Summer 8-19-2022

Decorrelated Deep Neural Networks: Learning Bias Invariant & Scanner Independent Features, and Causal relationships Using a novel deep learning methods based on Distance Correlation

Pranita Patil
pppatil@alumni.harrisburgu.edu

Decorrelated Deep Neural Networks

Learning Bias Invariant & Scanner Independent Features, and Causal relationships

Using a novel deep learning methods based on Distance Correlation

By

Pranita Patil

A thesis submitted to
Harrisburg University of Science and Technology
for the degree of
DOCTOR OF PHILOSOPHY

Department of Analytics
Harrisburg University of Science and Technology
August 2022

ABSTRACT

Advancements in deep learning or deep neural networks have made it possible to reach expert-level performance in a variety of applications, even in challenging situations. However, a central challenge in all deep learning, as well as machine learning applications, is dealing with its dependency on the quality of data which can be significantly impacted by biases, confounders, and irrelevant variations in data which leads to spurious relationships and erroneous decisions. The main purpose of this dissertation is to build a robust deep learning model which considers and mitigates these biases. Another challenge with the deep learning model is learning associations present in the data rather than causations. This also leads to bias problems and non-interpretable systems. So, the purpose of this dissertation also includes introducing causality in the deep learning models. Thus, developing a novel deep learning model to learn bias invariant features and learn causal discoveries are promising areas of research with high potential impact.

The overarching goal of this dissertation is to improve the performance, reliability, and generalization ability of deep learning even in the presence of biases and spurious associations in the data. This entails several research directions. First, we introduce a decorrelated framework that addresses the imbalanced and scanner dependencies issues present in the Parkinson's Disease (PD) dataset. Second, we further define the general formulation of decorrelated deep learning models. This provides the foundation for generic bias mitigation analysis and the design of robust decorrelated deep learning models. This dissertation also focuses on the topic of Granger Causality (GC) introduction in the deep learning model. Thus, the third research direction includes extending the LSTM-based Granger Causality framework to incorporate Graph Neural Network (GNN) and distance correlation which enables improvement in the performance of the deep learning model and provides interpretable GC interactions.

We propose a novel bias mitigation method for deep learning models by leveraging the distance correlation function to decorrelate the features and biases to provide a robust solution. We explore the use of this method in neuroimaging study settings for disease classification. We also derive the generic decorrelation-based bias mitigation framework for different data scenarios and different deep learning architectures. These results show how our approach provides a robust, flexible, scalable, and generic framework that improves the performance of deep learning models while reducing bias effects on model predictions. In addition to this, we define a mathematical framework to introduce the fusion of GC with GNN and distance correlation and showcase their success in learning complex non-linear Granger causal connections. We study the implications of our work in the deep learning field and discuss future work to further leverage this robust decorrelated framework and improve the performance irrespective of the quality of data.

DEDICATION

To my family

Ph.D. COMMITTEE APPROVAL

To the Faculty of Harrisburg University of Science and Technology:

The members of the Committee appointed to examine the dissertation of PRANITA PATIL find it satisfactory and recommend that it be accepted.

Rand Ford, Ph.D., Chair

Kevin Purcell, Ph.D.

Glenn Mitchell, MD

Maria Vaida, Ph.D.

Sridhar Reddy Ravula, Ph.D.

ACCEPTANCE PAGE

Harrisburg University of Science and Technology

As chair of the candidate's graduate committee, I have read the thesis of Pranita Patil in its final form and have found that (1) its format, citations, and bibliographical style are consistent and acceptable and fulfill university and department style requirements; (2) its illustrative materials including figures, tables, and charts are in place; and (3) the final manuscript is satisfactory to the graduate committee and is ready for submission to the university library.

Rand Ford, Ph.D.
Chair, Graduate Committee
Harrisburg University of Science and Technology

Accepted for the Department of Analytics

Rand Ford, Ph.D.
Chair, Department of Analytics
Harrisburg University of Science and Technology

Accepted for the University

Bilita Mattes, D.Ed,
Provost
Harrisburg University of Science and Technology

ACKNOWLEDGMENTS

My deepest gratitude to Dr. Rand Ford for his mentoring and guidance, and would also like to express my gratitude to my committee members: Dr. Kevin Purcell, Dr. Glenn Mitchell, Dr. Maria Vaida, and Dr. Sridhar Reddy Ravula for being a part of my committee as well as their help and support.

Table of Contents

List of Figures

List of Tables

Chapter One

A General Overview of Deep Learning Methods

1.1 Introduction

Deep Neural Networks are deep, which means having multiple hidden layers in the neural network architecture and mimicking the structure of the human brain. It performs a parameterized non-linear mapping between sets of input and output variables. These networks have proved their ability to achieve successful results in almost every field for various applications such as image recognition, medical, and finances with the help of high computational power and large data sets. Although deep neural networks have received popularity over the past few years, they do face some challenges such as lack of explainability, need for large datasets, computationally expensive, and prone to biased decisions. In recent years, bias problems in deep learning have received a lot of interest and have been a growing field of research. This work will introduce a novel technique based on distance correlation to resolve the issue of the influence of bias on network performance by mitigating the bias while maintaining performance on the main task of interest. Distance correlation captures how closely two variables of any dimension are linearly or non-linearly related to each other. Another major problem found in deep learning is that it is often based on a statistical association between

variables than underlying causal relationships. This work presents an integration of deep learning models and Granger causality with the aim of learning causal relationships. The following section provides a brief introduction to different types of deep neural networks, bias issues, and Granger causality.

1.1.1 Deep Neural Networks

Deep learning is a subset of machine learning which extracts complex patterns or features from the data to perform tasks such as classification, detection, regression, clustering, generation of new samples, etc. Training in deep learning can be done using Supervised Learning (labeled data), Unsupervised Learning (unlabeled data), or Reinforcement Learning. Deep Learning represents a powerful method and can be a positive path in Neuroimaging Diagnosis. Advances in Deep Learning are helping to achieve remarkable breakthroughs in the medical field. There are several types of deep learning, such as Convolutional Neural Networks (CNNs), convolutional gated recurrent unit-convolutional neural networks (convGRU-CNN), Recurrent Neural Networks (RNN), Long short-term memory (LSTM), and Graph Neural Networks discussed in this work.

1.1.1.1 Convolutional Neural Networks

The most commonly used deep learning architecture for computer vision and classification tasks is a Convolutional Neural Network (CNN). The architecture of CNN was inspired by the organization of the cat's visual system, and it provides a more scalable and flexible approach to identifying patterns within images. The research work of Lecun (LeCun, Boser, et al., 1989) laid the foundation for CNN in 1989, which is similar to Neocognitron (Fukushima, Miyake, and Ito, 1983), which was developed by Fukushima in 1980. CNNs mainly consist of three types of layers such as convolution, pooling, and fully connected layers. CNNs have also been used to classify time series data, audio, and speech data. The first two layers

or stack of these layers i.e. convolution and pooling layers, act as feature extractor which automatically extracts features from simple low-level to complex high-level patterns. The last fully connected layer or stack of these layers acts as a classifier and maps the input image to the output. Unlike fully connected multilayer networks, it uses shared weights and fewer connections. Different hyperparameters such as a number of kernel/filter, filter size, stride, and paddings need to be set for the convolutional layer. The pooling layer is used for a downsampling operation along the spatial dimensions, which reduces the dimension and the number of learnable parameters. There are different types of pooling, including maximum pooling, in which the maximum value is determined among all of the elements in each feature map, and average pooling, in which the average value of all the elements in each feature map is calculated. The Convolutional and pooling layer not only extracts features from input data but also obtains small shift and distortion invariance by reducing the spatial resolution of the feature map. The fully connected layers are like artificial neural networks where the input used is one dimensional, and every input is connected to every output. These layers classify the input based on features extracted by previous layers and output the probabilities for each class. The non-linear activation functions such as sigmoid, hyperbolic tangent (tanh), and Rectified linear unit(ReLU) are used to pass the output of operations such as convolution and fully connected layers. The activation for the last layer is dependent on the type of task. Different loss functions and optimization algorithms based on gradient descent are used to train the networks. These models have achieved state-of-the-art performance in many domains.

1.1.1.2 Long Short Term Memory Network

Long Short Term Memory Network (LSTM) (Hochreiter and Schmidhuber, 1997) is a special type of powerful recurrent neural network (RNN) which are designed to reduce vanishing gradient problem while learning the long-term dependencies. RNNs are basically similar to traditional feedforward neural networks but with a recurrent hidden state whose input is

dependent on the previous time or previous hidden state output. But RNNs are not capable of processing long-term dependencies since it stops learning due to vanishing gradient problem. LSTMs have three types of non-linear gates which modulate the flow of information through the cell state. Gates are nothing but a sigmoid neural net layer and a pointwise multiplication operation. Forget gate decides which information in the sequence to be forgotten from the previous cell state, whereas the input and output gate decides which information to keep or store and update. This is potentially significant for sequence-based tasks with long-term dependencies. The modified vanilla LSTM model is a variant of LSTM with the addition of a peephole connection, which adds cell state information to gating layer inputs. These models are trained using Backpropagation Through Time algorithm to update weight parameters. Many modifications to the LSTM architecture have been proposed, which leads to different types of LSTMs such as Depth Gated RNNs (K. Yao et al., 2015), Bidirectional LSTM (Graves and Schmidhuber, 2005), Gated Recurrent Unit (GRU), convolutional LSTM (ConvLSTM), and convolutional gated recurrent unit (ConvGRU), etc. Most often, these networks are used to process temporal sequences, so they are widely used for time series prediction, speech analysis, image and video captioning, language translation or processing, sound or text recognition, etc.

1.1.1.3 Convolutional Gated Recurrent Unit

In order to learn non-linear spatio-temporal features, a combination of convolutional gated recurrent unit (ConvGRU) and CNN are used in Gessert et al., 2018. This notion is inspired by the convolutional LSTM (ConvLSTM) architecture (X. Shi et al., 2015). Fully connected layers in this architecture are replaced by a convolutional structure. ConvGRU (Ballas et al., 2015) is similar to ConvLSTM, except GRU is used instead of LSTM. GRU (Chung et al., 2014) can be used to capture dependencies of varying time scales and has less number of gates, mainly update and reset gates, than LSTM, which has input, output, and forget gates. This helps GRU to have lesser learnable parameters and ultimately reduces the execution time

and less memory requirement. But the major problem with LSTM and GRU is their inability to handle spatio-temporal data processing simultaneously. To overcome this problem, the addition of a convolution structure in GRU helps to preserve spatio-temporal patterns. In ConvGRU-CNN, the first ConvGRU layer processes the temporal information present in the data, and then this processed information is fed to normal CNN architecture to process spatial information and generate a final prediction. This model is trained end-to-end.

1.1.1.4 Graph Neural Networks

Graph neural networks (GNNs) have been receiving more attention since it deals with complex graph-structured data, which are represented by a set of nodes and edges. Adjacency matrices are used to show which nodes are connected to each other. Unlike CNNs which operate on 1D and 2D grids in euclidean space, GNNs operate on graph domain which is an irregular, non-euclidean space, with no spatial locality, permutation invariant, and non-stationary. CNN uses localized convolutional filters and pooling function to extract local spatial features, which makes it difficult to apply in non-euclidean space and also assumes that input instances are independent, but that is not the case with graphs since nodes are related to each other by different types of connections. GNN consists of different types of layers, such as a convolutional or recurrent layer to aggregate the information from all neighbors and a pooling layer to extract high-level information, just like any other deep neural network. GNNs follow a message passing technique (Gilmer et al., 2017) between the adjacent nodes of graphs to aggregate information or features from neighbors using summation or any differentiable and permutation invariant function, and the new node features are passed through the learned neural network to update features. This new node discovers some structure of the graph which depends on the information of two adjacent nodes sharing the edge. For graph classification or regression, global pooling is used to compute a single feature from the whole graph. Convolutional and attention functions are also used where weights are dependent only on the structure of the graph and the features, respectively. Graph Convolu-

tional Networks (GCN) a very popular network is built to generalize to the graph domain i.e. non-euclidean space in an end-to-end manner. Two different types of convolution operations are performed in the spatial domain through direct use of convolution in graph space and in the spectral domain through Fourier and inverse Fourier transform. The loss functions used in these networks are dependent on task prediction types such as node-level, edge-level, and graph-level. GNNS are very powerful networks and have achieved great success in applications like geospatial analysis, traffic prediction, recommender systems, healthcare data analysis, and social influence prediction.

1.1.2 Bias Issues in Deep Learning

Even though Deep learning have been brought successful advances in recent years and is increasingly used in high-stake applications, bias or fairness issues in deep learning remains a problem. Since deep learning is data-driven learning and learn directly from data, the quality of the dataset is very crucial. Collecting such well-distributed and bias-free data requires great effort and time. Hence, the biases or spurious relationships present in the dataset lead to erroneous decisions by exploiting and amplifying this information that hinders the relationship between input and output. There are different types of biases that cause unfairness or bias issues and create algorithmic discrimination. Data bias is already included in the dataset, which is caused by biased device measurements, and biased towards privileged or unprivileged groups such as race, gender, and age. Exclusion bias is caused by missing data and thus making it not a good representative of the population. Algorithm bias is a bias introduced by Algorithmic objective errors that create unfair decisions by favoring one class over another. Some of the examples of biases that influence the results of model prediction are age, gender, race, skin color, religion, language, culture, economic condition, imbalanced dataset, credit history, device bias, marital status, and other demographic information.

Recently many methods have been proposed to mitigate biases and improve the perfor-

mance of the deep learning models. These methods can be categorized into pre-processing, in-processing, and post-processing (Mehrabi et al., 2021). Pre-processing methods make changes in the training dataset in order to remove bias before the model training, in-processing methods modify algorithmic approach or learning process to mitigate bias, whereas post-processing tries to modify the biased output of the model on holdout dataset based on a function after the model training. Many metrics have been proposed to address different bias and discrimination issues. The most commonly used metrices are Equality of Opportunity, Demographic Parity, Disparity Impact, and Equalized Odds. It is also important to note that different applications need different metrices to measure bias mitigation.

1.1.3 Distance Correlation

In order to resolve bias issues mentioned in the above section, this work use distance correlation to find the complex dependencies between learned features of the model and biases. Distance correlation (Székely, Rizzo, and Bakirov, 2007) not only measures statistical independence but also captures non-linear dependencies, unlike Pearson Correlation. Distance correlation measures the joint independence of two random variables in arbitrary dimensions and is free of normal distribution assumptions. The distance correlation is non-negative and ranges between 0 and 1. The zero value of distance correlation implies that two random variables are independent. Distance correlation is estimated using distance covariance, which is calculated using centered Euclidean distances. The predictive power of the model increases as distance correlation increases which means dependency between variables also increases. Due to these advantages and higher statistical power, distance correlation has been proved more practical and valuable in data analysis compared to other correlation methods such as Pearson product-moment correlation and Spearman's rank-order correlation. The only limitation of distance correlation is it doesn't provide positive and negative associations and requires large computation time due to $O(n^2)$ operation where n is a sample size. Distance

correlation has been used in different tasks such as a test of goodness of fit, test dependence of time series data in different contexts, supervised dimensionality reduction using distance correlation maximization/autoencoders(R. Wang, A.-H. Karimi, and Ghodsi, 2018), bayesian approach with distance correlation for medical data analysis (Bhattacharjee, 2014), and gene co-expression network analysis (Hou et al., 2022).

1.1.4 Granger Causality

The causal discovery on time series data is crucial to understanding and interpreting the underlying mechanisms of a system. Granger causality (GC) is a popular method in causality that is based on linear predictability. Causal relationships play an important role in prediction. In this relationship, one variable causes and influences the prediction of another variable. A causal relationship among time series cannot simply be inferred from observational data. It may be necessary to conduct new experiments, intervene, and develop a known mechanism for observed data. A noble prize winner Clive Granger (Granger, 1969) leverages the temporal ordering in time series and introduced a hypothesis test called the Granger causality test, which helps in determining whether one time series causes and helps in forecasting another time series. The main idea behind GC is if the prediction of one variable x or future values of time series improves by the inclusion of past values of time series or variable y, then y Granger causes x. The main focus of this testing is not actual causality but predictive causality, which requires stationary and time-invariant data. GC is relatively simple and normally estimated using the vector autoregressive model (VAR). Mathematically for bivariate and multivariate VAR models, lag polynomial matrices can be used to specify how time lag k affects the prediction of future time series. The values in these matrices determine Granger-causal interactions in the model. Zero value in these matrices indicates Granger non-causal condition. A value greater than zero indicates the Granger-causal condition. A higher value will indicate a stronger GC. When there are more than

two-time series, the bivariate approach is implemented using pairwise analysis, but it leads to erroneous causal inference by increasing estimating errors and parameter inconsistency. Hence a conditional GC is proposed to resolve this problem by calculating GC from x to y conditional on z. This allows to find if there is direct influence or is mediated by z. In traditional linear VAR models, lag selection needs to be specified for estimating GC. Smaller lag values will ignore GC interactions at longer lags, and higher lag values will cause an overfitting issue. Regularization-based approaches such as group lasso penalty will help to find the optimal value for lag from the data. The inconsistent estimation of GC interactions might occur in real-world applications that involve non-linear interactions since GC can only be used for linear models. The recent developments in GC offer component-wise functions using neural networks such as MLP or RNN, or LSTM and implement sparsity-inducing penalty weights to detect Granger non-causal interactions. GC has been used not only in economics but also in various different types of fields such as neuroscience, complex industrial process analysis, climate science, social media analysis, finance, and genomics.

1.2 Prior Work

This section defines the deep learning models and causal and bias-related concepts and gives the background of these models.

1.2.1 Convolutional Neural Networks

The most dominant method used in deep learning is the convolutional neural network (CNN) which has achieved state-of-the-art performance on various tasks. Fukushima, Miyake, and Ito, 1983 first time introduced the neocognitron architecture in 1983, inspired by the animal visual cortex mechanism. The convolution was introduced for the first time in the field of artificial neural networks. LeCun, Boser, et al., 1989 proposed convolutional neural network which is deeper, i.e seven layered network called LeNet-5 which resembles the Neocognitron

and then improved in LeCun, Bottou, et al., 1998. These LeNet architectures are trained using a backpropagation algorithm and are capable of automatically extracting features to recognize visual patterns from raw images without any preprocessing. These architectures encounter problems when the number of layers increases such as overfitting due to lack of sufficient training data, lack of computing power, and gradient vanishing and exploding problems.

To overcome these problems and to further enhance the performance of CNN, Krizhevsky, Sutskever, and Hinton, 2012 proposed AlexNet, which is similar to LeNet but has more deeper structure. The architecture consists of five convolutional and pooling layers followed by three fully-connected layers and has achieved astonishing results in the ImageNet Large Scale Visual Recognition Competition(ILSVRC) (Russakovsky et al., 2015) for object recognition. The authors also introduced a new activation function, Rectified Linear Units (ReLU), to speed up the training process on large datasets and achieved a 16.4% error rate on the ImageNet benchmark database. The dropout layer is also implemented in fully connected layers to reduce overfitting problems and helps to learn more robust features.

With this success, numerous studies have been proposed to improve CNN performance across a wide range of fields. Different variants of AlexNet such as ZFNet (Zeiler and Fergus, 2014), VGGNet (Simonyan and Zisserman, 2014), GoogleNet (Szegedy et al., 2015), and ResNet (He et al., 2016) were proposed. These architectures are more deeper than previous ones, which allows the increasing capability of capturing non-linearity and robust feature representations. ZFnet architecture is similar to AlexNet, but design modification was introduced to improve performance by visualizing intermediate feature layers in order to understand the operation of the classifier. ZFNet makes two modifications in AlexNet, such as changes in the 1st layer filter size from 11x11 to 7x7 and filter stride from 4 to 2. This network reduces the error rate to 11.7% on the ImageNet database. VGGNet aim to increase the depth of CNN architecture by replacing large filter size with small 3×3 convolution filters with 16-19 convolutional layers. Different variants of VGG networks are developed by using

a different number of convolutional and fully connected layers. This network achieves an error rate of 6.8% on the ImageNet database. GoogleNet is employed using the Inception module. These networks are an ensemble of 6 CNN with 22 convolutional layers, and so the authors increase the depth and also the width of the network. The 1×1, 3×3, and 5×5 convolution and global average pooling enable to creat deeper network while keeping the computational cost low. This network reduces the error rate to 6.7% on the ImageNet database. ResNet is around 20 times deeper than AlexNet and eight times deeper than VGGNet but still has lower complexity. It consists of 152 layers. The authors introduced an identity shortcut connection that skips double or triple layers and helps to reduce gradient vanishing problems due to deeper structure. The residual block has two 3x3 convolutional layers and allows residual mapping to learn identity mapping that might provide reasonable preconditioning for desired underlying mapping. This network further reduces the error rate to 3.57% on the ImageNet database.

The 2D and 3D CNNs have also been transformed into 1D CNN for signal processing or 1D applications. Authors in Kiranyaz et al., 2015 explored compact and adaptive implementation of 1D Convolutional Neural Networks (CNNs) for ECG classification and anomaly detection. In this study, instead of using 2D convolutions, 1D convolution is implemented with only three convolutional layers without fully connected layers, which reduces the computation burden. This approach has achieved the highest performance on MIT/BIH arrhythmia database. Another 1D CNN architecture is introduced to detect damage in bearings in W. Zhang et al., 2018. Authors used single as well as an ensemble of 1D CNNs to not only detect but also localize and quantify bearing faults without denoising preprocessing. The architecture includes six large convolutional layers followed by two fully connected layers and uses data augmentation to have noise invariant features.

CNNs have been widely used in medical research since their ability to achieve expert-level performance. Gulshan et al., 2016 used a deep convolutional neural network and trained on retinal images for diabetic retinopathy screening. The architecture used in this study is

Inception-v3 architecture. Numerous studies in the medical field such as Yasaka et al., 2018, Bejnordi et al., 2017, Mohana et al., 2022, and Lakhani and Sundaram, 2017 demonstrated the use of deep CNN models for differentiating different types of liver masses in CT images, detecting lymph node metastases in tissue sections of women with breast cancer in whole-slide pathology images, detecting the presence of arrhythmia and heart failure in ECG signal collected directly from the IoT devices, and classifying tuberculosis (TB) on chest radiographs respectively. These CNN models have been also utilized to segment cardiac MR images into left and right ventricular cavities and myocardium in Baumgartner et al., 2017. In this work, 2D U-Net and 3D U-Net CNN-based architectures are investigated along with the use of batch normalization. The performance of 3D U-Net was lower than 2D U-Net due to a reduction in training data size, loss of information due to less resolution, and significant downsampling requirements. Different variants of U-Net CNN based have been explored in many publications such as Milletari, Navab, and Ahmadi, 2016, Gordienko et al., 2018, Zhou et al., 2018, Ibtehaz and Rahman, 2020, and Lou, Guan, and Loew, 2021 for segmentation tasks.

1.2.2 Long Short-Term Memory and ConvGRU

To overcome the problem of gradient vanishing and exploding in Recurrent neural network (RNN) when trained on longer dependencies, Long short-term memory (LSTM) was designed and first introduced in Hochreiter and Schmidhuber, 1997. This was possible since it enforces the constant error flow using a Multiplicative gate, namely the input gate and output gate, forming the memory cell units which learns to open and close the access to these errors. Since then, several LSTM variants have been proposed. The most commonly used vanilla LSTM (Gers, Schmidhuber, and Cummins, 2000) is the same as the original LSTM, except for the addition of an adaptive forget gate. The forget allows resetting the cell at the appropriate time i.e. when the information is irrelevant, which ultimately reset the memory cell. The

vanilla LSTM consists of three gates, namely input, forget, output, and a single cell i.e. the Constant Error Carousel (CEC).

Researchers explored eight different variants of vanilla LSTM for different applications. The one modification for the vanilla LSTM structure is to include peephole connections. The peephole connections (Gers and Schmidhuber, 2000) are added from the internal cell CEC to the gates to learn precise time intervals between events i.e. the size of time intervals. Graves and Schmidhuber, 2005 presented the use of bidirectional LSTM with two hidden LSTM layers and full backpropagation through time (BPTT) optimization algorithm. Just like bi-directional RNN, in bidirectional LSTM, training series in the backward and forward directions are provided to two separate LSTMs, and they are connected to the same output. This means complete sequential information before and after all data points are used. Furthermore, the authors suggest that bidirectional LSTMs are significantly more effective than unidirectional LSTMs, where context plays an important role. Another variant of LSTM is Gated Recurrent Unit (GRU) which is employed by Cho et al., 2014. The author simplified the architecture of LSTM by combining the input and the forget gate into an update gate. Hence these networks do not have memory cells. A reset gate is used instead of an output gate, which allows relevant information to store and generate the next sequence. This means it determines the importance of the hidden state. The authors also removed peephole connections and output activation functions. This leads to fewer parameters and, therefore, lesser memory which makes GRU faster than LSTM.

Since LSTM does not capture spatial information, convolutional LSTM (ConvLSTM) was developed by X. Shi et al., 2015 to capture spatiotemporal information. These models contain convolutional operations in the recurrent connections. This convolution operator is used in both the input-to-state and state-to-state transitions to calculate the future state of certain cells based on inputs and past states of its local neighbors. Authors used this model to predict the future rainfall intensity for a short period of time in a local region since this model can handle spatiotemporal sequence forecasting due to the addition of a convolutional structure.

Similar to ConvLSTM models, convolutional gated recurrent unit (ConvGRU) (Ballas et al., 2015, Tianqi Ma et al., 2020) combines convolution operation with GRU model to maintain spatial information of input while extracting spatial-temporal patterns. The architectures used in these studies consist of deep CNN such as VGG-16, AlexNet along with ConvGRU. The placement of ConvGRU in the architecture depends upon the complexity and type of task. The authors used these models for Human Action Recognition and Video Captioning tasks. ConvGRU has lesser parameters and gating mechanisms compared to ConvLSTM.

Due to the impressive learning abilities of LSTM and ConvGRU networks, many researchers have applied them in different domains. Sagheer and Kotb, 2019 developed a deep network by stacking LSTM layers in a hierarchical fashion for predicting petroleum production. The authors used a genetic algorithm to find optimal parameter values. They compared LSTM with Vanilla RNN, deep GRU, Nonlinear Extension for linear Arps decline model, and Higher-Order Neural Network and found that deep LSTM models outperform the rest approaches for long interval time data. Another example of the use of LSTM is in nuclear power plants for fault diagnosis (H. A. Saeed et al., 2020). For Full fault diagnosis, authors combined LSTM and CNN output using statistical analysis. Additionally, the interactive process involved with this model also eliminates the possibility of misdiagnosis by the system. LSTMs have also been used in natural language processing, such as (Sukhbaatar, Weston, Fergus, et al., 2015) to perform automatic system responses, (Z. Wang et al., 2019) generate responses for dialog systems, and (Sutskever, Vinyals, and Le, 2014) for translation tasks. LSTMs are also found helpful in the medical domain to assess small bowel motility by marking small bowel images to the corresponding diameters (Pei et al., 2017), to predict ischemic stroke recurrence using an adaptive particle swarm optimization (Q. Li et al., 2022), and to detect benign epilepsy with spinous waves in the central temporal region (BECT) using EEG spikes data (Z. Xu et al., 2021).

Most commonly, ConvGRUs are used in video and image-based applications due to their less memory and low computation cost requirement. Video anomaly detection using a hy-

brid model of ConvGRU and 3DCNN was introduced by X. Wang, Xie, and J. Song, 2018 to learn appearance and motion features. Bidirectional ConvGRU is implemented to capture global spatial and temporal features in the space-time dimension, whereas 3DCNN is used to capture local spatial information. To prevent the overfitting problem, transfer learning for 3DCNN is performed. The authors also used adjacent video frame velocity loss to learn robust temporal features along with reconstruction and prediction errors. Another example of ConvGRU is video classification (Linchao Zhu et al., 2020). Since the processing of smaller clips independently and then aggregating small clip-level predictions requires high computation, a fast version of ConvGRU is introduced. The ConvGRU can learn the integration of multiple spatio-temporal representations. The fast version also includes bottleneck structure in gating using $1 \times 1 \times 1$ convolution followed by ReLU, similar to ResNet. Multi-layered ConvGRU model is used in geo-spatio-temporal domains to predict crowd density (Zonoozi et al., 2018). This model captures the spatial and temporal correlations and preserves the periodic characteristics by dynamically saving them in a memory-based dictionary after a forward pass through these ConvGRU layers. The last step used in this model is to combine these periodic representations with the current output of ConvGRU by using a weighting-based fusion strategy. The weighting assists in indicating the importance of periodic representations to the current context. In order to reduce the computational cost, a pyramidal structure is used for multi-layer ConvGRU where lower layer outputs are concatenated before passing to higher layers. This will help to speed up learning without impacting performance. Another application of ConvGRU is the classification of abdominal adhesions on sagittal cine-MRI data (De Wilde, Broek, and Huisman, 2021). Authors developed a hybrid model consisting of ResNet followed by ConvGRU model, which improves the classification performance than standalone ResNet while adding only 5% of additional parameters due to ConvGRU. ConvGRU helps to aggregate temporal information while preserving spatial information in features.

1.2.3 Graph Neural Networks

The recursive neural networks were first introduced in Sperduti and Starita, 1997 and Frasconi, Gori, and Sperduti, 1998 for complex pattern recognition using directed acyclic graphs. The early studies of GNN (Gori, Monfardini, and Scarselli, 2005 and Scarselli et al., 2008) are based on learning node' representation using neighbor information in an iterative fashion. Advancement in CNN (LeCun and Cortes, 2010) and graph representation learning (W. L. Hamilton, R. Ying, and Leskovec, 2017) leads to the discovery of GNN. To address the issue of hindering non-Euclidean domains such as Graph and manifolds in CNN, GNN based on geometric deep learning has been introduced (Bronstein et al., 2017). The use of convolutional filters in the spectral domain using CNN on graphs was first proposed in the literature by Bruna et al., 2013. In this paper, two construction ideas for CNN were explored, namely, the spatial construction, which uses spatial convolutional structure and defines locally connected and pooling layers, and spectral construction, which performs convolution in the Fourier domain. The graph convolution in the spectral domain approach faces a problem of significant computational cost, whereas the graph convolution in the spatial domain faces the challenge of matching local neighborhoods.

The computational issue was addressed in Defferrard, Bresson, and Vandergheynst, 2016 by providing efficient numerical schemes which help to design fast localized convolutional filters on graphs. The spectral convolutional filtering approach in this paper consists of filter parametrization, which is K-polynomial filters using Chebyshev expansion instead of explicitly using the graph Fourier-based approach. This allows for performing localized filtering and providing a computationally efficient model. Authors have also used coarsening phase of the Graclus multilevel clustering algorithm to rearrange graph signals with pooling operation, which ultimately makes operation memory efficient and captures the hierarchical structure of the graphs. There have been many variants, improvements, and advancements for spectral-based approaches in GNN using different training algorithms. The main draw-

back of Fourier transform-based spectral GNN methods is it requires prior knowledge of the input graph structure. To overcome this drawback, Henaff, Bruna, and LeCun, 2015 proposed spectral graph convolutions by estimating the similarities in the data. The robust similarity weight matrix is estimated using a z-score, square correlation, and mutual information. Both supervised and unsupervised graph estimation approaches have been explored in this study, and the authors found that the former performs significantly better than the latter. Different training algorithms such as semi-supervised and stochastic training of graph convolution network (GCN) have been used in Kipf and Welling, 2016 and Jianfei Chen, J. Zhu, and L. Song, 2017 respectively. In order to train with large and dense graphs, FastGCN (Jie Chen, Tengfei Ma, and Xiao, 2018) is introduced, which considers spectral graph convolutions to be integral transforms of embedding functions under probability measures. This allows using the Monte Carlo estimator of the original convolution to reduce the computational burden.

The spatial-based approaches used the graph convolution operation as it is to generalize the different combinations of the graph signal within nodes and to define the learnable filters. MoNet (Monti et al., 2017) and GraphSAGE (W. Hamilton, Z. Ying, and Leskovec, 2017) use graph convolution in the spatial domain. MoNet is used as a generic spatial domain framework of mixture model networks. In this approach, instead of using fixed patches, a patch operator based on a parametric approach is proposed. Thus spatial convolutions operation is based on patch operator and defined using a template-matching procedure. Whereas GraphSage is an extension of spatial graph convolution that accounts for generating node embeddings by sampling and aggregating features from the local neighborhood of the node. Three different aggregator functions such as mean aggregator, LSTM aggregator, and Max-Pooling aggregator are trained to aggregate feature information from a node's local neighborhood. Among these, LSTM and pool-based aggregators performed the best. The authors also reduced the training runtime by sampling node neighborhoods.

