

**Using Transfer Learning to Leverage Large Un-labelled
Datasets to Improve Classification Models in Cases with Small-
Labelled Datasets: Application to Paediatric Diagnostic and
Prognostic Models**


Paul M. Mwaniki

A thesis submitted to the School of Mathematics, University of
Nairobi for the award of Doctor of Philosophy in Biostatistics.

Date: 31 August 2023

Declaration


This Thesis is my original work and has not been presented for a degree in any other University.

Signature  Date 04/09/2023

Paul M. Mwaniki

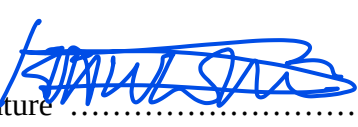
Registration number: I80/55134/2019

This thesis has been submitted for examination with our approval as University supervisors

Signature  Date 04/09/2023

Dr. Timothy Kamanu

School of Mathematics, University of Nairobi, Kenya

Signature  Date 04/09/2023

Prof. René Eijkemans

Julius Center for Health Sciences and Primary Care,

Department of Data Science and Biostatistics, Utrecht University, The Netherlands

Signature  Date 04/09/2023

Dr. Samuel Akech

KEMRI-Wellcome Trust Research Programme, Kenya

Abstract

Diagnostic and prognostic models based on machine learning models can improve diagnosis and identification of patients at risk of adverse health outcomes. Healthcare delivery can thus be improved in low and middle income countries (LMIC) settings, where making accurate diagnosis remains a challenge because of lack of essential laboratory tests and trained medical staff. Training machine learning models requires large labelled datasets which are often unavailable in LMICs. Moreover, models developed in say high-income settings/countries may not be generalizable to LMICs because of differences in setting/context where underlying data was collected. Transfer learning, which stores knowledge gained in solving one problem and incorporates that knowledge while solving a different but related problem, can overcome the challenges of training machine learning models using small labelled datasets. Transfer learning can extract knowledge from large un-labelled datasets or dataset from a different setting and incorporate that knowledge when training models using small labelled datasets, making it potentially applicable to settings with sparse or un-labelled data such as in LMIC. Transfer learning has been applied to natural images and natural language processing, but performance on healthcare data such as medical images, bio-signals and tabular datasets (e.g. clinical signs and symptoms) has not been evaluated.

This study evaluates the use of transfer learning in improving the performance of diagnostic and prognostic models fitted using small labelled datasets. Three types of datasets were evaluated. Firstly, paediatric chest x-rays were classified into WHO standardized categories for diagnosis of pneumonia. Secondly, physiological signals from a pulse oximeter were used to predict hospitalization status, and lastly, tabular data comprising clinical signs and symptoms were used to predict positive blood culture results (bacteremia). The performance of models fitted with and without transfer learning were compared for each dataset.

Transfer learning approaches using multi-task learning and pre-trained models (supervised and unsupervised pre-training) were used to leverage a large chest x-ray dataset from a high income setting to improve performance of models trained on a small chest x-ray dataset from seven LMICs. A novel method incorporating annotation from multiple human readers/annotators of chest x-rays is proposed and evaluated. Self-supervised learning (SSL) methods were used to extract features from pulse oximeter signals and to initialize end-to-end deep learning models for predicting hospitalization status (unsupervised pre-training). Features extracted using SSL were used to predict

hospitalization using logistic regression. Finally, deep learning models for predicting bacteremia using clinical signs and symptoms were compared with logistic regression models. The deep learning models were either initialized randomly or using weights from auto-encoders (unsupervised pre-training).

Supervised and unsupervised pre-training improved classification performance of chest x-rays marginally (accuracy 0.61 vs 0.59 and 0.60 vs 0.59, respectively). Multi-task learning did not improve classification of chest x-rays, while incorporating annotations from multiple human readers had higher performance (accuracy 0.62 vs 0.61). Features extracted from pulse oximeter signals using SSL models were predictive of hospitalization. The AUCs of logistic regression model trained on features extracted using SSL models were 0.83 and 0.80 for SSL model trained using labelled data only and SSL model trained using both labelled and unlabelled data, respectively. End-to-end deep learning models had AUCs of 0.73 when initialized randomly, 0.77 when initialized using SSL model trained using labelled data only, and 0.80 when initialized using both labelled and unlabelled pulse oximeter signals. Logistic regression models for predicting positive blood cultures performed better than deep learning for small training datasets (AUC 0.67 vs 0.62) and marginally worse for large datasets (AUC 0.70 vs 0.71). Initializing deep learning models using weights from auto-encoders did not have any effect on performance on models for predicting bacteremia.

Our results suggest that transfer learning can improve performance of models trained on homogenous data types such as medical images and bio-signals but may have no effect on a heterogeneous tabular data. SSL can be an effective technique for extracting features from bio-signals that could be used to predict various physiological parameters such as respiratory rate. Deep learning models perform worse than logistic regression in predicting bacteraemia using clinical signs and symptoms when the dataset is small.

