



DON HONORIO VENTURA STATE UNIVERSITY



DON HONORIO VENTURA STATE UNIVERSITY

Bacolor, Pampanga

College of Engineering and Architecture

Department of Electronics Engineering



***k*-Deep Autoencoder-Based Clustering of Channel Model Generated Multipath Components**

by

JIMENEZ, Ishi Jemila E.

LEONCIO, Timothy D.

MACAPAGAL, Faithlix Mark M.

MALIWAT, Jerome R.

MANALANG, Neera Mae C.

OLALIA, Jhonly A.

June, 2024



DON HONORIO VENTURA STATE UNIVERSITY



***k*-Deep Autoencoder-Based Clustering of Channel Model Generated Multipath Components**

A Thesis

Presented to the Faculty of the
Department of Electronics Engineering
College of Engineering and Architecture
Don Honorio Ventura State University

In Partial Fulfillment of the
Requirements for the Degree of
Bachelor of Science in Electronics Engineering

by

JIMENEZ, Ishi Jemila E.
LEONCIO, Timothy D.
MACAPAGAL, Faithlix Mark M.
MALIWAT, Jerome R.
MANALANG, Neera Mae C.
OLALIA, Jhonly A.

June, 2024



DON HONORIO VENTURA STATE UNIVERSITY

THESIS APPROVAL SHEET



Republic of the Philippines
DON HONORIO VENTURA TECHNOLOGICAL UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING AND ARCHITECTURE
ELECTRONICS ENGINEERING DEPARTMENT
SY 2023-2024

FINAL THESIS APPROVAL SHEET

This thesis entitled:

"k-Deep Autoencoder-Based Clustering of Channel Generated Multipath Components"

prepared and submitted by JIMENEZ, Ishi Jemila E., LEONCIO, Timothy D., MACAPAGAL, Faithlix Mark M., MALIWAT, Jerome R., MANALANG, Neera Mae C., OLALIA, Jhony A., (members) , in partial fulfilment of the requirements for the degree of Bachelor of Science in Electronics Engineering has been examined and is recommended for acceptance and approval for FINAL THESIS DEFENSE.

Engr. Emmanuel T. Trinidad
Adviser

Approved by the Committee on Oral Examination (Thesis) with a defense result of Pased at on April 29, 2024 (oral defense date).

Mr. Larie Joseph R. Lacsina
Member

Engr. Enmar T. Tuazon
Chair

Engr. Elias Francis G. Vergara
Member

Engr. Anthony S. Tolentino, MS-ECE
Capstone Project 1 Coordinator

Engr. Nilo Q. Manungag, PECE, MEP-EE
OIC Chairperson, ECE Department

Engr. Jun P. Flores, PECE, MEP-EE
Dean, College of Engineering and Architecture



DON HONORIO VENTURA STATE UNIVERSITY

2024

All Rights Reserved. No part of this publication may be reproduced, stored in an information retrieval system, or transmitted, in any form or by any means, electronic, mechanical, by photocopying, scanning, recording, or otherwise, except under the terms of the applicable law.



ACKNOWLEDGMENT

We want to extend our gratitude to our hardworking adviser, Engr. Emmanuel T. Trinidad, for sharing his valuable knowledge and expertise with us. Also, for sharing his skills on the L^AT_EX typesetting system, Python programming, and MATLAB. We appreciate and acknowledge the lessons that our adviser imparted to us. Moreover, for showing us that research can be enjoyable without any return rewards and by contributing a body of knowledge to the research community.

We want to thank the panel of examiners, Mr. Larie Joseph R. Lacsina and Engr. Elias Francis G. Vergara, chaired by Engr. Enmar T. Tuazon, for sharing and providing constructive comments to improve the thesis paper. Also, we want to express our gratitude to Dr. Lawrence Materum and Dr. Jojo Blanza for sharing their knowledge on this thesis.

To the family and friends who continually support us on the path that we enjoy, we send our deepest appreciation for financial and emotional support during thesis writing.

To DOST and CHED for extending their help through scholarships, thank you for the financial support on this thesis.

Above all, we are all grateful to the Almighty God for giving us the spiritual strength to finish this thesis with excellence.

Ishi Jemila E. Jimenez

Timothy D. Leoncio

Faithlix Mark M. Macapagal

Jerome R. Maliwat

Neera Mae C. Manalang

Jhonly A. Olalia



ABSTRACT

Channel modeling plays a significant role in designing and evaluating wireless communication systems. Geometric-Based Stochastic Channel Modelling (GBSCM) rely on the statistical distribution of multipath waves which arrive at the receiver with similar parameters and are called clusters. This study investigates the performance of k -Deep Autoencoder (k -DAE) in clustering the multipath components (MPC)s from the COST 2100 (C2CM) and International Mobile Telecommunications-2020 Channel Model (IMT-2020) channel models. Furthermore, the algorithm is modified by its activation function and optimizer, namely: Sigmoid, Rectified Linear Unit (ReLU), with Adaptive Momentum (ADAM) and Stochastic Gradient Descent (SGD) as the optimizers. The results showed that the Sigmoid-SGD configuration outperforms all the other configurations, both in C2CM and IMT-2020 datasets. The configuration increased in Jaccard Index (JI) by 0.016 at the 10th percentile, 0.019 for the median, and 0.3595 at the 90th percentile for the C2CM dataset compared to default configuration. On the other hand, the algorithm performance of the IMT-2020 data, increased in JI by 0.004, and 0.1, for the 10th, and 50th percentile, respectively.

Index Terms—clusters, multipath components, activation functions, optimizers, ReLU, sigmoid, autoencoder, algorithm.



TABLE OF CONTENTS

Thesis Approval Sheet	iv
Acknowledgment	v
Abstract	vi
Table of Contents	vii
List of Figures	xiii
List of Tables	xvii
Abbreviations	xix
Notation	xxiv
Glossary	xxv
Listings	xxvii
Chapter 1 INTRODUCTION	1
1.1 Background of the Study	2
1.2 Prior Studies	8
1.3 Problem Statement	9
1.3.1 General Problem	9
1.3.2 Specific Problems	9
1.4 Objectives and Deliverables	9
1.4.1 General Objective	10



1.4.2 Specific Objectives	10
1.4.3 Expected Deliverable	10
1.5 Significance of the Study	11
1.5.1 Technical Benefit	11
1.5.2 Social Impact	11
1.5.3 Environmental Impact	12
1.5.4 Academic Impact	12
1.6 Assumptions, Scope and Delimitations	12
1.6.1 Assumptions	12
1.6.2 Scope	12
1.6.3 Delimitation	13
1.7 Description and Methodology	13
1.7.1 Channel Generated Models	14
1.7.1.1 COST 2100 (C2CM)	14
1.7.1.2 International Mobile Telecommunications- 2020 (IMT-2020)	14
1.7.2 Activation Functions	15
1.8 Overview	16
Chapter 2 LITERATURE REVIEW	17
2.1 Existing Works	18
2.1.1 Existing Works in Clustering MPCs	18
2.1.2 Existing Works on Autoencoders	25
2.2 Lacking in the Approaches	32
2.3 Summary	32
Chapter 3 THEORETICAL CONSIDERATIONS	34
3.1 Radio Wave Propagation	35



3.2 MIMO Wireless Systems	35
3.2.1 Multipath Propagation	36
3.2.2 Channel Modelling of Wireless Systems	37
3.3 Clustering	38
3.3.1 Partition Method	39
3.4 Deep Learning	39
3.4.1 Activation Functions	40
3.4.2 Optimizer	40
3.5 Cluster Validity Index	40
3.6 Summary	41
Chapter 4 DESIGN CONSIDERATIONS	42
4.1 Channel Model Datasets	43
4.1.1 COST 2100 Channel Model	43
4.1.2 IMT-2020 Channel Model	44
4.2 k-Deep Autoencoder Clustering	45
4.2.1 Clustering Algorithm	45
4.3 Clustering Validation Index	46
4.3.1 Jaccard Index	48
4.3.2 Normalize Mutual Index	49
4.3.3 Adjusted Rand Index	49
4.3.4 Clustering Accuracy	50
4.4 Optimizers	51
4.4.1 Adaptive Momentum (ADAM)	51
4.4.2 Stochastic Gradient Descent (SGD)	51
4.5 Activation Functions	51
4.5.1 Rectified Linear Unit (ReLU)	52
4.5.2 Sigmoid	53



4.6	Python	54
4.6.1	Python IDE	54
4.6.2	Versions and Libraries	55
4.6.3	Flowchart of the Code	56
4.7	Summary	57
Chapter 5 METHODOLOGY		58
5.1	Implementation	60
5.1.1	Dataset Preparation	60
5.1.2	Code Modification	60
5.1.3	Optimizer	61
5.1.4	Activation Function	62
5.2	Evaluation	62
5.2.1	Evaluation of Results using Jaccard index	62
5.2.2	Statistical Treatment of Data	62
5.3	Summary	62
Chapter 6 RESULTS AND DISCUSSION		64
6.1	Implementation of k-DAE algorithm on C2CM and IMT-2020	67
6.1.1	Incorporation of required libraries and modules	67
6.1.2	Implementation of Channel Model Datasets	69
6.1.3	Implementation of Activation Functions and Optimizers	72
6.2	Clustering Accuracy of Modified k-DAE	74
6.2.1	Box Plots of CVIs of Different Configurations in C2CM	74
6.2.2	Box Plots of CVIs of Different Configurations in IMT-2020	78
6.2.3	Means and Standard Deviations of CVIs in C2CM	82
6.2.4	Means and Standard Deviations of CVIs in IMT-2020	84
6.3	Statistical Comparison of different Activation functions and Optimizers	85



6.3.1 Means and Standard Deviations of JIs in C2CM	86
6.3.2 Means and Standard Deviations of JIs in IMT-2020	87
6.3.3 ANOVA of JIs in C2CM	88
6.3.4 ANOVA of JIs in IMT-2020	91
6.3.5 Histogram on distribution of JI in C2CM and IMT-2020	95
6.3.6 ECDF of JIs in C2CM	99
6.3.7 ECDF of JIs in IMT-2020	102
6.4 Reconstruction Error	106
6.4.1 Correlations of CVIs of Different Configurations in C2CM . .	106
6.4.2 Correlations of CVIs of Different Configurations in IMT-2020	111
6.4.3 Correlation of JIs and REs in C2CM and IMT-2020	117
6.5 Summary	120
Chapter 7 CONCLUSIONS, RECOMMENDATIONS, AND FUTURE DIRECTIVES	122
7.1 Summary of Findings	123
7.2 Conclusion	123
7.3 Recommendations	124
7.4 Future Prospects	124
References	125
Appendix A THESIS WEEKLY REPORT	131
Appendix B ECE ADVISER ACCEPTANCE FORM	150
Appendix C MEMORANDUM OF UNDERSTANDING BETWEEN THE ECE THESIS ADVISEE AND THESIS ADVISER	152
Appendix D CERTIFICATE OF PERCENTAGE COMPLETION	154



Appendix E ANTI-PLAGIARISM FORM	156
Appendix F ORAL DEFENSE RECOMMENDATION SHEET	158
Appendix G THESIS PROPOSAL APPROVAL SHEET	160
Appendix H FINAL DEFENSE RECOMMENDATION SHEET	162
Appendix I GANTT-CHART	164
Appendix J MEMORANDUM OF AGREEMENT BETWEEN THESIS MEMBERS AND THESIS PANEL	166
Appendix K REVISION TO FINAL	168
Appendix L ANOVA OF COST 2100 USING JAMOVI	172
Appendix M ANOVA OF IMT-2020 USING JAMOVI	176
Appendix N DATASETS	180
Appendix O PROGRAM LISTING	183
00.1 Main Runfile	184
00.2 Utilities	188
00.3 k-DAE	194
00.4 Autoencoder	198
Appendix P APPROVAL SHEET	201
Appendix Q VITA	203



LIST OF FIGURES

1.1	Illustration of 3D MIMO Channel Model	3
1.2	Illustrations of clustering procedure for channel modeling	5
1.3	Illustration of Autoencoder Algorithm	7
1.4	Block Diagram of Methodology of the Study	13
3.1	Basic wireless propagation phenomena in confined environment	36
4.1	k -DAE Clustering Algorithm	46
4.2	Rectifier Linear Unit function	52
4.3	The demonstration of a Sigmoid Activation Function	53
4.4	Flowchart of the k -DAE code	56
5.1	Block Diagram of Methodology of the Study	60
6.1	Boxplots of C2CM's First Four Scenarios	75
6.2	Boxplots of C2CM's Last Four Scenarios	76
6.3	Boxplots of C2CM's Indoor, Outdoor, and All Scenarios	77
6.4	Boxplots of IMT-2020's First Four Scenarios	79
6.5	Boxplots of IMT-2020's Last Four Scenarios	80
6.6	Boxplots of IMT-2020's Indoor, Outdoor, and All Scenarios	81
6.7	C2CM ANOVA of All Indoor Scenarios	88
6.8	C2CM ANOVA of All Outdoor Scenarios	89
6.9	C2CM ANOVA of All Scenarios	89
6.10	IMT-2020 ANOVA of All Indoor Scenarios	92
6.11	IMT-2020 ANOVA of All Outdoor Scenarios	92
6.12	IMT-2020 ANOVA of All Scenarios	93



6.13 C2CM- Histogram of Jaccard Indices of All Scenarios	97
6.14 IMT-2020- Histogram of Jaccard Indices of All Scenarios	98
6.15 C2CM ECDF Comparison for All Indoor Scenarios	99
6.16 C2CM ECDF Comparison for All Outdoor Scenarios	99
6.17 C2CM ECDF Comparison for All Scenarios	100
6.18 IMT-2020 ECDF Comparison for All Indoor Scenarios	103
6.19 IMT-2020 ECDF Comparison for All Outdoor Scenarios	103
6.20 IMT-2020 ECDF Comparison for All Scenarios	104
6.21 C2CM-Correlation Plots of CVIs and Reconstruction Error Results in ReLU-ReLU-ADAM All Scenarios	107
6.22 C2CM-Correlation Plots of CVIs and Reconstruction Error Results in Sigmoid-Sigmoid-ADAM All Scenarios	108
6.23 C2CM-Correlation Plots of CVIs and Reconstruction Error Results in ReLU-ReLU-SGD All Scenarios	109
6.24 C2CM-Correlation Plots of CVIs and Reconstruction Error Results in Sigmoid-Sigmoid-SGD All Scenarios	110
6.25 IMT-2020-Correlation Plots of CVIs and Reconstruction Error Results in eLU-eLU-ADAM All Scenarios	112
6.26 IMT-2020-Correlation Plots of CVIs and Reconstruction Error Results in ReLU-ReLU-ADAM All Scenarios	113
6.27 IMT-2020-Correlation Plots of CVIs and Reconstruction Error Results in Sigmoid-Sigmoid-SGD All Scenarios	114
6.28 IMT-2020-Correlation Plots of CVIs and Reconstruction Error Results in ReLU-ReLU-SGD All Scenarios	115
6.29 IMT-2020-Correlation Plots of CVIs and Reconstruction Error Results in Sigmoid-Sigmoid-SGD All Scenarios	116
A.1 Weekly Report 1	132



A.2 Weekly Report 2	133
A.3 Weekly Report 3	134
A.4 Weekly Report 4	135
A.5 Weekly Report 5	136
A.6 Weekly Report 6	137
A.7 Weekly Report 7	138
A.8 Weekly Report 8	139
A.9 Weekly Report 10	140
A.10 Weekly Report 11	141
A.11 Weekly Report 12	142
A.12 Weekly Report 13	143
A.13 Weekly Report 14	144
A.14 Weekly Report 15	145
A.15 Weekly Report 16	146
A.16 Weekly Report 17	147
A.17 Weekly Report 18	148
A.18 Weekly Report 21	149
B.1 EcE Adviser Acceptance Form	151
C.1 MEMORANDUM OF UNDERSTANDING BETWEEN THE EcE THESIS ADVISEE AND THESIS ADVISER	153
D.1 Certificate of Percentage Completion	155
E.1 Anti-Plagiarism Form	157
F.1 ORAL DEFENSE RECOMMENDATION SHEET	159
G.1 Thesis Proposal Approval Sheet	161



H.1	Final Defense Recommendation Sheet	163
I.1	Gantt Chart	165
J.1	Memorandum of Agreement between Thesis Members and Thesis Panel	167
L.1	ANOVA of COST 2100 All Indoor Scenarios Using Jamovi	173
L.2	ANOVA of COST 2100 All Outdoor Scenarios Using Jamovi	174
L.3	ANOVA of COST 2100 All Scenarios Using Jamovi	175
M.1	ANOVA of IMT-2020 All Indoor Scenarios Using Jamovi	177
M.2	ANOVA of IMT-2020 All Outdoor Scenarios Using Jamovi	178
M.3	ANOVA of IMT-2020 All Scenarios Using Jamovi	179
N.1	COST 2100 datasets from IEEE Dataport	181
N.2	IMT-2020 datasets from IEEE Dataport	182
P.1	Thesis Approval Sheet	202



LIST OF TABLES

1.1	Related Studies on Multipath Clustering	8
1.2	Expected output per objective	10
1.3	Channel generated models' specifications	15
1.4	Accuracy of Significant Activation Functions using MNIST datasets . .	16
2.1	Existing Work on Multipath Clustering	21
2.2	Existing Work on Autoencoder Algorithm	29
3.1	k -DAE Results [OPOCHINSKY et al., 2020]	40
4.1	Channel generated models' comparison	43
4.2	Summary of CVIs	47
4.3	Summary of Versions and Libraries involved in the Study	55
5.1	Summary of Methods for Satisfying the Objectives	59
5.2	Different Configurations Implemented in the Study	61
6.1	Summary of Results per Objective	65
6.2	Mean and Standard Deviations of NMI results in C2CM	82
6.3	Mean and Standard Deviations of ACC results in C2CM	83
6.4	Mean and Standard Deviations of ARI results in C2CM	83
6.5	Mean and Standard Deviations of NMI results in IMT-2020	84
6.6	Mean and Standard Deviations ACC results in IMT-2020	84
6.7	Mean and Standard Deviations ARI results in IMT-2020	85
6.8	Mean and Standard Deviations of JI results in C2CM	86
6.9	Mean and Standard Deviations JI results in IMT-2020	87
6.10	Tabular Values of C2CM ANOVA in All Indoor Scenarios	90



6.11 Tabular Values of C2CM ANOVA in All Outdoor Scenarios	90
6.12 Tabular Values of C2CM ANOVA in All Scenarios	90
6.13 Tabular Values of IMT-2020 ANOVA in All Indoor Scenarios	93
6.14 Tabular Values of IMT-2020 ANOVA in All Outdoor Scenarios	93
6.15 Tabular Values of IMT-2020 ANOVA in All Scenarios	94
6.16 C2CM- Percentile Differences of ECDF in each Scenarios	100
6.17 IMT-2020- Percentile Differences of ECDF for each scenario	104
6.18 Correlation Coefficient (r) of JI and RE of C2CM	117
6.19 p-value and Correlation Coefficient (r) of JI and RE of IMT-2020	119
K.1 Revision to Final	169



ABBREVIATIONS

<i>k</i> -DAE	<i>k</i> -Deep Autoencoder	vi
<i>t</i> -SNE	<i>t</i> -distributed Stochastic Neighbor Embedding	20
2G	second-generation	2
3G	third-generation	2
3GPP	3rd Generation Partnership Project	8
4G	fourth-generation	2
5G	fifth-generation	2
6G	sixth generation	124
ACC	Clustering Accuracy	22
ADAGRAD	Adaptive Gradient	51
ADAM	Adaptive Momentum	vi
ADNI	Alzheimer's Disease Neuroimaging Initiative	25
AE+KM	Autoencoder+k-means	31
AG	Air-to-Ground	19
AI	Artificial Intelligence	5
AKPD	Adaptive Kernel Power Density	8
ANOVA	Analysis of Variance	13
AOA	Azimuth Angle of Arrival	43
AOD	Azimuth Angle of Departure	43
ARI	Adjusted Rand Index	47
B1	Band 1	15
B2	Band 2	15
BS	Base Station	14
C2CM	COST 2100	vi



CAE	Convolutional Autoencoder	27
CH	Calinski Harabasz	21
CONVLSTM	Convolutional Long Short-Term Memory	25
CVI	Clustering Validation Indices.....	10
DAMIC	Dataset for Activity Monitoring in Intelligent Environments with Context.....	28
DB	Davies-Bouldin	21
DCN	Deep Clustering Network	31
DCT	Directional Cosine Transform	69
DEC	Deep Embedded Clustering	26
DGG	Deep Clustering via a Gaussian-mixture variational autoencoder with Graph embedding	30
DNN	Deep Neural Network.....	25
DR	Dimensionality Reduction.....	20
ECDF	Empirical Cumulative Distribution Function.....	59
ELU	Exponential Linear Unit.....	82
EMBB	Enhanced Mobile Broadband	44
EOA	Elevation Angle of Arrival	43
EOD	Elevation Angle of Departure	43
GBSCM	Geometric-Based Stochastic Channel Modelling	vi
GBSM	Geometric-Based Stochastic Channel Model.....	14
GCN	Graph Convolutional Network.....	27
GD	Generalized Dunn	22
GMM	Gaussian Mixture Model.....	26
GMVAE	Gaussian Mixture Variational Autoencoder	31
GPU	Graphics Processor Unit	6
GUI	Graphical User Interface.....	24
HHAR	Human Activity Recognition	26



IDE	Integrated Development Environment	10
IEEE	Institute of Electrical and Electronics Engineers	11
IMT-2020	International Mobile Telecommunications-2020 Channel Model	
INH	vi Indoor Hotspot	15
ITU	International Telecommunication Union	44
JI	Jaccard Index	vi
KPD	Kernel Power Density	8
KPM	k -Power-Means	6
KSA	k -sparse Autoencoder	25
KT	Korea Telecommunication	8
LEO	Low-Earth Orbit	35
LOF	Local Outlier Factor	29
LOS	Line-of-Sight	15
MCD	multipath component distance	6
MGAE	Marginalized Graph Autoencoder	27
MIMO	multiple-input multiple-output	2
MIXAE	Mixture of Autoencoders	26
MMWAVE	millimeter-wave	19
MNIST	Modified National Institute of Standards and Technology ..	15
MPC	multipath components	vi
MRI	Magnetic Resonance Imaging	25
NLOS	Non-Line-of-Sight	15
NMI	Normalize Mutual Index	21
NORB	NYU Object Recognition Benchmark	29
NSL-KDD	Network Security Laboratory-Knowledge Discovery in Databases	27
O2I	Outdoor-to-Indoor	13



QUADRIGA	Quasi Deterministic Radio Channel Generator	8
RANDNET	Randomized Neural Network for Outlier Detection.....	28
RE	Reconstruction Error	46
RELU	Rectified Linear Unit.....	vi
RMA	Rural Macro	15
RMSPROP	Root Mean Square Propagation	51
SGD	Stochastic Gradient Descent	vi
SSP	Service Switching Point.....	44
S-VAE	Structured Variational Autoencoder	31
SAGE	Space-Alternating Generalized Expectation-Maximization .	23
SCA	Sparse Clustering Autoencoder.....	26
SCAMS	Simultaneous Clustering And Model Selection	8
SCAMSMA	Simultaneous Clustering and Model Selection Matrix Affinity	24
SD	standard deviation.....	13
SIMO	Single-Input Multiple-Output	21
SOM	Self Organizing Maps.....	20
STAE	Spatial-Temporal Autoencoder.....	30
STRIL	Spatial-Temporal Relation Learning.....	30
SVM-AKPD	Support Vector Machine Assisted AKPD	8
TCSL	Temporal Cluster-Spatial Lobe	19
UMA	Urban Macro	15
UMI	Urban Micro	15
USPS	United States Postal Service.....	28
VADE	Variational Deep Embedding.....	26
VAE	Variational AutoEncoder	8
VAE-GAN	Variational Autoencoder-Generative Adversarial Network..	31
VAED	Variational Autoencoder with Discriminative Learning .. .	30



VQ-VAE	Vector Quantized Variational Autoencoder	31
WAE	Wasserstein Autoencoder.....	31
WI-FI	Wireless Fidelity.....	35
XB	Xie-Beni.....	22



NOTATION

$\phi_{\ell, AoA}$	azimuth angle-of-arrival	37
$\phi_{\ell, AoD}$	azimuth angle-of-departure.....	37
C_{calc}	calculated multipath clusters.....	49
C_{ref}	reference multipath clusters	49
$\theta_{\ell, AoA}$	elevation angle-of-arrival	37
$\theta_{\ell, AoD}$	elevation angle-of-departure.....	37
η_{Jac}	Jaccard membership index	48
M_{00}	number of instances that are part of a different cluster	50
M_{01}	number of members present in C_{ref} but not in C_{calc}	49
M_{10}	number of members present in the C_{calc} but not in C_{ref}	49
M_{11}	number of members present in both C_{ref} and C_{calc}	49
τ_{ℓ}	delay.....	37
x_{ℓ}	multipath vector	37

Throughout this thesis, mathematical notations conform to ISO 80000-2 standard, e.g. variable names are printed in italics, the only exception being acronyms like e.g. SNR, which are printed in regular font. Constants are also set in regular font like j . Functions are also set in regular font, e.g. in $\sin(\cdot)$. Commonly used notations are t , f , $j = \sqrt{-1}$, n and $\exp(\cdot)$, which refer to the time variable, frequency variable, imaginary unit, n th variable, and exponential function, respectively.



GLOSSARY

<i>k</i> -Deep Autoencoder	It is a clustering approach based on unsupervised machine learning. The encoder and decoder network that make up this system work together to reassemble the input data. 6
activation function	An activation function is a mathematical operation used in artificial neural networks. It acts like a gate, allowing information to pass through only when the input is significant enough. 6
autoencoder	A type of artificial neural network used to learn efficient coding of unlabeled data having an encoder, bottleneck, and decoder. 6
azimuth angle	The angle measured counterclockwise in a horizontal plane. 37
channel model	A mathematical representation of how wireless signals undergo propagation effects while traversing a communication channel. 3
cluster	A group of multipath components that has similar parameters. 3
dimensionality	Number of attributes or features a dataset has. 6
elevation angle	The angle measured in a vertical plane. 37
Jaccard Index	Proximity measurement used to compute the similarity between two objects. 10



multipath Multiple-Input Multiple-Output	The radio signal propagation phenomenon caused by atmospheric effects or terrestrial objects, resulting in signals reaching the receiving antenna through multiple paths via reflection, refraction, diffraction, scattering, or similar factors. 3 Using two or more antenna elements at the transmitter and receiver. 2
---	--



LISTINGS

6.1 Under main run file script:	68
6.2 Under utilities script:	68
6.3 Under utilities script:	69
6.4 Under utilities script for IMT-2020:	69
6.5 Under main run file script:	70
6.6 Under main run file utilities:	70
6.7 Under <i>k</i> -DAE script:	71
6.8 The code contains the final settings:	71
6.9 Under autoencoder script:	72
6.10 Under autoencoder script:	73
6.11 Under <i>k</i> -DAE script:	73
O.1 Main Run file	185
O.2 Main Run file	186
O.3 Main Run file	187
O.4 Utilities Script	189
O.5 Utilities Script	190
O.6 Utilities Script	191
O.7 Utilities Script	192
O.8 Utilities Script	193
O.9 k-DAE file	195
O.10 k-DAE file	196
O.11 k-DAE file	197
O.12 Autoencoder file	199
O.13 Autoencoder file	200



Chapter 1

INTRODUCTION

1.1	Background of the Study	2
1.2	Prior Studies	8
1.3	Problem Statement	9
1.3.1	General Problem	9
1.3.2	Specific Problems	9
1.4	Objectives and Deliverables	9
1.4.1	General Objective	10
1.4.2	Specific Objectives	10
1.4.3	Expected Deliverable	10
1.5	Significance of the Study	11
1.5.1	Technical Benefit	11
1.5.2	Social Impact	11
1.5.3	Environmental Impact	12
1.5.4	Academic Impact	12
1.6	Assumptions, Scope and Delimitations	12
1.6.1	Assumptions	12
1.6.2	Scope	12
1.6.3	Delimitation	13
1.7	Description and Methodology	13
1.7.1	Channel Generated Models	14
1.7.1.1	COST 2100 (C2CM)	14
1.7.1.2	International Mobile Telecommunications- 2020 (IMT-2020)	14
1.7.2	Activation Functions	15
1.8	Overview	16



This chapter introduces the relevance and existence of MPC clustering in developing wireless channel model. Also, the problems in clustering MPC that should be administered. More so, the objectives, deliverables, and description of methodology used in the study.

1.1 Background of the Study

Wireless communication completely transformed the way people communicate and obtain information. Permeating aspects of modern life, such as, making phone calls up to complex networks that make up the international communication [Yen and Chou, 2001].

Moreover, wireless communication systems today are an integral part of many different wireless communication devices. Mobile phones are just one of the various wireless communication devices. Bluetooth, Wi-Fi, and third-generation (3G) and fourth-generation (4G) networks are features, to name a few. Since the turn of the twenty-first century, wireless communication technology has advanced quickly. From second-generation (2G) to 4G, the wireless communication bandwidth of networks became steadily wider and the data transfer rate dramatically increased [Zhao, 2019].

Furthermore, the continuous improvement of wireless communication systems led to the existence and development of fifth-generation (5G) mobile network standard. 5G aims to achieve faster data rates, reduced latency, and increased capacity compared to earlier generations of cellular networks. To achieve higher data rates, capacity, traffic density, and connection density in 5G, the use of massive Multiple-Input Multiple-Output technology is one of the keys enabling technology. multiple-input multiple-output (MIMO) systems provide significant capacity, strong diversity, and minimal interference, making them highly promising



for enhancing the data rate and performance of wireless communication systems. MIMO is highly important in modern wireless communication systems because they can significantly enhance spectral efficiency [Wu et al., 2014].

However, transmitted signals in MIMO do not traverse on a single path, sometimes they are reflected, refracted, diffracted, and scattered due to the objects that are present along the path between the transmitter and receiver. These objects cause the signals to have a continuous sequence of routes. Fig 1.1 shows a 3D MIMO channel model which displays the array of transmitting antennas signals in clusters characterized by their features and the array of receiving signals. Leading to an interesting case, where data analysis techniques are used in clustering MPCs, playing a crucial role for achieving accurate channel models [He et al., 2018].

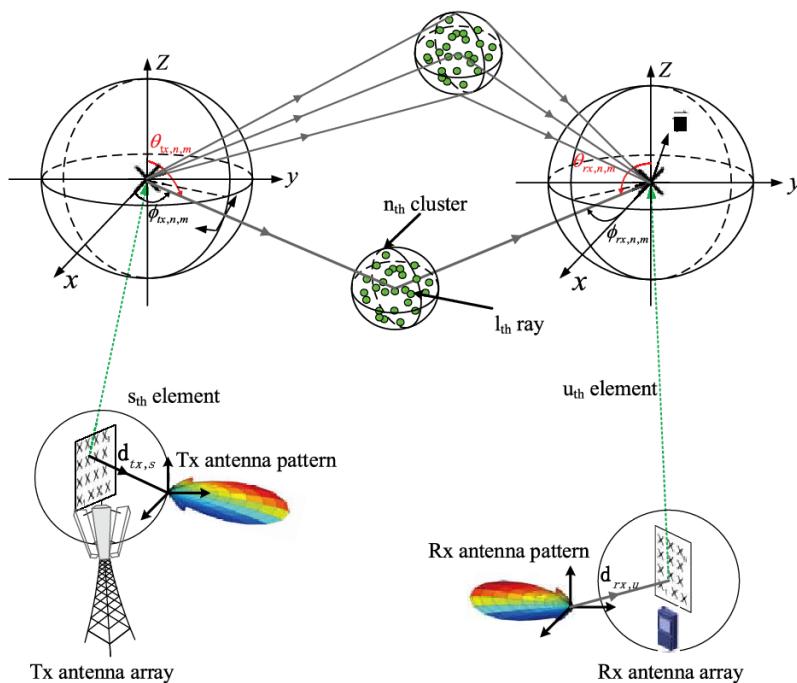


Fig. 1.1 Illustration of 3D MIMO Channel Model
[Zhang et al., 2017b]

Consequently, this occurrence of different paths traversed by the signal are called multipaths. The utilization of MIMO that provides multipath mitigation is also



constantly being researched and enhanced for use in real world situations. To mitigate this problem and exploit the multiple paths which can be used as reliable transmission paths, a statistical model of multipath signal reception is established by the attenuation, propagation delay, and carrier phase difference parameters of multipath signals [Xue et al., 2022].

In wireless communication systems, research on different channel models is in demand since it determines the performance of a system. Channel modelling is a process of exploring channel features for different real environments, to recognize the applicability of a system for a certain scenario. When a signal propagates in a channel for guided media, the signal attenuates due to the noise of the line. However, in unguided media, since it travels in free space, the signal attenuates depending on the environment, eg., diffraction, refraction, or reflection. In this case, this should be modelled to know the properties of each path, eg., delay and attenuation, for design optimization [Huang et al., 2019].

The goal of wireless channel modeling is to accurately represent the MPCs in wireless channels. Research has shown that MPCs often form clusters, meaning they group together based on similar properties such as angle and delay. By clustering MPCs, the main MPCs and their corresponding scatterers can be more easily identified, leading to a more detailed analysis of the physical environment at the propagation level. This method also allows for a more streamlined description by distinguishing between intra-cluster and inter-cluster statistics.

Analyzing measurement data typically involves three main steps as shown in the Fig 1.2. First, MPCs are extracted from raw data using estimation algorithms. Then, these MPCs are clustered based on their characteristics, a statistical process that greatly influences the resulting channel model by determining the features extracted and their presentation. The physical characteristics of the channels impact what constitutes effective MPC clustering, and validation criteria assist in



designing algorithms to achieve the desired outcomes. Finally, MPCs and clusters are characterized, which involves estimating parameters such as path loss, fading, and spreads in delay and angle [He et al., 2018].

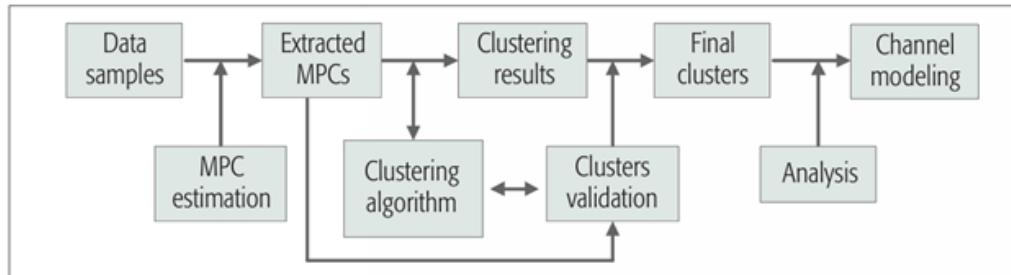


Fig. 1.2 Illustrations of clustering procedure for channel modeling
[He et al., 2018]

To obtain wireless channel models, the measured angular and temporal parameters of MPCs are grouped into clusters. Multipath component clustering is a technique that can significantly enhance the accuracy and efficiency of channel models for MIMO and 5G communication systems [Li et al., 2019]. It achieves this by grouping together MPCs with similar characteristics. This approach enables a clearer understanding of the propagation environment and facilitates the development of more precise and efficient channel models. Multipath component clustering can also help simplify and reduce the channel model parameters [Lu et al., 2021].

According to [Boban and Degli-Esposti, 2023], the "White Paper on Radio Channel Modeling and Prediction to Support Future Environment-aware Wireless Communication Systems", Artificial Intelligence (AI) is set to play an increasingly influential role in channel modeling. The paper proposes two key suggestions for future research.

First, data is essential for the efficacy of machine learning models. The primary sources of training data for machine learning in propagation are measurement data and simulation data. To enhance the quality and reduce the cost of measurement data, the development of novel channel measurement technologies is



recommended, with a focus on improvements such as denoising. Additionally, existing simulation software should be updated to support mainstream data formats compatible with popular deep learning frameworks like PyTorch and TensorFlow.

Second, considering that channel characteristics datasets are relatively small compared to typical image and voice datasets, the paper suggests using lightweight deep learning models. Large deep learning models tend to overfit when trained on small datasets. Lightweight models offer the added benefit of being easier to compute on base stations without Graphics Processor Unit (GPU)s. Furthermore, adopting distributed learning and federated learning approaches can help reduce communication costs, as fewer parameters need to be updated [Boban and Degli-Esposti, 2023].

Due to the high variety of MPCs data, finding a general clustering algorithm has been very challenging, and many user-specified parameters of the algorithms need to be manually adjusted [He et al., 2018]. Several algorithms have been proposed to cluster MPCs, considering factors such as power, delay, and angles. Examples include the k -Means algorithm and the k -Power-Means (KPM) algorithm, which seeks cluster centers based on MPC power and uses multipath component distance (MCD) to measure the similarity between MPCs. However, KPM has unsupervised learning characteristics and requires prior knowledge for initialization. This means that parameters need to be manually adjusted based on different data, making KPM somewhat subjective [He et al., 2017].

Furthermore, Fig 1.3 shows autoencoder which is a dimensionality reduction algorithm which encodes input data on different neural networks using a specified activation function. This reduces its dimension, and the data is compressed to a bottle neck. Moreover, the decoder on the output attempt to reconstruct the input using the compressed dimensions [Geng et al., 2019].

This paper aims to cluster MPCs automatically using k -Deep Autoencoder (a



k-means algorithm mixed with an autoencoder) with the use of different activation functions, measure their accuracy through JI, and evaluate and compare their clustering results.

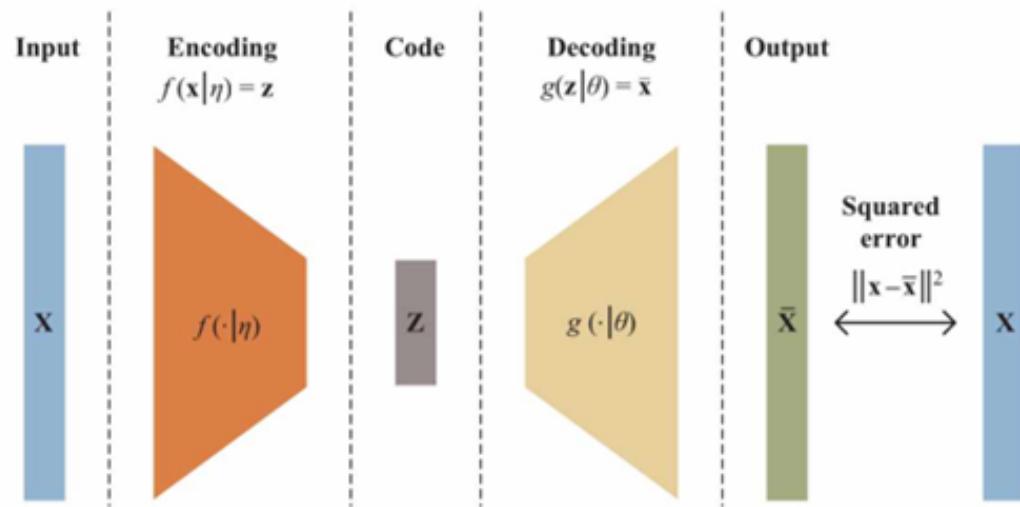


Fig. 1.3 Illustration of Autoencoder Algorithm
[Geng et al., 2019]



1.2 Prior Studies

Table 1.1 shows a brief comparison of several studies about MPCs and is further discussed in Chapter 2.

TABLE 1.1 RELATED STUDIES ON MULTIPATH CLUSTERING

Literature	Data Source	Clustering Algorithm	Type of Clustering	Frequency Range (GHz)
[Czink et al., 2006]	3GPP	KPM	Centroid-based	0.5-100
[Gustafson et al., 2013]	Measurement	KPM	Centroid-based	Not indicated
[He et al., 2017]	3GPP	KPD	Centroid-based	0.5-100
[Moayyed et al., 2019]	mmWave spectrum below 100 GHz	KPM	Centroid-based	<100
[Blanza and Materum, 2019a]	C2CM	SCAMS	Constraint-based	<6
[Lee, 2020]	Manual 3GPP Docomo KT	KPM	Centroid-based	0.5-100
[Materum and Teologo Jr, 2021]	C2CM	KPM	Centroid-based	<0.6
[Du et al., 2022]	QuaDRiGa	AKPD,SVM-AKPD,KPD,KPM, MCD	Centroid-based	0.45-100
[Baur et al., 2022]	3GPP	VAE	Dimension Reduction	0.5-100
[Trinidad and Materum, 2023]	C2CM	Manual Reclustering	Dimension Reduction	<0.6



1.3 Problem Statement

1.3.1 General Problem

In clustered-based channel models, several parameters are considered, such as delay, angle, or power. Consequently, the main concern of structured clusters is to incorporate these properties. However, clustering MPCs is still an open research, with no fixed algorithm available. Moreover, the modified k -DAE is designed to improve clustering performance by leveraging the capacity of deep learning to extract complex features from MPCs. The accuracy and reliability of MPC clustering must be significantly improved by the successful application of this algorithm, which may enhance the reliability of wireless communication networks.

1.3.2 Specific Problems

This study sought to address the following problems:

1. Sufficient activation function in k -DAE algorithm needs to be studied to recognize suitable function for clustering angular and time features of MPCs.
2. k -DAE's activation functions need to observe the clustering performance of C2CM and IMT-2020 channel models.
3. The accuracy and the associative correlation of reconstruction error and Jaccard index of k -DAE with ReLU and sigmoid activation functions in clustering the C2CM and IMT-2020 should be determined.

1.4 Objectives and Deliverables

Shown below are the general objective, specific objectives of the study, and summary of Sec 1.4.1 and Sec 1.4.2 are reflected on Table 1.2.



1.4.1 General Objective

GO: To cluster C2CM and IMT-2020 multipath components using modified deep autoencoder algorithm.

1.4.2 Specific Objectives

SO1: To have a sufficient activation function for deep autoencoder clustering algorithm.

SO2: To achieve close or higher Jaccard Index in clustering C2CM and IMT-2020.

SO3: To measure the JI performance and the associative correlation between autoencoder reconstruction error and JI.

1.4.3 Expected Deliverable

Table 2 shows the outputs, gains, results, realizations, and yields of the Thesis.

TABLE 1.2 EXPECTED OUTPUT PER OBJECTIVE

Objectives	Deliverable
GO: To cluster C2CM and IMT-2020 multipath components using modified deep autoencoder algorithm.	Statistical analysis and Cluster membership results of k -DAE using sigmoid and ReLU activation functions.
SO1: To have a sufficient activation function for deep autoencoder clustering algorithm.	Modify the k -DAE clustering algorithm code using Python IDE using sigmoid and ReLU activation functions to cluster MPCs.
SO2: To achieve close or higher JI in clustering C2CM and IMT-2020.	Statistics of the CVI using JI of modified algorithms using sigmoid and ReLU activation functions in clustering MPCs.

Continued on next page



Continued from previous page

Objectives	Deliverable
SO3: To measure the JI performance and the associative correlation between autoencoder reconstruction error and JI.	Compare the efficiency performance through JI and correlation between reconstruction error and JI of different modified k -DAE clustering algorithms using sigmoid and ReLU activation functions in clustering MPCs.

1.5 Significance of the Study

1.5.1 Technical Benefit

Network efficiency and reliability can be significantly improved by clustering MPCs. In line with, wireless system engineers and designers are the primary beneficiary that may be used in wireless system simulation. Wireless standardization bodies such as Institute of Electrical and Electronics Engineers (IEEE) and 3GPP are the secondary beneficiary. Moreover, channel emulator manufacturer vendors may use the channel models to make a hardware version of so that they do not need to have a field testing of equipment. The autoencoder is used to evaluate the gathered input for clustering data based on the predetermined traits of algorithm, factors, or parameters. As well as providing accurate data representations, such as potential outputs that are identical to the input.

1.5.2 Social Impact

In this modern era, many tasks and interactions are made through remote communication. Ensuring a strong, stable, and dependable wireless interconnectivity by accurate characterization of the wireless channel will not only allow a high data-rate transmission but will also make the users productive.



1.5.3 Environmental Impact

With the increasing demand for faster data rates and more dependable interconnections, traditional wireless system designs have reached their limits in mitigating non-ideal propagation conditions. In order to address these difficulties, more complicated designs and increased processing resources are necessary, which results in higher energy consumption. However, by conducting a practical examination of non-ideal conditions with lower energy use, the researchers can utilize more ecologically friendly and sustainable technology.

1.5.4 Academic Impact

Application of k -DAE in clustering channel generated datasets will contribute as reference data on the effectivity of the algorithm in clustering the specified channel models. Also, comparison and analysis of data serves as literature on clustering wireless MPCs using k -DAE.

1.6 Assumptions, Scope and Delimitations

1.6.1 Assumptions

- The obtained data from the C2CM and IMT-2020 channel Models with different scenarios are utilized as ground truth reference.
- The implementation of sigmoid and ReLU modified activation functions k -DAE algorithm can improve the accuracy of clustering the MPCs.

1.6.2 Scope

- Modifying activation functions using ReLU and sigmoid functions of clustering model k -DAE.



- JI is used for the validation of the clustered data.
- Comparative statistical analysis is used (Analysis of Variance (ANOVA), standard deviation (SD), and correlation coefficient).

1.6.3 Delimitation

- The algorithm applied was not developed from scratch.
- The obtained data clusters are limited to the C2CM and IMT-2020 channel models clustering results.
- The Outdoor-to-Indoor (O2I) scenarios on IMT-2020 are not considered in this study due to huge number of data.