GNN has been used in different applications such as natural language processing, medical

domains, recommender systems, and traffic forecasting and has been implemented successfully in different domains. But there is very few existing articles related to GNNs in causality. Authors in Phu and Nguyen, 2021 proposed GCN to predict causality between events by learning document context-augmented representations. This approach consists of a document encoder to convert words into representation vectors, an interaction graph generator, and Representation Regularization to regularize the representation vectors. In this work, GCN helps to learn abstract representation vectors for the nodes for causality prediction.

1.2.4 Bias Mitigation Approaches

Most machine learning and deep learning applications have a direct impact on our daily lives. A list of such machine learning applications is included in Howard and Borenstein, 2018, which has biases such as racial bias in face recognition applications, gender bias in voice recognition, social biases and stereotypes biases in search engine applications, and racial bias in justice system applications. IBM and Microsoft facial classifiers were biased towards race since they performed worst for darker females and performed best for lighter individuals and males overall (Buolamwini and Gebru, 2018). This study motivates the urgency for bias mitigation techniques, and several techniques have been introduced to mitigate different types of biases for different applications.

These techniques are divided into three categories such as pre-processing, in-processing, and post-processing (Mehrabi et al., 2021). Methods that manipulate or modify the training data come under pre-processing. The previous work (Kamiran and Calders, 2012) that pre-processes data includes massaging, reweighing, and resampling training data. The Massaging of the data is based on changing the class labels, and it used a combination of ranker and learner, whereas reweighing is weights are assigned w.r.t. bias/sensitive attribute and resampling is the sampling of the data with a replacement which creates four different groups such as DP (A deprived community with Positive class labels), DN (A deprived community

with Negative class labels), FP (A favored community with Positive class labels), and FN (A favored community with Negative class labels). Another study (Hajian and Domingo-Ferrer, 2012) introduced direct discrimination, which is based on sensitive attributes, and indirect discrimination, which is based on nonsensitive attributes that are strongly correlated with biased sensitive attributes. In this study, the authors modified the data points by measuring discrimination and identifying categories. These transformations are defined by using discriminatory rules such as changing the discriminatory values in some records and changing the class labels in some records. More advanced studies such as augmenting training data (Sharma et al., 2020) and learning fairness representations by modifying feature representations (Calmon et al., 2017) were also used so that distributions for both privileged and unprivileged groups become similar.

In order to mitigate the bias during training time, the in-processing algorithm in Kamishima et al., 2012 used the regularization approach and identified the causes of fairness, such as prejudice which refers to statistical dependence between sensitive/bias-related information and Target or non-bias-related information, underestimation refers to a non-convergence state, and negative legacy refers to the problems of unfair sampling or labeling. In this study, the addition of a bias removal regularizer in the objective function enforces independence between sensitive/bias-related information and classifier predictions by penalizing the mutual information. Quadrianto and Sharmanska, 2017 developed two techniques such as privileged learning and conditional distribution matching, to improve the performance when privileged information is only available at training but not at testing time. For the privileged learning model for achieving fairness, authors use protected/bias-related characteristics as privileged information. They also add fair impact and/or fair supervised performance constraints into the privileged learning model in order to prevent the risk of unfairness by proxy. This is achieved by using a distribution matching framework which ultimately leads to matching positive predictions, matching true positive rates, and matching false positive rates across the two demographics. The maximum Mean Discrepancy (MMD) metric is used to match

both these two privileged and unprivileged group distributions. J. Zhao et al., 2017 introduce the Reducing Bias Amplification (RBA) technique for bias mitigation. Authors suggest the use of corpus-level constraints in structured prediction models and an algorithm based on Lagrangian relaxation to reduce gender bias. The constraint makes sure that the bias ratio of each activity is within a given margin based on the statistics of the training data, whereas the Lagrangian relaxation algorithm helps to solve inference problem with constraints and ensures optimal solution if the algorithm converges. B. H. Zhang, Lemoine, and Mitchell, 2018 proposed an adversarial learning approach to mitigate biases such as zip code and gender. This learning referred to as adversarial debiasing, consists of two models; one is to predict the main task, and the other is to predict bias. So the objective of the model is to maximize the ability to predict the main task while minimizing the ability to predict bias/protected variable. Although this method is successful, it still produced somewhat biased results and is unstable.

Post-processing techniques are used on output results to make fairer decisions. Hardt, Price, and Srebro, 2016 recommends that the burden of uncertainty in classification should be shifted from the sensitive/protected/bias variable to the decision maker. But to use this method, it is important to have an unbiased output/target variable. To enhance equalized odds and equalized opportunity, constraints and thresholds for predicting scores using ROC (Receiver Operator Characteristic) curve for the different protected groups are implemented on the outcome of the classifier to ensure equal bias and equal accuracy in all demographics. The decoupling technique introduced in Dwork et al., 2018 is another post-processing technique where a separate classifier is trained on each group, and the joint loss function is used to penalize differences in classification statistics between groups which helps to capture fairness. Transfer learning is also introduced to enable training on a small-size dataset for mitigating bias. The Multiaccuracy Boost model in M. P. Kim, Ghorbani, and Zou, 2019 is used to classify subpopulations where the original model is biased. The original biased model is post-processed iteratively until unbiased predictions in each subgroup are achieved. The

post-processing algorithm is similar to gradient boosting, which uses a multiplicative weights framework to enhance predictions for subgroups and the addition of a "do-no-harm" guarantee. The "do-no-harm" makes sure that classification error does not increase drastically from the original classifier to the post-processed classifier.

It is very crucial to understand the challenges due to bias in deep learning and identify and mitigate those biases in the medical domain. A CNN-based approach called Skin Image Search (Kamulegeya et al., 2019) was used to classify skin lesions. But for images of Black patients, diagnostic accuracy reduces significantly, indicating biased toward white patients and not capturing patterns specific to the black patient. The authors suggest using diversity in the image dataset when training CNNs by including different skin complexities, and this will help to reduce the bias. Meyer et al., 2021 use intensity-based augmentation approach based on Gaussian Mixture Models (GMM-DA) to mitigate bias induced by multi-scanners. This method randomly modifies the individual tissue components of an MRI image while preserving structural information present in MRI. The main limitation of this method is that it is restricted to variations only in intensity and does not capture other distortions introduced by multi-scanners. Another limitation is that the representation is based on only Gaussian distributions.

1.2.5 Distance Correlation-Based Methods

Recently distance correlation (Székely, Rizzo, and Bakirov, 2007) has been explored in a few publications to achieve different goals. Vepakomma, Tonde, and Elgammal, 2018 proposed use of Statistical Distance Correlation for supervised dimensionality reduction. The authors suggest the addition of two Laplacian-based sample distance correlation to the objective function to measure dependencies between low-dimensional features and outputs/response variable and between low-dimensional features and inputs/covariates. The goal is to maximize the sum of squares of these two sample distance correlations by using the Gener-

alized Minorization-Maximization (G-MM) optimization algorithm. As this optimization method solves multiple optimization subproblems iteratively, it becomes difficult to deal with larger datasets. Also, this work is not able to learn an explicit mapping from input to low-dimensional features for new data points.

Another application of using distance correlation in regression and support vector machine setting is for feature selection (R. Li, Zhong, and Liping Zhu, 2012, Kong, S. Wang, and Wahba, 2015). Authors ranked the importance of the variables using distance correlations with the response in decreasing order (R. Li, Zhong, and Liping Zhu, 2012) and Kong, S. Wang, and Wahba, 2015 further improve the performance by adding variables until the distance covariance between variable and response does not decrease. These methods face a problem when dealing with an enormous number of variables and a small sample size. Authors recommend the use of stopping criteria for variable selection procedure and double greedy variable selection algorithm to tackle these difficulties. Two genetic risk problems in small round blue cell tumors and Ovarian cancer were studied for gene selection using a distance correlation-based variable selection approach.

Causal inference based on distance correlation in the discrete domain was proposed in Liu and L. Chan, 2016. Causal direction is determined based on distance correlation between the distribution of the cause and the conditional distribution mapping cause to effect. Pearson correlations are not used in this work since it would cause estimation bias when the sample size is not large. In this work, the direction that induces a smaller distance correlation value is inferred as the causal direction, while a larger value indicates the anti-causal direction. If variable x causes y, then the distance correlation between x and y should be smaller than between y and x. This indicates the smaller dependence coefficient inferred as the causal direction. The main drawback of this method is that it is not capable of capturing complex causal inferences present in high-dimensional data.

Distance correlation has been used in the deep learning domain for supervised dimensionality reduction in R. Wang, A.-H. Karimi, and Ghodsi, 2018. The authors used distance

correlation in the objective function of autoencoders. Maximizing the distance correlation is implemented by minimizing the negative log of distance correlation, and it allows the extraction of low-dimension features that have maximum linear as well as non-linear association with the output/response variable. The addition of the distance correlation improves the encoding capability of reconstructing the data point from its low dimension features by providing a good representation. LSTM and CNN networks were implemented for autoencoder using a stochastic gradient descent optimization algorithm, and a hyperparameter is used to denote the weight of the reconstruction loss. This method is not useful in the healthcare domain, where data sharing with privacy is a top most important aspect.

In order to prevent the reconstruction of raw data to avoid sensitivity and privacy issues, Vepakomma, Gupta, et al., 2019 proposes a framework that minimizes the reconstruction of raw data by minimizing the distance correlation measure between raw data and split layer features. The main goal of this study is to use distributed deep learning models for multimodal health data without sharing raw data and without affecting its performance. The authors used the split learning configuration where each client uses a partial deep network until the split layer to create features. Then these split layer features are propagated to another client/server to perform the rest of the training without looking at the original raw training dataset. The backpropagation is also performed in the same fashion but in a reversed direction. Most of the networks faced a data leakage issue at the split layer, and to remove this leakage issue from the networks, distance correlation is used. Authors successfully reduce the leakages while maintaining classification accuracy.

1.2.6 Granger Causality Frameworks

The Granger Causality (GC) was first introduced by Granger, 1969 which is based on how well the past and current information of times series predicts or causes the future information of another time series using a series of statistical tests. This is in contrast to other true causal

relations, and it can only be inferred using GC only under some specific conditions. GC has been primarily studied using the popular linear vector autoregressive (VARs) models and considers the only bivariate situation. These methods rely on heavy parameterization for the high-dimensional dataset, which makes them computationally expensive. Also, incorporating higher time series components in VAR models leads to more accurate results than a smaller number of components. To overcome these problems, several methods have been proposed.

Markov-switching (MS) VAR model is proposed by Psaradakis, Ravn, and Sola, 2005 with time-varying parameters to analyze changes in causality relationships over the sample period. This means causality is observed in some periods but not in others. Time-varying parameters are directly related to changes in causality relationships. One variable non-Granger causes another variable in some parts of the sample if the coefficients are zero for the same sample part. Authors assumed that changes in causality are non-constant, not stable, stochastic, and driven by a hidden Markov process. Since authors consider that these changes as random events vary with the hidden Markov process, probabilistic inference about causality can be calculated at each sample point. This allows for identifying the location and types of changes that occurred. However, it also leads to focusing only on linear relationships.

Graphical Granger Modeling is an extension of pairwise-GC where many time series variables are available. Lozano et al., 2009 developed the grouped graphical Granger modeling methods in order to leverage group structures among the lagged temporal data. In this approach, a group of lagged time series is considered in predictor selection. Different techniques for group variable selections such as group Lasso, boosting, Group Boosting, and Adaptive Group Boosting are implemented to improve the performance. These group penalties take into account the average effect of x on y over different periods of time. This helps to scale the GC estimation in VAR to higher dimensional data. However, the problem with this approach is the use of the Linear regression model, which only captures the linear Granger causal effect. Another study (Basu, Shojaie, and Michailidis, 2015) developed a generic group lasso regression regularization framework. This framework is called high-dimensional

Network Granger causality (NGC), which uses multiple time series i.e. more than two variables. Generic group lasso penalty properties exhibit direction consistency and deal with variable selection within groups while correctly estimating the sign of all the variables. The variant of group lasso penalties is a thresholded variant of group lasso which consistently learns the sparsity patterns within the group in addition to group level variable selection and corrects the misspecification in groups.

The approach for automatic lag selection was proposed by Nicholson, Bien, and Matteson, 2014 and Shojaie and Michailidis, 2010 is to use a hierarchical group lasso penalty and truncating penalties, respectively. A computationally efficient approach based on a truncating group lasso penalty not only automatically detects both nonlinear Granger causality and the lags of each inferred interaction but also provides forecasting for higher dimensional VAR models. The regular group lasso penalty does not provide information about the magnitude and sign of the GC interaction but truncating the lasso penalty addresses these issues by decreasing the number of effects as the time lag increases, which ultimately results in fewer effects in the estimates and forcing other remaining estimates to zero. The hierarchical group lasso is similar to truncating lasso, except it is calculated using a nested group structure and is convex. This nested structure leads to sparsity patterns that maintain the ordered structure inherent to VAR. This implies that one set of parameters being zero might lead to another set of parameters being zero, resulting in a more computationally efficient implementation.

Generalized Autoregressive Conditional Heteroskedasticity (GARCH) is another model that was utilized in Woźniak, 2015 to model the risk associated with financial time series and to analyze the GC interactions for the second conditional moments. The standard Bayesian approach is employed to calculate posterior odds ratios to test the second-order granger non-causality hypothesis. This test assumes that Granger causal interactions might present in the conditional mean process, which needs to be removed. The main limitation is that this method considers only two variables and analyzes all the second-order non-causality at

once for all future values.

Structural vector autoregressive (SVAR) model is a model used to estimate Granger causality under subsampled and mixed-frequency time series settings (Tank, Fox, and Shojaie, 2019). The authors demonstrate that structural vector autoregressive models assist in the identification of lagged Granger causality as well as instantaneous structural interactions under arbitrary subsampling and mixed-frequency conditions. Under the subsampling setting, instantaneous causal effects follow a directed acyclic graph without any prior information about the causal ordering of the variables, whereas the mixed-frequency settings leverage non-Gaussianity to provide causal ordering. A mixture of Gaussian distributions with a definite number of components is used to model the non-Gaussian errors. These errors help in detecting parameters that are difficult to identify from the first two moments since the non-Gaussianity of the structural model capture more realistic properties such as asymmetry, heavy tails, or stochastic volatility with subsampling or mixed frequencies. The authors also proposed an exact expectation-maximization algorithm for joint maximum likelihood estimation of parameters using both subsampled and mixed-frequency series. Two obstacles are observed while using this model. One is high complexity due to the use of the Kalman filter for computing mixture errors in the expectation-maximization algorithm, and the second is getting stuck in multiple local optima, which leads to a poor solution.

1.3 Objectives

The main theme of the research work presented in this dissertation is using the distance correlation function in deep learning for bias mitigation and learning Granger causal relationships. The main goal is to learn bias invariant features using deep neural networks on different types of datasets and to learn Granger causal interactions in order to improve the performance of deep neural networks. Nowadays, researchers are becoming more aware of biases present in different real-world applications and datasets. There are different sources

of these biases that can affect even state-of-the-art methods in machine learning and deep learning-based applications. Examples of different types of biases include data biases, algorithmic biases, and user-induced biases. Data biases arise due to biases that distort the representation of the dataset, such as measurement bias, exclusion bias, sampling bias, and representation bias (missing or under-represented group) (Suresh and Guttag, 2019). As the name suggests, algorithmic biases are biases introduced by the algorithms only, such as evaluation bias which is happened only during model evaluation (Suresh and Guttag, 2019). User-induced/Human biases are biases that are introduced by users during the data generation processes, such as population bias, social bias, Behavioral Biases, and content production bias (Olteanu et al., 2019).

Although Deep Learning algorithms have been developed rapidly, their use in a wide range of applications is resulting in a growing demand for algorithms that are robust, reliable, and generalized. Deep learning models are prone to biased decisions due to the presence of biases in the data. Since deep learning learns directly from the data, data quality and quantity are very important in terms of their robustness and fairness/unbiased results. There are different types of studies where authors identified different representation biases (Shankar et al., 2017) in two popular public data sets, ImageNet and Open Images and data biases such as age and race biases (Adeli et al., 2021) in Gender Shades Pilot Parliaments Benchmark (GS-PPB) dataset. Since these biases lead to erroneous decisions, they can be more dangerous to sensitive applications such as the health care domain. For example, researchers in the paper Fry et al., 2017 found the UK Biobank dataset that was collected to study middle and later-life diseases is not representative of the sampling population and provided evidence for "healthy volunteer" selection bias. Another example includes race/ethnic bias present in a largely white population data from the Framingham Heart study, which impacts the algorithm used to predict cardiovascular risk for Asian, Hispanic, and African American patients. Thus, developing deep learning models that can successfully learn bias invariant features is a promising area of research and has gained significant importance with high

potential impact.

We recognize our dissertation work can be divided into two broad areas of study. The first one includes the design and analysis of decorrelated deep learning architectures. It also involves generalizing the decorrelation notion across different domains such as medical, computer vision, and finance and using different biases such as class, scanner, color, gender, and age bias. The second area of study deals with the main limitation of the deep learning model i.e. learning by association which also causes discrimination and biases problems in deep learning algorithms. This study includes the first step toward providing an advanced solution that is based on Granger causation in the deep learning domain and developing a fusion deep learning model to estimate Granger Causal interactions in the data.

There are several objectives associated with this research work which are as follow:

The first is to develop a deep learning methodology to distinguish PD from the control group with the highest specificity and sensitivity by using neuroimaging data such as rs-fMRI along with patient information. PD is a very important area of study since there is an unknown factor in the cause of PD, and research is still going on to understand the mechanism underlying cognitive impairment for PD, which remains unclear. There have been very few studies conducted on neuroimaging data for the PD community. Most of them are driven by hypotheses and hand-crafted feature extraction methods which are based on pathology-related background knowledge. It is difficult to detect PD in a normal MRI scan since structural changes in the brain for early PD may not appear on MRI due to a subtle change. But rs-fMRI can help reveal structural and neural connectivity to understand the pathology of cognitive impairment in PD better. In addition to this, rs-fMRI is found to be easy to use and safer compared to other neuroimaging modalities, and hence its application in understanding the human brain structure and cognitive impairment has been increasing in recent past years. The main advantages of using resting-state functional magnetic resonance imaging (rs-fMRI) are it is a fast-developing research field, reveals the pathophysiology of cognitive symptoms in PD, facilitates early identification of PD patients with cognitive im-

pairment, and overcomes problems associated with other neuroimaging data such as lack of patterns, high cost, longer time period capture issue, etc. The main three challenges in Parkison's disease diagnosis are; one 30% misdiagnosis rate which is due to the fact that there is no precise test for Parkinson's, and different diagnosing doctors treat various indicators differently. For instance, none of the diagnostic tests are definitive, and so it is entirely up to the doctors how to review these tests to decide about diagnosis and medication or even which one to use to make a diagnosis. The second challenge is that PD varies from patient to patient, and symptoms overlap with other medical conditions. Hence, diagnosing diseases based on the diagnostic test and radiologists' reading is oftentimes prone to mistakes. It is still difficult to make an accurate prediction of PD. PD diagnosis is not easy to make because the cause of Parkinson's is unknown, and there are no proven ways to cure or avoid this disease. The third challenge is that generally, neurologists who are experts and familiar with this disease will most likely take a longer period to make this diagnosis. In addition to this, a whole range of other neurological disorders can have many of the same symptoms as Parkinson's, and as a result of this, a neurologist needs to be very careful and thorough in eradicating or excluding some of those other neurological disorders. In this scenario, a deep learning approach is an ideal approach since this approach not only overcome all three main challenges and makes accurate predictions but also is the quickest way to diagnose PD.

The second objective is an extension of the first one, which is to define a novel bias mitigation technique and construct these deep learning models for PD detection by using a single scanner and multi-scanner rs-fMRI datasets and to provide class and scanner invariant features without compromising the performance of the model. This will not only help to increase the dataset size but also build more robust and generalizable models. In order to do this, it will first be necessary to generate the bias mitigation concept and define a mathematical framework that can provide bias removal while simultaneously maintaining performance on the main task of interest.

Most of the novel techniques will work only on a particular set of data or in a particular

scenario. Hence, it is crucial to show that technique is generic, flexible, and applicable to a wide range of scenarios and fields. The third objective is to explore new application domains and new biases that will help to generalize the same novel bias mitigation concept beyond the medical domain. This will indicate that the same mathematical framework can be extended for the removal of different types of biases and can be adapted to many different data scenarios. It will help to show the flexibility and suitability of bias mitigation techniques for different applications in various fields and for different bias mitigations. It will also lay the groundwork for the use of bias mitigation techniques for different architectures of deep learning models and for analysis.

There are many deep learning applications, but all of them are based on association relationships present in the data. But we do not yet know how to introduce causality in deep learning. Causal discoveries will not only help to improve performance but also help to construct more interpretable and explainable systems. Thus, the last objective is to learn complex non-linear Granger causality (GC) interactions in temporal data using the integration of deep learning model and the mathematical component of the bias mitigation concept. This will also need to define a mathematical framework to introduce the fusion of GC with Graph Neural Network (GNN) and distance correlation.

Our main contributions in this dissertation are:

- We propose the use of rs-fMRI and patient data using a convolutional neural network and convolutional-gated recurrent unit-convolutional neural network (ConvGRU-CNN) for Parkinson's Disease (PD) recognition. This proposed approach is implemented for the first time. This approach emphasizes the ability of two different networks to process rs-fMRI neuroimaging data differently. We develop the ConvGRU-CNN model for processing temporal information first. Please refer to chapter 2

- We develop a framework based on the novel decorrelation loss function concept for deep learning models for PD classification to address not only imbalanced dataset is-

sues but also scanner dependencies issues. We explore different approaches to mitigate class and scanner biases using the decorrelation-based bias mitigation technique. These different approaches and deep learning architecture reduces the biases while still performing well for the PD classification task. In conjunction with the first contribution, we demonstrate how our proposed DcCNN models can produce class and scanner invariant feature representations while still completing the PD recognition task across different scanner acquisition protocols and majority class bias without compromising the performance. Please refer to chapter 2.

- We exploit a novel decorrelation loss function-based mathematical general framework for ANN, CNN, and DNN models to decorrelate bias from the learned features, to address bias issues. The introduction of a new loss function not only mitigates the biases but also helps to improve the performance of models. In particular, we generalize the idea of decorrelation function across five different domains and biases. Adopting the decorrelation-based bias mitigation concept to different data scenarios and different deep learning architectures shows the flexibility, scalability, and generalization ability of the framework. Furthermore, comparing our proposed DcDNN and DcANN methods to existing bias mitigation methods provides insights into the advantages of using our proposed models as opposed to existing models. Please refer to Chapter 3.

- We introduce a novel use of distance correlation and Graph Neural Network along with deep learning to estimate GC, which we refer to as Graph Granger Causality (GGC). In particular, we incorporate nonlinearities hidden in the data as weight initializers and model penalties using the mathematical component of the decorrelation function i.e. distance correlation. We extend the existing LSTM Granger Causality framework to GNN to account for Granger causal interactions that are present in transcriptomal time series dataset. We suggest a GGC framework that provides interpretable nonlinear Granger causality discovery. Please refer to Chapter 4.

1.4 Outline of Dissertation

In the next chapter i.e. Chapter 2, we discuss a methodology used to perform the classification o f P arkinson's d isease b ased o n t he n ovel d ecorrelated C onvolutional N eural Network(DcCNN). We introduce a problem of imbalanced and scanner-biased datasets and define a n ovel d ecorrelation a pproach t o m itigate c lass b ias a nd s canner b ias w hile simultaneously not impacting the process of distinguishing characteristics in resting-state functional MRI (rs-fMRI) data. Finally, we also describe the results of this proposed DcCNN deep learning method for Parkinson's disease recognition and show how the decorrelation function mitigates both biases. The work presented in chapter 2 was under review in the journal under the title "Decorrelated Convolutional Neural Networks for Parkinson's Disease Recognition using Imbalanced and Scanner-biased rs-fMRI Data" and will be published in the scientific journal, with Dr. Rand Ford as a collaborator.

In Chapter 3, the main focus is on generalizing the bias mitigation approach across different domains and for various types of biases. We propose the same decorrelation function used in chapter 1 with slight modifications. We present different deep learning and artificial neural network architectures along with decorrelation functions with which we can mitigate different types of biases for different applications while achieving high performance. We also compare the results with existing bias mitigation methods. We show how our proposed novel approach is simple, effective, and flexible and should be applicable to a wide range of applications. The work presented in Chapter 3 was accepted in the MDPI future internet 2022 journal under the title "Decorrelation-Based Deep Learning for Bias Mitigation" and was published in the scientific journal, with Dr. Kevin Purcell as a collaborator.

In chapter 4, we extend the idea of using distance correlation in the form of the objective function and weight initialization for the model. We consider the use of deep learning architectures in learning Granger causality interactions. We propose a novel GGC model, which is a fusion of graphs convolutions, LSTM, and nonlinear penalties for the objective of

learning Granger causal relationships among temporal elements in gene regulatory networks. Then, we validate the performance of our fusion model on a simulated dataset. We also study the novel use of distance correlation and graph convolutions to boost the performance of the model by capturing the true GC structure behind the time-series data. The work presented in Chapter 4 was presented at the Proceedings of the 14th International Conference on Bioinformatics and Computational Biology (BICOB 2022), and the paper titled "Learning Gene Regulatory Networks using Graph Granger Causality" was published in the conference proceedings, with Maria Vaida as a collaborator.

Finally, last chapter 5 will conclude this dissertation work and provide some perspectives for future work. We also highlight the main findings and discuss their impact.

Chapter Two

Decorrelated Convolutional Neural Networks for Parkinson's Disease Recognition using rs-fMRI Data: Learning Class Bias Invariant and Scanner Independent features

2.1 Abstract

Parkinson's Disease (PD) is a neurodegenerative and progressive disease that impacts the nerve cells in the brain and varies from person to person. The exact cause of PD is still unknown, and the diagnosis of PD does not include a specific objective test with certainty. Although deep learning has made great progress in medical neuroimaging analysis, these methods are very susceptible to biases present in neuroimaging datasets. An innovative decorrelated deep learning technique is introduced to mitigate class bias and scanner bias while simultaneously focusing on finding distinguishing characteristics in resting-state functional MRI (rs-fMRI) data, which assist in recognizing the PD with good accuracy. The

decorrelation function reduces the non-linear correlation between features and bias in order to learn bias invariant features. The Parkinson's Progression Markers Initiative (PPMI) dataset referred to as a single scanner imbalanced dataset in this study used to validate our method. The imbalanced dataset problem affects the performance of the deep learning framework by overfitting to the majority class. To resolve this problem, we propose a new Decorrelated Convolutional Neural Networks (DcCNN) framework by applying decorrelation-based optimization to Convolutional Neural Networks(CNN). An analysis of evaluation metrics comparisons shows that integrating the decorrelation function boosts the performance of PD recognition by removing class bias. Specifically, our DcCNN model performs significantly better than existing traditional approaches to tackle the imbalance problem. In addition to this, the same framework can be extended to create scanner invariant features without significantly impacting the performance of a model. The obtained dataset is a multi-scanner dataset which leads to scanner bias due to the differences in acquisition protocols and scanner. The multi-scanner dataset is a combination of two datasets namely PPMI and FTLDNI - frontotemporal lobar degeneration neuroimaging initiative (NIFD) dataset. The results of t-distributed stochastic neighbor embedding (t-SNE) and scanner classification accuracy of our proposed Feature Extraction-DcCNN (FE-DcCNN) model validated the effective removal of scanner bias. Our method achieves an average accuracy of 77.80% on a multi-scanner dataset for differentiating PD from healthy control which is superior to the DcCNN model trained on a single scanner imbalanced dataset.

2.2 Introduction

Parkinson's disease (PD) is characterized by the lack of dopamine transmitters due to the degeneration of melanin cells in the pars compacta (posterior part) of the substantia nigra, and PD patients show several cognitive deficits which include executive functioning, visuospatial abilities, and memory loss. The symptoms of PD include shaking, slow movements, walking

problems, behavioral problems, speech problems, etc. Diagnosis of PD generally includes assessment of behavior, neuroimaging, physical, biological sampling, and clinical data. The false-positive rate for PD is higher in the early stage and high at the final diagnostic stage. In the past few years, studies in neuroimaging modalities have provided more profound and valuable insights into the underlying mechanism of PD.

Parkinson's disease remains the second most common neurodegenerative disorder. But still, there is an unknown factor in the cause of PD, which makes PD a very important area of study. Motor symptoms, along with Cognitive impairment, are also found as common disabling symptoms in PD. The mechanism underlying cognitive dysfunction PD remains ambiguous, unlike motor symptoms. Many studies have been conducted on PD using clinical and biomarker data. Most of them are driven by hypotheses and hand-crafted feature extraction methods which are based on pathology-related background knowledge. Recently, neuroimaging is considered an important information source for neurodegenerative disease. Hence, it has also arisen considerable interest from the PD community. Diagnosing Parkinson's disease based on diagnostic tests and radiologists' reading on neuro-images is oftentimes prone to mistakes. So there is a gray area in the PD diagnosing research field where the unknown cause of PD, no precise test for PD, and a high misdiagnosis rate is present. There is a need for highly accurate and reliable results. This research may be of use to the medical community in a screening setting and to understand how and why PD develops and search for solutions to stop or avoid the progression of the disease.

Currently, no specific test exists to diagnose Parkinson's disease. There are few diagnostics tests that Physicians use to diagnose Parkinson's disease based on medical history, review of signs and symptoms, physical examination, blood test, and neuroimaging tests. As PD progresses, it becomes harder to prevent or slow the changes through medication. For this reason, in 2016, experts developed new criteria (Postuma et al., 2016). These include three steps. The first step includes accessing the probability based on the age that the diagnosis will be PD. In the second step, physicians access the information based on variables

such as whether the person is male or female, environmental risks, caffeine use, and smoking, genetic factors, family history, or genetic test. Sometimes findings based on these results of scans and other diagnostic tests show early signs and symptoms, which include constipation, loss of a sense of smell, and difficulty with movement. The third and final step consists of calculating the outcome by multiplying all the factors together and then comparing this total likelihood ratio with a threshold measure. If the comparison indicates a total likelihood ratio higher than 80 percent that PD is present, the physicians will diagnose that patient with the early stages of PD. Most commonly, a patient with a 75–80 percent total likelihood will have symptoms that may or may not relate to PD, e.g., constipation and depression whereas a patient with 95–97 percent total likelihood will have symptoms that are closely related to PD, e.g., Rapid eye movement (REM) sleep behavior disorder where a person experiences sudden and rapid movements and vocalizations during vivid dreams.

Deaths caused by PD have increased significantly over the years. The diagnosis of PD used in hospitals relies mainly on a combination of different diagnostic tests and symptoms assessment. It is still difficult to make an accurate prediction of PD. Neuroimaging data such as Magnetic Resonance Imaging (MRI), Resting-State Functional Magnetic Resonance Imaging (rs-fMRI), Single-photon emission tomography (SPECT), Dopamine transporter imaging (DAT), 123I-ioflupane-SPECT (DaTscan), Diffusion tensor imaging (DTI), A positron emission tomography (PET), Computed tomography (CT) scans can be used to diagnose PD. However, CT scans and MRI images sometimes do not show patterns in images to distinguish PD from a healthy patient. Whereas SPECT is a commonly used method but suffers from high cost and time issues and requires injection of radio-active material. Radiologists generally use one of these neuroimages to diagnose PD disease, but it is proved to be more prone to mistakes. Recent research and studies have shown that DTI and rs-fMRI can be used to predict PD and are found to be promising methods for the diagnosis of PD. But in order to capture DTI images, the patient will have to remain still for a longer period, i.e., half an hour. Since DTI is a relatively new technique, it is difficult to find hospitals equipped

with DTI scanners.

Current existing methodologies such as Rubbert et al., 2019 and Guo, Tinaz, and Dvornek, 2022 do not use rs-fMRI using CNN to detect PD. Therefore, processing of rs-fMRI with a single scanner and multi-scanner settings using CNN techniques to diagnose PD is not yet explored. This novel research study will evaluate the prediction of PD on noninvasive and comparatively less expensive neuroimaging data such as rs-fMRI in a single scanner and multi-scanner settings using a model that uses the convolutional neural network. Since available neuroimaging data is limited and the majority of the data is class imbalanced, this study will also provide a novel decorrelation-based deep learning fusion approach to mitigate class bias. Further, we will also explore the use of multi-scanner rs-fMRI data which is obtained by combining different datasets from different scanners not only to balance the dataset but also to increase the size of the dataset, and to improve the performance of the model. But this leads to an undesirable increase in variance caused by scanner and acquisition protocol differences including scanner upgrade, scanner drift, and gradient nonlinearities. The same framework of decorrelation-based deep learning is used to produce features that are invariant to scanner and acquisition protocol while still capable of not impacting the performance of PD recognition task.

