Understanding LOL projection method

Paper: Supervised dimension reduction method in big data (Option 1)

July 25, 2024

Table of contents

Study objective

Real life problem

- Small sample size with large number of dimensions (n << p)
- Lack of interpretable supervised dimensionality reduction method

Purpose

- To explore a new dimension reduction technique introduced by Vogelstein et al. for handling big data in neuroimaging.

Aim

High accuracy

Timely and resource efficiency

Literature review

Supervised dimensionality reduction for big data

Joshua T. Vogelstein [1,2], Eric W. Bridgeford[1,2], Minh Tang[1], Da Zheng[1], Christopher Douville[1], Randal Burns[1] & Mauro Maggioni[1]

nature COMMUNICATIONS

- Applied LOL as supervised feature reduction (FR) and LDA as classifier on brain MRI
- Compared LOL performance with various FR methods (PCA, RP, LDA, QOQ, CCA, etc.)
- n > 800
- Each brain volume ~ 150mil dimensions. Each 3D image ~ (193, 256, 256)
- Each dataset 42 ~ 400 samples (60 ~ 600 Gb)
- Authors developed LOL package themselves (R)
- 1 TB RAM

LOL Overview

Linear Optimal Low-Rank (LOL) Projection Method

WHAT		• To find the best projection matrix A that projects high-dimensional data into a lower-dimensional subspace, facilitating effective classification. • Combines the strengths of LDA and PCA
WHY		• To find the best projection matrix that reduces data dimensionality while preserving the most significant class-discriminative information. • Address limitations of PCA and LDA
HOW		• Used in high-dimensional biomedical data for effective classification and dimensionality reduction.

Steps in LOL Algorithm

LOL Algorithm

1. Compute Class-Conditional Means (μ_1, μ_2)

$$\mu_c = \frac{1}{n_c} \sum_{i:y_i=c} x_i.$$

2. Class Priors $\pi_c = \frac{n_c}{n}$

πc : Prior probability of class c
n is the total number of samples.

3. Sorting Class Means

Arrange the class means in descending order of class priors:

$$\pi_{(1)} > \pi_{(2)} > \cdots > \pi_{(C)}$$

4. Means Differences

For each class, compute the mean difference vector

$\delta_i = \mu_1 - \mu_i$ for i = 2, 3, …., C

The mean difference matrix

$\delta = [\delta_2, \delta_3 \ldots \delta_C]$ has rank C-1

5. Normalize δi

$$\tilde{\delta}_j = \frac{\delta_j}{||\delta_j||}.$$

Steps in LOL algorithm

Continued.

6. Compute Covariance Matrix

Calculate the overall mean μ:

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Compute the centered data matrix:

$$X_{\text{centered}} = X - \mu$$

Compute the covariance matrix Σ:

$$\Sigma = \frac{1}{n-1} X_{\text{centered}}^T X_{\text{centered}}$$

7. Eigen Decomposition

$$\Sigma = U \Lambda U^T$$

U contains the eigenvectors
λ is the diagonal matrix of eigenvalues.

8. Construct the LOL Projection Matrix:

$$A_{\text{LOL}} = [\tilde{\delta}, U_{d-(C-1)}]$$

9. Project the Data

$$Z = A_{\text{LOL}} X$$

Application & Workflow

Data Collection

The Autism Brain Imaging Data Exchange (ABIDE) aggregated data from 16 international imaging sites and shared it publicly via neurodata.io

~700 patients (50-50 case-control)

- **01** Barrow Neurological Institute (BNI)
- **02** Erasmus University medical Centre Rotterdam (EMC)
- **03** ETH Zürich (ETH)
- **04** Georgetown University (GU)
- **05** Indiana University (IU)
- **06** Institut Pasteur and Robert Debré Hospital (IP)
- **07** Katholieke Universiteit Leuven (KU)
- **08** Kennedy Krieger Institute (KKI)
- **09** NYU Langone Medical Center (NYU_1)
- **10** NYU Langone Medical Center (NYU_2)

Workflow

Data Collection

The Autism Brain Imaging Data Exchange (ABIDE) aggregated data from 16 international imaging sites and shared it publicly via neurodata.io

