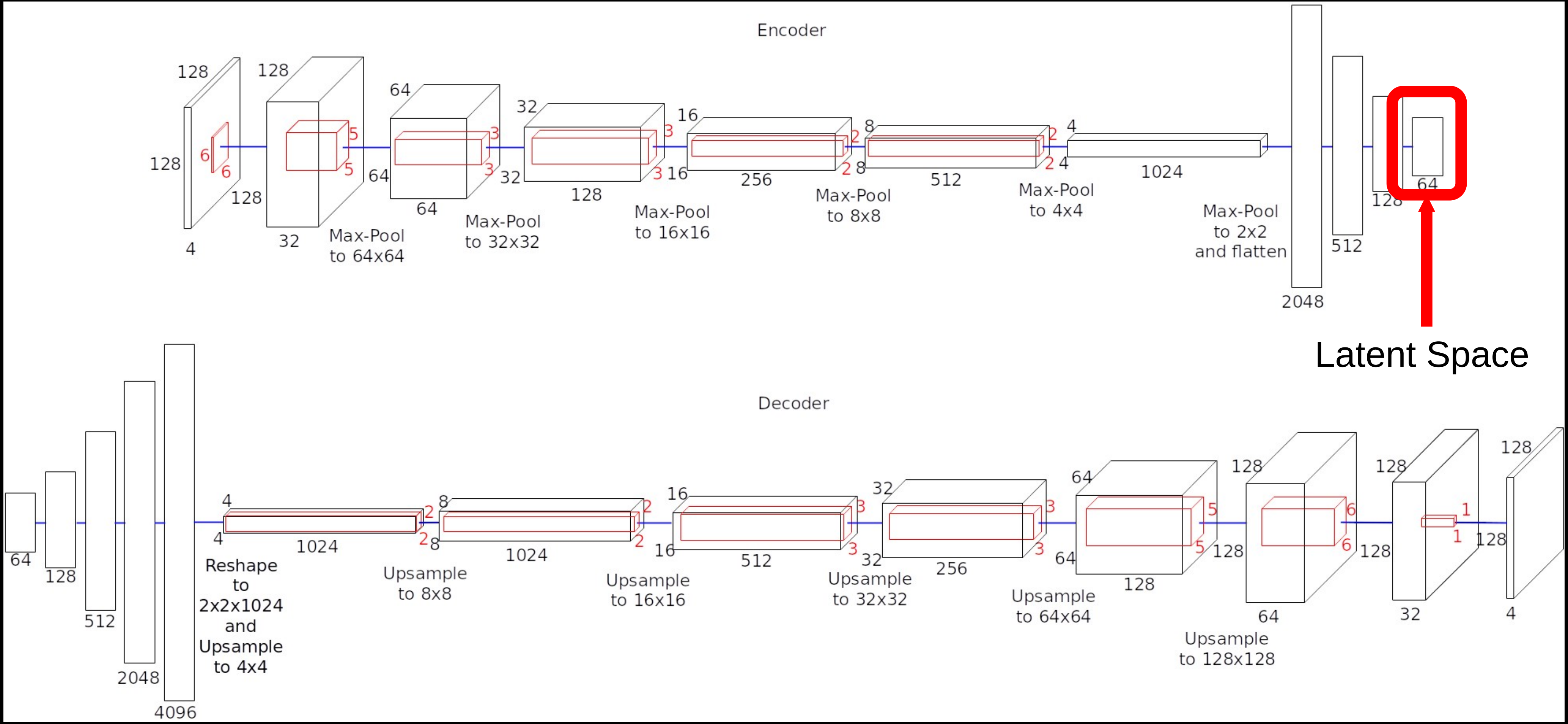
CNN



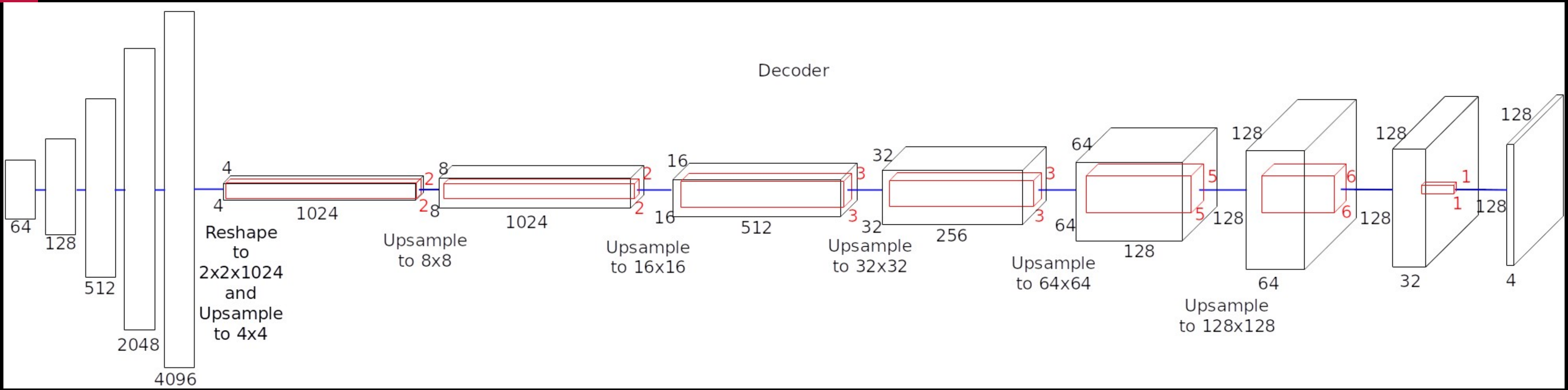
- Trained on MSE with ADAM optimiser

william.pearson@ncbj.gov.pl

Autoencoder



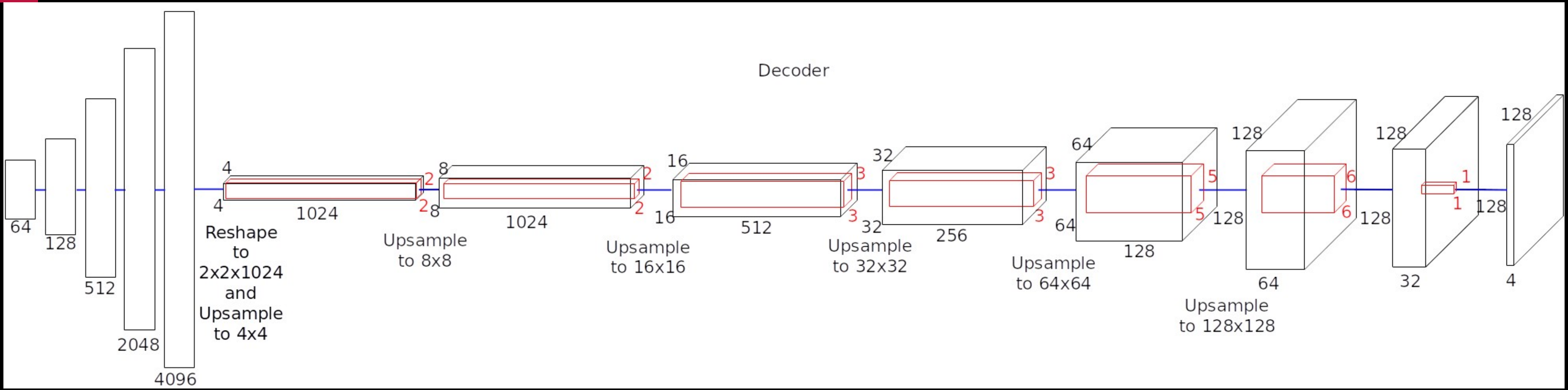
Autoencoder



- Input: 4 channel 128x128 image
- Output: 4 channel 128x128 image
  1 sigmoid latent space neuron - time