The rest of this paper is structured as follows: Section 2.3 briefly reviews the related work, whereas section 2.4 provides a brief description of the proposed methodologies, involved PD datasets, and preprocessing techniques; Section 2.5 reports the results and comparison with existing methodologies and Section 2.6 discusses the performance of our proposed method for the PD detection. Lastly, section 2.7 concludes the research and provides opportunities for future work.

2.3 Related Work

Two centuries ago, James Parkinson presented the first medical description of Parkinson's disease in 1817. Today, Parkinson's disease is the second most common neurodegenerative disorder. The pathophysiology of Parkinson's disease (PD) is the study of the functional processes that occur in a PD which is only partially understood. Currently, what we know about PD is that the loss of neurons in the Substantia Nigra pars compacta part of the brain and the presence of Lewy bodies leads to the loss of dopamine (a neurotransmitter). This damaged neurotransmitter ultimately prevents normal function in basal ganglia which causes the motor symptoms of PD and cognitive impairment. Common motor symptoms observed in PD include tremors, slowness, stiffness, rigidity, swallowing problems, balance problems, unpredictable movements, difficulty initiating or controlling movement, cramping, and speech problems. Cognitive issues, such as short-term memory loss, difficulty following complicated instructions, or a loss of multitasking ability, may also occur in PD patients. Some people will have several symptoms whereas others will have only a few. It has been observed that deaths caused by PD have increased significantly over the years. This is mainly because PD is difficult to diagnose and can be caused by a combination of environmental, genetic, or lifestyle factors. Male gender, gait disorder, and absent rest tremor are generally associated with poorer long-term survival. According to NIH, approximately 50,000 to 60,000 Americans are diagnosed with PD each year. Because of a lack of knowledge regarding which symptoms develop, and how severely and quickly symptoms develop, and since the symptoms of Parkinson's vary from patient to patient and often overlap with other medical conditions, PD is misdiagnosed up to 30 percent of the time. It has been observed that misdiagnosis of PD is very common. So there is a need for an automated diagnostic tool.

2.3.1 Pathology Driven Hypothesis

In the past few years, several studies have been done to explore the connection between clinical, biological, and imaging data to achieve an accurate diagnosis and early detection of PD. Most of these studies are driven by pathology or the underlying biology of PD and use hypotheses. According to S. Kim et al., 2019 and Braak et al., 2003, the α-Synuclein protein which is a major component of Lewy pathology accumulates and originates from cells in the gut and transmits to the brain via a vagus nerve in the patient with Parkinson's disease. The authors performed this study on a mouse model and supported the Braak hypothesis. This research might help to prevent or halt PD progression by blocking the vagal transmission pathway in an early stage. From a genetic contribution point of view, a paper published by Beilina and Cookson, 2016 suggests that protein products of genes help to identify the functionality of PD whereas El-Agnaf et al., 2006 have investigated the use of α-Synuclein protein as a biomarker for PD using hypothesis testing with around 85% specificity and 52% sensitivity. In the Trivedi et al., 2019 paper, innovative approach such as the use of sebum to diagnose the PD was used since a change in skin microflora and skin physiology can cause a change in odor in PD patients. The results (AUC 78%) to support this theory were achieved by collecting sebum samples from the participant's upper back and using combination of data processing techniques, such as olfactogram and chromatogram, and performing partial least-squares-discriminant analysis on this preprocessed data. The main limitation of this study is the smaller sample size. There are quite a few studies conducted to diagnose PD by using neuroimaging and clinical data.

Several papers such as Son, M. Kim, and Park, 2016, Cochrane and Ebmeier, 2013, Atkinson-Clement et al., 2017, Vaillancourt et al., 2009, Zheng et al., 2014 and U. Saeed et al., 2017 have suggested the use of DTI metrics can provide distinguishing features to detect PD or used as imaging biomarkers for PD. In recent years, there have been studies (Rolinski et al., 2014 and K. Li et al., 2018) in rs-fMRI which is a fast-developing research field

and helps in revealing cognitive dysfunction or increasing motor connectivity for early PD detection. All these studies perform hypothesis testing such as t-test, two-way mixed model ANOVA, comprehensive meta-analysis, etc. to find significant group differences between PD and control healthy groups. The cross-sectional study (H. Wilson et al., 2019) claimed that serotonergic pathology plays a vital early role in the progression of PD. This study provided evidence that loss in serotonin function is observed in the very early stages of PD by using PET and SPECT scans. To access molecular, clinical, and structural pathology, PET imaging was used. ANOVA and t-test were used for comparisons between the groups and suggested that serotonergic malfunction precedes the development of other PD symptoms such as motor and is related to the dopaminergic deficit by using the Braak staging scheme.

2.3.2 Data-Driven Models

Data-driven approaches, such as deep learning and machine learning are different than conventional statistical analyses. DaTscan SPECT image analysis with the one-layer artificial neural network is developed to classify PD versus normal with around 94% accuracy (Y. C. Zhang and Kagen, 2017). Machine learning-based approaches such as a support vector machine (D. Shi et al., 2022), a Naive Bayes classifier (Jiji, Rajesh, and Lakshmi, 2022), and a boosted logistic regression model (Rubbert et al., 2019) were also used for PD classification using rs-fMRI data but it was tested on very small datasets.

To overcome the drawback of feature-engineering or hand-crafted features, a few deep learning techniques have been deployed in the past decade. Choi et al., 2017 used SPECT data to detect PD over normal using deep 3D CNN architecture which achieved around 96% far higher than human evaluation accuracy and could be used for the SWEDD group. Another study in deep learning is carried out by X. Zhang et al., 2018 used graph convolutional deep networks (GCN) to fuse multiple modalities of MRI and DTI to detect PD cases and achieved around 95% AUC. In this study, a Brain Geometry graph (BGG) is obtained from

the Region of Interest of MRI and Brain Connectivity Graphs (BCGs) from the tractography of DTI and used as input to GCN to explore spatial and frequency spectrum information. Laplacian and Fourier transform-based graph convolution are performed on BGG and BCGs, and then the multi-view pooling is done to aggregate multi-view outputs of GCNs together. The authors also used pairwise matching between outputs of multi-view GCN to increase the amount of data. In the final step, a fully connected softmax network is used for classification by using pairwise matching layer output. Esmaeilzadeh, Yang, and Adeli, 2018 performed PD diagnosis using 3D Convolutional Neural Network(3D CNN) deep learning framework on 3D MRI and patient personal information such as age and gender. This work is primarily compared with Ahmed and Farag, 1997 and Gil and Manuel, 2009 work for performance comparison. The main goal of this pilot study is to integrate feature extraction and model learning into one framework to improve performance. Skull stripping by using the Brain Extraction Technique (BET) with Statistical Parametric Mapping (SPM) algorithms was used to remove non-cerebral tissue in order to improve the speed and accuracy of this study. Flipping of the right and left hemispheres was done in the data augmentation process. In their study, the authors claimed that using age alone in logistic regression to predict PD achieved 72% accuracy. The authors also performed image occlusion analysis to study important parts of the brain in PD diagnosis and suggested those parts are Basal Ganglia and Substantia Nigra along with the Superior Parietal part on the right hemisphere of the brain. Their proposed approach achieved 100% accuracy in distinguishing PD from healthy. The limitation of this study is that methodology has been tested on a small sample size dataset.

In the Rolinski et al., 2014 paper, the authors suggested that rs-fMRI can differentiate patients with early PD from healthy controls. Their study primarily consists of calculating connectivity scores based on three regions of interest, such as the caudate, putamen, and pallidum. This paper also recommended the use of rs-fMRI as a biomarker for early PD detection. Recently, rs-fMRI data was used in the early diagnosis of PD using a long short-term memory (LSTM) model by Guo, Tinaz, and Dvornek, 2022. This model achieved

around 72% accuracy with a small size of a dataset consisting of only 84 subjects. All the above studies are performed using identical data acquisition conditions and on a single scanner at the same site. However, larger multi-scanner and multi-site data are required to achieve higher generalization by building a more robust model. There are a few multi-site research Stöcker et al., 2005, Friedman, Glover, Krenz, et al., 2006, and Friedman, Glover, Consortium, et al., 2006 which are based on fMRI. These studies are focused on controlling scanner variations but these studies are performed using very small datasets and not for PD diagnosis. In addition to these studies, the ComBat harmonization approach (M. Yu et al., 2018) is also used for fMRI-derived connectivity measures in a multi-site study but can be used only on image-derived values and predefined relationships. Deep learning methods with the attention-based channel are used on large multi-site resting-state fMRI datasets without explicitly applying any scanner bias mitigation method (T. Zhang et al., 2020) to generalize models to multi-site datasets. The federated learning approach (X. Li, Gu, et al., 2020) with two domain adaptation techniques such as a mixture of experts domain adaptation to reduce the effect of a new domain on the global model and adversarial domain alignment to reduce the discrepancy between the source and target domains are used to resolve domain shift issue observed in multi-site fMRI datasets.

There have been many methods proposed for classifying PD using machine learning and deep learning. However, class imbalance and scanner bias remain issues in PD classification. Moreover, a minimal amount of previous research has used rs-fMRI to classify PD based on data-driven models. To the best of our knowledge, the proposed approach is the first to use a convolutional neural network and convolutional-gated recurrent unit-convolutional neural network (ConvGRU-CNN) to identify Parkinson's disease using Resting-State Functional Magnetic Resonance Imaging (rs-fMRI) data and patient information such as age and gender. Furthermore, a simple and effective distance correlation technique was used for the first time to address class imbalance and scanner bias issues in neuroimaging data which allows us to generalize the proposed model to larger multi-site and multi-scanner settings.

2.4 Methodology

Deep learning techniques in the medical domain have received increasing interest due to their ability to accurately perform tasks and for extracting meaningful features in neuroimaging datasets. However, the performance of the deep learning models is impacted by the imbalanced and multi-scanner datasets issues. Imbalanced datasets exhibit skewed class distributions whereas multi-scanner datasets exhibit data bias or confounding effect due to variance caused by differences in scanner and acquisition protocols. In this study, we aim to resolve two issues associated with rs-fMRI datasets of PD.

1. The dataset is highly imbalanced which introduces a class bias issue. Hence, deep learning models trained on this dataset are bias towards the majority class. In our study, the majority class is PD patients.

2. In order to improve the performance of deep learning, the datasets from two different scanners and different studies and sites are combined. But this leads to scanner-variant features and hence model predictions are dependent on scanner.

Our proposed method focuses on using distance correlation in the objective function to mitigate biased toward majority class and scanner dependencies from features learned by deep learning. In this method, we improve the classification performance on the imbalanced dataset by decorrelating class bias from learned features by model. Scanner dependencies on model performance are mitigated by decorrelating scanner configuration information from learned features to create scanner-invariant features. The proposed method is simple yet more effective and can be applied to the mitigation of a wide range of data bias, confounders, class bias, or a combination of all bias issues as shown in our previous work (Patil and Purcell, 2022). The proposed DcCNN framework in this study, on the other hand, is specifically designed to address scanner dependency and imbalance issues that are common in large clinical trials involving neuroimaging data. The proposed DcCNN model framework is shown

in Figure 2.1. The framework mainly consists of three steps: data preprocessing, balancing dataset using different sampling techniques and adding new dataset, and classification using DcCNN. Finally, the model is evaluated using different evaluation metrics and t-distributed stochastic neighbor embedding (t-SNE) plots.

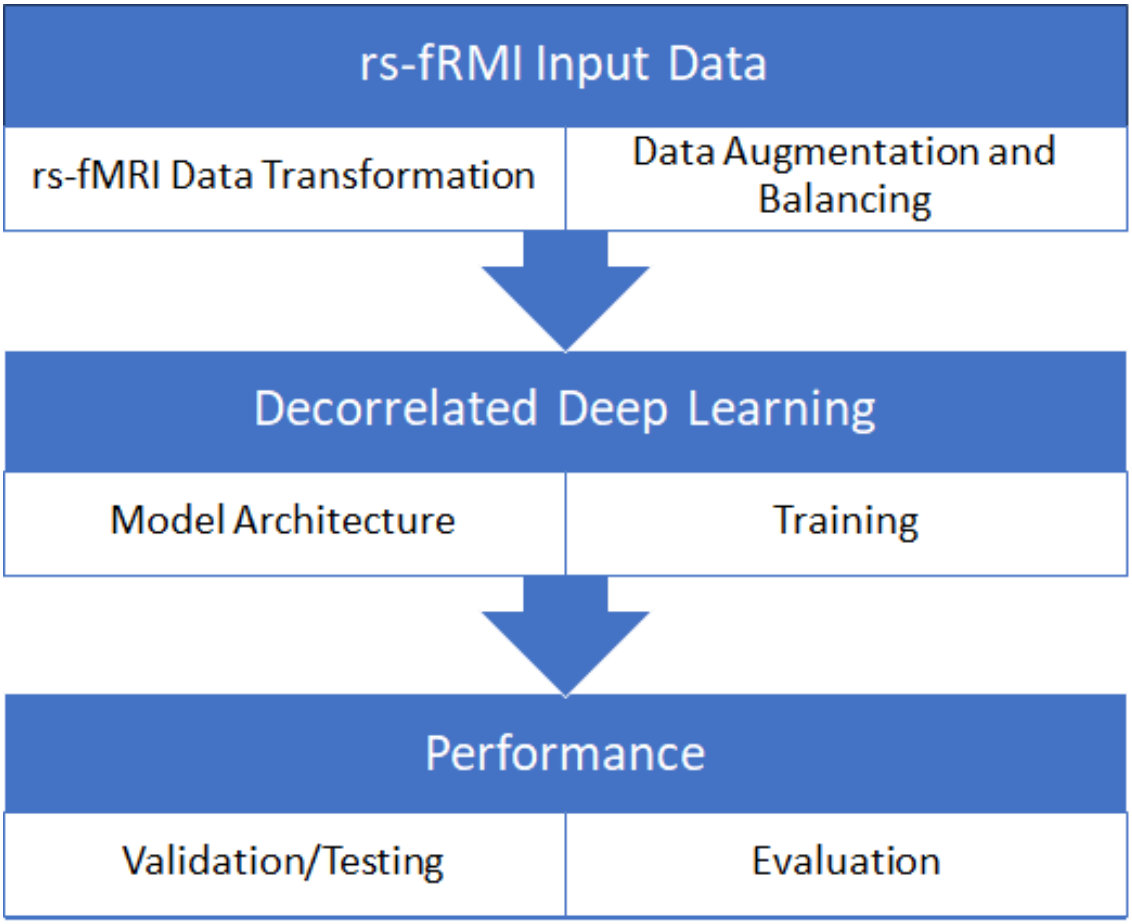


Figure 2.1 General Framework of the proposed method for classification of Parkinson's Disease.

2.4.1 Decorrelated Convolutional Neural Networks

Decorrelated Convolutional Neural Networks (DcCNN) are implemented by applying the decorrelation loss function to CNN architectures. We propose our DcCNN architecture as in Figure 2.2. We can use one or combinations of any layer outputs and concatenate them as features for the decorrelation function depending on the complexity of the task. Distance

correlation is used as a decorrelation function.

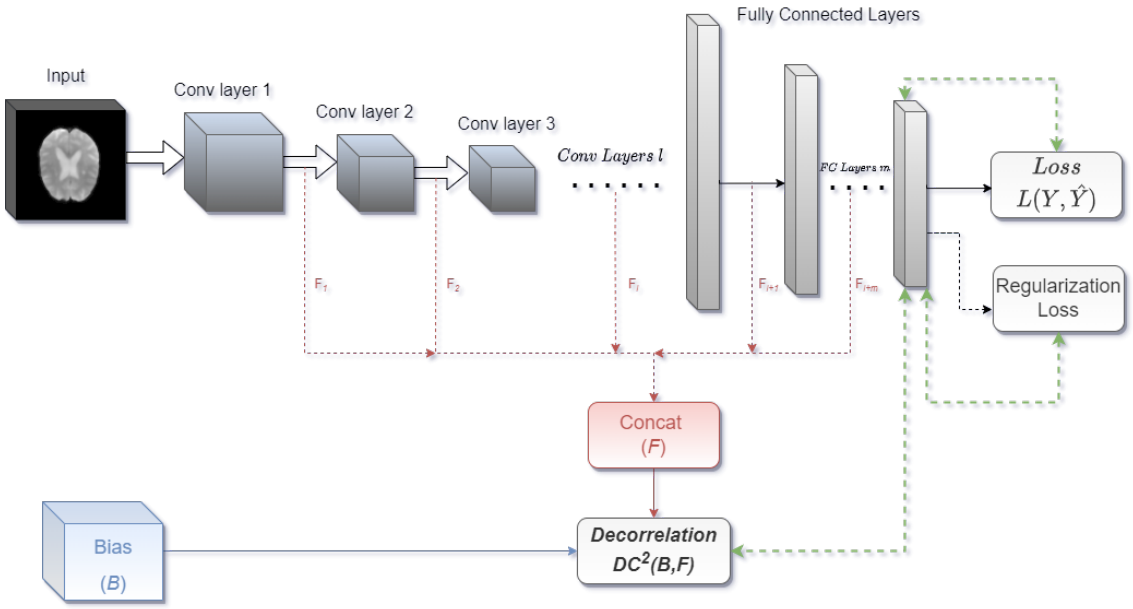


Figure 2.2 Proposed DcCNN architecture. Red dashed lines denote the output of convolutional layers l which are combined together to represent learned features. Green dashed lines indicate the start of the learning process where backward arrows show back-propagation using their respective gradient values while forward arrows show forward paths with updated parameters. Network parameters are updated as per the objective function.

Distance correlation calculates the association between two arbitrary dimension variables using the distances. In our proposed approach, $B_{1,\ldots,p}$ is the bias variable. $F_{1,\ldots,p}$ is features extracted from DNN and p is the total number of samples. The distance correlation is the square root of:

$$\mathcal{DC}^2(B,F) = \begin{cases} \frac{\mathcal{V}^2(B,F)}{\sqrt{\mathcal{V}^2(B,B)\mathcal{V}^2(F,F)}} & \text{if } \mathcal{V}^2(B,B)\mathcal{V}^2(F,F) > 0 \\ 0 & \text{else } 0 \end{cases} \tag{2.1}$$

where $DC(B,F)$ is bounded between 0 and 1. $DC(B,F) = 0$ only if the variables B and F are independent. $v^2(B,F)$ is the distance covariance between a pair of variables and $v^2(B,B)$, $v^2(F,F)$ is the distance variance as defined in Székely, Rizzo, and Bakirov, 2007.

The distance covariance is normalized by the distance variances. The Pearson correlation coefficient (Lee Rodgers and Nicewander, 1988) measures only linear dependencies but features extracted from CNN can have non-linear dependencies and hence distance correlation is more preferable since it measures not only linear but also non-linear dependencies between two random variables.

In our study, we use the squared distance correlation. Class weights are also used in the distance correlation loss function in some of the models to tackle the imbalance problem of scanner data. This function is minimized to reduce the distance correlation between features learned by the networks and the biases. This means that we want to find parameters of the network such that F features have a minimal distance correlation with the B bias variable. The decorrelation function term is added to the standard objective function for optimization.

2.4.2 Mitigation of Class Bias

Previous research work has shown that imbalanced datasets have a negative impact on the performance of CNNs due to bias towards the majority class. PPMI dataset used in this study is highly imbalanced and hence learning discriminating boundaries between Parkinson's Disease (PD) subjects and healthy control subjects could be more challenging. Our DcCNN models introduce the idea of using the decorrelation loss function along with a data sampling technique to address the class bias problem in deep learning due to an imbalanced dataset.

2.4.2.1 PPMI Dataset and Preprocessing

The PPMI dataset consists of around 183 subjects with follow-up visits. This dataset includes 164 PD patients and 19 healthy control subjects. The demographic information and box plot for the PPMI dataset is shown in Table 2.1 and Figure 2.3, respectively. The time required to collect the rs-fMRI data for each subject is around 8 min 4 sec. During data

collection, subjects are instructed to minimize all movements as well as to rest quietly with eyes open with a clear mind during the scan. They also instructed to not to fall asleep during this process. For a few subjects, data has been collected up to 1 to 3 years. In this study, imaging data associated with follow-ups are considered independent since they were scanned at different points in time. The size of each rs-fMRI slice is 68 x 66, and these images are grayscale. A total of 40 axial slices are captured for each subject. The scanner used to collect this dataset is the Tesla scanner manufactured by Siemens Medical Solutions. Functional scans are acquired using EPI sequence (Field Strength=3.0 tesla; Flip Angle=80.0 degree; Matrix X=476.0 pixels; Matrix Y=462.0 pixels; Mfg Model=TrioTim; Pixel Spacing X=3.2941 mm; Pixel Spacing Y=3.2941 mm; Pulse Sequence=EP; Volumes=210.0 time series ; Slice Thickness=3.2999 mm; TE=25.0 ms; TR=2400.0 ms).

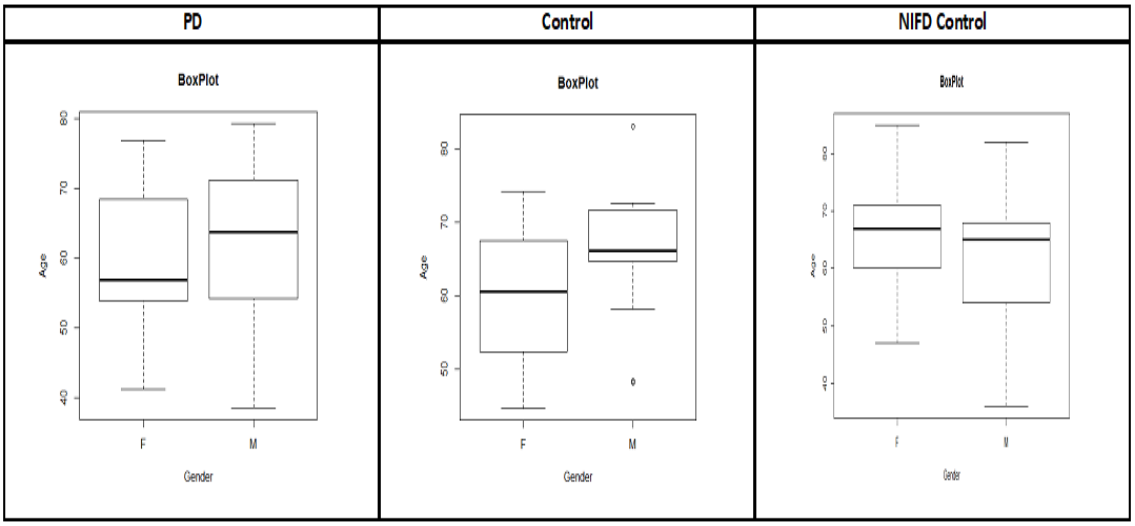


Figure 2.3 The boxplot of Age for Male and Female for PPMI and NIFD Datasets.

Table 2.1 Demographic information of Two datasets PPMI and NIFD.

Datasets	Total Subjects	Group	Subjects	Gender	Subjects	Mean of Age	SD of Age	Min of Age	Max of Age
PPMI	183	**PD**	164	**Male**	111	62.55	10.53	38.6	79.3
				Female	53	59.65	9.36	41.2	76.9
		Control	19	**Male**	15	65.98	9.04	48.1	83.1
				Female	4	59.95	12.1	44.6	74.2
NIFD	215	**Control**	215	**Male**	129	62.6	8.45	36	82
				Female	86	65.9	7.95	47	85

The preprocessing of rs-fMRI is done using FSL v6.0 (S. M. Smith et al., 2004). An FSL-BET extraction tool (S. M. Smith, 2002) is used to extract brain regions and remove skull and neck voxels. Motion correction is performed with the help of the FSL-MCFLIRT toolbox (Jenkinson et al., 2002) to remove motion artifacts introduced by head movement over time. Spatial smoothing of each volume is implemented using a gaussian kernel of 5 mm full width at half maximum to reduce noise without reducing the true underlying signal. High-pass temporal filtering with a cut-off frequency of 0.01 HZ (sigma = 90 seconds) is also applied to remove low-level noise. Since the first 10 slices and the last 5 slices of each subject contains no functional information, they are removed. The end results of this prepossessing for each subject are 66 x 66 PNG images with 25 slices and 210 volumes. The dataset is trained, validated, and tested using 70%, 15%, and 15% of the dataset respectively. To improve the generalization ability of DcCNN models, data augmentation methods such as random rotation, random translation, and elastic deformations (Ronneberger, Fischer, and Brox, 2015) are applied to the training dataset which helps to make the model shift, rotation, and deformation invariant.

Since the number the subjects are less and to minimize the overfitting issue, we use each slice and volume as independent 2D images. Table 2.2 provides the number of 2D images in the PPMI datasets before oversampling which clearly indicates an imbalanced dataset since the number of images for PD subjects is more as compared to healthy subjects. In order to resolve the class imbalance problem, different data oversampling techniques such as Random over-sampling (ROS), Synthetic Minority Over-Sampling Technique (SMOTE), and Stratified sampling are used. ROS (Batista, Prati, and Monard, 2004) is a simple method in which samples from monitory class are randomly increased by making exact copies of existing samples whereas SMOTE synthetically creates new minority samples by interpolating between minority class samples (Chawla et al., 2002) to balance class distribution. Disproportionate or Balanced Stratified Sampling is a sampling technique that randomly divides the data into different strata in such a way that it samples more data from the minority

class samples to balance the samples in the strata (Saleema et al., 2014). The total number of 2D rs-fMRI images after applying oversampling techniques is shown in Table 2.2. We also implement CNN as a feature extraction technique before applying data sampling methods to evaluate the performance of the model on a class imbalanced dataset (Salekshahrezaee, Leevy, and Khoshgoftaar, 2021). A simple method such as the weighted cross-entropy loss function is also implemented to boost the performance of the DcCNN model by providing more emphasis on the minority class. Our proposed method is a fusion model (Oversampling + Weighted loss + Decorrelation Loss) which applies oversampling technique and includes weighted cross-entropy along with decorrelation loss function to mitigate class bias.

Table 2.2 Class Distribution of PPMI training dataset Before and After Oversampling.

Class	Number of Images	
	Before OverSampling	After OverSampling
PD	818790	818790
Control	90930	818790

2.4.2.2 Decorrelation and Weighted Loss in Objective Function

The models tend to predict most images and subjects as PD patients due to class bias. This class bias is mainly caused by the higher number of PD patients compared to healthy control subjects. In order to represent the class bias condition quantitatively and to use it as a bias variable in the decorrelation function, we use a dummy bias variable based on discrete uniform distribution. PD patients group will have a wider discrete uniform distribution than the healthy control group which means the dummy variable would bias the classification results towards PD patients and create class bias. Minimizing the distance

correlation between this dummy bias variable and features will result in balanced true positive and true negative rates.

We introduce the objective function which consists of three main functions, namely, weighted cross entropy, decorrelation function and regularizer L2 loss function to mitigate class bias, and is defined as:

$$J(\theta) = \min_{\theta} L_{WCE}(Y, \hat{Y}) + \lambda \mathcal{DC}^2(B, F) + ||\theta||_2 \tag{2.2}$$

L_{WCE} in Equation 2.2 represents the weighted binary cross-entropy and Y and $\hat{Y}$ are true and classifier outputs, respectively. The weighted binary cross-entropy simply uses class weights to place more emphasis on minority class so that model learns equally from both classes. The decorrelation function is $DC^2(B, F)$ where B is the dummy class bias variable and F is features extracted from the model. The λ in the objective function is a hyperparameter that determines the relative importance of the decorrelation function in relation to the weighted cross-entropy loss function. The last term $||\theta||_2$ is a regularizer L2 loss function in the objective function for weight decay purposes which helps to avoid overfitting issues. Optimizing the decorrelation function along with the weighted cross-entropy loss helps to mitigate class bias.

2.4.2.3 Experimental Setup

The DcCNN model is built by applying decorrelation-based optimization to customized CNN architecture and is trained from scratch. It consists of stacks of 3 convolutional and max-pooling layers with ReLU activation and batch normalization layer, two fully connected layers, and SoftMax as the classifier. These three convolutional layers have 32, 64, and 128 filters respectively. We use a random oversampling technique to have an equal number of samples between two classes i.e. PD and healthy control. We use Root means square propagation (RMS prop) optimizer for optimization and weighted cross-entropy and decorrelation

function with $\lambda = 0.2$ as the loss function as mentioned in the subsection 2.4.2.2. Mini-batch size of 4000 and an exponential cyclical learning policy (L. N. Smith, 2017) which increases and decreases the learning rate by an exponential factor during the training is used. We observe that an exponential decaying learning rate leads to better generalization. For the decorrelation loss function, we use the outputs of fully connected layers and softmax layer as features F. For the evaluation of the DcCNN model, we use different evaluation metrics such as sensitivity, specificity, precision, and balanced accuracy (BC) calculated from the confusion matrix.

All models in this study are implemented from scratch in python using the TensorFlow platform (Martín Abadi et al., 2015) and cuDNN library (Chetlur et al., 2014) on a Linux instance. These experiments are conducted on the AWS Deep Learning AMIs (*Deep learning ami - Developer Guide* n.d.) to accelerate deep learning in the cloud using an Amazon EC2 P2 Instance. We use eight high-speed GPUs, parallel processing cores, and single and double-precision floating-point performance to train the dataset using deep learning. This helps to speed up the training processes.

2.4.3 Mitigation of Scanner Dependencies

A large and balanced neuroimaging dataset is important for deep learning and to improve its generalization ability. Hence, combining all available data from different sites and different scanners plays a vital part in achieving high performance. But it leads to an increase in variance due to differences in acquisition protocols and scanners. This includes scanner upgrades, scanner manufacturers, scanner strength, etc. We combine PPMI and healthy control subjects from the NIFD dataset to balance the dataset and improve the performance of deep learning to detect PD. The idea behind the proposed DcCNN models is to decorrelate the scanner information and features extracted from models to create scanner-invariant features. Three different variations of DcCNN models such as DcCNN, feature

extraction + DcCNN(FE-DcCNN) which extracts features from scanner classifier and use it as bias variable in DcCNN, and decorrelated convolutional-gated recurrent unit DcCNN (ConvGRU-DcCNN) which performs temporal processing are proposed to mitigate the scanner dependencies.

2.4.3.1 NIFD Datasets and Preprocessing

We use only rs-fMRI data for healthy controls from the NIFD dataset and it consists of 215 healthy control subjects with follow-up visits. Just like the PPMI dataset, the demographic information and box plot for the NIFD dataset is shown in Table 2.1 and Figure 2.3, respectively. We can see that there is no significant difference in age distribution between PPMI and NIFD datasets. The size of the rs-fMRI slice is 92 x 92, and the slices are grayscale. A total of 36 axial slices are captured for each subject. The scanner used to collect this dataset is the Tesla scanner manufactured by Siemens Medical Solutions. Functional scans are acquired using EPI sequence (Field Strength=3.0 tesla; Flip Angle=80.0 degree; Matrix X=552.0 pixels; Matrix Y=552.0 pixels; Mfg Model=TrioTim; Pixel Spacing X=2.5 mm; Pixel Spacing Y=2.5 mm; Pulse Sequence=EP; Volumes=240.0 time series ; Slice Thickness=3.0 mm; TE=27.0 ms; TR=2000.0 ms). As we can see, the scanner manufacturer for the NIFD dataset is the same as the PPMI dataset. However, scanner configurations such as TE, TR, slice thickness, voxel size, and the total number of slices and volumes are different. This might introduce the variance related to scanners which will ultimately mask the discriminating features between PD and healthy controls. The rs-fMRI data were preprocessed using the same library and steps as the PPMI dataset. Since the first 5 slices and the last 6 slices of each subject contains no functional information in the NIFD dataset, they are removed. In order to have the same and fixed size as the PPMI dataset, we also deleted the first 30 volumes in the NIFD dataset. So the preprocessed NIFD dataset has 66 x 66 PNG images with 25 slices and 210 volumes for each subject. The NIFD dataset is also divided into 70% training, 15% validation, and 15% testing dataset. After combing the PPMI and

NIFD datasets, a total of 2346750 images were produced and the class distribution of the combined dataset is provided in Table 2.3.

Table 2.3 Class Distribution of Combined PPMI and NIFD datasets.

Class	Number of Images		
	Training	Validation	Testing
PD	813750	141750	189000
Control	819000	178500	204750

2.4.3.2 Decorrelation in Objective Function

Deep learning models are extremely sensitive to non-biological variability such as acquisition and scanner settings in the field of neuroimaging data. One of the important problems in large clinical trials is the scanner dependencies/bias. To deal with the scanner dependencies issue, we introduce three types of scanner bias variables which contains: (i) scanner voxel size i.e. slice thickness and pixel spacing (Shafiq-ul-Hassan et al., 2017), (ii) features extracted from scanner classifier, and (iii) temporal standard deviation to represent scanner-to-scanner variability (Friedman, Glover, Consortium, et al., 2006).