Sample Size

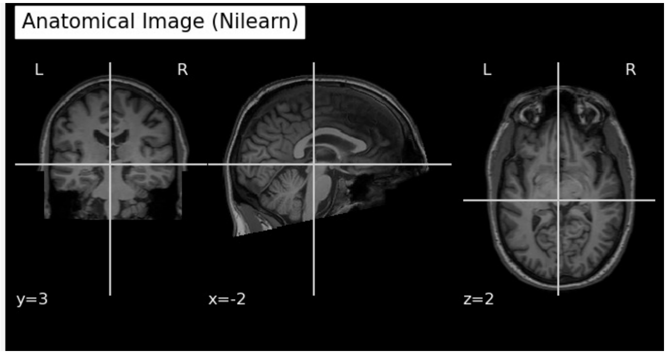
Total: N = 58 (age range 18-64 years)

Autism Spectrum Disorders (ASD): n = 13 (18-25 years), n = 16 (40-62 years)

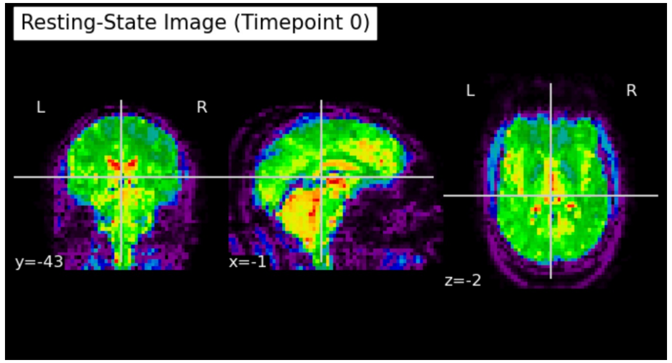
Typical Controls (TC): n = 10 (18-25 years), n = 19 (40-64 years)

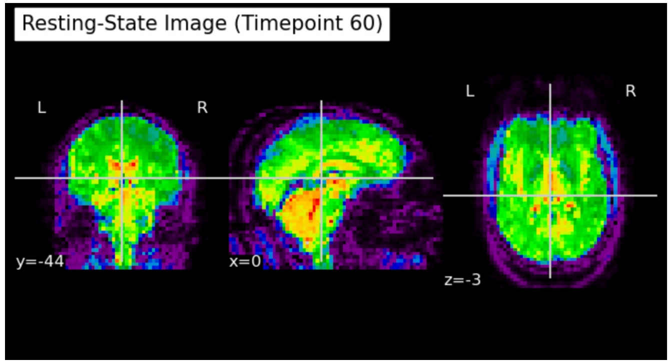
Pre-processing

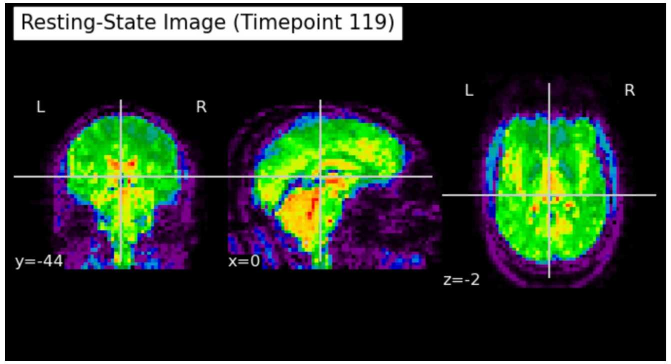
Anatomical Image



Time-series of Resting state Image



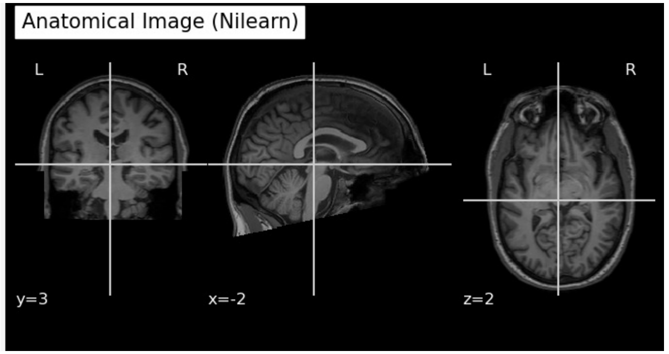




Workflow

Pre-processing

Resize and Flatten image



3D input
193x256x256

23	34	78
90	101	130
150	220	255

Resized image
64x64x64

Flattened 1D array
262,144 dimensions

90		150		255			23	...	...