william.pearson@ncbj.gov.pl

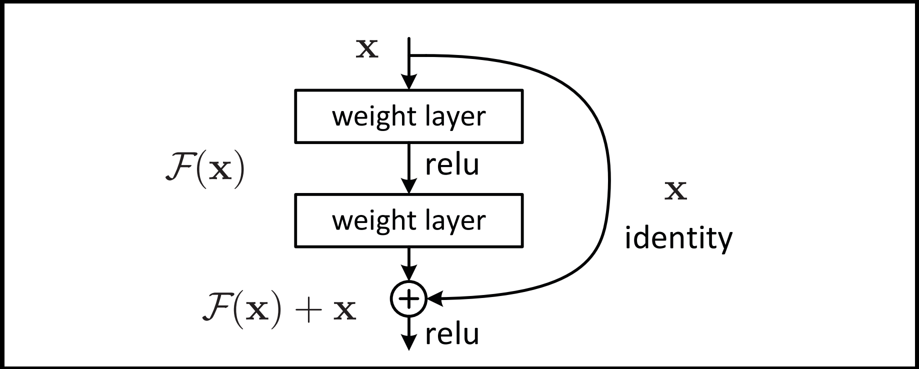
Autoencoder



- Trained on MSE of time + reproduced image with ADAM
- Selected on MSE of time

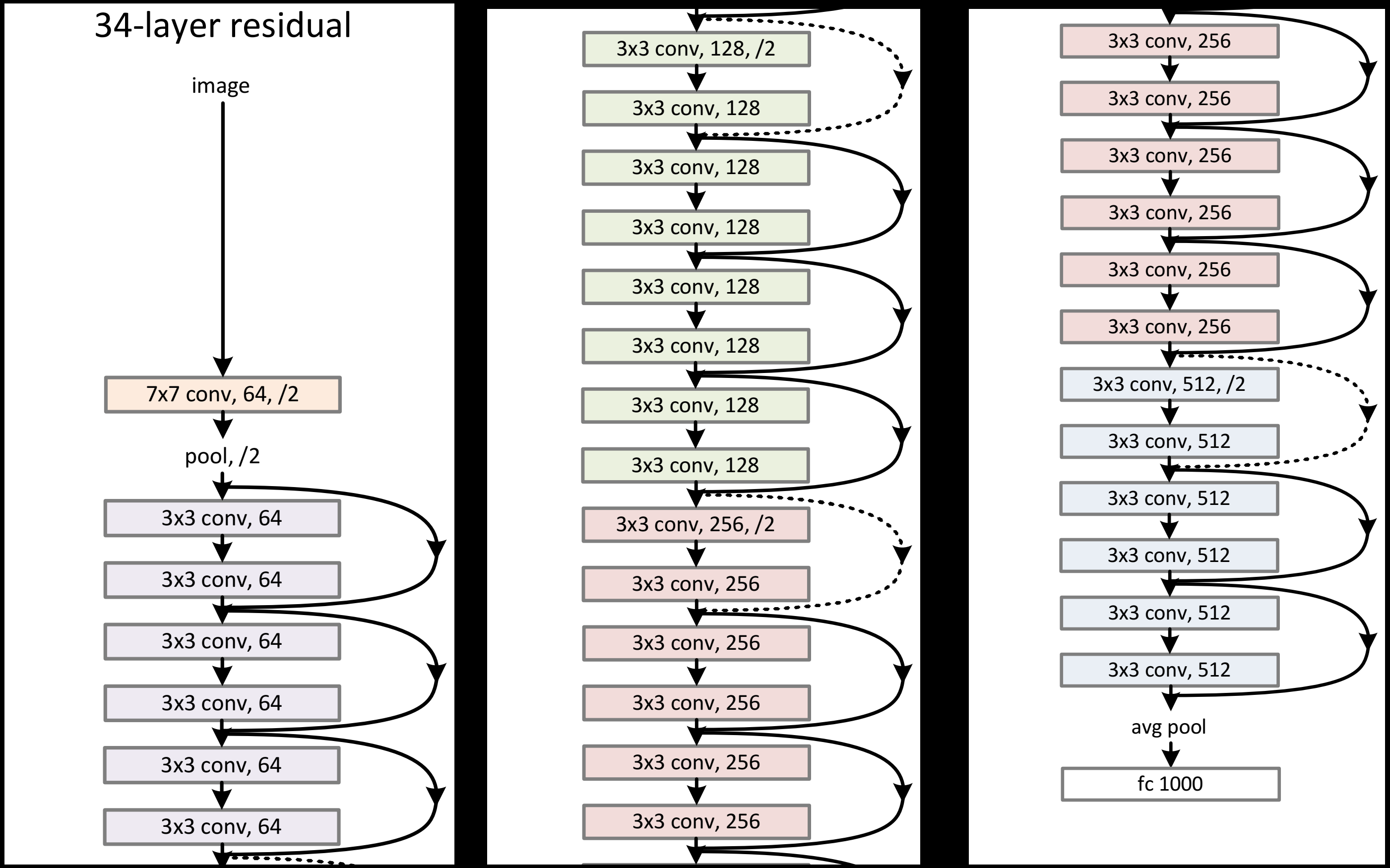
william.pearson@ncbj.gov.pl

Residual Network (ResNet)



He et al. 2016

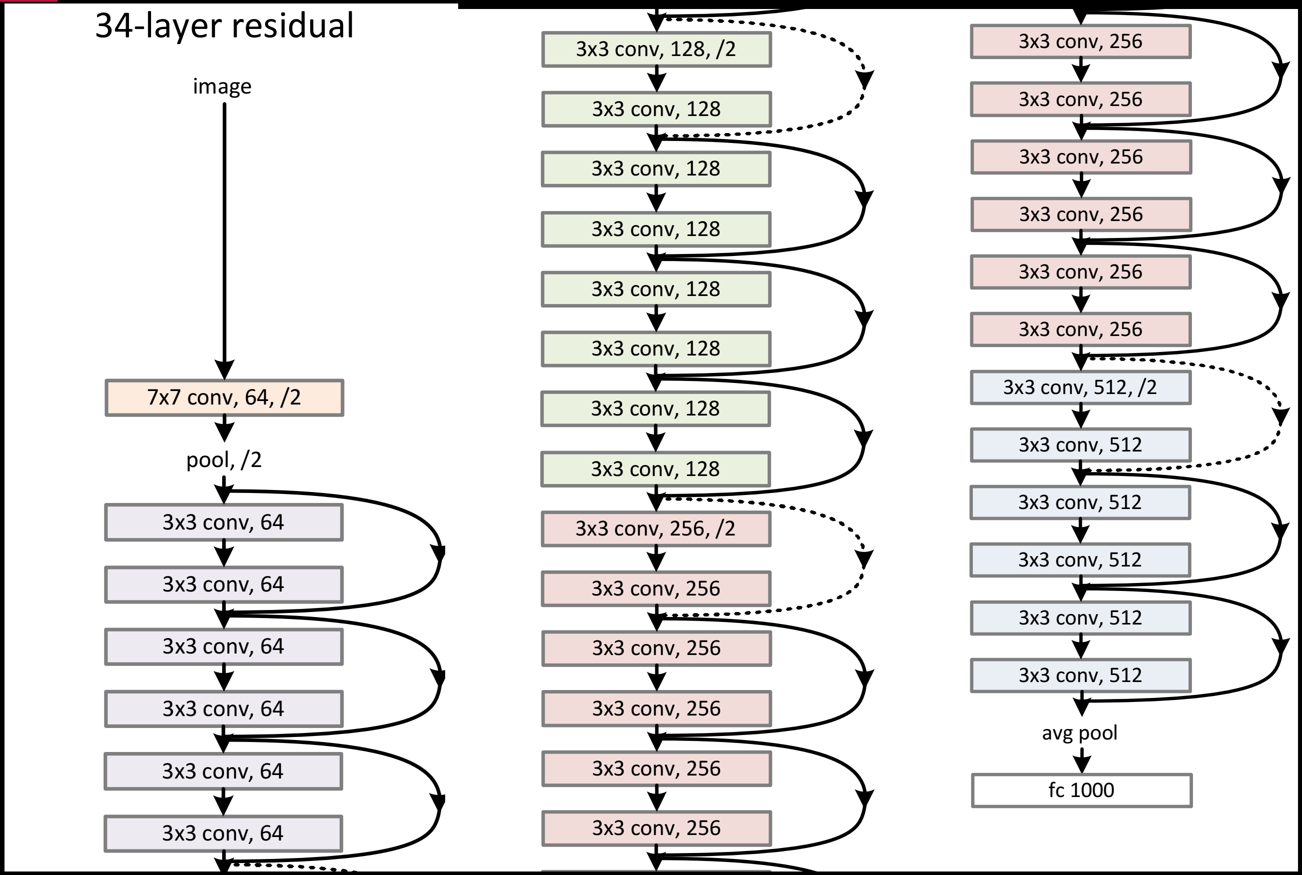
Residual Network (ResNet)



william.pearson@ncbj.gov.pl

He et al. 2016

Residual Network (ResNet)

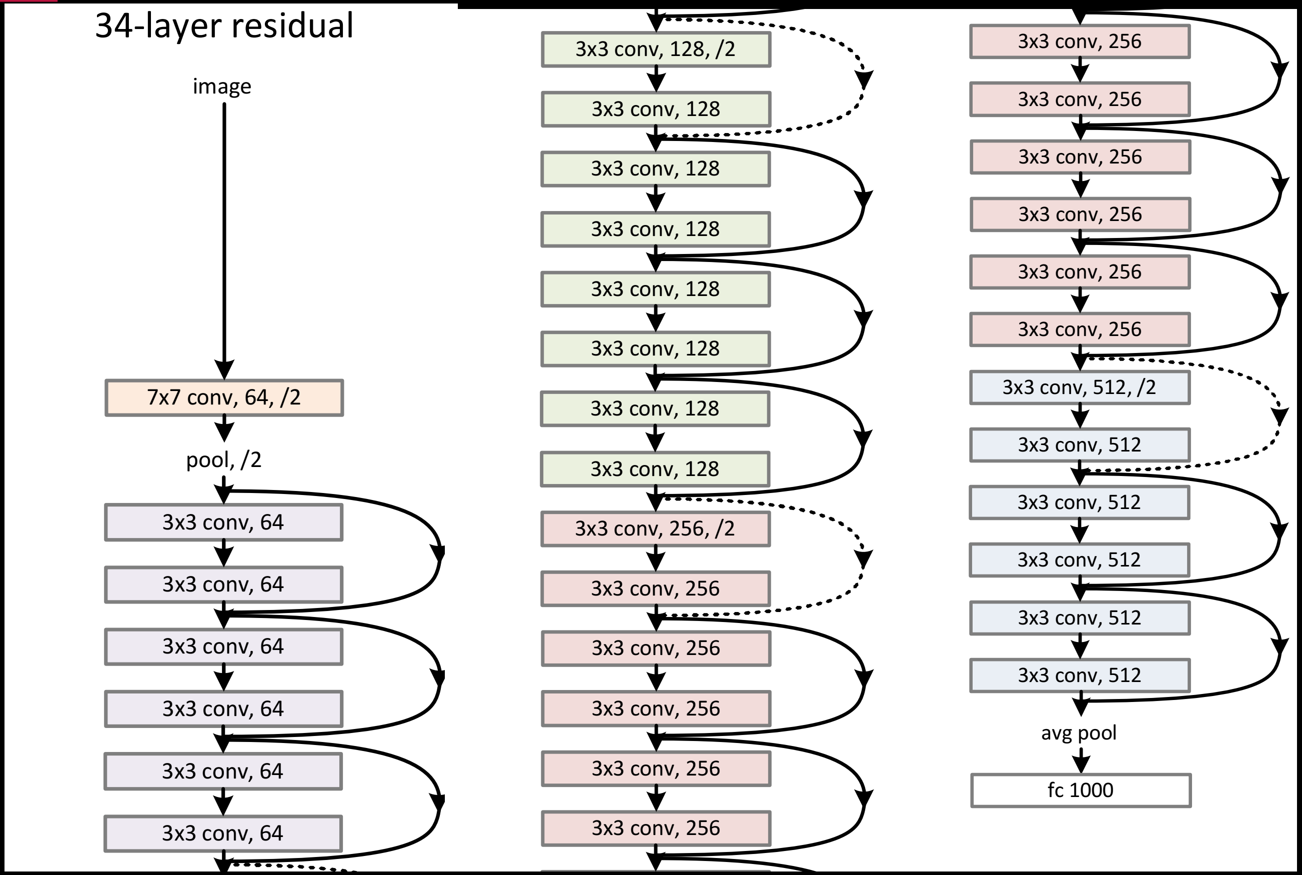


- ResNet50
- Input: 3 channel 128x128 image
- Output: 1 sigmoid neuron

He et al. 2016

william.pearson@ncbj.gov.pl

Residual Network (ResNet)