1.7 Description and Methodology

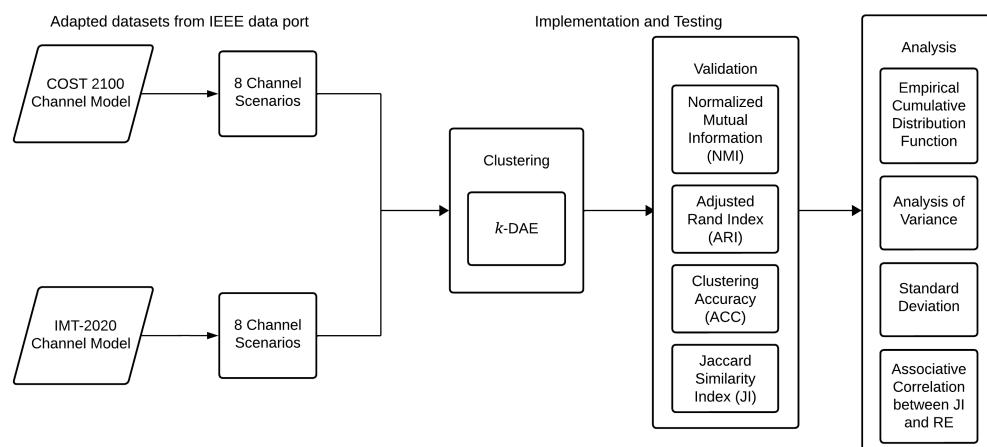


Fig. 1.4 Block Diagram of Methodology of the Study

Figure 1.4 shows an illustrated block diagram of the methodology of the study. It consists of three-parts the channel model datasets, implementation and testing of the clustering algorithm, and analysis of the gathered results.



1.7.1 Channel Generated Models

The following explains the channel generated models (adapted from IEEE data port) that is used in this study.

1.7.1.1 COST 2100 (C2CM)

C2CM channel model is from COST 259 and COST 273 which is a Geometric-Based Stochastic Channel Model (GBSM). The frequency band of this model is below 6 GHz. It involves three kinds of clusters i.e., local clusters, single-bound clusters, and twin clusters. Single-bound clusters and twin clusters are situated far from the Base Station (BS). The location of clusters is determined by two parameters angular and delay. The model can replicate the behavior of MIMO in terms of time, frequency, and space. Also, it allows the simulating of a non-stationary channel due to visibility region [Wang et al., 2018].

1.7.1.2 International Mobile Telecommunications- 2020 (IMT-2020)

The IMT-2020 channel model supports frequency band 0.5-100 GHz upon implementation. It involves 8 propagation scenarios and is considered as a GBSM based on adapting IMT-advanced channel model and 3GPP TR36.873 channel model. The number of clusters generated by IMT-2020 follows poisson distribution with a mean value of 3-10 [Wang et al., 2018].

This study utilized the datasets from two channel models, namely, C2CM (8 Channel scenarios) and IMT-2020 (8 channel scenarios) [Blanza et al., 2023]. These are clustered through modified activation function k -DAE algorithm.

After clustering, the gathered data are validated using CVIs i.e., JI. It was emphasized that the success of clustering data relies on the validation [Liu et al., 2010].

The similarities of cluster labels are calculated based on the ground truth from the datasets of C2CM and IMT-2020. The accuracy of the cluster membership is



determined using the JI formula. The JI can have values between 0 and 1, with 1 representing the maximum accuracy and 0 being the lowest.

TABLE 1.3 CHANNEL GENERATED MODELS' SPECIFICATIONS

Channel Models	Channel Scenario	Channel Bandwidth (GHz)
C2CM	Indoor B1 LOS Single Link Indoor B2 LOS Single Link Semi-Urban B1 LOS Single Link Semi-Urban B2 LOS Single Link Semi-Urban B1 NLOS Single Link Semi-Urban B2 NLOS Single Link Semi-Urban B1 LOS Multiple Links Semi-Urban B2 LOS Multiple Links	<6
IMT-2020	InH A LOS InH A NLOS RMa A LOS RMa A NLOS UMa A LOS UMa A NLOS UMi A LOS UMi A NLOS	0.5-100

1.7.2 Activation Functions

The standard state-of-the art activation functions are compared, and their accuracy is described. Significantly, ReLU and sigmoid are the activation functions that have a noteworthy accuracy using the Modified National Institute of Standards and Technology (MNIST) datasets [Apicella et al., 2021]. Table 1.4 shows the comparison of the accuracy of ReLU and sigmoid. Hence, ReLU and sigmoid are utilized in this study.



TABLE 1.4 ACCURACY OF SIGNIFICANT ACTIVATION FUNCTIONS USING MNIST DATASETS

Activation Functions		Accuracy %
Fixed Shape	<i>sigmoid</i>	97.9
Rectifier-based	<i>ReLU</i>	99.53

1.8 Overview

This thesis showed the effectivity of clustering wireless MPCs using modified activation functions of k -DAE algorithm. This study is organized as follows. Chapter 2 discusses the methods and related literature which highlighted the contribution of the study. Chapter 3 defines the theoretical ideas and concepts used in the study. Chapter 4 enriches the implementation of designing the output of the study. Chapter 5 describes the methods to accomplish the objectives in accordance with the study. Chapter 6 presents the accomplishments and resolutions to attain the specified objectives. Lastly, Chapter 7 concludes, presenting the observations and findings, and the future directive of the study for continuous study.



Chapter 2

LITERATURE REVIEW

2.1 Existing Works	18
2.1.1 Existing Works in Clustering MPCs	18
2.1.2 Existing Works on Autoencoders	25
2.2 Lacking in the Approaches	32
2.3 Summary	32



This section justifies the relevance of the study in clustering MPCs through existing literatures. The existing works in clustering MPCs are explained, summarized and tabulated. Furthermore, the previous studies on autoencoders are listed. Moreover, the lacking approaches from previous studies are discussed.

2.1 Existing Works

The initial application of automated clustering in analyzing and grouping of datasets in MIMO propagation channel was introduced by [Czink et al., 2006]. Clustering is performed by providing a framework and utilizing an algorithm to group similar data i.e., clustering wireless MPCs [Hu et al., 2020]. The channel parameters will be analyzed and subjected to algorithm using different visualization or plots [Czink et al., 2006]. The study of [Czink et al., 2006] successfully clustered MPCs for 3GPP channel model which enables continuous research on clustering MPCs.

2.1.1 Existing Works in Clustering MPCs

Clustering is one of the common types of unsupervised learning that divides the samples or data into groups called clusters, such that the clusters are more similar to each other and different from the other clusters [Geng et al., 2019]. Summarize on the Table 2.1 are the existing works in the clustering MPCs where different propagation scenarios were implemented, clustering algorithm, and frequency band. Observing the algorithm used in different studies, KPM is the majority used, as well as 3GPP and C2CM are majority used as data source. Since the first study on MPC [Czink et al., 2006], there is a significantly improvement on clustering algorithm used, the implementation on diverse various dataset scenarios, and clustering accuracy results for analyzing the effectivity of the implemented



algorithm on clustering membership using comparative similarity indices.

The first automated clustering of MPC was started by [Czink et al., 2006]. MPC clustering framework is presented, discard outliers, and decide on the number of clusters. This study used 3GPP channel model where, the frequency band is between 0.5-100 GHz. Furthermore, illustration of geometric and environmental factors that influence cluster angular characteristics were presented using KPM [Gustafson et al., 2013].

Development of a 3D statistical channel model using Temporal Cluster-Spatial Lobe (TCSL) clustering algorithm was used to cluster scenarios for simulating 28 GHz MIMO [Samimi et al., 2016]. Moreover, KPD was introduced in clustering MPCs which uses relative density and k -nearest MPCs in estimating density [He et al., 2017].

Clustering MPCs in millimeter-wave (mmWave) spectrum and below 100GHz was demonstrated by [Moayyed et al., 2019]. Narrow-beam antennas were analyzed on the effect of clustering and compared clustering characteristics of different types of antennas. The accuracy of KPM algorithm was evaluated in C2CM [Materum and Teologo Jr, 2021]. Furthermore, the number of cluster members were determined in C2CM datasets.

Spectral-based power weighted algorithm was first proposed in clustering MPCs. This study uses 11 GHz frequency band in which the proposed algorithm outperformed k -means, KPM, and traditional spectral-based clustering [Hu et al., 2020]. However, the first study of MPC clustering for time-varying Air-to-Ground (AG) channel model and 6.5 GHz data [Cui et al., 2022]. In addition, novel automatic clustering algorithm, AKPD and a SVM-AKPD were introduced for clustering MPCs [Du et al., 2014].

A notable study determined the number of clusters and cluster members [Blanza and Materum, 2019a]. Also, KPM is modified with incorporation of Mikowski



distance metric [Materum and Teologo Jr, 2021]. Moreover, mmWave O2I is explored propagation statistics for clustering MPCs [Lee, 2020].

On a recent study of [Ramírez-Arroyo et al., 2022], *t*-distributed Stochastic Neighbor Embedding (*t*-SNE) and VAE algorithms were introduced for Dimensionality Reduction (DR) and classification, which demonstrates an excellent performance in distinguishing between the recreation and real communication scenarios. Furthermore, a more flexible like hood model in using VAE decoder is a crucial part for effective clustering [Baur et al., 2022].

Continuous development on methods in clustering MPCs as [Africa et al., 2020] introduced graphical representation of multipath cluster dataset and determined most suitable visualization approach for MPC validation and interpretation. Also, [Alejandrino et al., 2022] proved the benefit of using Self Organizing Maps (SOM) in visualizing MPCs. Consequently, improved SCAMS results are achieved using DR techniques and interactive refinement of cluster membership accuracy [Trinidad and Materum, 2023].



TABLE 2.1 EXISTING WORK ON MULTIPATH CLUSTERING

Summary	Propagation Scenario/ Dataset	Clustering Algorithm	CVIs	Clustering accuracy
Presented a framework that is able to cluster MPCs, decide on number of clusters, and discard outliers [Czink et al., 2006].	3GPP	KPM	Combined Validate index CH index and DB index	Better than CH and DB for cluster angular spreads larger than 2.5 degrees
The MPCs were clustered using two new automatic clustering algorithms: an SVM-AKPD and an AKPD. Novel Automatic clustering algorithm, an AKPD and a SVM-AKPD were introduced in clustering MPCs [Du et al., 2014].	QuaDRiGa	AKPD SVM-AKPD KPD KPM MCD	Not indicated	Not indicated
Developed a 3-D statistical channel model for simulating 28 GHz MIMO system-level performance using TCSL clustering. Compared SIMO and MIMO channel capacities with Rician and Rayleigh distributions via Monte Carlo simulation [Samimi et al., 2016].	28 GHz small-scale track measurements	TCSL	Not indicated	Not indicated
Incorporated the modelled behavior of MPCs using Kernel density clustering technique. It uses relative density and uses only k -nearest MPCs to estimate density. To combine clusters, it implemented a heuristic approach [He et al., 2017]	3GPP	KPD	NMI	0.45

Continued on next page



Continued from previous page

Summary	Propagation Scenario/ Dataset	Clustering Algorithm	CVIs	Clustering accuracy
Demonstrates clustering process and its validation across a wide range of frequencies in mmWave spectrum below 100GHz. Also, narrower-beam antennas were analysed on how clustering solution changes and provided a comparison of the cluster characteristics for different types of antennas [Moayyed et al., 2019].	mmWave spectrum below 100 GHz	KPM	CH DB GD XB	CH (15 clusters): 1 XB- (6 clusters): 1 DB and GD- (2 clusters): 1
Determined number of clusters and cluster members in C2CM synthetic dataset [Blanza and Materum, 2019a].	C2CM	SCAMS	JI	Good clustering accuracy for the indoor scenarios (Indoor B1: 0.7305, Indoor B2: 0.7582), while improvements should be made for semi-urban scenarios.
A novel spectral clustering-based clustering algorithm is put proposed. The suggested technique performs better than k -means, KPM, and traditional spectrum-based clustering [Hu et al., 2020].	11 GHz Hall Environment	Spectral Based Power Weighted	CH ACC	>0.94
Explored propagation statistics based on clustering for mmWave O2I communication [Lee, 2020].	Manual 3GPP Docomo KT	KPM	CH DB	Not indicated

Continued on next page



Continued from previous page

Summary	Propagation Scenario/ Dataset	Clustering Algorithm	CVIs	Clustering accuracy
Modified the KPM method by implementation of weights or loads based on main component analysis. Also, incorporation of Mikowski distance metric instead of Euclidean distance [Materum and Teologo Jr, 2021]	C2CM	KPM	JI	Channel Scenario 1: 0.9471 Channel Scenario 2: 0.9481
First study on the MPC tracking and clustering for time-varying AG channel model using 500 MHz of bandwidth and 6.5 GHz measurement data [Cui et al., 2022].	Commercial ultra-wideband 6.5 GHz	SAGE KPM	Silhouette index DB index	Not indicated
Presented novel application of <i>t</i> -SNE clustering algorithm in telecommunication. Separated different environments into several clusters which allowed clear visualization. Using <i>t</i> -SNE and VAE for DR and classification demonstrates excellent performance in distinguishing between the recreation and real communication scenarios [Ramírez-Arroyo et al., 2022].	3GPP	<i>t</i> -SNE and VAE	Not indicated	Indoor: 0.975 Anechoic: 0.471 Reverberation: 0.982 Rooftop: 0.461
Proposed the use of VAE to group unlabeled channel state information based on model order in an unsupervised way. It was found that using a more flexible likelihood model for the VAE decoder is crucial for effective clustering [Baur et al., 2022].	3GPP	VAE	Not indicated	Not indicated

Continued on next page



Continued from previous page

Summary	Propagation Scenario/ Dataset	Clustering Algorithm	CVIs	Clustering accuracy
Improved SCAMS results from SCAMSMA and C2CM data set using DR algorithm techniques and incorporation of a GUI that projects MPCs and interactive refinement of cluster membership accuracy [Trinidad and Materum, 2023].	C2CM	Manual reclustering	JJ	Low value of 0.3368, median of 0.4697, and a high value of 0.8884 for all the datasets.



2.1.2 Existing Works on Autoencoders

The study conducted by [Bhatkoti and Paul, 2016] investigated the effectiveness of the Modified k -sparse Autoencoder (KSA) with a linear activation function for classifying Alzheimer's disease within a deep learning framework. This research adopted a practical approach by utilizing real Magnetic Resonance Imaging (MRI) images from patient screenings and comparing them with data from the Alzheimer's Disease Neuroimaging Initiative (ADNI) and data from previous studies. Another Study conducted by [Kannadasan et al., 2019], which demonstrated the utilization of the Deep Neural Network (DNN) framework using Linear activation function in diabetes data classification. Their approach involves the integration of stacked autoencoders within the DNN structure, enhanced through the incorporation of a softmax classifier. The study specifically applies this methodology to classify type 2 diabetes within the Pima Indian dataset. With a precision value of 90.66%, based on the experiments and observations, the model proved its importance in predicting the diabetes of a patient.

[Nayak et al., 2020] proposed an architecture in convolutional spatiotemporal autoencoder for video anomaly detection. The architecture comprises of three major sections, namely: spatial encoder, temporal encoder-decoder, and spatial decoder. The spatial encoder contains three layers of convolutional layers; the temporal encoder-decoder is realized with the assistance of Convolutional Long Short-Term Memory (ConVLSTM), gated with the tanh and sigmoid activation functions; and the spatial decoder is applied using three layers of the deconvolutional layers. The proposed model is trained using the datasets that composed the normal classes by minimizing the reconstruction error. After the model is trained, using the test dataset that contain anomalous activities, then high reconstruction error has resulted. Consequently, a high anomaly score and low regularity score is obtained. The proposed model is tested on UCSD Ped1 and Ped2 dataset



successfully. The result of the performance is found to be promising.

A novel Orthogonal Autoencoder-based deep clustering framework that uses ReLU as the activation function was proposed in the study of [Wang et al., 2017], that can extract the latent embedding and determine the clustering assignment simultaneously. The MNIST dataset was also utilized in the research conducted by [Kajó et al., 2020], introduced an innovative concept in cellular response analysis. The paper presented the Sparse Clustering Autoencoder (SCA) with a sigmoid activation function, a method designed to autonomously transform cellular responses into a unique graph-like representation known as Network State Transition Graphs. The SCA, as proposed, serves as an advanced clustering algorithm based on deep neural networks, specialized in translating network inputs into this distinct network state transition graph. This approach has the potential to enhance our comprehension of cellular behaviors and interactions, offering a new perspective on the complex world of biological systems.

In the paper of [Zhang et al., 2017a], the authors introduce Mixture of Autoencoders (MIXAE), a cutting-edge deep learning architecture that revolutionizes unsupervised data clustering. MIXAE stands out for its innovative blend of features, incorporating a mixture of autoencoders, a joint optimization framework, and ReLU as the activation function. This combination allows it to simultaneously learn manifold structures and clustering assignments. Along with other deep learning models like Deep Embedded Clustering (DEC) and Variational Deep Embedding (VaDE), MIXAE consistently outperformed traditional machine learning techniques such as k -means and Gaussian Mixture Model (GMM) in clustering MNIST, Reuters, and Human Activity Recognition (HHAR), demonstrating the significant advantage of leveraging autoencoders to extract latent features for challenging datasets. MIXAE's impressive performance on benchmark datasets showcases its ability to effectively cluster diverse writing styles despite minor



errors, reinforcing its potential in tackling complex data clustering tasks.

A Marginalized Graph Autoencoder (MGAE) algorithm for graph clustering is introduced. The graph structure and content are the input of the algorithm which will learn an augmented autoencoder upon the datasets with the Graph Convolutional Network (GCN) as a building block was proposed in the study of [Wang et al., 2017]. MGAE used a marginalization process to disturb the network information such that the content and structures can interact with each other to achieve optimized learning outcomes. As a result, MGAE delivered a stacked graph convolutional network architecture, with each layer integrating both structure and content information in a convolutional neural network. Such layered structures are flexibly stacked to support deep learning. MGAE utilized all useful information of the graph for representation learning in the spectral domain, in which the spectral clustering algorithm achieved superior clustering results. Experimental results and comparisons, using 11 state-of-the-art algorithms, validated the MGAE's performance.

Wireless connectivity has been the backbone of modern communication systems, the tremendous network traffic signifies and proves that it is popular to the users than wired network. However, in comparison to the standard wired network, wireless networks are more susceptible to attacks. In the study of [Chen et al., 2018], Autoencoder-based network anomaly detection, the main focus was the use of Convolutional Autoencoder (CAE) for DR. The convolutional layer has the capability to reduce the number of features, whereas the deconvolution layer can increase the feature count. Subsequently, CAE, employing a sigmoid activation function, was utilized to carry out DR, effectively capturing non-linear correlations among the features. When tested on the Network Security Laboratory-Knowledge Discovery in Databases (NSL-KDD) dataset, the assessment revealed that the anomaly detection approach based on Convolutional



Autoencoders outperformed other competing methods.

Moreover, study conducted by [Chen et al., 2017] introduced autoencoder ensembles for unsupervised outlier detection. Various randomly connected autoencoders with different structures and connection densities are used as base ensemble components. An adaptive sample size method within the ensemble framework is designed to achieve the dual goals of improved diversity and training time. Randomized Neural Network for Outlier Detection (RandNet) approach is used in this paper, an adaptive sampling with randomized model construction in order to achieve high-quality results. It was proven that the technique proposed improved significantly compared to the other neural network methods for outlier detection through random edge sampling with adaptive data sampling.

A recent study of [OPOCHINSKY et al., 2020] proposed a hybrid algorithm which consists of k -means and neural networks. The cluster are represented by autoencoders instead of centroids. They cluster three datasets MNIST, Fashion, and United States Postal Service (USPS) which consists of handwritten letters and numbers. The results show a significant improvement in clustering using the k -DAE algorithm. The k -DAE outperforms the other autoencoder algorithms on USPS data sets and has a similar clustering result to Dataset for Activity Monitoring in Intelligent Environments with Context (DAMIC) which is considered to have the most accurate clustering results.



TABLE 2.2 EXISTING WORK ON AUTOENCODER ALGORITHM

Summary	Activation Function	Algorithm	Dataset
The KSA algorithm's efficacy in classifying deep learning frameworks for Alzheimer's disease diagnosis [Bhatkoti and Paul, 2016].	Linear function	KSA	MNIST Object recognition Benchmark NORB
Developed an approach by using mixture of autoencoders to cluster data from low-dimensional nonlinear manifolds and learn the underlying manifolds of every cluster [Zhang et al., 2017a].	ReLU	MIXAE	MNIST Reuters HHAR
Autoencoder ensembles for unsupervised outlier detection [Chen et al., 2017].	Sigmoid ReLU	LOF Ensemble algorithm	UCI Machine Learning Repository Cardio, Ecoli, etc.
The MGAE a marginalized graph convolutional network for graph feature representations [Wang et al., 2017].	Linear activation function	MGAE	Cora Citeseer Wiki
The nonlinear correlation between features was successfully captured by the Autoencoder. The Convolutional Autoencoder used for DR was proven superior to the conventional autoencoder [Chen et al., 2018].	Sigmoid	CAE	NSL-KDD
Utilized DNN for the classification of type 2 diabetes in the Pima Indian dataset by employing a stacked autoencoder approach [Kannadasan et al., 2019].	Linear function	Stacked Autoencoders	Pima Indians Diabetes Dataset

Continued on next page



Continued from previous page

Summary	Activation Function	Algorithm	Dataset
This study proposed a DGG that unifies model and similarity-based clustering approaches and outperforms deep model-based clustering and deep spectral clustering [Yang et al., 2019].	ReLU	DGG	MNIST STL-10 Reuters HHAR
A DNN-based clustering algorithm - Sparse Clustering Autoencoder, capable of autonomously encoding cell behavior into a graph-like representation, the Network State Transition Graphs [Kajó et al., 2020].	Sigmoid	SCA	MNIST
The STRL framework for video anomaly detection utilizing and STAE [Nayak et al., 2020].	ReLU	STAE STRL	UCSD Ped1& Ped2 dataset CUHK Avenue dataset ShanghaiTech dataset
Inspired by [Song et al., 2013], which proposed to “artificially re-align each point in the latent space to an autoencoder to its nearest class neighbors during training”, the study proposed to generalize the approach in a generalized way, such that Euclidean distance in the latent space is represented by KL divergence, using probability distributions as inputs, and using the Bayesian Gaussian mixture model for clustering in the latent space [Lim et al., 2020].	Not Indicated	VAED	MNIST USPS SHVN Scene15 MIT67

Continued on next page



Continued from previous page

Summary	Activation Function	Algorithm	Dataset
This study provided an evaluation and comparison of the variations of VAEs based on their end goals and resulting architectures [Wei et al., 2020].	Not indicated	VaDE GMVAE INFO-VAE β -VAE VQ-VAE S-VAE VAE-GAN f-VAE-GAN-D2 WAE Zero-VAE-GAN S3VAE	MNIST
Proposed a deep clustering algorithm which is a combination of k -means and neural networks. The hybrid clustering algorithm outperforms later clustering algorithm such as Autoencoder followed by k -means AE+KM, DCN, DEC, and DAMIC [Opochinsky et al., 2020].	eLU Sigmoid	k -DAE	MNIST Fashion USPS



2.2 Lacking in the Approaches

The literature that was presented used different clustering algorithms in a variety of domains. In the field of medicine, autoencoders are frequently used to effectively regenerate images for purposes including anomaly detection, feature detection, and facial recognition without affecting the information. Comparing various methods that are already employed in wireless communication needs probable study. It is also notable that algorithms used in clustering are machine learning. In line with that, k -DAE is considered as a deep learning algorithm. Meanwhile, no study has been done to show that autoencoders are utilized in MPC clustering. Moreover, literatures cluster only a few channel model data sets. Reflecting on Table 2.1, C2CM and 3GPP are frequently clustered. Thus, this study aims to further investigate the accuracy of the clustering obtained by using the k -DAE algorithm for clustering wireless multipath channel models and incorporate different channel model models for comparison.

2.3 Summary

This chapter discussed the existence of study in clustering MPCs. Several literatures suggest that the topic needs further study since there is still area to improve the algorithm in clustering wireless MPCs. It was presented in the previous sections that some channel models are not used as dataset for clustering based on Table 2.1. Hence, the incorporation of IMT-2020 channel model.

Furthermore, MPC clustering in autoencoder algorithm is still a study that needs exploration. Literature presented implies that autoencoder algorithm is effective on reconstructing images in the medical field and effective for large amount of data. Similarly, when new channel models arrive, there is a huge amount of data that is not advisable for manual clustering. Hence, implementation



of autoencoder algorithm in clustering wireless MPC assumes that it may provide a clearer representation of clustering for detailed analysis of gathered data.



Chapter 3

THEORETICAL CONSIDERATIONS

3.1	Radio Wave Propagation	35
3.2	MIMO Wireless Systems	35
3.2.1	Multipath Propagation	36
3.2.2	Channel Modelling of Wireless Systems	37
3.3	Clustering	38
3.3.1	Partition Method	39
3.4	Deep Learning	39
3.4.1	Activation Functions	40
3.4.2	Optimizer	40
3.5	Cluster Validity Index	40
3.6	Summary	41



This chapter consists of the concepts and theories implemented to effectively achieve the objectives that supervised the development of the study methodology. The root concept of wireless communication then, MIMO system and multipath propagation are covered. Followed by, data representation, and clustering principles and parameters. The multiple concepts connected to channel modeling have also been covered, along with the k -DAE Algorithm. Also, the CVI for accuracy of cluster has been discussed.

3.1 Radio Wave Propagation

Wave propagation is the underlying concept behind wireless transmission and technology. [Frenzel, 2007] emphasized, wave propagation happens when the radio signal has been radiated by the transmitter antenna which travel on the free space to send to a receiver antenna. During propagation, the energy of the signal decreases given the distance from the transmitter. There are several factors that decrease the signal energy level, such as objects that can be encountered in the environment, trees, buildings, walls, etc. Hence, these factors form diffraction, reflection, and refraction of signal that causes the signal to have different paths.

3.2 MIMO Wireless Systems

Throughout the past decades, the advancement of communication over wireless networks technologies such as Low-Earth Orbit (LEO) satellites, Wireless Fidelity (Wi-Fi) 6E, and 5G has extensively altered the pace of life. One critical technology driving the development of energy-efficient 5G systems is MIMO. MIMO systems have revolutionized wireless communication by significantly increasing data speed, connection stability, and spectrum efficiency. To help develop future applications in wireless communications and automated mode of transportation,



MIMO antenna technology is used because it is capable to offer a significant data rate both in terms of spectral and energy efficiency without requiring extra bandwidth [Tran-Huy et al., 2024].

In a propagation scenario, channel models are essential for wireless system design in MIMO. With precise channel models, the function of wireless systems in the environment could be evaluated and developed even in the absence of technical implementation [Trinidad and Materum, 2023]. An understanding of the signal's behavior and the ways in which various elements, including noise, interference, and bandwidth, impact it, is obtained through the simulations conducted in the channel model scenarios.

3.2.1 Multipath Propagation

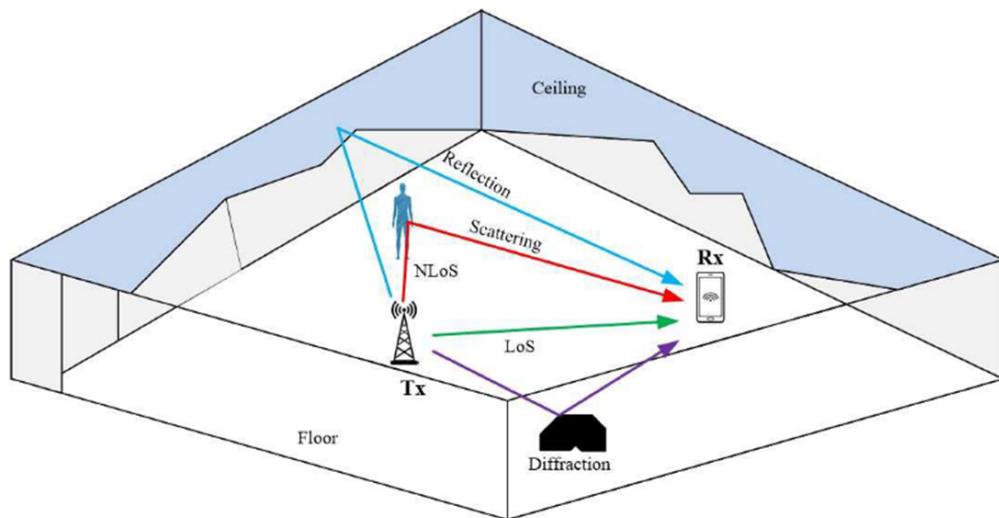


Fig. 3.1 Basic wireless propagation phenomena in confined environment
[Saleem et al., 2023]

Radio waves that encounter obstacles along their transmission path reflect, diffraction, and scatter, leading to multipath propagation [Faruque, 2017]. There are variations in signal strength due to the random phase and amplitudes of various MPCs, which results in small-scale fading. As electromagnetic waves go



through linear media, they can be scattered, absorbed, and refracted. This has the effect of allowing beams from a single transmitter to concurrently arrive at a receiver from several directions with varying Doppler shifts, delays, intensities, and polarizations. Due to their mobility and infrequently totally clear propagation paths, cellular phones frequently show significant multipath and fading effects.

A multipath vector x_ℓ is described by the equation:

$$x_\ell = \begin{bmatrix} \phi_{\ell,AoD} & \theta_{\ell,AoD} & \phi_{\ell,AoA} & \theta_{\ell,AoA} & \tau_\ell \end{bmatrix}^T \quad (3.1)$$

Which consists of azimuth angle of departure $\phi_{\ell,AoD}$ (AOD), elevation angle of AOD $\theta_{\ell,AoD}$, azimuth angle of arrival $\phi_{\ell,AoA}$ (AOA), elevation angle of AOA $\theta_{\ell,AoA}$, and multipath delay τ_ℓ . Since the vector is considered spherical, they are transformed in rectangular form using directional cosine.

$$\begin{aligned} x_x &= \sin \theta \cos \phi \\ x_y &= \sin \theta \sin \phi \\ x_z &= \cos \theta \end{aligned} \quad (3.2)$$

After the transformation, the equation below describes the vector consisting of seven dimensions of the MPC variables that are properly cluster using the clustering algorithm.

$$x_\ell = \begin{bmatrix} X_{\ell,x,AoD} & X_{\ell,y,AoD} & X_{\ell,z,AoD} & X_{\ell,x,AoA} & X_{\ell,y,AoA} & X_{\ell,z,AoA} & \tau_\ell \end{bmatrix}^T \quad (3.3)$$

3.2.2 Channel Modelling of Wireless Systems

Designing wireless communication systems requires a basic understanding of channel modeling. It is a standard procedure to collect an immense quantity of channel measurement data and then use statistical techniques to develop suitable channel models [Aldossari and Chen, 2019].

Effective wireless communication networks need the use of channel models, which provide an understanding of the communication environments. The channel



model is the mathematical formula that characterizes the behavior of the channel during the transmission of wireless signals. It is also known as the model of the medium between the transmitter and the receiver. Wireless channel models have been identified through a variety of methodologies and technologies since standardized approaches have not yet been established. Channel models fall into two categories: stochastic and deterministic. A stochastic model incorporates unpredictability in the process of creating solutions based on assumptions, whereas a deterministic model makes these hypotheses with certainty.

In a propagation scenario, channel models are essential for wireless system design. With precise channel models, the performance of wireless systems in the environment may be assessed and simulated even in the absence of hardware implementation [Trinidad and Materum, 2023]. An understanding of the signal's behavior and the ways in which various elements, including noise interference, and bandwidth, impact it, is obtained through the simulations conducted in the channel model scenarios.

3.3 Clustering

Clustering is part of unsupervised learning wherein the data is processed as unlabeled. Unsupervised learning is used for data without classification where it can imply different understanding of the results. Furthermore, clustering is dividing the samples in groups such that each member of the cluster is similar, but each cluster is different from one another. In these groups the data can be defined based on their properties and what makes them similar [Jin and Han, 2010]. There are several clustering algorithms, here is a well-known root algorithm used in this study.



3.3.1 Partition Method

Partitioning methods in clustering is an algorithm that divide the dataset into clusters, where each datapoint is one cluster. One cluster means nearest data points in the cluster centroid. One of the most popular partitioning algorithms is k -means [Tan et al., 2018]. It is by far the simplest clustering algorithm. This algorithm's purpose to determine the optimal amount of k clusters output, meaning finding the number of groups within the data, that could cluster the data with high accuracy. In k -means, each point on the data will be assigned to the k cluster distance it is nearest. The k clusters are normally found using silhouette method or elbow method [MacQueen et al., 1967].

3.4 Deep Learning

Within the field of machine learning is deep learning that applies neural systems consisting of several stages for gathering data representations. This has gained popularity and demonstrated notable applicability success due to its automatically learning detail interpretations and trends from complex data sets [LeCun et al., 2015].

Computational models driven by the architecture and performance of the human brain are called neural networks. Traditional neural networks are expanded using deep learning by adding more layers, enabling to learn hierachal representations of data [Ia, 2016]. Autoencoder algorithm is a neural network that is part of DR which reduces the dimensions of the data serving as the representation its representation. It consists of three parts, encoder, embedding layer, and decoder. Its goal is to find the best accuracy given less reconstruction error [He et al., 2018].

The emergence of a new clustering algorithm incorporating deep learning led this study to be conducted. The clustering algorithm first trains a single node



called neurons then, it will be encoded and be reduced to a bottle neck to specific number of k clusters. Afterwards, it will be decoded and the reconstruction error from the input should be less to find effective clustering [OPOCHINSKY ET AL., 2020].

The results of the clustering algorithm are notable, the researchers used three data sets MNIST, Fashion, and USPS with the following results:

TABLE 3.1 k -DAE RESULTS [OPOCHINSKY ET AL., 2020]

CVI	MNIST	Fashion	USPS
NMI	0.86	0.65	0.80
ARI	0.82	0.48	0.71
ACC	0.88	0.60	0.77

3.4.1 Activation Functions

Neural networks are presented with non-linear activation functions, which enables them to learn detailed patterns and representations from the input data. They determine the output of individual neurons and the overall behavior of the network [Nair and Hinton, 2010].

3.4.2 Optimizer

An optimizer is an algorithm that modifies a neural network's parameters (weights and biases) while on training to decrease the reduced function and enhance model performance. They adjust the parameters based on the gradients calculated based on the reduced function [Kingma and Ba, 2014] .

3.5 Cluster Validity Index

The CVI can be applied to mathematically verify the clustering accuracy. CVI is a metric that is used to evaluate the clustering efficiency of an algorithm which can



be classified by relative, internal, and external indices. External validity indices are used for clustering with fundamental accuracy of the clusters and comparing it with the function of the results. Internal validity indices examine the structure of the clusters based solely on their geometric properties of the original data. This study used external validation to compare the accuracy of cluster membership to the ground truth, specifically, the JI. Moreover, the different external validation indices were discussed in Chapter 4.

3.6 Summary

This study is developed through the topics of MIMO, channel modelling, k -DAE clustering algorithm, parameters in autoencoders, and CVIs. Through direct involvement and clustering by utilizing the k -DAE algorithm to the data sets and validating it by implementing CVI, the quality of the clusters can be improved.



Chapter 4

DESIGN CONSIDERATIONS

4.1	Channel Model Datasets	43
4.1.1	COST 2100 Channel Model	43
4.1.2	IMT-2020 Channel Model	44
4.2	k-Deep Autoencoder Clustering	45
4.2.1	Clustering Algorithm	45
4.3	Clustering Validation Index	46
4.3.1	Jaccard Index	48
4.3.2	Normalize Mutual Index	49
4.3.3	Adjusted Rand Index	49
4.3.4	Clustering Accuracy	50
4.4	Optimizers	51
4.4.1	Adaptive Momentum (ADAM)	51
4.4.2	Stochastic Gradient Descent (SGD)	51
4.5	Activation Functions	51
4.5.1	Rectified Linear Unit (ReLU)	52
4.5.2	Sigmoid	53
4.6	Python	54
4.6.1	Python IDE	54
4.6.2	Versions and Libraries	55
4.6.3	Flowchart of the Code	56
4.7	Summary	57



The previous chapter explains the theoretical considerations of this study. The design considerations are covered in this chapter where it is set to meet the objectives in Chapter 1. This includes the channel model datasets that are standards that need to be clustered. The Activation functions and Optimizers implemented in this study, and the formula of CVIs. Finally, the software that that is used as the user interface of the researchers.

4.1 Channel Model Datasets

TABLE 4.1 CHANNEL GENERATED MODELS' COMPARISON

Channel Models	Number of Channel Scenarios	Modelling Approach	Channel Bandwidth (GHz)
C2CM	8	GBSM	<6
IMT-2020	8	GBSM	0.5-100

4.1.1 COST 2100 Channel Model

An extension of the COST 273 model, the C2CM is a GBSM channel model built on the concept of clusters that could replicate MIMO channels' behavior across space, time, and frequency. Initially, it was intended to simulate a radio channel between mobile stations with multiple antennas and static stations with multiple antennas. Azimuth Angle of Departure (AoD), Elevation Angle of Departure (EoD), Azimuth Angle of Arrival (AoA), and Elevation Angle of Arrival (EoA) are the characteristics that define the MPCs [Liu et al., 2012]. The channel scenarios are the following:

1. Indoor B1 LOS Single Link
2. Indoor B2 LOS Single Link



3. Semi-Urban B1 LOS Single Link
4. Semi-Urban B2 LOS Single Link
5. Semi-Urban B1 NLOS Single Link
6. Semi-Urban B2 NLOS Single Link
7. Semi-Urban B1 LOS Multiple Links
8. Semi-Urban B2 LOS Multiple Links

4.1.2 IMT-2020 Channel Model

The term 5G refers to IMT-2020 which is a set of standards and specifications for 5G next generations of mobile broadband and IoT connectivity set by the International Telecommunication Union (ITU). IMT-2020 is responsible for establishing the worldwide criteria for 5G networks, which includes parameters such as peak data rate, user-experienced data rate, area traffic capacity, mobility, latency, energy efficiency, and density of connections.

The IMT-2020 channel model enables 3D MIMO by expanding the elevation domain. Based on the model, there are matching scenarios for the 5G use cases. For example, the InH improved wireless internet access Enhanced Mobile Broadband (eMBB), where UMa and UMi are the dense urban eMBB layers, with RMa serving as the rural eMBB representation.

An additional module below 6 GHz is a map-based hybrid channel module based on the ray-tracing model, and the IMT-2020 is also regarded as a GBSM for its primary module. Regarding its Service Switching Point (SSP)s, the hold-up and cluster power is represented by an exponential power delay profile and exponential delay distribution, respectively, while the power angle spectrum in the azimuth of each cluster is represented by a wrapped Gaussian or Laplacian. The Poisson



distribution is followed by the number of clusters produced by the IMT-2020. It also considers the impact of vegetation on mm-wave bands [Blanza et al., 2023].

The IMT-2020 dataset considered in this study consisting of 11 different scenarios with 30 sheets each are listed as follows:

1. InH A LOS propagation conditions.
2. InH A NLOS propagation conditions.
3. RMa A LOS propagation conditions.
4. RMa A NLOS propagation conditions.
5. UMa A LOS propagation conditions.
6. UMa A NLOS propagation conditions.
7. UMi A LOS propagation conditions.
8. UMi A NLOS propagation conditions.

4.2 k-Deep Autoencoder Clustering

4.2.1 Clustering Algorithm

The C2CM and IMT-2020 dataset were used by the researchers as an input to the k -DAE method. k -DAE algorithm was implemented k -means algorithm using neural networks. Instead of representing clusters by a centroid, it is represented by an autoencoder. It outperforms different autoencoder-based clustering algorithms with notable results (refer to 2.2). Hence, if the data is properly clustered, the points that are in the cluster are similar [Opochinsky et al., 2020]. There are three steps in the whole algorithm. These procedures are applied to every sheet in the datasets. The inputs initially undergo preliminary training. Following the



initialization of information, the autoencoder automatically cluster the datasets fed in the algorithm. The final stage, the k -DAE, received the autoencoder's output clusters and this clusters are compared to the ground truth using different CVIs. Also, the Reconstruction Error (RE)'s role is to know the performance of the system if the data are properly reconstructed, meaning the data from input and output are similar. If the value is approaching 0 the autoencoder is optimized.

The clustering Algorithm is summarized on the Fig 4.1 below:

Goal: clustering $x_1, \dots, x_n \in \mathbb{R}^d$ into k clusters.

Network components:

- A set of autoencoders (one for each cluster):

$$x \rightarrow \hat{x}(i) = f_i(x; \theta_i), \quad i = 1, \dots, k$$

Pre-training:

- Train a single autoencoder for the entire dataset.
- Apply a k -means algorithm in the embedded space.
- Use the k -means clustering to initialize the network parameters.

Training: clustering is obtained by minimizing the reconstruction error:

$$L(\theta_1, \dots, \theta_k) = \sum_{t=1}^n \min_i d(x_t, \hat{x}_t(i))$$

The final (hard) clustering is:

$$\hat{c}_t = \arg \min_{i=1}^k d(x_t, \hat{x}_t(i)), \quad t = 1, \dots, n$$

Fig. 4.1 k -DAE Clustering Algorithm
[Opochinsky et al., 2020]

4.3 Clustering Validation Index

The performance of the clustering results was assessed using a metric called the CVI. There are three different forms of CVI which are the relative, internal, and external indices. External validity indices compare the performance of the findings and are based on prior knowledge or ground truth of the clusters



[Trinidad and Materum, 2023]. External validity indices were used in this study.

The following are the external validity CVIs:

TABLE 4.2 SUMMARY OF CVIs

CVI	Definition	Equation	Range
NMI	Measures the amount of information shared between the predicted clustering and the ground truth clustering, normalized to ensure the result is between 0 and 1. NMI is derived from Mutual Information but is normalized to account for the size of the clusters and the total number of data points [Vinh et al., 2009]	$\text{NMI } (C_{\text{ref}}, C_{\text{calc}}) = \frac{MI(C_{\text{ref}}, C_{\text{calc}})}{\max(H(C_{\text{ref}}), H(C_{\text{calc}}))}$	[0,1]
ARI	It is an improved version of the Rand Index that adjusts for the possibility of random agreements. The ARI measures the similarity between two clusterings by adjusting the Rand Index (RI) for the chance grouping of elements [Hubert and Arabie, 1985]. The RI counts the number of pairs of elements that are either in the same cluster or in different clusters in both the predicted and true clusterings. The ARI corrects this measure to account for the expected similarity of random clusterings, providing a more reliable comparison [Vinh et al., 2009]	$\text{ARI} = \frac{\frac{M_{11} + M_{00}}{M_{00} + M_{01} + M_{10} + M_{11}} - \frac{M_{11} + M_{00}}{\binom{M}{2}}}{1 - \frac{M_{11} + M_{00}}{\binom{M}{2}}}$	[-1,1]
ACC	ACC measures how accurately the predicted clusters match the true labels by finding the best one-to-one correspondence between cluster labels and ground truth labels, and then computing the percentage of correctly assigned labels [Yang and Liu, 1999].	$\text{ACC} = \frac{\sum_{i=1}^N \delta(C_{\text{ref}}, \text{map}(C_{\text{calc}}))}{N}$	[0,1]

Continued on next page

*Continued from previous page*

CVI	Definition	Equation	Range
JI	Assesses the similarity between two sets of data points by comparing the intersection and union of the sets of point pairs that are grouped together in both clusterings. It quantifies how similar the sets of pairwise relationships are between the clusterings [Jaccard, 1912].	$\eta_{\text{Jac}} = \left \frac{C_{\text{ref}} \cap C_{\text{calc}}}{C_{\text{ref}} \cup C_{\text{calc}}} \right = \frac{M_{11}}{M_{11} + M_{10} + M_{01}}$	[0,1]

Differentiation among clustering evaluation metrics can be understood through their specific focuses and methodologies. NMI centers on mutual information and entropy, providing a normalized measure of the shared information between clusters and labels. ARI adjusts for randomness, offering a robust measure against chance groupings by considering the possibility of random assignments. ACC is based on the accuracy after optimal mapping between clusters and true labels, directly assessing how well the clusters correspond to the actual labels. Finally, the JI evaluates clustering by examining the similarity of sets of clustered point pairs, emphasizing pairwise relationships within the data.

4.3.1 Jaccard Index

JI determines how similar two data sets are to one another. When applying different clustering approaches to the same dataset, it can be utilized in multipath clustering to analyze the similarity of multiple cluster assignments [Fahad et al., 2014].

The definition of the similarity measure is:

$$\eta_{\text{Jac}} = \left| \frac{C_{\text{ref}} \cap C_{\text{calc}}}{C_{\text{ref}} \cup C_{\text{calc}}} \right| = \frac{M_{11}}{M_{11} + M_{10} + M_{01}}, \in [0, 1] \quad (4.1)$$

where $|\bullet|$ stands for cardinality,



C_{ref} is the reference clusters, and
 C_{calc} the calculated clusters, the membership of the clusters in JI
 M_{11} is the number of members present in the C_{ref} and C_{calc} clusters,
 M_{10} is the number of members present in the C_{calc} but not in C_{ref} clusters, and
 M_{01} is the number of members that are not present in the C_{calc} but present in
 C_{ref} clusters [Blanza and Materum, 2019b].
The result ranges between 0 for an empty set and 1 for a perfect match.

4.3.2 Normalize Mutual Index

The measure of similarity between two clustering findings that considers both cluster sizes and cluster assignments is called the NMI. In multipath clustering, the NMI is a statistic that is used to compare different community discovery strategies.

$$\text{NMI } (C_{\text{ref}}, C_{\text{calc}}) = \frac{MI(C_{\text{ref}}, C_{\text{calc}})}{\max(H(C_{\text{ref}}), H(C_{\text{calc}}))}, \in [0, 1] \quad (4.2)$$

where $H(C_{\text{ref}})$ and $H(C_{\text{calc}})$ are the entropies of C_{ref} and C_{calc}

C_{ref} represents the true or reference labels C_{calc} represents the predicted or calculated labels.

The NMI value of 1 show that the clustering solution matches the predefined label assignments perfectly, and close to 0 if the matching is short.

4.3.3 Adjusted Rand Index

In cluster analysis, the ARI is frequently employed to quantify the level of agreement between two data partitions. To gain a deeper understanding of this index, there has been interest in examining the extremes of agreement and disagreement under various conditions since its establishment [Chacón and Rastrojo, 2023]. The ARI considers the quantity of instances that appear in separate clusters as well as those that occur within the same cluster. The predicted value of such



a validation measure is not zero when comparing partitions [Fahad et al., 2014].