Workflow

Modeling

Feature reduction
- Linear Optimal Low-Rank projection (LOL)
- Linear Discriminant Analysis (LDA)
- Random Projection (RP)
- Principal Component Analysis (PCA)

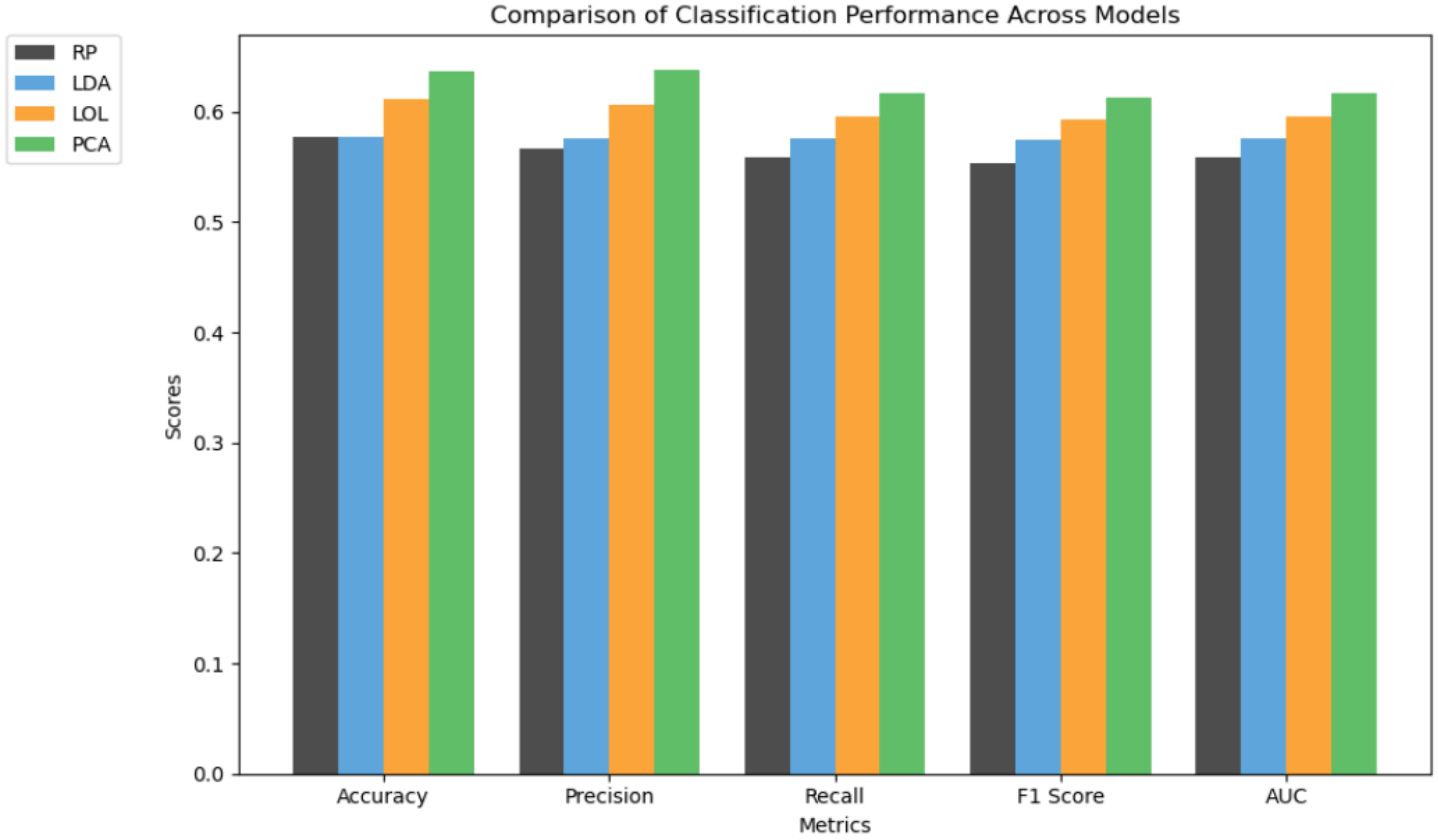
Model
- Random Forest Classifier
- LDA Classifier
- CNN model(1)