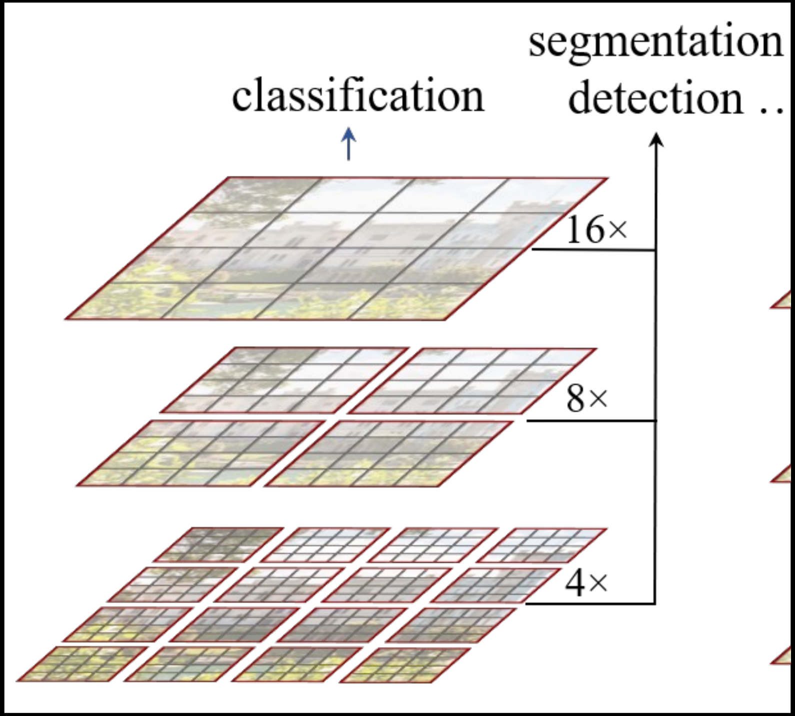
- Trained on MSE with ADAM

He et al. 2016

Swin Transformer

- Transformers come from Natural Language Processing
  - Use attention to model long-range dependencies
    - They can remember
- Swin Transformers apply this to images

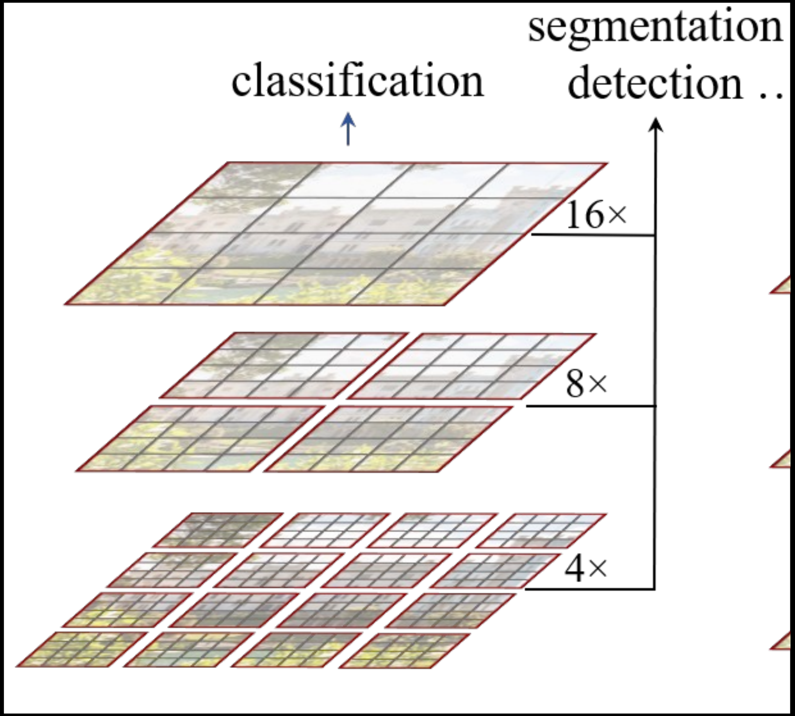
Swin Transformer

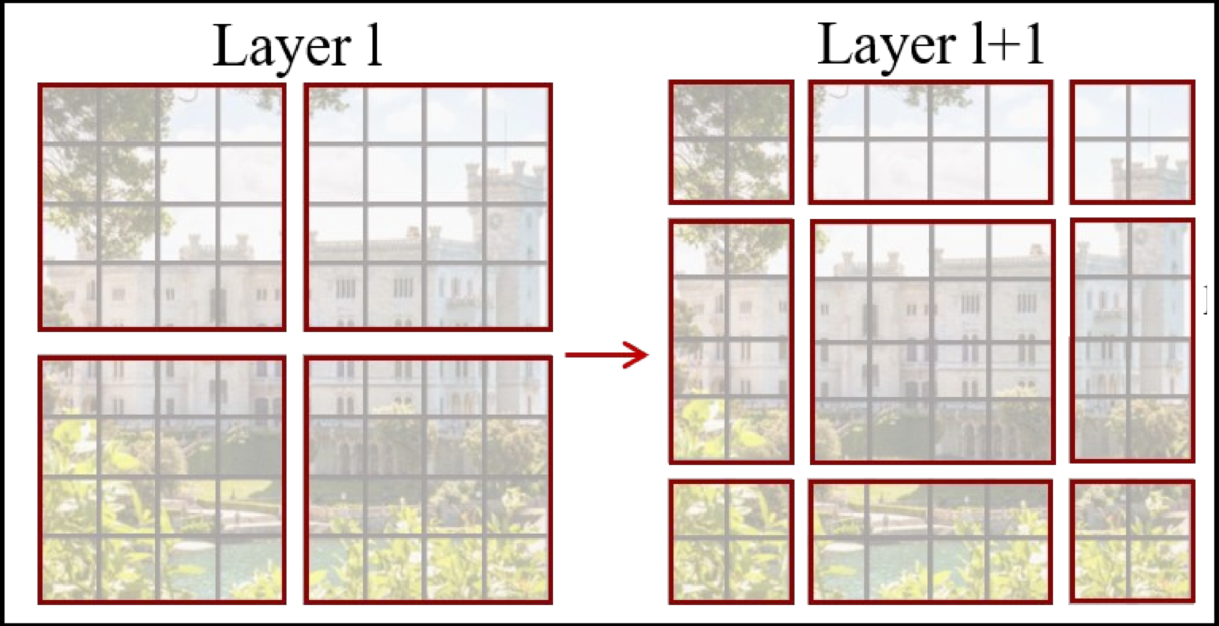


Liu et al. 2021

william.pearson@ncbj.gov.pl

Swin Transformer

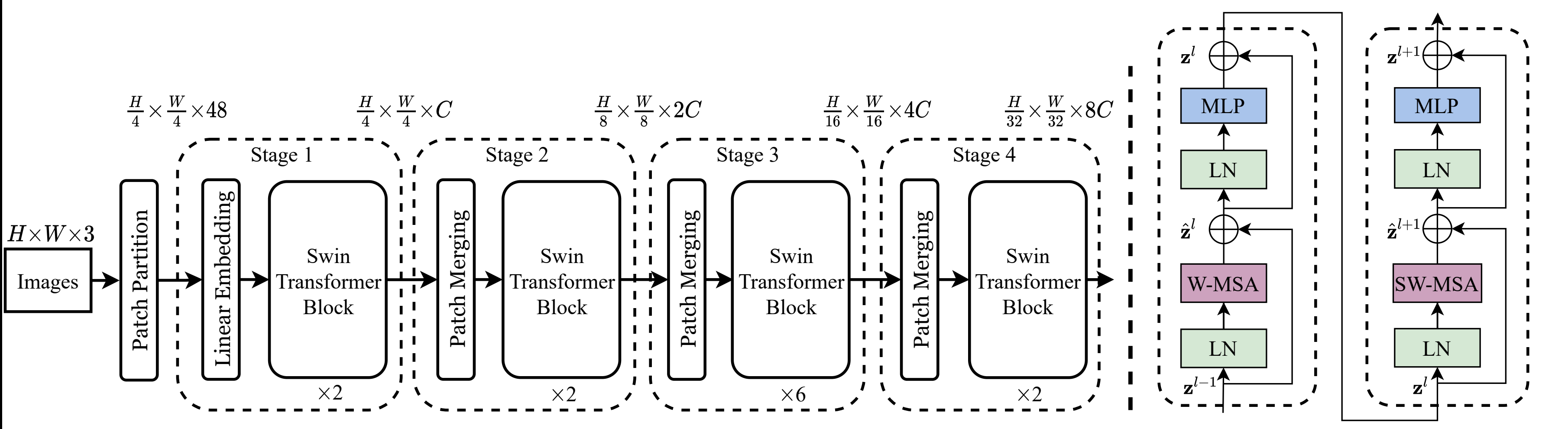




Liu et al. 2021

william.pearson@ncbj.gov.pl

Swin Transformer

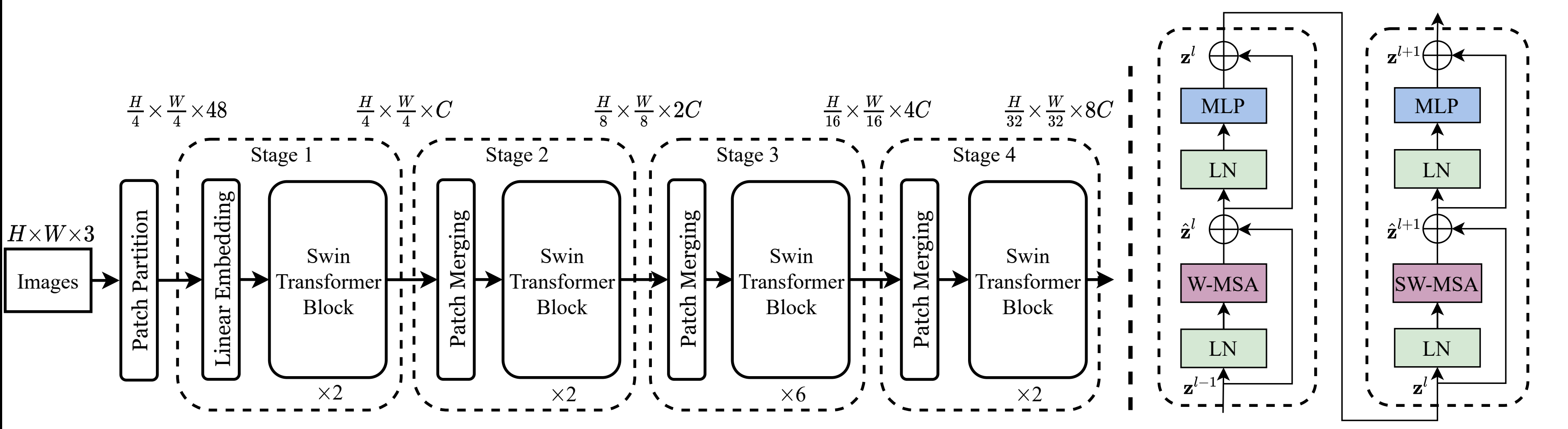


(a) Architecture (b) Two Successive Swin Transformer Blocks

Liu et al. 2021

- Input: 3 channel 224x224 image
- Output: 1 sigmoid latent space neuron - time

Swin Transformer



(a) Architecture

(b) Two Successive Swin Transformer Blocks

Liu et al. 2021

- Pre-trained on ImageNet-1K
- Trained on MSE of time with stochastic gradient descent (SDG)