The models are trained with a combination of cross entropy loss $L(Y, \hat{Y})$, the decorrelation loss $DC^2_{control}(B, F)$, and the regularizer L2 Loss $||\theta||_2$ functions. This objective function can be expressed as:

$$J(\theta) = \min_{\theta} \lambda_1 L(Y, \hat{Y}) + \lambda_2 \mathcal{DC}^2_{control}(B, F) + ||\theta||_2 \tag{2.3}$$

where L is the softmax cross-entropy loss and $||\theta||_2$ is regularizer L2 loss function. The decorrelation function is $DC^2_{control}(B, F)$ where B is scanner bias variable and F is features

extracted from the model and subscript *control* indicates the decorrelation function is only applied to control subjects since the healthy control subjects had been scanned using both the scanners with different acquisition protocols i.e. present in PPMI as well as NIFD datasets whereas PD subjects had been scanned using only one scanner out of two scanners i.e. present in only PPMI dataset. This will help models to remove scanner-related information than removing the main task i.e. PD detection-related information. The λ_1 and λ_2 in the objective function are hyperparameters that control the trade-off between the cross-entropy loss function and the decorrelation function. Since the number of healthy controls in the PPMI dataset is less compared to NIFD dataset, higher class weights are assigned to PPMI controls than NIFD controls to make decorrelation loss for PPMI controls larger than NIFD controls. This will help models to decorrelate features equally from both scanners and ultimately to resolve imbalanced scanner data problem for healthy controls.

2.4.3.3 Experimental Setup

We train three different DcCNN models with different architectures and scanner bias variables. The first model, abbreviated as DcCNN, has the same architecture as the DcCNN model used to mitigate class bias except for changes in the objective function to mitigate scanner dependencies and there are three stack convolutional layers with 32,16, and 16 filters and followed by two hidden layers with 40 and 100 neurons. We train DcCNN with a mini-batch size of 4000 and an exponential cyclical leaning policy using an RMS prop optimizer for optimization with a decay of 0.005. Hyperparameters $\lambda_1 = 0.5$ for cross-entropy loss and $\lambda_2 = 5.0$ for decorrelation function are used to control the trade-off between two loss functions as mentioned in subsection 2.4.3.2. The output of the first convolutional layer and fully connected layers are used as feature F, whereas slice thickness and pixel spacing are considered as scanner information and used as scanner bias variable B.

The second model (FE-DcCNN) has two models. The first model is built to predict the scanner which we refer it as Feature Extraction(FE) model. The dataset used to train

this model consists of only healthy control subjects from PPMI and NIFD datasets. once the training is done, features are extracted from the FE model and used as scanner bias variable in the second i.e. DcCNN model. Both FE and DcCNN models have the same architecture and the same training dataset. These models have 5 stacks of convolution, batch normalization, and max-pooling layers with ReLU activation as shown in Figure 2.4 followed by two fully connected layers with 40 and 100 neurons. Both models use 32, 16, 16, 8, and 8 filters to extract discriminative features for the detection of PD. The output of the fifth convolutional layer in the FE model is used as scanner bias variable B whereas the output of the fifth convolutional layer along with fully connected layers in the DcCNN model are used as feature F. The hyperparameters used in objective function are $\lambda_1 = 0.05$ and $\lambda_2 = 0.95$. We have used the dropout of 0.2 in the first four convolutional layers to reduce the overfitting problem in the model and the rest of the training configuration is the same as the first model DcCNN.

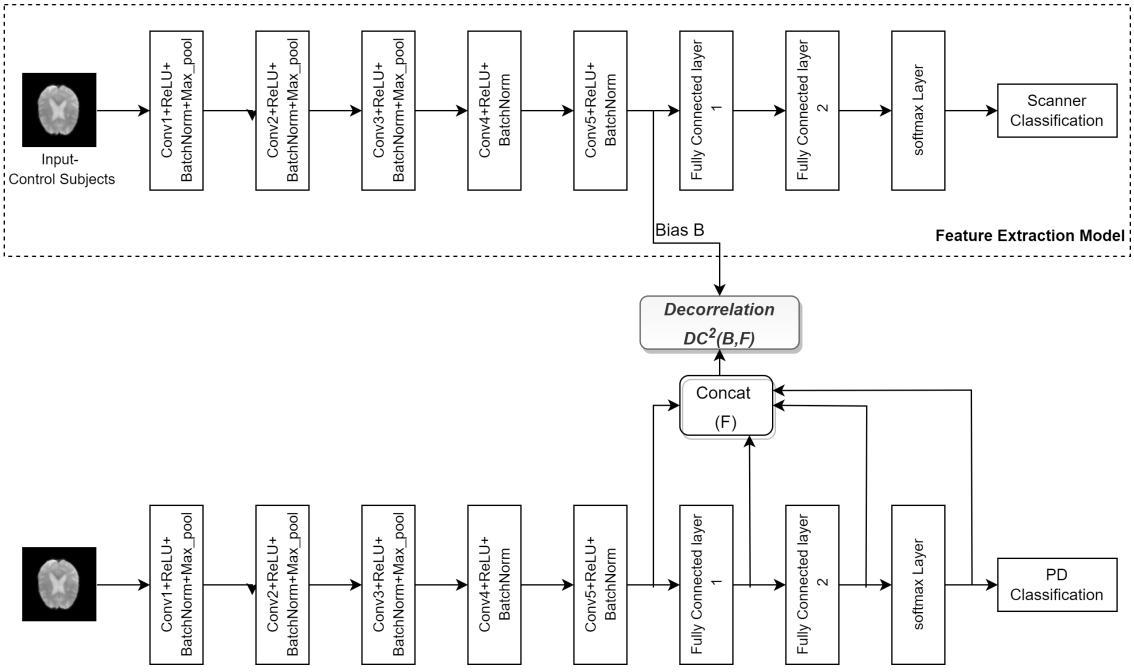


Figure 2.4 The architecture of the FE-DcCNN model.

To make use of temporal information present in rs-fMRI, we implement the third model

(ConvGRU-DcCNN) as shown in Figure 2.5. ConvGRU-DcCNN performs temporal processing first and uses a 3D image of size 66 x 66 x 210 as the input. Since we have to use temporal information for this model, we have to convert 2D images to 3D images and that produces a total of 11175 images, including 5450 PD and 5725 healthy control PNG samples. The core architecture consists of convolutional gated recurrent operations (convGRU) (Bengs, Gessert, and Schlaefer, 2020) as the first layer and followed by the DcCNN architecture. ConvGRU is used to perform temporal processing. The DcCNN part consists of three convolutional layers with filters 16, 32, and 32, followed by two fully connected layers with 1000 and 500 neurons. The model is trained using an Adam optimizer with a mini-batch size of 256 and a learning rate (lr) scheduler with an initial lr of 0.001 with a decay of 0.5. In addition to this, an optimizer weight decay of 0.005 is used. We use $\lambda_1 = 0.2$ and $\lambda_2 = 0.6$ in objective function. For decorrrelation loss, we use the output of the second convolutional layer and first fully connected layer as features F whereas temporal standard deviation (temporal fluctuations) is used as scanner bias B.

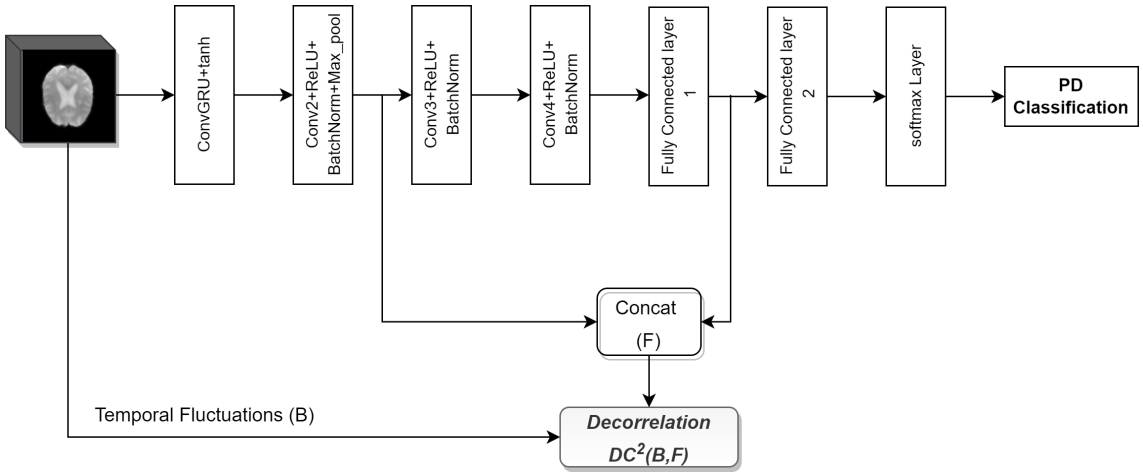


Figure 2.5 The architecture of the ConvGRU-DcCNN model.

2.5 Results

In order to access whether DcCNN models perform better to mitigate class bias and scanner bias, we apply our method and baseline model to the single scanner imbalanced PPMI dataset and combination of multi-scanner PPMI and NIFD datasets respectively.

Single Scanner Imbalanced Dataset

We access the performance of DcCNN to classify PD on PPMI imbalanced dataset. Our proposed fusion method aims to mitigate class bias. In order to show that DcCNN reduces the statistical dependence between features and class bias variables, we plot the distance correlation against iterations as shown in Figure 2.6. The plot shows that distance correlation decreases as the iteration increases for our fusion model as opposed to the oversampling method. We compare our fusion model with different CNN models and existing data-sampling techniques. The baseline model is a simple CNN model and has the same architecture as DcCNN where no data-sampling technique and class bias mitigation methods are applied. The existing data-sampling techniques (Leevy et al., 2018) such as smote and oversampling are implemented to address the class imbalance issue. We have also compared our model with a fusion of different combinations of existing class bias mitigation techniques such as fusion of oversampling and weighted loss functions, a fusion of feature extraction and smote, and stratified sampling.

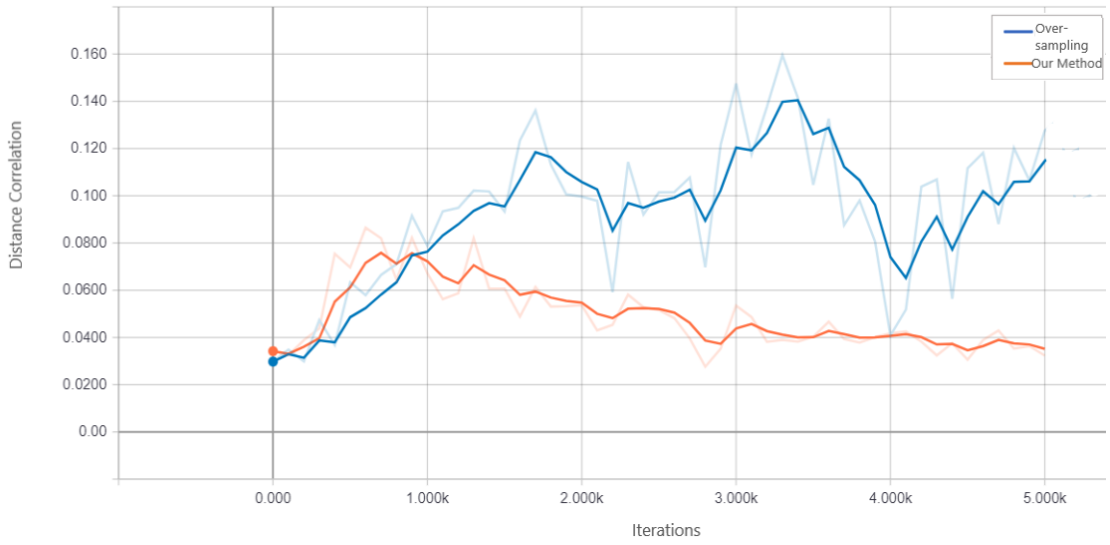


Figure 2.6 Distance correlation between learned features and class bias for the imbalanced dataset.

The results of the holdout testing dataset for each method are displayed in Table 2.4 and the performance of imbalanced classification is measured specifically by sensitivity, specificity, precision, and balanced accuracy (BA). As we can see from the results, our proposed fusion method significantly increases balanced accuracy as compared to other methods. This, therefore, suggests that by using the decorrelation function along with oversampling technique and weighted loss function creates features that are invariant to class bias. The precision and specificity are higher for our fusion model compared to other methods. Lower sensitivity and higher specificity for our fusion model indicates that model prediction is not biased towards the majority class i.e. PD subjects whereas higher sensitivity and lower specificity for methods such as baseline, smote, FE+smote, stratified sampling, and oversampling indicate model prediction is highly biased towards PD class. We notice that the weighted loss function helps the model to improve balanced accuracy. Figure 2.7 shows the confusion matrix of the baseline model and our proposed DcCNN model to classify slices into the PD and healthy controls. The confusion matrix for the baseline model clearly indicates that all subjects are

classified as PD due to the presence of class bias, while our proposed model classifies both classes almost equally by mitigating this class bias. Figure 2.8 illustrates the ROC curve of different methods. From this graph, we observe the superior performance of our fusion Dc-CNN model over traditional data-sampling methods. In both balanced accuracy and ROC metrics, our DcCNN fusion method clearly outperforms other methods.

Table 2.4 Performance Evaluation of PD Classification for imbalanced PPMI Dataset using different methods.

Methods	**Sensitivity**	**Specificity**	**Precision**	**BA**
Baseline	100.00%	0.01%	90.00%	50.01%
Smote	94.60%	8.60%	90.30%	51.60%
FE + Smote	93.60%	4.70%	89.80%	49.15%
Stratified	95.60%	4.50%	90.00%	50.05%
Oversampling	71.20%	34.90%	90.80%	53.05%
Oversampling+weighted loss	49.00%	59.20%	91.50%	54.10%
Our method	**58.47%**	**60.37%**	**93.07%**	**59.42%**

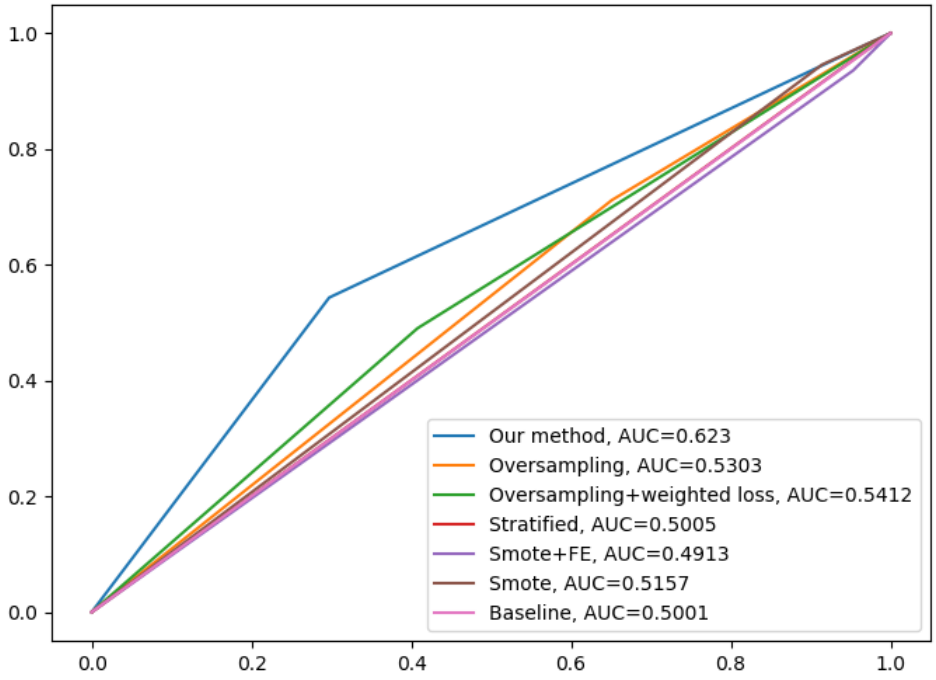


Figure 2.8 ROC curves of different methods for imbalanced dataset.

Due to a few labeled rs-fMRI images available at the subject level in the PPMI dataset, we train the models at the slice level which increase the data and avoid overfitting issue. The above reported results are for the slice-level classification. Since in the medical field, subject-level PD classification is important, we propose a global subject-level classification by using a max-wins voting strategy. In this strategy, all slices for each subject are classified and then the class with maximum votes for a given subject determines the global subject classification. This will allow to classify and assign PD or healthy control labels to a given subject. As shown in Table 2.5 applying the max-wins voting strategy for subject-level classification significantly improved accuracy by correcting a small number of misclassified slices. Our fusion DcCNN model achieves a subject-level balanced accuracy of 67% after applying a max-wins voting strategy.

Table 2.5 Sensitivity, Specificity, Precision, and Balanced accuracies of slicewise and subjectwise PD recognition for imbalanced PPMI testing Dataset (%). Results are mean across three initialization with 95% confidence interval.

Methods	**Sensitivity**	**Specificity**	**Precision**	**BA**
Slice-level	58.47±0.05	60.37±0.08	93.07±0.01	59.42±0.03
Subject-level	66.67±0.08	66.67±0.20	95.13±0.03	66.67±0.10

Multi-scanner Datasets

A DcCNN, an FE-DcCNN, and a ConvGRU-DcCNN are the three main models presented in this subsection to create features that are invariant to scanner and acquisition protocols while maintaining the performance of PD classification. This will reduce the influence of scanner on model predictions. We compare our proposed models with baseline models. In a similar way to the previous imbalanced dataset experiment, baseline models such as CNN and ConvGRU-CNN share the same architecture as DcCNN and ConvGRU-DcCNN, respectively, without any scanner bias mitigation methods being incorporated. Figure 2.9 shows that statistical dependence between learned features and scanner bias decreases as iteration increases for ConvGRU-DcCNN as opposed to the baseline ConvGRU-CNN model. The purpose of this plot is to observe the trend rather than to show the true difference between the distance correlation values of the ConvGRU-DcCNN model and the baseline model since weights have been assigned to calculate the decorrelation function used in ConvGRU-DcCNN versus the baseline model. We also evaluate the performance of scanner bias mitigation techniques using accuracy, scanner classification accuracy, and error rate for each dataset/scanner (since each dataset represents one scanner). The scanner classification accuracy indicates the scanner information present in features that influence the decision of model prediction.

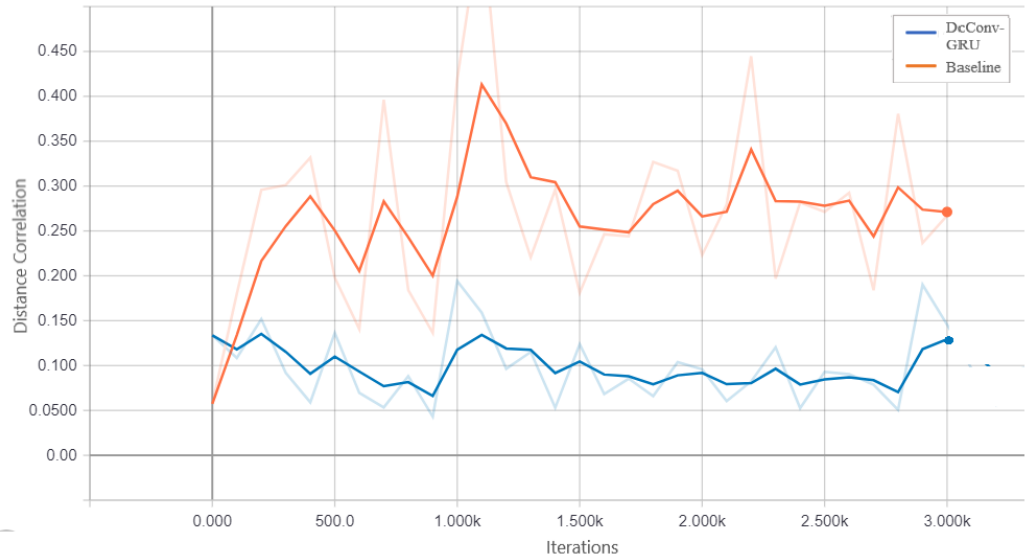


Figure 2.9 Decorrelation between learned features and scanner bias for baseline ConvGRU-CNN and ConvGRU-DcCNN models.

Table 2.6 presents the performance of different types of DcCNN models on a multi-scanner testing dataset. As expected, the scanner classification accuracy for baseline models is 100% which means models make predictions based on features that are dependent on scanner and not on the main task of PD recognition. With the FE model, the scanner classification was performed using only healthy control groups, and scanner-relevant features were extracted for the FE-DcCNN model to use as scanner bias variables. FE model results in an accuracy of 92.6% at slice level and 100% at subject level. All three types of DcCNN models reduce the scanner classification accuracy compared to baseline models indicating that DcCNN reduces scanner dependencies fairly with a slight reduction in accuracy. Accuracy for Baseline models is high due to the fact that all PD subjects in the dataset had been scanned on one scanner and the majority of the healthy control subjects had been scanned on another scanner. Thus, it makes the task more harder and we can see a reduction in accuracy for DcCNN when compared with Baseline models. Hence, for our multi-scanner dataset, we can say that raw classification accuracy is not only the consideration. The error rates for both datasets (i.e. both scanners) increase for DcCNN models indicating scanner bias removal

Table 2.6 Performance Evaluation of Baseline Models and DcCNN models using PPMI and NIFD datasets.

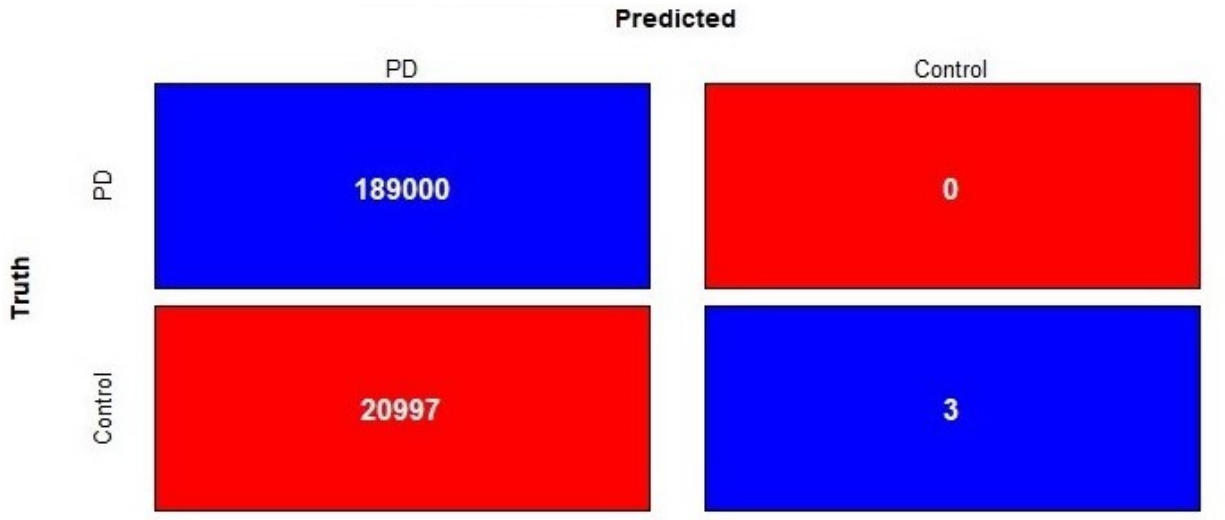
Models	**Accuracy**	**Scanner Classification Accuracy**	**NIFD Error rate**	**PPMI Error rate**
Baseline Models:				
CNN	94.70%	100.00%	0.00	0.00
ConvGRU-CNN	94.70%	100.00%	0.00	0.00
Our Models:				
DcCNN	**80.47%**	83.10%	0.25	0.06
FE-DcCNN	**77.80%**	**80.43%**	0.30	0.17
ConvGRU-DcCNN	65.77%	**63.13%**	0.46	0.28

is performed. ConvGRU-DcCNN model performs poorer compared to the DcCNN and FE-DcCNN models in terms of accuracy, possibly because it removes information related to the main task while reducing scanner dependencies. The ConvGRU-DcCNN performs poorly, most likely due to four factors: removal of PD-relevant features, decorrelation penalization leading to a negative influence on predictive accuracy, reduction in data size, and inclusion of PD information in scanner bias variable. DcCNN and FE-DcCNN models have similar accuracy while substantially decreasing the scanner dependencies.

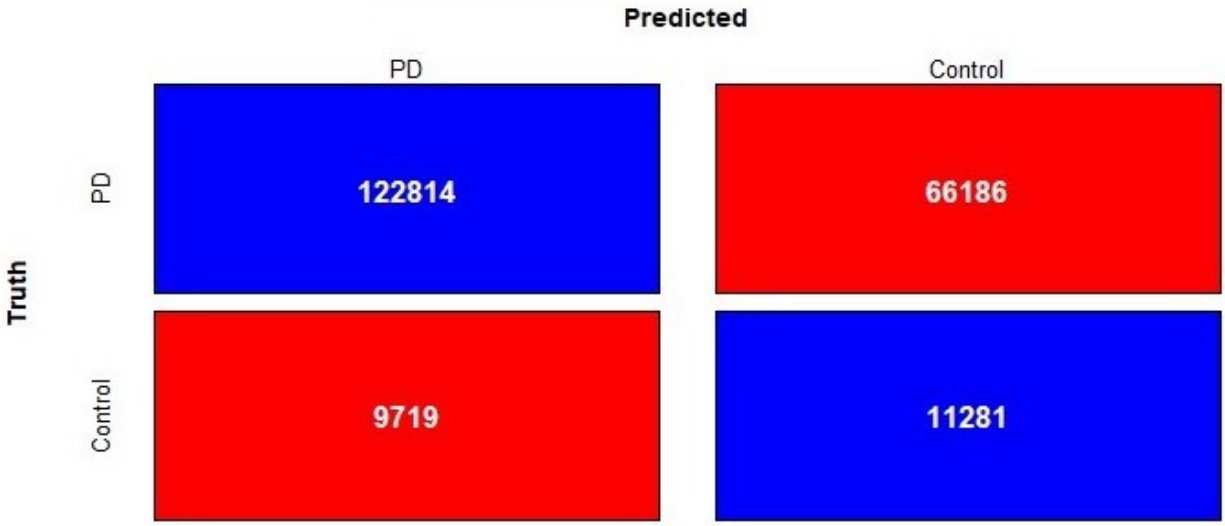
Finally, these above results are further supported by t-distributed stochastic neighbor embedding (t-SNE) visualizations of the learned fully connected layer features as shown in

Figure 2.10. Since only healthy control subjects had been scanned using both scanners and present in both datasets, we plot tSNE visualization for the healthy control group. We observe that the baseline models such as CNN and ConvGRU-CNN have a clear association with scanner since the PPMI dataset is grouped on the right side, while the NIFD dataset is grouped on the left side of the Figure 2.10a. But scanner features become jointly embedded for DcCNN, FE-DcCNN, and ConvGRU-DcCNN models which indicate no apparent bias towards scanner. This suggests that our proposed DcCNN models successfully create features that are invariant with respect to scanner without compromising the performance of PD classification. For the FE-DcCNN model, data points in Figure 2.10b are largely indistinguishable across all two scanners compared to the DcCNN model in Figure 2.10c. This can also be confirmed by scanner classification accuracy for FE-DcCNN is lower than the DcCNN model. Similar to FE-DcCNN, the features learned by the ConvGRU-DcCNN model spread uniformly across all scanners indicating successful mitigation of scanner dependencies but the ConvGRU-DcCNN model results in a drastic loss in accuracy indicating the removal of information related to the main task.

For subject-level classification, we use the same max-wins voting strategy as defined for a single scanner imbalanced dataset. The above reported results for multi-scanner datasets are for the subject-level classification. The evaluation metrics for slice-level and subject-level classification are summarized in Table 2.7. All these results show that the FE-DcCNN model not only successfully mitigates the scanner bias but also achieves high performance in comparison with DcCNN and ConvGRU-DcCNN models respectively. FE-DcCNN model achieves a subject-level accuracy of 78% after applying a max-wins voting strategy and scanner classification accuracy of 80%.



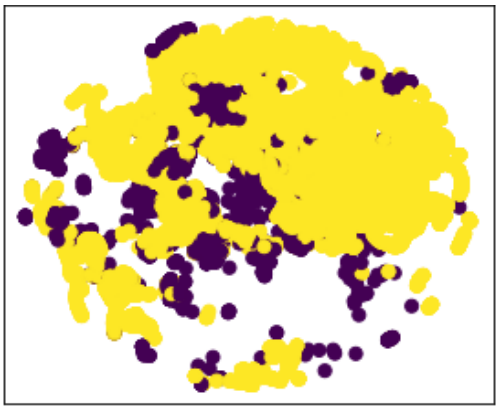
(a) Baseline



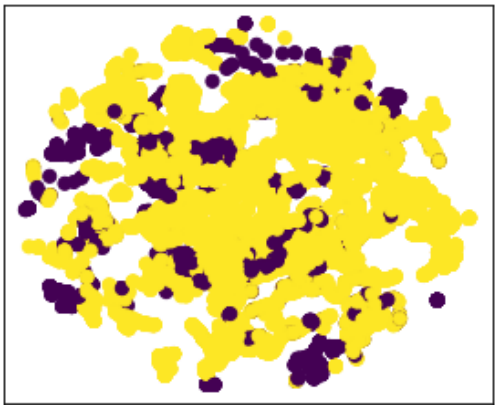
(b) Our method

Figure 2.7 Confusion matrix of baseline and our method(ROS + weighted loss + DcCNN) with two classes for imbalanced PPMI testing dataset(Slice-level PD recognition).

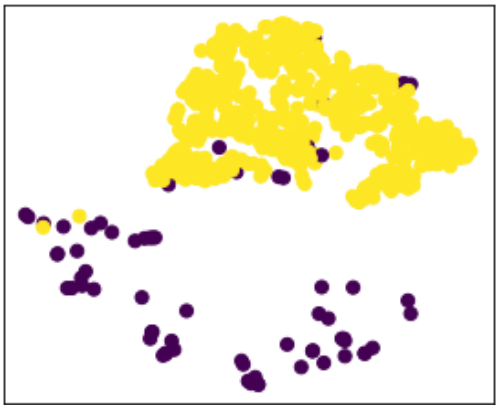
(a) Baseline:CNN



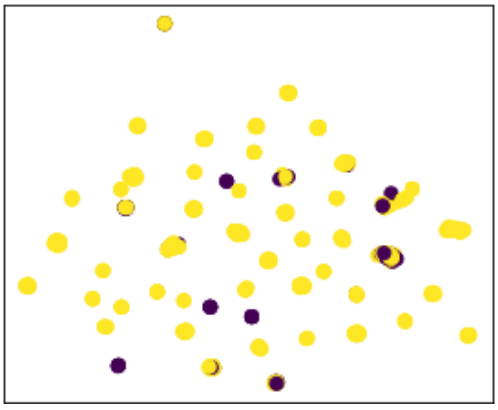
(b) DcCNN



(c) FE-DcCNN



(d) Baseline:ConvGRU-CNN



(e) ConvGRU-DcCNN

Figure 2.10 tSNE plot of the learned fully connected layer features for healthy control data. The yellow color indicates the NIFD dataset scanner and the purple color indicates the PPMI dataset scanner

Table 2.7 Sensitivity, Specificity, Precision, F1, and accuracies of slicewise and subjectwise PD recognition for PPMI and NIFD testing Datasets (%). Results are mean across three initialization with 95% confidence interval.

Models	**Methods**	**Sensitivity**	**Specificity**	**Precision**	**F1**	**Accuracy**
DcCNN	Slicewise	76.87+0.06	78.00±0.08	76.90±0.05	76.60±0.01	77.47±0.02
	Subjectwise	79.63+0.09	81.20±0.10	80.43±0.06	79.57±0.03	80.47±0.03
FE-DcCNN	Slicewise	83.40+0.14	71.00±0.06	72.70±0.02	77.20±0.05	76.95±0.03
	Subjectwise	80.53+0.16	75.20±0.05	75.03±0.01	77.13±0.07	77.80±0.05
ConvGRU-DcCNN	Slicewise/ Subjectwise	74.07+0.03	58.13±0.01	62.00±0.002	67.50±0.01	65.77±0.01

2.6 Discussion

This study presents a decorrelation-based bias mitigation technique that can be applied to deep learning architectures such as CNN, ConvGRU, and fusion methods to mitigate not only class bias but also scanner bias by creating class and scanner invariant features. We have demonstrated that our decorrelation technique can be applied to any architecture and provides a high level of flexibility. The hyperparameter $\lambda > 0$ plays a vital role in deciding the importance of decorrelation and regular loss function. When $\lambda = 0$, it means it is a baseline model with no bias mitigation technique applied. Extreme high values of λ will cause unstable training and poor classification performance. Hence, finding optimal values for hyperparameters λ is crucial and can be achieved by trying different values of λ. We notice that increasing the batch size improves the stability of the decorrelation function during training. In addition, it provides unbiased estimates of distance covariance when the batch size is larger. Similar to hyperparameter λ, we experience that finding the optimal combination of the output of layers as feature F helps in improving the performance of the bias mitigation technique. The choice of feature F depends on the type of bias mitigation technique and model architecture. As stated in our previous work (Patil and Purcell, 2022), the bias variable B should provide more precise bias-relevant information.