Performance comparison
- AUC
- Cohen kappa
- Time
- Accuracy

1) CNN model using resized 3D input with full features

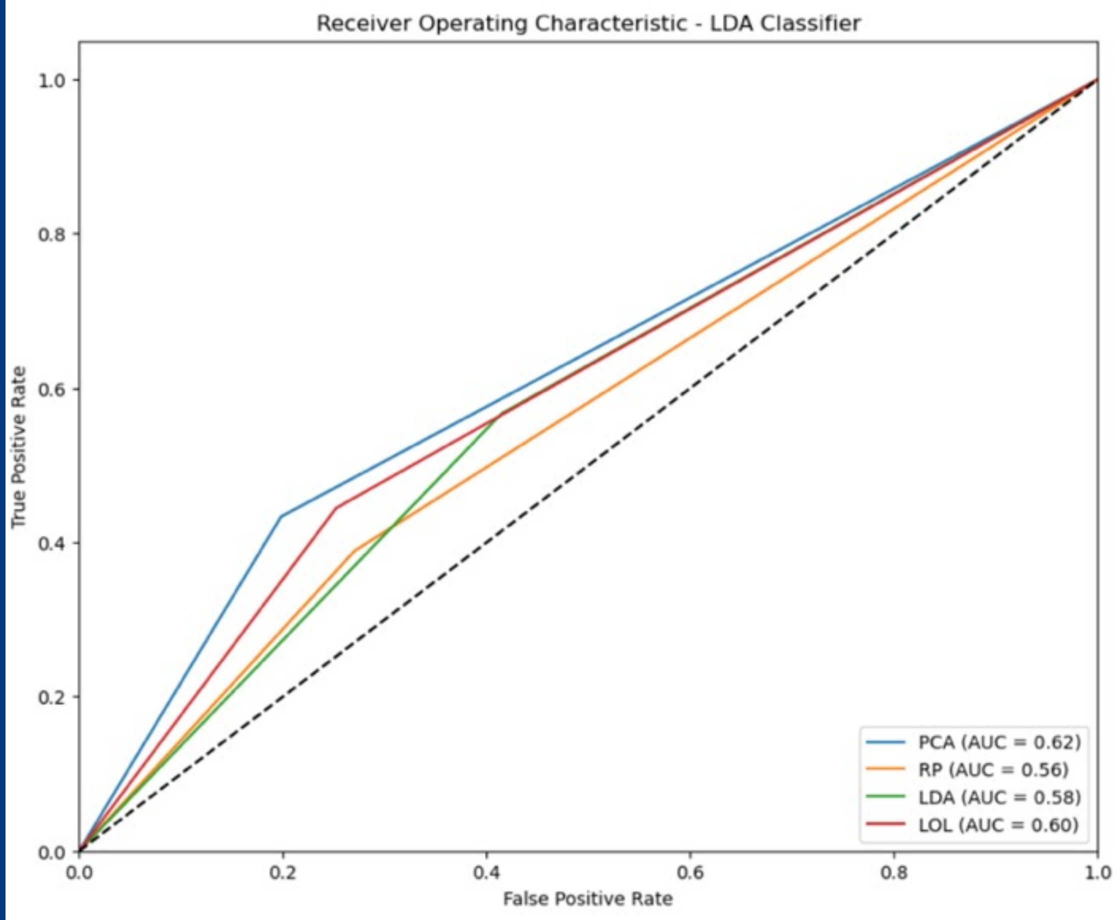
Performance

Unlike the paper, PCA showed best results

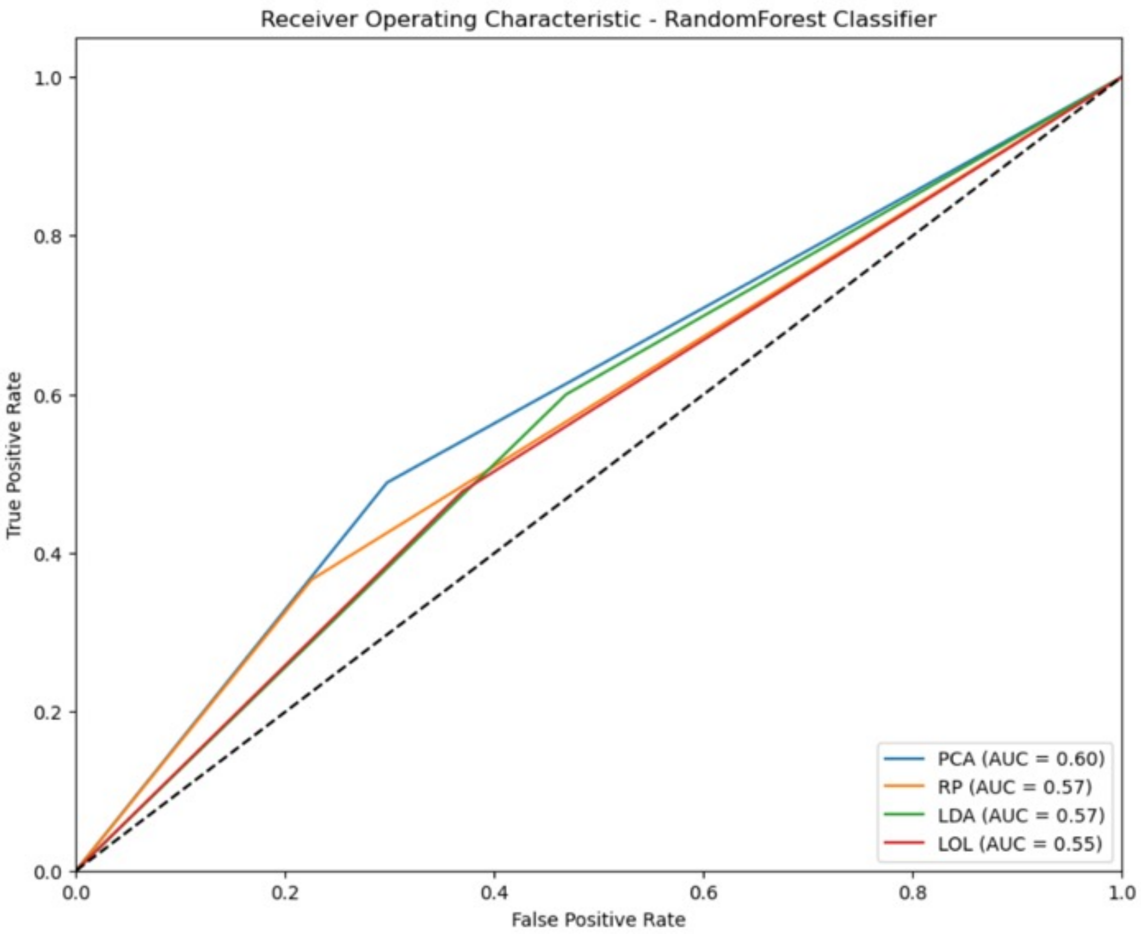


Highest AUC for PCA + LDA

LDA classifier



RF classifier



Cohen's kappa value

Cohen Kappa Statistic

Measure The Performance of Classification Models

Kappa Statistic

$$k = \frac{2 * (TP * TN - FN * FP)}{(TP + FP) * (FP + TN) + (TP + FN) * (FN + TN)}$$

Assess the level of agreement between an actual and predicted

Predicted (rater 2) \ Actual (rater 1)	YES	NO	
YES	45 (**TP**)	15 (**FN**)	60
NO	25 (**FP**)	15 (**TN**)	40
	70	30	