Acknowledgments

I would like to express my deepest appreciation to my supervisors Dr Timothy Kamanu, Dr Samuel Akech and the late Prof René Eijkemans for providing guidance and feedback throughout the research period. I am grateful to Sub-Sahara Consortium for Advanced Biostatistics (SSACAB) and Initiative to Develop African Research Leaders (IDEAL) whose funding made this research project possible.

I am grateful to KEMRI-Wellcome Trust Research Programme for hosting me and providing the data used in this project. I also thank my colleagues at KEMRI who were always more than willing to provide assistance with procurement, academic writing, IT support, and explanation of various medical concepts.

Lastly, I would like to thank my entire family and friends for their constant encouragement during my entire PhD journey. I would also like to thank fellow PhD students Naomi Muinga and Peter Nguhiu for their moral support through weekly meeting.

List of publications/manuscripts

1. **Mwaniki P**, Kamanu T, Akech S and Eijkemans MJC. Using Machine Learning Methods Incorporating Individual Reader Annotations to Classify Paediatric Chest Radiographs in Epidemiological Studies [version 2; peer review: 2 approved]. Wellcome Open Res 2022, 6:309 (<https://doi.org/10.12688/wellcomeopenres.17164.2>)
2. **Mwaniki P**, Kamanu T, Akech S et al. Using self-supervised feature learning to improve the use of pulse oximeter signals to predict paediatric hospitalization [version 2; peer review: 2 approved]. Wellcome Open Res 2023, 6:248 (<https://doi.org/10.12688/wellcomeopenres.17148.2>)

Table of Contents

Declaration.....	i
Abstract.....	ii
Acknowledgments.....	iv
List of publications/manuscripts.....	v
1 Introduction.....	1
1.1 Background.....	1
1.2 Statement of the problem.....	5
1.3 Objectives of the study.....	6
General objectives.....	6
Specific objectives.....	6
1.4 Significance of the study.....	6
2 Literature Review.....	8
2.1 Introduction.....	8
2.2 Machine learning models.....	8
2.3 Machine learning models in healthcare.....	9
2.4 Semi-supervised learning.....	13
2.4.1 Self-training.....	13
2.4.2 Co-training.....	14
2.4.3 Graph-base self-supervised learning.....	14
2.4.4 Self-supervised deep learning.....	15
2.5 Transfer learning.....	18
2.5.1 Instance-based transfer learning.....	19
2.5.2 Parameter-based transfer learning.....	20
2.5.3 Feature-based transfer learning.....	20
2.5.4 Transfer learning using deep learning.....	21
2.6 Transfer learning for medical images, bio-signals and clinical data.....	24
3 Methods.....	27
3.1 Analysis of Bio-signals.....	27
3.1.1 Data sources.....	27
3.1.2 Signal pre-processing.....	28
3.1.3 Data Augmentation.....	29
3.1.4 Models.....	30

3.1.5 Feature extraction using SSL.....	30
Encoder architecture and training.....	30
Model training and Hyper-parameter optimization.....	31
Feature extraction.....	32
3.1.6 Classification and regression using Linear models.....	32
3.1.7 End to end deep learning.....	33
3.2 Analysis of medical images (Chest radiographs).....	34
3.2.1 Data.....	35
3.2.2 Baseline models.....	38
3.2.3 Pre-trained models.....	39
Supervised pre-training.....	39
Self-supervised pre-training.....	40
3.2.4 Multi-task learning.....	41
3.2.5 Incorporating individual reader annotations.....	43
3.3 Analysis of clinical data (Prediction of positive blood cultures).....	45
3.3.1 Data sources.....	46
Labelled data.....	46
Unlabelled data.....	46
3.3.2 Data imputation and pre-processing.....	47
3.3.3 Baseline model for predicting blood culture results.....	47
3.3.4 Multi-layer perceptron (MLP).....	48
3.3.5 Self-supervised pre-training.....	51
3.3.6 Self-Training.....	52
4 Results.....	54
4.1 Analysis of bio-signals.....	54
4.1.1 Feature learning using contrastive learning.....	54
4.1.2 Classification and regression models using extracted features.....	57
4.1.3 End to end Deep Learning.....	59
4.2 Analysis of chest radiographs.....	60
4.2.1 Supervised Pre-training models.....	60
4.2.2 Unsupervised pre-training.....	61
4.2.3 Multi-task Analysis.....	65
4.2.4 Model with highest performance on PERCH CXRs.....	65
4.3 Analysis of clinical data.....	67

4.3.1 Missing data and imputation.....	68
4.3.2 Auto-encoders.....	71
4.3.3 Prediction of blood culture results.....	72
5 Discussion.....	76
5.1 Analysis of bio-signals.....	76
5.2 Analysis of chest radiographs.....	78
5.3 Analysis of clinical data.....	83
6 Conclusion and Recommendations.....	88
References.....	91
Appendix A: Map of CIN and KHDSS hospitals.....	117
Appendix B: Performance of models predicting blood culture results using clinical signs and symptoms.....	118