One definition of the ARI is:

$$ARI = \frac{M_{11} + M_{00}}{M_{00} + M_{01} + M_{10} + M_{11}} = \frac{M_{11} + M_{00}}{\binom{M}{2}}, \in [-1, 1] \quad (4.3)$$

where M_{11} is the number of pairs that are part of the same cluster

M_{00} is the number of instances that are part of a different cluster

M_{01} is the number of pairs that are in the reference cluster but not in the clustering output

M_{10} is the number of pairs that are in the output cluster but not in the reference cluster [Fahad et al., 2014].

The value of ARI ranges from -1 to 1, with higher value indicating that all data instances are clustered correctly, and the cluster holds only pure instances.

4.3.4 Clustering Accuracy

ACC that assesses the percentage of data points for which the produced clusters can be used to assign to ground-truth classes. It is employed to ascertain the overall correctness of a clustering solution [OPOCHINSKY et al., 2020]. In multipath clustering, ACC is applied to assess the accuracy of clustering of the clustering results obtained from many routes or runs of the clustering algorithm. ACC can be used in multipath clustering to compare different clustering methods or parameter settings and assess the stability of the clustering results over a variety of runs or datasets. A higher ACC score denotes better clustering accuracy and can be used to guide the selection of the optimal clustering parameters or procedure. The ACC can be defined as:

$$ACC = \frac{\sum_{I=1}^N \delta(C_{\text{ref}}, \text{map}(C_{\text{calc}}))}{N}, \in [0, 1] \quad (4.4)$$

where N is the total number of data and $\delta(x, y)$ is the delta function that equals



one if $x = y$ and equals zero otherwise, and

$\text{map}(C_{\text{calc}})$ is the permutation mapping function that maps each cluster label C_{calc} to the equivalent label from C_{ref} [Cai et al., 2010].

ACC values are between 0 and 1, where 0 represents an ineffective clustering algorithm and 1 for an effective clustering algorithm.

4.4 Optimizers

4.4.1 Adaptive Momentum (ADAM)

ADAM is an adaptive learning rate optimization algorithm that combines the advantages of Adaptive Gradient (AdaGrad) and Root Mean Square Propagation (RMSProp) [Kingma and Ba, 2014]. It is well-suited for training deep neural networks and offers fast convergence. ADAM is the optimization used by the previous study [Opochinsky et al., 2020]. It is an appropriate match for training deep neural networks and offers fast convergence.

4.4.2 Stochastic Gradient Descent (SGD)

SGD is a traditional optimization technique, to modify the model's parameters according to the loss function's gradient regarding the training data [Ruder, 2016]. SGD is computationally efficient and widely used in machine learning. This study introduced this optimization algorithm in clustering for channel model generated multipath components.

4.5 Activation Functions

An activation function is a mathematical operation used in artificial neural networks. It acts like a gate, allowing information to pass through only when the



input is significant enough to be mapped on the function. Part of the scope of this research is modifying activation functions using ReLU and Sigmoid functions of clustering model k -DAE.

4.5.1 Rectified Linear Unit (ReLU)

ReLU is a common activation function in deep learning models. It outputs the input if it's positive, otherwise zero. This is mathematically represented as:

$$f(x) = \max(0, x) \quad (4.5)$$

ReLU helps minimize the vanishing gradient issue that arises when using other functions, such as the hyperbolic tangent or sigmoid, to facilitate the training of deep neural networks.

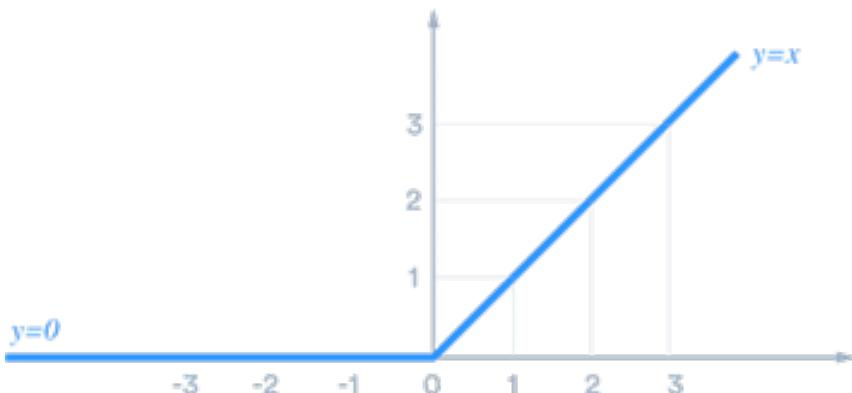


Fig. 4.2 Rectifier Linear Unit function
[Szandała, 2021]

ReLU is a simple but efficient activation function that preserves positive values while setting negative values to zero [Nair and Hinton, 2010]. It helps alleviate the vanishing gradient problem and simplifies the training process. This activation function employed from the previous investigation was ReLU [Opochinsky et al., 2020].



4.5.2 Sigmoid

The logistic function, which is also known as the sigmoid function, is a popular activation function often employed at the last layer in the binary classification models and occasionally in the neural networks' concealed layers. The mathematical representation of the sigmoid function is:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (4.6)$$

In this equation, x is the input to the function, and $\sigma(x)$ is the output. The sigmoid function can take any real number and compress it into a range between 0 and 1. The result of the sigmoid function can be understood as a probability, which is particularly beneficial in binary classification tasks where the objective is to determine if an input is a member of one of two possible classes.

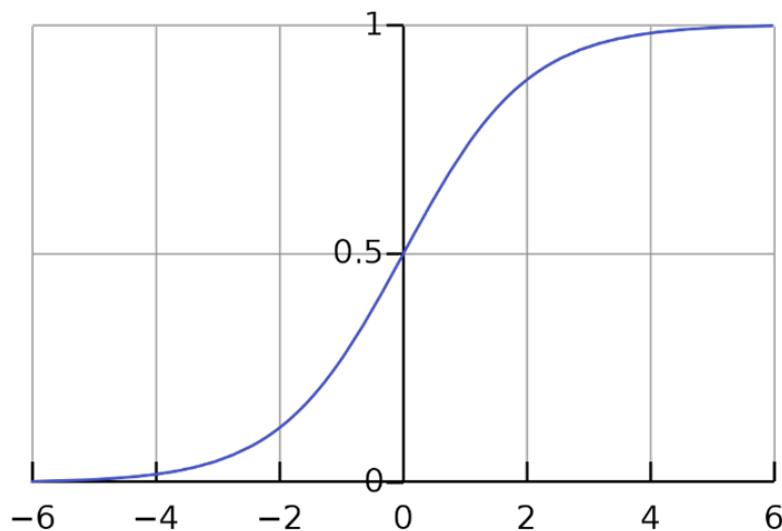


Fig. 4.3 The demonstration of a Sigmoid Activation Function
[Szandała, 2021]

Moreover, sigmoid squashes the output of each neuron to the range [0,1]. It is particularly useful in binary classification tasks and for producing probabilities as output [Glorot et al., 2011]. In this investigation, sigmoid is used due to the



presence of negative values of the datasets in the channel generated multipath components.

4.6 Python

Python is a universal, high-level, open-source programming language designed with an emphasis with its syntax and code readability that allows the expression of concepts in fewer lines make it one of the popular choices among programmers. With its comprehensive standard libraries, Python is used in different applications such as building software, websites, machine learning data analysis, scientific computing, artificial intelligence, machine learning, therefore the latter was used in doing the data gathering for our thesis.

4.6.1 Python IDE

Python has various IDEs that are made for different needs and preferences. For this research, we used the Spyder IDE – Spyder is a Python-specific IDE that offers a comprehensive platform optimized for scientific workflows and data analysis. Spyder integrates with SciPy, NumPy, and Matplotlib which are Python's data science libraries. Spyder environment features a variable explorer, IPython console, and integrated documentation.

- Code Editor: Spyder code editor features syntax highlighting, code completion, and built-in code analysis support.
- Interactive Console: The IDE includes an interactive IPython console, enabling users to run code in real-time, execute scripts, and explore data with ease.



- Variable Explorer: This feature allows the user to examine and interact with variables within the workspace that is proven to be useful especially for data analysis and debugging.
- Debugger: The built-in debugger allows the programmer to modify the code, set breakpoints, and inspect variables to identify and rectify issues in the code.
- Plots and Visualization: The IDE has inherent support for plotting and data visualization using libraries like Matplotlib and other popular plotting tools.
- Conda Integration: Spyder integrates smoothly with Conda, a widely used package and environment management system, simplifying project dependency management.

4.6.2 Versions and Libraries

TABLE 4.3 SUMMARY OF VERSIONS AND LIBRARIES INVOLVED IN THE STUDY

Library	Function	Version
Python	Programming language	3.11.2
TensorFlow	Open-source software library for machine learning and AI (artificial intelligence)	2.14.0
TensorFlow-estimator	A high level TensorFlow API that encapsulates training, evaluation, prediction and export	2.14.0
Keras	Provides a Python interface for artificial neural networks. This works with TensorFlow	2.14.0
NumPy	Used for working with multi-dimensional arrays and matrices and high-level mathematical functions	1.24.2
scikit-learn	Provides tools for predictive data analysis	1.3.2
SciPy	Provides optimizations of algorithms	1.10.1
openpyxl	Used to be able to read or write on Excel files.	3.1.2
pandas	Data analysis and manipulation	2.2.0



4.6.3 Flowchart of the Code

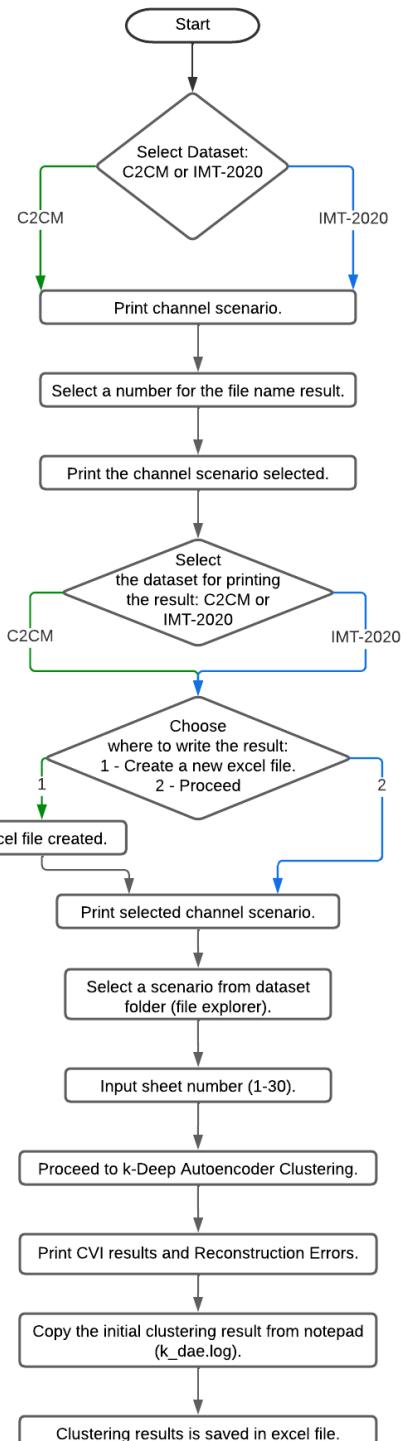


Fig. 4.4 Flowchart of the *k*-DAE code



4.7 Summary

This chapter discussed the channel model datasets, k -DAE, CVIs, optimizers, activation functions, Python usage, and the conceptual framework that is used in implementing the study. The mentioned concepts are defined to meet the objectives of the study.



Chapter 5

METHODOLOGY

5.1	Implementation	60
5.1.1	Dataset Preparation	60
5.1.2	Code Modification	60
5.1.3	Optimizer	61
5.1.4	Activation Function	62
5.2	Evaluation	62
5.2.1	Evaluation of Results using Jaccard index	62
5.2.2	Statistical Treatment of Data	62
5.3	Summary	62



This chapter discusses the proper implementation of the objectives in Chapter 1. This chapter is divided into two parts, the implementation which include the instrument to be used, how to modify the code, the optimizers and activation function used, and the data clustering of the two types of channel model. After clustering, evaluation of the results of clustering algorithm are analyzed using JI and other statistical treatments.

TABLE 5.1 SUMMARY OF METHODS FOR SATISFYING THE OBJECTIVES

Objective	Methods	Location
GO: To cluster C2CM and IMT-2020 multipath components using modified deep autoencoder algorithm.	Applied the k -DAE algorithm in clustering the datasets from C2CM and IMT-2020 channel models.	Sec 5.3 p. 62
SO1: To have a sufficient activation function for deep autoencoder clustering algorithm.	Modified the k -DAE clustering algorithm code using the configurations stated on Table 5.2 to cluster MPCs.	Sec 5.1.2 p. 60
SO2: To achieve close or higher JI in clustering C2CM and IMT-2020.	Comparison of the JI results of different configurations in Table 5.2 using the One-Way ANOVA and ECDF.	Sec 5.2.2 p. 62
SO3: To measure the JI performance and the associative correlation between autoencoder reconstruction error and JI.	Observed the relationship between the JI and reconstruction error.	Sec 5.2.2 p. 62

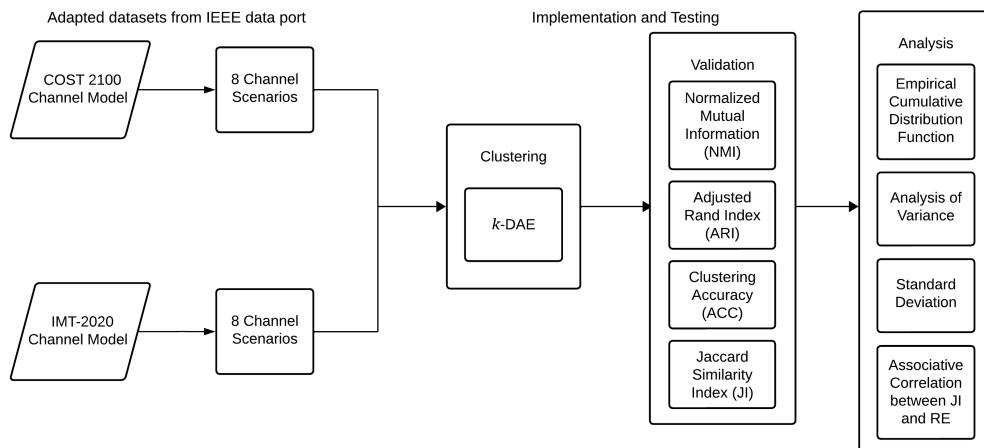


Fig. 5.1 Block Diagram of Methodology of the Study

5.1 Implementation

5.1.1 Dataset Preparation

Datasets were from C2CM and IMT-2020 Channel models that were extracted from IEEE DataPort. The C2CM involves 8 channel scenarios which includes two indoor and six outdoor channel scenarios. Also, IMT-2020 involves 8 channel scenarios and includes two indoor, three rural, and six urban scenarios. Furthermore, each scenario contains 30 individual sheets. These sheets represent a single snapshot, as a result having 480 total number of data sheets. Each channel scenario are be fed to the algorithm to undergo clustering one data sheet at a time. The datasets are already Directional Cosine Transformed, thus there is no need for normalization.

5.1.2 Code Modification

The Algorithm was adapted and accessed on Git-Hub, it is an opensource community of programmers on collaboration in software development. The researchers



modified the code with the following functions:

1. Incorporate C2CM and IMT-2020 datasets.
2. Select excel files and individual sheets using pandas library.
3. Define the number of neurons, layers, and epoch.
4. Test the sufficient combination activation functions (ReLU and Sigmoid). The following are the configurations:

TABLE 5.2 DIFFERENT CONFIGURATIONS IMPLEMENTED IN THE STUDY

Configuration	Encoder	Embedding Layer	Decoder	Optimizer
1	ReLU	ReLU	Sigmoid	ADAM
2	Sigmoid	Sigmoid	Sigmoid	ADAM
3	ReLU	ReLU	Sigmoid	SGD
4	Sigmoid	Sigmoid	Sigmoid	SGD

5. Extract Reconstruction errors.
6. Incorporate JI for validation of clusters.
7. Save the data clusters, JI, and reconstruction errors on a separate excel file.

5.1.3 Optimizer

The implementation of clustering in k -DAE incorporates two optimization algorithms: ADAM and SGD. The ADAM or SGD optimizer in k -DAE, is integrated into the training process by using TensorFlow library in Python language. Additionally, binary cross-entropy is also used during the training for the loss function.



5.1.4 Activation Function

In a k -DAE algorithm, ReLU and Sigmoid activation functions were implemented using the TensorFlow library in Python language. In either case, the activation function is specified using the Dense layer. The input dimension and encoding dimension is defined based on the k -DAE architecture.

5.2 Evaluation

5.2.1 Evaluation of Results using Jaccard index

This study used JI to compare the accuracy of cluster membership to the ground truth. The similarities of cluster labels were calculated based on the ground truth from the datasets of C2CM and IMT-2020. Different JI results of each configuration were also compared.

5.2.2 Statistical Treatment of Data

After obtaining the JI of the clustering algorithms, statistical analysis, and visualization using Box Plots was used for each scenario in the C2CM and IMT-2020 datasets. See Section 4.3 for more details on the CVIs.

In comparing the results of the JI of the algorithms used in clustering datasets from the channel models, One-Way ANOVA, SD and ECDF are used. Additionally, Scatter Plots were used to visualize the comparison of the JI results.

5.3 Summary

This chapter discusses the outline on the fulfillment of the objectives in Chapter 1. The implementation on modifying the code that is adapted. Activation



functions and Optimizers were taken note for experimentation of different configuration on clustering the data. Also, the statistical treatment and visualization on analyzing the data gathered from the clustered data.



Chapter 6

RESULTS AND DISCUSSION

6.1	Implementation of k-DAE algorithm on C2CM and IMT-2020	67
6.1.1	Incorporation of required libraries and modules	67
6.1.2	Implementation of Channel Model Datasets	69
6.1.3	Implementation of Activation Functions and Optimizers	72
6.2	Clustering Accuracy of Modified k-DAE	74
6.2.1	Box Plots of CVIs of Different Configurations in C2CM	74
6.2.2	Box Plots of CVIs of Different Configurations in IMT-2020	78
6.2.3	Means and Standard Deviations of CVIs in C2CM	82
6.2.4	Means and Standard Deviations of CVIs in IMT-2020	84
6.3	Statistical Comparison of different Activation functions and Optimizers	85
6.3.1	Means and Standard Deviations of JIs in C2CM	86
6.3.2	Means and Standard Deviations of JIs in IMT-2020	87
6.3.3	ANOVA of JIs in C2CM	88
6.3.4	ANOVA of JIs in IMT-2020	91
6.3.5	Histogram on distribution of JI in C2CM and IMT-2020	95
6.3.6	ECDF of JIs in C2CM	99
6.3.7	ECDF of JIs in IMT-2020	102
6.4	Reconstruction Error	106
6.4.1	Correlations of CVIs of Different Configurations in C2CM	106
6.4.2	Correlations of CVIs of Different Configurations in IMT-2020	111
6.4.3	Correlation of JIs and REs in C2CM and IMT-2020	117
6.5	Summary	120



The objectives of this study were accomplished based on the concepts from the previous chapters. This chapter provides the implementation of the chosen channel models, activation functions, and optimizers. Correspondingly, the clustering accuracy scores obtained from simulation of the system, the comparison of JI results of different activation function and optimizer configurations, and statistics of the obtained CVIs. Finally, the associative correlation of JI and RE for each channel model. Table 6.1 presents the key results of the following objective in this study.

TABLE 6.1 SUMMARY OF RESULTS PER OBJECTIVE

Objective	Key Result	Location
GO: To cluster C2CM and IMT-2020 MPCs using modified deep autoencoder algorithm.	The integration of modified k -DAE algorithm was applied in clustering MPCs and it showed that the Sigmoid-Sigmoid-SGD configuration attained the highest accuracy among all the configurations in all the scenarios in C2CM and IMT-2020. Specifically, in C2CM all scenarios it achieved 0.6 in NMI, 0.29 in ARI, 0.43 in ACC, and 0.19 in JI. In IMT-2020 all scenarios, it achieved 0.03 in NMI, 0.01 in ARI, 0.10 in ACC, and 0.084 JI.	Sec 6.5 p. 120
SO1: To have a sufficient activation function for deep autoencoder clustering algorithm.	The modification of activation functions and optimizers of the k -DAE algorithm on the implementation of the C2CM and IMT-2020 datasets uncovered that sigmoid is the most sufficient activation function for k -DAE algorithm in clustering MPCs.	Sec 6.1 p. 67 Sec 6.2 p. 74

Continued on next page

*Continued from previous page*

Objective	Key Result	Location
SO2: To achieve close or higher JI in clustering C2CM and IMT-2020.	The combination of sigmoid and SGD configuration gave the highest JI result among the different configurations of k -DAE in clustering the C2CM and IMT-2020 datasets. Comparing the combination of Sigmoid to the previous configuration of k -DAE, it increased by 14.63% in 10th percentile, 30.34% in 50th percentile, and 50.70% for indoor scenarios in C2CM. For outdoor scenarios, it increased by 1.46% in 10th percentile, 1.46% in 50th percentile, and 1% in 90th percentile. For all scenarios, it increased by 1.59% in 10th percentile, 1.19% in 50th percentile, and 35.95% in 90th percentile in C2CM. Also, in IMT-2020, it increased by 1% in 10th percentile, 1.1% in 50th percentile, and 1.2 % in 90th percentile in indoor scenarios. For outdoor scenarios, it increased 0.2% for 10th percentile and 0.4% in 50th percentile. For all scenarios, it increased 0.4% for 10th percentile and 1% for 50th percentile. Thus, Sigmoid-Sigmoid-ADAM is sufficient among all configurations.	Sec 6.3 p.85
SO3: To measure the JI performance and the associative correlation between autoencoder RE and JI.	In C2CM, Sigmoid configurations showed a negative linear correlation of JI and RE with overall scenarios, indicating a negative linear relationship. Sigmoid-Sigmoid-ADAM displayed a weak correlation coefficient of -0.16, while Sigmoid-Sigmoid-ADAM displayed a moderate value of -0.5 correlation coefficient. Thus, in Sigmoid configurations, as the JI result increases, the corresponding RE decreases. However, for IMT-2020, correlation coefficients still vary depending on the scenario and the specific configuration used. The lack of statistical significance may indicate that the observed correlations could be due to random chance rather than a true relationship between configurations and measures.	Sec 6.4 p.106



6.1 Implementation of k-DAE algorithm on C2CM and IMT-2020

This section satisfied the first objective of this study. To modify and have a sufficient activation function for *k*-DAE Algorithm. The researchers modified the Python script in Spyder IDE using combination of ReLU and Sigmoid activation functions. The following subsections explained how the first objective was executed and attained.

6.1.1 Incorporation of required libraries and modules

The code algorithm was adapted from GitHub and accessed and run on Anaconda environment, particularly, Spyder IDE. Installation of several packages such as libraries and modules were installed on the devices that were used to run the code. The version of these packages is stated in Table 4.3, these were installed to properly run the algorithm.

The algorithm is divided into four python scripts: main.py, utils.py, autoencoder.py, and kdae.py. The (1) main.py script is responsible for compiling and running the code to execute the whole program. Also, the epochs and selection of datasets are in this script. In (2) utilities, the argument for the display and selection of datasets from two channel models are coded and the opening of file explorer for accessing the datasets. The (3) autoencoder script, Activation functions on encoder, decoder and output layer are initiated. Also, the inclusion of optimizer. Lastly, the (4) *k*-DAE algorithm code itself. Originally, the code adapted without modification contained MNIST, Fashion, and USPS datasets. Hence, the algorithm can choose between the three datasets. After installing the required libraries, the researchers checked if the algorithm and scripts are functional. Subsequently, the researchers decided to modify the code by removing the previous datasets



then adding the two datasets in main run file, “cost” for C2CM and “imt2020” for IMT-2020.

Listing 6.1 Under main run file script:

```
1 if __name__ == '__main__':
2     parser = argparse.ArgumentParser()
3
4     parser.add_argument('-sd', '--save_dir',
5                         type=str,
6                         default='save',
7                         help='path to save')
8
9     parser.add_argument('-dn', '--dataset_name',
10                        choices=['cost', 'imt2020'],
11                        default='imt2020',
12                        help='dataset name [cost, imt2020]')
```

Listing 6.2 Under utilities script:

```
1 if data_name == 'cost':
2     print('Select an Excel File in the File Explorer\n')
3     Tk().withdraw()
4     filepath = filedialog.askopenfilename(
5         initialdir=r'C:\Users\Mark Macapagal\Desktop\MasterCode2\
6             DATASETS',
7         title="Open Excel file",
8         filetypes=(("Excel files", "*.xlsx"), ("all files", "*.*")))
9
10    if not filepath:
11        print("NO FILE SELECTED.")
12        return filepath
13    elif filepath:
14        sheet_Num=str(input('Enter Sheet Number: '))
15        print()
16        t=pd.read_excel(filepath,sheet_name='Sheet'+sheet_Num)
17        print(t)
18        data=t.to_numpy()
19        x_train=data[:, :-2]
20        y_train = relabel(data[:, -1])
21        return [x_train, y_train]
22
23    if data_name == 'imt2020':
24        print('Select an Excel File in the File Explorer\n')
25        Tk().withdraw()
26        filepath = filedialog.askopenfilename(
27            initialdir=r'C:\Users\Mark Macapagal\Desktop\MasterCode2\
28                DATASETS',
29            title="Open Excel file",
30            filetypes=(("Excel files", "*.xlsx"), ("all files", "*.*")))
31
32        if not filepath:
33            print("NO FILE SELECTED.")
34            return filepath
35        elif filepath:
36            sheet_Num=str(input('Enter Sheet Number: '))
37            print()
38            t=pd.read_excel(filepath,sheet_name='Sheet'+sheet_Num)
39            print(t)
40            data=t.to_numpy()
41            x_train=data[:, :-2]
42            y_train = relabel(data[:, -1])
43            return [x_train, y_train]
```



6.1.2 Implementation of Channel Model Datasets

The modification of the code starts by properly accessing and reading the two Channel Model Datasets stored in excel files. Also, correct assignment of variables was applied. The importation of pandas and os libraries was initiated for reading of excel file format.

Listing 6.3 Under utilities script:

```
1 #import excel file
2 import pandas as pd
3 #Removing Initial Cluster
4 import os
```

After the data were read, the data were converted to numerical python-format for each Channel Models with their respective variable assignments to integrate them to the system for clustering. Also, since IMT-2020 is first time to be implemented, the code was modified adding the channel model.

Listing 6.4 Under utilities script for IMT-2020:

```
1 Tk().withdraw()
2 filepath = filedialog.askopenfilename(
3     initialdir=r'C:\Users\Mark\Macapagal\Desktop\MasterCode2\
4         DATASETS',
5     title="Open Excel file",
6     filetypes=(( "Excel files", "*.xlsx"), ("all files", "*.*"))
7 )
8 if not filepath:
9     print("NO FILE SELECTED.")
10    return filepath
11 elif filepath:
12     sheet_Num=str(input('Enter Sheet Number: '))
13     print()
14     t=pd.read_excel(filepath,sheet_name='Sheet'+sheet_Num)
15     print(t)
16     data=t.to_numpy()
17     x_train=data[:, :-2]
18     y_train = relabel(data[:, -1])
19     return [x_train, y_train]
```

After the following modifications, the C2CM and IMT-2020 datasets from IEEE DataPort that were whitened and Directional Cosine Transform (DCT) transformed (normalized) were fed to the system to check if the data from the excel files are read and variables are properly assigned, respectively.



The simulation of clustering C2CM and IMT-2020 run by inputting excel and saving the CVIs and other variables to a separate excel file, where the first sheet includes the y_train, y_initial, and y_pred and the second sheet contains the CVIs.

Specifically, the execution of the code started by asking to input a dataset either “cost” or “imt2020”. After inputting the chosen dataset, it printed the scenarios that were included on that Channel Model. Subsequently, it asked what scenario to cluster from the C2CM or IMT-2020.

The system asked again what Channel Model to cluster to access and remove initial cluster. Afterwards, it asked to create an excel file or proceed clustering if excel file has been created.

Listing 6.5 Under main run file script:

```
1 xlsxname = 'Sample'
2 xlsxname = menu(xlsxname)
3 xlsxcreate = 'Sample2'
4 xlsxcreate = excel()
5 option1 = int(input('Select your choice: '))
6 while option1 !=2:
7     if option1 == 1:
8         print(xlsxname)
9         df1 = pd.DataFrame()
10        df1.to_excel(str(xlsxname)+'.xlsx')
11        break
12    else:
13        print('Invalid Option.')
14        print()
15        excel()
16        option1 = int(input('Enter your option: '))
17    print()
```

Listing 6.6 Under main run file script:

```
1 def excel():
2     data = str(input('Select excel datasets cost or imt2020: '))
3     if data == 'cost':
4         file_path = r'C:\Users\Mark\Macapagal\Desktop\MasterCode2\
5                     save\cost\initial_cluster.npy'
6     elif data == 'imt2020':
7         file_path = r'C:\Users\Mark\Macapagal\Desktop\MasterCode2\
8                     save\imt2020\initial_cluster.npy'
9     if os.path.exists(file_path):
10        os.remove(file_path)
11        print('Initial Cluster has been Deleted')
12    else:
13        print('No Initial Cluster Found.')
14    print()
15    print('1 - Create a New Excel File')
16    print('2 - Proceed')
17    print()
```



To continue, the system proceeded clustering after choosing an excel file and sheet. There was a sound indicator once clustering is finished. The CVIs and clusters was saved on the separate excel file created including NMI, ARI, ACC, JI, RE, and Time elapsed and analyzed in Sec 6.2, Sec 6.3, Sec 6.4.

Another modification of the code includes, number of epochs, which were adapted from the pre-coded k -DAE algorithm itself. The (1) pre-train epoch produced the initial result of the algorithm, (2) autoencoder epoch assigned the labels on each data point to minimize the error upon reconstruction, (3) k -DAE epoch reconstructed the data belonging on a particular cluster. However, the number of neurons and layers were adapted from [Song et al., 2013], which used Autoencoder the k -means Algorithm.

Listing 6.7 Under k -DAE script:

```
1 class Kdae:  
2     def __init__(self, number_cluster, dataset_name='temp', k_dae_epoch  
3                  =5, ae_initial_dim=(1024,256,10,256,1024),  
4                  initial_epoch=100, ae_dim=(1024,256,10,256,1024),  
5                  epoch_ae=20, batch_size=256, save_dir='save'):
```

Listing 6.8 The code contains the final settings:

```
1 k_dae_epoch=5, epoch_ae=20, initial_epoch=100;  
2 ae_dim=(1024,256,10,256,1024), ae_initial_dim=(1024, 256, 10,256,1024)
```

Upon experimentation of the following parameters the following are observed:

1. Increasing number of k -DAE epoch has no significant increase in clustering accuracy. However, increasing number of epochs increased the duration of the time consumed. Thus, the researchers decided to retain the k_dae epoch.
2. Changing the number of layers and neurons adapted from the architecture of [Song et al., 2013] significantly improved clustering accuracy.



6.1.3 Implementation of Activation Functions and Optimizers

This section mainly satisfied the first objective of this study. It is to modify activation functions using (1) ReLU and (2) Sigmoid to cluster MPCs. The code used three different activation functions, (1) encoder, (2) embedded layer, and (3) decoder. These parameters will be modified using the configurations in Table 5.2.

Listing 6.9 Under autoencoder script:

```
1      for j, i in enumerate(self.hidden_dim[:-1]):
2          x = Dense(i, kernel_initializer=init, name='encoder_%d' % j
3                     )(x)
4          if self.batch_normalize:
5              x = BatchNormalization()(x)
6          #x = keras.layers.ELU()(x)
7          x = Activation('sigmoid')(x)
8      embedding_layer = Dense(self.hidden_dim[-1], kernel_initializer
9                             =init, name='embedding_layer')(x)
10     x = embedding_layer
11     for j, i in enumerate(self.hidden_dim[1:-1]):
12         x = Dense(i, kernel_initializer=init, name='decode_%d' % j
13                    )(x)
14         if self.batch_normalize:
15             x = BatchNormalization()(x)
16         #x = keras.layers.ELU()(x)
17         x = Activation('sigmoid')(x)
18         x = Dense(self.data_dim, kernel_initializer=init, activation=
19                   'sigmoid', name='decoder_0')(x)
20     decode = x
21     model = Model(inputs=inputs, outputs=decode)
22     embedding_model = Model(inputs=inputs, outputs=embedding_layer)
23     #plot_model(model, to_file='autoencoder.png', show_shapes=True)
24     self.model = model
25     self.embedding_model = embedding_model
```

Upon implementation of Optimizers, the researchers observed a significant increase in accuracy of clustering using ADAM and SGD. Thus, the researchers used this variable to combine it with activation functions to observe its effect in accuracy of clustering.



Listing 6.10 Under autoencoder script:

```
1 for SGD
2 class AutoEncoder:
3     def __init__(self, data_dim, hidden_dim, batch_normalize=True,
4                  epoch=100,
5                  batch_size=256, loss='binary_crossentropy', optimizer='
6                  SGD',
7                  save_name='temp', verbose=1, save_model=False):
8
9 #for ADAM
10 class AutoEncoder:
11     def __init__(self, data_dim, hidden_dim, batch_normalize=True,
12                  epoch=100,
13                  batch_size=256, loss='binary_crossentropy', optimizer='
14                  ADAM',
15                  save_name='temp', verbose=1, save_model=False):
16
17 #for SGD
18 def fit(self, x_train, patience=10):
19     sgd = SGD() # Create SGD optimizer object
20     self.model.compile(loss=self.loss, optimizer=sgd)
21
22 #for ADAM
23 def fit(self, x_train, patience=10):
24     self.model.compile(loss=self.loss, optimizer=self.optimizer)
```

Listing 6.11 Under *k*-DAE script:

```
1 #for SGD
2 self.k_dae_model.compile(optimizer='SGD', loss=self.k_dae_loss)
3 #for ADAM
4 self.k_dae_model.compile(optimizer='ADAM', loss=self.k_dae_loss)
```

After implementation of activation functions and optimizers, the following were observed:

1. Sigmoid and ReLU brought significantly higher accuracy. Hence, ReLU was compared to Sigmoid to establish which activation function configuration was sufficient.
2. SGD and ADAM were utilized to compare their effect in clustering MPCs.

In summary, this subsection discussed the implementation of activation functions and optimizers in the *k*-DAE Python script, the addition of IMT-2020 Channel Model, saving of clustering results, and the settings used in the *k*-DAE algorithm specific in this study.



6.2 Clustering Accuracy of Modified k-DAE

To attain the second specific objective of this study, which is to achieve close or higher JI in clustering C2CM and IMT-2020, Python was used to enhance the results of the *k*-DAE by modifying the membership of clusters, and MATLAB software was utilized to evaluate the results of each scenario by creating box plots.

6.2.1 Box Plots of CVIs of Different Configurations in C2CM

The following Figures show the NMI, ARI, ACC, and JI of the different activation functions in C2CM.

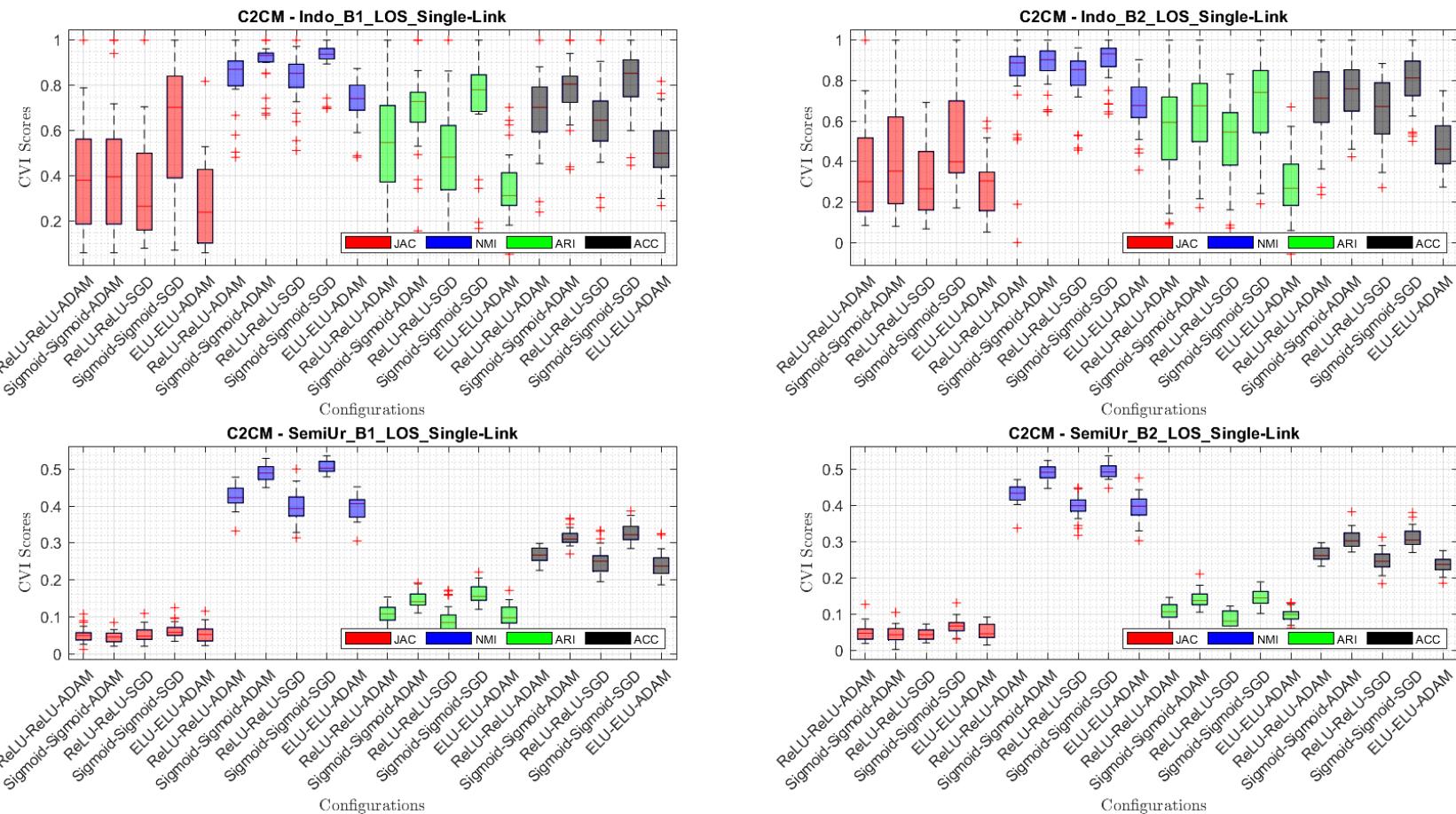


Fig. 6.1 Boxplots of C2CM's First Four Scenarios

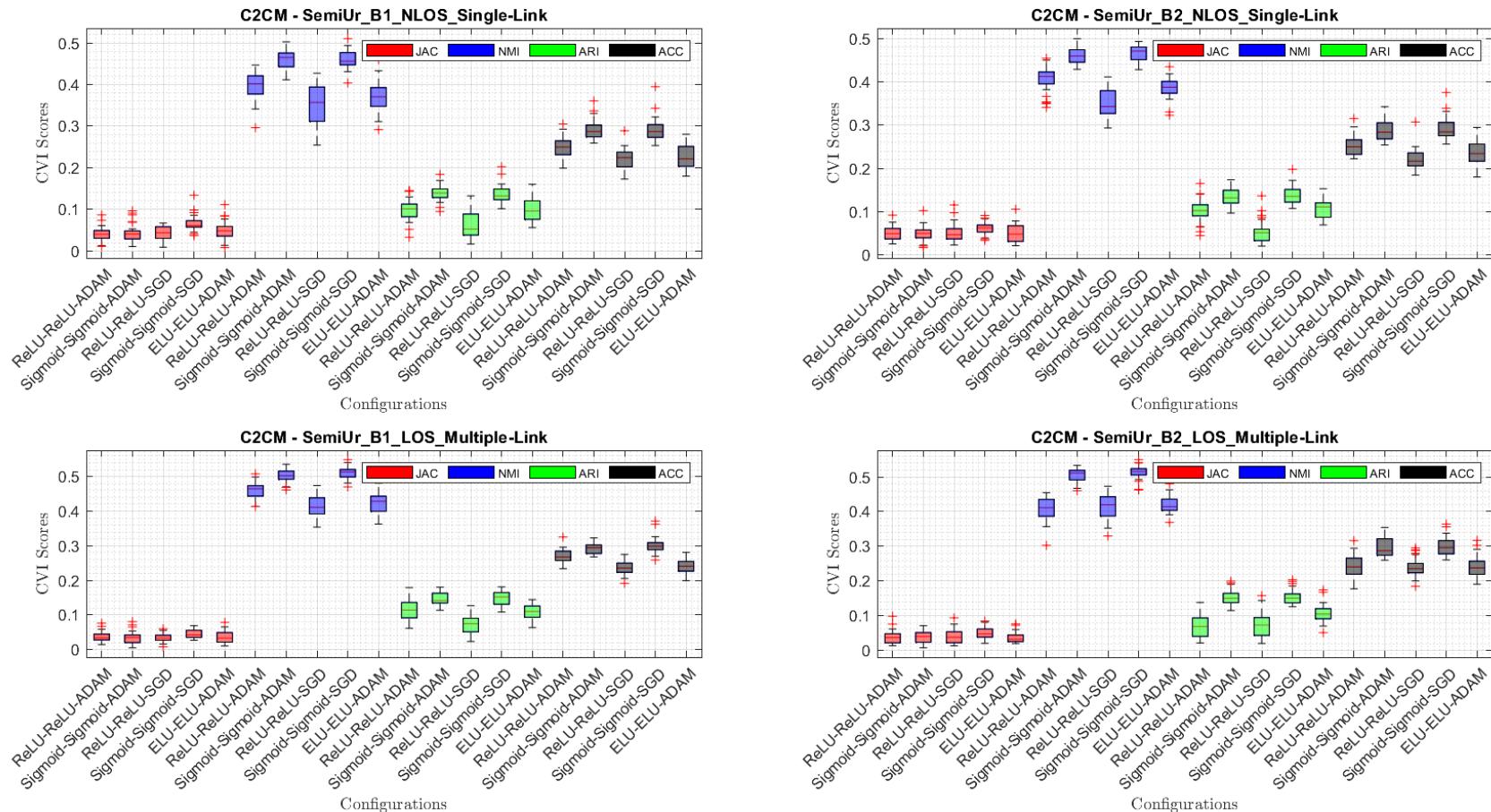


Fig. 6.2 Boxplots of C2CM's Last Four Scenarios

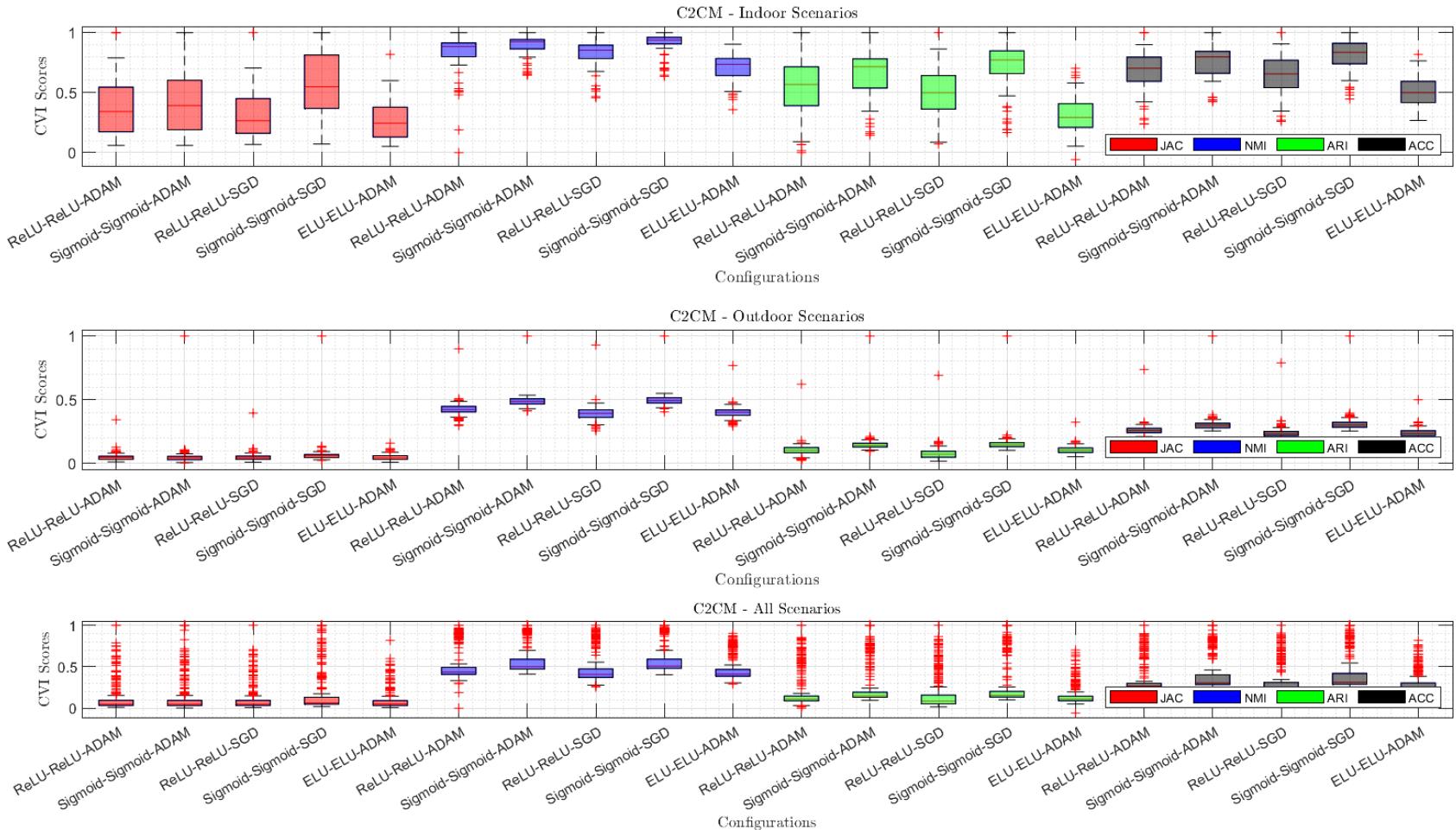


Fig. 6.3 Boxplots of C2CM's Indoor, Outdoor, and All Scenarios



In Fig 6.1 and Fig 6.2, the CVI results of different configurations were juxtaposed in every scenario of C2CM. The JI result of the different configurations is highlighted in the left part of the graph with red in color. The Sigmoid-Sigmoid-SGD configuration achieved the highest JI result among all the other configurations. The interquartile ranges of NMI and ACC are shorter or smaller among the CVIs. This shows that NMI and ACC have more consistent results. Additionally, Fig 6.3 provides the summary of all indoor, all outdoor scenarios and the overall scenarios. In summary, the Sigmoid-Sigmoid-SGD gives the highest JI result among the different configurations of k -DAE algorithm in clustering the C2CM datasets.