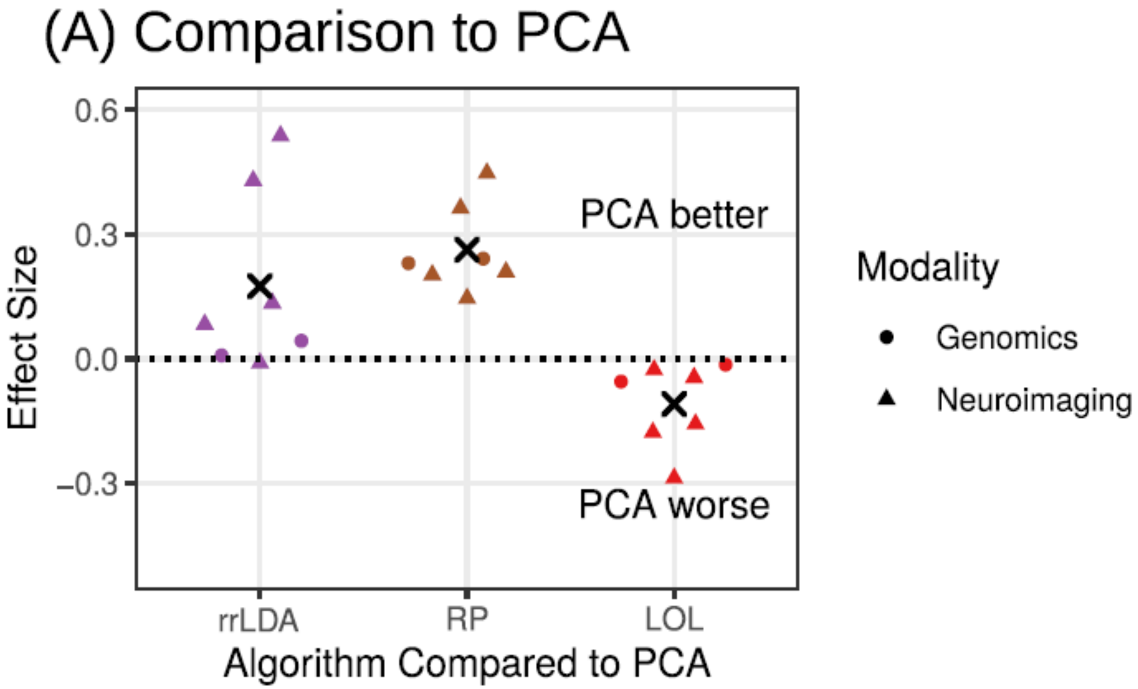
$$k = \frac{2 * (45 * 15 - 15 * 25)}{(45 + 25) * (25 + 15) + (45 + 15) * (15 + 15)} = 0.13 \ (13\%)$$

Kappa Score Interpretation

Kappa	**Agreement**
<0	Less than chance agreement
0.01-0.20	Slight agreement
0.21-0.40	Fair agreement
0.41-0.60	Moderate agreement
0.61-0.80	Substantial agreement
0.81-0.99	Almost perfect agreement

Discussion & Result

Cohen's kappa value of PCA remains the best over RP, LOL and LDA



3D CNN deep learning showed poorer classification results

Model: "sequential"

Layer (type)	Output Shape	Param #
conv3d (Conv3D)	(None, 126, 126, 126, 32)	896
max_pooling3d (MaxPooling3D)	(None, 63, 63, 63, 32)	0
batch_normalization (BatchNormalization)	(None, 63, 63, 63, 32)	128
conv3d_1 (Conv3D)	(None, 61, 61, 61, 64)	55,360
max_pooling3d_1 (MaxPooling3D)	(None, 30, 30, 30, 64)	0
batch_normalization_1 (BatchNormalization)	(None, 30, 30, 30, 64)	256
conv3d_2 (Conv3D)	(None, 28, 28, 28, 128)	221,312
max_pooling3d_2 (MaxPooling3D)	(None, 14, 14, 14, 128)	0
batch_normalization_2 (BatchNormalization)	(None, 14, 14, 14, 128)	512
flatten (Flatten)	(None, 351232)	0
dense (Dense)	(None, 512)	179,831,296
dropout (Dropout)	(None, 512)	0
dense_1 (Dense)	(None, 1)	513