List of Figures

Transfer learning methods.....	24
PPG signal analysis.....	28
PPG data augmentation.....	29
Contrastive learning.....	32
Number (%) of chest radiographs from the seven countries.....	36
Random sample of PERCH CXR images.....	38
SSL using contrastive learning.....	41
Multi-task learning.....	43
Model classifying PERCH CXRs conditional on reader.....	45
Nested K-fold cross-validation.....	48
Multi-layer perceptron (MLP) used as backbone of classifiers and auto-encoders.....	50
Denosing and sparse auto-encoders.....	53
Validation accuracy and NCE loss of SSL model trained using contrastive learning.....	55
Dimensionality reduction of featured extracted using contrastive learning using t-SNE.....	56
Dimensionality reduction or raw PPG signals using PCA.....	57
Image1.....	58
t-SNE plot of hidden representation learned using self-supervised learning.....	62
Dis-aggregated AUCs of models trained to classify PERCH CXRs.....	63
Boxplots of model performance on PERCH dataset categorized by model architectures and initialization schemes.....	64
Confusion matrix and lowes plot of classification accuracy against age of model with highest accuracy and AUC.....	66
Grad-CAM visualization of randomly selected images that were correctly classified by the best model.....	67
Proportion of missing data in the predictor variables.....	69
Performance of sparse and denoisiness auto-encoders on the validation set.....	71
Performance of MLPs with different initialization schemes on the validation data.....	74
Map of CIN and KHDSS hospitals.....	117

Acronyms

ANN Artificial Neural Network

ASHA Asynchronous Successive Halving

AUC Area Under the Curve (Receiver Operating Characteristic Curve)

CIN Clinical Information Network

CNN Convolutional Neural Network

CT Computed Tomography

CXR Chest-X-Ray

ECG Electrocardiogram

EEG Electroencephalogram

EHR Electronic Health Record

FDA Food and Drugs Administration

GAN Generative Adversarial Network

GAP Global Average Pooling

MAD Mean Absolute Difference

MCAR Missing Completely At Random

MLM Machine Learning Models

MMD Maximum Mean Discrepancy

MRI Magnetic Resonance Imaging

MUAC Mid-Upper Arm Circumference

KL Kullback–Leibler

LMIC Low and Middle Income Countries

MLP Multi-layer Perceptron

MSE Mean Squared Error

NCE Noise Contrastive Estimation

NIH National Institute of Health
PACS Picture Archiving and Communication Systems
PBT Population Based Training
PCA Principal Component Analysis
PERCH Pneumonia Etiology Research for Child Health
PPG Photoplethysmography
SGD Stochastic Gradient Descent
SD Smart Discharges
SpO2 Peripheral Oxygen Saturation
SSL Self-Supervised Learning
SVM Support Vector Machine
t-SNE t-distributed Stochastic Network Embeddings
WHO World Health Organization

1 Introduction

1.1 Background

Diagnostic and prognostic models developed using machine learning models can improve diagnosis and prediction of clinical outcomes (F. S. Collins & Varmus, 2015; Yadav et al., 2021). Diagnostic models predict the presence or absence of a medical condition of interest, presence of disease-causing organism (etiology), or abnormality. Prognostic scores predict the likelihood that a patient will experience an outcome (e.g. discharged alive/dead or experiencing an adverse event after receiving a treatment). Diagnostic and prognostic models can be useful in low and middle income countries (LMICs) where diagnostic laboratories are poorly equipped and medical specialist are often few or lacking (Yadav et al., 2021). Accurate machine learning models for medical applications are difficult to develop because training the models require large labelled datasets (Althnian et al., 2021; Raghu et al., 2019; Sordo & Zeng, 2005).

Large training datasets are often difficult to obtain in medical settings and confidentiality concerns has been cited as a major cause (Price & Cohen, 2019). Furthermore, there are cost constraints of labelling medical data because that requires specialists or expensive and time consuming tests. Medical images for instance may require trained radiologists to annotate each image which is expensive and time-consuming. Data collected in routine healthcare setting may not lend itself in forms fit for training machine learning models because they are often intended for administrative purposes. Available data have frequent missingness and lack standardized definitions of various clinical outcomes and procedures, undermining their utility for machine learning (Lujic et al., 2014; Rumisha et al., 2020).

Training of machine learning models using small datasets is challenging because it may limit the complexity of models that can be derived without over-fitting, i.e., the model performs well on the training dataset but poorly on unseen/test data (Kohavi, 1996; Luxburg & Schölkopf, 2008). Model complexity refers to how flexible the model's decision boundary can be. More complexity in machine learning models is desirable because it allows capturing the complex relationship between

the predictors and the outcome (Gilad-Bachrach et al., 2003). More complexity is useful with highly structured datasets such as images and physiological signals where the relationship between the raw data and the output cannot be captured by simpler models (LeCun et al., 1999). Lack of sufficient model complexity may result to models with high bias, whereby the difference between the observed and predicted values are large on average.