6.2.2 Box Plots of CVIs of Different Configurations in IMT-2020

The following Figures show the NMI, ARI, ACC, and JI of the different activation functions in IMT-2020.

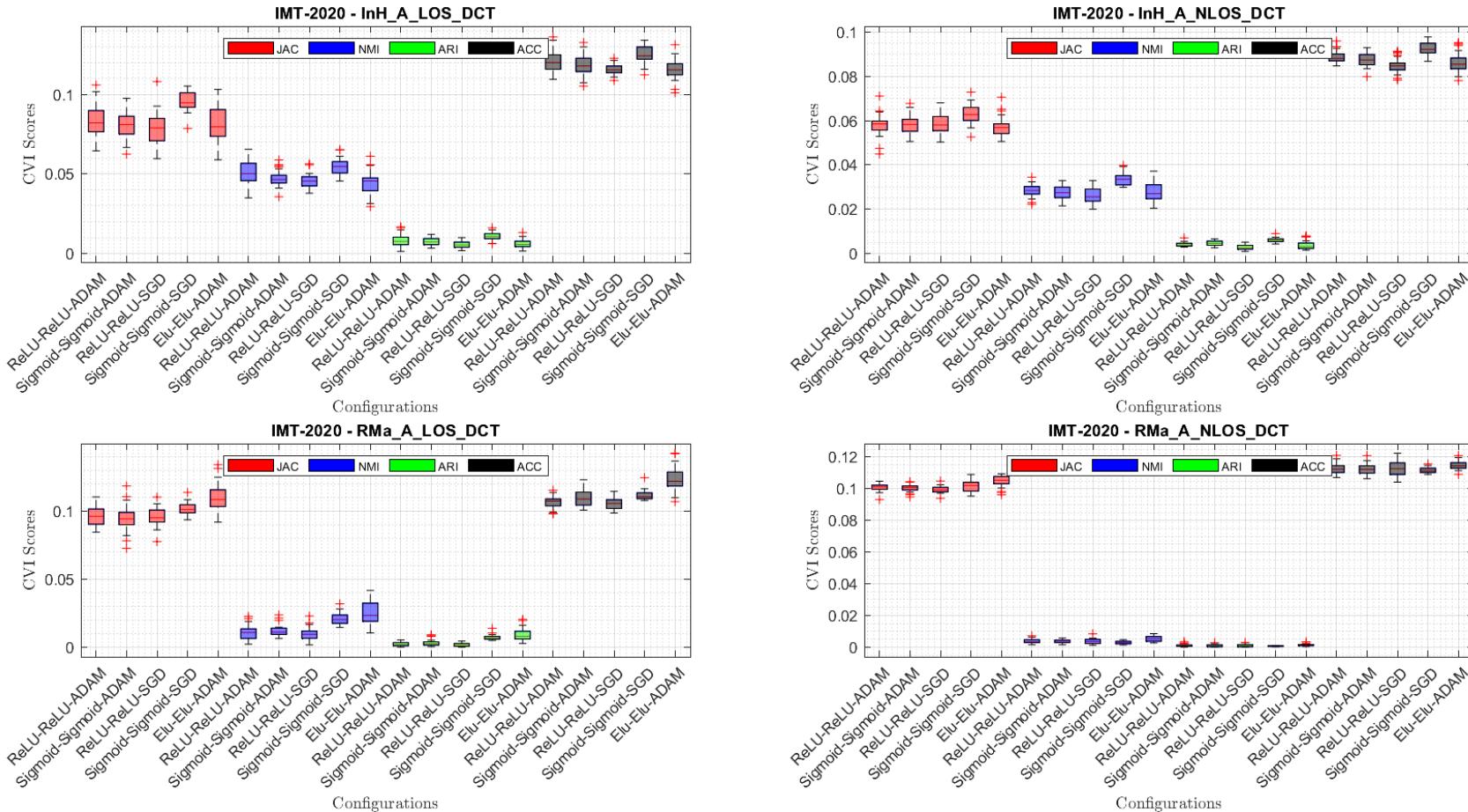


Fig. 6.4 Boxplots of IMT-2020's First Four Scenarios

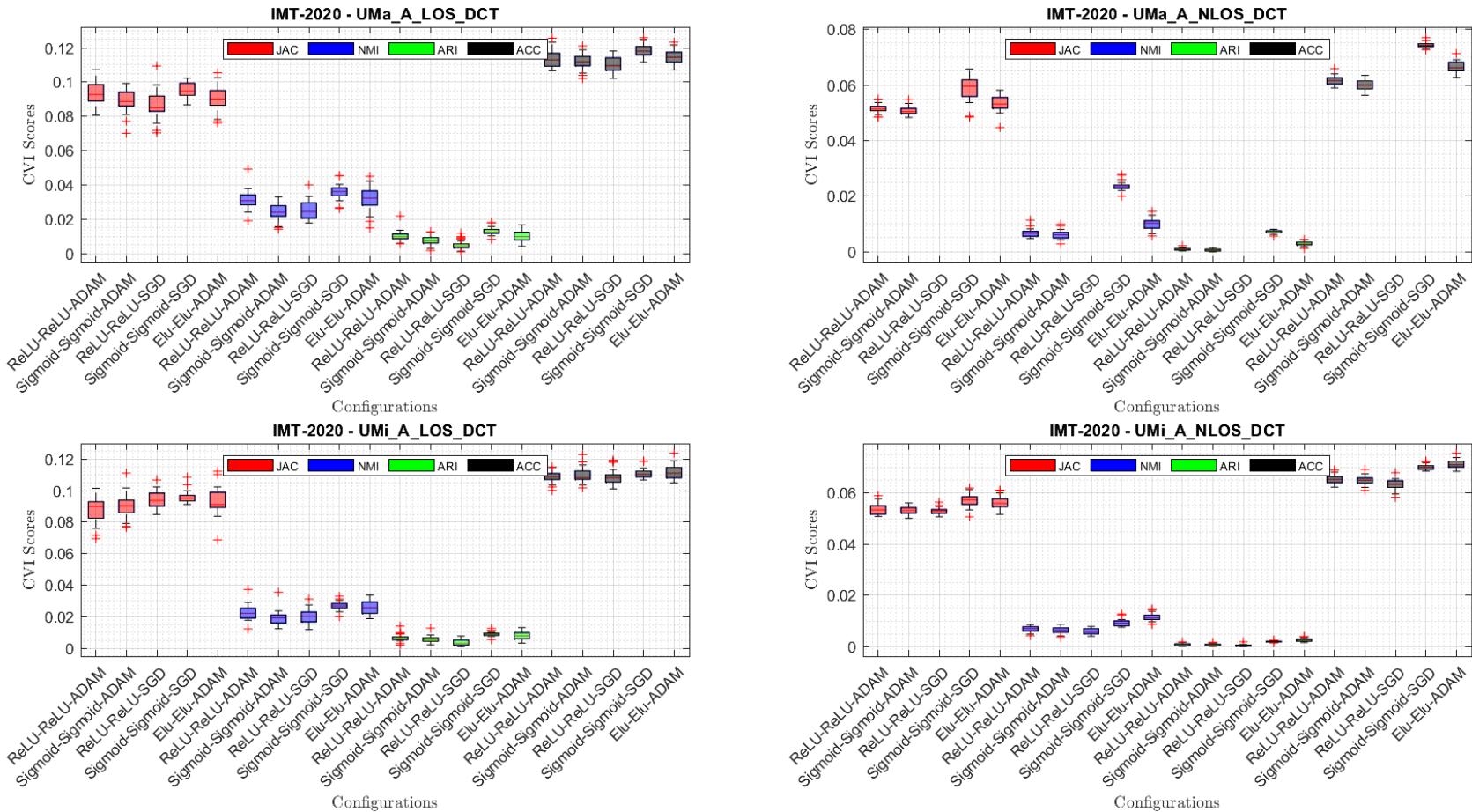


Fig. 6.5 Boxplots of IMT-2020's Last Four Scenarios

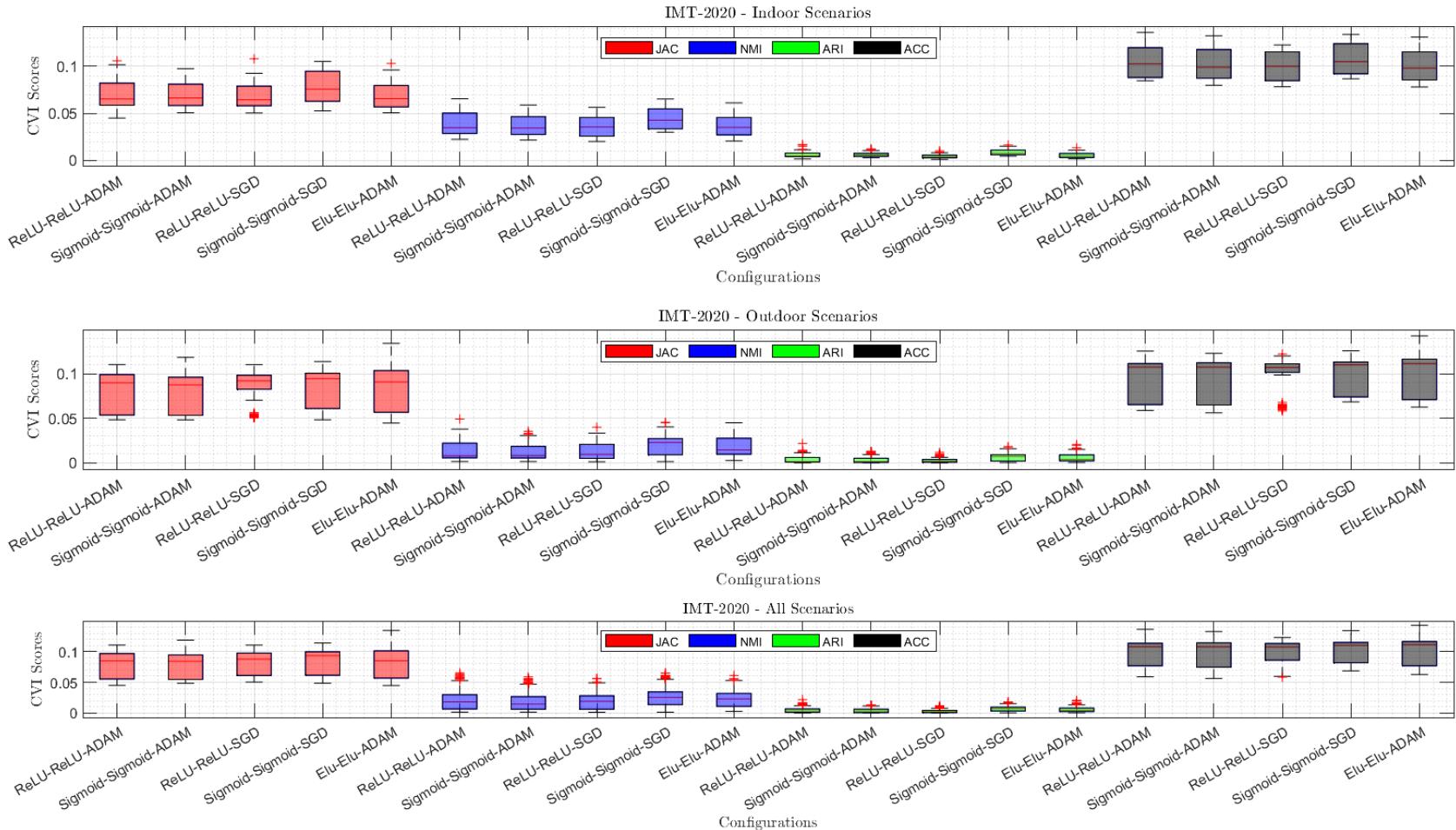


Fig. 6.6 Boxplots of IMT-2020's Indoor, Outdoor, and All Scenarios



In Fig 6.4 and Fig 6.5, each scenario of IMT-2020 is depicted with boxplots of the CVI results of different configurations. The JI results of the different configurations are highlighted in red on the left side of the figures. Among these configurations, Sigmoid-Sigmoid-SGD exhibits the highest JI results in InH-A LOS, InH A NLOS, UMa A LOS, UMa A NLOS, and UMi A NLOS scenarios. Conversely, Exponential Linear Unit (eLU)-eLU-ADAM achieved the highest JI index in other scenarios. Notably, the ReLU-ReLU-SGD configuration failed to process datasets for the UMa A NLOS scenario due to its significant data volume. Additionally, Fig 6.6 presents a summary of all indoor, outdoor, and overall scenarios. In summary, Sigmoid-Sigmoid-SGD configuration attained the highest JI index across all indoor scenarios of IMT-2020. However, for outdoor scenarios, the eLU-eLU-ADAM configuration prevailed with the highest JI index.

6.2.3 Means and Standard Deviations of CVIs in C2CM

The following Figures show the means and standard deviations of the different activation functions in C2CM.

TABLE 6.2 MEAN AND STANDARD DEVIATIONS OF NMI RESULTS IN C2CM

Channel Scenario		NMI									
		ReLU-ReLU-ADAM		Sigmoid-Sigmoid-ADAM		ReLU-ReLU-SGD		Sigmoid-Sigmoid-SGD		ELU-ELU-ADAM	
		Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Band 1	Indoor SL LOS	0.8357	0.1235	0.9003	0.0860	0.8244	0.1123	0.9147	0.0840	0.7341	0.0971
	Semi-Urban SL LOS	0.4257	0.0282	0.4897	0.0225	0.3983	0.0429	0.5063	0.0156	0.3952	0.0312
	Semi-Urban SL NLOS	0.3975	0.0332	0.4594	0.0222	0.3515	0.0490	0.4623	0.0225	0.3717	0.0384
	Semi-Urban ML LOS	0.4601	0.0233	0.5015	0.0178	0.4136	0.0314	0.5102	0.0185	0.4237	0.0277
Band 2	Indoor SL LOS	0.8005	0.2271	0.8816	0.1002	0.8117	0.1393	0.8965	0.1082	0.6776	0.1253
	Semi-Urban SL LOS	0.4314	0.0263	0.4905	0.0211	0.3972	0.0311	0.4934	0.0199	0.3960	0.0333
	Semi-Urban SL NLOS	0.4060	0.0289	0.4599	0.0182	0.3517	0.0314	0.4661	0.0172	0.3864	0.0240
	Semi-Urban ML LOS	0.4086	0.0340	0.5053	0.0194	0.4119	0.0344	0.5130	0.0194	0.4184	0.0228
Mean		0.5207	0.0656	0.5860	0.0384	0.4950	0.0590	0.5953	0.0382	0.4754	0.0500



TABLE 6.3 MEAN AND STANDARD DEVIATIONS OF ACC RESULTS IN C2CM

Channel Scenario		ACC									
		ReLU-ReLU-ADAM		Sigmoid-Sigmoid-ADAM		ReLU-ReLU-SGD		Sigmoid-Sigmoid-SGD		ELU-ELU-ADAM	
		Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Band 1	Indoor SL LOS	0.6649	0.1667	0.7760	0.1317	0.6462	0.1627	0.8206	0.1367	0.5213	0.1336
	Semi-Urban SL LOS	0.2670	0.0201	0.3152	0.0207	0.2523	0.0356	0.3283	0.0255	0.2423	0.0322
	Semi-Urban SL NLOS	0.2504	0.0248	0.2926	0.0229	0.2209	0.0246	0.2921	0.0278	0.2283	0.0277
	Semi-Urban ML LOS	0.2686	0.0203	0.2930	0.0157	0.2374	0.0201	0.3018	0.0230	0.2393	0.0191
Band 2	Indoor SL LOS	0.6755	0.1986	0.7555	0.1531	0.6422	0.1616	0.7965	0.1502	0.4901	0.1232
	Semi-Urban SL LOS	0.2645	0.0187	0.3070	0.0240	0.2474	0.0264	0.3120	0.0260	0.2364	0.0209
	Semi-Urban SL NLOS	0.2526	0.0221	0.2877	0.0226	0.2209	0.0230	0.2933	0.0268	0.2342	0.0283
	Semi-Urban ML LOS	0.2418	0.0302	0.2970	0.0255	0.2403	0.0271	0.3016	0.0266	0.2391	0.0291
Mean		0.3606	0.0627	0.4155	0.0520	0.3384	0.0601	0.4308	0.0553	0.3039	0.0518

TABLE 6.4 MEAN AND STANDARD DEVIATIONS OF ARI RESULTS IN C2CM

Channel Scenario		ARI									
		ReLU-ReLU-ADAM		Sigmoid-Sigmoid-ADAM		ReLU-ReLU-SGD		Sigmoid-Sigmoid-SGD		ELU-ELU-ADAM	
		Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Band 1	Indoor SL LOS	0.5405	0.2117	0.6776	0.1986	0.5033	0.2102	0.7331	0.2057	0.3447	0.1514
	Semi-Urban SL LOS	0.1057	0.0265	0.1470	0.0204	0.0900	0.0373	0.1612	0.0244	0.1015	0.0270
	Semi-Urban SL NLOS	0.0991	0.0255	0.1394	0.0183	0.0629	0.0317	0.1370	0.0210	0.0985	0.0284
	Semi-Urban ML LOS	0.1135	0.0261	0.1465	0.0178	0.0726	0.0272	0.1491	0.0191	0.1090	0.0228
Band 2	Indoor SL LOS	0.5348	0.2660	0.6444	0.2288	0.5090	0.2063	0.6966	0.2404	0.2892	0.1593
	Semi-Urban SL LOS	0.1085	0.0213	0.1398	0.0231	0.0843	0.0250	0.1458	0.0209	0.0976	0.0180
	Semi-Urban SL NLOS	0.1020	0.0258	0.1352	0.0213	0.0536	0.0251	0.1373	0.0210	0.1069	0.0213
	Semi-Urban ML LOS	0.0715	0.0339	0.1528	0.0226	0.0758	0.0363	0.1549	0.0209	0.1071	0.0260
Mean		0.2094	0.0796	0.2728	0.0689	0.1814	0.0749	0.2894	0.0717	0.1568	0.0568

Table 6.2, Table 6.3, and Table 6.4 show the accomplished results of all scenarios in all validation indices, thus, comparing the five configurations of k -DAE in clustering C2CM. Among all the configurations, the Sigmoid-Sigmoid-SGD attained the highest mean value in all the scenarios consistently. Proving that Sigmoid-Sigmoid-SGD showed favorable results in clustering C2CM dataset followed by ReLU-ReLU-SGD. Additionally, Sigmoid-Sigmoid-SGD has generally lower standard deviation values compared to ReLU-ReLU-SGD, indicating more stable performance. Therefore, based on the provided data, the Sigmoid-Sigmoid-SGD configuration appeared to have the highest CVI scores.



6.2.4 Means and Standard Deviations of CVIs in IMT-2020

The following Tables show the means and SDs of the different activation functions in IMT-2020.

TABLE 6.5 MEAN AND STANDARD DEVIATIONS OF NMI RESULTS IN IMT-2020

Channel Scenario	NMI										
	ReLU-ReLU-ADAM		Sigmoid-Sigmoid-ADAM		ReLU-ReLU-SGD		Sigmoid-Sigmoid-SGD		ELU-ELU-ADAM		
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD	
LOS	Indoor Hotspot	0.0510	0.0071	0.0471	0.0048	0.0453	0.0045	0.0546	0.0051	0.0446	0.0072
	Rural Macro	0.0106	0.0050	0.0119	0.0040	0.0096	0.0044	0.0208	0.0037	0.0280	0.0046
	Urban Macro	0.0312	0.0051	0.0244	0.0050	0.0253	0.0053	0.0359	0.0042	0.0252	0.0077
	Urban Micro	0.0224	0.0044	0.0189	0.0044	0.0201	0.0043	0.0270	0.0025	0.0052	0.0016
NLOS	Indoor Hotspot	0.0283	0.0026	0.0274	0.0028	0.0262	0.0033	0.0335	0.0027	0.0318	0.0070
	Rural Macro	0.0036	0.0013	0.0035	0.0010	0.0035	0.0015	0.0027	0.0009	0.0100	0.0019
	Urban Macro	0.0067	0.0013	0.0061	0.0015	–	–	0.0236	0.0015	0.0255	0.0043
	Urban Micro	0.0068	0.0010	0.0064	0.0012	0.0059	0.0010	0.0092	0.0013	0.0114	0.0014
Mean		0.0201	0.0035	0.0182	0.0031	0.0194	0.0035	0.0259	0.0027	0.0227	0.0044

TABLE 6.6 MEAN AND STANDARD DEVIATIONS ACC RESULTS IN IMT-2020

Channel Scenario	ACC										
	ReLU-ReLU-ADAM		Sigmoid-Sigmoid-ADAM		ReLU-ReLU-SGD		Sigmoid-Sigmoid-SGD		ELU-ELU-ADAM		
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD	
LOS	Indoor Hotspot	0.1205	0.0062	0.1186	0.0062	0.1157	0.0030	0.1251	0.0055	0.1155	0.0061
	Rural Macro	0.1068	0.0042	0.1094	0.0056	0.1055	0.0040	0.1114	0.0034	0.0860	0.0043
	Urban Macro	0.1134	0.0046	0.1120	0.0041	0.1101	0.0043	0.1181	0.0036	0.1234	0.0085
	Urban Micro	0.1088	0.0034	0.1097	0.0045	0.1083	0.0044	0.1109	0.0028	0.1147	0.0025
NLOS	Indoor Hotspot	0.0889	0.0025	0.0874	0.0030	0.0848	0.0032	0.0924	0.0029	0.1144	0.0040
	Rural Macro	0.1129	0.0032	0.1125	0.0034	0.1125	0.0046	0.1117	0.0017	0.0665	0.0018
	Urban Macro	0.0615	0.0015	0.0599	0.0018	–	–	0.0743	0.0009	0.1117	0.0045
	Urban Micro	0.0653	0.0017	0.0650	0.0016	0.0632	0.0019	0.0701	0.0010	0.0712	0.0015
Mean		0.0973	0.0034	0.0968	0.0038	0.1000	0.0037	0.1018	0.0027	0.1004	0.0041



TABLE 6.7 MEAN AND STANDARD DEVIATIONS ARI RESULTS IN IMT-2020

Channel Scenario		ARI									
		ReLU-ReLU-ADAM		Sigmoid-Sigmoid-ADAM		ReLU-ReLU-SGD		Sigmoid-Sigmoid-SGD		ELU-ELU-ADAM	
		Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
LOS	Indoor Hotspot	0.0080	0.0036	0.0074	0.0022	0.0054	0.0019	0.0109	0.0023	0.0060	0.0025
	Rural Macro	0.0020	0.0014	0.0030	0.0022	0.0017	0.0012	0.0070	0.0017	0.0037	0.0018
	Urban Macro	0.0101	0.0028	0.0077	0.0027	0.0049	0.0025	0.0129	0.0021	0.0094	0.0045
	Urban Micro	0.0066	0.0022	0.0056	0.0020	0.0036	0.0018	0.0089	0.0012	0.0013	0.0006
NLOS	Indoor Hotspot	0.0041	0.0009	0.0046	0.0012	0.0028	0.0011	0.0060	0.0009	0.0099	0.0031
	Rural Macro	0.0009	0.0007	0.0008	0.0006	0.0009	0.0008	0.0005	0.0002	0.0029	0.0007
	Urban Macro	0.0009	0.0003	0.0006	0.0004	–	–	0.0071	0.0005	0.0079	0.0025
	Urban Micro	0.0008	0.0004	0.0006	0.0003	0.0004	0.0003	0.0020	0.0002	0.0025	0.0006
Mean		0.0042	0.0015	0.0038	0.0015	0.0028	0.0014	0.0069	0.0011	0.0054	0.0020

Tables 6.5, 6.6, and 6.7 present the achieved results for NMI, ACC, and ARI, comparing the performance of five configurations of k -DAE in clustering IMT-2020 dataset. Across all scenarios, Sigmoid-Sigmoid-SGD consistently obtained the highest mean value, indicating its superior performance in clustering IMT-2020 data. Based on the table, Sigmoid-Sigmoid-SGD recorded a mean value of 0.0259, 0.1018, and 0.0069 in NMI, ACC, and ARI, respectively. Moreover, Sigmoid-Sigmoid-SGD exhibited the lowest SD values overall, suggesting a more stable performance compared to other configurations recording a standard deviation value of 0.0027 in NMI and ACC, and 0.0011 in ARI. Therefore, based on the provided data, the Sigmoid-Sigmoid-SGD configuration appeared to yield the highest scores in NMI, ACC, and ARI.

6.3 Statistical Comparison of different Activation functions and Optimizers

This section provides statistical analyses on the comparison of different configuration of activation functions and optimizers to find the suitable configuration for clustering MPCs. This section specifically compares the JIs of different configu-



rations to achieve closer or higher clustering membership accuracy in clustering C2CM and IMT-2020. Thus, this section provided resolution for the second specific objective of the study.

6.3.1 Means and Standard Deviations of JIs in C2CM

TABLE 6.8 MEAN AND STANDARD DEVIATIONS OF JI RESULTS IN C2CM

Channel Scenario		JI									
		ReLU-ReLU-ADAM		Sigmoid-Sigmoid-ADAM		ReLU-ReLU-SGD		Sigmoid-Sigmoid-SGD		ELU-ELU-ADAM	
		Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Band 1	Indoor SL LOS	0.3876	0.2306	0.4310	0.2583	0.3395	0.2239	0.6296	0.2752	0.2665	0.1841
	Semi-Urban SL LOS	0.0505	0.0199	0.0454	0.0151	0.0520	0.0193	0.0629	0.0192	0.0530	0.0217
	Semi-Urban SL NLOS	0.0410	0.0156	0.0435	0.0211	0.0429	0.0150	0.0646	0.0188	0.0487	0.0227
	Semi-Urban ML LOS	0.0374	0.0140	0.0354	0.0175	0.0349	0.0118	0.0465	0.0128	0.0364	0.0171
Band 2	Indoor SL LOS	0.3558	0.2330	0.4450	0.2927	0.2985	0.1778	0.5301	0.2788	0.2828	0.1366
	Semi-Urban SL LOS	0.0514	0.0229	0.0462	0.0233	0.0450	0.0146	0.0679	0.0209	0.0516	0.0219
	Semi-Urban SL NLOS	0.0502	0.0162	0.0490	0.0178	0.0527	0.0213	0.0615	0.0143	0.0500	0.0204
	Semi-Urban ML LOS	0.0399	0.0189	0.0388	0.0167	0.0409	0.0197	0.0509	0.0157	0.0375	0.0147
Mean		0.1267	0.0714	0.1418	0.0828	0.1133	0.0629	0.1893	0.0820	0.1033	0.0549

Table 6.8 shows the summary of JI mean and SD of all scenarios in C2CM. It can be observed that the Sigmoid-Sigmoid-SGD configuration JI stood out due to the highest total mean value of all scenarios with the score 0.1893. Also, among the five configurations Sigmoid-Sigmoid-SGD has the highest mean value JI in all the scenarios belonging to B1-Indoor Single Link LOS that garnered a mean JI score of 0.6296 and SD of 0.2752. In Semi-Urban Outdoor Scenarios, the highest JI mean score came from Sigmoid-Sigmoid-SGD belonging to B2- Semi-Urban Single Link LOS having a mean of 0.0679 and SD of 0.0209. Furthermore, in Semi-Urban Multiple Link Sigmoid-Sigmoid-SGD has the highest mean score of 0.0509 with SD of 0.0157. However, eLU-eLU-ADAM has the lowest SD among all the configurations exhibiting a total mean SD score of 0.054942.

Overall, the Sigmoid-Sigmoid-SGD has the highest mean value for all indoor and outdoor scenarios. It can be observed that it has favorable JI mean scores in



all scenarios making it the most sufficient activation functions and optimizer out of all configurations in C2CM.

6.3.2 Means and Standard Deviations of JIs in IMT-2020

TABLE 6.9 MEAN AND STANDARD DEVIATIONS JI RESULTS IN IMT-2020

Channel Scenario		JI									
		ReLU-ReLU-ADAM		Sigmoid-Sigmoid-ADAM		ReLU-ReLU-SGD		Sigmoid-Sigmoid-SGD		ELU-ELU-ADAM	
		Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
LOS	Indoor Hotspot	0.0836	0.0103	0.0807	0.0082	0.0788	0.0103	0.0955	0.0061	0.0808	0.0098
	Rural Macro	0.0964	0.0070	0.0945	0.0093	0.0959	0.0066	0.1014	0.0045	0.0573	0.0044
	Urban Macro	0.0931	0.0067	0.0892	0.0061	0.0868	0.0081	0.0955	0.0042	0.1101	0.0096
	Urban Micro	0.0883	0.0075	0.0897	0.0070	0.0943	0.0054	0.0959	0.0036	0.1048	0.0034
NLOS	Indoor Hotspot	0.0579	0.0048	0.0581	0.0042	0.0583	0.0044	0.0629	0.0043	0.0902	0.0070
	Rural Macro	0.1008	0.0023	0.1005	0.0022	0.0995	0.0021	0.1016	0.0038	0.0534	0.0027
	Urban Macro	0.0513	0.0015	0.0506	0.0014	—	—	0.0586	0.0040	0.0934	0.0082
	Urban Micro	0.0536	0.0020	0.0532	0.0014	0.0529	0.0012	0.0572	0.0025	0.0562	0.0023
	Mean	0.0781	0.0053	0.0771	0.0050	0.0828	0.0054	0.0836	0.0041	0.0808	0.0059

Table 6.9 shows the summary of JI mean and SD of all scenarios in IMT-2020. It can be observed that the Sigmoid-Sigmoid-SGD configuration JI stood out due to the highest total mean value of all scenarios with the score 0.0836. Also, in Sigmoid-Sigmoid-SGD, the highest mean value JI in all the scenarios belonging to NLOS-RMa that garnered a mean JI score of 0.1016 and SD of 0.0038. In Indoor Scenarios, the highest JI mean score came from Sigmoid-Sigmoid-SGD belonging to LOS- InH having a mean of 0.0956 and SD of 0.0061. Furthermore, in UMa Sigmoid-Sigmoid-SGD has the highest mean score of 0.0955 with SD of 0.0042. Moreover, in UMa, Sigmoid-Sigmoid-SGD has favorable JI mean score belonging to LOS- UMa with the mean score 0.0959 and SD 0. 0036. Consequently, Sigmoid-Sigmoid-SGD exhibits the lowest SD score among all the configurations garnering a total mean SD score of 0.0041.

Overall, the Sigmoid-Sigmoid-SGD has the highest mean value for all indoor and outdoor scenarios. It can be observed that it has promising JI mean scores in



all scenarios making it the most sufficient activation functions and optimizer out of all configurations in IMT-2020 Channel Model. Thus, it can be conjectured that it is the highest score among all configurations.

6.3.3 ANOVA of JIs in C2CM

This section shows the ANOVA of each scenario in C2CM Channel Model. Below are the ANOVA plots of different scenarios in C2CM Channel Model.

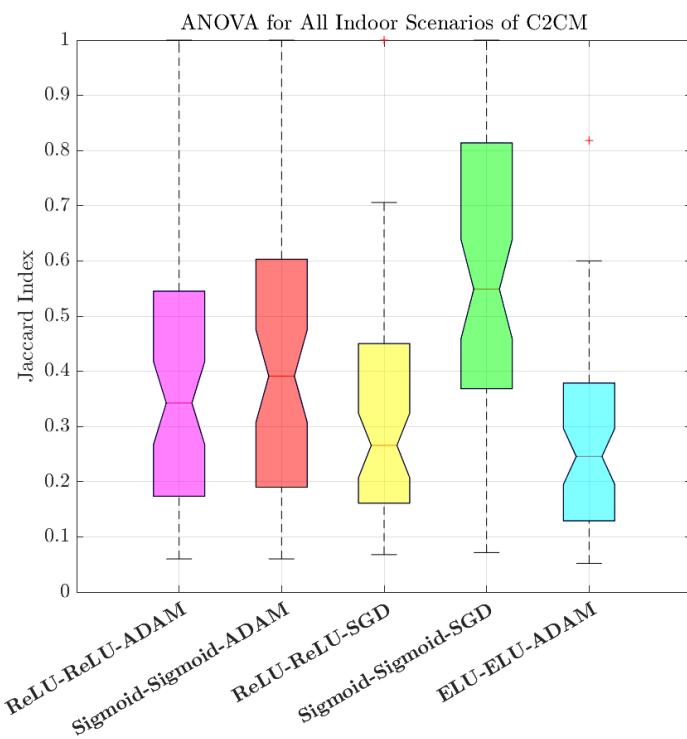


Fig. 6.7 C2CM ANOVA of All Indoor Scenarios

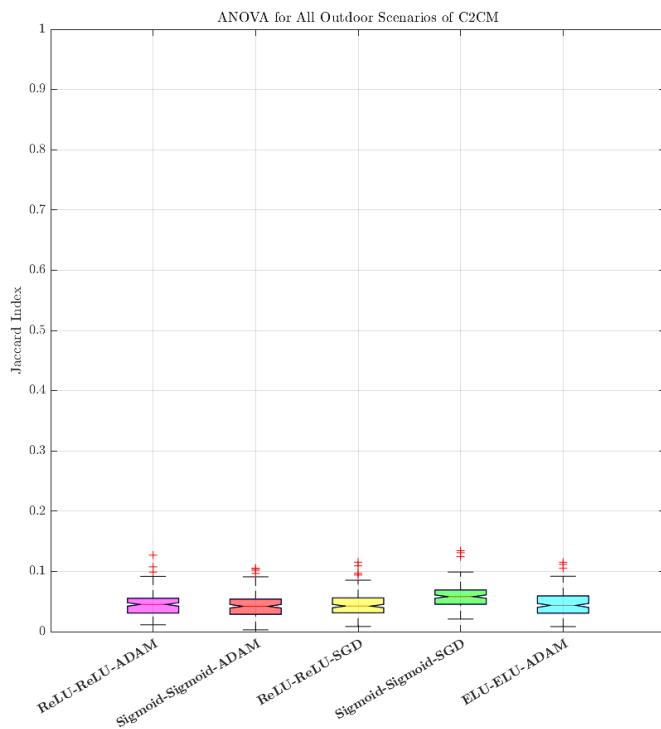


Fig. 6.8 C2CM ANOVA of All Outdoor Scenarios

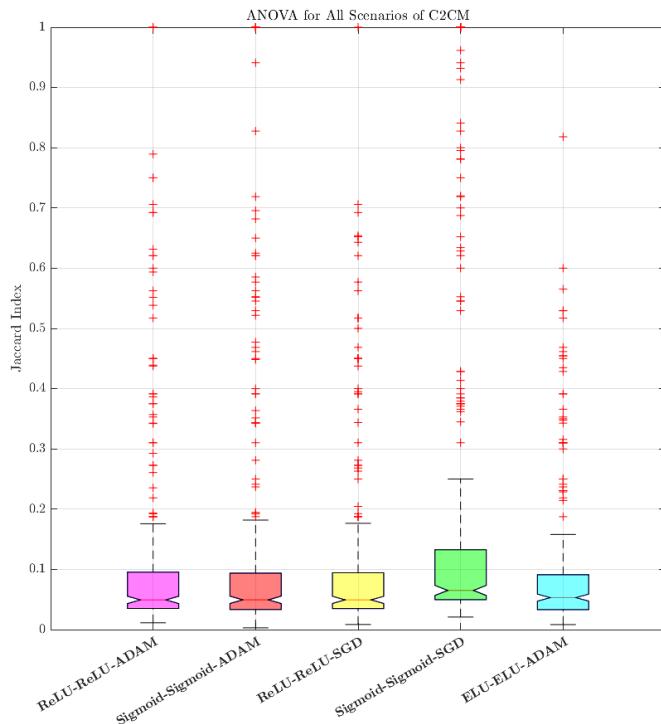


Fig. 6.9 C2CM ANOVA of All Scenarios



The following tables shows the summarize ANOVA for indoor, outdoor, and all scenarios.

TABLE 6.10 TABULAR VALUES OF C2CM ANOVA IN ALL INDOOR SCENARIOS

Configuration	N	Mean	Std. deviation	Std. error
RELU-RELU-ADAM	60	0.3717	0.234	0.0303
RELU-RELU-SGD	60	0.3190	0.205	0.0265
SIGMOID-SIGMOID-ADAM	60	0.438	0.279	0.0360
SIGMOID-SIGMOID-SGD	60	0.5798	0.284	0.0366
ELU-ELU-ADAM	60	0.2746	0.164	0.0211

TABLE 6.11 TABULAR VALUES OF C2CM ANOVA IN ALL OUTDOOR SCENARIOS

Configuration	N	Mean	Std. deviation	Std. error
RELU-RELU-ADAM	180	0.0451	0.0191	0.00143
RELU-RELU-SGD	180	0.0448	0.0185	0.00138
SIGMOID-SIGMOID-ADAM	180	0.0431	0.0194	0.00145
SIGMOID-SIGMOID-SGD	180	0.0591	0.0189	0.00141
ELU-ELU-ADAM	180	0.0462	0.0211	0.00158

TABLE 6.12 TABULAR VALUES OF C2CM ANOVA IN ALL SCENARIOS

Configuration	N	Mean	Std. deviation	Std. error
RELU-RELU-ADAM	240	0.1267	0.184	0.01189
RELU-RELU-SGD	240	0.1133	0.157	0.01016
SIGMOID-SIGMOID-ADAM	240	0.1418	0.221	0.01426
SIGMOID-SIGMOID-SGD	240	0.1893	0.267	0.01723
ELU-ELU-ADAM	240	0.1033	0.130	0.00836



Based on the data presented on the overall indoor ANOVA, the Sigmoid-Sigmoid-SGD exhibited the highest JI median value of 0.549 with 25th percentile of 0.368 and 75th percentile of 0.814 as seen in Fig 6.7. In Table 6.10 Sigmoid-Sigmoid-SGD exhibited the highest JI mean and SD values of 0.5798 and 0.284, respectively. This unveiled that the Sigmoid-Sigmoid-SGD configuration was more effective in clustering than the four other configurations but more dispersed data due to high SD.

In the overall outdoor ANOVA , the Sigmoid-Sigmoid-SGD also leads, garnering the highest JI median value of 0.058 with 25th percentile of 0.045 and 75th percentile of 0.069 as seen in Fig 6.8. Also, in Table 6.11, Sigmoid-Sigmoid-SGD took the highest JI mean of 0.0591 and SD value of 0.0189. This showed that the Sigmoid-Sigmoid-SGD configuration was more effective in clustering than the four other configurations in outdoor scenarios in C2CM.

In terms of overall scenarios in ANOVA, the Sigmoid-Sigmoid-SGD has effective clustering performance compared to the four other configurations. Exhibiting a median value of 0.065, 25th percentile equal to 0.05, and 75th percentile equal to 0.133 based on Fig 6.9 Furthermore, in Table 6.12, the Sigmoid-Sigmoid-SGD configuration has the highest JI mean value of 0.1893. However, the SD is 0.267 making the variability more dispersed.

6.3.4 ANOVA of JIs in IMT-2020

This section shows the ANOVA of each scenario in IMT-2020 Channel Model. Below are the ANOVA plots of different scenarios in the IMT-2020 Channel Model.

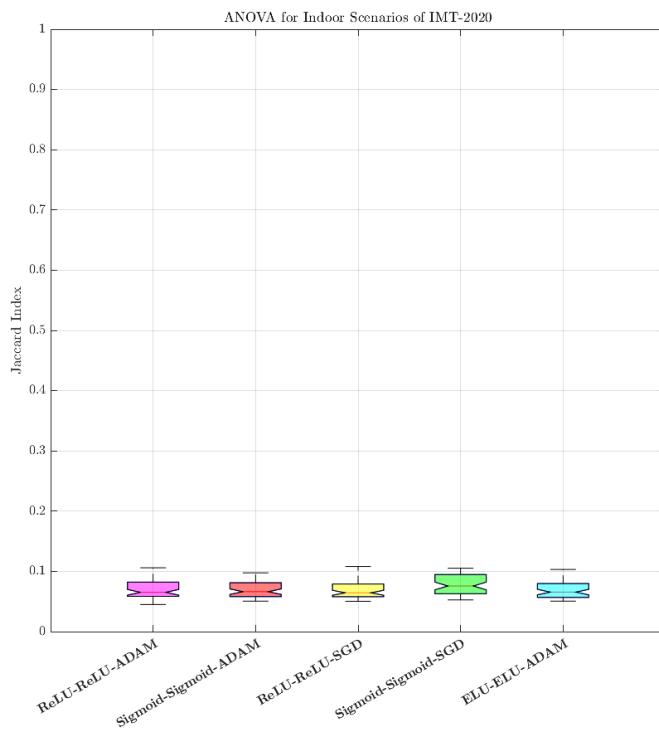


Fig. 6.10 IMT-2020 ANOVA of All Indoor Scenarios

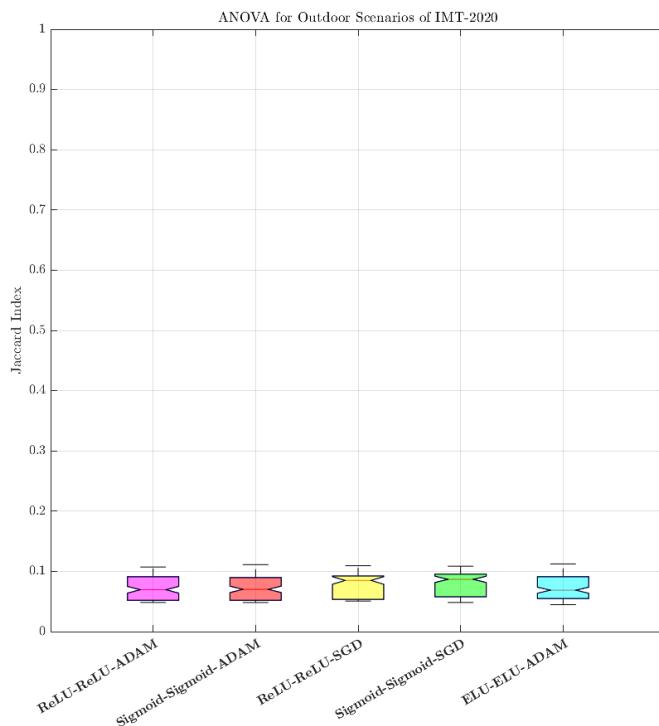


Fig. 6.11 IMT-2020 ANOVA of All Outdoor Scenarios

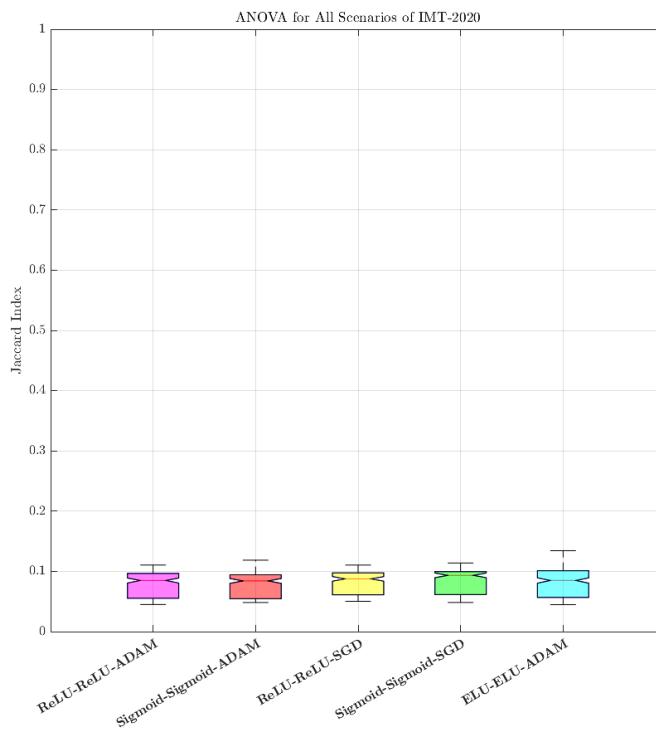


Fig. 6.12 IMT-2020 ANOVA of All Scenarios

TABLE 6.13 TABULAR VALUES OF IMT-2020 ANOVA IN ALL INDOOR SCENARIOS

Configuration	N	Mean	Std. deviation	Std. error
RELU-RELU-ADAM	60	0.0708	0.0153	0.00198
RELU-RELU-SGD	60	0.0686	0.0131	0.00169
SIGMOID-SIGMOID-ADAM	60	0.0695	0.0132	0.00170
SIGMOID-SIGMOID-SGD	60	0.0793	0.0173	0.00223
ELU-ELU-ADAM	60	0.0691	0.0141	0.00183

TABLE 6.14 TABULAR VALUES OF IMT-2020 ANOVA IN ALL OUTDOOR SCENARIOS

Configuration	N	Mean	Std. deviation	Std. error
RELU-RELU-ADAM	180	0.0806	0.0210	0.00156

Continued on next page

*Continued from previous page*

Configuration	N	Mean	Std. deviation	Std. error
RELU-RELU-SGD	150	0.0859	0.0179	0.00146
SIGMOID-SIGMOID-ADAM	180	0.0797	0.0208	0.00155
SIGMOID-SIGMOID-SGD	180	0.0851	0.0198	0.00147
ELU-ELU-ADAM	180	0.0847	0.0231	0.00172

TABLE 6.15 TABULAR VALUES OF IMT-2020 ANOVA IN ALL SCENARIOS

Configuration	N	Mean	Std. deviation	Std. error
RELU-RELU-ADAM	240	0.0782	0.0201	0.0013
RELU-RELU-SGD	210	0.0810	0.0184	0.00127
SIGMOID-SIGMOID-ADAM	240	0.0771	0.0196	0.00127
SIGMOID-SIGMOID-SGD	240	0.0836	0.0193	0.00125
ELU-ELU-ADAM	240	0.0808	0.0223	0.00144

Based on the data presented on the overall indoor ANOVA, the eLU-eLU-ADAM exhibited the highest JI median value of 0.0655 with 25th percentile of 0.0569 and 75th percentile of 0.08 as seen in Fig 6.10. In Table 6.13 Sigmoid-Sigmoid-SGD exhibited the highest JI mean and SD values of 0.0793 and 0.0173, respectively. This revealed that the Sigmoid-Sigmoid-SGD configuration was more effective in clustering in indoor scenarios of IMT-2020 than the four other configurations.