The rs-fMRI original imaging data is organized in 4D matrices which contain spatial as well as temporal information. Due to high dimensionality and small dataset size, deep learning models face problems like overfitting when 4D data is used. This would only be solved by adding more data. However, 2D and 3D rs-fMRI data used in this study show the applicability of using this data for PD classification while significantly mitigating the class and scanner bias. We also find that the ConvGRU-DcCNN model almost exhibits similar performance with and without class weights for decorrelation function, since using temporal information reduces the size of the dataset and ultimately the imbalance ratio between PPMI controls and NIFD controls. Out of the three types of scanner bias variables used to mitigate

scanner bias, features extracted from the scanner classifier bias variable provide more accurate scanner-relevant information since the FE-DcCNN model yields optimal results which reduces scanner dependence without removing much PD-specific information.

The results from the class bias mitigation study show that not only we are able to achieve high performance than existing traditional approaches but also successfully mitigate bias towards the majority class. We have also shown that the same decorrelation function technique can be used to remove scanner dependencies. The scanner classification accuracy and tSNE plots confirm that scanner dependencies have been reduced. Since existing harmonization and domain adaptation methods approach scanner mitigation differently than our method, we do not directly compare them to our method. Additionally, our proposed model differs from previous methods in that it is designed for rs-fMRI data collected from a single scanner with identical acquisition protocols and a single site rather than from multi-scanner and multi-site data. The presented method suggests that combining multi-scanner data and increasing the size of the dataset improve the performance of PD classification compared to single scanner imbalanced data.

2.7 Conclusion

The performance of deep learning models is highly impacted by bias variability and class imbalance present in data. We introduce a novel decorrelation approach, which reduces the distance correlation between the features learned by deep learning models and biases. The main goal of this approach is to mitigate scanner dependencies and class bias which will help the model to generalize to multi-scanner and multi-center datasets. The proposed framework includes extensive data preprocessing modules and decorrelated deep learning-based classifiers to distinguish PD patients from healthy controls using rs-fMRI data. We evaluated our four different models on single scanner imbalanced and multi-scanner datasets. On a single scanner imbalanced PPMI datasets, our proposed DcCNN model significantly

improves performance by alleviating bias toward the majority class, whereas our proposed FE-DcCNN model produces scanner-invariant features without affecting accuracy much on multi-scanner PPMI and NIFD datasets. Furthermore, the rs-fMRI dataset is used for the first time to train CNN models for PD classification. These simple and yet efficient proposed DcCNN models perform better than previous approaches and baseline models to mitigate the bias and require fewer hyperparameters to optimize. We additionally verify from the results that using a multi-scanner and larger dataset results in significantly better performance when compared with a single scanner imbalanced dataset. This study also demonstrated that subject-level classification results in an even more robust model and improves accuracy using a max-wins voting strategy.

An immediate next step would be using advanced visualization techniques such as saliency maps, DeepLIFT, and occlusion maps. A combination of these precise and detail-oriented visualization techniques may help in characterizing fMRI biomarkers for PD. Our proposed models also demonstrate the potential for predicting stages in the progression of PD, which could be addressed in future studies. Additional future direction works also include collecting a larger dataset and more information related to patients along with individual rs-fMRI slices and temporal information to achieve higher accuracy and reliability. A larger dataset and increased computation complexity will also enhance the overall performance of 4D-DcCNN models by taking advantage of using the inherent spatial-temporal information in 4D rs-fMRI data. Moreover, it would be interesting to investigate how by applying the proposed decorrelation approach to pre-trained models and to different types of data variations and biases would impact performance.

Chapter Three

Decorrelation-Based Deep Learning for Bias Mitigation: Learning Generic Bias Invariant Feature

3.1 Abstract

Although deep learning has proven to be tremendously successful, the main issue is the dependency of its performance on the quality and quantity of training datasets. Since the quality of data can be affected by biases, a novel deep learning method based on decorrelation is presented in this study. The decorrelation specifically learns bias invariant features by reducing the non-linear statistical dependency between features and bias itself. This makes the deep learning models less prone to biased decisions by addressing data bias issues. We introduce Decorrelated Deep Neural Networks (DcDNN)/ Decorrelated Convolutional Neural Networks (DcCNN) and Decorrelated Artificial Neural Networks (DcANN) by applying decorrelation-based optimization to Deep Neural Networks (DNN)/Convolutional Neural Network(CNN) and Artificial Neural Networks (ANN), respectively. Previous bias mitigation methods result in a drastic loss in accuracy at the cost of bias reduction. Our study aims to resolve this by controlling how strongly the decorrelation function for bias reduction and

loss function for accuracy affect the network objective function. The detailed analysis of the hyperparameter shows that for the optimal value of hyperparameter, our model is capable of maintaining accuracy while being bias invariant. The proposed method is evaluated on several benchmark datasets with different types of biases such as age, gender, and color. Additionally, we test our approach along with traditional approaches to analyze the bias mitigation in deep learning. Using simulated datasets, the results of t-distributed stochastic neighbor embedding (t-SNE) of the proposed model validated the effective removal of bias. An analysis of fairness metrics and accuracy comparisons shows that using our proposed models reduces the biases without compromising accuracy significantly. Furthermore, the comparison of our method with existing methods shows the superior performance of our model in terms of bias mitigation, as well as simplicity of training.

3.2 Introduction

Modern machine learning techniques, especially deep learning models, have shown tremendous improvement in various fields using limited, as well as large-scale, datasets to perform different types of tasks. However, the reliability of these models is based solely on the quality of training datasets. The quality of datasets can be dramatically affected by different types of biases such as representation, measurement, algorithmic, temporal, social, etc.(Mehrabi et al., 2021). These biases can induce irrelevant information in the training dataset and affect model generalization and the performance of deep learning. Collecting datasets that are free of bias and are well distributed is expensive and very time-consuming (e.g., medical datasets). There are pre-existing large-scale datasets such as Yahoo YFCC100M Flickr (Kärkkäinen and Joo, 2019) and ImageNet (Tommasi et al., 2017) that already contain different types of biases and recollecting these datasets is cumbersome and can be impossible.

Convolutional Neural Networks (CNN) (LeCun, Boser, et al., 1989) and Deep Neural Networks (DNN) are rapidly evolving as an automated method of extracting high-level features

from 2D and 3D data. However, these features are prone to biases when dataset collections are not properly controlled. Recent work has focused on methods such as pre-processing of datasets, sampling and reweighting (Kamiran and Calders, 2012), adversarial training to mitigate bias (B. Kim et al., 2019), and others (Adeli et al., 2021). However, these methods face the problem of instability and require additional careful fine-tuning of hyperparameters. In order to resolve these issues, our work aims at creating simple and stable models to mitigate biases while achieving high performance. This study introduces a novel technique based on a distance correlation loss function to decorrelate the features learned by the model with a bias. We term this model the Decorrelated Deep Neural Network (DcDNN) and Decorrelated Artificial Neural Network (DcANN) when applied to DNN and Artificial Neural Network (ANN) (Hagan, Demuth, and Beale, 1997), respectively. To the best of our knowledge, this study is the first example in which a simple and effective distance correlation technique was used for bias mitigation in a deep learning context.

In this study, we will mainly focus on attributes related to data for bias mitigation. These biases are color, gender, and age. These biases can impose severe challenges to the decisions made by deep learning. The experiments were performed on five datasets to show that our method can be generalized across different domains and different deep learning models. For our proposed method, we assumed that the existence of data bias is known for the training dataset. The main objective of our proposed models is to minimize the correlation between the high-level features learned by the model and the bias variable. Bias variables used in our study are color information, age, and gender.

Our main contributions in this study are:

- The introduction of a new loss function to ANN, CNN, and DNN to decorrelate bias from the learned features, which helps in mitigating bias;
- Generalizing the idea of decorrelation across different domains and biases;
- Comparing our proposed DcDNN and DcANN methods to existing methods.

In all experiments with different datasets, we showed that our methods achieved better performance as compared to existing methodologies. DcDNN and DcANN methods are able to learn more relevant information for a given task by mitigating irrelevant bias-related features. We can validate this by studying the t-SNE plots. We also show that using our proposed method, accuracy is not largely compromised even after mitigating the biases. In concurrent work, a similar notion of using distance correlation as a regularizer term was developed, but it is used to achieve stability of network prediction and compared against adversarial methods(Kasieczka and Shih, 2020). However, the ability of the distance correlation function is not fully explored due to limiting the dimensions of input variables to be one-dimensional.

The rest of this paper is structured as follows: Section 3.3 presents a literature review and focuses on the pros and cons of existing methods, whereas section 3.4 outlines the proposed methodologies and used datasets; Section 3.5 discusses the results, evaluation metrics, and comparison with existing methodologies; Section 3.6 discusses the performance of our proposed method and Section 3.7 provides a conclusion and remarks and opportunities for future work.

3.3 Related Work

Data-driven deep learning frameworks are widely used in complex real-world applications, and the bias and the fairness of these frameworks is still an active and popular topic of research in the field. Most machine learning algorithms fall into three categories: pre-processing, in-processing, and post-processing, depending on how they tackle bias and unfairness issues (Mehrabi et al., 2021). We focus on in-processing learning algorithms in this paper.

An algorithmic solution of reweighing or resampling the data to remove bias from the training dataset is provided by Kamiran and Calders, 2012. However, this study is limited to

using only binary bias variables and a binary classification problem. Calmon et al. (Calmon et al., 2017) demonstrated an optimized pre-processing method that uses an optimization algorithm to transform datum probabilistically to have a fairer classification. In order to minimize representation bias, Y. Li and Vasconcelos, 2019 investigated a data resampling technique called Representation Bias Removal (REPAIR). In this technique, optimization is performed by minimizing the representation bias to learn weights that penalize misclassified examples and maximizing the classification loss on the reweighted dataset.

Recent studies Dinsdale, Jenkinson, and Namburete, 2021; B. H. Zhang, Lemoine, and Mitchell, 2018; Mandis, n.d. used adversarial learning based on the min–max objective to remove confounds, such as scanners variation in medical data, by applying the domain adaption framework to remove gender bias from word embeddings, or to remove race from a hiring employees dataset and loan approvals using the Generative Adversarial Network (GAN) framework. The Bias-Resilient Neural Network (BR-Net) is another adversarial training-based approach used to learn bias-invariant features. The BR-Net applies adversarial maximization of linear correlation between bias prediction and protected bias variable and minimization of cross-entropy or mean squared error (MSE) loss for the classification task (Adeli et al., 2021). The basic foundation of the BR-net is based on GANs (Goodfellow et al., 2014) used for domain-adaptation. Similar approaches based on adversarial training to predict the bias variable were proposed in Sadeghi, R. Yu, and Boddeti, 2019; T. Zhao et al., 2022; T. Wang et al., 2019. Most of the adversarial methods require two separate neural networks, which results in higher hyperparameters, requires extreme fine-tuning, and is very unstable.

The domain and task-based approach for neural networks is implemented to remove known bias and variations from the feature representations by using the joint learning and unlearning algorithm (Alvi, Zisserman, and Nellåker, 2018). In this algorithm, they used a joint loss function which includes softmax loss for classifier prediction and cross-entropy loss between classifier output and uniform distribution for the unlearning of spurious variations. Another way of using a joint loss function to include distance correlation in deep learning is

explored by R. Wang, A.-H. Karimi, and Ghodsi, 2018. This study used autoencoders with distance correlation as an objective function for dimensionality reduction. By maximizing the distance correlation loss function, autoencoders were able to extract high-quality latent features representation, and it was also easily scalable to large high-dimensional datasets.

Our method, unlike previous works, focuses on explicitly mitigating bias in a simple, stable, and more effective way. The optimization used in this study does not rely on min–max optimization or adversarial optimization which are unstable. We also went further to show that the method can be generalized across different dataset sizes and dimensions, domains, and biases.

3.4 Methodology

Our proposed method focuses on using distance correlation in the objective function to decorrelate bias from features learned by CNN and ANN architectures. To generalize our proposed method across different domains, we used different datasets with various biases and also implemented different architectures. This opens up new opportunities to utilize this proposed approach across different deep learning or neural network architecture to mitigate different types of biases.

3.4.1 Distance correlation

Distance correlation measures not only linear, but also non-linear dependencies between two random variables $B_{1,\ldots,p}$ and $F_{1,\ldots,p}$, unlike the Pearson correlation coefficient (Lee Rodgers and Nicewander, 1988) which measures only linear dependencies. In our proposed approach, B is the dataset bias variable whereas F is features extracted from ANN and CNN and p is the total number of samples. The distance correlation is the square root of:

$$\mathcal{DC}^2(B,F) = \begin{cases} \frac{\mathcal{V}^2(B,F)}{\sqrt{\mathcal{V}^2(B,B)\mathcal{V}^2(F,F)}} & \text{if } \mathcal{V}^2(B,B)\mathcal{V}^2(F,F) > 0 \\ 0 & \text{else } 0 \end{cases} \tag{3.1}$$

where $DC^2(B,F)$ varies between 0 and 1 and indicates that variables B and F have dependencies, and $DC(B,F) = 0$ only when the variables B and F are independent. $v^2(B,F)$ is the distance covariance between a pair of variables and $v^2(B,B)$, $v^2(F,F)$ is the distance variance as defined in Székely, Rizzo, and Bakirov, 2007. The distance covariance is normalized by the distance variances.

3.4.2 Decorrelation in Objective Function

In our study, we use the squared distance correlation as a decorrelation function. This function is minimized to decorrelate features learned by the networks from the biases. This means that we want to find parameters of the network, such as F features, have a minimal distance correlation with the B bias variable. We added the decorrelation function term to the standard objective function. The objective function is given as:

$$J(\theta) = \min_{\theta}(1-\lambda)L(Y,\hat{Y}) + \lambda\mathcal{DC}^2(B,F) \tag{3.2}$$

The regular loss function (L) in Equation (3.2) could be binary cross-entropy, softmax loss, or mean-squared error depending on the nature of the tasks. The λ in the objective function is a hyperparameter that controls the relative importance of the decorrelation function in relation to the loss function. Y and $\hat{Y}$ are true and classifier outputs, respectively, whereas B is bias variable and F is features extracted from the model. Optimizing the combination of these two losses not only helps to mitigate bias but also tries to achieve higher classification accuracy. Depending on the size of the dataset and the overfitting issue, one can also add a regularizer L2 loss function in the objective function for weight decay purposes.

3.4.3 DcANN and DcCNN

ANNs are suited for modeling complex small datasets. For DcANN, we use the same architecture used in ANN which consists of an input layer, multiple hidden layers $h(1, \ldots, l)$, and an output layer with only the difference of using hidden layer output values as feature F in decorrelation loss function. The other input to the decorrelation loss function is bias B which can be N-dimensional and include more than one bias type. The framework of our model DcANN is shown in Figure 3.1. The output of first hidden layer (Patil, 2013) is given by:

$$\begin{aligned} h_j^1 &= \sum_{j=1}^{p} \mathrm{w}_{ij}^1 \, \mathrm{x}_i + b_j^1 \text{ where } j = 1, .., s \\ \mathrm{F} &= h^{1,\ldots,l} \end{aligned} \tag{3.3}$$

Here x is the input size of p and after applying a transfer function to $f(h_j^1)$ becomes the output of the first hidden layer, whereas w and b are weights and biases, i.e., θ parameters of the neural network. The variable (s) denotes the total number of hidden units in the first hidden layer and (l) denotes the total number of layers. We use the output of the first hidden layer h^1 as input F to our decorrelation function to reduce the dependencies of these output values on biases. Of course, it may be more appropriate to use just one or combinations of other layer outputs depending on the types of applications.

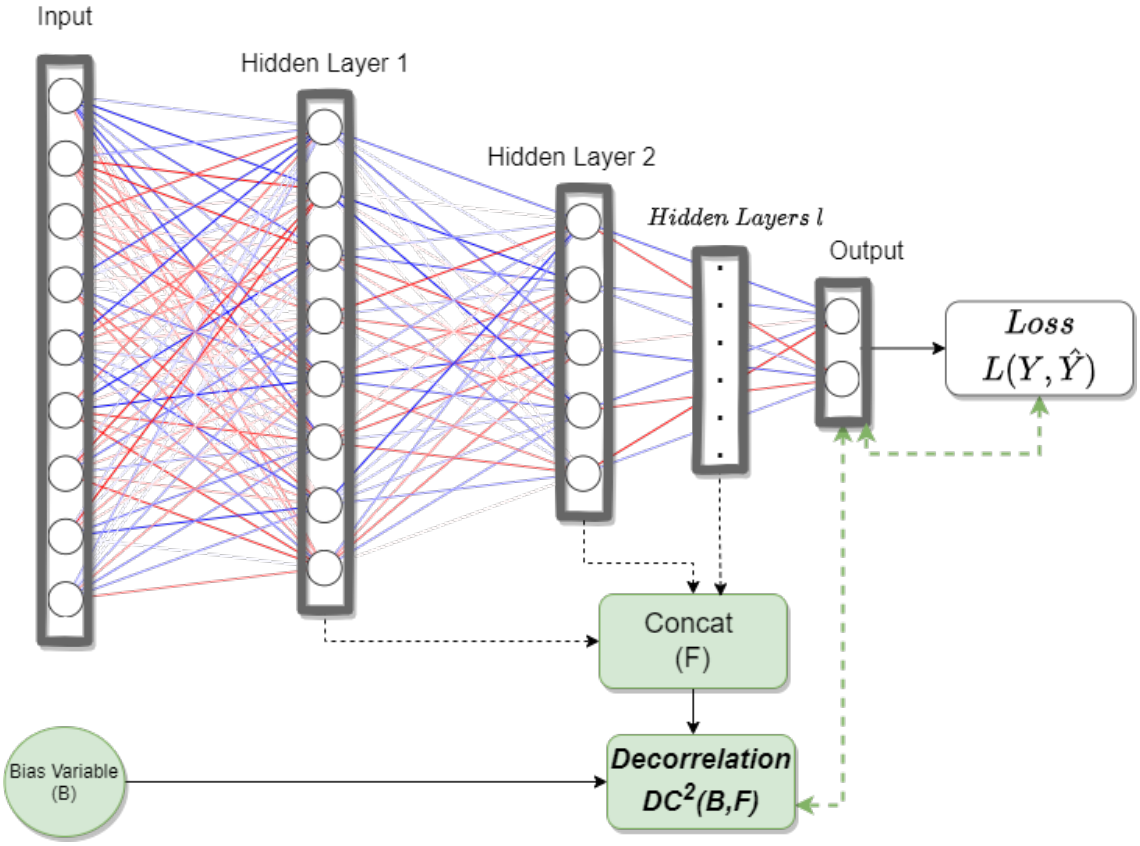


Figure 3.1 Proposed DcANN architecture: Black dashed lines denote the output of hidden layers l which are combined together to represent learned features. Green dashed lines indicate the start of the learning process where backward arrows show back-propagation using their respective gradient values while forward arrows show forward paths with updated parameters. Network parameters are updated as per the objective function.

For DcCNN or DcDNN, we follow the same technique used in ANN. We implemented decorrelation loss function and applied either a CNN or DNN architecture. We propose our DcCNN architecture as in Figure 3.2. The output of the first convolutional layer (Patil, 2013) is given by:

$$\begin{aligned} \mathrm{Z}^1 &= \mathrm{W}^1 * \mathrm{X} \\ \mathrm{F} &= \mathrm{Z}^{1,\ldots,l} \end{aligned} \tag{3.4}$$

In Equation (3.4), X is the two-dimensional or three-dimensional input size of p. $*$ denotes the convolution operation. We apply a transfer function and max-pooling to $\mathrm{Z}^{1,\ldots,l}$ to the

output of convolutional layers and concatenate them together to use as features for the decorrelation function. We use the first layer output Z^1 as input F to our decorrelation function to reduce the dependencies of these output values on biases. As in the DcANN case, using just one or combinations of other layer outputs as features might be more beneficial depending on the complexity of task.

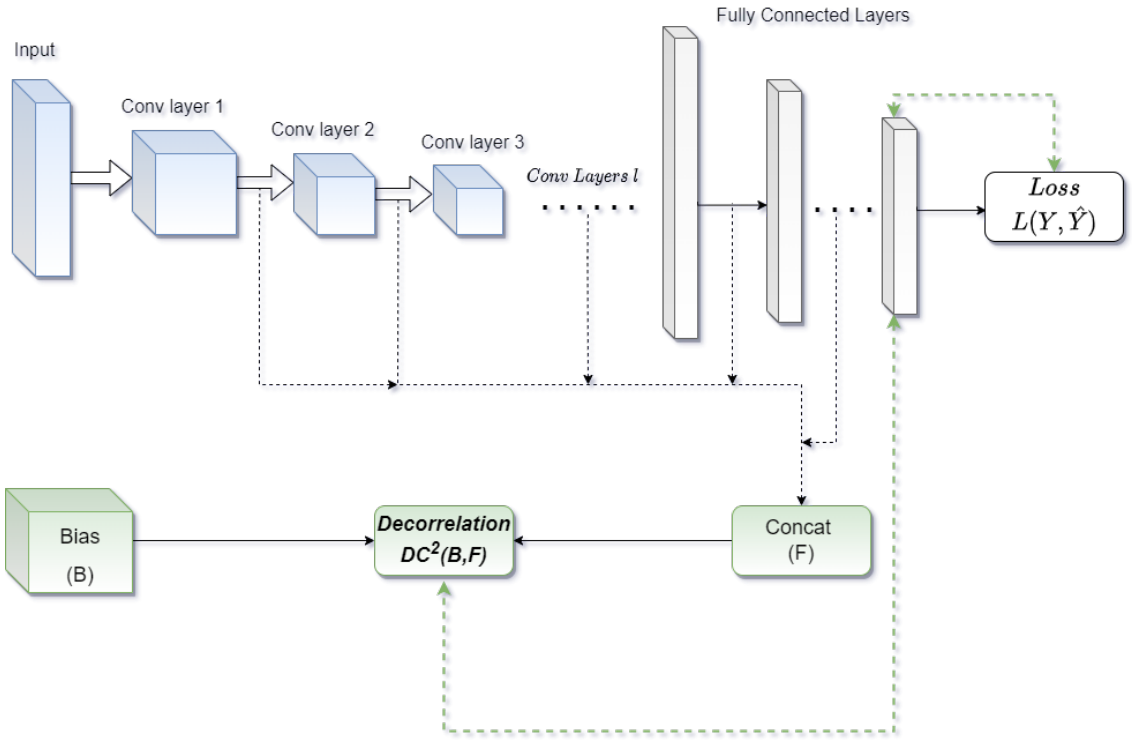


Figure 3.2 Proposed DcDNN architecture. Black dashed lines denote the output of convolutional layers l which are combined together to represent learned features. Green dashed lines indicate the start of the learning process where backward arrows show back-propagation using their respective gradient values while forward arrows show forward paths with updated parameters. Network parameters are updated as per the objective function.

3.4.4 Experimental Setup

In order to evaluate our proposed generic method, we explore five different scenarios and different types of biases. To validate our proposed approach, we utilize a simulated biased dataset (Adeli et al., 2021) generated specifically to check the performance of the model in

mitigating the bias. Datapoints are generated using four Gaussians whose magnitudes m_1 and m_2 are controlled by sampling from two different uniform distributions to classify into two groups. We implement three layers of 3×3 convolutions followed by tanh activation and max-pooling. This is followed by one hidden layer with 16 dimensions. The output of the last convolutional layer is used as feature F, whereas m_2 is a bias variable B since we assume m_1 is the main reason for discrimination between two groups. We use a mini-batch size of 256 and the hyperparameter (λ) of 0.7.

We consider commonly used standard datasets such as the German credit dataset and UCI adult dataset (Dua and Graff, 2017) which have been examined for biases. We consider age as a bias variable for the German credit datasets and gender as a bias variable for the UCI Adult dataset as shown in Table 3.1. The basic three-layer and two-layer DcANN model is implemented for the adult and German datasets, respectively, and compared with existing mitigation algorithms. For the German dataset, there are two layers with 50 and 10 dimensions with ReLU activations except the last layer with sigmoid activation. We use a mini-batch size of 100 with a dropout of 0.5 and the hyperparameter (λ) of 0.9. Detailed analysis of λ is give in Section 3.5. The regularization weight decay parameter is set to 0.05. For the adult dataset, we construct three layers with 200, 100, and 50 dimensions. The rest of the configurations used for the adult dataset is the same as for the German dataset except for the mini-batch size of 1024. The output of the first hidden layer is used as feature F, whereas age and gender are used as bias variable B for the German and adult dataset, respectively.

Table 3.1 Description of German Credit and Adult Datasets.

Datasets	**Bias Variable**	**Class Labels**
German Credit	Age	Good and Bad Credit
Adult Data	Gender	Income: >50K and ≤50K

We also use the MNIST image dataset (LeCun and Cortes, 2010) to check the performance of the proposed approach of DcCNN. Since the dataset is grayscale images with no bias and specifically designed for digit classification, we decided to utilize (B. Kim et al., 2019) the approach of intentionally planting color bias in the MNIST dataset. A few examples from the color-biased MNIST dataset are shown in Figure 3.3a. The training dataset consists of colored digits which are randomly sampled from the normal distribution of the corresponding mean and variance. So, the ten colors with their mean color value are assigned to each digit. The variance values such as 0.02, 0.03, 0.035, 0.045, 0.05 are also explored in this experiment. The smaller variance value means more color bias. These variance values controlled the amount of color bias in the training dataset. The testing dataset is unbiased. Colors are assigned randomly to each digit in the testing dataset. For this dataset, we apply the DcCNN method and implemented the same network architecture as in B. Kim et al., 2019, i.e., four convolution layers, followed by average pooling. We use softmax loss and decorrelation loss with $\lambda = 0.9$. For decorrrelation loss, we use the output of the first convolutional layer as features F whereas mean RGB color values are used as bias B. RMSprop optimizer with a mini-batch size of 1200 and a learning rate (lr) scheduler with an initial lr of 0.01 with a decay of 0.5 is used.

Another way of adding color bias in the MNIST dataset (Arjovsky et al., 2019) is dividing the dataset to predict binary labels where 0–4 digits are assigned as labels 0 and 5–9 digits are assigned as label one and then flipping the label with a 25% probability and color with

probability value which depends on the training environment. We combine these two training environments in one for our proposed method. According to labels in the training dataset, digit groups (i.e., 0–4 digits group and 5–9 digits group) are assigned red or green colors in a way that is strongly correlated with label 0 and label 1. For the testing dataset, the direction of correlation is changed; for example, if label 0 is red, then in the testing dataset, label 0 is green. We term this as the reversed color-biased MNIST dataset since the relation with bias variable in training and the testing dataset is exactly opposite. Figure 3.3b shows some of the samples taken from the reversed color-biased MNIST dataset. We apply DcCNN with CNN architecture by using two convolution layers and two fully connected layers. We use the same batch size and optimizer as in the color-biased MNIST dataset. The network is trained with $\lambda = 0.99$ and a learning rate (lr) scheduler with an initial lr of 0.001 with a decay of 0.5. In addition to this, an optimizer weight decay of 0.005 is used. Similar to the color-biased MNIST dataset, the output of the first convolutional layer is used to reduce their association with mean RGB color values.

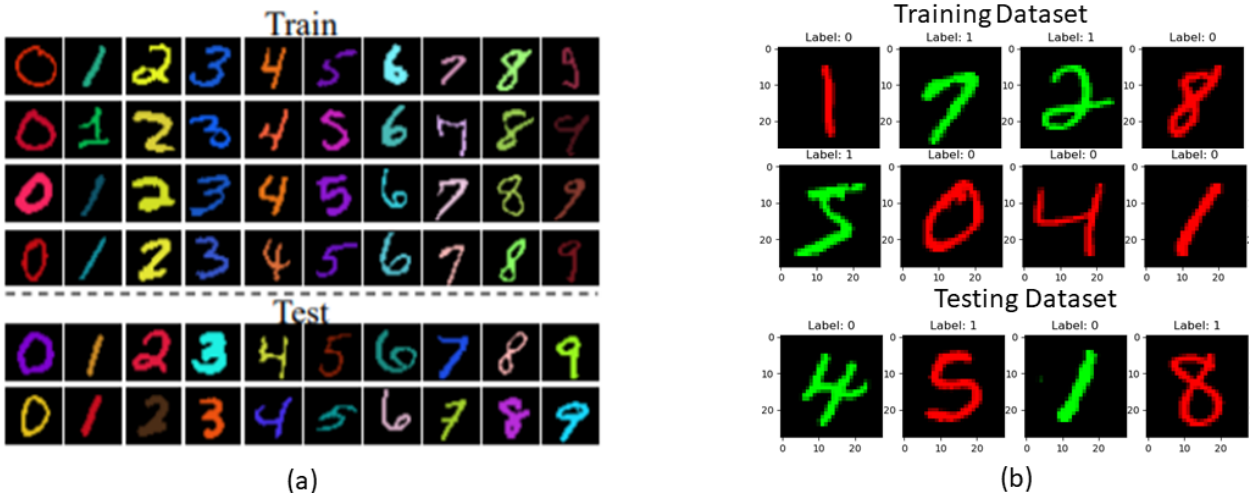


Figure 3.3 Colored MNIST Training and Testing Datasets Examples with color bias: (**a**) Some image examples of color-biased MNIST dataset - Modified MNIST dataset with a color bias for each digit. Taken from B. Kim et al., 2019 (**b**) Some image examples of reversed color-biased MNIST dataset - Binary group based color bias prepared for IRM (Arjovsky et al., 2019).

We implemented all our DcCNNs and DcANNs from scratch in python on the AWS Deep

Learning AMIs (*Deep learning ami - Developer Guide* n.d.) to accelerate deep learning in the cloud, at any scale using the TensorFlow platform (Martín Abadi et al., 2015) and cuDNN library (Chetlur et al., 2014). An Amazon EC2 P2 Instance is used to train the dataset using deep learning. P2 instances provide eight high-speed GPUs, parallel processing cores, and single and double-precision floating-point performance to speed up the training processes.

3.5 Experimental Results

3.5.1 Simulated Dataset

To validate the performance of our proposed approach DcCNN, we apply our method and baseline to simulated dataset to mitigate bias planted in the dataset. The baseline model is trained with $\lambda = 0$ where decorrelation is not performed to mitigate the bias m_2. Figure 3.4 shows tSNE plots of learned features for baseline as well as DcCNN models. The color bar indicates the value of bias variable m_2. From the plots, we can see that there is a correlation between features and m_2 for the baseline model whereas features learned by DcCNN have a roughly uniform distribution of features across all values of m_2 indicating no dependency of features on bias m_2. This indicates that our proposed DcCNN successfully mitigates the bias present in the dataset.

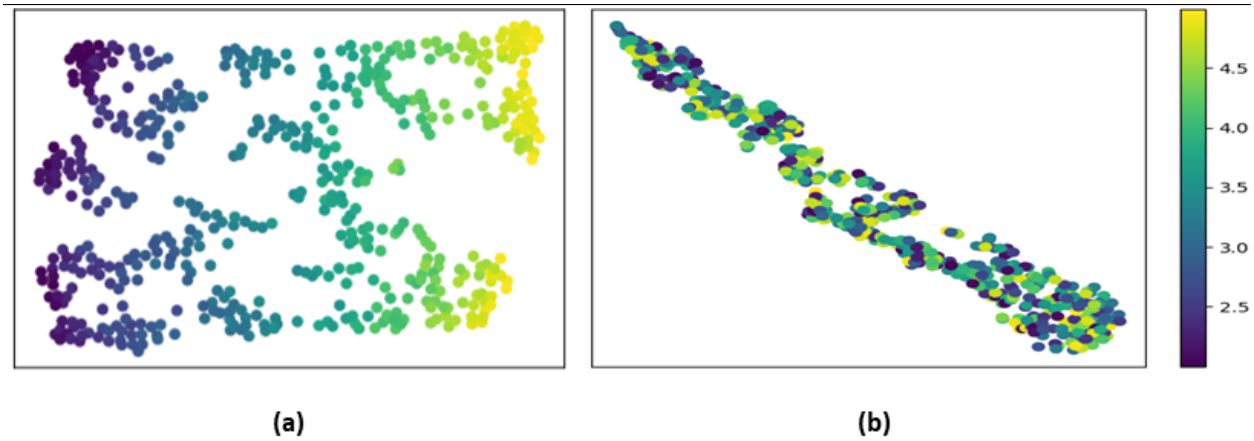


Figure 3.4 tSNE plots of learned features for different methods: (**a**) For Baseline CNN model (Adeli et al., 2021) (**b**) For DcCNN model.