Training machine learning models with small dataset is also challenging when the number of predictor variables is large relative to the number of observations, a phenomenon known as curse of dimensionality (Bellman, 1961). Datasets such as medical images, omics and physiological signals are high dimensional. The higher the dimensions of the input data, the larger the size of training data required to train the models. The number of observations required in the training datasets increases exponentially with the dimensions of the inputs. The exponential growth in number of observations required can be illustrated using a simple classifier formulated such that the input space is split into cells of unit size and prediction for a new point is obtained by averaging the class of training data points in the cell containing the new point. The number of cells would increase exponentially with increase in number of dimension reducing the average number of points in each cell, and increasing the number of cells without any training data points. Furthermore, it has been shown that in high dimension, any new observation is likely to lie at the boundary rather than within the rest of the data points (Balestriero et al., 2021). Therefore, making predictions in tasks involving high dimensional input space often involves extrapolation as opposed to interpolation, making generalization more difficult.

The challenges of training machine learning models using small labelled datasets are compounded by model development pipelines that require splitting the data into training, validation and test sets, which further reduces the number of data points available for estimating the parameters and fine-tuning machine learning models. The test data is required for estimation of model performance on yet unseen data, and give an indication of how well the model will perform once deployed. The validation data is required for hyper-parameter optimization which cannot be estimated when training the model. For instance, finding optimal hyper-parameters for regularization can not be estimated during training because hyper-parameter configuration that result to the highest performance on the training dataset would be selected, which would likely result to over-fitting.

Large training datasets are often available in high income settings, however, models fitted using data from high income setting may not generalize to LMICs because datasets from different settings may follow different distributions (Shimodaira, 2000). There are external validation studies showing degradation of model performance on datasets originating from settings different from those the training data originate (G. S. Collins & Altman, 2009; Ogero et al., 2023). Differences in the distribution of the outcome such as differences in prevalence of a condition of interest manifests as poor model calibration. Poor calibrated models with acceptable discriminative performance can be corrected using calibration techniques (Kull et al., 2017), but calibration cannot address covariance shift, where the distribution of predictors change. Covariance shift can arise due to differences in populations sampled or differences in equipment used to measure the predictors (e.g. differences in chest radiographs machines used, or procedures used in preparing the medical images). Furthermore, even within the same context, the distribution of data can change over if there is concept drift, wherein models trained using data from the setting of interest may not generalize to the same setting after a while (Gama et al., 2014). Concept drift can occur in medical diagnostic models where new interventions are implemented from time to time. For example, introduction of pneumococcal and other vaccines can change the distribution of causes/etiologies of infectious diseases such as pneumonia and invasive bacterial diseases. Therefore, if the performance of a prediction model is sensitive to changes in etiologies, then any intervention that changes the distribution of etiologies would degrade the performance of such models.

Lack of transferability of models fitted in one setting to other settings can have adverse effects to certain patient populations; It has been argued that pulse oximeters, whose development require calibration using data, may not be as accurate for patients with dark skin because data from such minorities is not included in calibrating the devices during the development stage (Sjoding et al., 2020). Variation in performance of machine learning models due to race has also been observed in dermatology (Adamson & Smith, 2018; Navarrete-Dechent et al., 2018). Race disparity in performance of the models for dermatology can be explained by variability in skin complexion, but performance disparities have also been observed in medical images where information about race is not expected to be present (Seyyed-Kalantari et al., 2021). Banerjee (2021) was able to predict patients' race from chest X-rays despite there being no known clinical features related to race in the medical images. The models could still predict patients' race after data augmentation, despite data augmentation being recommended as solution for developing more generalizable models. Data augmentation involves generating new data point by perturbing existing data with random noise.

Issues that affect transferability of models from one context to another may require refitting the models in each context the model will be used, and thus require training data from each context. Collecting sufficient training data in each context of interest may be unfeasible due to cost and time constraints. This study investigates transfer learning using large unlabelled datasets or labelled dataset from different contexts to address the lack of transferability of machine learning models across different contexts. Transfer learning may improve performance of models fitted using small labelled datasets by incorporating knowledge obtained from models fitted using data from a different context or models trained to solve a different but related tasks (Farahani et al., 2021; Zhuang et al., 2020). Model trained using data from one context can be modified to work in a different context through domain adaptation (Sukhija et al., 2016). A different form of transfer learning involves modifying models trained to solve one task (source task) to perform a different task (target task). For instance, a model developed to classify chest radiographs for one condition can be adapted to classify chest radiographs for a different condition. The inputs for the source and target tasks can come from the same distribution but most observations may be missing the target outcome. Such scenarios where most of the observations are missing the outcome of interest can arise in many medical applications where the inputs/predictors are collected as part of routine care but obtaining the outcome requires specialists or expensive tests. For instance, large databases of chest radiographs may arise naturally from digital chest x-ray machines, but specialists such as radiologists may be required to annotate each image. Clinical notes accompanying such chest radiographs may not be suitable labels for training machine learning models because of lack of standardization.