In the overall outdoor ANOVA, the Sigmoid-Sigmoid-SGD leads, garnering the highest JI median value of 0.0946 with 25th percentile of 0.0611 and 75th percentile of 0.1007 as seen in Fig 6.11. Furthermore, in Table 6.14, it can be observed that the number of samples in ReLU-ReLU-SGD was only 150 due to inapplicability of this configuration on the UMa A NLOS which has 30 excel sheets. In this case, Sigmoid-Sigmoid-SGD took the highest JI mean of 0.0851 and SD value of 0.0198. This showed that the Sigmoid-Sigmoid-SGD configuration was



more effective in clustering than the four other configurations in outdoor scenarios in IMT-2020.

In terms of overall scenarios in ANOVA, the Sigmoid-Sigmoid-SGD was the most effective configuration in clustering compared to the four other configurations. Exhibiting a median value of 0.0934, 25th percentile equal to 0.0614, and 75th percentile equal to 0.0995 based on Fig 6.12. Furthermore, in Table 6.15, the Sigmoid-Sigmoid-SGD configuration has the highest JI mean value of 0.0836 and the standard deviation was 0.0193 making the variability more compacted with each other.

Similarly, the comparison of the variability of C2CM and IMT-2020 showed their variability of the data which influenced the clustering scores. It is evident that IMT-2020 showed consistency in values while C2CM has larger variability. Overall, Sigmoid-Sigmoid-SGD still leads in IMT-2020 considering all the scenarios on the dataset.

6.3.5 Histogram on distribution of JI in C2CM and IMT-2020

Fig 6.13 showed the distribution of accumulated JIs in C2CM using different configurations. It shows how the mean values were affected by the distribution of JIs in terms of all scenarios. It is evident that the graph is rightly skewed meaning most data were positioned on the left part of the graph. In the graph, it can be observed that Sigmoid-Sigmoid-SGD and Sigmoid-Sigmoid-ADAM exhibited outlier values ranging from 0.9 to 1. From the past observations, Indoor scenarios have the highest mean in JI accumulation. Therefore, these values mostly came from indoor scenarios. Also, ReLU-ReLU-SGD and eLU-eLU-ADAM accumulated JIs values that reside on the left most side of the histogram making them the lowest total mean of JIs.

Moreover, Fig 6.14 shows the distribution of accumulated JIs in IMT-2020



using different configurations. It showed how the mean values are affected by the distribution of JIs in terms of all scenarios. It is evident that the values of the data are mostly positioned in the 0.05 to 0.1 range. In the graph, it can be observed that eLU-eLU-ADAM exhibited values ranging from 0.1 to 0.15. From the past observations, eLU-eLU-ADAM has the highest mean in indoor scenarios. Therefore, these values mostly came from indoor scenarios. Moreover, consistent values ranging 0.05-0.1 shown are garnered by Sigmoid-Sigmoid-SGD. Also, it is evident that the data exist on a compressed range making the variability of the data consistent with its values.

In summary, the data from C2CM has a larger variability than in IMT-2020. Also, in C2CM the data is widely distributed and affected by indoor scenarios. However, in IMT-2020 the data is compact and low variability making the values consistent. Also, the data that made the highest JI scores observed belong to the Indoor Scenarios.

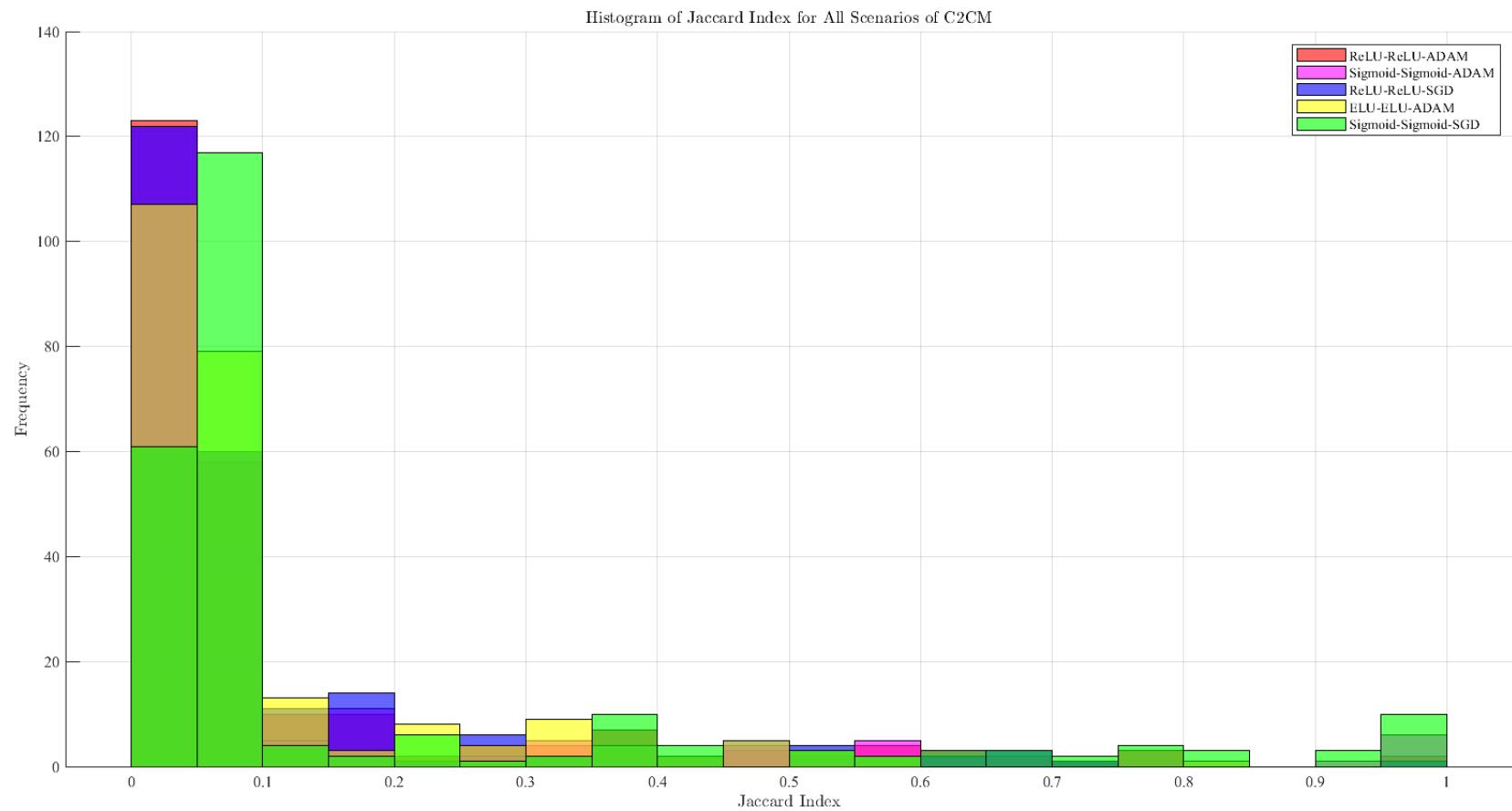


Fig. 6.13 C2CM- Histogram of Jaccard Indices of All Scenarios

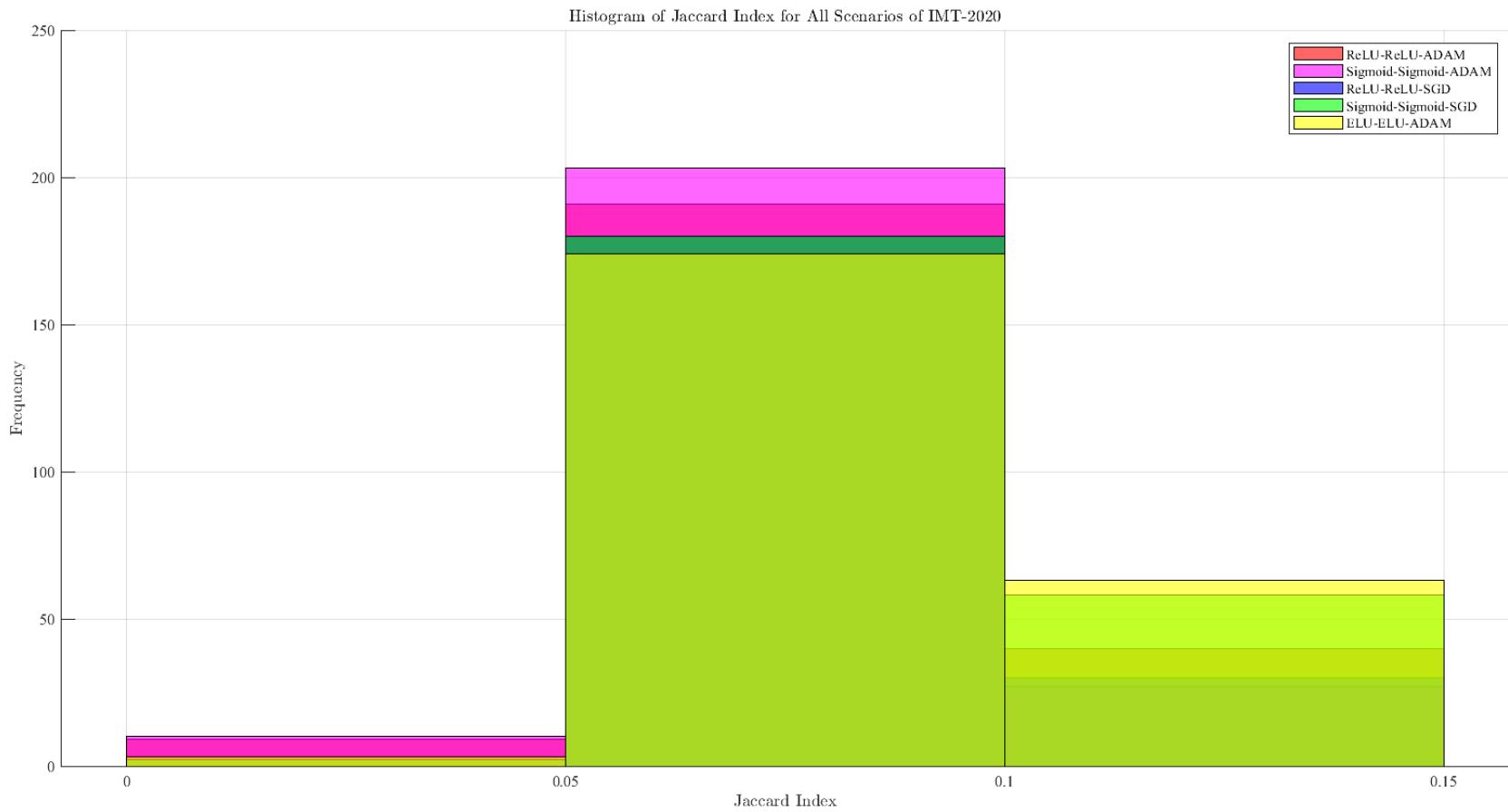


Fig. 6.14 IMT-2020- Histogram of Jaccard Indices of All Scenarios



6.3.6 ECDF of JIs in C2CM

This section shows the ECDF of JIs in C2CM Channel Model. It achieved the second specific objective of this study, to provide statistical comparison of JIs of different configurations using Sigmoid and ReLU activation functions and ADAM and SGD optimizers.

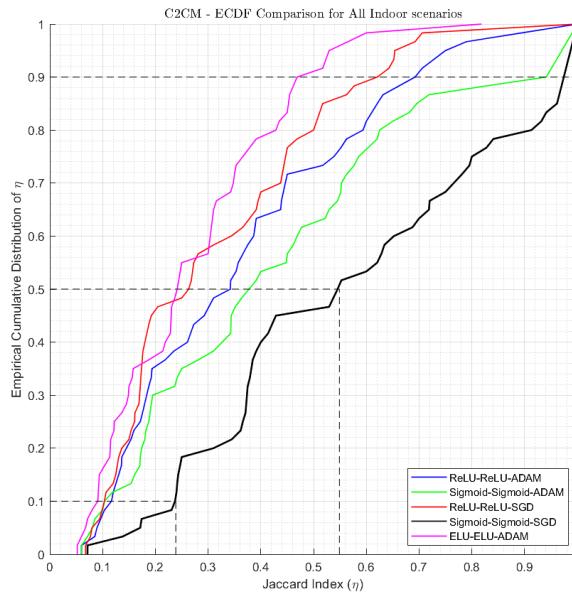


Fig. 6.15 C2CM ECDF Comparison for All Indoor Scenarios

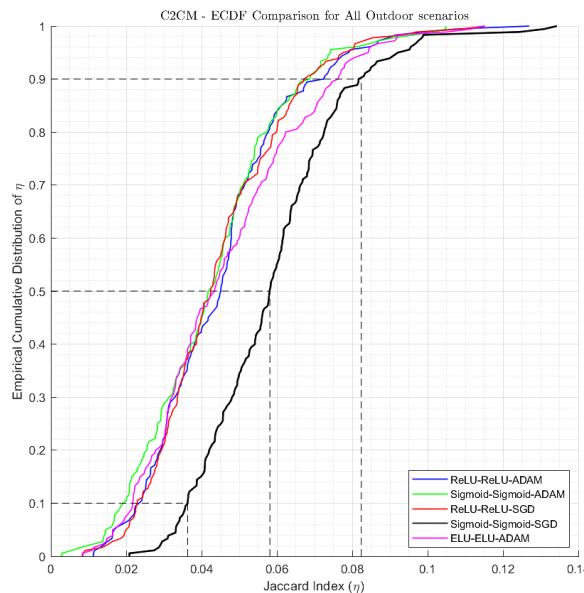


Fig. 6.16 C2CM ECDF Comparison for All Outdoor Scenarios

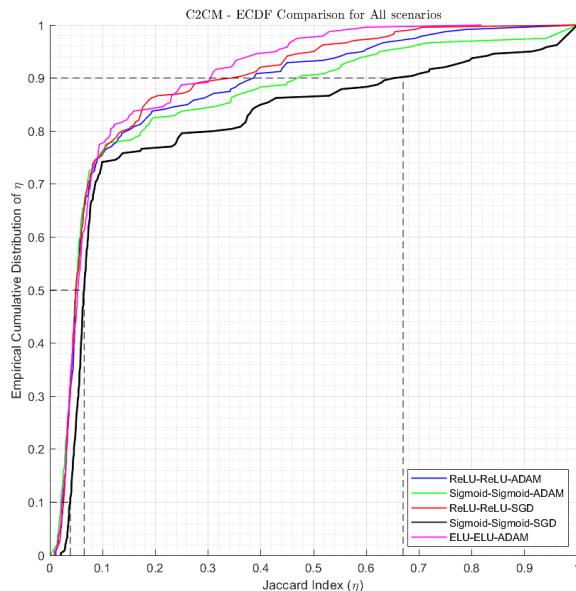


Fig. 6.17 C2CM ECDF Comparison for All Scenarios

The Table 6.16 shows the percentile differences of ECDF in each scenario in C2CM.

TABLE 6.16 C2CM- PERCENTILE DIFFERENCES OF ECDF IN EACH SCENARIOS

Scenarios	Configuration	10th	50th	90th
All Indoor Scenarios	ReLU-ReLU-ADAM	0.118442	0.342481	0.699095
	Sigmoid-Sigmoid-ADAM	0.111046	0.391304	0.970588
	ReLU-ReLU-SGD	0.104916	0.265725	0.631773
	Sigmoid-Sigmoid-SGD	0.238644	0.549043	1
	ELU-ELU-ADAM	0.09233	0.24569	0.492996
All Outdoor Scenarios	ReLU-ReLU-ADAM	0.023778	0.045081	0.072546
	Sigmoid-Sigmoid-ADAM	0.019629	0.042084	0.068792
	ReLU-ReLU-SGD	0.022945	0.042429	0.067728
	Sigmoid-Sigmoid-SGD	0.036323	0.058126	0.082377

Continued on next page

*Continued from previous page*

Scenarios	Configuration	10th	50th	90th
All Scenarios	eLU-eLU-ADAM	0.021771	0.043496	0.076333
	ReLU-ReLU-ADAM	0.02527	0.049355	0.388834
	Sigmoid-Sigmoid-ADAM	0.020932	0.049447	0.473011
	ReLU-ReLU-SGD	0.026224	0.049357	0.354802
	Sigmoid-Sigmoid-SGD	0.038699	0.065045	0.669837
	eLU-eLU-ADAM	0.022758	0.053108	0.310345

Fig 6.15 shows the ECDF for the JI of all indoor scenarios in C2CM. The ECDF shows that Sigmoid-Sigmoid-SGD significantly has the highest 10th, 50th, and 90th percentiles than all configuration, specifically on eLU-eLU-ADAM. The Sigmoid-Sigmoid-SGD configuration has an increase of 0.238644 in 10th percentile, 0.549043 in 50th, and 1 in 90th percentile. Therefore, the Sigmoid-Sigmoid-SGD is the leading configuration for all indoor scenarios in C2CM channel model.

Furthermore, Fig 6.16 shows the ECDF for the JI of all outdoor scenarios in C2CM. The ECDF shows that Sigmoid-Sigmoid-SGD also significantly ranks the highest 10th, 50th, and 90th percentiles than all configuration, specifically on eLU-eLU-ADAM. The Sigmoid-Sigmoid-SGD configuration has an increase of 0.036323 in 10th percentile, 0.058126 in 50th, and 0.082377 in 90th percentile. It can be observed the decrease in scores in outdoor scenario having a 0.058126 50th percentile. The data implies that Sigmoid-Sigmoid-SGD is the leading configuration for all outdoor scenarios in C2CM channel model.

Finally, Fig 6.17 shows the ECDF for the JI of all scenarios in C2CM. The ECDF shows that Sigmoid-Sigmoid-SGD is the superior configuration and significantly ranks the highest for 10th, 50th, and 90th percentiles than all configuration making it a sufficient configuration for C2CM channel model. The Sigmoid-Sigmoid-SGD configuration has an increase of 0.038699 in 10th percentile, 0.065045 in 50th,



and 0.669837 in 90th percentile. It can be observed that the 50th percentile having a score of 0.065045 shows the increase in percentile than the outdoor scenarios because of the indoor scenarios. The data implies that Sigmoid-Sigmoid-SGD is the superior configuration for overall scenarios in C2CM channel model. Thus, making it the sufficient activation function and increasing the JI score from the past study that uses eLU-eLU-ADAM configuration.

In summary, comparing Sigmoid-Sigmoid-SGD to the previous configuration of k -DAE (eLU-eLU-ADAM), it increased by 14.63% in 10th percentile, 30.34% in 50th percentile, and 50.70% for indoor scenarios in C2CM. For outdoor scenarios, it increased by 1.46% in 10th percentile, 1.46% in 50th percentile, and 1% in 90th percentile. For all scenarios, it increased by 1.59% in 10th percentile, 1.19% in 50th percentile, and 35.95% in 90th percentile. Thus, Sigmoid-Sigmoid-SGD is sufficient configuration for C2CM channel model.

6.3.7 ECDF of JIs in IMT-2020

This section shows the ECDF of JIs in IMT-2020 Channel Model. This section is the continuation of Sec 6.3.6. It achieved the second specific objective of this study, to provide statistical comparison of JIs of different configurations using Sigmoid and ReLU activation functions and ADAM and SGD optimizers.

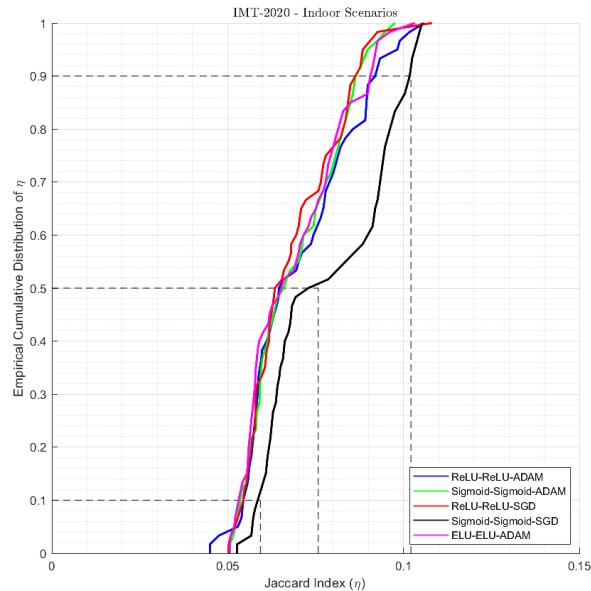


Fig. 6.18 IMT-2020 ECDF Comparison for All Indoor Scenarios

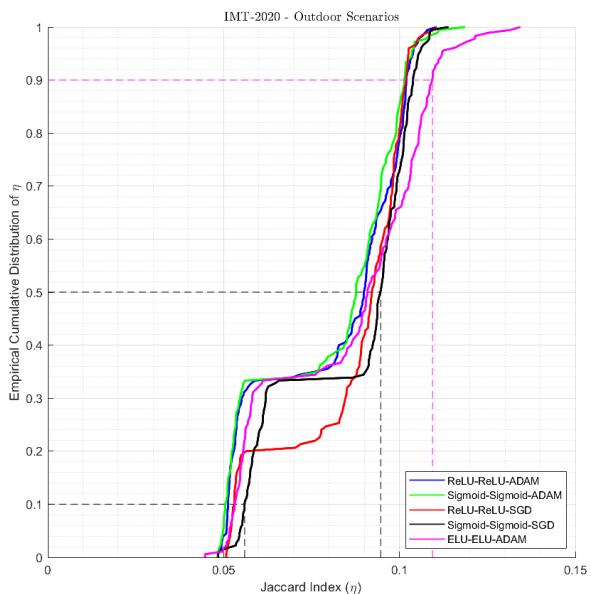


Fig. 6.19 IMT-2020 ECDF Comparison for All Outdoor Scenarios

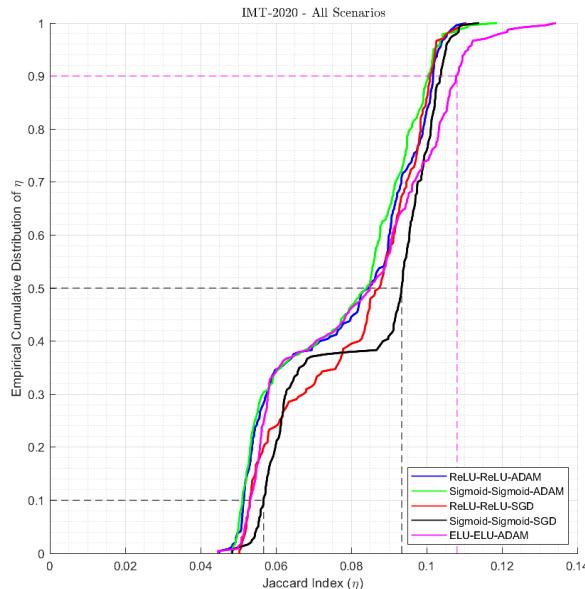


Fig. 6.20 IMT-2020 ECDF Comparison for All Scenarios

The Table 6.17 below shows the percentile differences of ECDF in each scenario in IMT-2020.

TABLE 6.17 IMT-2020- PERCENTILE DIFFERENCES OF ECDF FOR EACH SCENARIO

Scenarios	Configuration	10th	50th	90th
All Indoor Scenarios	ReLU-ReLU-ADAM	0.055070603	0.065263158	0.092280702
	Sigmoid-Sigmoid-ADAM	0.053915276	0.066324347	0.087017544
	ReLU-ReLU-SGD	0.054557125	0.064441592	0.087017544
	Sigmoid-Sigmoid-SGD	0.059306804	0.075755242	0.102105263
	ELU-ELU-ADAM	0.053658537	0.06558836	0.090877193
All Outdoor Scenarios	ReLU-ReLU-ADAM	0.051287882	0.089953658	0.101973684
	Sigmoid-Sigmoid-ADAM	0.050570391	0.087674543	0.101466165
	ReLU-ReLU-SGD	0.053024352	0.092188017	0.10189755

Continued on next page

*Continued from previous page*

Scenarios	Configuration	10th	50th	90th
All Scenarios	Sigmoid-Sigmoid-SGD	0.055980486	0.094656979	0.10387218
	ELU-ELU-ADAM	0.053200932	0.090944173	0.109379699
	ReLU-ReLU-ADAM	0.051483591	0.08483947	0.101717721
	Sigmoid-Sigmoid-ADAM	0.051114856	0.084160874	0.100451128
	ReLU-ReLU-SGD	0.053086369	0.08755379	0.100989405
All Scenarios	Sigmoid-Sigmoid-SGD	0.05675272	0.093393862	0.103752995
	ELU-ELU-ADAM	0.053200932	0.084928392	0.108079379

Fig 6.18 shows the ECDF for the JI of all indoor scenarios in IMT-2020. The ECDF shows that Sigmoid-Sigmoid-SGD significantly has the highest 10th, 50th, and 90th percentiles than all configurations. The Sigmoid-Sigmoid-SGD configuration has an increase of 0.059306804 in 10th percentile, 0.075755242 in 50th, and 0.102105263 in 90th percentile. Therefore, the Sigmoid-Sigmoid-SGD is the leading configuration for all indoor scenarios in IMT-2020 channel model.

Furthermore, Fig 6.19 shows the ECDF for the JI of all outdoor scenarios in IMT-2020. The ECDF shows that Sigmoid-Sigmoid-SGD also significantly ranks the highest 10th and 50th percentiles than all configurations. The Sigmoid-Sigmoid-SGD configuration has an increase of 0.055980485 in 10th percentile and 0.094656979 in 50th. However, eLU-eLU-ADAM scored 0.1094 in 90th percentile. It can be observed that the scores in outdoor scenario increased having a 0.094656979 50th percentile. This implied that k -DAE is more effective in outdoor scenario than C2CM. Nevertheless, the data suggests that Sigmoid-Sigmoid-SGD is the leading configuration for all outdoor scenarios in IMT-2020 channel model.

Finally, Fig 6.20 shows the ECDF for the JI of all scenarios in IMT-2020. The ECDF shows that Sigmoid-Sigmoid-SGD is the superior configuration for 10th and 50th percentiles than all configurations making it also a sufficient configuration



for IMT-2020 channel model. The Sigmoid-Sigmoid-SGD configuration has an increase of 0.05675272 in 10th percentile and 0.093393862 in 50th. However, eLU-eLU-ADAM scored 0.1081 in 90th percentile. This implies eLU-eLU-ADAM has higher extreme values however low median and minimum values. It can be observed that there is no significant increase of values compared to indoor and outdoor scenarios. This indicates that the data has a more consistent variability than C2CM. In summary, the data infers that Sigmoid-Sigmoid-SGD is the superior configuration for overall scenarios in IMT-2020 channel model. Thus, making it the sufficient activation function configuration.

In summary, comparing of Sigmoid-Sigmoid-SGD to eLU-eLU-ADAM in IMT-2020, it increased 1% in 10th percentile, 1.1% in 50th percentile, and 1.2 % in 90th percentile in indoor scenarios. For outdoor scenarios, it increased 0.2% for 10th percentile and 0.4% in 50th percentile. For all scenarios, it increased 0.4% for 10th percentile and 1% for 50th percentile. Thus, Sigmoid-Sigmoid-SGD is sufficient among all configurations.

6.4 Reconstruction Error

To attain the third specific objective of this study, which is to observe the associative correlation between the autoencoder reconstruction error and JI, Python was used to record the RE of the modified k -DAE algorithm in clustering the C2CM and IMT-2020, MATLAB software was utilized to evaluate the correlation of the RE and CVIs by creating correlation plots.

6.4.1 Correlations of CVIs of Different Configurations in C2CM

The following Figures show the correlation plots of each CVI and RE of all scenarios in C2CM. The correlation plot of every CVI with RE is highlighted in the rightmost part of the graph with the data plotted in black.

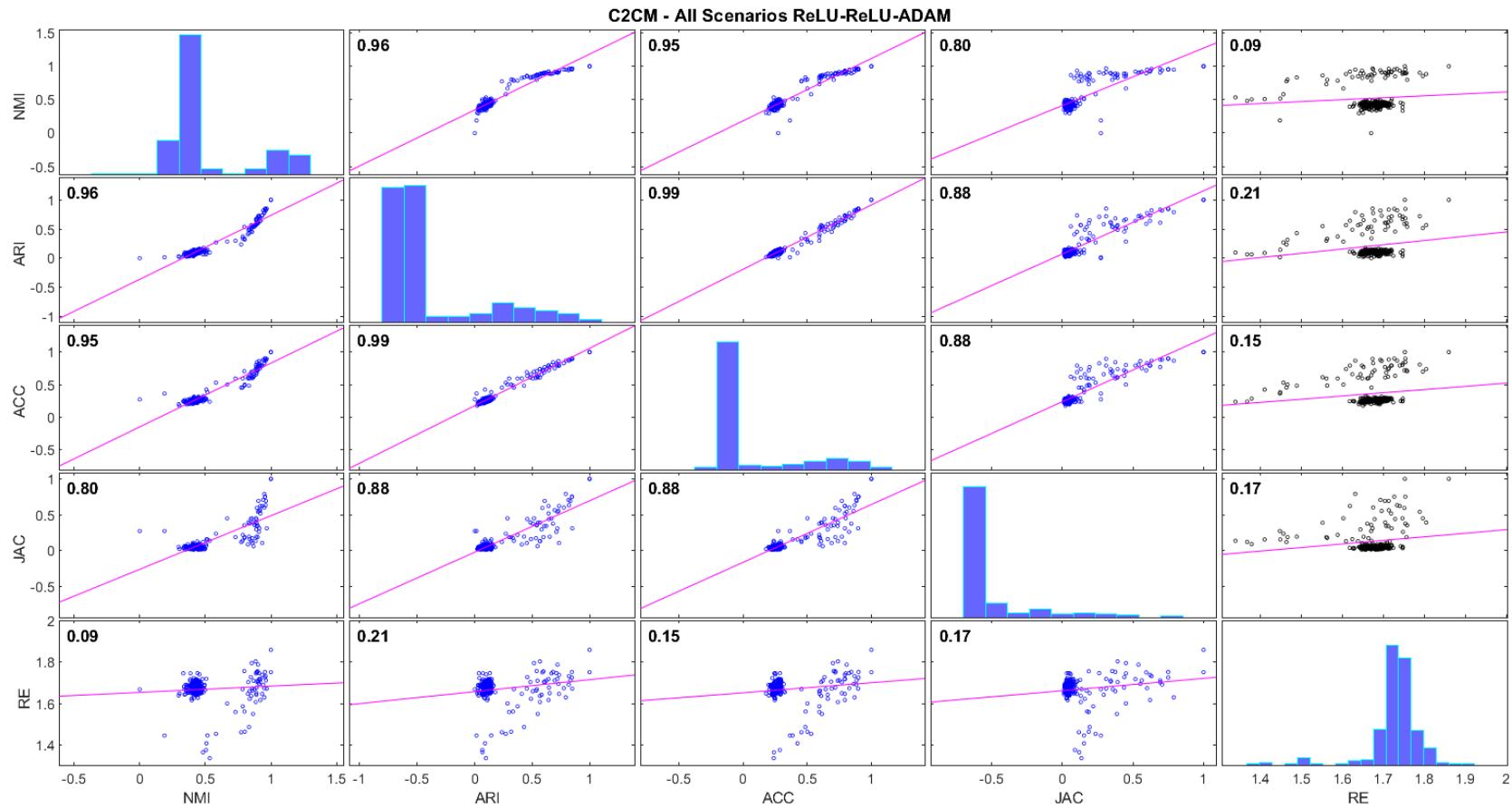


Fig. 6.21 C2CM-Correlation Plots of CVIs and Reconstruction Error Results in ReLU-ReLU-ADAM All Scenarios

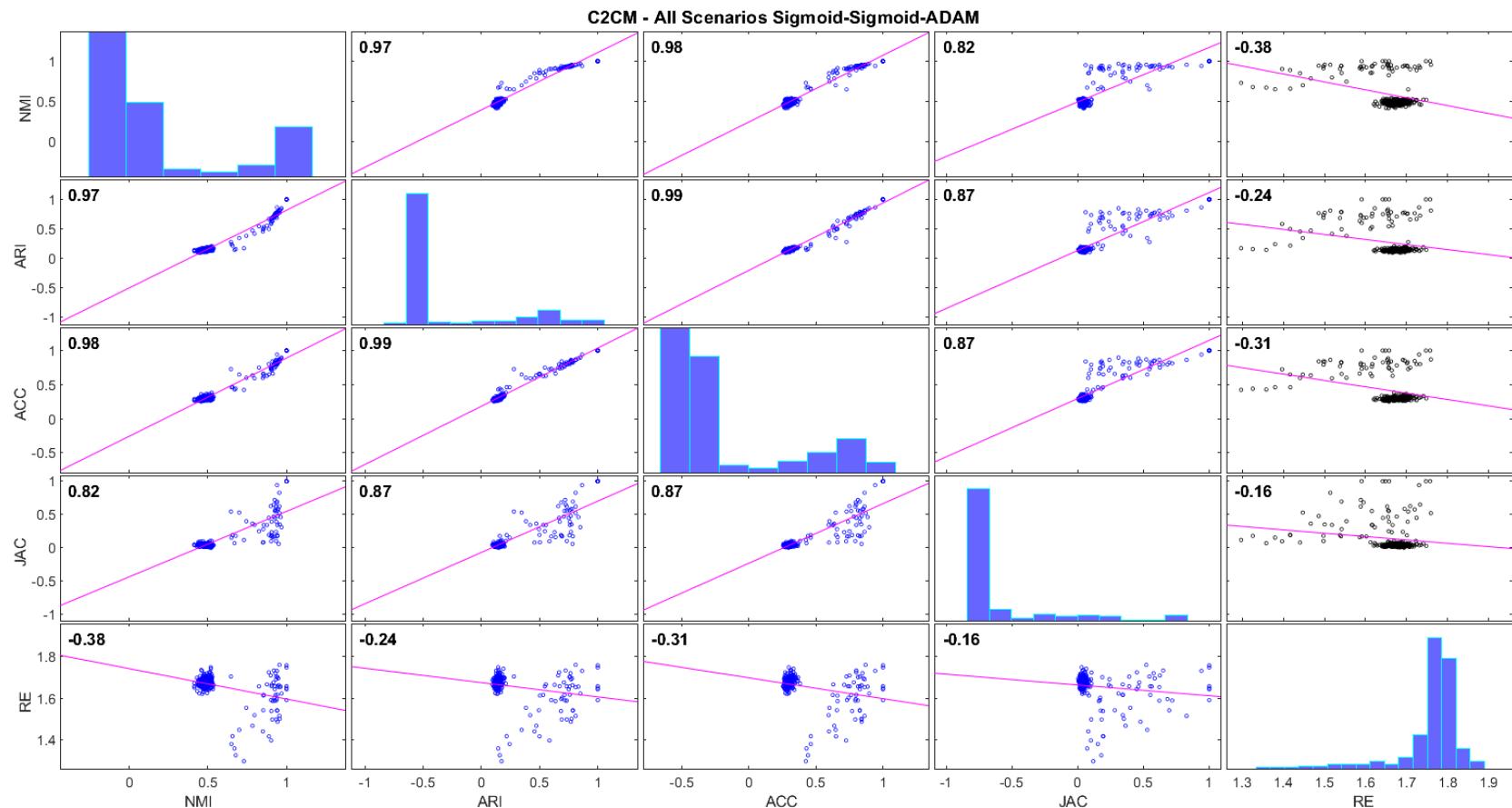


Fig. 6.22 C2CM-Correlation Plots of CVIs and Reconstruction Error Results in Sigmoid-Sigmoid-ADAM All Scenarios

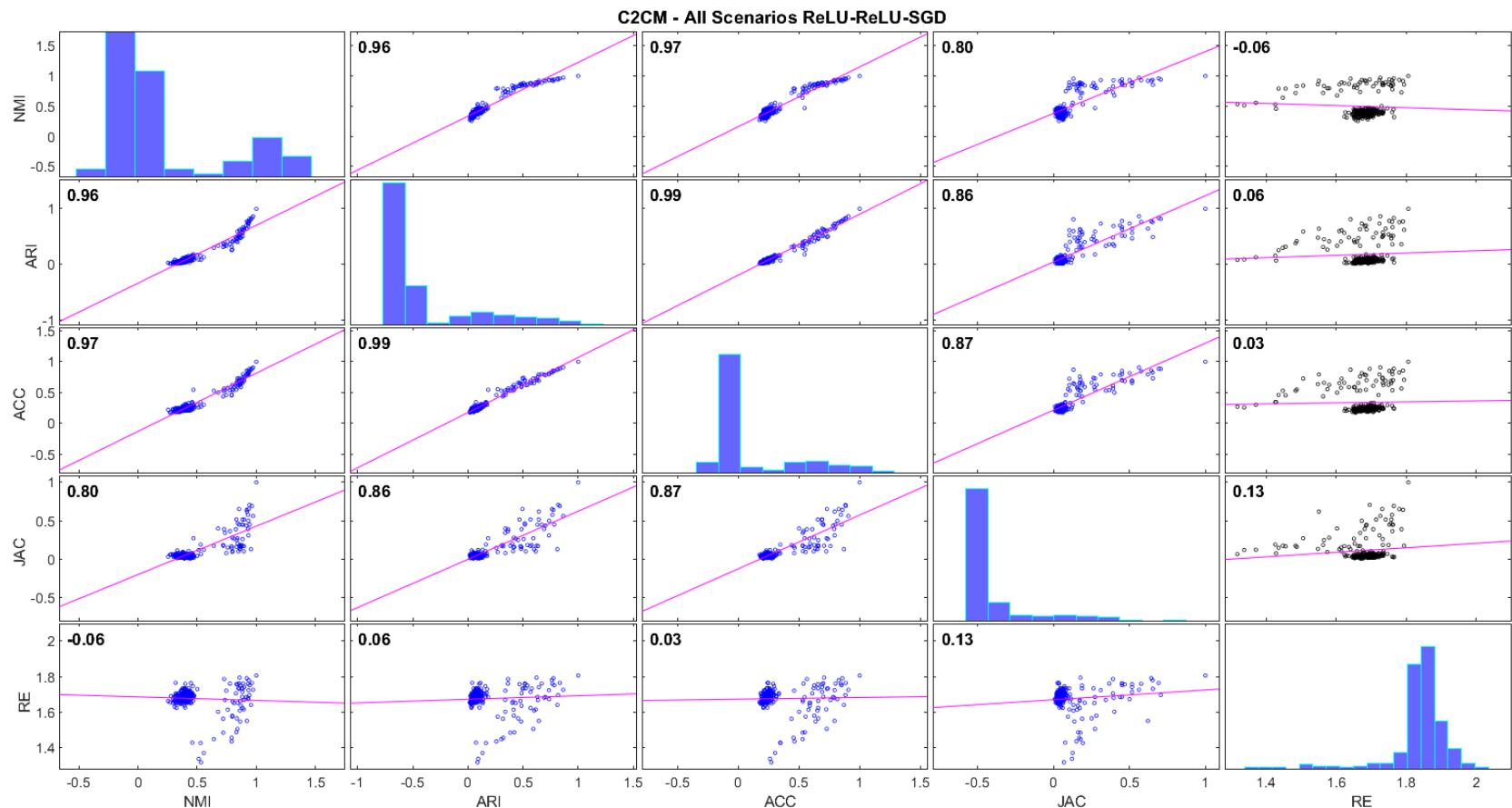


Fig. 6.23 C2CM-Correlation Plots of CVIs and Reconstruction Error Results in ReLU-ReLU-SGD All Scenarios

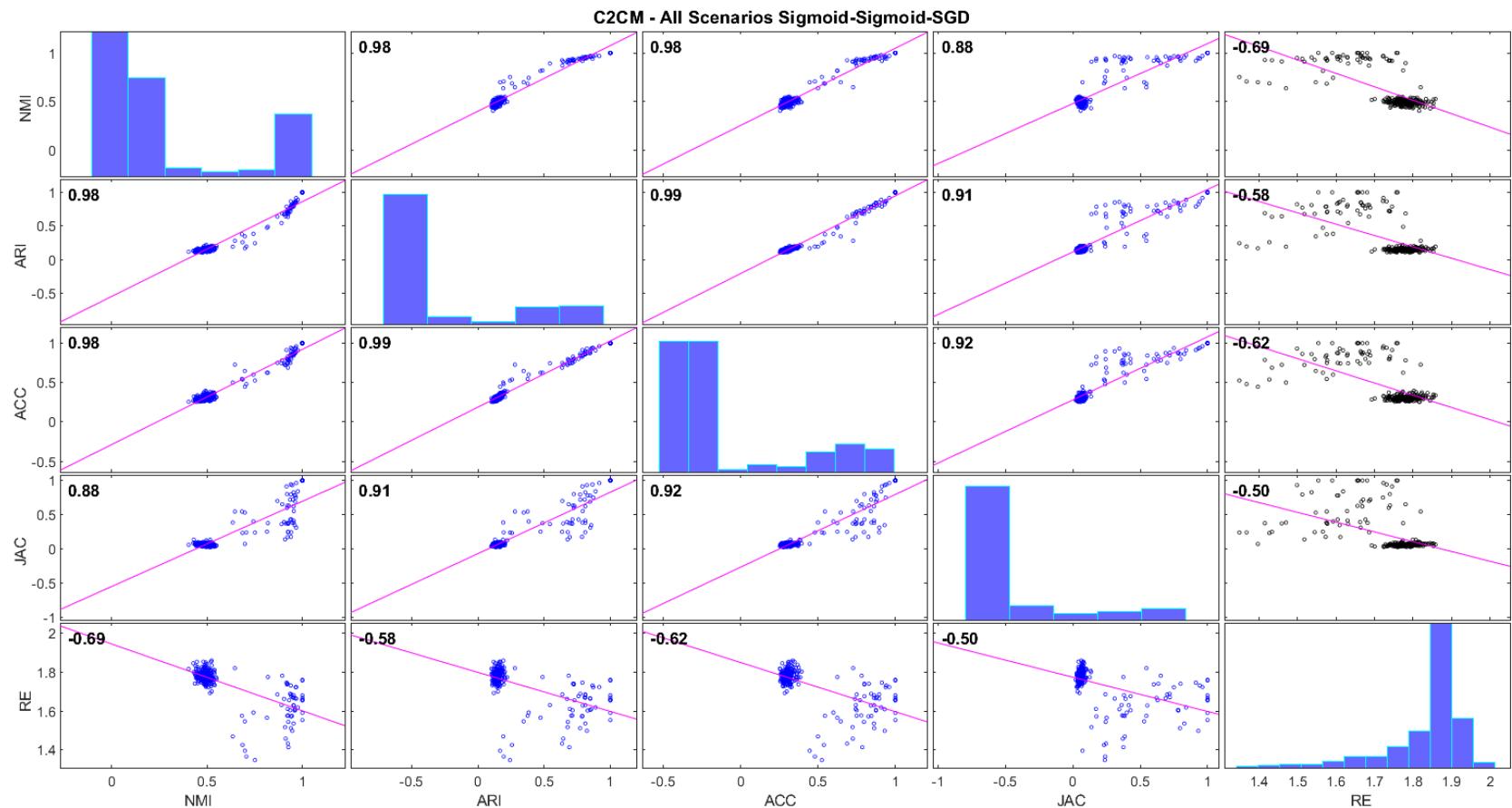


Fig. 6.24 C2CM-Correlation Plots of CVIs and Reconstruction Error Results in Sigmoid-Sigmoid-SGD All Scenarios



Fig 6.21, Fig 6.22, Fig 6.23, and Fig 6.24 display the correlation plots of the CVIs and RE of each configuration in all scenarios of C2CM. The results revealed that the correlation between each CVI and RE varies across different scenarios. Some showed a very weak positive correlation, others exhibited negative correlations. In general, ReLU-ReLU configurations showed very weak positive correlations between CVIs and RE, suggesting there was little to no linear relationship between the given metrics. However, Sigmoid-Sigmoid configurations, especially with the SGD as optimizer, showed moderate to strong negative correlations, indicating that as the CVI increases, the RE tends to decrease. With the result provided, negative correlations in the Sigmoid-Sigmoid-SGD optimizer are particularly interesting and may warrant further investigation to understand the underlying cause.

6.4.2 Correlations of CVIs of Different Configurations in IMT-2020

The following Figures show the correlation plots of each CVI and RE of all scenarios in IMT-2020. The correlation plot of every CVI with RE is highlighted in the rightmost part of the graph with the data plotted in black.