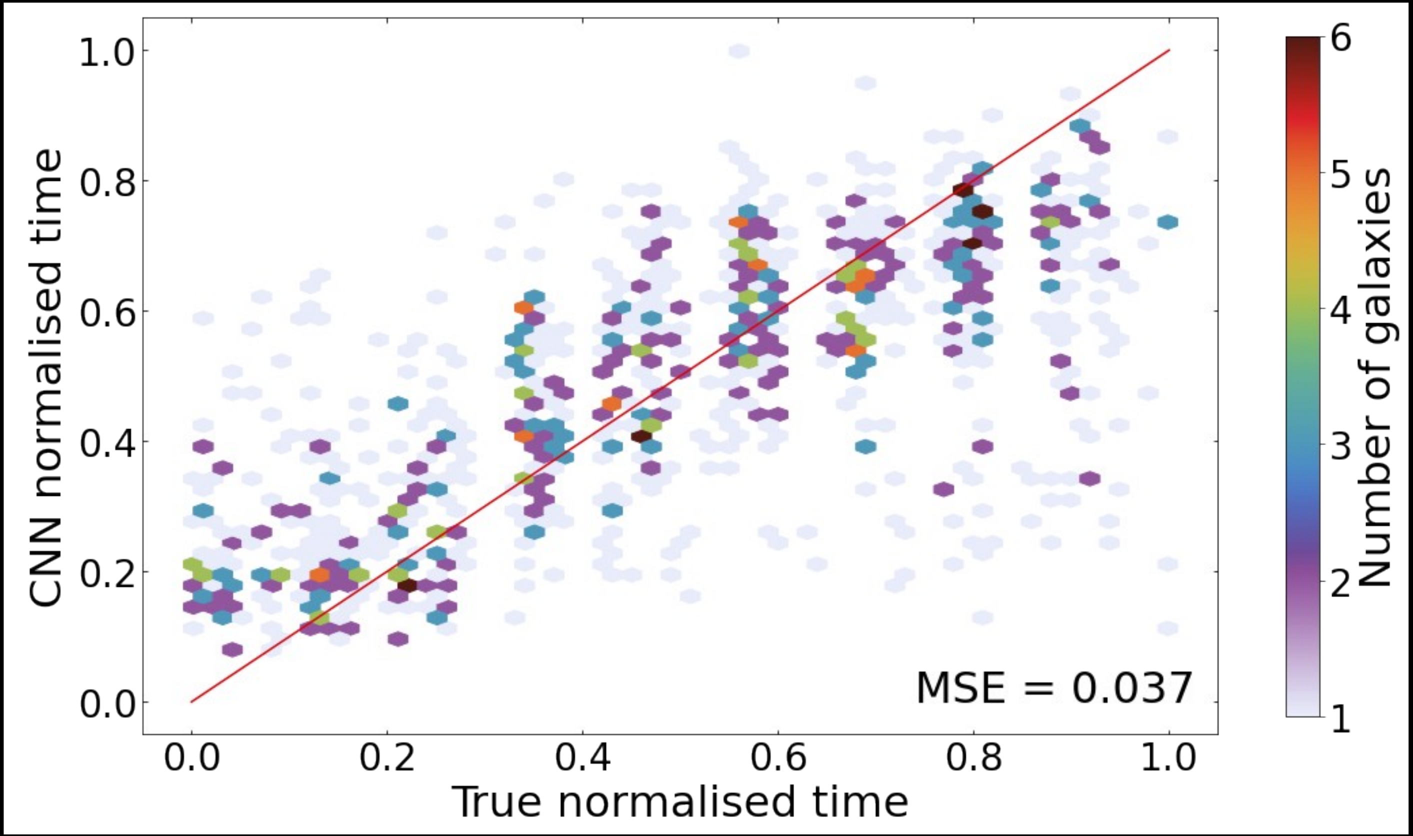
Results

Architecture	Training MSE	Validation MSE	Validation error	
			Mean [a]	Median [a]
ResNet50	0.075	0.065	327	324
Swin	0.034	0.042	236	193
CNN	0.040	0.037	215	165
Autoencoder	0.036	0.038	222	160

Notes. [a] Values in Myr.

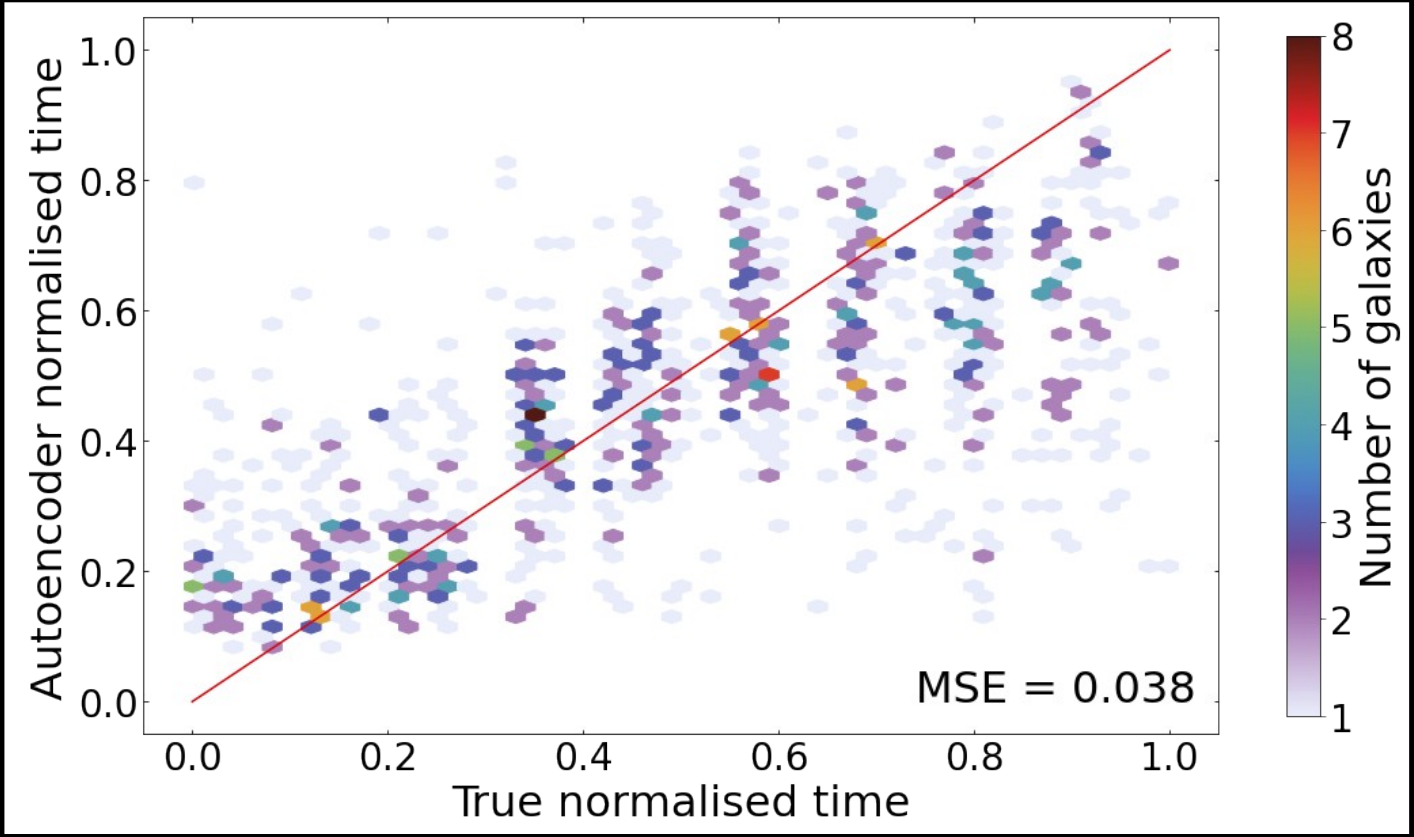
william.pearson@ncbj.gov.pl

Results - CNN



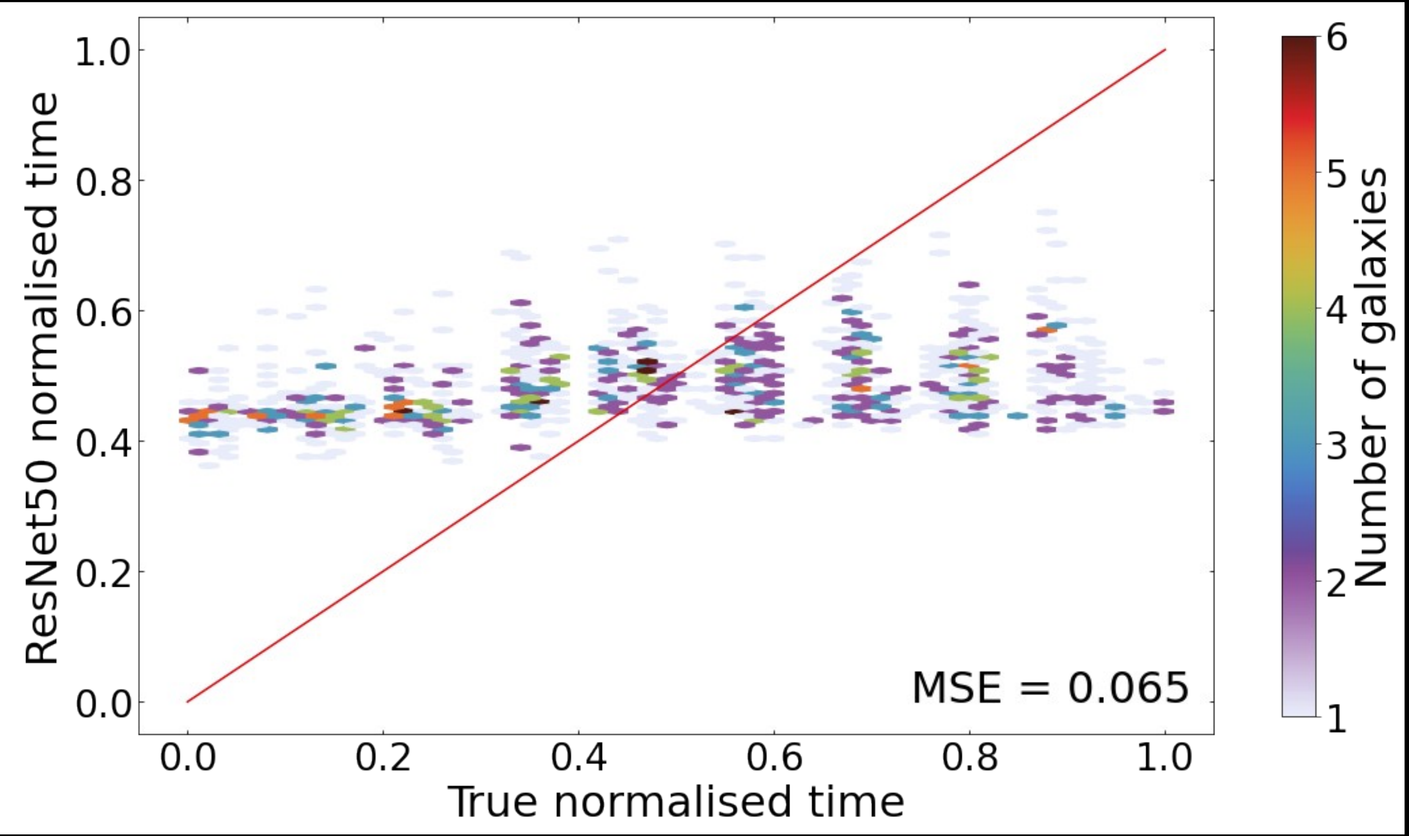
CNN normalised time
True normalised time
MSE = 0.037
Number of galaxies
0.0
0.2
0.4
0.6
0.8
1.0
1
2
3
4
5
6