In this study we evaluated the use of transfer learning in improving diagnostic and prognostic models trained using small labelled datasets. Transfer learning was used to extract information from large unlabelled datasets or large datasets from a different context with the aim on improving models fitted using small labelled datasets from LMICs. Three datasets available in LMIC settings with potential of improving delivery of care were used: bio-signals, medical images and tabular data. The bio-signal dataset comprised of photoplethysmogram (PPG) signals obtained using a pulse oximeter, a cheap and non-invasive device used routinely in clinical practice to measure oxygen saturation and heart rate. We trained machine learning models to predict hospitalization of children seeking care at an outpatient department of a public hospital in Kenya using PPG signals as predictors. The medical images dataset comprised of chest radiographs (CXRs) used in diagnosis of

various chest conditions. We trained machine learning models to classify the CXR images according to World Health Organization (WHO) standardized methodology of classifying CXR for pneumonia diagnosis in children. The tabular dataset comprised of clinical signs and symptoms (fever, diarrhoea, difficulty breathing, e.t.c) commonly evaluated by clinicians during assessment to aid in making various clinical decisions including diagnosis. We used the clinical signs and symptoms as predictors of bacteremia (measured using a blood culture test), a life-threatening condition where bacterial infections enter the bloodstream and cause disease. For each of the three data sets, a larger unlabelled data-set or a data-set from a different settings was obtained with the aim of improving performance of classification models fitted with the smaller labelled dataset. A larger unlabelled PPG signals datasets was obtained from the Smart Discharge (SD) study in Uganda (Wiens et al., 2021a). Additional publicly available CXR dataset from a high income settings and comprising mostly of adult CXRs was downloaded from the internet, while an unlabelled tabular data on clinical signs and symptoms was obtained from Clinical Information Network (CIN), a dataset collected from 14 public hospitals in Kenya (Tuti et al., 2016).

1.2 Statement of the problem

Machine learning models can improve the accuracy of making diagnoses and hence result in improved healthcare delivery in LMICs, where clinical teams often encounter problems making correct and timely diagnosis. For example, clinicians can be provided with timely and accurate determination of possible etiologies for common ailments such as pneumonia, meningitis and sepsis, and accurate and rapid interpretation of medical images such as chest x-rays.

Large datasets required to fit machine learning models with high sensitivity and specificity are however often unavailable, incomplete, or originate from different contexts. For instance, when fitting models for predicting etiology using a set of clinical signs and symptoms, the clinical data may be easy to collect from medical records, but the laboratory results may be unavailable because the required laboratory facilities are lacking in low resource settings. As a result, majority of the observations in the training data are unlabelled. Large datasets may be available in high income settings, but machine learning models work best when the training and testing data come from similar distributions, limiting generalizability of models fitted using data from high income settings to LMICs. Transfer learning might circumvent issues arising from insufficient labelled datasets

while fitting machine learning models (MLMs) in LMICs by incorporating knowledge gained when fitting similar models using data from high income settings or large unlabelled datasets.

1.3 Objectives of the study

General objectives

This study seeks to use transfer learning to leverage large unlabelled datasets to improve performance of paediatric diagnostic and prognostic models when only small labelled datasets are available. The study considers three types of datasets: physiological signals, medical images, and clinical signs and symptoms.

Specific objectives

The specific objectives of the study are to:

- i. Develop MLMs for predicting pediatric admission based on physiological signals obtained from a pulse oximeter (Photoplethysmograph, i.e. PPG).
- ii. Develop models for classifying paediatric chest x-ray images into WHO standardized classification for pediatric chest x-rays.
- iii. Predict positive blood cultures using clinical signs and symptoms for children admitted in public hospitals in LMICs.

1.4 Significance of the study

The results of this study can inform on how performance of diagnostic and prognostic models developed in LMICs can be improved using transfer learning. Such models may be useful in standardizing interpretation of medical images such as chest x-rays, improving estimation of the burden of pneumonia as well as evaluation of efficacy of interventions for pneumonia such as vaccines.

The methods developed here for classifying pulse oximeter signals can allow novel uses of pulse oximeters beyond estimation of oxygen saturation and heart rate. Features extracted from raw pulse

oximeter signals using the approaches developed here may improve the development of prediction models for diverse medical outcomes such as hospitalization and classification of severity of illness.

The diagnostic models can provide accurate diagnoses that are provided by medical specialists such as radiologists or medical tests that can be expensive, unavailable, or results come after several days. In such cases the model would prevent delay in provision of potentially lifesaving interventions. Such models can be especially useful in low income settings where essential medical tests are often unavailable or too expensive for most patients exacerbating inequalities in access to healthcare (Petti et al., 2006; Wilson et al., 2018). For example, lack of diagnostic services has been associated with over-use of antimicrobials which is associated with anti-microbial resistance (Ayukekbong et al., 2017). Machine learning models may also offer alternatives to medical tests that require invasive procedures and therefore reduce discomfort or infection to patients.