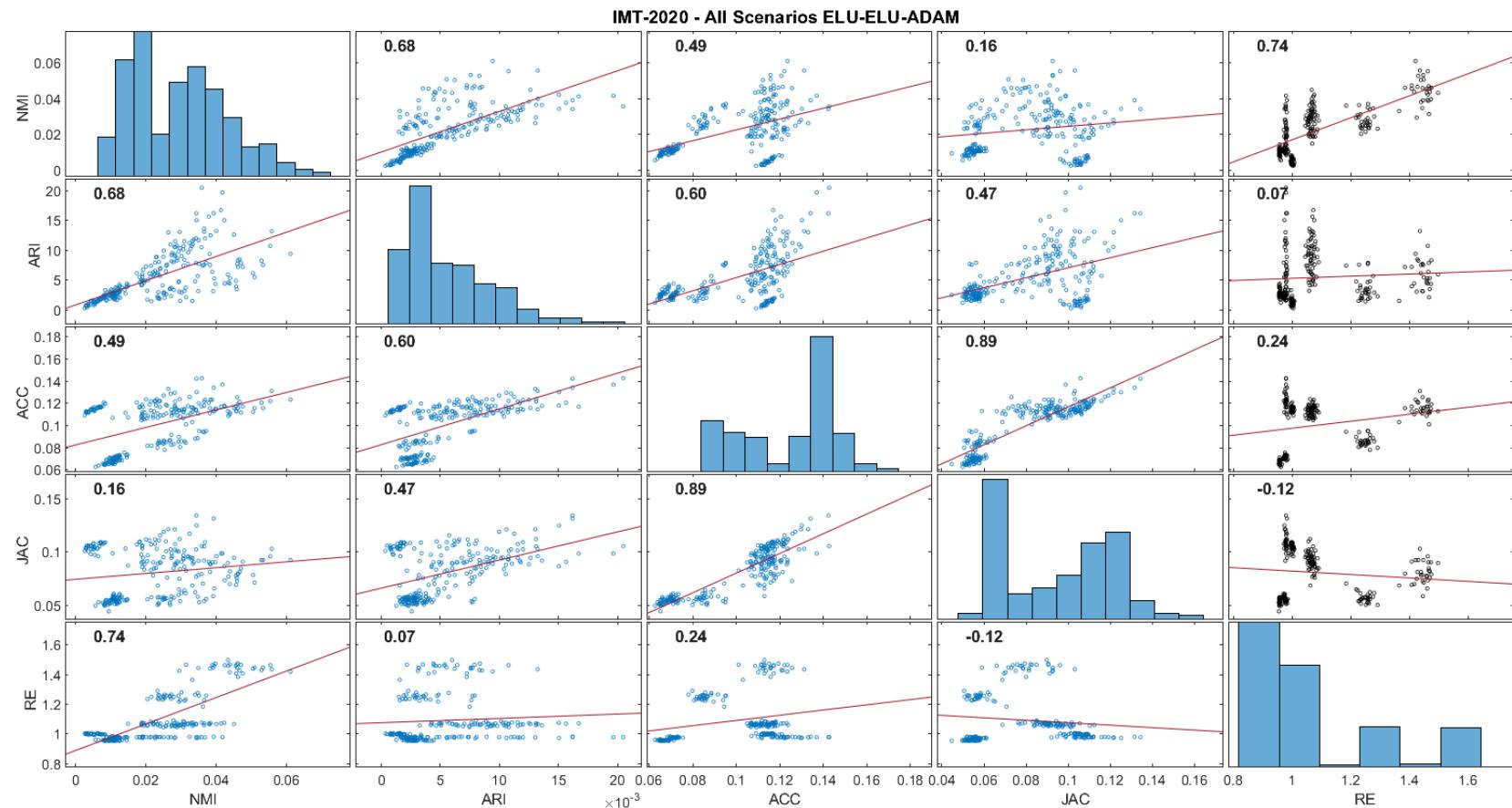


Fig. 6.25 IMT-2020-Correlation Plots of CVIs and Reconstruction Error Results in eLU-eLU-ADAM All Scenarios

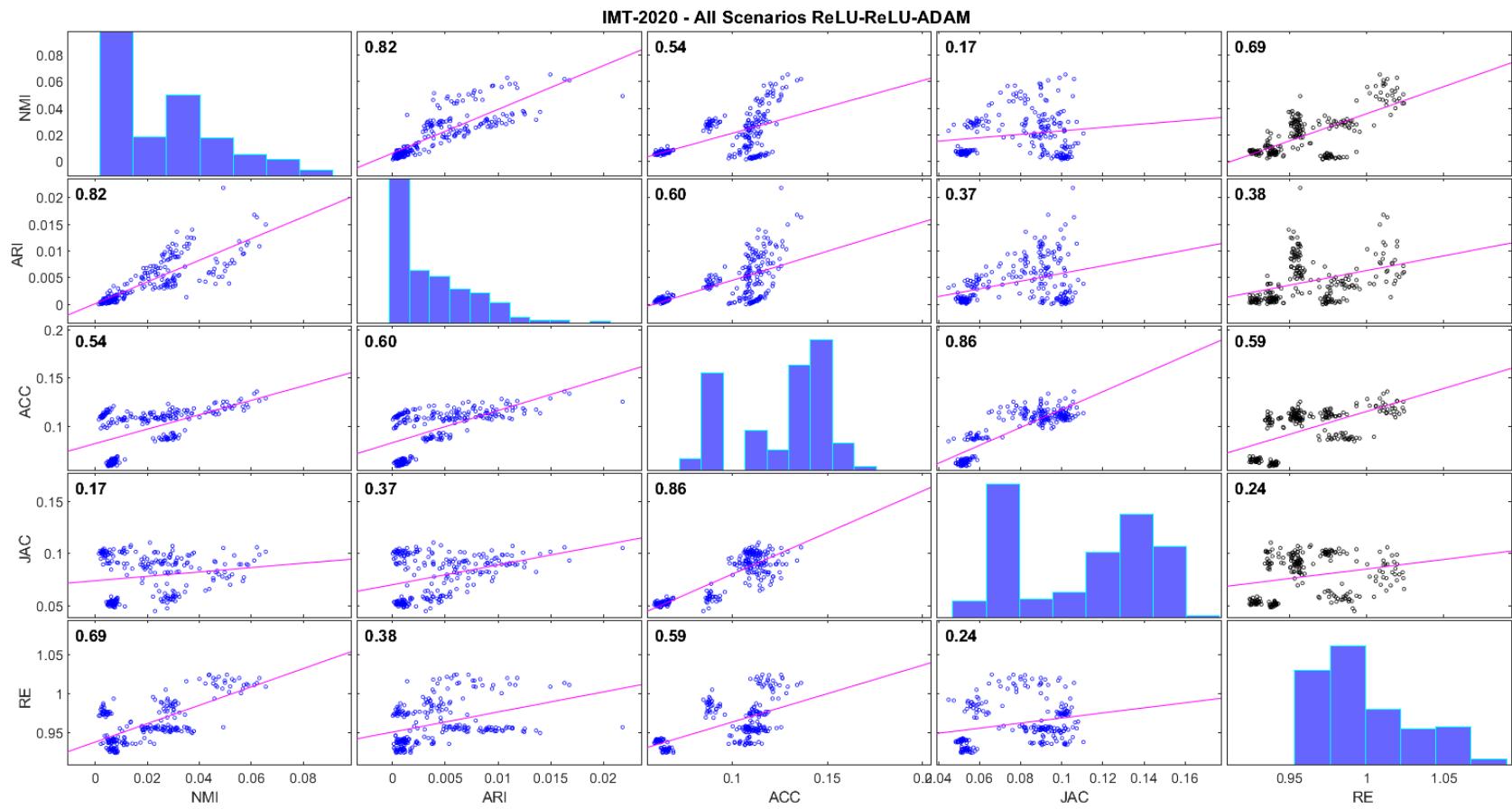


Fig. 6.26 IMT-2020-Correlation Plots of CVIs and Reconstruction Error Results in ReLU-ReLU-ADAM All Scenarios

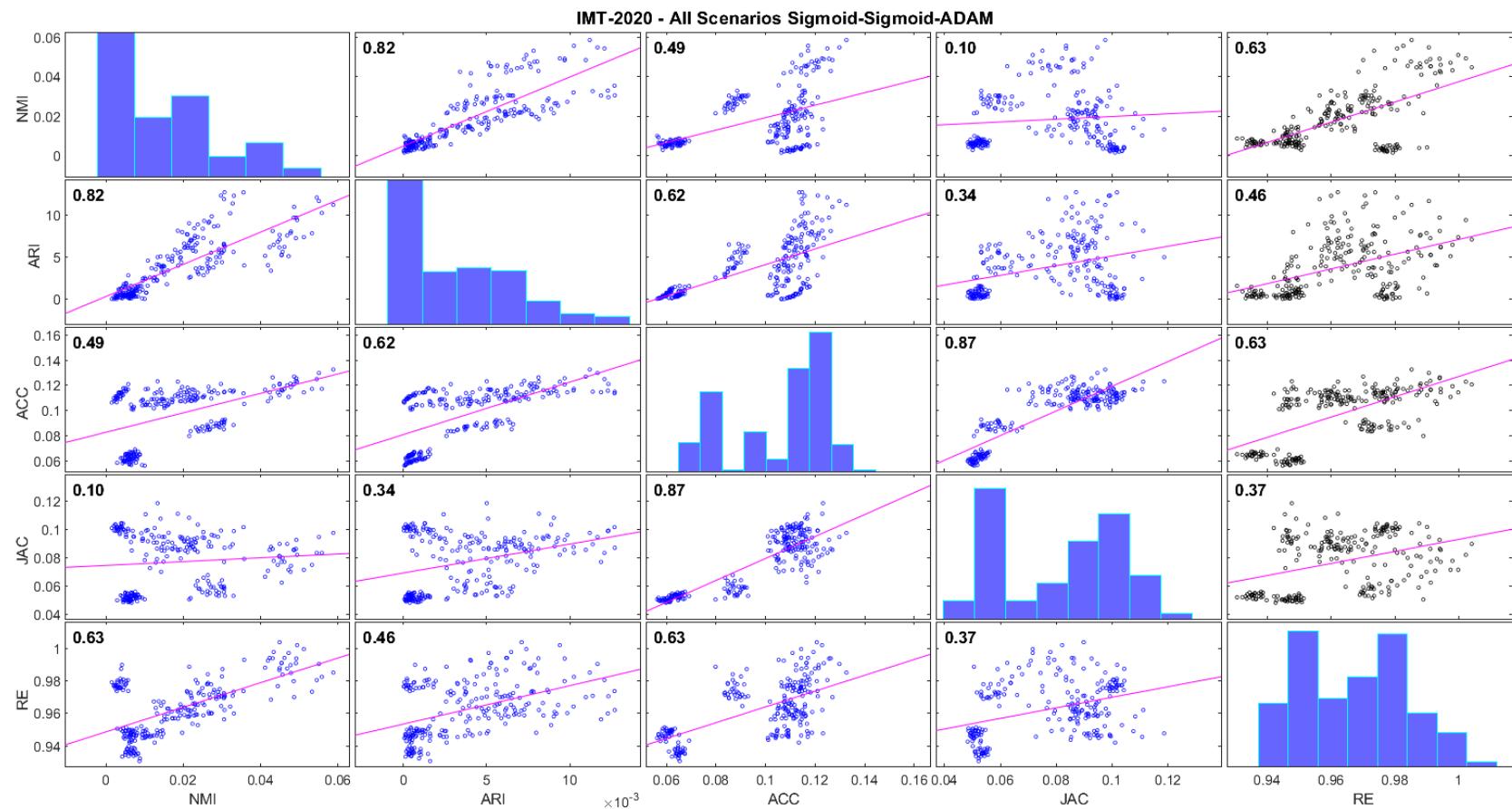


Fig. 6.27 IMT-2020-Correlation Plots of CVIs and Reconstruction Error Results in Sigmoid-Sigmoid-ADAM All Scenarios

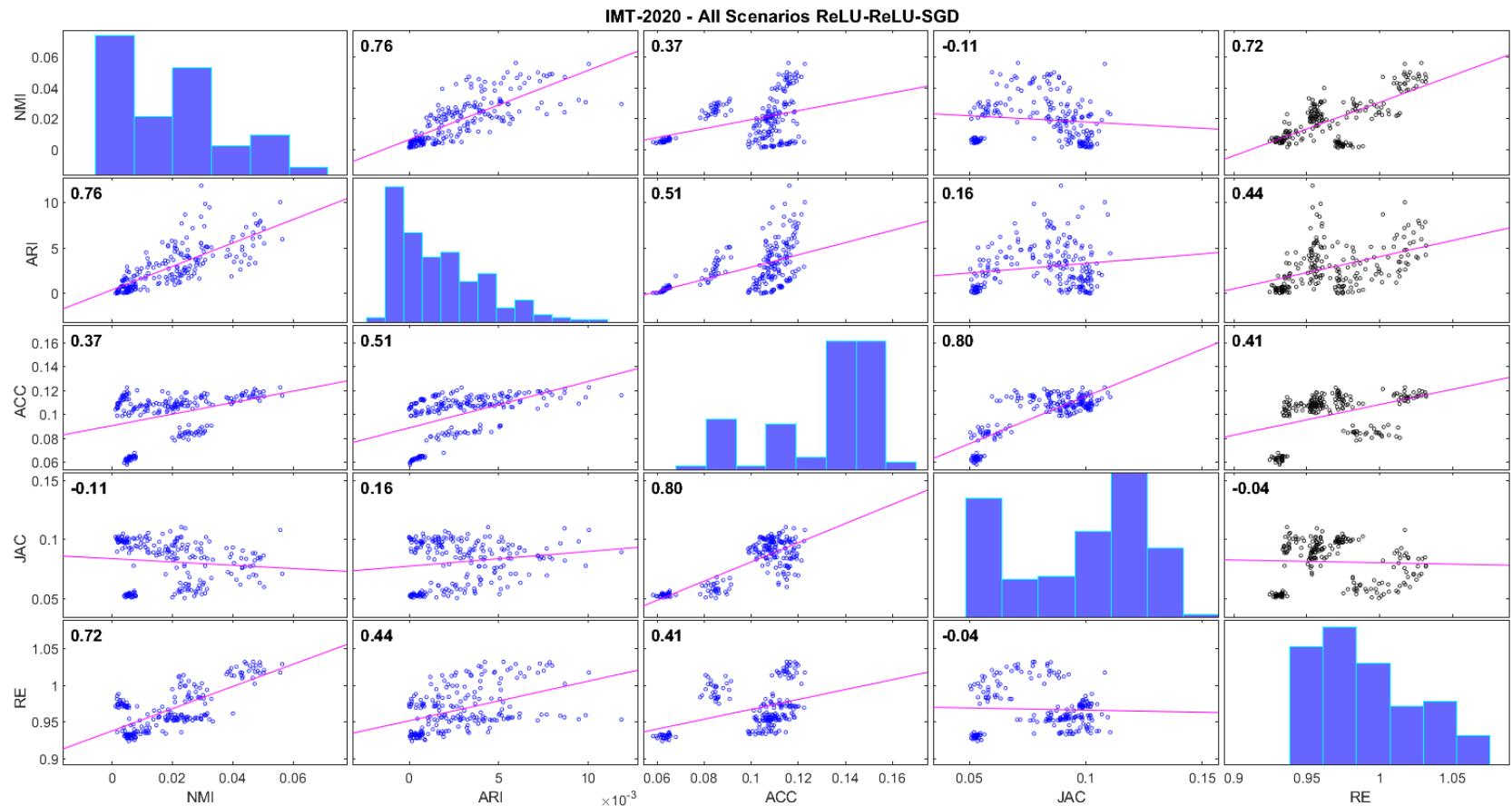


Fig. 6.28 IMT-2020-Correlation Plots of CVIs and Reconstruction Error Results in ReLU-ReLU-SGD All Scenarios

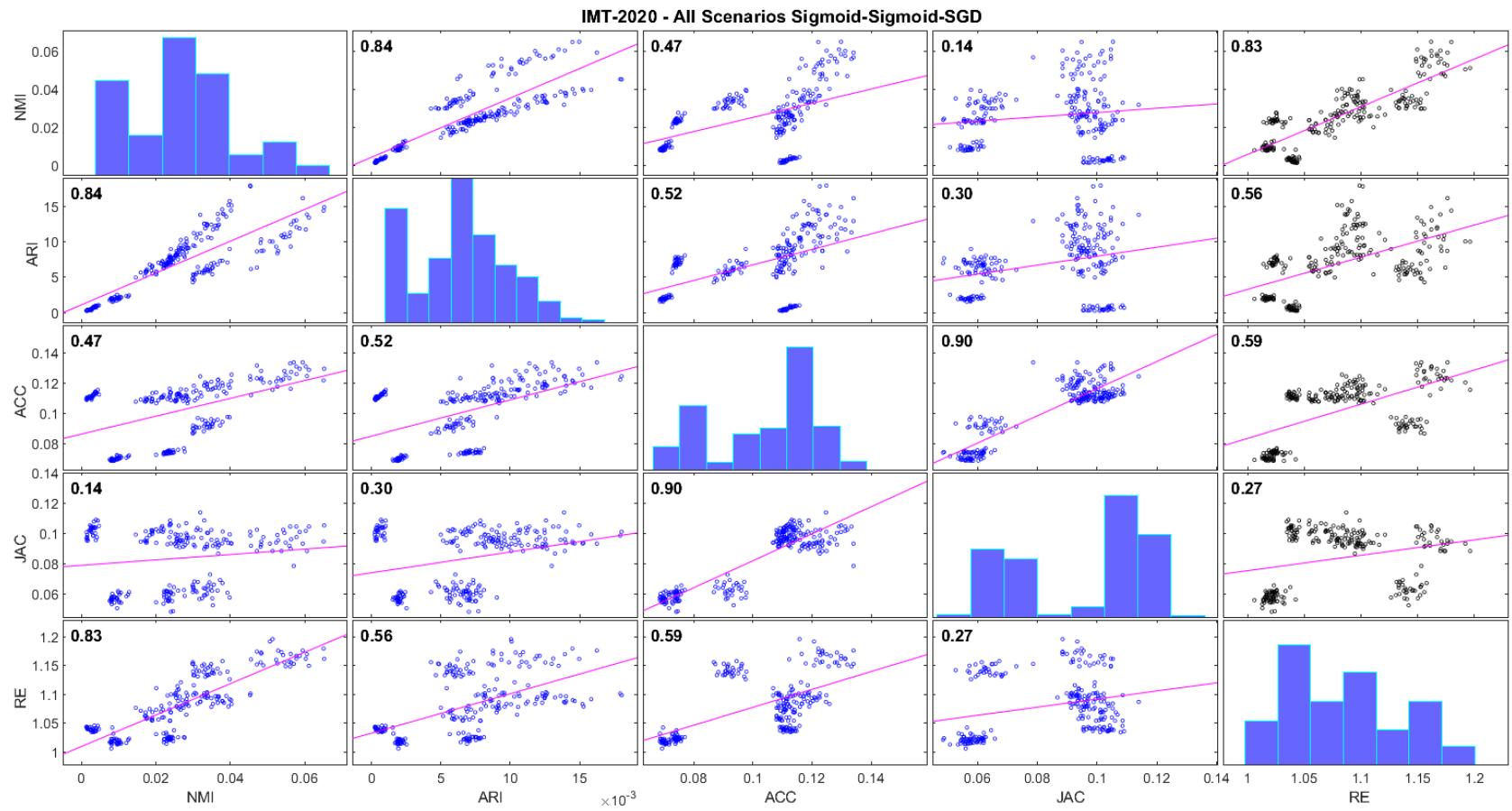


Fig. 6.29 IMT-2020-Correlation Plots of CVIs and Reconstruction Error Results in Sigmoid-Sigmoid-SGD All Scenarios



Fig 6.25, Fig 6.26, Fig 6.27, Fig 6.28, and Fig 6.29 illustrate correlation plots depicting the relationship between each CVI and RE for every configuration across all scenarios of IMT-2020. Generally, there existed a trend of moderate to strong positive correlation between CVIs and RE across all configurations. However, for Sigmoid-Sigmoid-SGD, the correlation between JI and RE appeared to be the weakest compared to other CVIs such as NMI, ARI, and ACC.

Specifically, in eLU-eLU-ADAM, NMI, ARI, and ACC demonstrated a weak to strong positive relationship with RE, while JI showed a weak negative relationship. In ReLU-ReLU-ADAM, all CVIs exhibited a moderate to strong positive relationship with RE. Conversely, Sigmoid-Sigmoid-ADAM displayed a weak to moderate negative relationship between each CVI and RE. In ReLU-ReLU-SGD, only NMI demonstrated a very weak negative relationship with RE, while the remaining CVIs showed a very weak positive relationship. Notably, Sigmoid-Sigmoid-SGD showcased a strong negative relationship between each CVI and RE, indicating distinct performance characteristics.

6.4.3 Correlation of JIs and REs in C2CM and IMT-2020

The Table 6.18 and Table 6.19 show the correlation coefficients (r) of each CVIs and REs in C2CM.

TABLE 6.18 CORRELATION COEFFICIENT (r) OF JI AND RE OF C2CM

C2CM	Configuration	Correlation Coefficient (r)
All Indoor Scenarios	ReLU-ReLU-ADAM	0.91
	Sigmoid-Sigmoid-ADAM	1
	ReLU-ReLU-SGD	0.96
	Sigmoid-Sigmoid-SGD	0.66
All Outdoor Scenarios	ReLU-ReLU-ADAM	0.26
	Sigmoid-Sigmoid-ADAM	0.03

Continued on next page

*Continued from previous page*

C2CM	Configuration	Correlation Coefficient (r)
All Scenarios	ReLU-ReLU-SGD	0.58
	Sigmoid-Sigmoid-SGD	0.34
	ReLU-ReLU-ADAM	0.17
	Sigmoid-Sigmoid-ADAM	-0.16
	ReLU-ReLU-SGD	0.13
	Sigmoid-Sigmoid-SGD	-0.5

Table 6.18 showcases the associative relationships between the JI and the RE for various configurations across all indoor, all outdoor, and all scenarios of C2CM.

In indoor scenarios, configurations such as ReLU-ReLU-ADAM and ReLU-ReLU-SGD demonstrated strong to very strong positive associations with correlation coefficient of 0.91 and 0.96, respectively, indicating robust correlations between JI and RE. Sigmoid-Sigmoid-ADAM configuration exhibited a perfect positive association, suggesting complete correlation, while Sigmoid-Sigmoid-SGD configuration showed a moderate positive association, indicating a moderate correlation.

For outdoor scenarios, ReLU-ReLU-SGD and Sigmoid-Sigmoid-SGD configurations demonstrated moderate positive associations with correlation coefficients of 0.58 and 0.34, respectively, suggesting reasonably correlated relationships. ReLU-ReLU-ADAM configuration showed a weak positive association with a value of 0.26, indicating a slight correlation, while Sigmoid-Sigmoid-ADAM configuration exhibited a very weak positive association, indicating minimal correlation.

Overall, ReLU-ReLU-ADAM configuration displayed a weak positive association, while Sigmoid-Sigmoid-SGD configuration presented a moderate negative association, suggesting a notable inverse correlation. Other configurations showed varying degrees of association between JI and RE.



These findings offered valuable insights into the relationship between JI and RE across different configurations and scenarios, contributing to a deeper understanding of clustering performance and reconstruction quality in various settings.

TABLE 6.19 P-VALUE AND CORRELATION COEFFICIENT (R) OF JI AND RE OF IMT-2020

IMT-2020	Configuration	Correlation Coefficient (r)
All Indoor Scenarios	eLU-eLU-ADAM	0.82
	ReLU-ReLU-ADAM	0.71
	Sigmoid-Sigmoid-ADAM	0.58
	ReLU-ReLU-SGD	0.58
	Sigmoid-Sigmoid-SGD	0.67
All Outdoor Scenarios	eLU-eLU-ADAM	0.39
	ReLU-ReLU-ADAM	0.7
	Sigmoid-Sigmoid-ADAM	0.71
	ReLU-ReLU-SGD	0.68
	Sigmoid-Sigmoid-SGD	0.69
All Scenarios	eLU-eLU-ADAM	-0.12
	ReLU-ReLU-ADAM	0.24
	Sigmoid-Sigmoid-ADAM	0.37
	ReLU-ReLU-SGD	-0.04
	Sigmoid-Sigmoid-SGD	0.27

Table 6.19 showcases the associative relationships between the JI and the RE for various configurations across all indoor, all outdoor, and all scenarios of IMT-2020.

For all indoor scenarios, the eLU-eLU-ADAM configuration demonstrated the strongest positive relationship with a coefficient of 0.82 between JI and RE, followed by ReLU-ReLU-ADAM at 0.71, both indicating a strong positive association. Sigmoid-Sigmoid-SGD and ReLU-ReLU-SGD configurations share a moderate



positive association with coefficients of 0.58. The Sigmoid-Sigmoid-ADAM configuration exhibited a slightly lower coefficient of 0.67 compared to eLU-eLU-ADAM, indicating a moderate positive association.

In contrast, for all outdoor scenarios, eLU-eLU-ADAM shows a weak positive association with a coefficient of 0.39, while ReLU-ReLU-ADAM and Sigmoid-Sigmoid-ADAM displayed strong positive associations with coefficients of 0.70 and 0.71, respectively. Both ReLU-ReLU-SGD and Sigmoid-Sigmoid-SGD configurations presented moderate positive associations with coefficients of 0.68 and 0.69, respectively.

Across all scenarios, eLU-eLU-ADAM exhibited a weak negative association with a coefficient of -0.12, while ReLU-ReLU-ADAM showed a weak positive association with a coefficient of 0.24. Sigmoid-Sigmoid-ADAM configuration demonstrated a moderate positive association with a coefficient of 0.37. ReLU-ReLU-SGD exhibited a very weak negative association with a coefficient of -0.04, and Sigmoid-Sigmoid-SGD configuration presented a weak positive association with a coefficient of 0.27.

These observations offered insights into the varying degrees of association between JI and RE across different configurations and scenarios in IMT-2020.

6.5 Summary

The key results on the objectives were accomplished following the methods of the study. The k -DAE script modification according to the C2CM and IMT-2020 Channel models was discussed. Also, the modification of suitable activation functions and optimizers was presented. It was revealed that the combination of sigmoid and SGD optimizer is sufficient for clustering the two channel models. Furthermore, the sigmoid and SGD configuration gives the highest JI result among



the different configurations of k -DAE in clustering the C2CM and IMT-2020 datasets in indoor, outdoor and all scenarios. However, it was observed that C2CM has wider variability clustering scores than IMT-2020 due to scedasticity. Moreover, the associative correlation of C2CM and JI in all scenarios showed moderate to strong negative correlations, indicating that as the CVI increases, the RE tends to decrease. Lastly, in IMT-2020, in Sigmoid-Sigmoid-SGD, RE and JI approaches inverse relationship.



Chapter 7

CONCLUSIONS, RECOMMENDATIONS, AND FUTURE DIRECTIVES

7.1	Summary of Findings	123
7.2	Conclusion	123
7.3	Recommendations	124
7.4	Future Prospects	124



7.1 Summary of Findings

Based on the results of the study, sigmoid activation function is the most sufficient activation function combined with SGD optimizer. Sigmoid-SGD obtained higher CVI results for indoor, outdoor, and all scenarios in C2CM and IMT-2020 channel models, specifically in JI scores. The modified k -DAE increased by 1.6% in 10th percentile, 1.2% in 50th percentile, and 35.95% in 90th percentile for all scenarios from the previous configuration in C2CM. Moreover, in IMT-2020 all scenarios, k -DAE increased by 0.4% for 10th percentile and 1% for 50th percentile using the previous configuration. In C2CM all scenarios, RE and JI correlation showed an inverse relationship. Results suggest that the observed relationship varies depending on the scenario and configuration of the algorithm. Overall, the relationship between JI and RE is associative, meaning the correlation does not imply causation.

7.2 Conclusion

This study evaluates the clustering results of the modified k -DAE algorithm on the C2CM dataset and IMT-2020 channel model dataset. ReLU and Sigmoid are utilized for the first time as the activation functions in k -DAE, while ADAM and SGD are used as optimizers. The following modifications of activation functions and optimizers used are, ReLU-ReLU-ADAM, Sigmoid-Sigmoid-ADAM, ReLU-ReLU-SGD, and Sigmoid-Sigmoid-SGD. These configurations of autoencoders are utilized for the first time in wireless MPC clustering. In conclusion, the validation and accuracy of clusters have been improved after utilizing the modified autoencoder in C2CM scenarios, with a success of implementation in IMT-2020. The Sigmoid-Sigmoid configuration outperforms all the other configurations, both in C2CM and IMT-2020 datasets. Sigmoid-Sigmoid-SGD maintained to provide



the highest JI result among the different configurations of k -DAE. Although there are improvements, the increase is not significant comparing it to previous study. The researchers suggests to explore hyperparameters in tuning the k -DAE for future work.

7.3 Recommendations

Exploring different hyperparameter in tuning the k -DAE algorithm can be considered. Also, modifying the code using different programming languages for comparison of their optimization. Lastly, execute and compare the time elapsed in using Linux and Windows operating systems.

7.4 Future Prospects

Based on the findings of this study, several prospects for future studies are listed:

1. Apply the approach in channel measurements in the beyond 100 GHz or sixth generation (6G) frequency bands.
2. Comparison of IMT-2020 clustering results to different studies for validation.
3. Establish a General User Interface for automation of clustering using k -DAE algorithm.



REFERENCES

- [Africa et al., 2020] Africa, A., Trinidad, E., and Materum, L. (2020). Projection of wireless multipath clusters using multi-dimensional visualisation techniques. *Advances in Science, Technology and Engineering Systems Journal (ASTESJ)*, 5(6):1064–1070.
- [Aldossari and Chen, 2019] Aldossari, S. M. and Chen, K.-C. (2019). Machine learning for wireless communication channel modeling: An overview. *Wireless Personal Communications*, 106:41–70.
- [Alejandrino et al., 2022] Alejandrino, J., Trinidad, E., Concepcion, R., Sybingco, E., Palconit, M. G., Materum, L., and Dadios, E. (2022). Utilization of self-organizing maps for map depiction of multipath clusters. In *Intelligent Computing & Optimization: Proceedings of the 4th International Conference on Intelligent Computing and Optimization 2021 (ICO2021) 3*, pages 417–426. Springer.
- [Apicella et al., 2021] Apicella, A., Donnarumma, F., Isgrò, F., and Prevete, R. (2021). A survey on modern trainable activation functions. *Neural Networks*, 138:14–32.
- [Baur et al., 2022] Baur, M., Würth, M., Koller, M., Andrei, V.-C., and Utschick, W. (2022). Csi clustering with variational autoencoding. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5278–5282. IEEE.
- [Bhatkoti and Paul, 2016] Bhatkoti, P. and Paul, M. (2016). Early diagnosis of alzheimer’s disease: A multi-class deep learning framework with modified k-sparse autoencoder classification. In *2016 international conference on image and vision computing New Zealand (IVCNZ)*, pages 1–5. IEEE.
- [Blanza et al., 2023] Blanza, J., Trinidad, E., and Materum, L. (2023). Scedasticity descriptor of terrestrial wireless communications channels for multipath clustering datasets. *International Journal of Electrical and Computer Engineering (IJECE)*, 13(6):6547–6557.
- [Blanza and Materum, 2019a] Blanza, J. F. and Materum, L. (2019a). Grouping of cost 2100 indoor multipaths using simultaneous clustering and model selection. *International Journal of Emerging Trends in Engineering Research*, 7(8):197.
- [Blanza and Materum, 2019b] Blanza, J. F. and Materum, L. (2019b). Wireless propagation multipath clustering: On simultaneously solving the membership and the number of clusters. *International Journal of Advanced Trends in Computer Science and Engineering*, 8(5):1914.
- [Boban and Degli-Esposti, 2023] Boban, M. and Degli-Esposti, V. (2023). White paper on radio channel modeling and prediction to support future environment-aware wireless communication systems. *arXiv preprint arXiv:2309.17088*.
- [Cai et al., 2010] Cai, D., He, X., and Han, J. (2010). Locally consistent concept factorization for document clustering. *IEEE Transactions on Knowledge and Data Engineering*, 23(6):902–913.
- [Chacón and Rastrojo, 2023] Chacón, J. E. and Rastrojo, A. I. (2023). Minimum adjusted rand index for two clusterings of a given size. *Advances in Data Analysis and Classifi-*



- cation, 17(1):125–133.
- [Chen et al., 2017] Chen, J., Sathe, S., Aggarwal, C., and Turaga, D. (2017). Outlier detection with autoencoder ensembles. In *Proceedings of the 2017 SIAM international conference on data mining*, pages 90–98. SIAM.
- [Chen et al., 2018] Chen, Z., Yeo, C. K., Lee, B. S., and Lau, C. T. (2018). Autoencoder-based network anomaly detection. In *2018 Wireless telecommunications symposium (WTS)*, pages 1–5. IEEE.
- [Cui et al., 2022] Cui, Z., Guan, K., Oestges, C., Briso-Rodríguez, C., Ai, B., and Zhong, Z. (2022). Cluster-based characterization and modeling for uav air-to-ground time-varying channels. *IEEE Transactions on Vehicular Technology*, 71(7):6872–6883.
- [Czink et al., 2006] Czink, N., Cera, P., Salo, J., Bonek, E., Nuutinen, J.-P., and Ylitalo, J. (2006). A framework for automatic clustering of parametric mimo channel data including path powers. In *IEEE Vehicular Technology Conference*, pages 1–5. IEEE.
- [Du et al., 2014] Du, D., Zhang, J., Pan, C., and Zhang, C. (2014). Cluster characteristics of wideband 3d mimo channels in outdoor-to-indoor scenario at 3.5 ghz. In *2014 IEEE 79th Vehicular Technology Conference (VTC Spring)*, pages 1–6. IEEE.
- [Du et al., 2022] Du, M., Zhao, J., Sun, J., and Dong, Y. (2022). M3w: Multistep three-way clustering. *IEEE Transactions on Neural Networks and Learning Systems*.
- [Fahad et al., 2014] Fahad, A., Alshatri, N., Tari, Z., Alamri, A., Khalil, I., Zomaya, A. Y., Foufou, S., and Bouras, A. (2014). A survey of clustering algorithms for big data: Taxonomy and empirical analysis. *IEEE transactions on emerging topics in computing*, 2(3):267–279.
- [Faruque, 2017] Faruque, S. (2017). *Radio frequency modulation made easy*. Springer.
- [Frenzel, 2007] Frenzel, L. (2007). *Principles of electronic communication systems*. McGraw-Hill, Inc.
- [Geng et al., 2019] Geng, Y.-A., Liu, M., Li, Q., and He, R. (2019). *Introduction of machine learning*, pages 1–65.
- [Glorot et al., 2011] Glorot, X., Bordes, A., and Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323. JMLR Workshop and Conference Proceedings.
- [Gustafson et al., 2013] Gustafson, C., Haneda, K., Wyne, S., and Tufvesson, F. (2013). On mm-wave multipath clustering and channel modeling. *IEEE transactions on antennas and propagation*, 62(3):1445–1455.
- [He et al., 2018] He, R., Ai, B., Molisch, A. F., Stuber, G. L., Li, Q., Zhong, Z., and Yu, J. (2018). Clustering enabled wireless channel modeling using big data algorithms. *IEEE Communications Magazine*, 56(5):177–183.
- [He et al., 2017] He, R., Li, Q., Ai, B., Geng, Y. L.-A., Molisch, A. F., Kristem, V., Zhong, Z., and Yu, J. (2017). A kernel-power-density-based algorithm for channel multipath components clustering. *IEEE Transactions on Wireless Communications*, 16(11):7138–7151.



- [Hu et al., 2020] Hu, M., Ye, Y., He, R., Ai, B., Huang, C., and Zhong, Z. (2020). A novel power weighted multipath component clustering algorithm based on spectral clustering. In *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, pages 1–5. IEEE.
- [Huang et al., 2019] Huang, H., Savkin, A. V., Ding, M., and Huang, C. (2019). Mobile robots in wireless sensor networks: A survey on tasks. *Computer Networks*, 148:1–19.
- [Hubert and Arabie, 1985] Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of classification*, 2:193–218.
- [Ia, 2016] Ia, G. (2016). Deep learning/ian goodfellow, yoshua bengio and aaron courville.
- [Jaccard, 1912] Jaccard, P. (1912). The distribution of the flora in the alpine zone. 1. *New phytologist*, 11(2):37–50.
- [Jin and Han, 2010] Jin, X. and Han, J. (2010). K-Means Clustering. In Sammut, C. and Webb, G. I., editors, *Encyclopedia of Machine Learning*, pages 563–564. Springer US, Boston, MA.
- [Kajó et al., 2020] Kajó, M., Schultz, B., and Carle, G. (2020). Deep clustering of mobile network data with sparse autoencoders. In *NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium*, pages 1–6. IEEE.
- [Kannadasan et al., 2019] Kannadasan, K., Edla, D. R., and Kuppili, V. (2019). Type 2 diabetes data classification using stacked autoencoders in deep neural networks. *Clinical Epidemiology and Global Health*, 7(4):530–535.
- [Kingma and Ba, 2014] Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- [LeCun et al., 2015] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436–444.
- [Lee, 2020] Lee, J. (2020). Cluster-based millimeter-wave outdoor-to-indoor propagation characteristics based on 32 ghz measurement analysis. *IEEE Antennas and Wireless Propagation Letters*, 20(1):73–77.
- [Li et al., 2019] Li, J., Ai, B., He, R., Yang, M., Zhong, Z., and Hao, Y. (2019). A cluster-based channel model for massive mimo communications in indoor hotspot scenarios. *IEEE Transactions on Wireless Communications*, 18(8):3856–3870.
- [Lim et al., 2020] Lim, K.-L., Jiang, X., and Yi, C. (2020). Deep clustering with variational autoencoder. *IEEE Signal Processing Letters*, 27:231–235.
- [Liu et al., 2012] Liu, L., Oestges, C., Poutanen, J., Haneda, K., Vainikainen, P., Quitin, F., Tufvesson, F., and De Doncker, P. (2012). The cost 2100 mimo channel model. *IEEE Wireless Communications*, 19(6):92–99.
- [Liu et al., 2010] Liu, Y., Li, Z., Xiong, H., Gao, X., and Wu, J. (2010). Understanding of internal clustering validation measures. In *2010 IEEE international conference on data mining*, pages 911–916. IEEE.
- [Lu et al., 2021] Lu, Y., Liu, L., and Fu, L. (2021). Clustering-based nonstationary massive mimo channel modeling at 1.4725 ghz. *Applied Sciences*, 11(11):5083.
- [MacQueen et al., 1967] MacQueen, J. et al. (1967). Some methods for classification and



- analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA.
- [Materum and Teologo Jr, 2021] Materum, L. and Teologo Jr, A. (2021). An improved k-power means technique using minkowski distance metric and dimension weights for clustering wireless multipaths in indoor channel scenarios. *Journal of Information and Communication Technology*, 20(04):541–563.
- [Moayyed et al., 2019] Moayyed, M. T., Antonescu, B., and Basagni, S. (2019). Clustering algorithms and validation indices for mmwave radio multipath propagation. In *2019 Wireless Telecommunications Symposium (WTS)*, pages 1–7. IEEE.
- [Nair and Hinton, 2010] Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814.
- [Nayak et al., 2020] Nayak, R., Pati, U. C., and Das, S. K. (2020). Video anomaly detection using convolutional spatiotemporal autoencoder. In *2020 International Conference on Contemporary Computing and Applications (IC3A)*, pages 175–180. IEEE.
- [Opochinsky et al., 2020] Opochinsky, Y., Chazan, S. E., Gannot, S., and Goldberger, J. (2020). K-autoencoders deep clustering. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4037–4041. IEEE.
- [Ramírez-Arroyo et al., 2022] Ramírez-Arroyo, A., García, L., Alex-Amor, A., and Valenzuela-Valdés, J. F. (2022). Artificial intelligence and dimensionality reduction: Tools for approaching future communications. *IEEE Open Journal of the Communications Society*, 3:475–492.
- [Ruder, 2016] Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.
- [Saleem et al., 2023] Saleem, A., Zhang, X., Xu, Y., Albalawi, U., and Younes, O. (2023). A critical review on channel modeling: Implementations, challenges and applications. *electronics* 2023, 12, 2014.
- [Samimi et al., 2016] Samimi, M. K., Sun, S., and Rappaport, T. S. (2016). Mimo channel modeling and capacity analysis for 5g millimeter-wave wireless systems. In *2016 10th European Conference on Antennas and Propagation (EuCAP)*, pages 1–5. IEEE.
- [Song et al., 2013] Song, C., Liu, F., Huang, Y., Wang, L., and Tan, T. (2013). Auto-encoder based data clustering. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 18th Iberoamerican Congress, CIARP 2013, Havana, Cuba, November 20-23, 2013, Proceedings, Part I 18*, pages 117–124. Springer.
- [Szandała, 2021] Szandała, T. (2021). Review and comparison of commonly used activation functions for deep neural networks. *Bio-inspired neurocomputing*, pages 203–224.
- [Tan et al., 2018] Tan, P. Y., Wong, N. H., Tan, C. L., Jusuf, S. K., Chang, M. F., and Chiam, Z. Q. (2018). A method to partition the relative effects of evaporative cooling and shading on air temperature within vegetation canopy. *Journal of Urban Ecology*, 4(1):juy012.



- [Tran-Huy et al., 2024] Tran-Huy, H., Thi, T. N., Aslam, M., Tuan, H. N., and Nguyen, T. T.-L. (2024). A large-scale mimo antenna system for 5g iot applications. *Plos one*, 19(1):e0292383.
- [Trinidad and Materum, 2023] Trinidad, E. T. and Materum, L. (2023). High-accuracy visualization-based grouping of mimo multipath waves. *J. Commun.*, 18(1):68–75.
- [Vinh et al., 2009] Vinh, N. X., Epps, J., and Bailey, J. (2009). Information theoretic measures for clusterings comparison: is a correction for chance necessary? In *Proceedings of the 26th annual international conference on machine learning*, pages 1073–1080.
- [Wang et al., 2017] Wang, C., Pan, S., Long, G., Zhu, X., and Jiang, J. (2017). Mgae: Marginalized graph autoencoder for graph clustering. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 889–898.
- [Wang et al., 2018] Wang, C.-X., Bian, J., Sun, J., Zhang, W., and Zhang, M. (2018). A survey of 5g channel measurements and models. *IEEE Communications Surveys & Tutorials*, 20(4):3142–3168.
- [Wei et al., 2020] Wei, R., Garcia, C., El-Sayed, A., Peterson, V., and Mahmood, A. (2020). Variations in variational autoencoders-a comparative evaluation. *Ieee Access*, 8:153651–153670.
- [Wu et al., 2014] Wu, S., Wang, C.-X., Alwakeel, M. M., He, Y., et al. (2014). A non-stationary 3-d wideband twin-cluster model for 5g massive mimo channels. *IEEE journal on selected areas in communications*, 32(6):1207–1218.
- [Xue et al., 2022] Xue, G., Yang, M., Guo, Q., Yuan, S., et al. (2022). Performance of spectrum sharing in hybrid satellite terrestrial network with opportunistic relay selection. *Wireless Communications and Mobile Computing*, 2022.
- [Yang et al., 2019] Yang, L., Cheung, N.-M., Li, J., and Fang, J. (2019). Deep clustering by gaussian mixture variational autoencoders with graph embedding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6440–6449.
- [Yang and Liu, 1999] Yang, Y. and Liu, X. (1999). A re-examination of text categorization methods. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 42–49.
- [Yen and Chou, 2001] Yen, D. C. and Chou, D. C. (2001). Wireless communication: the next wave of internet technology. *Technology in Society*, 23(2):217–226.
- [Zhang et al., 2017a] Zhang, D., Sun, Y., Eriksson, B., and Balzano, L. (2017a). Deep unsupervised clustering using mixture of autoencoders. *arXiv preprint arXiv:1712.07788*.
- [Zhang et al., 2017b] Zhang, J., Zhang, Y., Yu, Y., Xu, R., Zheng, Q., and Zhang, P. (2017b). 3-d mimo: How much does it meet our expectations observed from channel measurements? *IEEE Journal on Selected Areas in Communications*, 35(8):1887–1903.
- [Zhao, 2019] Zhao, J. (2019). A survey of intelligent reflecting surfaces (irss): Towards 6g wireless communication networks. *arXiv preprint arXiv:1907.04789*.



DON HONORIO VENTURA STATE UNIVERSITY

Produced: June 3, 2024, 21:04



DON HONORIO VENTURA STATE UNIVERSITY

Appendix A

THESIS WEEKLY REPORT



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT
1st Semester SY 2023-2024

Date of Submission: October 11, 2023 Schedule: Tuesday 10:30-1:30 PM Report No: 1
Thesis Title: Autoencode-based Clusterization of Cannel Model Generated Wireless Multipath Components
Group No: 7 Thesis Adviser: Engr. Emmanuel T. Trinidad
For the week: October 2, 2023 to October 9, 2023 (Signature over Printed Name)

I. Accomplishments for the week:

Started writing the introduction section's first draft. The introduction provides a clear and concise overview of the research topic. Also, conducted a thorough analysis on the relevant literature review. Identified important theories, models, and earlier research that will serve as a guide in the prior studies.

II. Deviation from Planning (from the last report):

III. Corrective Action:

IV. Plans for the next week:

Revision of introduction section and continue composing the remaining sections of the chapter 1. Schedule regular meetings with the thesis adviser to seek advice, check progress, and discuss any problems that may arise.

V. Specific issues to be discussed:

Encountered challenges in obtaining suitable related literature relevant to the research topic. Generating preliminary phases of the statement of the problem and the specific objectives and deliverables.

VI. Contribution of each member:

Jimenez, Ishi Jemila - Background of the Study (Channel Model, Wave Propagation)

Leoncio, Timothy - Prior Studies

Macapagal, Faithlix Mark - Prior Studies

Maliwat, Jerome - Background of the Study (5G, MIMO, Clustering Multipath Components)

Manalang, Neera Mae - Background of the Study (Wireless network, WLAN, Multipath)

Olalia, Jhonly - Prior Studies

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

CS CamScanner

Fig. A.1 Weekly Report 1



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT
1st Semester SY 2023-2024

Date of Submission: October 18, 2023 Schedule: Tuesday 10:30-1:30 PM Report No: 2
Thesis Title: Autoencode-based Clusterization of Cannel Model Generated Wireless Multipath Components
Group No: 7
For the week: October 10, 2023 to October 17, 2023 Thesis Adviser: Engr. Emmanuel T. Trinidad
(Signature over Printed Name)

I. Accomplishments for the week:

The successful continuation and completion of Chapter 1 for our proposal paper was the highlight of this week's accomplishments. This important section, which gives a thorough description of the research study context, objectives and significance, providing the basis of the topic.