Results - Autoencoder

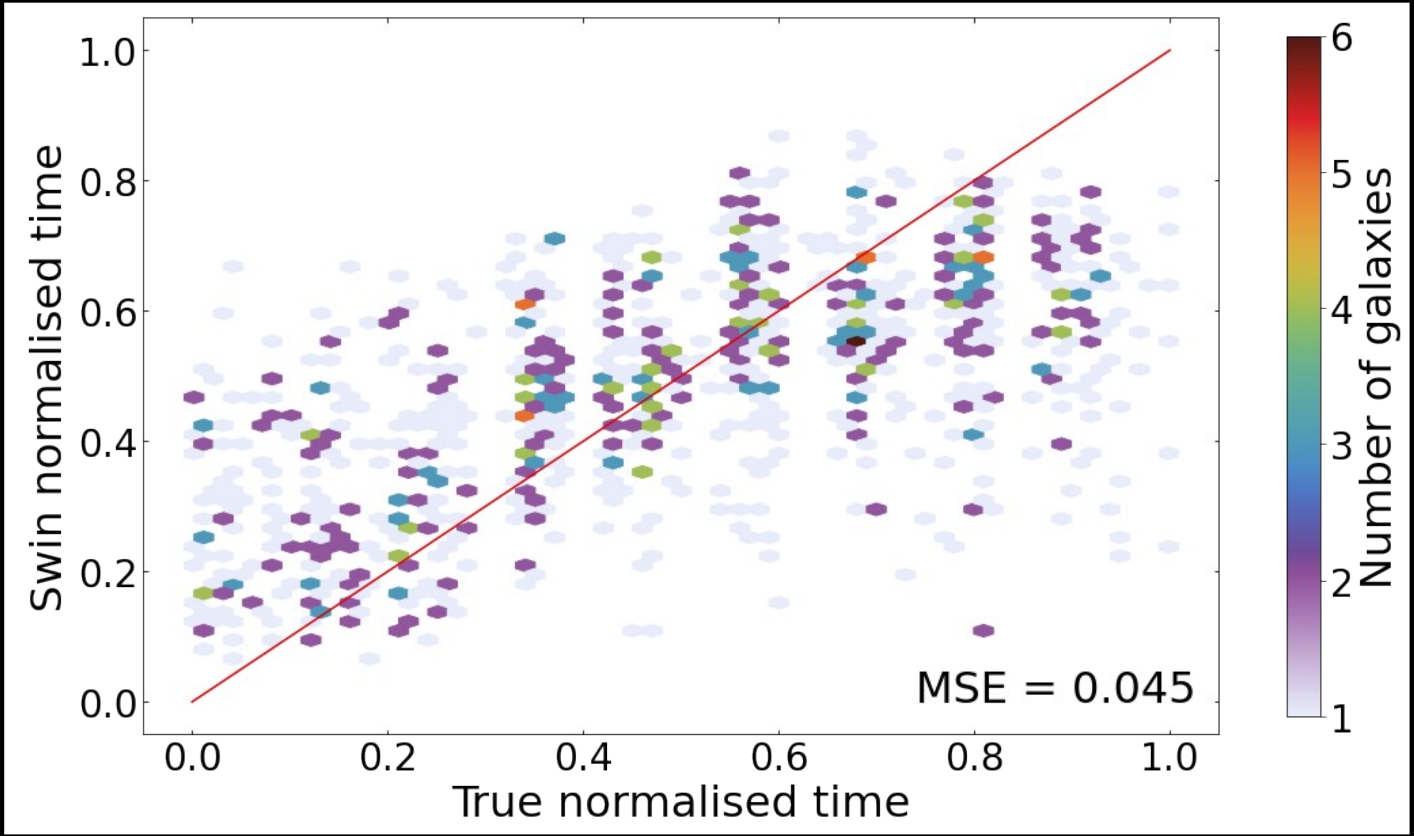


william.pearson@ncbj.gov.pl

Results - ResNet50



Results - Swin Transformer



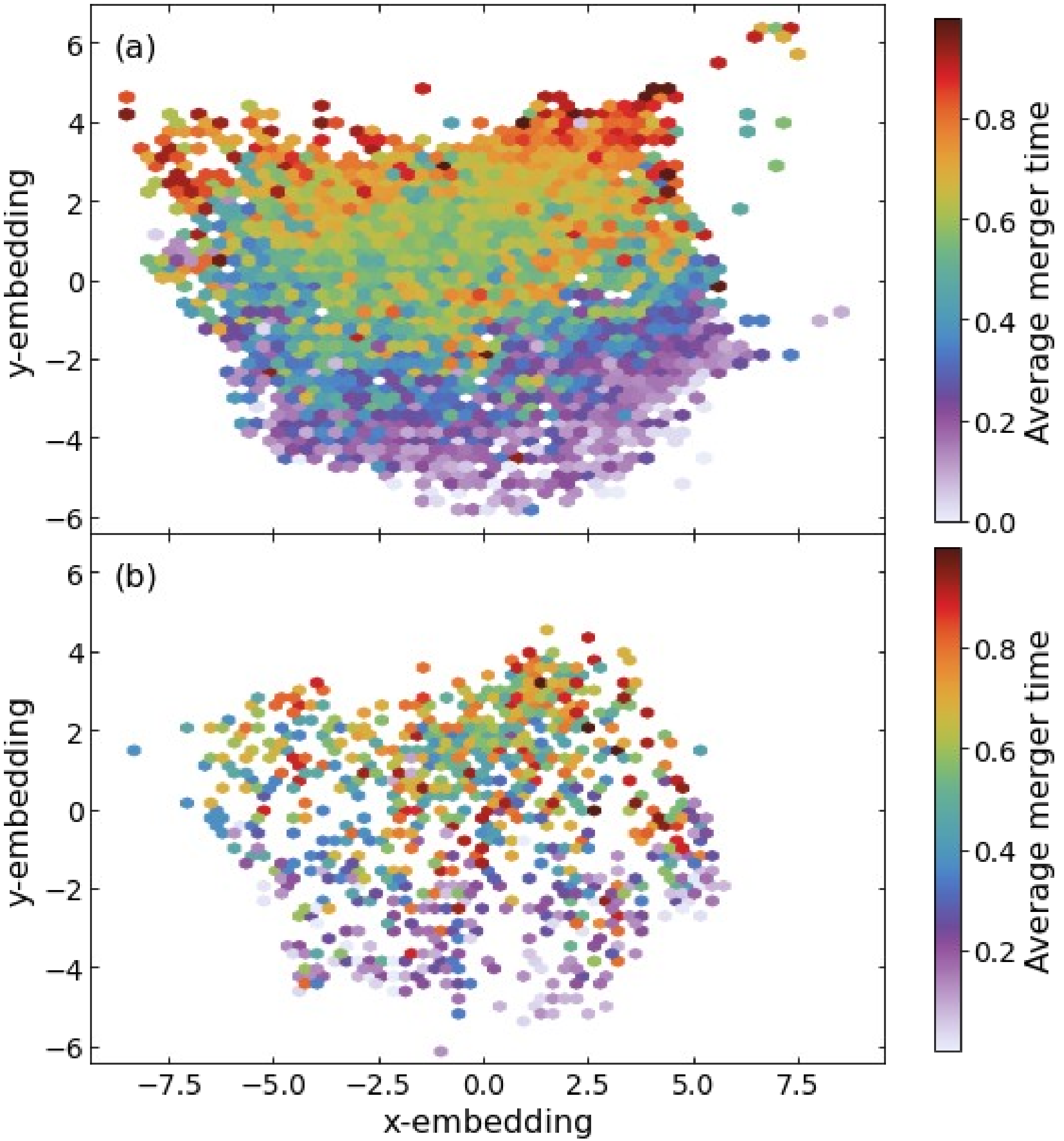
william.pearson@ncbj.gov.pl

Latent Space

- Isomap (Tenenbaum et al. 2000, scikit learn)
- Linear Discriminant Analysis (LDA, scikit learn)
- Neighbourhood Components Analysis (NCA, Goldberger et al. 2004, scikit learn)
- Sparse Random Projection (SRP, Lie et al. 2006, scikit learn)
- Truncated Singular Value Decomposition (TSVD, Halko et al. 2011, scikit learn)
- Uniform Manifold Approximation and Projection (UMAP, McInnes et al. 2018)