This study also seeks to demonstrate how large unlabeled datasets can be leveraged to improve performance of prediction models when labeled data is scarce, given that high quality medical data is often expensive to collect with respect to time and monetary cost.

2 Literature Review

2.1 Introduction

This review describes existing literature on the types of MLMs, application of MLMs in diagnostic and prognostic models, and various types of semi-supervised and transfer learning methods. Sections 2.1 and 2.3 describes the main branches of machine learning and applications of machine learning models in healthcare, while sections 2.4 and 2.5 describe semi-supervised learning and transfer learning methods that can be used to leverage unlabelled data-sets to improve performance of models fitted using small labelled data-sets. Applications of transfer learning models in bio-signals, medical images and tabular data are described in section 2.6.

2.2 Machine learning models

MLMs are developed by learning a function that accomplishes a task of interest. using data known as the training set. The generalizability of MLMs is determined by assessing their performance on test dataset. Inductive machine learning models such as logistic regression learn general rules using the training data that can later be applied to yet unseen test data. On the other hand, transductive learning model such as Transductive Support Vector Machine (TSVM), require the model to be refitted every time a new test observation is encountered (Kondratovich et al., 2013; Vapnik, 1998). Machine learning models can broadly be classified into supervised learning, unsupervised learning, and reinforcement learning models (Awad & Khanna, 2015).

Supervised learning involves training machine learning models with input-output pairs and learns a function that maps the inputs to outputs. Such models performs a classification task if the output is categorical or regression if continuous. Supervised learning algorithms are categorized into generative and discriminative algorithms (Ng & Jordan, 2002). Given a set of inputs X and an outcome y , generative models learn the class conditional probabilities $p(X|y)$ and apply Bayes theorem to make predictions of the outcome given the inputs $p(y|X)$. Examples of generative supervised learning algorithms are naive bayes classifier and linear discriminant analysis (H. Zhang, 2004). Discrimination models avoid the challenge of estimating class conditional probabilities by estimating the conditional distribution $p(y|X)$ directly. Examples of discriminative models include logistic regression and artificial neural networks.

Unsupervised learning models, sometime referred to as self-supervised learning models, are provided with the inputs only and learn a compressed representation of the inputs. The compressed representation may be distributed or non-distributed (I. Goodfellow et al., 2016). For non distributed representation, each input is mapped to a single category. Clustering algorithms such as the K-means clustering that assigns each input into one of several categories are examples of non-distributed representations. On the other hand, distributed representation learning algorithms such as Principal Component Analysis (PCA), map each input into a space with multiple dimensions. Each dimension can be thought of as representing different latent attributes of the inputs.

Reinforcement learning models learn the optimal sequence of actions in a given environment that maximizes a reward (Sutton, 1998). Unlike supervised learning where each input is paired with an output, rewards in reinforcement learning are only available at the end of an episode (series of steps). The model, often referred to as an agent, takes in measurements of the environment at each step and makes a prediction on the optimal action. At the end of an episode, the model receives feedback on whether the actions taken resulted in a desirable outcome. Reinforcement learning is widely used in robotics (Kormushev et al., 2013).

2.3 Machine learning models in healthcare

Predictive models in healthcare have traditionally been developed using linear and logistic regression models for continuous and categorical outcomes, respectively. However, linear models are sub-optimal for non-linearly separable data, and their use might result to under-fitting. Moreover, the relationship between the inputs and the outcomes cannot be captured by linear models for highly structured datasets such as medical images and bio-signals (LeCun et al., 2015). For such datasets, extensive feature engineering is required to extract hand-crafted features from the inputs in order to use such features as predictors with linear models. Hand-crafted feature engineering requires extensive domain knowledge about the relationship between the inputs and the outcomes which may hinder the development of models for novel outcomes/tasks. For instance, classification of images involved hand-crafted feature extractors for edges, corners, blobs and ridges (Mahony et al., 2020). In addition, hand-crafted features are often task specific and therefore require development of new features for different tasks. For example, features that are relevant in classifying images of animals may not be relevant for classifying medical or satellite images. The development of hand-crafted features for bio-signals requires knowledge on signal processing

techniques and intimate knowledge about human physiology (Supratak et al., 2016; Tomé et al., 2013). In contrast, over-parameterized machine learning models such as deep learning do not require hand-crafted features and these allow models with similar architecture to classify images from different domains without requiring any modification. Furthermore, deep learning models used with tasks involving images can also be used for tasks involving physiological signals, natural text, or time series data.

MLMs have been applied in all aspects of healthcare delivery including administrative tasks, disease outbreak prediction, drug development, development of diagnostic devices, treatment recommendations and prediction of disease progression and other endpoints such as re-hospitalization and mortality (Cuttillo et al., 2020; Davenport & Kalakota, 2019; Rajkomar et al., 2019). MLMs can ease administrative burden arising from management of patient records through automatic transcription of unstructured clinician notes or audio recordings into data formats that are compatible with electronic health records (Kaufman et al., 2016; Willyard, 2019). MLMs can also be used to link medical records stored in diverse databases, and thus enable tracking of patients across visits or points of care (Redfield et al., 2020; Sauleau et al., 2005). MLMs have also been used to automate medical insurance claims processing and detect fraudulent claims, which has hastened processing of claims and reduced operational costs (Singh & Urolagin, 2021).