II. Deviation from Planning (from the last report):

Accomplished the planned task, which included writing the other parts of chapter 1 and revising the manuscript in response to the thesis adviser's suggestions.

III. Corrective Action:

Revising the chapter 1 to ensure that the concepts are presented in a clear and logical order, beginning with the introduction whereas in the background of the study, revising the paper by adding figures to properly discuss MIMO and multipath as well as adding brief explanations in channel modeling and autoencoders and its use in clustering. Then in the prior studies, revised by adding a frequency band and the type of clustering used and removing the similarity index.

IV. Plans for the next week:

Start drafting and outlining chapter 2 while making sure it is in line with the research objectives and follows logically from the previous sections. Schedule a meeting with the thesis adviser to discuss the progress of the research and seek his guidance and feedback on the draft.

V. Specific issues to be discussed:

Evaluate whether the problem statement and research objectives are clearly defined and aligned with the overall goal of the study. Mention the progress that has been made in the analysis and synthesis of the relevant literature. Provide the basis regarding including or excluding important theories, concepts, and articles relevant to the study.

VI. Contribution of each member:

Jimenez, Ishi Jemila- Estimated work schedule? Gantt chart and Budget
Leoncio, Timothy- Make tables of deliverables and provided significance of the study in Environmental Welfare
Macapagal, Faithlix Mark- Description of Methodology and revising the background of the study and prior studies.

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

Fig. A.2 Weekly Report 2



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT
1st Semester SY 2023-2024

Date of Submission: October 18, 2023 Schedule: Tuesday 10:30-1:30 PM Report No: 3
Thesis Title: k-Deep Autoencoder-Based Clustering of Channel Model Generated Multipath Components
Group No: 7
For the week: October 17, 2023 to October 24, 2023 Thesis Adviser: Engr. Emmanuel T. Trinidad
(Signature over Printed Name)

I. Accomplishments for the week:

Accomplished Flow Chart Midterm Exam in Methods of research. This Flow Chart shows the methodology of the thesis, the general and specific objectives and problems, variables used, locale/participants, data gathering tools, and statistical tools.

II. Deviation from Planning (from the last report):

Accomplished the planned task, which included writing the remaining parts of chapter 1.

III. Corrective Action:

Revision of the Problem Statement and Specific Problems.

IV. Plans for the next week:

Schedule a meeting with the thesis adviser to discuss the progress of the research and seek his guidance and feedback on the draft.

V. Specific issues to be discussed:

Revision of the Thesis title approved by the adviser.

VI. Contribution of each member:

Jimenez, Ishi Jemila- General and Specific Objectives
Leoncio, Timothy- Statement of the Problems
Macapagal, Faithlix Mark- Local/Participants and Data Gathering tools
Maliwat, Jerome- Data Gathering tools and Statistical tools
Manalang, Neera Mae- Research design
Olalia, Jhonly A.- Variables and Hypothesis

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

Fig. A.3 Weekly Report 3



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT 1st Semester SY 2023-2024

Date of Submission: October 31, 2023 Schedule: Tuesday 10:30-1:30 PM Report No: 4
Thesis Title: k-Deep Autoencoder-Based Clustering of Channel Model Generated Multipath Components
Group No: 7
For the week: October 24, 2023 to October 31, 2023 Thesis Adviser: Engr. Emmanuel T. Trinidad
(Signature over Printed Name)

I. Accomplishments for the week:

An online meeting was held with the adviser for the plan of the remaining days of thesis proposal paper writing. Also, reminders about the deadlines, accomplishments, and completion of the paper were discussed.

II. Deviation from Planning (from the last report):

Accomplished the planned task, which included writing the remaining parts of chapter 1.

III. Corrective Action:

Revision of the Problem Statement and Specific Problems.

IV. Plans for the next week:

Start writing chapter 2 while making sure it is in line with the research objectives and follows logically from the previous sections. Schedule a meeting with the thesis adviser to discuss the progress of the research and seek his guidance and feedback on the draft.

V. Specific issues to be discussed:

Revision of the Description of methodology to achieve the specific objectives of the study with the proper measures and techniques.

VI. Contribution of each member:

Jimenez, Ishi Jemila- Gantt chart and Budget
Leoncio, Timothy- Delimitations
Macapagal, Faithlix Mark- Description of Methodology.
Maliwat, Jerome- Description and Methodology.
Manalang, Neera Mae- Assumptions
Olalia, Jhonly A.- Scope

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

Fig. A.4 Weekly Report 4



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT
1st Semester SY 2023-2024

Date of Submission: November 6, 2023 Schedule: Tuesday 10:30-1:30 PM Report No: 5
Thesis Title: K-Deep Autoencoder-Based Clustering of Channel Model Generated Multipath Components
Group No: 7
For the week: October 31, 2023 to November 7, 2023 Thesis Adviser: Engr. Emmanuel T. Trinidad
(Signature over Printed Name)

I. Accomplishments for the week:

Conducted regular meeting with the thesis adviser to schedule the remaining days writing the thesis proposal paper. Reminders regarding the paper's completion, accomplishments, and deadlines were also included.

II. Deviation from Planning (from the last report):

Accomplished the assigned work which involved finishing writing the chapter 1 and 2 remaining sections.

III. Corrective Action:

Explaining the methods briefly and inserting the theoretical considerations in chapter 1.

IV. Plans for the next week:

Moving forward to focus on completing the remaining sections of Chapter 2, followed by a comprehensive review and revision process and will also be seeking feedback from the thesis adviser

V. Specific issues to be discussed:

Revision of the chapter 2 involving the literature review together with the existing work, lacking in the approaches and summary.

VI. Contribution of each member:

Jimenez, Ishi Jernila- Existing Work in Clustering using Autoencoders and Activation Functions used, Summary

Leoncio, Timothy- Existing Work in Clustering MPC, Lacking in the Approaches

Macapagal, Faithlix Mark- Description of Methodology, Overview

Maliwat, Jerome- Description and Methodology (methods block diagram)

Manalang, Neera Mae- Existing Work in Clustering MPC, Lacking in the Approaches, Progress Report #5

Olalia, Jhony A.- Existing Work in Clustering using Autoencoders and Activation Functions used, Summary

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

CS CamScanner

Fig. A.5 Weekly Report 5



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT 1st Semester SY 2023-2024

Date of Submission: November 14, 2023 Schedule: Tuesday 10:30-1:30 PM Report No: 6
Thesis Title: k-Deep Autoencoder-Based Clustering of Channel Model Generated Multipath Components
Group No: 7
For the week: November 8, 2023 to November 13, 2023 Thesis Adviser: Engr. Emmanuel T. Trinidad
(Signature over Printed Name)

I. Accomplishments for the week:

Completion of the thesis proposal paper, scheduled a regular meeting with the thesis adviser for revisions and polishing the chapter 1 and 2 before submitting the thesis proposal paper.

Accomplished the assigned task which involved finishing writing the chapter 2.

II. Corrective Action:

Revising specific question in the chapter 1, adding recent studies, repeat header and adding CVI used and Clustering Accuracy on the existing work for MPCS under the literature review.

III. Plans for the next week:

Proceeding on preparing for the thesis proposal defense.

IV. Specific issues to be discussed:

Revision and final touch for chapter 1 chapter 2 and start working on with paper works needs to be signed by the adviser and panel before the proposal defense.

V. Contribution of each member:

Jimenez, Ishi Jemila- Add CVI used and Clustering Accuracy on the Existing Work for MPCs

Leoncio, Timothy- Add CVI used and Clustering Accuracy on the Existing Work for MPCs

Macapagal, Faithlix Mark- Repeat header, Compiled paper for the thesis proposal paper

Maliwat, Jerome- Revise specific question, Existing works in Autoencoders

Manalang, Neera Mae- Add CVI used and Clustering Accuracy on the Existing Work for MPCs

Olalia, Jhony A.- Add CVI used and Clustering Accuracy on the Existing Work for MPCs

Members: (signature over printed name)

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

CS CamScanner

Fig. A.6 Weekly Report 6



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT
1st Semester SY 2023-2024

Date of Submission: November 21, 2023 Schedule: Tuesday 10:30-1:30 PM Report No: 7
Thesis Title: k-Deep Autoencoder-Based Clustering of Channel Model Generated Multipath Components
Group No: 7
For the week: November 14, 2023 to November 20, 2023 Thesis Adviser: Engr. Emmanuel T. Trinidad
(Signature over Printed Name)

I. Accomplishments for the week:

Completion of the required documents such as the appendices and also the thesis proposal paper both soft copy and hard copy.

II. Corrective Action:

Revising some minimal terms and adding the list of table, figures and appendices together with the glossary on the paper.

III. Plans for the next week:

Proceeding on preparing for the oral presentation of the thesis proposal defense.

IV. Specific issues to be discussed:

Start preparing for the thesis proposal defense by considering possible questions or inquires from the panel regarding the research topic.

V. Contribution of each member:

Jimenez, Ishi Jemila- List of Appendices, Abbreviations and Acronyms
Leoncio, Timothy- List of figures and tables
Macapagal, Faithlix Mark- Table of Contents
Maliwat, Jerome- Notations
Manalang, Neera Mae- Glossary, Progress Report #7
Olalia, Jhonly A.- Glossary

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

Fig. A.7 Weekly Report 7



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT 1st Semester SY 2023-2024

Date of Submission: March 27, 2024 Schedule: Monday 11:30-1:00 PM, Friday 7:30-9:00 AM Report No: 20

Thesis Title: k-Deep Autoencoder-Based Clustering of Channel Generated Multipath Components

Group No: 7

For the week: March 20, 2024 to March 28, 2024 Thesis Adviser: Engr. Emmanuel T. Trinidad
(Signature over Printed Name)

I. Accomplishments for the week:

Completed the data gathering and conducted a face-to-face interview with Engr. Arlon Calma regarding on discussing his insights about our research topic.

II. Deviation from Planning (from the last report):

Conducted a regular weekly meeting together with the thesis adviser regarding the discussion of planning to work out on the data analysis as well as starting the chapters 6 and 7 of the thesis paper.

III. Corrective Action:

Finished running all the data sets even the O2I data sets and compiled all the results from the C2CM and IMT2020 in an excel sheet.

IV. Plans for the next week:

Start working out on the data analysis with the use of MATLAB and also, starting the chapters 6 and 7 of the thesis paper.

V. Specific issues to be discussed:

Observe the general and specific objectives of the thesis paper and identify the parts that needed to be discussed.

VI. Contribution of each member:

Jimenez, Ishi Jemila- Finished running the remaining data sets from the code on a device.

Leoncio, Timothy- Finished running the remaining data sets from the code on a device.

Macapagal, Faithix Mark- Conducted and provide the questionnaires on the interview.

Maliwat, Jerome- Conducted and provide the questionnaires on the interview.

Manalang, Neera Mae- Provide the Progress Report #20.

Olalia, Jhony- Finished running the remaining data sets from the code on a device.

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

CamScanner

Fig. A.8 Weekly Report 8



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT 1st Semester SY 2023-2024

Date of Submission: December 10, 2023 Schedule: Tuesday 10:30-1:30 PM Report No: 10
Thesis Title: k-Deep Autoencoder-Based Clustering of Channel Model Generated Multipath Components
Group No: 7
For the week: December 5, 2023 to December 11, 2023 Thesis Adviser: Engr. Emmanuel T. Trinidad
(Signature over Printed Name)

I. Accomplishments for the week:

Conducted a virtual meeting with the thesis adviser together with Engr. Materum to discuss the revisions of the thesis paper.

II. Corrective Action:

Discussed the revision of the paper based on the comments of the panels and how it will be addressed in the paper.

III. Plans for the next week:

Proceeding on start revising the thesis paper based on the notes from the thesis adviser in relate to the comments of the panels.

IV. Specific issues to be discussed:

Revision of specific objectives where it has to be concise and measurable, percentage improvement on the Jaccard Index, conduct an interview based on the given questions.

V. Contribution of each member:

Jimenez, Ishi Jemila- Identified the corresponding part/page of the paper to revise/edit based on the panels' comments.
Leoncio, Timothy- Taken notes of the comments of panel #3.

Macapagal, Faithlix Mark- Generate steps on how the comments will be addressed.

Maliwat, Jerome- Summarized and compiled the panel's comments.

Manalang, Neera Mae- Taken notes of the comments of panel #2., Progress Report #10

Olalia, Jhony A.- Taken notes of the comments of panel #1.

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

CS CamScanner

Fig. A.9 Weekly Report 10



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT 1st Semester SY 2023-2024

Date of Submission: January 24, 2024 Schedule: _____ Report No: 11
Thesis Title: Autoencode-based Clusterization of Cannel Model Generated Wireless Multipath Components
Group No: 7
For the week: January 17, 2024 to January 19, 2024 Thesis Adviser: Engr. Emmanuel T. Trinidad
(Signature over Printed Name)

I. Accomplishments for the week:

Conducted a meeting regarding on utilizing the given code in Python and attempted to run the data sets from the code to check its function.

II. Deviation from Planning (from the last report):

Scheduled an online meeting together with the thesis adviser regarding on utilizing the code.

III. Corrective Action:

Comparing the open-source libraries used in the Python from the previous studies (Lacsina, LJ., et.al) by installing the libraries such as the Keras, TensorFlow, TensorFlow-estimator, NumPy, scikit-learn, SciPy, to identify whether the code will execute.

IV. Plans for the next week:

Start learning the code thoroughly and observe and analyze the errors when simulating it. Also, start working on the Chapters 3 and 4 on the paper.

V. Specific issues to be discussed:

Evaluate the encountered error from the code and analyze why it fails to operate. Use trial and error by installing other software programs such as Spyder and Anaconda.

VI. Contribution of each member:

Jimenez, Ishi Jemila- Identify the error from the code while running it from the device.

Leoncio, Timothy- Utilize the code from the device and attempted to run the data sets.

Macapagal, Faithlix Mark- Use trial and error to determine the code will execute from other software program.

Maliwat, Jerome- Utilize the code from the device and attempted to run the data sets.

Manalang, Neera Mae- Completed the Progress Report #11.

Olalia, Jhonly- Utilize the code from the device and attempted to run the data sets.

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

Fig. A.10 Weekly Report 11



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT 1st Semester SY 2023-2024

Date of Submission: January 31, 2024 Schedule: _____ Report No: 12
Thesis Title: Autoencode-based Clusterization of Cannel Model Generated Wireless Multipath Components
Group No: 7
For the week: January 23, 2024 to January 30, 2024 Thesis Adviser: Engr. Emmanuel T. Trinidad
(Signature over Printed Name)

I. Accomplishments for the week:

Scheduled an online meeting together with the thesis adviser regarding on the concerns on working out the given code on other software program.

II. Deviation from Planning (from the last report):

Utilized trial and error which is installing other software programs that could operate the given code.

III. Corrective Action:

Observing the low results of the Jaccard Index on the sigmoid indoor scenarios when incorporating the k-DAE autoencoder.

IV. Plans for the next week:

Start working out on the Chapters 3, 4 and 5 on the paper specifically on the Methodology that needs to be improved.

V. Specific issues to be discussed:

Initiate the process of learning to run the code on exploring different parameters of autoencoders to improve the Jaccard Index results.

VI. Contribution of each member:

Jimenez, Ishi Jemila- Identify the error from the code while running it from the device.

Leoncio, Timothy- Utilized the code from the device and attempted to run the data sets.

Macapagal, Faithix Mark- Incorporate the code and evaluate the low results of Jaccard Index

Maliwat, Jerome- Utilized the code from the device and attempted to run the data sets.

Manalang, Neera Mae- Completed the Progress Report #12.

Olalia, Jhonly- Utilize the code from the device and attempted to run the data sets.

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

Fig. A.11 Weekly Report 12



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT
1st Semester SY 2023-2024

Date of Submission: February 7, 2024 Schedule: Monday 11:30-1:00 PM, Friday 7:30-9:00 AM Report No: 13

Thesis Title: Autoencode-based Clusterization of Cannel Model Generated Wireless Multipath Components

Group No: 7

For the week: January 29, 2024 to February 2, 2024 Thesis Adviser: Engr. Emmanuel T. Trinidad
(Signature over Printed Name)

I. Accomplishments for the week:

Utilized and set up the code by installing required python versions and its libraries to the members' laptop devices. Also, conducted the initial testing of the program.

II. Deviation from Planning (from the last report):

Started the process of operating the code and testing with different autoencoder parameters to improve the Jaccard Index results.

III. Corrective Action:

Evaluating the low performance of accuracy of the Jaccard Index by exploring different parameters to improve it.

IV. Plans for the next week:

Begin working on the Chapters 3, 4 and 5 on the paper as soon as the code effectively generates the intended results.

V. Specific issues to be discussed:

Improving the accuracy of the Jaccard Index when integrating k-DAE autoencoder and activation function.

VI. Contribution of each member:

Jimenez, Ishi Jemila- Set-up the code and operate it from the device.

Leoncio, Timothy- Tested running the data sets using the code from the device.

Macapagal, Faithlix Mark- Configuring the code on improving the accuracy of the results.

Maliwat, Jerome- Tested running the data sets using the code from the device.

Manalang, Neera Mae- Completed the Progress Report #13.

Olalia, Jhony- Set-up the code and operate it from the device.

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A - Thesis Weekly Progress Report
Rev. 09-2023

CS CamScanner

Fig. A.12 Weekly Report 13



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT 1st Semester SY 2023-2024

Date of Submission: February 14, 2024 Schedule: Monday 11:30-1:00 PM, Friday 7:30-9:00 AM Report No: 14

Thesis Title: Autoencode-based Clusterization of Cannel Model Generated Wireless Multipath Components

Group No: 7

For the week: February 8, 2024 to February 14, 2024 Thesis Adviser: Engr. Emmanuel T. Trinidad
(Signature over Printed Name)

I. Accomplishments for the week:

Finalize the modified code and apply the Stochastic Gradient Descent as an optimizer to increase the percentage of accuracy of the Jaccard Index results.

II. Deviation from Planning (from the last report):

Conducted a weekly meeting together with the thesis adviser regarding on improving the accuracy of Jaccard Index results.

III. Corrective Action:

Identify the proper optimizer to increase the percentage of accuracy of the Jaccard Index results.

IV. Plans for the next week:

Start working out on the Chapters 3, 4 and 5 on the paper and begin gathering data sets on the modified code.

V. Specific issues to be discussed:

Incorporate the gathered data sets on the modified code and operate it from the device.

VI. Contribution of each member:

Jimenez, Ishi Jemila- Set-up the modified code and operate it from the device.

Leoncio, Timothy- Determine an optimizer that would increase the percentage of accuracy results.

Macapagal, Faithlix Mark- Improved the percentage of accuracy results by applying Stochastic Gradient Descent as the optimizer.

Maliwat, Jerome- Determine an optimizer that would increase the percentage of accuracy results.

Manalang, Neera Mae- Completed the Progress Report #14.

Olalia, Jhony- Set-up the modified code and operate it from the device.

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

CS CamScanner

Fig. A.13 Weekly Report 14



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT
1st Semester SY 2023-2024

Date of Submission: February 21, 2024 Schedule: Monday 11:30-1:00 PM, Friday 7:30-9:00 AM Report No: 15

Thesis Title: Autoencode-based Clusterization of Cannel Model Generated Wireless Multipath Components

Group No: 7

For the week: February 15, 2024 to February 22, 2024

Thesis Adviser: Engr. Emmanuel T. Trinidad

(Signature over Printed Name)

I. Accomplishments for the week:

Set up the finalize modified code on the device together with operating the gathered data sets. The output data values from the four configurations remain on both initial and final clustering when the modified code operates on the device.

II. Deviation from Planning (from the last report):

Scheduled a weekly meeting together with the thesis adviser regarding on incorporating the gathered data sets on the modified code and operating it from the device.

III. Corrective Action:

Observe the output data values from the four configurations that remain on both initial and final clustering.

IV. Plans for the next week:

Continue working on the Chapters 3, 4 and 5 on the paper and begin revising together with the thesis adviser before the submission date.

V. Specific issues to be discussed:

Evaluate the data results from the initial and final clustering and start working out the on the incomplete parts of the thesis paper.

VI. Contribution of each member:

Jimenez, Ishi Jemila- Incorporate the gathered data sets and operate the modified code from the device.

Leoncio, Timothy- Set up the modified code and operate it from the device.

Macapagal, Faithlix Mark- Examine the output data values from the four configurations that remain from both initial and final clustering.

Maliwat, Jerome- Incorporate the gathered data sets and operate the modified code from the device.

Manalang, Neera Mae- Completed the Progress Report #15.

Olalia, Jhony- Incorporate the gathered data sets and operate the modified code from the device.

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

CamScanner

Fig. A.14 Weekly Report 15



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT 1st Semester SY 2023-2024

Date of Submission: February 28, 2024 Schedule: Monday 11:30-1:00 PM, Friday 7:30-9:00 AM Report No: 16

Thesis Title: Autoencode-based Clusterization of Cannel Model Generated Wireless Multipath Components

Group No: 7

For the week: February 22, 2024 to February 29, 2024

Thesis Adviser: Engr. Emmanuel T. Trinidad

[Signature over Printed Name]

I. Accomplishments for the week:

Completed the drafts of the thesis paper's chapters 3, 4, and 5 and emailed them to the thesis adviser for revision before the deadline.

II. Deviation from Planning (from the last report):

Conducted a regular meeting together with the thesis adviser regarding the completion of chapters 3, 4 and 5 of the thesis paper.

III. Corrective Action:

Revising some parts of the paper after checking the drafts by the thesis adviser.

IV. Plans for the next week:

Proceed working with the data sets by running them on the modified code.

V. Specific issues to be discussed:

Revised the parts that needs revision in accordance with the thesis adviser's feedback on thesis paper.

VI. Contribution of each member:

Jimenez, Ishi Jemila- Incorporate the Chapter 4 which is the Design Considerations.

Leoncio, Timothy- Provide the Chapter 3 which is the Theoretical Considerations.

Macapagal, Faithlix Mark- Construct the Chapter 5 which is the Methodology and also provide the Conceptual Framework.

Maliwat, Jerome- Construct the Chapter 5 which is the Methodology and also provide the Conceptual Framework.

Manalang, Neera Mae- Provide the Chapter 3 which is the Theoretical Considerations.

Olalia, Jhony- Incorporate the Chapter 4 which is the Design Considerations.

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

CamScanner

Fig. A.15 Weekly Report 16



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT
1st Semester SY 2023-2024

Date of Submission: March 6, 2024 Schedule: Monday 11:30-1:00 PM, Friday 7:30-9:00 AM Report No: 17

Thesis Title: k-Deep Autoencoder-Based Clustering of Channel Generated Multipath Components

Group No: 7

For the week: February 29, 2024 to March 7, 2024 Thesis Adviser: Engr. Emmanuel T. Trinidad
(Signature over Printed Name)

I. Accomplishments for the week:

Submitted the chapters 3, 4 and 5 on time and received the results of the similarity reports from the three chapters.

II. Deviation from Planning (from the last report):

Scheduled a regular meeting together with the thesis adviser regarding revising some parts of chapters 3, 4 and 5 of the thesis paper.

III. Corrective Action:

Revising some parts of each chapter in accordance with the results of the similarity report.

IV. Plans for the next week:

Proceed on revising the chapters 3, 4 and 5 until the desired overall similarity percentage is achieved and continue working with the data sets by running them on the modified code.

V. Specific issues to be discussed:

Revise and incorporate some adjustments on some parts of the chapters 3, 4 and 5 in accordance with the results of the similarity report.

VI. Contribution of each member:

Jimenez, Ishi Jemila- Revise the Chapter 4 which is the Design Considerations

Leoncio, Timothy- Revise the Chapter 3 which is the Theoretical Considerations.

Macapagal, Faithlix Mark- Revise the Chapter 4 and 5 which is the Design Considerations and the Methodology.

Maliwat, Jerome- Revise the Chapter 4 and 5 which is the Design Considerations and the Methodology.

Manalang, Neera Mae- Revise the Chapter 3 which is the Theoretical Considerations and provide the Progress Report #17

Olaia, Jhony- Revise the Chapter 4 which is the Design Considerations.

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

CS CamScanner

Fig. A.16 Weekly Report 17



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT 1st Semester SY 2023-2024

Date of Submission: March 13, 2024 Schedule: Monday 11:30-1:00 PM, Friday 7:30-9:00 AM Report No: 18

Thesis Title: k-Deep Autoencoder-Based Clustering of Channel Generated Multipath Components

Group No: 7

For the week: March 7, 2024 to March 14, 2024 Thesis Adviser: Engr. Emmanuel T. Trinidad
(Signature over Printed Name)

I. Accomplishments for the week:

Submitted the certificate of percentage of completion and also installed the driver called CUDA in order to set up the GPU from the device to facilitate on running the data sets from the code. Furthermore, the data gathering is almost complete.

II. Deviation from Planning (from the last report):

Conducted a regular meeting together with the thesis adviser regarding the installation of CUDA and implementation of GPU to make it easier to run the data sets from the code.

III. Corrective Action:

Installation of CUDA driver to set up the GPU from the device to reduce the time it takes on running the data sets from the code.

IV. Plans for the next week:

Proceed on revising some chapters from the similarity report results until the desired overall percentage is achieved and continue to run the remaining data sets from the code.

V. Specific issues to be discussed:

Evaluate the encountered error on some sheet numbers while running the data sets from the code.

VI. Contribution of each member:

Jimenez, Ishi Jemila- Set up the GPU from the device to facilitate on running the data sets from the code.

Leoncio, Timothy- Set up the GPU from the device to facilitate on running the data sets from the code.

Macapagal, Faithlix Mark- Set up the GPU from the device to facilitate on running the data sets from the code.

Maliwat, Jerome- Set up the GPU from the device to facilitate on running the data sets from the code.

Manalang, Neera Mae- Provide the Progress Report #18

Olalia, Jhonly- Set up the GPU from the device to facilitate on running the data sets from the code.

*Attached copies of the data, tables, pictures, figures, and other document supporting the report

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

CamScanner

Fig. A.17 Weekly Report 18



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING ARCHITECTURE
Department Of Electronics Engineering

ECE DES 1 WEEKLY PROGRESS REPORT 1st Semester SY 2023-2024

Date of Submission: April 3, 2024 Schedule: Monday 11:30-1:00 PM, Friday 7:30-9:00 AM Report No: 21

Thesis Title: k-Deep Autoencoder-Based Clustering of Channel Generated Multipath Components

Group No: 7

For the week: March 27, 2024 to April 4, 2024 Thesis Adviser: Engr. Emmanuel T. Trinidad
(Signature over Printed Name)

I. Accomplishments for the week:

Completed of the data analysis with the use of MATLAB and the data results from the data gathering had been incorporated in the Chapter 6 of the thesis paper.

II. Deviation from Planning (from the last report):

Scheduled a regular weekly meeting together with the thesis adviser regarding the revision of Chapter 6 which is the Results and Discussion.

III. Corrective Action:

Revising some parts in accordance with the thesis adviser's feedback on the Results and Discussion which is the Chapters 6 before the submission date.

IV. Plans for the next week:

Start preparing for the oral presentation in the final defense and conduct a mock defense that serves as a practice.

V. Specific issues to be discussed:

Determine the recommendations by the panel to be included in the Chapter 7.

VI. Contribution of each member:

Jimenez, Ishi Jemila- Compiled all the data results in an excel sheet.

Leoncio, Timothy- Compiled all the data results in an excel sheet.

Macapagal, Faithlix Mark- Conducted the data analysis. Also, provide and revised the Chapter 6.

Maliwat, Jerome- Conducted the data analysis. Also, provide and revised the Chapter 6.

Manalang, Neera Mae- Revised the Chapter 3 and 4 and provide the Progress Report #21.

Olalia, Jhonly- Compiled all the data results in an excel sheet.

**Attached copies of the data, tables, pictures, figures, and other document supporting the report*

Appendix A – Thesis Weekly Progress Report
Rev. 09-2023

Fig. A.18 Weekly Report 21



DON HONORIO VENTURA STATE UNIVERSITY

Appendix B

ECE ADVISER ACCEPTANCE FORM



DON HONORIO VENTURA STATE UNIVERSITY



Don Honorio Ventura State University
College of Engineering & Architecture
Department of Electronics Engineering

September 21, 2023

To : Anthony S. Tolentino, MS-ECE
Design Capstone 1 Coordinator
Department of Electronics Engineering
[Signature] *09/28/2023*

Fr : Engr. Emmanuel T. Trinidad (to be signed)
EcE Thesis Adviser

This is to certify that I have accepted the group composed of

Jimenez, Ishi Jemila E. *[Signature]*
Leoncio, Timothy D. *T. Leoncio*
Macapagal, Faithlix Mark M. *[Signature]*
Maliwat, Jerome R. *[Signature]*
Manalang, Neera Mae C. *[Signature]*
Olalia, Jhonly A. *[Signature]*

section 4B whose topic is Autoencoder-based Clusterization of channel Model Generated wireless multipath components

to be their permanent thesis adviser for Design Capstone 1 and Design Capstone 2 for A.Y 2023-2024.

Appendix B - EcE Adviser's Form
Revised 2023

Fig. B.1 EcE Adviser Acceptance Form



Appendix C

MEMORANDUM OF UNDERSTANDING BETWEEN THE ECE THESIS ADVISEE AND THESIS ADVISER



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
 DON HONORIO VENTURA STATE UNIVERSITY
 Bacolor, Pampanga

COLLEGE OF ENGINEERING AND ARCHITECTURE
 ELECTRONICS ENGINEERING DEPARTMENT
 SY 2023-2024

**MEMORANDUM OF UNDERSTANDING BETWEEN
 THE ECE THESIS ADVISEE AND THESIS ADVISER**

This Memorandum of Understanding (this "Memorandum of Understanding" or this "MOU") is made and entered into as of September 28, 2024 by Jerome R. Maliwat, Ishi Jemila E. Jimenez, Timothy D. Leoncio, Faithlix Mark M. Macapagal, Neera Mae C. Manalang, Jhonly A. Olalia (names of the thesis group leader and members) and between Engr. Emmanuel T. Trinidad (name/s of thesis adviser/s) of EeE Department, DHVSU. This Memorandum of Understanding sets forth the understanding of the parties related to the function of the Thesis Adviser and the function of the Thesis advisee, and this Memorandum of Understanding shall expire upon the earlier of June 30, 2024; following the Effective Date, unless it is extended by written modification, or a written successor agreement pertaining to the subject matter of this MOU.

WHEREAS, in connection with the establishment of this thesis functions the parties wish to set forth their current understanding and certain agreements made to date with respect to such function.

The Thesis Adviser shall fulfill the following functions:

- Advising the student in the preparation of the thesis final project and document.
- Guiding and monitoring students' undergraduate research.
- Endorsing the research work for the defense.
- Checking the format of the manuscript.
- Providing the General Editing of the thesis.
- Providing guidance in data analysis.
- Checking the organization and the contents of the work.
- Providing ample time to his/her advisee in relation to thesis work.
- Orienting the advisee on what might/will transpire in the defense session, and
- Present for the duration of thesis advisee defense, giving cues or signals to the group when necessary, or clarify questions for the group but is not allowed to answer nor object for the candidates during the defense.

The Thesis Advisee shall fulfill the following functions:

- Inform the adviser, at least twice a week of all the important matters for the conduct of research.
- Coordinate with the adviser regarding the changes, suggestions, and recommendations, and
- Meet the adviser through different online platforms for thesis consultations

IN WITNESS WHEREOF, by their signatures below, the parties have caused this Memorandum of Understanding to be executed and effective as of the Effective Date.

Engr. Emmanuel T. Trinidad
 Thesis Adviser

Appendix C –
 Design Capstone Project Thesis Advisee-Adviser Rev 09-2023

Thesis Member
 (Signature over printed name)
 1. JIMENEZ, Ishi Jemila E.
 2. LEONCIO, Timothy D.
 3. MACAPAGAL, Faithlix Mark M.
 4. MALIWAT, Jerome R.
 5. MANALANG, Neera Mae C.
 6. OLALIA, Jhonly A.

CamScanner

Fig. C.1 MEMORANDUM OF UNDERSTANDING BETWEEN THE ECE THESIS ADVISEE AND THESIS ADVISER



DON HONORIO VENTURA STATE UNIVERSITY

Appendix D

CERTIFICATE OF PERCENTAGE COMPLETION



CERTIFICATE OF PERCENTAGE OF COMPLETION

This is to CERTIFY that Group Number 7 who are composed of BS Electronics Engineering students from Don Honorio Ventura State University, have partially completed their project for at least 80%. (Indicate Percentage).

REMARKS: (Please indicate in writing your comments)

Complete initial results of remaining datasets.

Enable GPU processing..

Perform statistical analysis and provide graphs.

Date of Checking/ Inspection: March 7, 2024

Place of Checking/Inspection: Fabrication Laboratory 2

Signed by: Emmanuel T. Trinidad 07/07/24 (Thesis Adviser)

Fig. D.1 Certificate of Percentage Completion



DON HONORIO VENTURA STATE UNIVERSITY

Appendix E

ANTI-PLAGIARISM FORM



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING AND ARCHITECTURE
ELECTRONICS ENGINEERING DEPARTMENT
2nd Semester SY 2023-2024

ANTI-PLAGIARISM WAIVER

We,

JIMENEZ, Ishi Jemila E.
LEONCIO, Timothy D.
MACAPAGAL, Faithlix Mark M.
MALIWAT, Jerome R.
MANALANG, Neera Mae C.
OLALIA, Jhonly A.

with group number 7, with a thesis entitled **k-Deep Autoencoder-Based Clustering of Channel Model Generated Multipath Components**, in partial fulfillment of the requirements for the degree in Bachelor of Science in Electronics Engineering agree to comply with all the following conditions:

- 1.) We shall refrain from un-credited verbatim copying of a full paper or the verbatim copying of a major portion of an original document.
- 2.) We shall refrain from un-credited verbatim copying of individual elements (e.g., paragraphs, sentences, figures).
- 3.) We shall refrain from un-credited improper paraphrasing of pages or paragraphs.
- 4.) We shall refrain from credited verbatim copying of a major portion of a paper without clear delineation (e.g., quotes or indents).
- 5.) We shall commit ourselves to the highest ethical and professional conduct.
- 6.) We shall agree to be honest and realistic in stating claims and/or estimates based on available data.

Date: May 21, 2024

Appendix J – Anti-Plagiarism Waiver

CamScanner

Fig. E.1 Anti-Plagiarism Form



Appendix F

ORAL DEFENSE RECOMMENDATION SHEET



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA TECHNOLOGICAL UNIVERSITY
Bacolor, Pampanga
COLLEGE OF ENGINEERING AND ARCHITECTURE
ELECTRONICS ENGINEERING DEPARTMENT
SY 2023-2024

THESIS PROPOSAL APPROVAL SHEET

This thesis entitled:

"k-Deep Autoencoder-Based Clustering of Channel Generated Multipath Components"

prepared and submitted by JIMENEZ, Ishi Jemila E., LEONCIO, Timothy D., MACAPAGAL, Faithlix Mark M., MALIWAT, Jerome R., MANALANG, Neera Mae C., OLALIA, Jhony A. (members), in partial fulfillment of the requirements for the degree of Bachelor of Science in Electronics and Communications Engineering has been examined and is recommended for acceptance and approval for THESIS PROPOSAL DEFENSE.

Engr. Emmanuel T. Trinidad
Adviser

Approved by the Committee on Oral Examination (Thesis Proposal) with a defense result of _____ on November 29, 2023 (oral defense date).

Engr. Kirson B. Serrano

Chair

Engr. Mary Anne M. Sahagun
Member

Engr. Anthony S. Tolentino, MS-ECE
Capstone Project 1 Coordinator

Engr. Enmar T. Tuazon
Member

Engr. Mary Anne M. Sahagun, PECE, MEP-EE
Chairperson, ECE Department
12.20.2023

Appendix L –
Design Capstone Project : Thesis Proposal Approval Sheet Rev 09-2023

CamScanner

Fig. F.1 ORAL DEFENSE RECOMMENDATION SHEET



DON HONORIO VENTURA STATE UNIVERSITY

Appendix G

THESIS PROPOSAL APPROVAL SHEET



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA TECHNOLOGICAL UNIVERSITY
Bacolor, Pampanga
COLLEGE OF ENGINEERING AND ARCHITECTURE
ELECTRONICS ENGINEERING DEPARTMENT
SY 2023-2024

THESIS PROPOSAL APPROVAL SHEET

This thesis entitled:

"k-Deep Autoencoder-Based Clustering of Channel Generated Multipath Components"

prepared and submitted by JIMENEZ, Ishi Jemila E., LEONCIO, Timothy D., MACAPAGAL, Faithlix Mark M., MALIWAT, Jerome R., MANALANG, Neera Mae C., OLALIA, Jhony A. (members), in partial fulfillment of the requirements for the degree of Bachelor of Science in Electronics and Communications Engineering has been examined and is recommended for acceptance and approval for THESIS PROPOSAL DEFENSE.

Engr. Emmanuel T. Trinidad
Adviser

Approved by the Committee on Oral Examination (Thesis Proposal) with a defense result of _____ on November 29, 2023 (oral defense date).

Engr. Kirson B. Serrano

Chair

Engr. Mary Anne M. Sahagun
Member

Engr. Anthony S. Tolentino, MS-ECE
ECE No. 0050341
Capstone Project 1 Coordinator

Engr. Enmar T. Tuazon
Member

Engr. Mary Anne M. Sahagun, PECE, MEP-EE
Chairperson, ECE Department
12-20-2023

Appendix L –
Design Capstone Project : Thesis Proposal Approval Sheet Rev 09-2023

CamScanner

Fig. G.1 Thesis Proposal Approval Sheet



Appendix H

FINAL DEFENSE RECOMMENDATION SHEET



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA STATE UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING AND ARCHITECTURE
ELECTRONICS ENGINEERING DEPARTMENT
SY 2023-2024

FINAL DEFENSE RECOMMENDATION SHEET

This thesis, entitled k-Deep Autoencoder-Based Clustering of Channel Generated Multipath Components, prepared and submitted by Group 7 (Thesis Group No.), composed of:

JIMENEZ, Ishi Jemila E.

LEONCIO, Timothy D.

MACAPAGAL, Faithlix Mark M.

MALIWAT, Jerome R.

MANALANG, Neera Mae C.

OLALIA, Jhonly A.

in partial fulfillment of the requirements for the degree of Bachelor of Science in Electronics Engineering (BSECE) has been examined and is recommended for acceptance and approval for FINAL THESIS DEFENSE.


Engr. Emmanuel T. Trinidad
Adviser

Appendix M
Design Capstone Project : Oral Defense Recommendation Sheet Rev 03-2022

CS CamScanner

Fig. H.1 Final Defense Recommendation Sheet



Appendix I GANTT-CHART

**GANTT CHART**

THESIS TITLE: AUTO ENCODE-BASED CLUSTERIZATION OF CAHNNEL MODEL GENERATED WIRELESS MULTIPATH COMPONENTS

GROUP MEMBERS: JIMENEZ, ISHI JEMILA
LEONCIO, TIMOTHY
MACAPAGAL, FAITHLI MARK
MALIWAT, JEROME
MANALANG, NEERA MAE
OLAILIA, JHONLY A.

Week #	Target Activity	Expected Deliverables (must reflect in your weekly progress reports)
1	START OF DRAFTING THE CHAPTER 1 FOR PROPOSAL PAPER.	BACKGROUND OF THE STUDY
2	CONTINUATION AND COMPLETION OF THE CHAPTER 1 FOR PROPOSAL PAPER	DESCRIPTION OF METHODOLOGY
3	MIDTERM EXAM	N/A
4	START OF WRITING THE DRAFT OF CHAPTER 2	LITERATURE REVIEW
5	CONTINUATION AND COMPLETION OF CHAPTER 2	LITERATURE REVIEW
6	START OF REVISION OF CHAPTERS 1 AND 2	FINAL REVISED BACKGROUND OF THE STUDY AND LITERATURE REVIEW
7	DEADLINE OF SUBMISSION OF PROPOSAL PAPER	PRINTED COPY OF FINAL PROPOSAL PAPER
8	PREPARATION FOR THE PROPOSAL DEFENSE	POWERPOINT PRESENTATION
9	PROPOSAL DEFENSE	PROPOSED THESIS PROJECT
10	REVISIONS OF CHAPTERS 1 AND 2	REVISED PAPER
11	FINAL THESIS TITLE AND START OF DRAFTING REMAINING PARTS OF RESEARCH	PRINTED REVISED PAPER
12	FINAL EXAMS	N/A

Noted by

Engr. Emmanuel Trinidad
Adviser

Appendix P Gantt Chart Rev 09-2023

Fig. I.1 Gantt Chart



Appendix J

MEMORANDUM OF AGREEMENT

BETWEEN THESIS MEMBERS AND THESIS

PANEL



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA TECHNOLOGICAL UNIVERSITY
Bacolor, Pampanga
COLLEGE OF ENGINEERING AND ARCHITECTURE
ELECTRONICS ENGINEERING DEPARTMENT
SY 2023-2024

MEMORANDUM OF AGREEMENT BETWEEN THESIS MEMBERS AND THESIS PANEL

This thesis entitled:

"k-Deep Autoencoder-Based Clustering of Channel Generated Multipath Components"

prepared and submitted by JIMENEZ, Ishi Jemila E., LEONCIO, Timothy D., MACAPAGAL, Faithlix Mark M., MALIWAT, Jerome R., MANALANG, Neera Mae C., OLALIA, Jhony A. (members), in partial fulfillment of the requirements for the degree of Bachelor of Science in Electronics and Communications Engineering has been examined and is recommended for acceptance and approval for THESIS PROPOSAL DEFENSE.


Engr. Emmanuel T. Trinidad
Adviser

Approved by the Committee on Oral Examination (Thesis Proposal) with a defense result of _____ on November 29, 2023 (oral defense date).