We also plotted the decorrelation function against iterations in Figure 3.5 to compare the performance of models in regards to reducing the statistical dependence between features and bias variables. This figure shows the unsmoothed distance correlation values in light blue and light orange colors. In contrast, dark blue and dark orange colors indicate the smoothed distance correlation values which are calculated using exponential moving average. Smoothing is used to observe the overall trend. It shows that the distance correlation between features and bias variable decreases as the number of iterations increases for DcCNN as opposed to the baseline model.

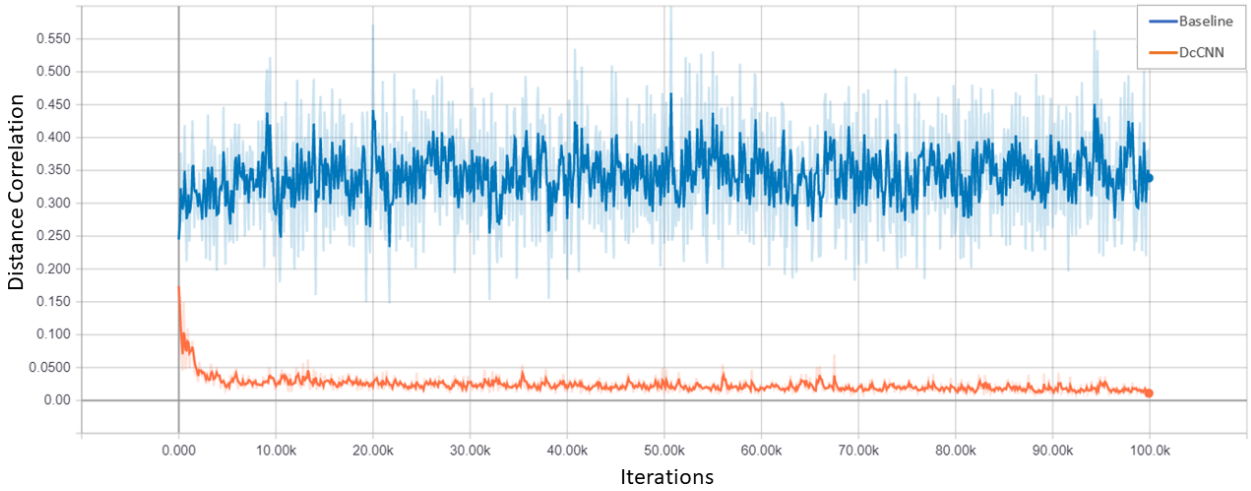


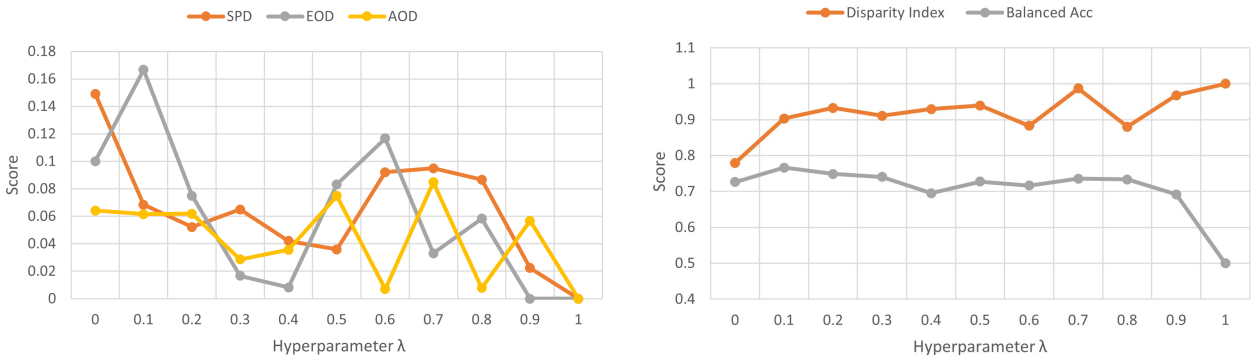
Figure 3.5 Distance correlation between learned features and bias m_2 for the simulated dataset.

3.5.2 Age Biased German Dataset

For the age-biased German dataset, we train a model to classify credit score levels and to reduce the age bias of the baseline model. Furthermore, we analyze the performance of different bias mitigation approaches using fairness metrics.

3.5.2.1 Hyperparamter λ Analysis

The hyperparameter $\lambda > 0$ defines the strength or relative importance of the decorrelation function in relation to the loss function, and hence, it plays a crucial role in deciding the importance of the decorrelation task for bias mitigation. The higher value of λ means features learned by the network are highly decorrelated with bias which might impact the ability of the network to do certain tasks such as classification. A lower value of λ would mean less bias reduction relative to a higher value. The better performance is achieved by trying different values of λ depending upon the requirement of applications.



(a) **(b)**

Figure 3.6 Fairness scores and accuracies for different values of λ: (**a**) SPD, EOD and AOD scores Vs λ (**b**) DI scores and Accuracies Vs λ.

Figure 3.6 plots fairness scores and accuracies for different values of λ for the age-biased German dataset. We can infer from the plot Figure 3.6a that for the German dataset, SPD, EOD, and AOD values somewhat decrease as the value of λ increases. The main reason behind this is the decorrelation between features learned by the model and age bias in the German dataset increases as the value of λ increases. In Figure 3.6b, we observe that DI

increases as the value of λ increases whereas there is a slight drop in balanced accuracy as the value of λ increases. In order to achieve maximum fairness or bias reduction, we select $\lambda = 0.9$ since the lowest values of SPD and EOD and the highest value of DI are observed for the same. The values for λ may vary for different tasks depending upon network architecture and the complexity of the task.

3.5.2.2 Evaluation Fairness metrics

In general, to access the performance of the fair model which indicates no discrimination against the bias or protected attribute, we report widely-used fairness metrics for bias mitigation methods. These protected attributes include gender, age, color, race, religion, etc. In this study, we will focus on metrics to evaluate group fairness based on age and gender. The fair model will provide predictions that are not influenced by protected attributes. Four metrics have been selected to evaluate the bias mitigation ability of the proposed approach since testing datasets of age-biased German and gender-biased adult datasets are not unbiased and contain age and gender bias, respectively.

We use the Demographic Parity or Statistical Parity Difference (SPD) fairness metric (Kusner et al., 2017) to check if decisions are independent of protected attributes. Disparate Impact (DI) (M. Feldman et al., 2015) is the same as SPD but formulated as proportion, respectively. SPD and DI are given as:

$$\text{SPD} \;=\; \text{P}(\hat{Y} = 1|B = 0) - \text{P}(\hat{Y} = 1|B = 1) \tag{3.5}$$

$$\text{DI} \;=\; \frac{\text{P}(\hat{Y} = 1|B = 1)}{\text{P}(\hat{Y} = 1|B = 0)} \tag{3.6}$$

The Equality of Odds Difference (EOD) metric (X. Li, Cui, et al., 2021) is used for separation, i.e., to check the independence of the decision and protected attribute separately

for individuals. EOD measures the difference in true positive rates for protected and unprotected groups whereas Average Odds Difference (AOD) (X. Li, Cui, et al., 2021) measures the difference between the true-positive rates as well as false-positive rates for each group. EOD and AOD are formulated as:

$$\text{EOD} \;=\; \text{TPR}_{B=0} - \text{TPR}_{B=1} \tag{3.7}$$

$$\text{AOD} \;=\; 0.5 * [(\text{FPR}_{B=0} - \text{FPR}_{B=1}) + (\text{TPR}_{B=0} - \text{TPR}_{B=1})] \tag{3.8}$$

In the above Equations (3.5)–(3.8), B is bias variable which can be age or gender and is 0 when it represents a privileged group and 1 when it represents an unprivileged group. P is the classification probability whereas $\hat{Y}$ is model prediction. FPR and TPR represent a false positive rate and true positive rate, respectively. Lower values of SPD, EOD, and AOD indicate less bias, and higher values of DI show more fairness and less bias. Balanced accuracy is calculated as the average of sensitivity and specificity.

3.5.2.3 Comparative Evaluations

We compare the performance of our proposed DcANN with other existing pre-processing methods as shown in Table 3.2 on a holdout testing dataset. We report all fairness metrics and balanced accuracies for all the methods. The existing methods such as the baseline model, reweighing, optimized pre-processing, and adversarial debiasing are implemented using the AIF360 (Bellamy et al., 2018) open source Python toolkit. The baseline model is simple the ANN model where unprocessed data are used and bias mitigation method is not applied. Baseline DcANN (B-DcANN) has the same architecture as DcANN but with $\lambda = 0$ which means decorrelation is not present for bias mitigation. The main difference between the baseline model and B-DcANN is that B-DcANN uses regularization parameters such as

dropout and weight decay. We notice that regularization helps the model in achieving higher accuracy as can be seen in Table 3.2.

Table 3.2 Fairness scores and Balanced accuracies of predictions for Age Biased German Dataset for different methods.

Methods	**SPD**	**EOD**	**AOD**	**DI**	**BA**
Baseline	-0.3162	-0.318	-0.2876	0.3112	0.6534
Reweighing	-0.2049	-0.2318	-0.2016	0.6229	0.6687
Optimized pre	-0.0351	0.0254	-0.0639	0.9421	0.6872
Advers-Debias	0.0713	0.0393	0.0931	**1.0834**	0.6633
B-DcANN (λ=0)	-0.1491	-0.1	-0.0642	0.7798	0.7262
DcANN	**-0.03144**	**-0.03918**	**0.06223**	**0.95927**	**0.7093**

As we can see, our proposed method DcANN significantly reduces SPD, EOD, and AOD as compared to other methods without compromising the accuracy. The DI is higher for our DcANN method compared to other methods except for adversarial debiasing. However, as we can see from other fairness metrics and especially accuracy, adversarial debiasing does not provide a significant reduction in bias without significantly comprising accuracy when compared to our DcANN method. In general, we can say that our model performs best on all fairness metrics.

3.5.3 Gender Biased Adult Dataset

The UCI adult dataset is used to classify income levels and we consider gender as bias. We use the same evaluation metrics as defined in Section 3.5.2.2 and compare with the same existing mitigation methods mentioned in Section 3.5.2.3 on a holdout testing dataset for the gender-biased adult dataset. We also compare the performance of our model with the method of fusion introduced in T. Feldman and Peake, 2021. Authors used the fusion of different combinations of existing bias mitigation methods such as Disparate Impact Remover (DIR) (M. Feldman et al., 2015), Adversarial Debiasing (Advers-Debias) (B. H. Zhang, Lemoine, and Mitchell, 2018), and Calibrated Equalized Odds (CEO) (Pleiss et al., 2017) to provide end-to-end bias mitigation. We use their best method for the comparison.

The results for each method are displayed in Table 3.3. The results show that the DcANN model achieves the lowest EOD and AOD and highest DI amongst all methods while the balanced accuracy slightly decreases. The SPD and EOD scores of Adversarial Debiasing are almost similar to the DcANN method. However, as it is seen in the German dataset case, Adversarial Debiasing helps to reduce bias but at the cost of a significant reduction in the accuracy. The fusion model (IR + Advers-Debias + CEO) has the lowest SPD and highest accuracy but it does not perform well on other fairness metrics. Thus, the results suggest that DcANN reduces bias fairly by achieving good results on almost all fairness metrics without significantly compromising balanced accuracy.

Table 3.3 Fairness scores and balanced accuracies of predictions for gender-biased adult dataset for different models.

Methods	**SPD**	**EOD**	**AOD**	**DI**	**BA**
Baseline	-0.3752	-0.3716	-0.3258	0.2876	0.7472
Reweighing	-0.2924	-0.3815	-0.3234	0.3831	0.7110
Optimized pre	-0.2144	-0.1991	-0.1945	0.568	0.7231
Advers-Debias	**-0.0876**	**-0.0592**	-0.0373	0.5775	0.6656
DIR+Advers-Debias+CEO	**-0.0301**	0.0785	0.051	-	**0.8113**
B-DcANN (λ=0)	-0.3394	-0.1545	-0.1965	0.3025	0.8226
DcANN	**-0.0964**	**0.0657**	**0.0325**	**0.8063**	**0.7747**

3.5.4 Color Biased MNIST Dataset

The color introduced in the dataset misled the model while performing the digit classification task. The model learns the color features instead of learning digit features to categorize the digits. We use the DcCNN model to remove color bias from features learned by the network. The performance of the DcCNN model is compared with existing methods such as Adversarial Training (B. Kim et al., 2019) and the Blind Eye method (Alvi, Zisserman, and Nellåker, 2018). Adversarial Training without Pre-trained model (Advers Training-no Pretrain) is the

same as the Adversarial Training model but it is trained from scratch without using any pre-trained parameters. The baseline model is trained with no decorrelation function, i.e., $\lambda = 0$ which means bias mitigation is not performed. The results are included in the Table 3.4 and the variance values (Var) control the amount of color bias in the dataset.

Table 3.4 Comparison of accuracies for color-biased MNIST dataset for different values of variances among existing methods. Results are calculated on the testing dataset.

Methods	**Var=0.02**	**Var=0.03**	**Var=0.035**	**Var=0.045**	**Var=0.05**
Baseline	0.4055	0.5996	0.6626	0.7973	0.845
BlindEye	0.6741	0.7883	0.8203	0.8927	0.9159
Advers Training	**0.8185**	**0.9137**	**0.9306**	**0.9555**	**0.9618**
Advers Training-no Pretrain	0.7336	0.8516	0.8781	0.9277	0.9429
DcCNN	**0.8100**	**0.8910**	**0.9250**	**0.9500**	**0.9604**

The results show that our model DcCNN achieves almost the same accuracies as adversarial training and is better than all other methods for all values of variances. However, in order to achieve the same results using the Adversarial Training method, we need to use pre-trained parameters. If we don't utilize a pre-trained model and train the model from scratch, then there is a lot of fluctuation in the accuracies. In fact, as shown in Table 3.4, accuracies dropped by a significant amount for low variance values for Adversarial Training without using the pre-trained model. This indicates that the Adversarial training algorithm

is very unstable, and it also requires a lot of fine-tuning. Thus, the DcCNN model is simple and requires less fine-tuning since it has only one hyperparameter (λ) for fine-tuning to successfully mitigate the color bias while achieving high performance.

3.5.5 Reversed Color Biased MNIST Dataset

To analyze the reversed effect of bias and the proposed approach, we use the reversed color-biased MNIST dataset where the bias present in the testing dataset is exactly the opposite of the bias present in the training dataset. This is to validate how well the proposed approach generalizes to the unseen test dataset. In the paper, Arjovsky et al., 2019 used an Invariant Risk Minimization (IRM) causal-invariant based approach in multiple training environments to promote out-of-distribution (OOD) generalization by assuming the different environments share the same underlying structural equation model. An ANN classifier is implemented to achieve the same. However, for comparison purposes, we use the same CNN architecture as mentioned in Section 3.4.4 for all existing methods. Empirical Risk Minimization (ERM) combines the data from all the training environments and uses all features which is similar to the baseline model principle.

Table 3.5 Comparison of Accuracies for Reversed Color Biased MNIST Dataset among different methods. Results are calculated on testing dataset.

Methods	**Accuracy**
Baseline: ERM	0.1115
IRM	0.6208
DcCNN	**0.6630**

The Table 3.5 presents the comparison where the ERM method classifies digits based on color bias and hence the lowest accuracy whereas IRM and DcCNN remove the color bias

information from the features and classify based on features relevant to digits. The results show that DcCNN achieved an accuracy of 66.30% which is the best across all methods and can successfully mitigate the color bias by learning more digit-relevant features.

3.6 Discussion

One of the crucial aspects of our proposed method is to mitigate bias without compromising the performance of the model by optimizing the decorrelation loss along with loss related to the task and tuning hyperparameter (λ). The choice of λ depends on the complexity of the task and network architecture. The results from all five datasets outperformed the traditional approaches in mitigating different types of biases. This higher performance indicates that the DcCNN and DcANN models significantly mitigate the bias and present relevant feature information to the network compared to other methods. Further, this also demonstrates the generalization ability of the proposed approach across different domains for bias mitigation.

In Figure 3.5, we observe the oscillations in distance correlation values. From our experiments, we verify that these oscillations are due to the size of mini-batches. Increasing the batch size not only reduces the oscillations but also leads to an unbiased estimate of distance correlation. We also notice that regularization such as dropout and weight decay helps the baseline model to improve its performance. As for our proposed approach, the input bias variable also plays a vital role and it should clearly define the bias present in the data or task. For example, a possible concern is our proposed method might not show significant improvement for the domain adaptation tasks due to the lack of enough quantitative information about different domains and a limited number of domains. We apply our method for digit domain adaptation tasks (Ganin et al., 2016) where we use MNIST, USPS, SVHN, and synthetic numbers datasets as training and validation datasets. We evaluated the results on the MNIST-M dataset as a test dataset. However, we observed that the improvement using our approach is not that significant. For such cases, we simply recommend collecting and

using more domain-relevant information as a bias variable or using domain distributions as a bias variable.

3.7 Conclusions

The performance of deep learning mainly depends on the quality of data. Failure to account for the quality of data, e.g., biased data in deep learning can lead to erroneous decisions. We propose a new method based on the core idea of reducing the association between features learned by the ANN or CNN models and bias. Additionally, we evaluated proposed models, which we name the DcANN and DcCNN, on five different datasets with different biases such as age, gender, and color. The experimental results demonstrate that features learned by our models are statistically independent of biases or confounds present in the dataset. Our proposed method leverages the ability of the distance correlation function to decorrelate features from data bias without significantly impacting the performance of a network. Furthermore, we observe that our method also performs better than previous approaches to mitigate the bias. Our models are easy, simple, and require fewer hyperparameters to optimize compared to adversarial training. Thus, our models DcCNN and DcANN, despite having numerous methods to achieve bias mitigation, is a promising and effective novel method. Future work will investigate the use of DcDNN in the medical domain to mitigate bias or confounding effects or any irrelevant dependency issues. In addition, we plan to further evaluate the expansion of the proposed method by applying it to pre-trained models and to different types of data variations. Although we did not observe significant change in training times for all our proposed models in comparison with baseline models, we intend to perform a time complexity analysis in the future by measuring the whole training process in terms of training time as the number of dimensions, the complexity of tasks, and the number of layers increases.

Chapter Four

Learning Gene Regulatory Networks using Graph Granger Causality: Learning Granger Causal Relationships

4.1 Abstract

Interacting systems such as gene regulatory networks have the ability to respond to individual component changes, propagate these changes throughout the network, and affect the temporal trajectories of other network elements. Causality techniques are frequently employed to investigate the interconnection between variables in complex dynamical systems. However, the vast majority of causality models are rooted in regression techniques such as Vector Autoregression Models and Bootstrap Elastic net regression from Time Series framework, and there is very limited research in the space of deep learning, particularly graph neural networks. In this paper, we explore in more depth the concept of Granger causality in deep learning and propose Granger causality deep learning framework using graphs convolutions, LSTM, and nonlinear penalties for the objective of learning causal relationships between temporal elements in gene regulatory networks. The deep learning architecture proposed here for studying causality in dynamic networks has achieved high results on sim-

ulated networks as well as on more challenging Dream3 gene regulatory networks time-series datasets.

4.2 Introduction

The advances in the field of mRNA sequencing have been opening the doors for a wide range of applications in the research and medical communities. mRNA sequencing and analysis hold the potential to unlock great insights into disease causes and progression, carrier status, cancer, and infectious diseases research, as well as other fields such as agrigenomics. Understanding the causes that lead to changes in gene regulatory expression levels over time and across conditions is vital in studying the genesis, progression, and ultimately treatments of diseases.

Gene regulatory network inference methods seek to uncover these complex relationships among gene pathways and predict the consequences of perturbations. Existing causality frameworks focus on studying the associations between elements in an interacting system using statistical quantitative techniques that require apriori assumptions on the data. Moreover, these models are not able to fully capture the nonlinearities hidden in the latent variables, and the vast amounts of available data that are idiosyncratic to the mRNA analysis can generate high levels of false positives. Data science on the other hand can overcome these limitations while learning latent structural properties in connected networks. More specifically, graph neural networks have been proven an essential tool in studying networks, encoding and reconstructing structural characteristics of nodes, and predicting network linkage.

This work extends causality inference to graph neural networks and combines them with existing LSTM causality frameworks introduced by Tank, Covert, et al., 2018, with the aim of learning causal relation—ships from transcriptomal time series data based on Granger causality. Graph Granger Causality (GGC) framework proposed here also incorporates non-

linearities hidden in the data as weight initializers and model penalties in the unsupervised task of learning latent causal structure in temporal gene regulatory networks. The concept of node-node similarities in graph networks have been extensively used in graph neural networks by Henaff, Bruna, and LeCun, 2015; X. Chen and L. Huang, 2017; Tengfei Ma et al., 2018; Shuman, Ricaud, and Vandergheynst, 2016. In the field of causality, correlation-based similarity measures have been used by Ji et al., 2018 and Dong et al., 2017.

4.3 Related Work

Interacting biological systems such as neurons, proteins, and genes have a strong interdependence expressed through various mechanisms that propagate individual component perturbations across the entire network. However, the latent causal relationships between components of these systems are hidden. To understand network dynamics one must infer causal relations from the available time-based observational data (Wismüller et al., 2021). Time series causality inference quantifies the degree to which one variable evolution in time impacts another variable trajectory. Such a causal relationship can be translated into the ability of the first variable to explain or predict the the second variable (Wiener, 1956, Granger, 1969). A variable Z is considered Granger causal of variable X if the past values of $Z_{t,\dots,t-1}$ improves the prediction accuracy of future values of X_{t+1} when compared with a model in which $Z_{t,\dots,t-1}$ is not included.

In the gene regulatory networks (GRN) research, the Granger causality space has been mostly focused around regression techniques. For instance, Bootstrap Elastic net regression from Time Series (BETS) infers causal relationships in gene regulatory networks using elastic net regression and stability selection from bootstrapped samples. In addition to causal relationships prediction, BETS also infers the directionality of effect, namely whether causal effects are activating or inhibitory (Lu et al., 2021). Sliding Window Inference for Network Generation (SWING) is another time series windowed network inference approach based on

Granger causality. SWING identifies associations between genes by applying an ensemble of regression-based models over time-series gene expression data using flexible time-series windows (Finkle, J. J. Wu, and Bagheri, 2018). For each time window, SWING infers a ranked list of time-delayed causal edges between variables.

The alternative techniques used for Granger causality are based on prior information and conditional probabilities such as transfer entropy (Vicente et al., 2011) and models based on maximum likelihood estimation (Okatan, M. A. Wilson, and Brown, 2005). A more advanced method such as Gene regulatory networks on transfer entropy (GRNTE) is implemented to infer gene regulatory interactions by Castro et al., 2019. This method uses partial mutual information between pairs of genes and higher values of transfer entropy correspond to stronger interaction. Prior knowledge-driven Granger causality model (S. Yao, Yoo, and D. Yu, 2015) uses the prior knowledge as the directed or undirected weighted graph between the gene. This conditional Granger causality model also incorporates ridge regularization to resolve the problem of data size limitation.

More recently Wismüller et al., 2021, Ren, B. Li, and Han, 2020 and Siggiridou and Kugiumtzis, 2015 proposed causality frameworks that account for network nonlinearities. Large-scale nonlinear Granger causality (lsNGC) (Wismüller et al., 2021) models high dimensional limited-time series interval data with nonlinear dimensionality reduction techniques using radial basis functions to identify statistically significant casual relations. lsNGC is applied to functional Magnetic Resonance Imaging (fMRI) data to detect causality in brain tissues. Hilbert–Schmidt independence criterion Lasso Granger causality (HSIC-Lasso-GC) (Ren, B. Li, and Han, 2020) extracts nonlinear time series intimation using stationarity test and state-space reconstruction functions and applies a HSIC-Lasso model to learn the causality structure. Backward-in-Time Selection Conditional Granger Causality index (BTS-CGCI), defines Granger causality as the logarithm of the ratio of the error variances of leave-one-variable-out and the unrestricted ordinary least squares (OLS) models. BTS-CGCI incorporates the assumption that the variables at smaller lags are more explanatory to the

response variable than variables at larger lags. Thus conditioning on the variables at smaller lags the variables at larger lags may only enter the model if they have genuine contribution not already contained in the selected variables of smaller lags (Siggiridou and Kugiumtzis, 2015).

All previous models mentioned above either face problem of a data size limitation or are not able to capture linear dependencies or curse of dimensionality. Tank, A. et all. were the first to introduce a time series causality inference deep learning architecture. Their model titled Neural Granger Causality (NGC) combines fully connected layers and long-short term memory convolutions (LSTM) with group-lasso penalties to extract the Granger causal structure from network parameters without any supervised causality loss measures (Tank, Covert, et al., 2018). Our models build on this deep learning framework by integrating graph neural networks with LSTM convolutions to better capture network structure, connectivity, and time series interaction patterns in gene regulatory networks. We also extend the notion of penalties to include nonlinear similarities between variables, and to offer the model data-driven weights initialization. The framework of our model is shown in Figure 4.1.

4.4 Methodology

Graph Granger Causality (GGC) model combines graph neural networks over temporal data with Granger causality principle to measure the causal effects in a series of data points. Given a model with N variable of size p and time points T, Granger causality defines a causal relation between a pair of variables $N_{1i}, ...,_{Ti}$ and $N_{1j}, ...,_{Tj}$ by minimizing the reconstruction error of $M_{N_1,..,N_i,..,N_p}$ models. The model applies LSTM and graph convolutions over a set of variables expressed as a sequence of time points, by learning to reconstruct the sliding window in the time series.

Each model corresponds to one output variable, thus the number of variables $N_{1,...,p}$ dictates the total number of models $M_{N_{1,...,p}}$ the network will learn over as in Tank, Covert,

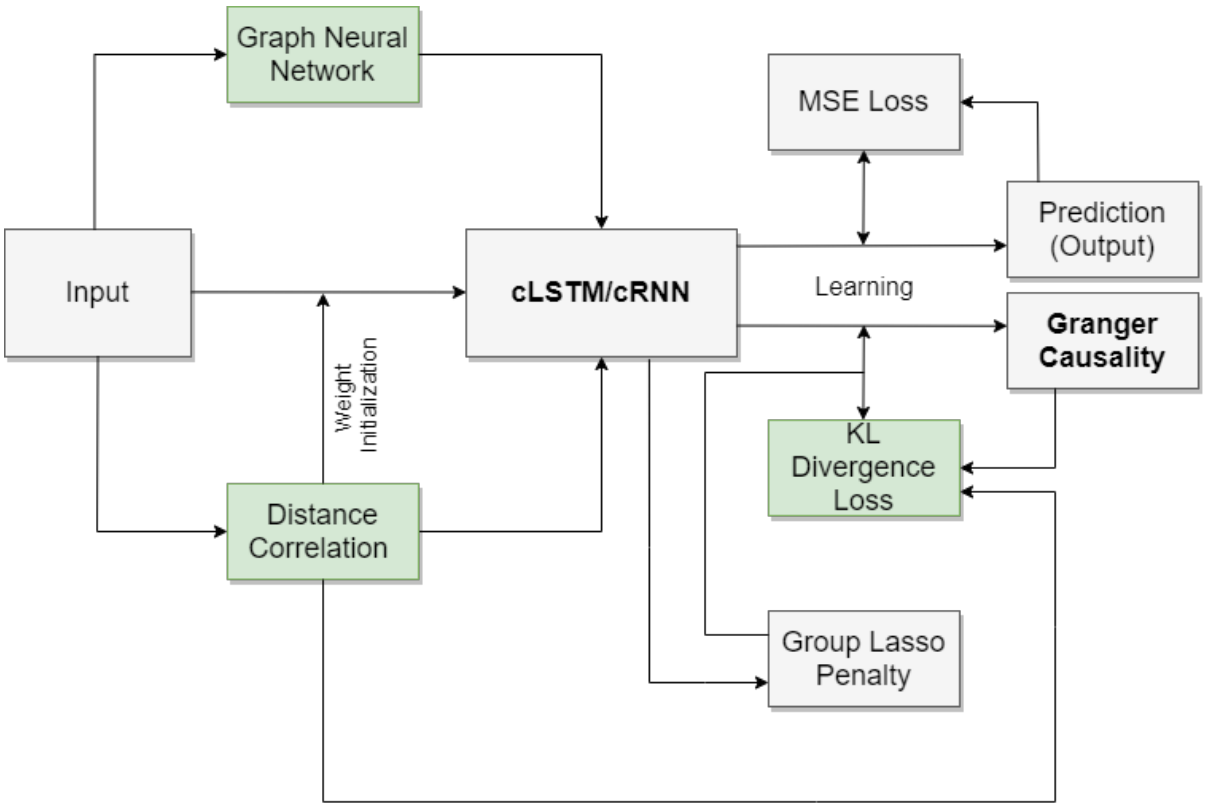


Figure 4.1 GGC Model per one predictor variable.

et al., 2018. Since recurrent models perform impressively for modeling time series data even with longer time series dependencies, we applied component-wise models with graph neural network to the recurrent neural network (RNN) and long-short term memory network (LSTM) and term those as Graph Granger Causality-component wise models such as GGC-cRNN and GGC-CLSTM models respectively. For each variable, the network convolves over the entire time-series and graph datasets and outputs the reconstruction values for one variable.

We also implemented a leave-one-out variation of the model (GGC-LOO), which includes an additional global model, thus in total the network having $M_{N_{1,\ldots,p}} + 1$ models. The global model computes the loss over all variables and predicts each variable, while the remaining M models masks each of the $N_1, ..., p$ variable at a time to predict each variable. If we train separate models for all variables except jth variable to predict each variable i and the loss of the global model is less than the loss for jth variable N_j of the masked models, then we can conclude that the masked variable N_j is causal of N_i since its presence in the network reduces the reconstruction error. The input to this model is two streams of data: a time series dataset and a graph adjacency matrix of the variables.

The graph dataset is defined as a weighted undirected graph $\mathrm{G} = (\mathcal{V},\mathrm{E_w})$, where $\mathcal{V} = \{\upsilon_1, ..., \upsilon_p\}$ is a set of vertices, and $\mathrm{E}_w = e_{\imath j}$ represents the set of edges. Each node in the graph $\upsilon_\imath \in \mathcal{V}$ corresponds to a gene and each weighted edge encodes the normalized gene expression values. The structure of the graph is represented by its symmetric adjacency matrix $\mathrm{A} \in \mathrm{R}^{p\times p}$, where $\mathrm{A}_{\imath j} = W$, ($W\imath j$ is the edge weight between vertices), if the two genes, $\imath j$, are expressed together otherwise $\mathrm{A}_{\imath j} = 0$.

The normalized adjacency matrix is defined as a weighted adjacency matrix derived from the time-series data matrix as follows:

$$\begin{aligned} \mathrm{A} &= N^T N \\ \mathrm{A}_{\imath j} &= \sum_{v=1}^{T} N_{iv} N_{vj} \text{ where } i,\, j = 1, .., p \end{aligned} \tag{4.1}$$

where $\mathrm{A}_{\imath j}$ represents the weights of the edges i and j if there is a common vertex v incident these edges. The edges are aggregated over all time points T . The resulting adjacency matrix values are then normalized.

The models consists of 3 layers: LSTM layer, GNN layer, and a Conv1D unit. The LSTM layer initiates its initial hidden states to the distance correlation of the data, which measures the nonlinear dependencies between variables and is computed as:

$$\mathcal{C}^2(X,Y) = \begin{cases} \frac{\mathcal{V}^2(X,Y)}{\sqrt{\mathcal{V}^2(X,X)\mathcal{V}^2(Y,Y)}} & \text{if } \mathcal{V}^2(x,x)\mathcal{V}^2(y,y) > 0 \\ 0 & \text{else } 0 \end{cases} \tag{4.2}$$

By taking the square root of $C^2(X,Y)$, distance correlation $C(X,Y)$ is calculated. $C(X,Y)$ is between 0 and 1 when the variables have a level of interdependence, and $C(X,Y) = 0$ when the variables are independent. $v^2(x,y)$ is the distance covariance between a pair of variables as defined in Székely, Rizzo, and Bakirov, 2007. All other LSTM components are as per Sak, Senior, and Beaufays, 2014.