MLMs have been used in pharmaceutical industry to identify new drug candidates by predicting properties of various compounds using their molecular structures (Pham et al., 2021). MLMs have been used to model pharmaco-kinetics and toxicological properties of drug candidates, hastening pre-clinical and clinical stages of drugs development (McComb et al., 2021; Miljković et al., 2021). MLMs can benefit drug manufacturing by predicting properties of chemical reactions which is essential for drug manufacturing (Schwaller et al., 2018, 2021). Moreover, machine learning models have been used to identify new uses for approved drugs by identifying related disease pathology (Rodriguez et al., 2021; Urbina et al., 2021; F. Yang et al., 2022).

MLMs for clinical decision making have been applied at population or individual level. Population level MLMs aim at making predictions on groups of individuals at a given time or/and geographical location. On the other hand, individual level MLMs make predictions about a single individual given that individual's characteristics. Individual level models are behind research in personalized medicine.

MLMs have enhanced population level prediction by enabling analysis of unstructured and semi-structured data types such as images, videos, text and audio, which classical statistical models are not equipped to handle. For instance, machine learning models for natural language processing (NLP) have been used to predict disease outbreaks using media articles and social media posts, thus enabling fast and cheap detection of possible disease outbreaks (Ghosh et al., 2017; Kim & Ahn, 2021; Şerban et al., 2019; Y. Zhang et al., 2020). MLMs have been used to model the distribution of disease vectors to shed light on the dynamics of infectious diseases transmission (Ding et al., 2018; G. Ren & Wang, 2014; Tripathi et al., 2020). MLMs have also been used in public health to estimate burden of non-communicable diseases and public health surveillance (Brownstein et al., 2009; Haneef et al., 2021; Mooney & Pejaver, 2018).

Machine learning models for individual clinical decision making utilize individual level predictors such as demographic information, clinical signs and symptoms, biomedical images, bio-signals, omics, wearable sensors, and laboratory measured biomarkers as inputs (L. Chen, 2020). These models can be classified as diagnostic or prognostic models. Diagnostic models predict the presence of a condition while prognostic model predict an outcome of interest at a future time (Hendriksen et al., 2013). Diagnostic models are useful in diagnoses of diverse conditions ranging from infectious diseases, oncology, dermatology, and mental health (Jaimes et al., 2004; Y. Liu et al., 2017; Navarrete-Dechent et al., 2018). There are over 138 machine learning applications that have been approved for clinical use by the Food and Drugs Administration (FDA), most of which are related to radiology (Wald et al., 2021). The complexity of diagnostic and prognostic models range from systems that make predictions about a single medical condition/disease to decision support systems covering multiple aspects of clinical care (Giordano et al., 2021; Lynn, 2019).

The performance of most machine learning models depend on how the input data is represented (Bengio et al., 2014). Therefore, extensive feature engineering is often necessary before training the machine learning models. Feature engineering in clinical prediction modeling often requires domain knowledge about the relationship between the inputs and the outcome of interest (Roe et al., 2020). Such domain knowledge can be lacking in novel applications, and thus hinder the use of machine learning methods.

Structured data such as clinical signs and symptoms have been analyzed using a wide spectrum of machine learning models including linear/logistic models, decision trees, random forest, artificial neural networks, boosted trees, and support vector machines (Christodoulou et al., 2019). On the other hand, developing machine learning models using inputs that are not structured is challenging. Text, medical images and bio-signals require extensive feature engineering to transform them into feature vectors, which limits the types of machine learning models that can be used. As a result, machine learning methods that have been used to develop diagnostic and prognostic models depend heavily on the structure of input data. Machine learning models for analysis of bio-signals have relied on classical signal processing techniques to extract features from raw signals (Krishnan & Athavale, 2018; Rajoub, 2020). The extracted features are then used as predictors of outcomes of interest using diverse regression and classification models. Features are extracted in time domain or from the frequency domain following signal decomposition using Fourier or wavelet transforms (Krishnan & Athavale, 2018). Recent approaches of analyzing bio-signals and medical images avoid manual feature engineering by using deep learning (Wu et al., 2018).

Deep learning models are inspired by biological neurons and consist of artificial neurons that are stacked to form layers such that the outputs of one layer acts as inputs for the subsequent layer (I. Goodfellow et al., 2016). Each artificial neuron performs a linear transformation of the input followed by a non-linear transformation. The stacked layers of a deep learning model enables the model to approximate highly non-linear functions, and are capable of representing any continuous function given enough artificial neurons in the hidden layers (Cybenko, 1989). Consequently, deep learning models are capable of approximating the relationship between the inputs and the outputs regardless of structure of the inputs. There are different architectures of deep learning model that can be broadly classified as fully connected neural networks, convolutional neural networks (CNNs), recurrent neural networks (RNNs), and graph neural networks (GNNs). CNNs are often used to classify images, while RNNs are used in analysis of time-series data (Barragán-Montero et al., 2021; Erickson et al., 2017; S. K. Zhou et al., 2021). GNNs are used to analyze data that can be organized in graphs where entities are represented by the node of the graphs and the relationship between entities are represented as edges (J. Zhou et al., 2020).