Total params: 180,110,273 (687.07 MB)

Trainable params: 180,109,825 (687.06 MB)

Non-trainable params: 448 (1.75 KB)

```
Epoch 1/10
3/3 ━━━━━━━━━━━━━━━━━━━━ 139s 49s/step - accuracy: 0.4108 - loss: 21.6151 - val_accuracy: 0.6176 - val_loss: 0.7170
Epoch 2/10
3/3 ━━━━━━━━━━━━━━━━━━━━ 125s 38s/step - accuracy: 0.5516 - loss: 10.7422 - val_accuracy: 0.4412 - val_loss: 0.7315
Epoch 3/10
3/3 ━━━━━━━━━━━━━━━━━━━━ 124s 40s/step - accuracy: 0.6757 - loss: 5.4900 - val_accuracy: 0.4412 - val_loss: 0.7567
Epoch 4/10
3/3 ━━━━━━━━━━━━━━━━━━━━ 94s 25s/step - accuracy: 0.5722 - loss: 6.2629 - val_accuracy: 0.4412 - val_loss: 0.7154
Epoch 5/10
3/3 ━━━━━━━━━━━━━━━━━━━━ 107s 34s/step - accuracy: 0.5853 - loss: 8.0299 - val_accuracy: 0.4118 - val_loss: 0.6953
Epoch 6/10
3/3 ━━━━━━━━━━━━━━━━━━━━ 119s 40s/step - accuracy: 0.6579 - loss: 6.4935 - val_accuracy: 0.4412 - val_loss: 0.7031
Epoch 7/10
3/3 ━━━━━━━━━━━━━━━━━━━━ 139s 42s/step - accuracy: 0.6475 - loss: 4.9050 - val_accuracy: 0.5588 - val_loss: 0.9233
Epoch 8/10
3/3 ━━━━━━━━━━━━━━━━━━━━ 105s 33s/step - accuracy: 0.7443 - loss: 2.9353 - val_accuracy: 0.5588 - val_loss: 0.9612
Epoch 9/10
3/3 ━━━━━━━━━━━━━━━━━━━━ 109s 33s/step - accuracy: 0.6771 - loss: 1.5363 - val_accuracy: 0.5588 - val_loss: 1.1903
Epoch 10/10
3/3 ━━━━━━━━━━━━━━━━━━━━ 105s 31s/step - accuracy: 0.8204 - loss: 1.4579 - val_accuracy: 0.5588 - val_loss: 1.4481
2/2 ━━━━━━━━━━━━━━━━━━━━ 4s 235ms/step - accuracy: 0.5600 - loss: 1.4441
Validation loss: 1.4481221437454224, Validation accuracy: 0.5588235259056091
```

Discussion & Result

Processing time

Method	No. of components	Run time
RP	50	2.3s
LDA	1	18.8s
LOL	40	26s
PCA	77	5s
CNN	2,097,152	18m25s

Recommended system configuration:
CPU: 12th Gen Intel(R) Core (TM) i7-12800H 2.40 GHz
RAM: 32GB

Filename	Number of observations
Barrow Neurological Institute (BNI)	58
Erasmus University Medical Center Rotterdam (EMC)	54
ETH Zürich (ETH)	37
Georgetown University (GU)	106
Indiana University (IU)	40
Institut Pasteur and Robert Debré Hospital (IP)	56
Katholieke Universiteit Leuven (KU)	28
Kennedy Krieger Institute (KKI)	211
NYU Langone Medical Center: Sample 1 (NYU_1)	78
NYU Langone Medical Center: Sample 2 (NYU_2)	27
Total	**667**

Conclusion

Our findings

- ✓ Successfully replicated:
  - 3D image transformation
  - LOL as feature reduction
  - Cohen's kappa comparison

- ✓ Handle 3D images better after this project

- ✓ Contradictory results compared to the original paper might be due to the limitation of computing resources (resized)
  - Explore other methods that can handle timeseries data
  - Follow author's suggestions for future work

Thanks for listening