Graph layer integrates the structural properties of the network with the LSTM embeddings, which are then passed to the Conv1D layer. This last layer concatenates the embeddings of all variables into one vector (which is representative of the gene time-series) the respective model is being optimized over. The complete model takes the following form:

$$h_i^{t(k)} \;=\; \sum_{j=1}^{p} \mathrm{A}_{\imath j}\; \mathrm{h}_j^{t(k-1)} \text{ where } i = 1, .., p \tag{4.3}$$

where $h_i^{t(k)}$ is the k window of time t output for i gene in the network, $\mathrm{h}_j^{t(k-1)}$ is the LSTM hidden state for time t window of $k-1$, $A_{\imath j}$ is the graph adjacency matrix described

in equation 4.1, and $\sum_{j=1}^{p}$ is summation over all variables which is nothing but genes. This last layer is convolution layer which condenses the dimensionality of the data from N variables to 1 variable window series that is being reconstructed for each k window.

In the optimization task, the loss function applies to mean squared error(MSE) between network predictions and output values for each time-series window and for each variable. Similar to Tank, Covert, et al., 2018, we optimize the models simultaneously by adding group lasso penalty. To further optimize the networks to learn the latent Granger structure, we add two additional loss constraints in the form of the distance correlation matrix described in equation 4.2, and Triplet loss (Vassileios Balntas and Mikolajczyk, 2016). Distance correlation also serves as the LSTM hidden input weights initialization as discussed earlier. Triplet loss minimizes the distance between similar samples and increases the distance between negative and positive samples. The complete loss function is computed as:

$$L = \min_{w} \sum_{t=2}^{T} \left(y_t^k - \hat{y}_t^k\right)^2 + \lambda_1 \sum_{j=1}^{p} ||W_1^j||_2^2 + \lambda_2 \sum_{n=1}^{p} y_n \cdot (\log y_n - x_n) + max(d(a, po) - d(a, ne) + R) \tag{4.4}$$

where $\sum \left(y_t^k - \hat{y}_t^k\right)^2$ is the MSE loss, $\lambda \sum ||W_1^j||_2^2$ is the Lasso penalty, $y_n \left(\log\ y_n - x_n\right)$ is the Kullback-Leibler divergence between the distance correlation and the reconstructed latent Granger causality matrix. $Max(d(a, po) - d(a, ne) + R)$ defines the Triplet loss, with $d(x_i, y_i) = ||x_i - y_i||_p$ measuring the relative similarity between the anchor a, positive examples po and negative examples ne, and $R = 1$ is the soft margin distance between positive and negative examples. λ_1 is regularization parameters and controls the sparsity of Granger causal connections whereas λ_2 controls tradoff between observed and predicted non-linear dependence.

The time-series dataset for these proposed models is segmented into p X p sliding windows with a lag of 1 timepoint for each window. For each variable in the dataset, a model convolves over all variables and all windows and outputs the reconstruction time-series values for the

next window in the series.

4.5 Experimental Results

We conducted experiments on 1 simulated Lorenz-96 dataset (A. Karimi and Paul, 2010) and 5 gene regulatory network datasets that were published as part of the Dream3 challenges (*Dream Challenges* n.d.). The Lorenz-96 model is used to simulate non-linear time series data with p=10 and data time series length, T=1000. The forcing constant used is 10. The Dream3 simulated data mimics gene expression and regulation dynamics, encoding latent nonlinear causal relationships, as described in Tank, Covert, et al., 2018. The Dream challenge datasets include three sets of data, each with two E. Coli data sets and three Yeast networks. This dataset includes 10 and 100 genes spanning over 966 timepoints each and it is specifically designed to be a more challenging non-linear dataset (Prill et al., 2010).

The weighted adjacency matrices of the 5 gene networks and 1 simulated dataset are shown in Figures 4.3, 4.4, 4.15, 4.9, 4.10, and 4.18. By looking at these adjacency matrices A_{10X10}, we can interpret that if there is an edge or not from node N_i to N_j. If value is zero that means there is no edge connecting N_i and N_j. Convolution operators compute time series embeddings by infusing the global structural properties of each graph encoded in the adjacency matrices into the calculations. The graphical representations 4.5, 4.7, 4.16, 4.11,4.13, and 4.19 are force-directed drawing which applies the ForceAtlas2 algorithm to learn the connections between nodes in order to create a structural map of the network. The generated visual graphs can be interpreted as structural similarity densities. The closer the nodes are, the weaker the attraction force is, subject to the edge weights factor. The repulsion force decreases when distances increase (Jacomy et al., 2014). Figures 4.5, 4.7, 4.16, 4.11,4.13, and 4.19 display network weight initial values modeled as the distance correlation plots derived from the time series data. These figures are distance correlation matrix C_{10X10} where each row and column corresponds to a gene. Higher values closer to 1 represents

dependence between different genes whereas lower values closer to 0 denote genes are independent. As with the graphical data, distance correlation values capture similar dynamics between variables across time points.

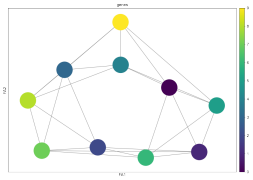


Figure 4.4 Lorenz graph.

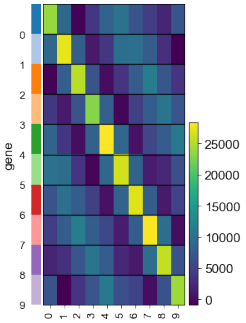


Figure 4.2 Weighted Lorenz adjacency matrix.

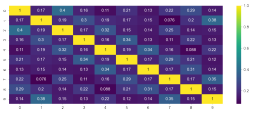


Figure 4.5 Lorenz Distance Correlation.

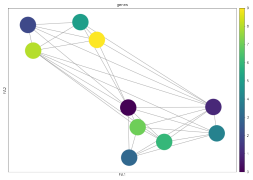


Figure 4.6 Yeast1 graph.

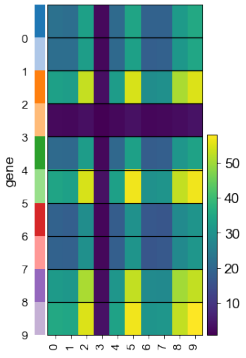


Figure 4.3 Weighted Yeast1 adjacency matrix.

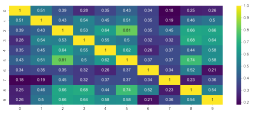


Figure 4.7 Yeast1 Distance Correlation.

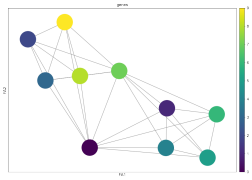


Figure 4.10 Yeast2 graph.

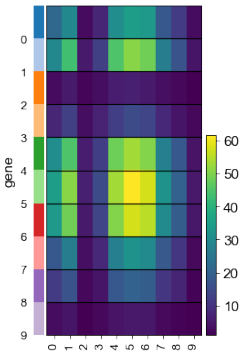


Figure 4.8 Weighted Yeast2 adjacency matrix.

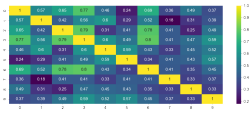


Figure 4.11 Yeast2 Distance Correlation.

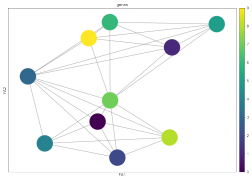


Figure 4.12 Yeast3 graph.

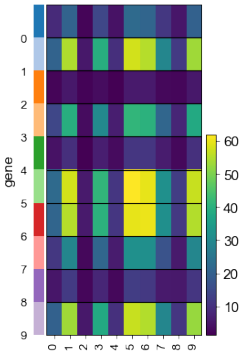


Figure 4.9 Weighted Yeast3 adjacency matrix.

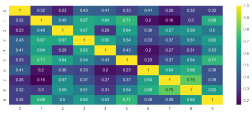


Figure 4.13 Yeast3 Distance Correlation.

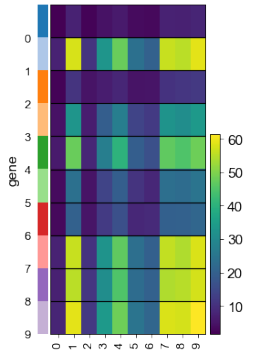


Figure 4.14 Weighted Ecoli1 adjacency matrix.

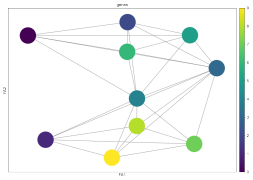


Figure 4.15 Ecoli1 graph.

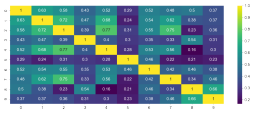


Figure 4.16 Ecoli1 Distance Correlation.

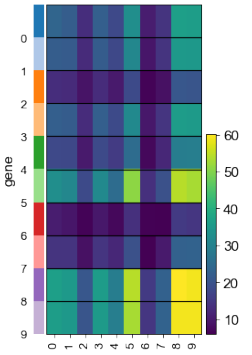


Figure 4.17 Weighted Ecoli2 adjacency matrix.

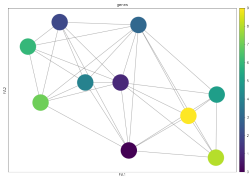


Figure 4.18 Ecoli2 graph.

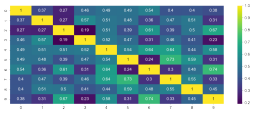


Figure 4.19 Ecoli2 Distance Correlation.

We implemented GGC-cLSTM, GGC-cRNN, and GGC-LOO with 10 hidden units to detect complex non-linear dependencies in Dream3 10-gene datasets and Lorenz-96 datasets and compared our GGC approach with previously published results of models GC-cLSTM (Granger Causal-cLSTM) and GC-cRNN (Granger Causal-cRNN) (Tank, Covert, et al., 2018). Accuracies for all models for all datasets are shown in Table 4.1. In terms of accuracy, our models GGC-cLSTM outperforms GC-cLSTM and GC-cRNN across all six datasets by a wide margin. GGC-cRNN follows a similar architecture with GGC-cLSTM, replacing the LSTM layer with a Recurrent Neural network (RNN) layer. This model outperformed all other models on the Yeast2 data (Table 4.1). These results indicate that including graphical interactions and non-linear dependencies as input to GGC-cLSTM networks boost the performance in recovering the interpretable non-linear interactions. The reason behind GGC-LOO not performing well might be the presence of a masked variable in inter-dependencies between remaining variables which are not masked.

In Figures 4.20 and 4.21, we also compute the AUROC and AUPR for all models across all five Dream 10-gene datasets since these performance metrics are commonly used for Dream datasets. As expected, the AUROC plot and the AUPR plot indicate that the GGC-cLSTM

and GGC-cRNN perform better than the other models. Except for the yeast-2 dataset where GGC-cRNN slightly underperforms GC-cRNN.

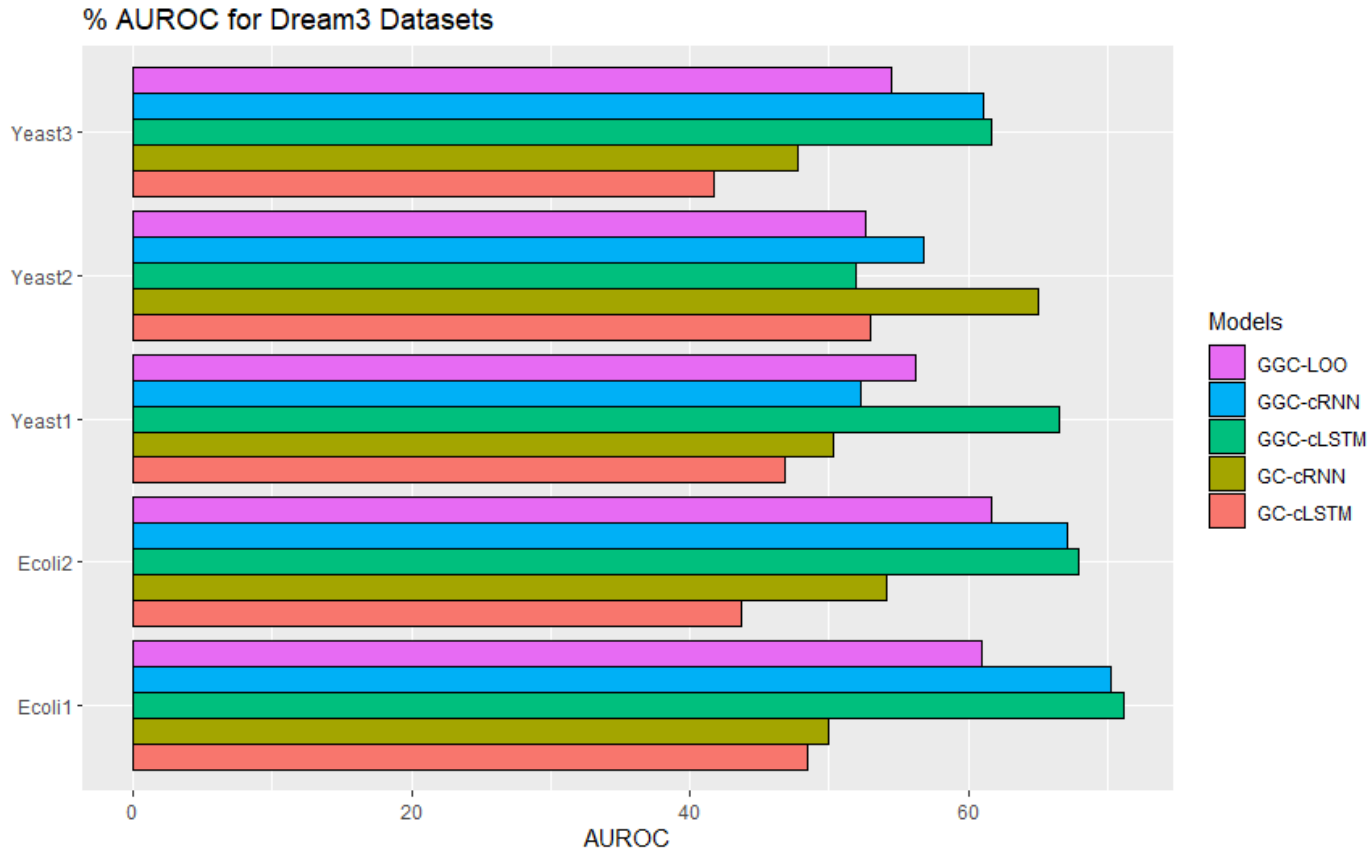


Figure 4.20 AUROC in percentage for Dream3 10-gene Datasets.

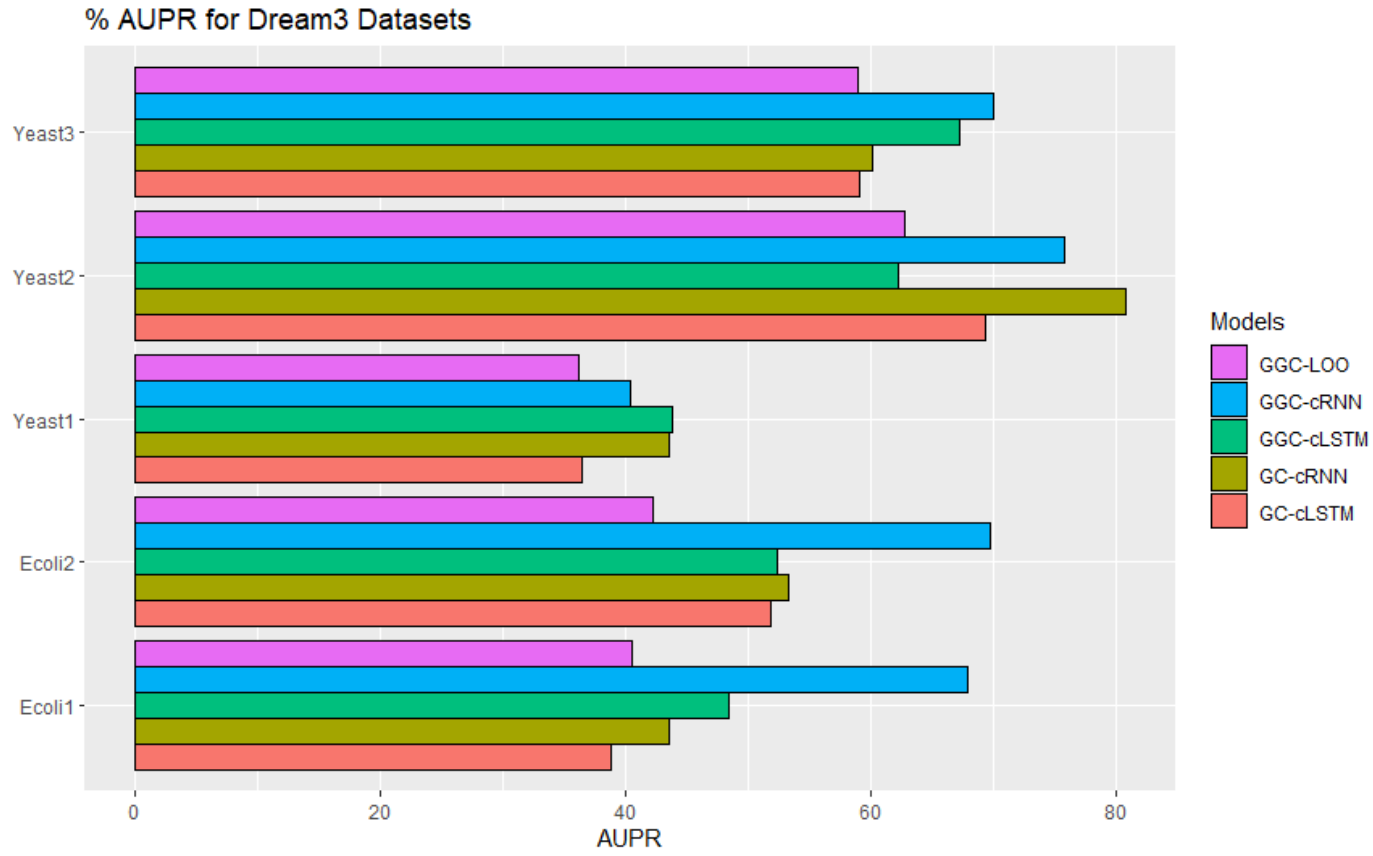


Figure 4.21 AUPR in percentage for Dream3 10-gene Datasets.

To check the performance of GGC-cLSTM model on a higher dimensional time series dataset with a larger number of hidden units, we implemented 100 hidden units for the GGC-cLSTM for five 100-gene Dream Dataset. The results in Table 4.2 reveals that a higher number of hidden units improves the performance of GCC-cLSTM across all five 100-gene datasets as compared to 10-gene datasets. We also observed the computational cost of models increases with the number of variables with the 10-variable models being the most efficient.

Model	Lorenz	Ecoli1	Ecoli2	Yeast1	Yeast2	Yeast3
GGC-cLSTM	**.99**	**.70**	**.64**	**.71**	.56	**.60**
GC-cLSTM	.97	.56	.42	.57	.55	.46
GGC-cRNN	.90	.68	.60	.60	**.59**	.57
GC-cRNN	.95	.56	.56	.58	.55	.51
GGC-LOO	.60	.69	.54	.52	.54	.58

Table 4.1 Graph Ganger Cusality accuracy on the 5 Dream3 10-gene datasets and 1 simulated dataset. Comparable results from GC-cLSTM and GC-cRNN (Tank, Covert, et al., 2018) are listed.

Model	Ecoli1	Ecoli2	Yeast1	Yeast2	Yeast3
GGC-cLSTM	.93	.93	.94	.89	.79

Table 4.2 Graph Ganger Causality cLSTM accuracy on the 5 Dream3 100-gene datasets.

4.6 Conclusions

Graph Granger Causality framework discussed here combines LSTM and graph convolutions with the objective of learning latent causal relationships in gene regulatory networks. Causal dynamics in networks are oftentimes rooted in interaction patterns that can be captured through various nonlinear functions such as distance correlation metrics. GGC captures these intra-gene nonlinearities through initialization and penalties strategies. First, the input weights of the LSTM layers are given more informative initialization values by sharing them with the distance correlation function. And secondly, the optimization step penalizes

these weights if they steer too far away from these values. We found that using informative network weights, the model tends to better capture the true causality structure behind the time-series data. The deep learning architecture proposed here for studying causality in dynamic networks was able to reconstruct the Lorenz-96 simulated dataset to 99% accuracy and has achieved good results on more challenging Dream3 gene regulatory networks time-series datasets. Future work can include the use of complex models with more hidden layers to detect the large-range dependencies in the high-dimensional setting to further enhance the performance and interpretation of the Granger causality framework. To resolve computation issues in high-dimensional and large dataset settings, parallel and distributed cloud computing can be helpful.

Chapter Five

Conclusions

Deep learning models have been proven to be very powerful in a wide range of applications, but there are problems that all model architects encounter; for example, 1) biased decisions toward the majority class due to an imbalanced dataset, 2) erroneous classification/prediction decision due to model dependency on biases such as device configuration, age, gender, and color biases, etc., 3) decision based on association and not understanding causal relationships and thus not able to understand the mechanisms underlying the dynamics. Throughout this dissertation, we studied the bias mitigation methods and Granger Causality using the deep learning models to tackle these diverse sets of problems. In this research process, we developed a number of novel methods to learn bias invariant features and infer GC interactions while revisiting and comparing the traditional approaches. We proposed simple yet effective methods and also explored the use of these methods beyond just one single application, i.e., we tested these methods across different application domains. Thus, the proposed method is flexible, generic, and robust and should be applicable to a wide range of applications.

5.1 Learning Class Bias and Scanner Invariant Features

Neuroimaging datasets are often imbalanced, and sometimes different datasets are combined between scanners and acquisition protocols to improve the performance of deep learning models when the limited dataset is available. As a result, models trained on these types of datasets introduce a prediction bias for the majority class and scanner type, which ultimately adversely impacts the model performance. In chapter 2, we introduced a novel decorrelation approach, which reduces dependencies between the features learned by deep learning models and biases. The approach was formulated by adding a simple regularisation term based on the distance correlation function. This allows us to mitigate scanner dependencies and class bias which helped the model to generalize to multi-scanner and multi-center datasets. We further proposed four different deep learning architectures for single scanner imbalanced and multi-scanner datasets. Our approach performed better compared to previous approaches and baseline models while requiring fewer hyperparameters to optimize. We also discovered that using multiple scanners and a larger dataset resulted in better performance when compared to a single scanner imbalanced dataset.

We consider two bias problems, class bias and scanner bias, present in the PD rs-fMRI dataset. Given that deep learning models are data-driven, and their performance is impacted by the quality of data, we observe that the performance of the baseline deep learning model gets compromised for the single-scanner imbalanced dataset and multi-scanner dataset. We thus propose a fusion model by using sampling and weighted strategy along with decorrelation function to address the imbalanced issue in the single scanner dataset and a fusion of scanner feature extractor model and PD classifier model to address the scanner bias issue in the multi-scanner dataset. This work has shown how a simple regularization function based on decorrelation can achieve state-of-the-art performance while mitigating biases.

5.2 Generic Framework for Learning Bias Invariant Features

In Chapter 3, we further proposed a generic decorrelation-based framework to expand the scope of decorrelated deep learning model robustness to more settings. Our approach based on the decorrelation function was motivated by the simple and effective idea of reducing the association between features learned by the ANN, CNN, or deep learning models and bias. In order to capture linear as well as non-linear associations, the distance correlation function is used in the decorrelation function. We additionally explored different deep learning architectures such as CNN, ConvGRU-CNN, and ANN to mitigate different types of biases such as age, gender, scanner, and color. The optimization process used in the proposed approach is simpler with fewer hyperparameters while obtaining powerful results as compared to traditional and adversarial network-based approaches. By virtue of the generic framework, the proposed bias mitigation approach is flexible, scalable, and generic and is also ready for applications in a wide range of scenarios and fields where bias removal is crucial without significantly impacting the performance of deep learning models. Thus, decorrelated based deep learning models such as DcCNN, DcANN, and ConvGRU-DcCNN are effective and promising new models which address problems of bias and fairness of decision algorithms in various applications. We also studied how the hyperparameter value controls the tradeoff between accuracy and bias reduction. However, hyperparameter value relies on the deep learning architecture and the complexity of the task.

5.3 Learning Granger Causal Relationship

The majority of causality models rely on linear-based approaches such as regression techniques to learn Granger causality, whereas non-linear deep learning models learn from associations present in the data. In order to take benefit from both these models, deep learning

and the Granger causality method are combined together to develop the fusion model. In order to extract Granger Causal structure, sparsity-inducing group-lasso penalties are used to force the weights of deep learning models such as RNN and LSTM to be zero. In chapter 4, we extend this fusion model by further combining with GNN and using distance correlation metrics with the objective of learning latent causal relationships in gene regulatory networks by capturing intra-gene nonlinearities. We established a distance correlation function to capture nonlinearities and developed an effective method by integrating distance correlation which assisted our proposed approach to provide informative weight initialization and to better capture true non-linear predictive causality structure. Graph convolution-based GNN further helped our proposed approach to capture network structure, connectivity, and interaction patterns.

5.4 Future Research Directions

The ideas presented in this dissertation not only lead to a more robust, generic, flexible, and simple yet powerful bias mitigation technique that can be applicable to a wide range of studies in different domains but also indicates the importance of introducing causality in deep learning models. Moreover, it also includes the first-time application of using CNN in classifying PD patients from normal subjects using rs-fMRI data. We believe that these ideas lay the groundwork for building more generalizable and robust deep learning models using bias mitigating techniques that allow us to use all available datasets and also understand the mechanism underlying the systems using causality. The future work includes;

- Implementation of advanced visualization techniques for deep learning in the detection of PD will not only help in characterizing fMRI biomarkers for PD but also in understanding the underlying working mechanism of the deep learning model. This will helpful for diagnosing Parkinson's in order to improve patient treatment strategies. This research also provides a foundation for future research into predicting stages of

Parkinson's disease progression.

- It would be an interesting line of future work to expand the proposed method by applying it to pre-trained models and to different types of data variations. The notion of using a feature extractor model to encode a bias variable in our proposed approach could be leveraged to tackle domain adaption problems in deep learning.

- Using parallel and distributed cloud computing for the deep complex models in high-dimensional and large dataset settings will be helpful to further enhance the performance and interpretation of the Granger causality framework. Since Granger causality does not always imply true causality, cause and effect study based on a structural causal model (SCM) or based on different types of causal inference concepts should be introduced in deep learning algorithms to retain crucial information and not just predictive causality information. We believe it is an interesting and major new area of research.

APPENDICES

Appendix A

Distance Correlation

We use the definition of distance correlation introduced by Székely, Rizzo, and Bakirov, 2007. B and F are random vectors representing bias and features. n denotes the sample size, $\|.\|$ indicates L2 norm and $k, l = 1, 2, 3, ..., n$.The distance dependence is calculated as follows in Equation A.1 and Equation A.2:

$$\begin{aligned} b_{kl} &= \|x_k - x_l\| \\ \overline{b}_{k.} &= \frac{1}{n}\sum_{l=1}^{n} b_{kl} \\ \overline{b}_{.l} &= \frac{1}{n}\sum_{k=1}^{n} b_{kl} \\ \overline{b}_{..} &= \frac{1}{n^2}\sum_{k,l=1}^{n} b_{kl} \\ B_{kl} &= b_{kl} - \overline{b}_{k.} - \overline{b}_{.l} + \overline{b}_{..} \end{aligned} \tag{A.1}$$

$$\begin{aligned} f_{kl} &= \|y_k - y_l\| \\ \overline{f}_{k.} &= \frac{1}{n}\sum_{l=1}^{n} b_{kl} \\ \overline{f}_{.l} &= \frac{1}{n}\sum_{k=1}^{n} b_{kl} \\ \overline{f}_{..} &= \frac{1}{n^2}\sum_{k,l=1}^{n} b_{kl} \\ F_{kl} &= f_{kl} - \overline{f}_{k.} - \overline{f}_{.l} + \overline{f}_{..} \end{aligned} \tag{A.2}$$

The empirical estimate for distance covariance is calculated using distance dependence statistics. It is given as:

$$V_n^2(B,F) = \frac{1}{n^2}\sum_{k,l=1}^{n} B_{kl}F_{kl} \tag{A.3}$$

Similarly, the empirical estimate for distance variance $V^2(B,B)$ and $V^2(F,F)$ are calculated. The distance correlation $DC^2(B,F)$ between random variables B and F is defined as:

$$\mathcal{DC}^2(B,F) = \begin{cases} \frac{\mathcal{V}^2(B,F)}{\sqrt{\mathcal{V}^2(B,B)\mathcal{V}^2(F,F)}} & \text{if } \mathcal{V}^2(B,B)\mathcal{V}^2(F,F) > 0 \\ 0 & \text{else } \mathcal{V}^2(B,B)\mathcal{V}^2(F,F) = 0 \end{cases} \tag{A.4}$$

$DC^2(B,F)\varepsilon[0,1]$ demonstrates statistical dependency between variables B and F. $DC(B,F) = 0$ only when the variables B and F are independent. The distance covariance is normalized by using the distance variances.

Appendix B

Color Bias in MNIST Dataset

In color biased MNIST dataset, ten colors are selected for each digit, and color bias is intentionally induced in the training dataset (B. Kim et al., 2019). The testing dataset is independent of color bias. The colors and their values for each digit in the training dataset are listed in Table B.1.

In reversed color biased MNIST dataset, we randomly divide the dataset into two environments, unlike the three environments used in Arjovsky et al., 2019. Main four steps used to create reversed color biased dataset are as follows:

1. Assign output label based on a digit. If a digit is between 0-4, then assign the output label as zero otherwise, assign the label as one.

2. All labels are flipped with a 25% probability

3. Red and green colors are added to each grayscale image according to flipped label.

4. Color of images in training sets is flipped with 30% probability, whereas the color of images in the testing is flipped with 90% probability which results in a reversed direction in the testing dataset.

Table B.1 Color Bias information of MNIST Dataset.

Digit	**Color Name**	**Color Value**
0	Crimson	(220, 20, 60)
1	Teal	(0,128,128)
2	Lemon	(253,233, 16)
3	Bondi Blue	(0,149,182)
4	Carrot orange	(237,145, 33)
5	Strong Violet	(145, 30,188)
6	Cyan	(70,240,240)
7	Pink	(250,197,187)
8	Lime	(210,245, 60)
9	Maroon	(128, 0, 0)

References

Adeli, Ehsan et al. (2021). "Representation learning with statistical independence to mitigate bias". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2513–2523.

El-Agnaf, Omar MA et al. (2006). "Detection of oligomeric forms of α-synuclein protein in human plasma as a potential biomarker for Parkinson's disease". In: *The FASEB journal* 20.3, pp. 419–425.

Ahmed, Mohamed N and Aly A Farag (1997). "Two-stage neural network for volume segmentation of medical images". In: *Proceedings of International Conference on Neural Networks (ICNN'97)*. Vol. 3. IEEE, pp. 1373–1378.

Alvi, Mohsan, Andrew Zisserman, and Christoffer Nellåker (2018). "Turning a blind eye: Explicit removal of biases and variation from deep neural network embeddings". In: *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pp. 0–0.

Arjovsky, Martin et al. (2019). "Invariant risk minimization". In: *arXiv preprint arXiv:1907.02893*.

Atkinson-Clement, Cyril et al. (2017). "Diffusion tensor imaging in Parkinson's disease: Review and meta-analysis". In: *NeuroImage: Clinical* 16, pp. 98–110.

Ballas, Nicolas et al. (2015). "Delving deeper into convolutional networks for learning video representations". In: *arXiv preprint arXiv:1511.06432*.

Basu, Sumanta, Ali Shojaie, and George Michailidis (2015). "Network granger causality with inherent grouping structure". In: *The Journal of Machine Learning Research* 16.1, pp. 417–453.

Batista, Gustavo EAPA, Ronaldo C Prati, and Maria Carolina Monard (2004). "A study of the behavior of several methods for balancing machine learning training data". In: *ACM SIGKDD explorations newsletter* 6.1, pp. 20–29.

Baumgartner, Christian F et al. (2017). "An exploration of 2D and 3D deep learning techniques for cardiac MR image segmentation". In: *International Workshop on Statistical Atlases and Computational Models of the Heart*. Springer, pp. 111–119.

Beilina, Alexandra and Mark R Cookson (2016). "Genes associated with Parkinson's disease: regulation of autophagy and beyond". In: *Journal of neurochemistry* 139, pp. 91–107.