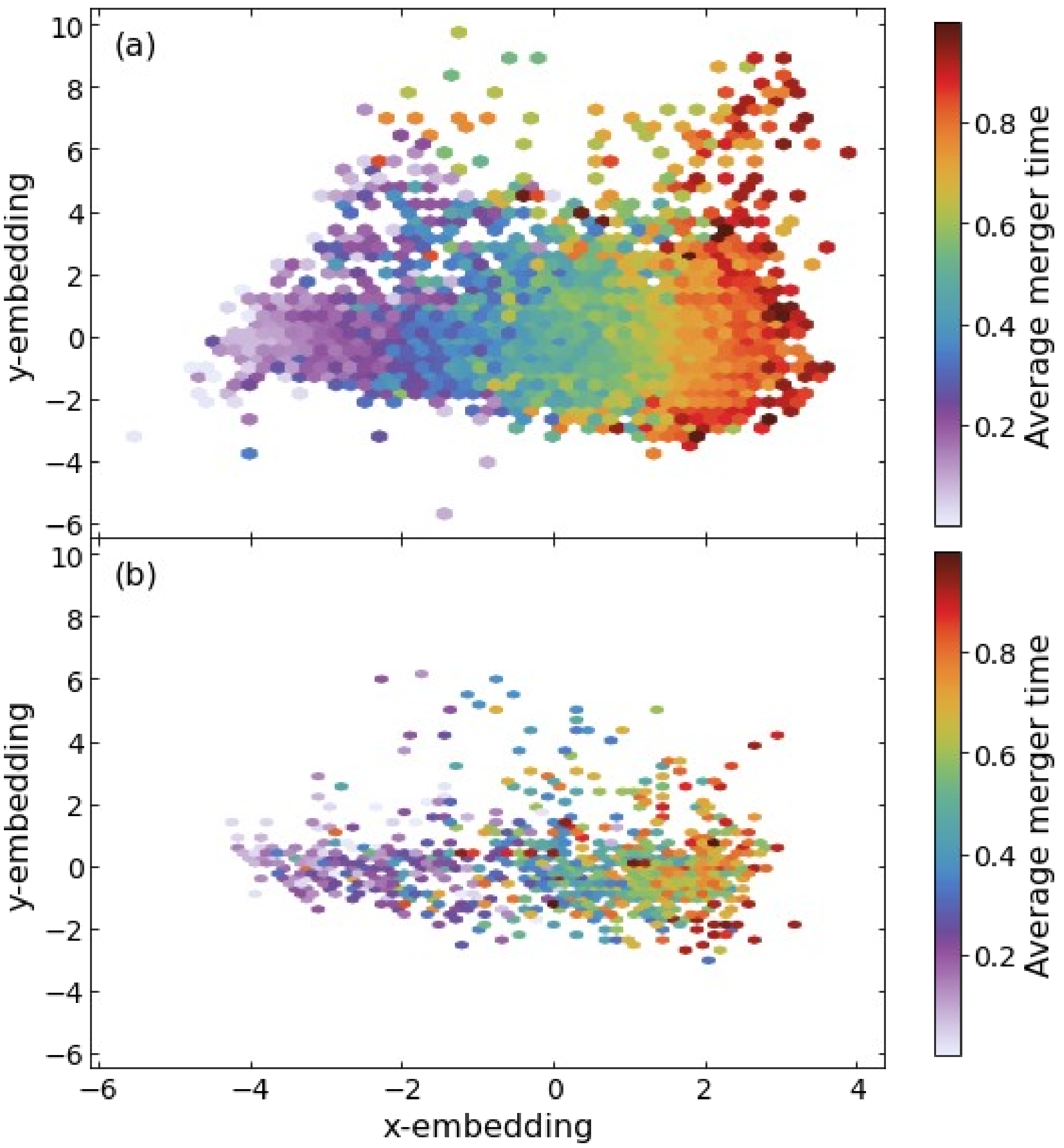
Isomap

- Projects multi-dimension space into lower dimensions in a non-linear way (unsupervised)
  - Construct neighbourhood map
  - Find shortest paths between each pair
  - Use multidimensional scaling to reduce dimensions using shortest paths


(a)
(b)
y-embedding
x-embedding
Average merger time
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
6
4
2
0
−2
−4
−6
0.0
0.2
0.4
0.6
0.8

LDA

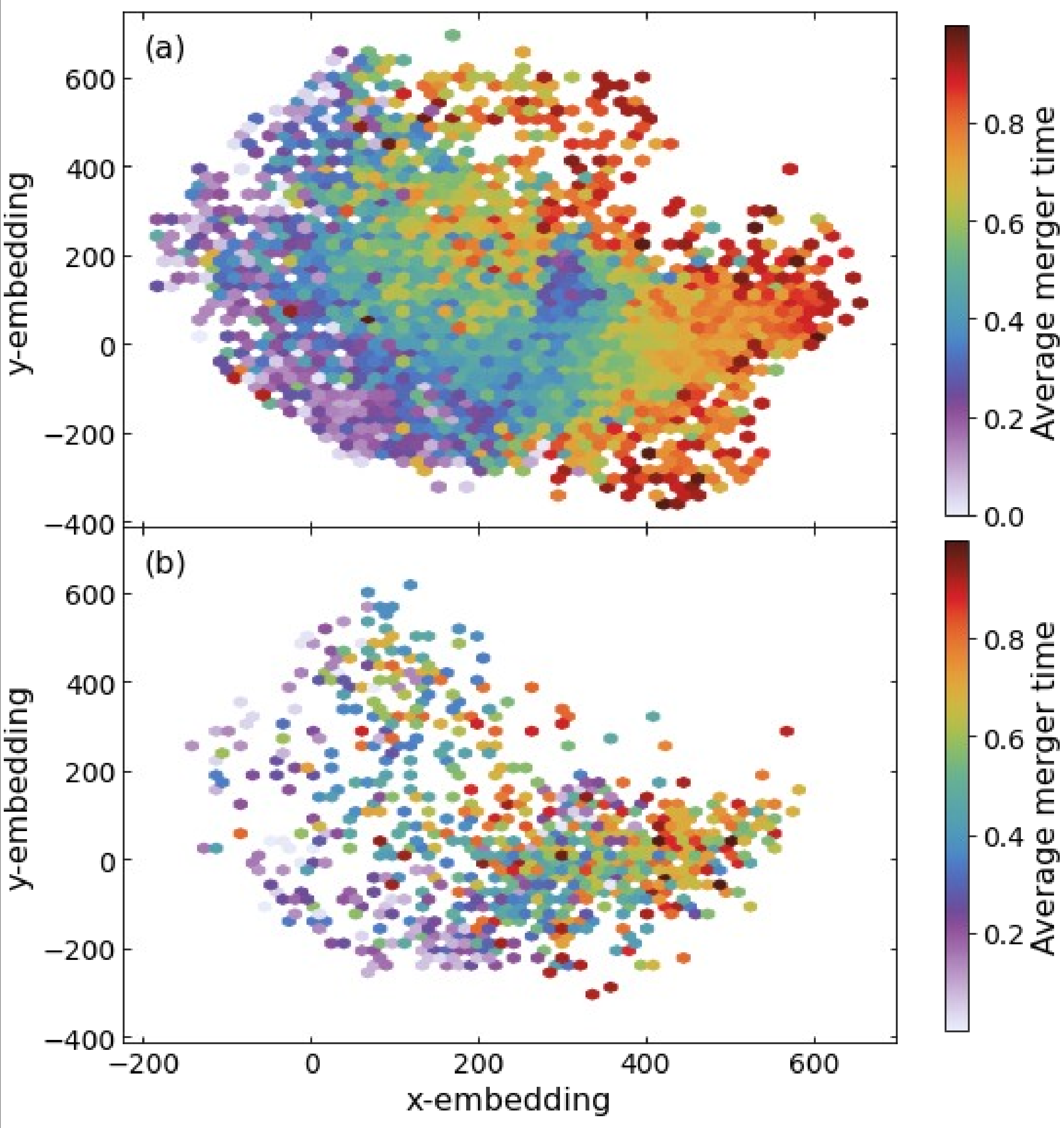
- Projects multi-dimension space to lower dimensions space linearly (supervised)
  - Maximised separation between different classes
  - Embedding dimension can be at most 1 less than the number of classes
  - Assumes each class has a Gaussian density distribution that share a covariance matrix



(a)
(b)
y-embedding
x-embedding
Average merger time
10
8
6
4
2
0
−2
−4
−6
0.2
0.4
0.6
0.8

NCA

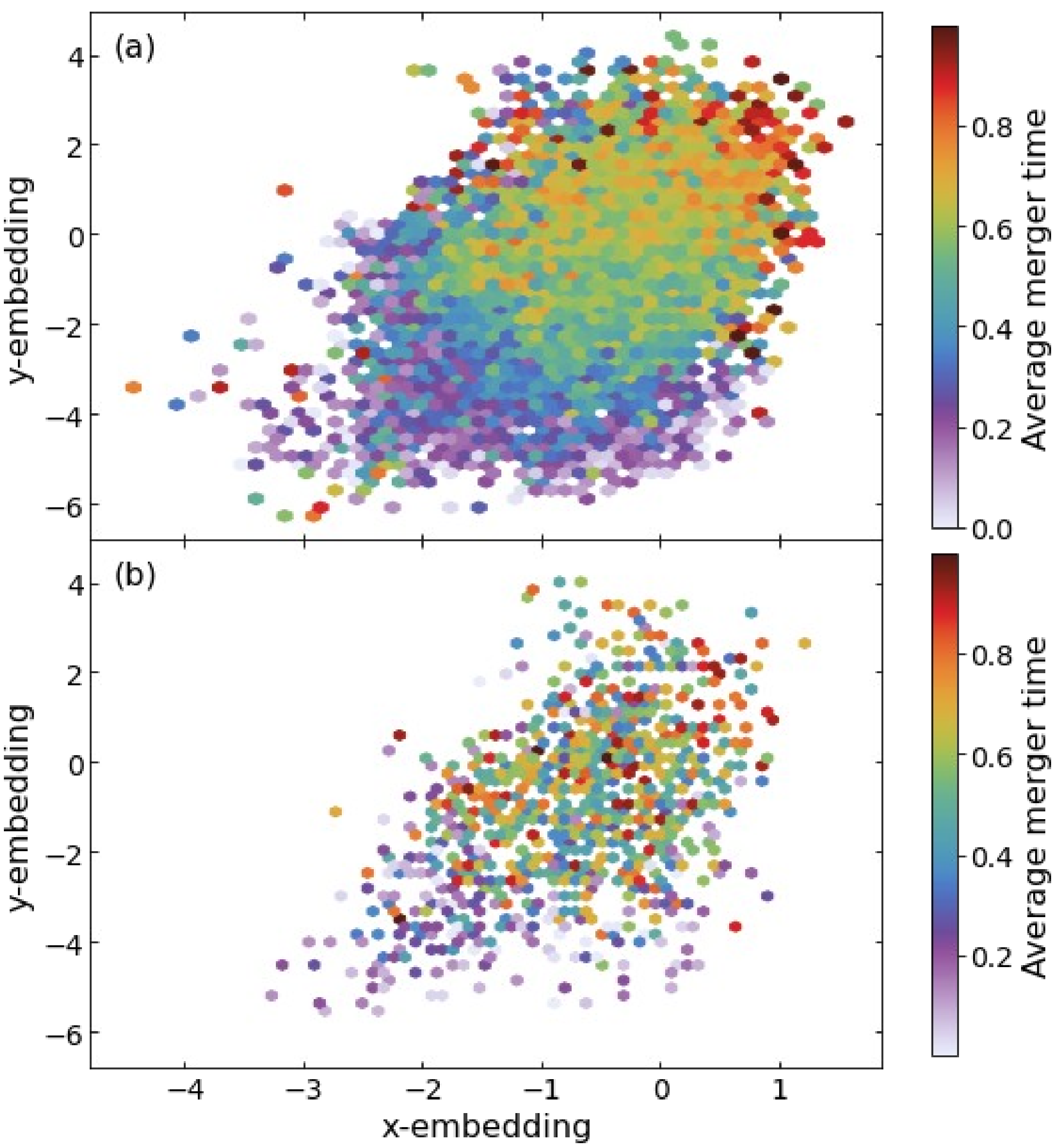
- Projects multi-dimension space to lower dimensions space (supervised)
  - Reduces the distance between objects of the same classification
  - Increases distance between objects of different classifications
  - Maximise the probability that an object is correctly classified