With the following Requirements and Recommendations for FINAL DEFENSE

1. Incorporate the importance of channel modeling. (Rationale of the Study)
2. Conduct interview. (Appendix)
3. State on who will benefit on the study. (Significance of the study)
4. Identify Research Gap – Why there is a need for another study on clustering. (Background of the Study)
5. Revise Objective (Concise, Specific, Measurable). Make the objectives more specific to implement accordingly (Objectives).
6. Why use this (Jaccard Index) validation Index? (Background of the Study) *but all possible validation index justify the reason of Jaccard index advantages over the other validation*
7. Comparative Analysis of why k-DAE? (Existing Works on k-DAE)

Appendix Q –
MOA Between Thesis Members and Panel: Rev 09-2023

CS CamScanner

Fig. J.1 Memorandum of Agreement between Thesis Members and Thesis Panel



DON HONORIO VENTURA STATE UNIVERSITY

Appendix K REVISION TO FINAL



TABLE K.1 REVISION TO FINAL

Examiner	Comments	Summary of how the comment was addressed	Location
Engr. Enmar T. Tuazon	<ol style="list-style-type: none">1. You said that the method you used is something new, how can we say that your produced output is reliable? How can you validate that?2. Given the datasets, can you describe what they are and what the "error" (reconstruction error in result sheets) is referring to?3. What is the acceptable Jaccard Index value?4. Expand on the background of the study, answering "what is the significance of the study"5. What does the NMI, ARI, ACC, and JI explain? Explain in simple terms.-table6. Make your tables more presentable, notably Table 22 and Table 23.7. Improve Chapter 6 so that it is easier to read and understand. Broaden your explanations and interpretations.8. Clearly state the future directives of the study. Include all the recommendations that were mentioned that you were not able to produce and accomplish.9. Special request under chapter 6 summary- elaborate your significant findings and describe their meaning and use case of these results and what they might contribute to and to whom.	<ol style="list-style-type: none">1. Showed the role of CVIs.2. Clarified RE in design considerations.3. Showed the computation of JI.4. Revised and added significance to the background of the study.5. Explained CVIs using a summarize table.6. Addressed redundant tables and table captions.7. Improved and revised Chapter 6.8. Added and revised recommendations.9. Improved and revised Chapter 6.	<ol style="list-style-type: none">1. Sec. 4.3 p.462. Subsec. 4.2.1 p.453. Subsec. 4.3.1 p.484. Sec. 1.1 p.25. Sec. 4.3 p.466. Subsec. 6.2.3 p.82 Subsec. 6.2.4 p. 847. Chap. 6 p.648. Sec. 7.3 p.1249. Chap. 6 p.64

Continued on next page



DON HONORIO VENTURA STATE UNIVERSITY

Continued from previous page

Examiner	Comments	Summary of how the comment was addressed	Location
Engr. Elias Francis G. Vergara	<ol style="list-style-type: none">1. Have you considered using TensorFlow with R or Java?2. Have you considered adding a UI?3. Have you considered making the code so that it runs all the sheets automatically?4. Have you considered running your code in Linux?5. Add to recommendations, try in other OS, and UI, ways on making the algorithm run more smoothly and faster in larger datasets.	<ol style="list-style-type: none">1. Added to recommendations.2. Added to future directive.3. Added to future directive.4. Added to recommendations.5. Added to future directive.	<ol style="list-style-type: none">1. Sec. 7.3 p.1242. Sec. 7.4 p.1243. Sec. 7.4 p.1244. Sec. 7.3 p.1245. Sec. 7.4 p.124

Continued on next page



DON HONORIO VENTURA STATE UNIVERSITY

Continued from previous page

Examiner	Comments	Summary of how the comment was addressed	Location
Mr. Larie Joseph R. Lacsina	<ol style="list-style-type: none">1. How many actual scenarios does the IMT-2020 have? Explain the use of 8 scenarios instead of 11.2. Have you tried using the ELU-ELU-ADAM configuration with the IMT-2020 dataset? Can you assume that the result would be the same if it was performed?3. What is your basis of using Jaccard Index? Justify more on why you use Jaccard Index.4. Validation of your results in statistics.5. In the summary of results justify if IMT2020 has a negative correlation because I don't see it as a negative correlation.6. If also possible, try to simulate the IMT2020 in default k-DAE configuration to have a comparison to prove that sigmoid-sigmoid-SGD improved the results. That default result then will be your baseline on your thesis.7. Add recommendations, to compare the clustering results of IMT2020.	<ol style="list-style-type: none">1. Added to delimitation.2. Added results to Chapter 6.3. Showed comparison of table in design considerations4. Validated results using Jamovi.5. Clarified in chapter 6.6. Same with item 2.7. Added to recommendations.	<ol style="list-style-type: none">1. Subsec. 1.6.3 p.132. Chap. 6 p.643. Sec. 4.3 p.464. Subsec. 6.3.3 p.88 Subsec. 6.3.4 p.915. Subsec. 6.4.2 p.1116. Chap. 6 p.647. Sec. 7.3 p.124



Appendix L

ANOVA OF COST 2100 USING JAMOVI



Results

One-Way ANOVA

One-Way ANOVA (Welch's)

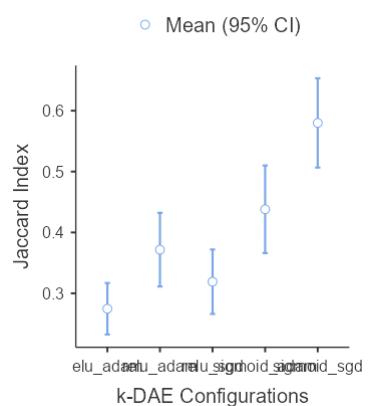
	F	df1	df2	p
Jaccard Index	14.7	4	146	< .001

Group Descriptives

	k-DAE Configurations	N	Mean	SD	SE
Jaccard Index	elu_adam	60	0.275	0.164	0.0211
	relu_adam	60	0.372	0.234	0.0303
	relu_sgd	60	0.319	0.205	0.0265
	sigmoid_adam	60	0.438	0.279	0.0360
	sigmoid_sgd	60	0.580	0.284	0.0366

Plots

Jaccard Index



Post Hoc Tests

Fig. L.1 ANOVA of COST 2100 All Indoor Scenarios Using Jamovi



Results

One-Way ANOVA

One-Way ANOVA (Welch's)

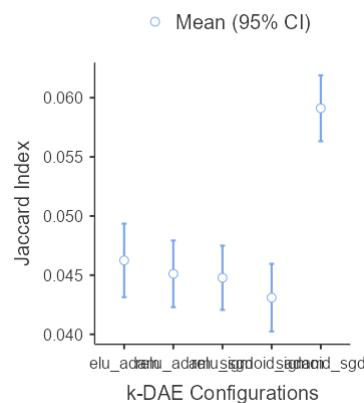
	F	df1	df2	p
Jaccard Index	20.9	4	447	< .001

Group Descriptives

	k-DAE Configurations	N	Mean	SD	SE
Jaccard Index	elu_adam	180	0.0462	0.0211	0.00158
	relu_adam	180	0.0451	0.0191	0.00143
	relu_sgd	180	0.0448	0.0185	0.00138
	sigmoid_adam	180	0.0431	0.0194	0.00145
	sigmoid_sgd	180	0.0591	0.0189	0.00141

Plots

Jaccard Index



Post Hoc Tests

Fig. L.2 ANOVA of COST 2100 All Outdoor Scenarios Using Jamovi



Results

One-Way ANOVA

One-Way ANOVA (Welch's)

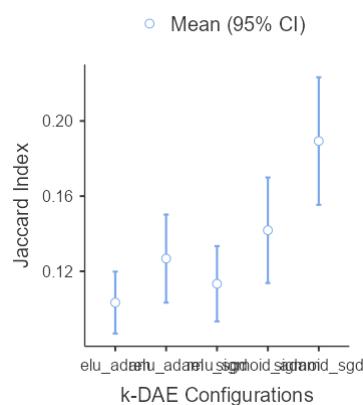
	F	df1	df2	p
Jaccard Index	5.74	4	589	< .001

Group Descriptives

	k-DAE Configurations	N	Mean	SD	SE
Jaccard Index	elu_adam	240	0.103	0.130	0.00836
	relu_adam	240	0.127	0.184	0.01189
	relu_sgd	240	0.113	0.157	0.01016
	sigmoid_adam	240	0.142	0.221	0.01426
	sigmoid_sgd	240	0.189	0.267	0.01723

Plots

Jaccard Index



Post Hoc Tests

Fig. L.3 ANOVA of COST 2100 All Scenarios Using Jamovi



Appendix M

ANOVA OF IMT-2020 USING JAMOVI



Results

One-Way ANOVA

One-Way ANOVA (Welch's)

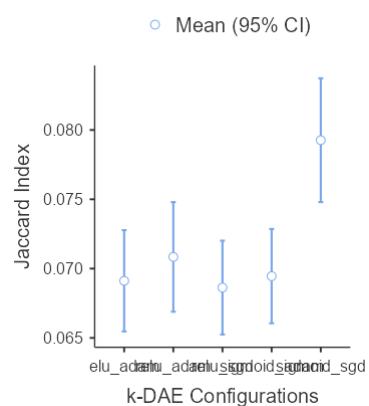
	F	df1	df2	p
Jaccard Index	4.32	4	147	0.002

Group Descriptives

	k-DAE Configurations	N	Mean	SD	SE
Jaccard Index	elu_adam	60	0.0691	0.0141	0.00183
	relu_adam	60	0.0708	0.0153	0.00198
	relu_sgd	60	0.0686	0.0131	0.00169
	sigmoid_adam	60	0.0695	0.0132	0.00170
	sigmoid_sgd	60	0.0793	0.0173	0.00223

Plots

Jaccard Index



Post Hoc Tests

Fig. M.1 ANOVA of IMT-2020 All Indoor Scenarios Using Jamovi



Results

One-Way ANOVA

One-Way ANOVA (Welch's)

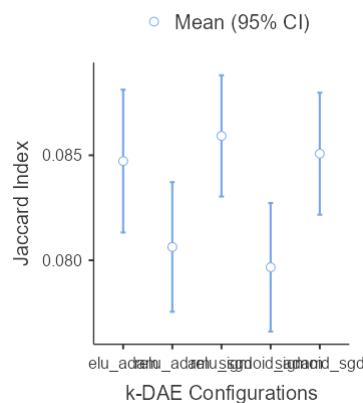
	F	df1	df2	p
Jaccard Index	3.42	4	431	0.009

Group Descriptives

	k-DAE Configurations	N	Mean	SD	SE
Jaccard Index	elu_adam	180	0.0847	0.0231	0.00172
	relu_adam	180	0.0806	0.0210	0.00156
	relu_sgd	150	0.0859	0.0179	0.00146
	sigmoid_adam	180	0.0797	0.0208	0.00155
	sigmoid_sgd	180	0.0851	0.0198	0.00147

Plots

Jaccard Index



Post Hoc Tests

Fig. M.2 ANOVA of IMT-2020 All Outdoor Scenarios Using Jamovi



Results

One-Way ANOVA

One-Way ANOVA (Welch's)

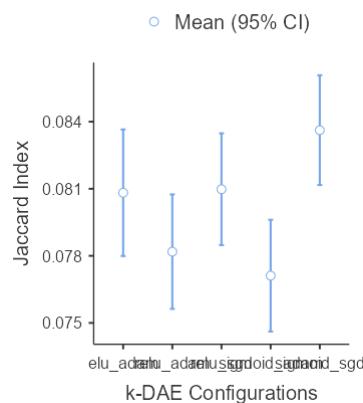
	F	df1	df2	p
Jaccard Index	4.09	4	581	0.003

Group Descriptives

	k-DAE Configurations	N	Mean	SD	SE
Jaccard Index	elu_adam	240	0.0808	0.0223	0.00144
	relu_adam	240	0.0782	0.0201	0.00130
	relu_sgd	210	0.0810	0.0184	0.00127
	sigmoid_adam	240	0.0771	0.0196	0.00127
	sigmoid_sgd	240	0.0836	0.0193	0.00125

Plots

Jaccard Index



Post Hoc Tests

Fig. M.3 ANOVA of IMT-2020 All Scenarios Using Jamovi



Appendix N DATASETS



The screenshot shows the IEEE DataPort homepage with a search bar at the top. Below it, there are navigation links for DATASETS, SUBMIT A DATASET, COMPETITIONS, and SEARCH. The IEEE logo is also present. A large banner features the word "Datasets" over a globe graphic with binary code. A "Open Access" badge is visible on the right side of the banner. Below the banner, a title reads "6 DATASETS FOR MULTIPATH CLUSTERING AT 285 MHZ AND 5.3 GHZ BANDS BASED ON COST 2100 MIMO CHANNEL MODEL". To the left of the title is a thumbnail image of the Earth from space. To the right of the title are details about the dataset, including citation information, submission details, and metadata like views, categories, and keywords. At the bottom of the page, there are buttons for CITE and SHARE/EMBED.

Citation Author(s): Jojo Blanza (*De La Salle University*)
Antipas Jr. Teologo (*De La Salle University*)
Lawrence Materum (*De La Salle University*)

Submitted by: Antipas Jr. Teologo

Last updated: Mon, 01/20/2020 - 00:14

DOI: 10.21227/4cb9-hf81

Data Format: xlsx

License: Creative Commons Attribution

1264 Views

Categories: Communications

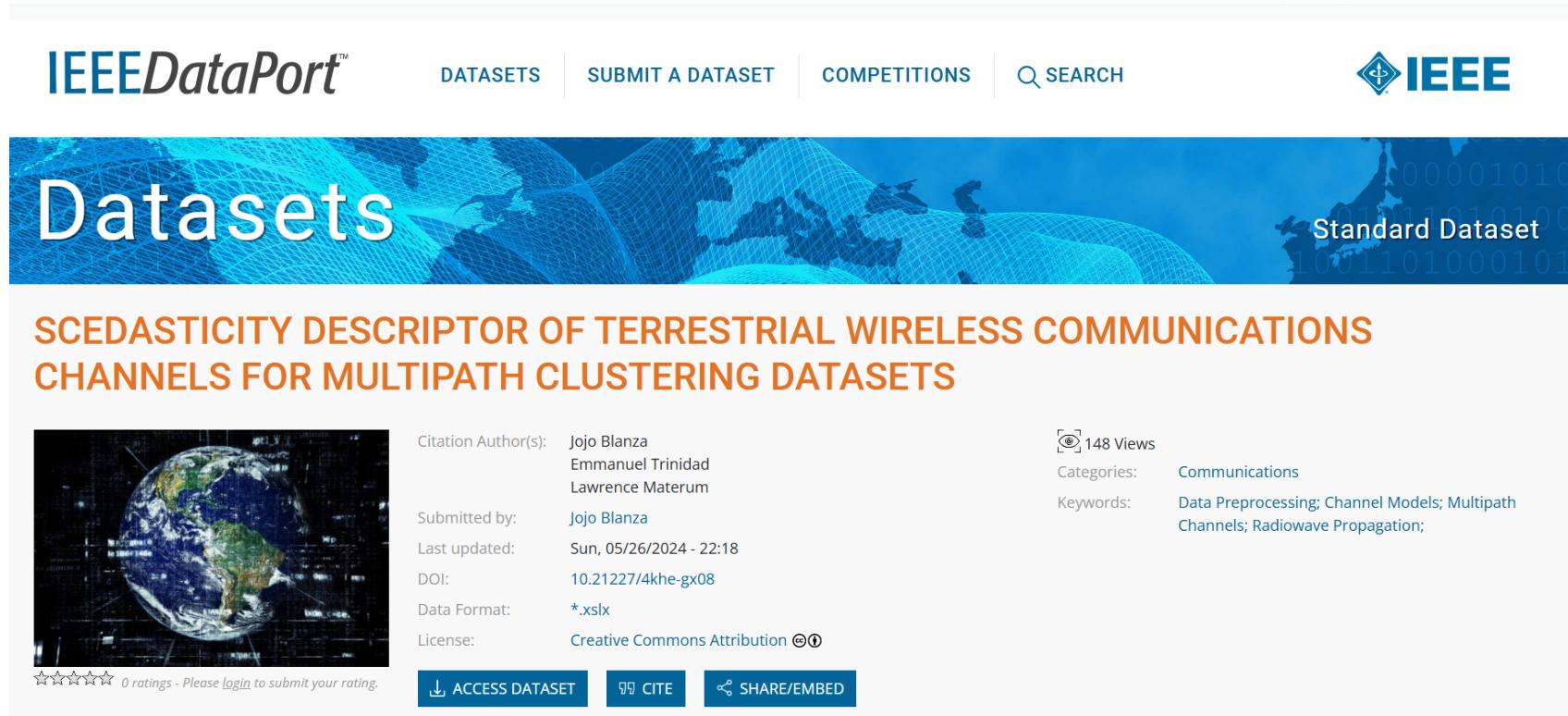
Keywords: Data Handling, Data Models, Data Preprocessing; Channel Models; Multipath Channels; Radiowave Propagation; Clustering Methods

★★★★★ 1 rating - Please [login](#) to submit your rating.

CITE SHARE/EMBED

Fig. N.1 COST 2100 datasets from IEEE Dataport

The C2CM dataset used in this study are found on IEEE Dataport using the link: C2CM Datasets [Blanza and Materum, 2019a]



The screenshot shows the IEEE DataPort homepage with a search bar and navigation links for Datasets, Submit a Dataset, Competitions, and Search. The main banner features a globe and the word "Datasets". Below the banner, the title of the dataset is displayed in large orange text: "SCEDASTICITY DESCRIPTOR OF TERRESTRIAL WIRELESS COMMUNICATIONS CHANNELS FOR MULTIPATH CLUSTERING DATASETS". To the right of the title, there is a "Standard Dataset" badge with binary code. The dataset details include:
Citation Author(s): Jojo Blanza, Emmanuel Trinidad, Lawrence Materum
Submitted by: Jojo Blanza
Last updated: Sun, 05/26/2024 - 22:18
DOI: 10.21227/4khe-gx08
Data Format: *.xlsx
License: Creative Commons Attribution 
The dataset has 0 ratings and a "Please login to submit your rating" message.
Action buttons include: ACCESS DATASET, CITE, and SHARE/EMBED.

Fig. N.2 IMT-2020 datasets from IEEE Dataport

The IMT-2020 dataset used in this study are found on IEEE Dataport using the link: IMT-2020 Datasets [Blanza et al., 2023]



Appendix O PROGRAM LISTING



00.1 Main Runfile

This script is the main code that the researchers run when running the algorithm.



Listing O.1 Main Run file

```
1 from src.k_dae import KDAE #external library
2 from src.utils import load_data, cluster_performance, menu, excel, acc
3     #external library
4 from scipy.spatial.distance import hamming #library in python
5 from sklearn import metrics #library in python
6 from pandas import ExcelWriter #library in python
7 import numpy as np #library in python
8 import pandas as pd #library in python
9 import logging #python module
10 from pathlib import Path
11 import os
12 import argparse
13 import timeit
14 import winsound
15 os.environ["CUDA_VISIBLE_DEVICES"] = "0"
16
17 #logs the data to the k_dae.log notepad
18 def config_logger(log_path='k_dae.log'):
19     logging.basicConfig(filename=log_path, level=logging.DEBUG,
20                         filemode='w',
21                         format='%(asctime)s - %(name)s - %(levelname)s -\n%(message)s')
22
23 def relabel(labels):
24     keys = list(dict.fromkeys(labels))
25     return [keys.index(label) for label in labels]
26
27 if __name__ == '__main__':
28     parser = argparse.ArgumentParser()
29
30     parser.add_argument('-sd', '--save_dir',
31                         type=str,
32                         default='save',
33                         help='path to save')
34
35     parser.add_argument('-dn', '--dataset_name',
36                         choices=['cost', 'imt2020'],
37                         default='imt2020',
38                         help='dataset name [cost, imt2020]')
39
40     FFLAGS, unparsed = parser.parse_known_args()
41     save_dir_name = FFLAGS.save_dir
42     dataset_name = FFLAGS.dataset_name
43     path_dir = Path(os.path.join(save_dir_name, dataset_name))
44     path_dir.mkdir(parents=True, exist_ok=True)
45     log_path = os.path.join(save_dir_name, dataset_name, 'k_dae.log')
46     config_logger(log_path)
47     logging.debug('Start running dataset name {}'.format(dataset_name))
48
49     xlsxname = 'Sample'
50     xlsxname = menu(xlsxname)
51     xlsxcreate = 'Sample2'
52     xlsxcreate = excel()
53     option1 = int(input('Select your choice: '))
54     while option1 != 2:
55         if option1 == 1:
56             print(xlsxname)
57             df1 = pd.DataFrame()
58             df1.to_excel(str(xlsxname)+'.xlsx')
59             break
60         else:
```



Listing O.2 Main Run file (continued)

```
1     print('Invalid Option.')
2     print()
3     excel()
4     option1 = int(input('Enter your option:'))
5     print()
6
7     [x_train, y_train] = load_data(dataset_name)
8     tic = timeit.default_timer()
9     n_cluster = len(np.unique(y_train))
10    model = KDAE(number_cluster=n_cluster, k_dae_epoch=5, epoch_ae=20,
11                  initial_epoch=100, dataset_name=dataset_name)
12    model = KDAE(number_cluster=n_cluster, dataset_name=dataset_name)
13    model.fit(x_train, y_train, dataset_name=dataset_name)
14    y_pred = model.predict(x_train)
15    reconstruction_errors = model.get_reconstruction_errors()
16    print("Reconstruction Errors:", reconstruction_errors)
17    cluster_performance(y_pred, y_train)
18
19    duration = 1000 # milliseconds
20    freq = 520 # Hz
21    freq1 = 800 # Hz
22    winsound.Beep(freq, duration)
23    winsound.Beep(freq1, duration)
24
25    n_pred = len(np.unique(y_pred))
26    k_means_nmi = metrics.normalized_mutual_info_score(y_train, y_pred)
27    k_means_ari = metrics.adjusted_rand_score(y_train, y_pred)
28    k_means_acc = acc(np.int0(y_train), y_pred)
29    k_means_jac = 1 - hamming(np.int0(y_train), relabel(y_pred))
30    toc = timeit.default_timer()
31    mon, sec = divmod(toc-tic, 60)
32    hr, mon = divmod(mon, 60)
33
34    print()
35    print("'''Initial Clustering Only'''")
36    y_initial_raw = input("Copy and Paste the y_initial from k_dae log here:")
37    y_initial=y_initial_raw.split()
38    y_initial=list(map(int, y_initial))
39    y_initial = np.array(y_initial)
40    n_initial = len(np.unique(y_initial))
41
42    k_means_nmi_ini = metrics.normalized_mutual_info_score(y_train,
43                                              y_initial)
44    k_means_ari_ini = metrics.adjusted_rand_score(y_train, y_initial)
45    k_means_acc_ini = acc(np.int0(y_train), y_initial)
46    k_means_jac_ini = 1 - hamming(np.int0(y_train), relabel(y_initial))
47
48    print()
49    col_num = int(input('Enter Column Number to be save range: 1~99: '))
50    df1 = pd.DataFrame({ 'y_train':y_train, 'y_initial':y_initial, 'y_pred':
51                         ':y_pred'})
52    df2 = pd.DataFrame(data=None, columns=["Sheet "+str(col_num)])
53    df3 = pd.DataFrame(data=None, columns=[', 'NMI_ini', 'ARI_ini', ,
54                                         'ACC_ini', 'JAC_ini',
55                                         'NMI', 'ARI', 'ACC', 'JAC', 'TRUE
56                                         ', 'INITIAL',
57                                         'PRED', 'TIME', 'Reconstruction
58                                         _errors'])
59
60    df4 = pd.DataFrame({':Sheet '+str(col_num), 'NMI_ini':
61                         k_means_nmi_ini, 'ARI_ini':
```



Listing O.3 Main Run file (continued)

```
1      :k_means_ari_ini,'ACC_ini':k_means_acc_ini,'
2          JAC_ini':
3      k_means_jac_ini,'NMI':k_means_nmi,'ARI':
4          k_means_ari,'ACC',
5      :k_means_acc,'JAC':k_means_jac,'TRUE':n_cluster
6          ,INITIAL,
7      :n_initial,'PRED':n_pred,'TIME':'%d:%02d:%02d"
8          % (hr, mon, sec),
9      'Reconstruction_Error': reconstruction_errors},
10         index=[0])
11 writer = ExcelWriter(xlsxname+'.xlsx', mode='a', if_sheet_exists='
12         overlay')
13 df1.to_excel(writer, sheet_name='Sheet1', index=False, startrow=1,
14             startcol=((col_num - 1)*3))
15 df2.to_excel(writer, sheet_name='Sheet1', index=False, startrow=0,
16             startcol=((col_num - 1)*3))
17 df3.to_excel(writer, sheet_name='Sheet2', index=False, startrow=1,
18             startcol=0)
19 df4.to_excel(writer, sheet_name='Sheet2', index=False, startrow=(
20                 col_num + 1),
21                 startcol=0, header=False)
22 writer.close()
23
24 print("Elapsed_time:", toc, tic)
25 print("Elapsed_time_during_the_whole_program_in_seconds:", "%d:%02d
26         :%02d" % (hr, mon, sec))
27 duration = 1000 # milliseconds
28 freq = 520 # Hz
29 freq1 = 800 # Hz
30 winsound.Beep(freq, duration)
31 winsound.Beep(freq1, duration)
```



00.2 Utilities

This script is the utilities code, which is responsible for loading of excel datasets C2CM and IMT-2020.



Listing O.4 Utilities Script

```
1 from sklearn.cluster import KMeans
2 from sklearn import metrics
3 import numpy as np
4 import logging
5 import tensorflow as tf
6 from scipy.optimize import linear_sum_assignment as linear_assignment
7 from scipy.spatial.distance import hamming
8
9 #import excel file
10 import pandas as pd
11
12 #select excel file
13 from tkinter import filedialog, Tk
14
15 #Removing Initial Cluster
16 import os
17
18 def menu(ExcelName):
19     data = str(input('Select datasets cost or imt2020: '))
20     print()
21     if data == 'cost':
22         ExcelName = 'Ex'
23         print('1 - 01_Indoor_B1_LOS_Single_Results')
24         print('2 - 02_Indoor_B2_LOS_Single_Results')
25         print('3 - 03_SemiUrban_B1_LOS_Single_Results')
26         print('4 - 04_SemiUrban_B2_LOS_Single_Results')
27         print('5 - 05_SemiUrban_B1_NLOS_Single_Results')
28         print('6 - 06_SemiUrban_B2_NLOS_Single_Results')
29         print('7 - 07_SemiUrban_B1_LOS_Multiple_Results')
30         print('8 - 08_SemiUrban_B2_LOS_Multiple_Results')
31         print()
32     option = int(input('Select a number for the filename: '))
33     while option != 999:
34         if option == 1:
35             ExcelName = '01_Indoor_B1_LOS_Single_Results'
36             print(ExcelName + '\n')
37             return ExcelName
38         elif option == 2:
39             ExcelName = '02_Indoor_B2_LOS_Single_Results'
40             print(ExcelName + '\n')
41             return ExcelName
42         elif option == 3:
43             ExcelName = '03_SemiUrban_B1_LOS_Single_Results'
44             print(ExcelName + '\n')
45             return ExcelName
46         elif option == 4:
47             ExcelName = '04_SemiUrban_B2_LOS_Single_Results'
48             print(ExcelName + '\n')
49             return ExcelName
50         elif option == 5:
51             ExcelName = '05_SemiUrban_B1_NLOS_Single_Results'
52             print(ExcelName + '\n')
53             return ExcelName
54         elif option == 6:
55             ExcelName = '06_SemiUrban_B2_NLOS_Single_Results'
56             print(ExcelName + '\n')
57             return ExcelName
58         elif option == 7:
59             ExcelName = '07_SemiUrban_B1_LOS_Multiple_Results'
60             print(ExcelName + '\n')
61             return ExcelName
62         elif option == 8:
63             ExcelName = '08_SemiUrban_B2_LOS_Multiple_Results'
```



Listing O.5 Utilities Script (continued)

```
1     print(ExcelName+'\n')
2     return ExcelName
3 else:
4     print('Invalid Option.')
5     print()
6     menu(ExcelName)
7     option = int(input('Select a number for the filename: '))
8     print()
9
10 elif data == 'imt2020':
11     ExcelName = 'Ex'
12     print('1 - 01_InH_A_LOS_Results')
13     print('2 - 02_InH_A_NLOS_Results')
14     print('3 - 03_RMa_A_LOS_Results')
15     print('4 - 04_RMa_A_NLOS_Results')
16     print('5 - 05_UMa_A_LOS_Results')
17     print('6 - 06_UMa_A_NLOS_Results')
18     print('7 - 07_UMi_A_LOS_Results')
19     print('8 - 08_UMi_A_NLOS_Results')
20     print()
21     option = int(input('Select a number for the filename: '))
22     while option !=999:
23         if option == 1:
24             ExcelName = '01_InH_A_LOS_Results'
25             print(ExcelName+'\n')
26             return ExcelName
27         elif option == 2:
28             ExcelName = '02_InH_A_NLOS_Results'
29             print(ExcelName+'\n')
30             return ExcelName
31         elif option == 3:
32             ExcelName = '03_RMa_A_LOS_Results'
33             print(ExcelName+'\n')
34             return ExcelName
35         elif option == 4:
36             ExcelName = '04_RMa_A_NLOS_Results'
37             print(ExcelName+'\n')
38             return ExcelName
39         elif option == 5:
40             ExcelName = '05_UMa_A_LOS_Results'
41             print(ExcelName+'\n')
42             return ExcelName
43         elif option == 6:
44             ExcelName = '06_UMa_A_NLOS_Results'
45             print(ExcelName+'\n')
46             return ExcelName
47         elif option == 7:
48             ExcelName = '07_UMi_A_LOS_Results'
49             print(ExcelName+'\n')
50             return ExcelName
51         elif option == 8:
52             ExcelName = '08_UMi_A_NLOS_Results'
53             print(ExcelName+'\n')
54             return ExcelName
55     else:
56         print('Invalid Option.')
57         print()
58         menu(ExcelName)
59         option = int(input('Select a number for the filename: '))
60         print()
```



Listing O.6 Utilities Script (continued)

```
1 def excel():
2     data = str(input('Select\u00a0excel\u00a0datasets\u00a0cost\u00a0or\u00a0imt2020:\u00a0'))
3     if data == 'cost':
4         file_path = r'C:\Users\Mark\Macapagal\Desktop\MasterCode2\
5             save\cost\initial_cluster.npy'
6     elif data == 'imt2020':
7         file_path = r'C:\Users\Mark\Macapagal\Desktop\MasterCode2\
8             save\imt2020\initial_cluster.npy'
9     if os.path.exists(file_path):
10        os.remove(file_path)
11        print('Initial\u00a0Cluster\u00a0has\u00a0been\u00a0Deleted')
12    else:
13        print('No\u00a0Initial\u00a0Cluster\u00a0Found.\u00a0')
14    print()
15    print('1\u2014\u2014Create\u00a0a\u00a0New\u00a0Excel\u00a0File')
16    print('2\u2014\u2014Proceed')
17    print()
18
19
20 def acc(y_true, y_pred):
21     """ Calculate clustering accuracy
22     Require scikit-learn installed
23
24     :param y_true: true labels
25     :param y_pred: predicted labels
26     :return: accuracy, in [0,1]
27     """
28     y_true = y_true.astype(np.int64)
29     assert y_pred.size == y_true.size
30     D = max(y_pred.max(), y_true.max()) + 1
31     w = np.zeros((D, D), dtype=np.int64)
32     for i in range(y_pred.size):
33         w[y_pred[i], y_true[i]] += 1
34
35     ind = linear_assignment(w.max() - w)
36     ind=np.array(list(zip(*ind)))
37     return sum([w[i, j] for i, j in ind]) * 1.0 / y_pred.size
38
39
40 def k_means(x_train, n_class, n_init=100):
41     """ compute k_means algorithm
42
43     use scikit-learn to compute k-means
44
45     :param x_train: data points
46     :param n_class: number of clusters
47     :param n_init: The number of different initialization
48     :return: k_means model
49     """
50     k_mean = KMeans(n_clusters=n_class, n_init=n_init)
51     km_model = k_mean.fit(x_train)
52     return km_model
53
54 def relabel(labels):
55     keys = list(dict.fromkeys(labels))
56     return [keys.index(label) for label in labels]
57
58 def cluster_performance(y_pred, y_train, label='kmean'):
59     """ calculate performance of clustering
```



Listing O.7 Utilities Script (continued)

```
1      :param y_pred: Predication vector
2      :param y_train: Ground truth vector
3      :param label: Method name
4      :return: NMI, ACC, ARI
5      """
6
7      k_means_nmi = metrics.normalized_mutual_info_score(y_train, y_pred)
8      k_means_ari = metrics.adjusted_rand_score(y_train, y_pred)
9      k_means_acc = acc(np.int0(y_train), y_pred)
10     k_means_jac = 1 - hamming(np.int0(y_train), relabel(y_pred))
11     print('{} NMI is {}'.format(label, k_means_nmi))
12     print('{} ARI is {}'.format(label, k_means_ari))
13     print('{} Acc is {}'.format(label, k_means_acc))
14     print('{} Jac is {}'.format(label, k_means_jac))
15     #logging.info("NMI - {:.2f}, ARI - {:.2f}, ACC - {:.2f}, JAC - {:.2f}"
16     #              .format(k_means_nmi, k_means_ari, k_means_acc,
17     #                     k_means_jac))
17     logging.info("NMI - {:.0f}, ARI - {:.0f}, ACC - {:.0f}, JAC - {:.0f}"
18     #              .format(k_means_nmi, k_means_ari, k_means_acc, k_means_jac))
19     logging.info("Numbers in num_list are: {}".format(' '.join(map(str,
20     y_pred))))
21     return k_means_nmi, k_means_acc, k_means_ari, k_means_jac
22
23 def load_data(data_name):
24
25     if data_name == 'cost':
26         print('Select an Excel File in the File Explorer\n')
27         Tk().withdraw()
28         filepath = filedialog.askopenfilename(
29             initialdir=r'C:\Users\Mark Macapagal\Desktop\MasterCode2\
30             DATASETS',
31             title="Open Excel file",
32             filetypes=(("Excel files", "*.xlsx"), ("all files", "*.*")))
33     if not filepath:
34         print("NO FILE SELECTED.")
35         return filepath
36     elif filepath:
37         sheet_Num=str(input('Enter Sheet Number: '))
38         print()
39         t=pd.read_excel(filepath,sheet_name='Sheet'+sheet_Num)
40         print(t)
41         data=t.to_numpy()
42         x_train=data[:, :-2]
43         y_train = relabel(data[:, -1])
44         return [x_train, y_train]
45
46     if data_name == 'imt2020':
47         print('Select an Excel File in the File Explorer\n')
48         Tk().withdraw()
49         filepath = filedialog.askopenfilename(
50             initialdir=r'C:\Users\Mark Macapagal\Desktop\MasterCode2\
51             DATASETS',
52             title="Open Excel file",
53             filetypes=(("Excel files", "*.xlsx"), ("all files", "*.*")))
54     if not filepath:
55         print("NO FILE SELECTED.")
56         return filepath
57     elif filepath:
58         sheet_Num=str(input('Enter Sheet Number: '))
59         print()
60         t=pd.read_excel(filepath,sheet_name='Sheet'+sheet_Num)
```



Listing O.8 Utilities Script (continued)

```
1     print(t)
2     data=t.to_numpy()
3     x_train=data[:, :-2]
4     y_train = relabel(data[:, -1])
5     return [x_train, y_train]
```



00.3 k-DAE

This script is the k -DAE code, which is adapted algorithm to cluster C2CM and IMT-2020.



Listing O.9 k-DAE file

```
1 from src.autoencoder import AutoEncoder
2 from src import utils
3 import logging
4 import numpy as np
5 from keras.layers import Input, concatenate, Reshape
6 from keras.models import Model
7 import tensorflow as tf
8 from keras.callbacks import ModelCheckpoint, Callback
9 import os
10 #tf.config.set_visible_devices([], 'GPU')
11 os.environ["CUDA_VISIBLE_DEVICES"] = "0"
12
13
14 class ChangeCluster(Callback):
15     def __init__(self, k_dae, x_train, y_train, y_pred=None):
16         super(ChangeCluster, self).__init__()
17         self.k_dae = k_dae
18         self.y_pred = y_pred
19         self.x_train = x_train
20         self.y_train = y_train
21
22     def on_epoch_end(self, epoch, logs=None):
23         if epoch % 5 == 0:
24             y_pred = self.k_dae.predict(self.x_train)
25             logging.info("Result after {} epochs".format(epoch))
26             utils.cluster_performance(y_pred, self.y_train)
27             #print("the number of not sample change ae is {}".format(
28             #    sum(y_auto_pred == self.y_pred)))
29             self.y_pred = y_pred
30
31
32 class KDae:
33     def __init__(self, number_cluster, dataset_name='temp', k_dae_epoch
34 = 5, ae_initial_dim=(1024, 256, 10, 256, 1024),
35                 initial_epoch=100, ae_dim=(1024, 256, 10, 256, 1024),
36                 epoch_ae=20, batch_size=256, save_dir='save'):
37         self.number_cluster = number_cluster
38         self.k_dae_epoch = k_dae_epoch
39         self.ae_initial_dim = ae_initial_dim
40         self.initial_epoch = initial_epoch
41         self.ae_dim = ae_dim
42         self.epoch_ae = epoch_ae
43         self.batch_size = batch_size
44         self.dataset_name = dataset_name
45         self.save_dir = save_dir
46         self.initial_ae = None
47         self.initial_label = None
48         self.ae_models_list = []
49         self.k_dae_model = None
50         self.reconstruction_errors = []
51
52     def _initial_clustering(self, x_data):
53         _, input_dim = x_data.shape
54         self.initial_ae = AutoEncoder(input_dim, self.ae_initial_dim,
55                                         epoch=self.initial_epoch)
56         self.initial_ae.auto_encoder_model()
57         embed = self.initial_ae.fit(x_data)
58         k_means_initial_model = utils.k_means(embed, self.
59                                             number_cluster)
60         return k_means_initial_model.labels_
61
62     def _create_combination_model(self, input_dim):
63         logging.info("Create the k_dae model")
```



Listing O.10 k-DAE file (continued)

```
1  inputs = Input(shape=(input_dim,), name='Input_layer')
2  ae_list = []
3  for i in range(self.number_cluster):
4      ae_list.append(self.ae_models_list[i].model(inputs))
5      ae_list[i] = Reshape((1, input_dim))(ae_list[i])
6  out = concatenate(ae_list, axis=1, name='Output')
7  model = Model(inputs=inputs, outputs=out)
8  return model
9
10 @staticmethod
11 def k_dae_loss(y_true, y_pred):
12     """ loss of the k_dae
13
14     :param y_true: The output of the k_dae model np.array with
15         shape(batch_size, self.number_cluster, input_dim)
16     :param y_pred: x_data reshape to (batch_size, self.
17         number_cluster, input_dim)
18     :return:
19     """
20     diff = y_true - y_pred
21     reconstruction_error = tf.linalg.norm(diff, axis=-1)
22     min_value = tf.reduce_min(reconstruction_error, axis=-1,
23         keepdims=True)
24     return min_value
25
26 def get_reconstruction_errors(self):
27     return self.reconstruction_errors
28
29 def fit(self, x_data, y_data=None, dataset_name='temp',
30         save_init_label=True, is_pre_train=False):
31     input_size, input_dim = x_data.shape
32     if is_pre_train:
33         try:
34             self.initial_label = np.load(os.path.join(self.save_dir
35                                         , dataset_name, 'initial_cluster.npy'))
36         except FileNotFoundError as e:
37             logging.warning("initial clustering didn't start
38                             initial AE")
38         self.initial_label = self._initial_clustering(x_data)
39     else:
40         self.initial_label = self._initial_clustering(x_data)
41     if y_data is not None:
42         logging.info("##### Initial Clustering Results #####")
43         _ = utils.cluster_performance(self.initial_label, y_data)
44     if save_init_label:
45         logging.info("Save the initial clustering")
46         np.save(os.path.join(self.save_dir, dataset_name,
47                             'initial_cluster'), self.initial_label)
48         _ = y_data
49
50     for i in range(self.number_cluster):
51         logging.info("model number {} create".format(i))
52         self.ae_models_list.append(AutoEncoder(input_dim, self.
53             ae_dim, epoch=self.epoch_ae, verbose=1))
54         self.ae_models_list[i].auto_encoder_model()
55         # train each ae with the initial clustering
56         self.ae_models_list[i].fit(x_data[self.initial_label == i])
57     if y_data is not None:
58         y_predict = self.predict(x_data)
59         logging.info("Clustering result after each ae train
60                     separately")
61         _ = utils.cluster_performance(y_predict, y_data)
```



Listing O.11 k-DAE file (continued)

```
1     check = ModelCheckpoint('k_dae_' + dataset_name, monitor='loss',
2                               save_best_only=True)
3     cb = [check]
4     if y_data is not None:
5         cb.append(ChangeCluster(self, x_data, y_data))
6     self.k_dae_model = self._create_combination_model(input_dim)
7     self.k_dae_model.compile(optimizer='ADAM', loss=self.k_dae_loss
8                               )
9     x_repeat = np.repeat(x_data[:, np.newaxis, :], self.
10                           number_cluster, axis=1)
11    #tf.config.experimental.list_physical_devices(device_type=None)
12    #with tf.devices("/cpu:0"):
13    self.k_dae_model.fit(x_data, x_repeat, epochs=self.k_dae_epoch,
14                          batch_size=self.batch_size,
15                          callbacks=cb)
16    reconstruction_errors = self.k_dae_model.history.history['loss']
17    ]
18    self.reconstruction_errors.append(reconstruction_errors)
19
20    def predict(self, x_data):
21        reconstruction_norm = []
22        for i in range(self.number_cluster):
23            reconstruct = self.ae_models_list[i].model.predict(x_data)
24            delta = x_data - reconstruct
25            reconstruction_norm.append(np.linalg.norm(delta, axis=1))
26        reconstruction_norm_array = np.array(reconstruction_norm)
27        y_predict = np.argmin(reconstruction_norm_array, axis=0)
28        return y_predict
```



00.4 Autoencoder

This script is the autoencoder code, which is a supplementary script for k -DAE algorithm to cluster C2CM and IMT-2020.



Listing O.12 Autoencoder file

```
1 from keras.layers import Dropout, BatchNormalization
2 from keras.layers import Input, Dense, GaussianNoise
3 from keras.models import Model
4 import keras
5 from keras.callbacks import EarlyStopping, ModelCheckpoint
6 from keras.layers import Activation
7 from keras.optimizers import SGD
8
9
10 class AutoEncoder:
11     def __init__(self, data_dim, hidden_dim, batch_normalize=True,
12                  epoch=100, batch_size=256, loss='binary_crossentropy',
13                               optimizer='SGD', save_name='temp', verbose=1,
14                               save_model=False):
15         self.data_dim = data_dim
16         self.hidden_dim = hidden_dim
17         self.batch_normalize = batch_normalize
18         self.epoch = epoch
19         self.batch_size = batch_size
20         self.loss = loss
21         self.optimizer = optimizer
22         self.save_name = save_name
23         self.save_model = save_model
24         self.verbose = verbose
25         self.model = None
26         self.embedding_model = None
27
28     def auto_encoder_model(self):
29         print(self.hidden_dim)
30
31         init = 'glorot_uniform'
32         inputs = Input(shape=(self.data_dim,), name='z')
33         # ran = tf.random_normal(shape=(self.data_dim,), mean=0, stddev
34         # =0.1)
35         x = inputs
36         # x = GaussianNoise(stddev=0.1)(x)
37         for j, i in enumerate(self.hidden_dim[:-1]):
38             x = Dense(i, kernel_initializer=init, name='encoder_%d' % j
39             )(x)
40             if self.batch_normalize:
41                 x = BatchNormalization()(x)
42                 #x = keras.layers.ELU()(x)
43                 x = Activation('sigmoid')(x)
44             embedding_layer = Dense(self.hidden_dim[-1], kernel_initializer
45             =init, name='embedding_layer')(x)
46             x = embedding_layer
47             for j, i in enumerate(self.hidden_dim[1:-1]):
48                 x = Dense(i, kernel_initializer=init, name='decode_%d' % j
49                 )(x)
50                 if self.batch_normalize:
51                     x = BatchNormalization()(x)
52                     #x = keras.layers.ELU()(x)
53                     x = Activation('sigmoid')(x)
54             x = Dense(self.data_dim, kernel_initializer=init, activation=
55             'sigmoid', name='decoder_0')(x)
56             decode = x
57             model = Model(inputs=inputs, outputs=decode)
58             embedding_model = Model(inputs=inputs, outputs=embedding_layer)
59             #plot_model(model, to_file='autoencoder.png', show_shapes=True)
60             self.model = model
61             self.embedding_model = embedding_model
62
63     def fit(self, x_train, patience=10):
```



Listing O.13 Autoencoder file (continued)

```
1      sgd = SGD() # Create SGD optimizer object
2      self.model.compile(loss=self.loss, optimizer=sgd)
#self.model.compile(loss=self.loss, optimizer=self.optimizer)
3      early_stopping = EarlyStopping(monitor='loss', patience=
4          patience)
5      callback_name = [early_stopping]
6      if self.save_model:
7          path_name = self.save_name + "u-{epoch:02d}-{loss:.2f}.hdf5"
8          modelcheckpoint = ModelCheckpoint(path_name, monitor='loss',
9              , save_best_only=True)
10         callback_name.append(modelcheckpoint)
11         self.model.fit(x_train, x_train, epochs=self.epoch, batch_size=
12             self.batch_size,
13                 callbacks=callback_name, verbose=self.verbose)
14         encoder_output = self.embedding_model.predict(x_train)
15         # if to_save_model:
16         #     aetSNE_utils.save_keras_model(encoder, save_name)
17         return encoder_output
18
19
20
21 if __name__ == '__main__':
22     pass
23     # from src import utils
24     # x_train, y_train = utils.load_data('mnist')
25     # ae = AutoEncoder(784, [500, 500, 2000, 10], epoch=1)
26     # ae.auto_encoder_model()
27     # embed = ae.fit(x_train)
```



DON HONORIO VENTURA STATE UNIVERSITY

Appendix P APPROVAL SHEET



DON HONORIO VENTURA STATE UNIVERSITY



Republic of the Philippines
DON HONORIO VENTURA TECHNOLOGICAL UNIVERSITY
Bacolor, Pampanga

COLLEGE OF ENGINEERING AND ARCHITECTURE
ELECTRONICS ENGINEERING DEPARTMENT
SY 2023-2024

FINAL THESIS APPROVAL SHEET

This thesis entitled:

"k-Deep Autoencoder-Based Clustering of Channel Generated Multipath Components"

prepared and submitted by JIMENEZ, Ishi Jemila E., LEONCIO, Timothy D., MACAPAGAL, Faithlix Mark M., MALIWAT, Jerome R., MANALANG, Neera Mae C., OLALIA, Jhony A., (members) , in partial fulfillment of the requirements for the degree of Bachelor of Science in Electronics Engineering has been examined and is recommended for acceptance and approval for FINAL THESIS DEFENSE.

Engr. Emmanuel T. Trinidad
Adviser

Approved by the Committee on Oral Examination (Thesis) with a defense result of Pased at on April 29, 2024 (oral defense date).

Engr. Enmar T. Tuazon
Chair

Mr. Larie Joseph R. Lacsina
Member

Engr. Bias Francis G. Vergara
Member

Engr. Anthony S. Tolentino, MS-ECE
Capstone Project 1 Coordinator

Engr. Nilo Q. Manungag, PECE, MEP-EE
OIC Chairperson, ECE Department

Engr. Jun P. Flores, PECE, MEP-EE
Dean, College of Engineering and Architecture

Appendix X –
Design Capstone Project : Final Thesis Approval Sheet Rev 09-2023

Fig. P.1 Thesis Approval Sheet



Appendix Q

VITA



Ishi Jemila E. Jimenez graduated Senior High School at Bloomingfields Educational Foundation, Inc., Buensuceso, Arayat, Pampanga under the Science, Technology, Engineering and Mathematics (STEM) strand. She is currently taking up her Bachelor of Science in Electronics Engineering at Don Honorio Ventura State University. Her fields of interest are programming and communications.



Timothy D. Leoncio graduated Senior High School under Technical Vocational Livelihood(TVL) Strand at Don Honorio Ventura State University, DHVSU, Bacolor Pampanga. He is currently taking up his Bachelor of Science in Electronics Engineering at Don Honorio Ventura State University. His fields of interest are Communications and Electronics.



Faithlix Mark M. Macapagal graduated Senior High School under Science, Technology, Engineering, and Mathematics (STEM) Strand at Pampanga State Agricultural University, PAC, Magalang, Pampanga. He is currently taking up his Bachelor of Science in Electronics Engineering at Don Honorio Ventura State University. His specializations are Programming, Electronics, and Mathematics.



Jerome R. Maliwat graduated Senior High School under Science, Technology, Engineering, and Mathematics (STEM) Strand at Pampanga State Agricultural University, PAC, Magalang, Pampanga. He is currently taking up his Bachelor of Science in Electronics Engineering at Don Honorio Ventura State University. His fields of interest are Mathematics, Programming, Electronics, and Data Analysis.



Neera Mae C. Manalang graduated Senior High School under Science, Technology, Engineering, and Mathematics (STEM) Strand at University of the Assumption, UA, San Fernando, Pampanga. She is currently taking up her Bachelor of Science in Electronics Engineering at Don Honorio Ventura State University. Her fields of interest are Communications and Data Analysis.



Jhonly A. Olalia graduated Senior High School under Science, Technology, Engineering, and Mathematics (STEM) Strand at Our Lady of Fatima University, OLFU, San Fernando, Pampanga. He is currently taking up his Bachelor of Science in Electronics Engineering at Don Honorio Ventura State University. His fields of interest are Communications and Electronics.