Bejnordi, Babak Ehteshami et al. (2017). "Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer". In: *Jama* 318.22, pp. 2199–2210.

Bellamy, Rachel KE et al. (2018). "AI Fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias". In: *arXiv preprint arXiv:1810.01943*.

Bengs, Marcel, Nils Gessert, and Alexander Schlaefer (2020). "4d spatio-temporal deep learning with 4d fmri data for autism spectrum disorder classification". In: *arXiv preprint arXiv:2004.10165*.

Bhattacharjee, Atanu (2014). "Distance correlation coefficient: An application with bayesian approach in clinical data analysis". In: *Journal of Modern Applied Statistical Methods* 13.1, p. 23.

Braak, Heiko et al. (2003). "Staging of brain pathology related to sporadic Parkinson's disease". In: *Neurobiology of aging* 24.2, pp. 197–211.

Bronstein, Michael M et al. (2017). "Geometric deep learning: going beyond euclidean data". In: *IEEE Signal Processing Magazine* 34.4, pp. 18–42.

Bruna, Joan et al. (2013). "Spectral networks and locally connected networks on graphs". In: *arXiv preprint arXiv:1312.6203*.

Buolamwini, Joy and Timnit Gebru (2018). "Gender shades: Intersectional accuracy disparities in commercial gender classification". In: *Conference on fairness, accountability and transparency*. PMLR, pp. 77–91.

Calmon, Flavio et al. (2017). "Optimized pre-processing for discrimination prevention". In: *Advances in neural information processing systems* 30.

Castro, Juan Camilo et al. (2019). "Gene regulatory networks on transfer entropy (GRNTE): a novel approach to reconstruct gene regulatory interactions applied to a case study for the plant pathogen Phytophthora infestans". In: *Theoretical Biology and Medical Modelling* 16.1, pp. 1–15.

Chawla, Nitesh V et al. (2002). "SMOTE: synthetic minority over-sampling technique". In: *Journal of artificial intelligence research* 16, pp. 321–357.

Chen, Jianfei, Jun Zhu, and Le Song (2017). "Stochastic training of graph convolutional networks with variance reduction". In: *arXiv preprint arXiv:1710.10568*.

Chen, Jie, Tengfei Ma, and Cao Xiao (2018). "Fastgcn: fast learning with graph convolutional networks via importance sampling". In: *arXiv preprint arXiv:1801.10247*.

Chen, Xing and Li Huang (2017). "LRSSLMDA: Laplacian regularized sparse subspace learning for MiRNA-disease association prediction". In: *PLoS computational biology* 13.12, e1005912.

Chetlur, Sharan et al. (2014). "cudnn: Efficient primitives for deep learning". In: *arXiv preprint arXiv:1410.0759*.

Cho, Kyunghyun et al. (2014). "Learning phrase representations using RNN encoder-decoder for statistical machine translation". In: *arXiv preprint arXiv:1406.1078*.

Choi, Hongyoon et al. (2017). "Refining diagnosis of Parkinson's disease with deep learning-based interpretation of dopamine transporter imaging". In: *NeuroImage: Clinical* 16, pp. 586–594.

Chung, Junyoung et al. (2014). "Empirical evaluation of gated recurrent neural networks on sequence modeling". In: *arXiv preprint arXiv:1412.3555*.

Cochrane, Claire J and Klaus P Ebmeier (2013). "Diffusion tensor imaging in parkinsonian syndromes: a systematic review and meta-analysis". In: *Neurology* 80.9, pp. 857–864.

De Wilde, Bram, Richard PG ten Broek, and Henkjan Huisman (2021). "Cine-MRI detection of abdominal adhesions with spatio-temporal deep learning". In: *arXiv preprint arXiv:2106.08094*.

Deep learning ami - Developer Guide (n.d.). URL: https://docs.aws.amazon.com/dlami/latest/devguide/dlami-dg.pdf.

Defferrard, Michaël, Xavier Bresson, and Pierre Vandergheynst (2016). "Convolutional neural networks on graphs with fast localized spectral filtering". In: *Advances in neural information processing systems* 29.

Dinsdale, Nicola K, Mark Jenkinson, and Ana IL Namburete (2021). "Deep learning-based unlearning of dataset bias for MRI harmonisation and confound removal". In: *NeuroImage* 228, p. 117689.

Dong, Jie et al. (2017). "Joint data-driven fault diagnosis integrating causality graph with statistical process monitoring for complex industrial processes". In: *IEEE Access* 5, pp. 25217–25225.

Dream Challenges (n.d.). https://dreamchallenges.org/dream-3-in-silico-network-challenge/. Accessed: 2021-08-30.

Dua, Dheeru and Casey Graff (2017). *UCI Machine Learning Repository*. URL: http://archive.ics.uci.edu/ml.

Dwork, Cynthia et al. (2018). "Decoupled classifiers for group-fair and efficient machine learning". In: *Conference on fairness, accountability and transparency*. PMLR, pp. 119–133.

Esmaeilzadeh, Soheil, Yao Yang, and Ehsan Adeli (2018). "End-to-End Parkinson Disease Diagnosis using Brain MR-Images by 3D-CNN". In: *arXiv preprint arXiv:1806.05233*.

Feldman, Michael et al. (2015). "Certifying and removing disparate impact". In: *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 259–268.

Feldman, Tal and Ashley Peake (2021). "End-To-End Bias Mitigation: Removing Gender Bias in Deep Learning". In: *arXiv preprint arXiv:2104.02532*.

Finkle, Justin D, Jia J Wu, and Neda Bagheri (2018). "Windowed Granger causal inference strategy improves discovery of gene regulatory networks". In: *Proceedings of the National Academy of Sciences* 115.9, pp. 2252–2257.

Frasconi, Paolo, Marco Gori, and Alessandro Sperduti (1998). "A general framework for adaptive processing of data structures". In: *IEEE transactions on Neural Networks* 9.5, pp. 768–786.

Friedman, Lee, Gary H Glover, Fbirn Consortium, et al. (2006). "Reducing interscanner variability of activation in a multicenter fMRI study: controlling for signal-to-fluctuation-noise-ratio (SFNR) differences". In: *Neuroimage* 33.2, pp. 471–481.

Friedman, Lee, Gary H Glover, Diana Krenz, et al. (2006). "Reducing inter-scanner variability of activation in a multicenter fMRI study: role of smoothness equalization". In: *Neuroimage* 32.4, pp. 1656–1668.

Fry, Anna et al. (2017). "Comparison of sociodemographic and health-related characteristics of UK Biobank participants with those of the general population". In: *American journal of epidemiology* 186.9, pp. 1026–1034.

Fukushima, Kunihiko, Sei Miyake, and Takayuki Ito (1983). "Neocognitron: A neural network model for a mechanism of visual pattern recognition". In: *IEEE transactions on systems, man, and cybernetics* 5, pp. 826–834.

Ganin, Yaroslav et al. (2016). "Domain-adversarial training of neural networks". In: *The journal of machine learning research* 17.1, pp. 2096–2030.

Gers, Felix A and Jürgen Schmidhuber (2000). "Recurrent nets that time and count". In: *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium*. Vol. 3. IEEE, pp. 189–194.

Gers, Felix A, Jürgen Schmidhuber, and Fred Cummins (2000). "Learning to forget: Continual prediction with LSTM". In: *Neural computation* 12.10, pp. 2451–2471.

Gessert, Nils et al. (2018). "Needle tip force estimation using an oct fiber and a fused convgru-cnn architecture". In: *International conference on medical image computing and computer-assisted intervention*. Springer, pp. 222–229.

Gil, David and Devadoss Johnson Manuel (2009). "Diagnosing Parkinson by using artificial neural networks and support vector machines". In: *Global Journal of Computer Science and Technology* 9.4.

Gilmer, Justin et al. (2017). "Neural message passing for quantum chemistry". In: *International conference on machine learning*. PMLR, pp. 1263–1272.

Goodfellow, Ian et al. (2014). "Generative adversarial nets". In: *Advances in neural information processing systems* 27.

Gordienko, Yu et al. (2018). "Deep learning with lung segmentation and bone shadow exclusion techniques for chest X-ray analysis of lung cancer". In: *International conference on computer science, engineering and education applications*. Springer, pp. 638–647.

Gori, Marco, Gabriele Monfardini, and Franco Scarselli (2005). "A new model for learning in graph domains". In: *Proceedings. 2005 IEEE international joint conference on neural networks*. Vol. 2. 2005, pp. 729–734.

Granger, C (1969). "investigating causal relations by econometric models and crossspectral methods". In: *Econometrica* 37, pp. 424–438.

Graves, Alex and Jürgen Schmidhuber (2005). "Framewise phoneme classification with bidirectional LSTM and other neural network architectures". In: *Neural networks* 18.5-6, pp. 602–610.

Gulshan, Varun et al. (2016). "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs". In: *Jama* 316.22, pp. 2402–2410.

Guo, Xueqi, Sule Tinaz, and Nicha C Dvornek (2022). "Early Disease Stage Characterization in Parkinson's Disease from Resting-state fMRI Data Using a Long Short-term Memory Network". In: *arXiv preprint arXiv:2202.12715*.

Hagan, Martin T, Howard B Demuth, and Mark Beale (1997). *Neural network design*. PWS Publishing Co.

Hajian, Sara and Josep Domingo-Ferrer (2012). "A methodology for direct and indirect discrimination prevention in data mining". In: *IEEE transactions on knowledge and data engineering* 25.7, pp. 1445–1459.

Hamilton, Will, Zhitao Ying, and Jure Leskovec (2017). "Inductive representation learning on large graphs". In: *Advances in neural information processing systems* 30.

Hamilton, William L, Rex Ying, and Jure Leskovec (2017). "Representation learning on graphs: Methods and applications". In: *arXiv preprint arXiv:1709.05584*.

Hardt, Moritz, Eric Price, and Nati Srebro (2016). "Equality of opportunity in supervised learning". In: *Advances in neural information processing systems* 29.

He, Kaiming et al. (2016). "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

Henaff, Mikael, Joan Bruna, and Yann LeCun (2015). "Deep convolutional networks on graph-structured data". In: *arXiv preprint arXiv:1506.05163*.

Hochreiter, Sepp and Jürgen Schmidhuber (1997). "Long short-term memory". In: *Neural computation* 9.8, pp. 1735–1780.

Hou, Jie et al. (2022). "Distance correlation application to gene co-expression network analysis". In: *BMC bioinformatics* 23.1, pp. 1–24.

Howard, Ayanna and Jason Borenstein (2018). "The ugly truth about ourselves and our robot creations: the problem of bias and social inequity". In: *Science and engineering ethics* 24.5, pp. 1521–1536.

Ibtehaz, Nabil and M Sohel Rahman (2020). "MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation". In: *Neural networks* 121, pp. 74–87.

Jacomy, Mathieu et al. (2014). "ForceAtlas2, a continuous graph layout algorithm for handy network visualization designed for the Gephi software". In: *PloS one* 9.6, e98679.

Jenkinson, Mark et al. (2002). "Improved optimization for the robust and accurate linear registration and motion correction of brain images". In: *Neuroimage* 17.2, pp. 825–841.

Ji, Qiang et al. (2018). "Network causality structures among Bitcoin and other financial assets: A directed acyclic graph approach". In: *The Quarterly Review of Economics and Finance* 70, pp. 203–213.

Jiji, Wiselin, A Rajesh, and M Maha Lakshmi (2022). "Diagnosis of Parkinson's Disease Using EEG and fMRI". In.

Kamiran, Faisal and Toon Calders (2012). "Data preprocessing techniques for classification without discrimination". In: *Knowledge and information systems* 33.1, pp. 1–33.

Kamishima, Toshihiro et al. (2012). "Fairness-aware classifier with prejudice remover regularizer". In: *Joint European conference on machine learning and knowledge discovery in databases*. Springer, pp. 35–50.

Kamulegeya, Louis Henry et al. (2019). "Using artificial intelligence on dermatology conditions in Uganda: A case for diversity in training data sets for machine learning". In: *BioRxiv*, p. 826057.

Karimi, Alireza and Mark R Paul (2010). "Extensive chaos in the Lorenz-96 model". In: *Chaos: An interdisciplinary journal of nonlinear science* 20.4, p. 043105.

Kärkkäinen, Kimmo and Jungseock Joo (2019). "Fairface: Face attribute dataset for balanced race, gender, and age". In: *arXiv preprint arXiv:1908.04913*.

Kasieczka, Gregor and David Shih (2020). "DisCo Fever: Robust Networks Through Distance Correlation". In: *arXiv preprint arXiv:2001.05310*.

Kim, Byungju et al. (2019). "Learning not to learn: Training deep neural networks with biased data". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9012–9020.

Kim, Michael P, Amirata Ghorbani, and James Zou (2019). "Multiaccuracy: Black-box post-processing for fairness in classification". In: *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 247–254.

Kim, Sangjune et al. (2019). "Transneuronal Propagation of Pathologic α-Synuclein from the Gut to the Brain Models Parkinson's Disease". In: *Neuron*, pp. 1–15. ISSN: 08966273. DOI: 10.1016/j.neuron.2019.05.035. URL: https://linkinghub.elsevier.com/retrieve/pii/S089662731930488X.

Kipf, Thomas N and Max Welling (2016). "Semi-supervised classification with graph convolutional networks". In: *arXiv preprint arXiv:1609.02907*.

Kiranyaz, Serkan et al. (2015). "Convolutional neural networks for patient-specific ECG classification". In: *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, pp. 2608–2611.

Kong, Jing, Sijian Wang, and Grace Wahba (2015). "Using distance covariance for improved variable selection with application to learning genetic risk models". In: *Statistics in medicine* 34.10, pp. 1708–1720.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton (2012). "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems* 25.

Kusner, Matt J et al. (2017). "Counterfactual fairness". In: *Advances in neural information processing systems* 30.

Lakhani, Paras and Baskaran Sundaram (2017). "Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks". In: *Radiology* 284.2, pp. 574–582.

LeCun, Yann, Bernhard Boser, et al. (1989). "Handwritten digit recognition with a back-propagation network". In: *Advances in neural information processing systems* 2.

LeCun, Yann, Léon Bottou, et al. (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11, pp. 2278–2324.

LeCun, Yann and Corinna Cortes (2010). "MNIST handwritten digit database". In: URL: http://yann.lecun.com/exdb/mnist/.

Lee Rodgers, Joseph and W Alan Nicewander (1988). "Thirteen ways to look at the correlation coefficient". In: *The American Statistician* 42.1, pp. 59–66.

Leevy, Joffrey L et al. (2018). "A survey on addressing high-class imbalance in big data". In: *Journal of Big Data* 5.1, pp. 1–30.

Li, Kai et al. (2018). "Resting State fMRI: A Valuable Tool for Studying Cognitive Dysfunction in PD". In: *Parkinson's Disease* 2018.

Li, Qingjiang et al. (2022). "Prediction Model of Ischemic Stroke Recurrence Using PSO-LSTM in Mobile Medical Monitoring System". In: *Computational Intelligence and Neuroscience* 2022.

Li, Runze, Wei Zhong, and Liping Zhu (2012). "Feature screening via distance correlation learning". In: *Journal of the American Statistical Association* 107.499, pp. 1129–1139.

Li, Xiaoxiao, Ziteng Cui, et al. (2021). "Estimating and improving fairness with adversarial learning". In: *arXiv preprint arXiv:2103.04243*.

Li, Xiaoxiao, Yufeng Gu, et al. (2020). "Multi-site fMRI analysis using privacy-preserving federated learning and domain adaptation: ABIDE results". In: *Medical Image Analysis* 65, p. 101765.

Li, Yi and Nuno Vasconcelos (2019). "Repair: Removing representation bias by dataset resampling". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9572–9581.

Liu, Furui and Laiwan Chan (2016). "Causal inference on discrete data via estimating distance correlations". In: *Neural computation* 28.5, pp. 801–814.

Lou, Ange, Shuyue Guan, and Murray Loew (2021). "DC-UNet: rethinking the U-Net architecture with dual channel efficient CNN for medical image segmentation". In: *Medical Imaging 2021: Image Processing*. Vol. 11596. SPIE, pp. 758–768.

Lozano, Aurelie C et al. (2009). "Grouped graphical Granger modeling methods for temporal causal modeling". In: *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 577–586.

Lu, Jonathan et al. (2021). "Causal network inference from gene transcriptional time-series response to glucocorticoids". In: *PLoS computational biology* 17.1, e1008223.

Ma, Tengfei et al. (2018). "Drug similarity integration through attentive multi-view graph auto-encoders". In: *arXiv preprint arXiv:1804.10850*.

Ma, Tianqi et al. (2020). "ConvGRU in Fine-grained Pitching Action Recognition for Action Outcome Prediction". In: *arXiv preprint arXiv:2008.07819*.

Mandis, Isabella S (n.d.). "Reducing Racial and Gender Bias in Machine Learning and Natural Language Processing Tasks Using a GAN Approach". In: ().

Martín Abadi et al. (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. URL: https://www.tensorflow.org/.

Mehrabi, Ninareh et al. (2021). "A survey on bias and fairness in machine learning". In: *ACM Computing Surveys (CSUR)* 54.6, pp. 1–35.

Meyer, Maria Ines et al. (2021). "A Contrast Augmentation Approach to Improve Multi-Scanner Generalization in MRI". In: *Frontiers in neuroscience*, p. 1048.

Milletari, Fausto, Nassir Navab, and Seyed-Ahmad Ahmadi (2016). "V-net: Fully convolutional neural networks for volumetric medical image segmentation". In: *2016 fourth international conference on 3D vision (3DV)*. IEEE, pp. 565–571.

Mohana, J et al. (2022). "Application of internet of things on the healthcare field using convolutional neural network processing". In: *Journal of Healthcare Engineering* 2022.

Monti, Federico et al. (2017). "Geometric deep learning on graphs and manifolds using mixture model cnns". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5115–5124.

Nicholson, William B., Jacob Bien, and David S. Matteson (2014). "Hierarchical Vector Autoregression". In: *arXiv: Methodology*.

Okatan, Murat, Matthew A Wilson, and Emery N Brown (2005). "Analyzing functional connectivity using a network likelihood model of ensemble neural spiking activity". In: *Neural computation* 17.9, pp. 1927–1961.

Olteanu, Alexandra et al. (2019). "Social data: Biases, methodological pitfalls, and ethical boundaries". In: *Frontiers in Big Data* 2, p. 13.

Patil, Pranita (2013). *Flexible image recognition software toolbox (first)*. Oklahoma State University.

Patil, Pranita and Kevin Purcell (2022). "Decorrelation-Based Deep Learning for Bias Mitigation". In: *Future Internet* 14.4, p. 110.

Pei, Mengqi et al. (2017). "Small bowel motility assessment based on fully convolutional networks and long short-term memory". In: *Knowledge-Based Systems* 121, pp. 163–172.

Phu, Minh Tran and Thien Huu Nguyen (2021). "Graph convolutional networks for event causality identification with rich document-level structures". In: *Proceedings of the 2021*

Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 3480–3490.

Pleiss, Geoff et al. (2017). "On fairness and calibration". In: *Advances in neural information processing systems* 30.

Postuma, Ronald B et al. (2016). "The new definition and diagnostic criteria of Parkinson's disease". In: *The Lancet Neurology* 15.6, pp. 546–548.

Prill, Robert J et al. (2010). "Towards a rigorous assessment of systems biology models: the DREAM3 challenges". In: *PloS one* 5.2, e9202.

Psaradakis, Zacharias, Morten O Ravn, and Martin Sola (2005). "Markov switching causality and the money–output relationship". In: *Journal of Applied Econometrics* 20.5, pp. 665–683.

Quadrianto, Novi and Viktoriia Sharmanska (2017). "Recycling privileged learning and distribution matching for fairness". In: *Advances in Neural Information Processing Systems* 30.

Ren, Weijie, Baisong Li, and Min Han (2020). "A novel Granger causality method based on HSIC-Lasso for revealing nonlinear relationship between multivariate time series". In: *Physica A: Statistical Mechanics and its Applications* 541, p. 123245. ISSN: 0378-4371. DOI: https://doi.org/10.1016/j.physa.2019.123245. URL: https://www.sciencedirect.com/science/article/pii/S0378437119318217.

Rolinski, Michal et al. (2014). "Resting State Fmri Discerns Early Parkinson's From Controls". In: *J Neurol Neurosurg Psychiatry* 85.10, e4–e4.

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox (2015). "U-net: Convolutional networks for biomedical image segmentation". In: *International Conference on Medical image computing and computer-assisted intervention*. Springer, pp. 234–241.

Rubbert, Christian et al. (2019). "Machine-learning identifies Parkinson's disease patients based on resting-state between-network functional connectivity". In: *The british journal of radiology* 92.1101, p. 20180886.

Russakovsky, Olga et al. (2015). "Imagenet large scale visual recognition challenge". In: *International journal of computer vision* 115.3, pp. 211–252.

Sadeghi, Bashir, Runyi Yu, and Vishnu Boddeti (2019). "On the global optima of kernelized adversarial representation learning". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7971–7979.

Saeed, Hanan A et al. (2020). "Novel fault diagnosis scheme utilizing deep learning networks". In: *Progress in Nuclear Energy* 118, p. 103066.

Saeed, Usman et al. (2017). "Imaging biomarkers in Parkinson's disease and Parkinsonian syndromes: current and emerging concepts". In: *Translational neurodegeneration* 6.1, p. 8.

Sagheer, Alaa and Mostafa Kotb (2019). "Time series forecasting of petroleum production using deep LSTM recurrent networks". In: *Neurocomputing* 323, pp. 203–213.

Sak, Haşim, Andrew Senior, and Françoise Beaufays (2014). "Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition". In: *arXiv preprint arXiv:1402.1128*.

Saleema, JS et al. (2014). "Cancer prognosis prediction using balanced stratified sampling". In: *arXiv preprint arXiv:1403.2950*.

Salekshahrezaee, Zahra, Joffrey L Leevy, and Taghi M Khoshgoftaar (2021). "Feature extraction for class imbalance using a convolutional autoencoder and data sampling". In: *2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE, pp. 217–223.

Scarselli, Franco et al. (2008). "The graph neural network model". In: *IEEE transactions on neural networks* 20.1, pp. 61–80.

Shafiq-ul-Hassan, Muhammad et al. (2017). "Intrinsic dependencies of CT radiomic features on voxel size and number of gray levels". In: *Medical physics* 44.3, pp. 1050–1062.

Shankar, Shreya et al. (2017). "No classification without representation: Assessing geodiversity issues in open data sets for the developing world". In: *arXiv preprint arXiv:1711.08536*.

Sharma, Shubham et al. (2020). "Data augmentation for discrimination prevention and bias disambiguation". In: *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pp. 358–364.

Shi, Dafa et al. (2022). "Machine Learning for Detecting Parkinson's Disease by Resting-State Functional Magnetic Resonance Imaging: A Multicenter Radiomics Analysis". In: *Frontiers in aging neuroscience* 14, p. 806828.

Shi, Xingjian et al. (2015). "Convolutional LSTM network: A machine learning approach for precipitation nowcasting". In: *Advances in neural information processing systems* 28.

Shojaie, Ali and George Michailidis (2010). "Discovering graphical Granger causality using the truncating lasso penalty". In: *Bioinformatics* 26.18, pp. i517–i523.

Shuman, David I, Benjamin Ricaud, and Pierre Vandergheynst (2016). "Vertex-frequency analysis on graphs". In: *Applied and Computational Harmonic Analysis* 40.2, pp. 260–291.

Siggiridou, Elsa and Dimitris Kugiumtzis (2015). "Granger causality in multivariate time series using a time-ordered restricted vector autoregressive model". In: *IEEE Transactions on Signal Processing* 64.7, pp. 1759–1773.

Simonyan, Karen and Andrew Zisserman (2014). "Very deep convolutional networks for large-scale image recognition". In: *arXiv preprint arXiv:1409.1556*.

Smith, Leslie N (2017). "Cyclical learning rates for training neural networks". In: *2017 IEEE winter conference on applications of computer vision (WACV)*. IEEE, pp. 464–472.

Smith, Stephen M (2002). "Fast robust automated brain extraction". In: *Human brain mapping* 17.3, pp. 143–155.

Smith, Stephen M et al. (2004). "Advances in functional and structural MR image analysis and implementation as FSL". In: *Neuroimage* 23, S208–S219.

Son, Seong-Jin, Mansu Kim, and Hyunjin Park (2016). "Imaging analysis of Parkinson's disease patients using SPECT and tractography". In: *Scientific reports* 6, p. 38070.

Sperduti, Alessandro and Antonina Starita (1997). "Supervised neural networks for the classification of structures". In: *IEEE Transactions on Neural Networks* 8.3, pp. 714–735.

Stöcker, Tony et al. (2005). "Automated quality assurance routines for fMRI data applied to a multicenter study". In: *Human brain mapping* 25.2, pp. 237–246.

Sukhbaatar, Sainbayar, Jason Weston, Rob Fergus, et al. (2015). "End-to-end memory networks". In: *Advances in neural information processing systems* 28.

Suresh, Harini and John V Guttag (2019). "A framework for understanding unintended consequences of machine learning". In: *arXiv preprint arXiv:1901.10002* 2, p. 8.

Sutskever, Ilya, Oriol Vinyals, and Quoc V Le (2014). "Sequence to sequence learning with neural networks". In: *Advances in neural information processing systems* 27.

Szegedy, Christian et al. (2015). "Going deeper with convolutions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9.

Székely, Gábor J, Maria L Rizzo, and Nail K Bakirov (2007). "Measuring and testing dependence by correlation of distances". In: *The annals of statistics* 35.6, pp. 2769–2794.

Tank, Alex, Ian Covert, et al. (2018). "Neural Granger Causality". In: *arXiv preprint arXiv:1802.05842*.

Tank, Alex, Emily B Fox, and Ali Shojaie (2019). "Identifiability and estimation of structural vector autoregressive models for subsampled and mixed-frequency time series". In: *Biometrika* 106.2, pp. 433–452.

Tommasi, Tatiana et al. (2017). "A deeper look at dataset bias". In: *Domain adaptation in computer vision applications*. Springer, pp. 37–55.

Trivedi, Drupad K et al. (2019). "Discovery of volatile biomarkers of Parkinson's disease from sebum". In: *ACS Central Science*.

Vaillancourt, DE et al. (2009). "High-resolution diffusion tensor imaging in the substantia nigra of de novo Parkinson disease". In: *Neurology* 72.16, pp. 1378–1384.

Vassileios Balntas Edgar Riba, Daniel Ponsa and Krystian Mikolajczyk (Sept. 2016). "Learning local feature descriptors with triplets and shallow convolutional neural networks". In: *Proceedings of the British Machine Vision Conference (BMVC)*. Ed. by Edwin R. Hancock Richard C. Wilson and William A. P. Smith. BMVA Press, pp. 119.1–119.11. ISBN: 1-901725-59-6. DOI: 10.5244/C.30.119. URL: https://dx.doi.org/10.5244/C.30.119.

Vepakomma, Praneeth, Otkrist Gupta, et al. (2019). "Reducing leakage in distributed deep learning for sensitive health data". In: *arXiv preprint arXiv:1812.00564* 2.

Vepakomma, Praneeth, Chetan Tonde, and Ahmed Elgammal (2018). "Supervised dimensionality reduction via distance correlation maximization". In: *Electronic Journal of Statistics* 12.1, pp. 960–984.

Vicente, Raul et al. (2011). "Transfer entropy—a model-free measure of effective connectivity for the neurosciences". In: *Journal of computational neuroscience* 30.1, pp. 45–67.

Wang, Rick, Amir-Hossein Karimi, and Ali Ghodsi (2018). "Distance correlation autoencoder". In: *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, pp. 1–8.

Wang, Tianlu et al. (2019). "Balanced datasets are not enough: Estimating and mitigating gender bias in deep image representations". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5310–5319.

Wang, Xin, Weixin Xie, and Jiayi Song (2018). "Learning spatiotemporal features with 3DCNN and ConvGRU for video anomaly detection". In: *2018 14th IEEE International Conference on Signal Processing (ICSP)*. IEEE, pp. 474–479.

Wang, Zongsheng et al. (2019). "Enhancing generative conversational service agents with dialog history and external knowledge". In: *Computer Speech & Language* 54, pp. 71–85.

Wiener, Norbert (1956). "The theory of prediction. Modern mathematics for engineers". In: *New York* 165.

Wilson, Heather et al. (2019). "Serotonergic pathology and disease burden in the premotor and motor phase of A53T α-synuclein parkinsonism: a cross-sectional study". In: *The Lancet Neurology*.

Wismüller, Axel et al. (2021). "Large-scale nonlinear Granger causality for inferring directed dependence from short multivariate time-series data". In: *Scientific reports* 11.1, pp. 1–11.

Woźniak, Tomasz (2015). "Testing causality between two vectors in multivariate GARCH models". In: *International Journal of Forecasting* 31.3, pp. 876–894.

Xu, Zhendi et al. (2021). "BECT spike detection based on novel EEG sequence features and LSTM algorithms". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 29, pp. 1734–1743.

Yao, Kaisheng et al. (2015). "Depth-gated recurrent neural networks". In: *arXiv preprint arXiv:1508.03790* 9, p. 98.

Yao, Shun, Shinjae Yoo, and Dantong Yu (2015). "Prior knowledge driven Granger causality analysis on gene regulatory network discovery". In: *BMC bioinformatics* 16.1, pp. 1–18.

Yasaka, Koichiro et al. (2018). "Deep learning with convolutional neural network for differentiation of liver masses at dynamic contrast-enhanced CT: a preliminary study". In: *Radiology* 286.3, pp. 887–896.

Yu, Meichen et al. (2018). "Statistical harmonization corrects site effects in functional connectivity measurements from multi-site fMRI data". In: *Human brain mapping* 39.11, pp. 4213–4227.

Zeiler, Matthew D and Rob Fergus (2014). "Visualizing and understanding convolutional networks". In: *European conference on computer vision*. Springer, pp. 818–833.

Zhang, Brian Hu, Blake Lemoine, and Margaret Mitchell (2018). "Mitigating unwanted biases with adversarial learning". In: *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 335–340.

Zhang, Tao et al. (2020). "Separated channel attention convolutional neural network (SC-CNN-attention) to identify ADHD in multi-site rs-fMRI dataset". In: *Entropy* 22.8, p. 893.

Zhang, Wei et al. (2018). "A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load". In: *Mechanical Systems and Signal Processing* 100, pp. 439–453.

Zhang, Xi et al. (2018). "Multi-View Graph Convolutional Network and Its Applications on Neuroimage Analysis for Parkinson's Disease". In: *arXiv preprint arXiv:1805.08801*.

Zhang, Yi C and Alexander C Kagen (2017). "Machine learning interface for medical image analysis". In: *Journal of digital imaging* 30.5, pp. 615–621.

Zhao, Jieyu et al. (2017). "Men also like shopping: Reducing gender bias amplification using corpus-level constraints". In: *arXiv preprint arXiv:1707.09457*.

Zhao, Tianxiang et al. (2022). "Towards Fair Classifiers Without Sensitive Attributes: Exploring Biases in Related Features". In.

Zheng, Zhong et al. (2014). "DTI correlates of distinct cognitive impairments in Parkinson's disease". In: *Human brain mapping* 35.4, pp. 1325–1333.

Zhou, Zongwei et al. (2018). "Unet++: A nested u-net architecture for medical image segmentation". In: *Deep learning in medical image analysis and multimodal learning for clinical decision support*. Springer, pp. 3–11.

Zhu, Linchao et al. (2020). "Faster recurrent networks for efficient video classification". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. 07, pp. 13098–13105.

Zonoozi, Ali et al. (2018). "Periodic-CRN: A convolutional recurrent model for crowd density prediction with recurring periodic patterns." In: *IJCAI*, pp. 3732–3738.

View results

Respondent

6 Bilita Mattes

06:31
Time to complete

1. Student Name *

Pranita Patil

2. Student Program *

- (●) Data Sciences
- ( ) Information Systems Engineering and Management
- ( ) Computational Sciences

3. Dissertation Title *

Decorrelated Deep Neural Networks: Learning Bias Invariant & Scanner Independent Features, and Causal Relationships Using a Novel Deep Learning Methods Based on Distance Correlation

4. Provost Recommendation *

- (●) Approve
- ( ) Reject
- ( ) Approve with Edits

More option

5. If Approve with Edits, please use the text box below to include any notes and/or recommendations

6. Provost Acknowledgement: I, the Provost, confirm that my recommendation entered above is complete and accurate. The approval recommendations from the majority of the committee and the Provost certifies the completion of the dissertation defense and accepts and approves the dissertation. *

- (●) I confirm