(a)
(b)
x-embedding
y-embedding
y-embedding
Average merger time
Average merger time

SRP

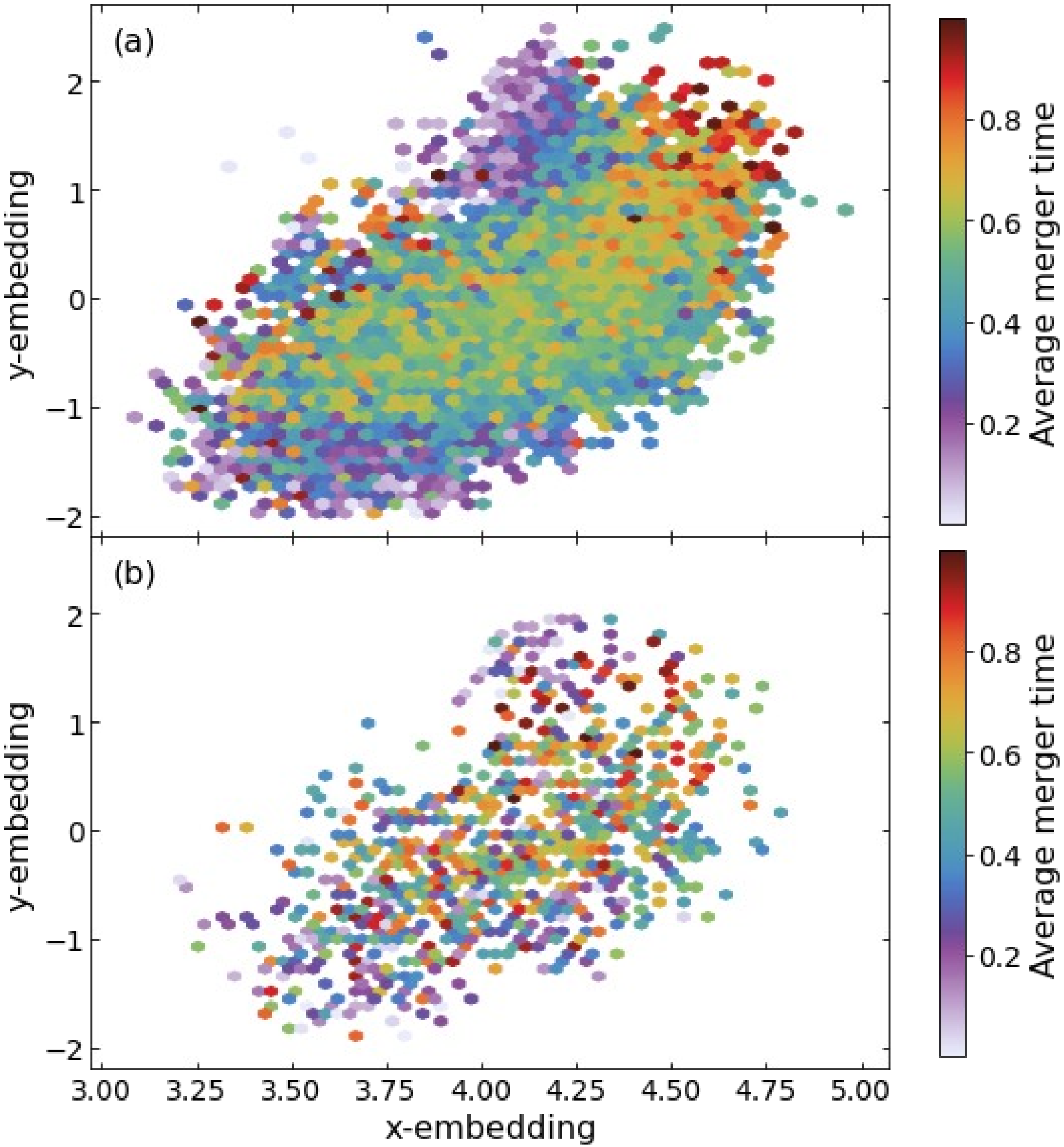
- Projects multi-dimension space to lower dimensions space randomly* (unsupervised)
  - Multiply the input space by a matrix with random values sampled from $\{-1, 0, 1\}$ with probabilities $\{1/2\sqrt{D}, 1-1/\sqrt{D}, 1/2\sqrt{D}\}$
    - D is the number of input dimensions
  - Method is much faster than Random Projection but slightly less accurate



(a)
(b)
y-embedding
x-embedding
Average merger time
4
2
0
−2
−4
−6
−4
−3
−2
−1
0
1
0.0
0.2
0.4
0.6
0.8

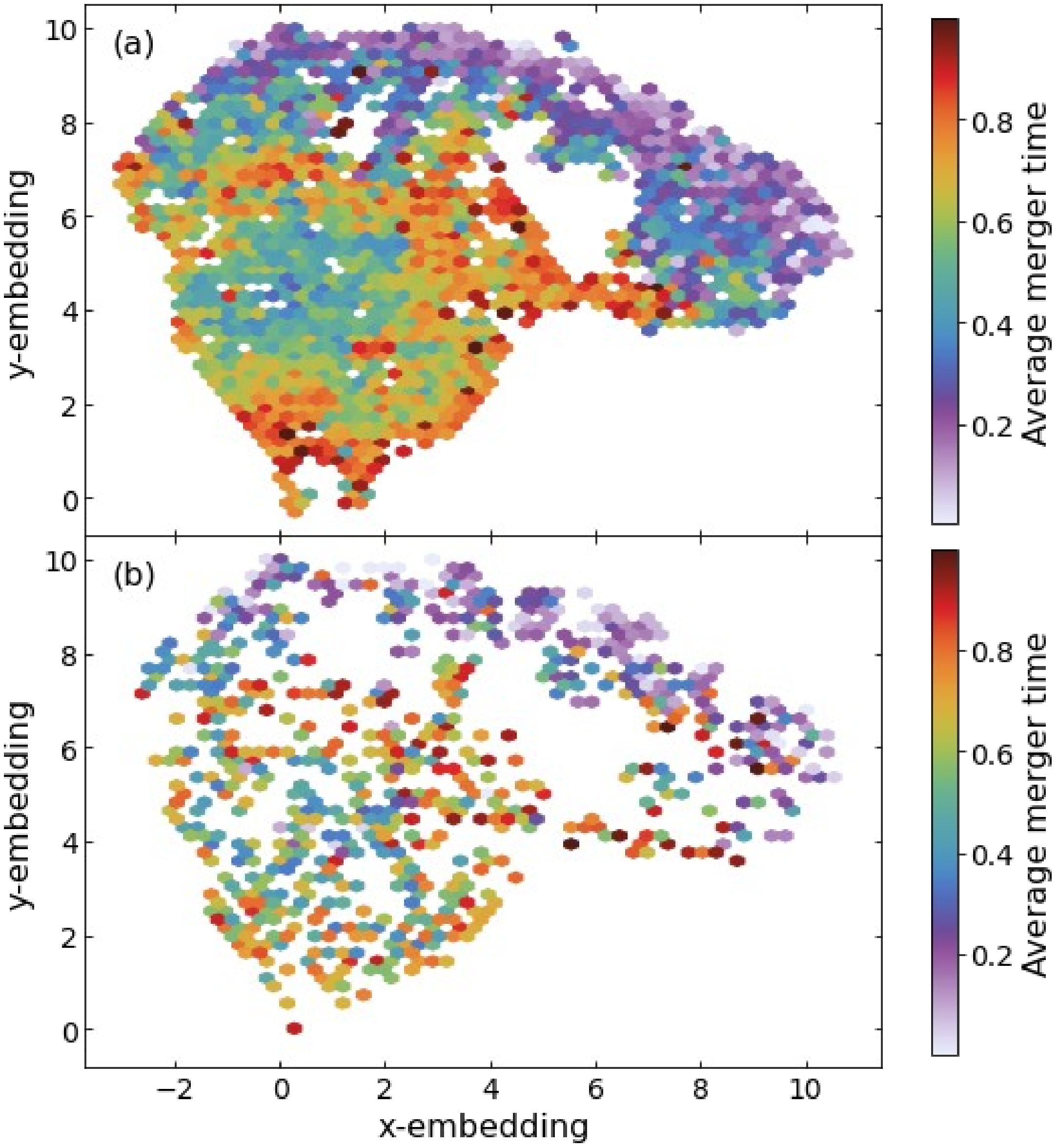
TSVD

- Projects multi-dimension space to lower dimensions space linearly (unsupervised)
  - Similar to PCA but does not require data to be centred
  - In certain cases, is identical to PCA


(a)
(b)
y-embedding
x-embedding
Average merger time
0.2
0.4
0.6
0.8
−2
−1
0
1
2
3.00
3.25
3.50
3.75
4.00
4.25
4.50
4.75
5.00

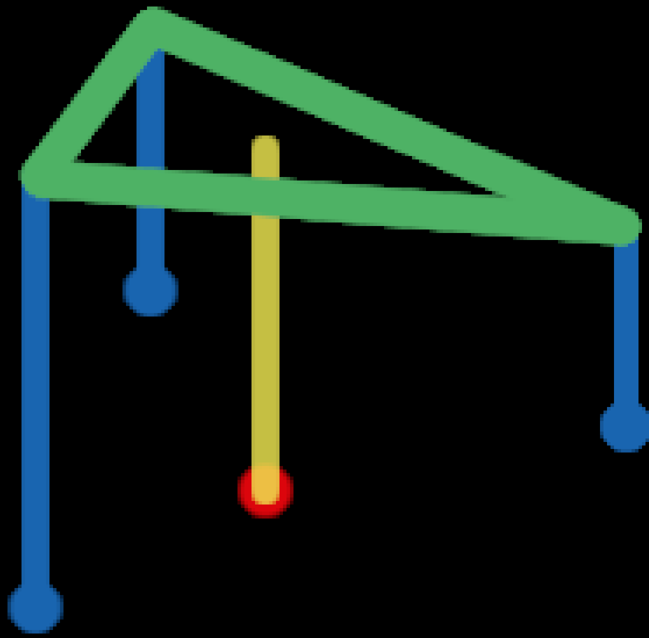
UMAP

- Projects multi-dimension space into lower dimensions in a non-linear way (unsupervised)
  - Assumes data is uniformly distributed on an Riemannian manifold, the Riemannian metric is (approximately) locally constant, and the manifold is locally connected
  - Separation between nearby points in projection maintain their high-dimension separation



(a)
(b)
y-embedding
x-embedding
Average merger time
0.2
0.4
0.6
0.8
−2
0
2
4
6
8
10

Latent Space Time



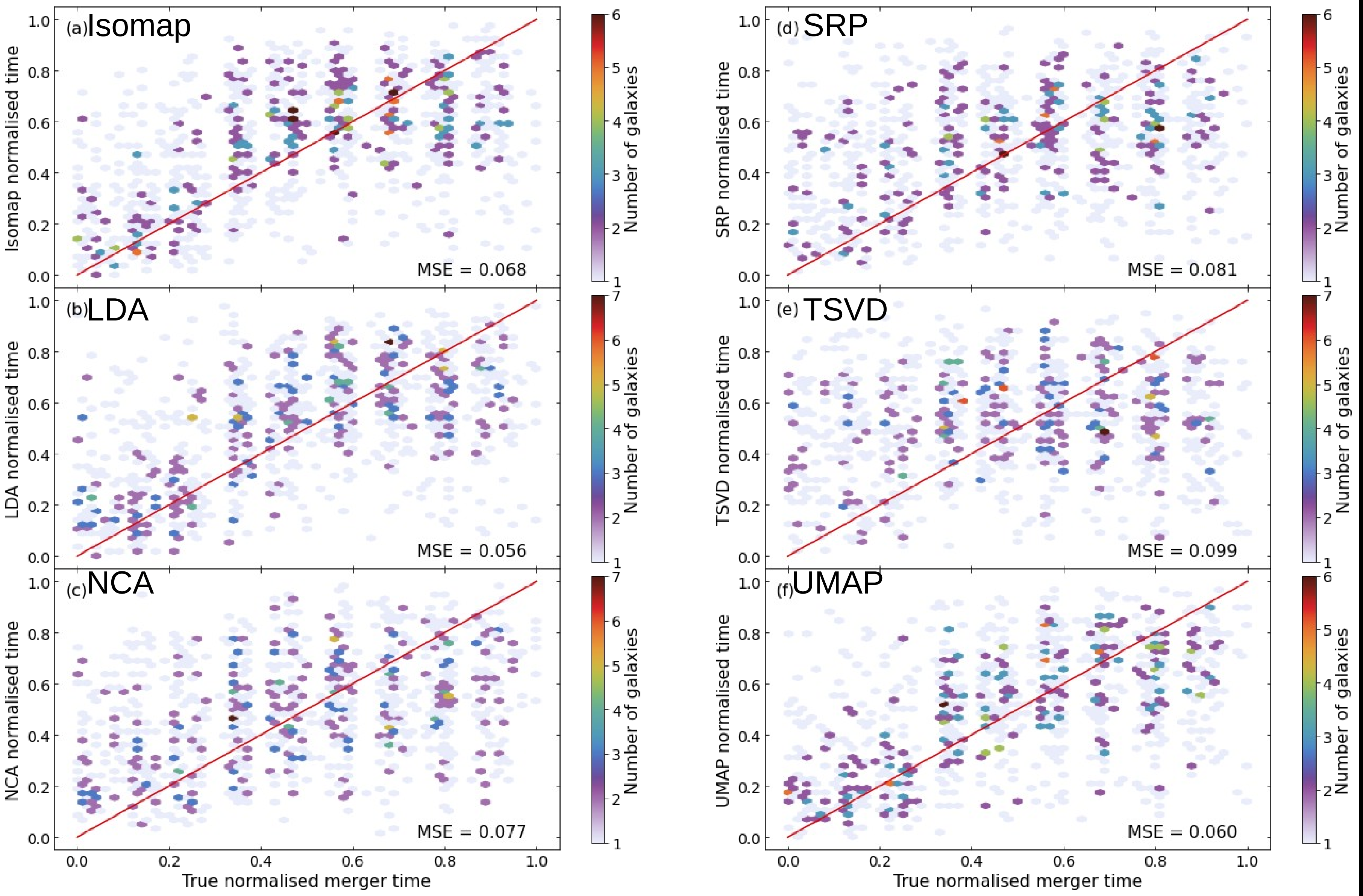
william.pearson@ncbj.gov.pl

Results

Embedding	Validation	Validation error	
	MSE	Mean [a]	Median [a]
Isomap	0.068	291	231
LDA	0.056	283	226
NCA	0.077	332	277
SRP	0.081	338	287
TSVD	0.099	378	318
UMAP	0.060	282	223

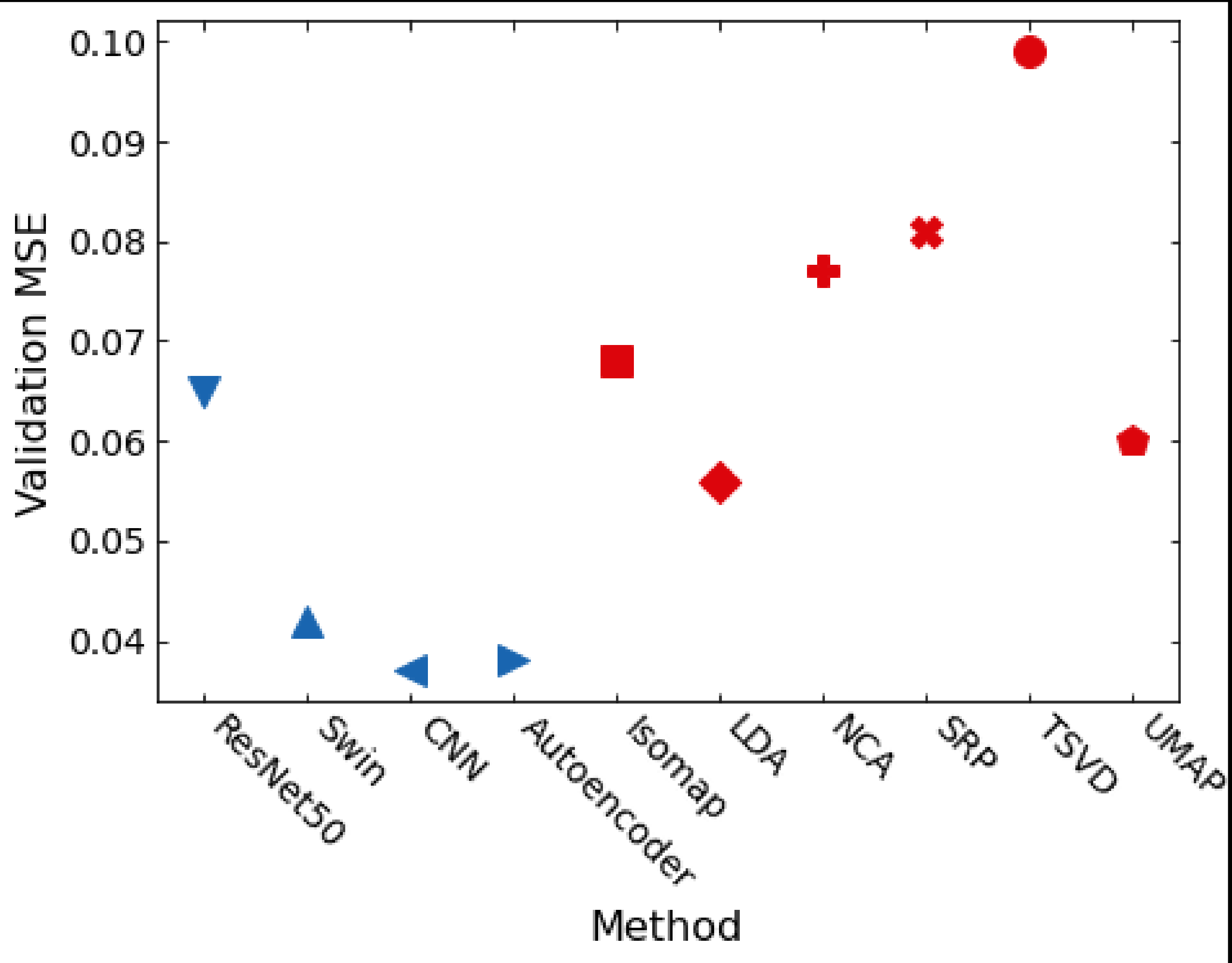
Notes. [a] Values in Myr.

william.pearson@ncbj.gov.pl



(a) Isomap
Isomap normalised time
MSE = 0.068
(b) LDA
LDA normalised time
MSE = 0.056
(c) NCA
NCA normalised time
MSE = 0.077
(d) SRP
SRP normalised time
MSE = 0.081
(e) TSVD
TSVD normalised time
MSE = 0.099
(f) UMAP
UMAP normalised time
MSE = 0.060
Number of galaxies
True normalised merger time

Results



Validation MSE
0.10
0.09
0.08
0.07
0.06
0.05
0.04
ResNet50
Swin
CNN
Autoencoder
Isomap
LDA
NCA
SRP
TSVD
UMAP
Method

Summary

- Used machine learning methods to find time before/after a merger event in simulations
- CNN, Autoencoder, ResNet, Swin Transformer
- Isomap, LDS, NCA, SRP, TSVD, UMAP
- CNN worked the best, ResNet was useless
- Could not recover missing data from Latent Space

william.pearson@ncbj.gov.pl

- Baugh, C. M, 2006, Reports on Progress in Physics, Volume 69, Issue 12, pp. 3101-3156
- Goldberger, J., Hinton, G. E., Roweis, S., & Salakhutdinov, R. R. 2004, in Advances in Neural Information Processing Systems, eds. L. Saul, Y. Weiss, & L. Bottou (MIT Press), 17
- Halko, N., Martinsson, P. G., & Tropp, J. A. 2011, SIAM Rev., 53, 217
- He, K., Zhang, X., Ren, S., & Sun, J. 2016, in 2016 IEEEConference on Computer Vision and Pattern Recognition (CVPR), 770
- Li, P., Hastie, T. J., & Church, K. W. 2006, in Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’06 (New York, USA: Association for Computing Machinery), 287
- Liu, Z., Lin, Y., Cao, Y., et al. 2021, in Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)
- McInnes, L., Healy, J., & Melville, J. 2018, arXiv e-prints [arXiv:1802.03426]
- Nelson, D., Springel, V., Pillepich, A., et al. 2019, Comput. Astrophys. Cosmol., 6, 2
- Tenenbaum, J. B., de Silva, V., & Langford, J. C. 2000, Science, 290, 2319

william.pearson@ncbj.gov.pl