CNNs have been used to analyze chest radiographs (CXRs), Magnetic Resonance Imaging (MRI), Computed Tomography (CT) scans, and pathology slides (Castiglioni et al., 2021). Machine

learning applications for medical images can be classified into detection and segmentation tasks. Detection tasks involve recognizing presence of conditions of interests on medical images such as pneumonia from chest radiographs. Segmentation of medical images involve creating a bounding box of a region of interest that can be overlaid on the original image. Such regions may include an organ such as liver or abnormalities like tumors.

Machine learning models that have been developed for diagnostic and prognostic models have often employed supervised learning algorithms that require each observation in the training data to have a corresponding outcome (labels). However, obtaining labels for every observations in the training data is often expensive and not feasible at scale. There are many situations where a large dataset exist but only a small fraction of the dataset is labelled. In such cases, semi-supervised learning can be used to exploit the unlabelled data to improve performance of fitted models.

2.4 Semi-supervised learning

Semi-supervised learning fall between supervised and self-supervised learning in that not all inputs are labelled (Chapelle et al., 2006; Zhu, 2005). Semi-supervised learning models are believed to be useful if the marginal distribution of the inputs $p(X)$ is informative of the distribution of the outcome given the inputs $p(y|X)$. Semi-supervised learning algorithms make implicit assumptions about the relationship between data points that are close to each other in the input space (Chapelle et al., 2006). The smoothness or continuity assumption states that data point that are close to each other in the input space are also likely to share the same outcome (same label for classification tasks or similar values for regression tasks). The smoothness assumption gives rise to clustering assumption which states that points in the same cluster are likely to have the same outcome. Clustering assumption does not imply that each cluster consists of observations from the same class, but the decision boundary of the classifier should lie in low density regions. Other semi-supervised learning algorithm assume that the data lie on a low dimensional manifold, and data points that are closer in the low manifold are likely to have similar labels.

2.4.1 Self-training

Self-training is among the first semi-supervised learning method described in literature (Agrawala, 1970; Chapelle et al., 2009; Fralick, 1967; Scudder, 1965). A self-training models is first developed using the labelled data only, and the model is then used to predict the outcome on the unlabelled data to create pseudo-labels. A fraction of the unlabelled data which the model is most confident

about and corresponding pseudo labels are added to the training data. Self-training is based on the assumptions that predictions with high confidence are also more likely to be correct. For instance, in binary classification, predictions with probabilities close to 0 or 1 are more likely to be correct compared to predictions with probabilities close to 0.5. The machine learning model is then refitted with the expanded training data. The process of adding more observations from the yet to be labelled dataset is repeated until all unlabelled data is added to the training set or model performance does not improve. Self-training is easy to implement because any supervised learning algorithm can be used as a base learner. However, self-training models can over-fit to incorrect pseudo labels due to confirmation bias (Arazo et al., 2020).

2.4.2 Co-training

Co-training is a semi-supervised learning method closely related to self-training. Co-training assumes that the input variables can be divided into two views such that each view is sufficient for predicting the outcome (Blum & Mitchell, 1998; National Research Council, 2004). Ideally the two views should be conditionally independent given the outcome. Co-training has been used to classify web pages where hyperlinks pointing to a page and the text in the web page were treated as two views (Blum & Mitchell, 1998). Two classifiers are fitted using the two views and predictions are then made on random and disjoint observations from the unlabelled data. For each classifier, the observations the model is most confident about are added to the labelled data. The process of fitting the two classifiers and adding observations to the labelled data dataset is repeated like in self-training. Co-training can still be useful if a natural split in the inputs is not known in what is referred to as single view co-training (J. Du et al., 2011) . However, it is not clear whether the assumption that the two views are conditionally independent given the outcome can be relaxed without compromising performance of co-training, and the method is only useful if the two classifiers give differing predictions in some observations where one classifier is correct (Kroegel & Scheffer, 2004).

2.4.3 Graph-base self-supervised learning

Other approaches for semi-supervised learning are based on graph algorithms (Bengio et al., 2006). A graph is constructed using both labelled and unlabelled data, with node representing observations and edges representing similarity between observations. Labels are then propagated from labelled nodes to all nodes in the graph. For a graph with a weight matrix W , with elements w_{ij} representing the similarity between two inputs x_i and x_j , edges are constructed for all nodes with $w_{i,j}>0$. The

weight matrix can be constructed by computing the distance (e.g. euclidean) between each element and all other elements in the dataset, resulting in a dense graph where each node is connected to all other nodes. However, such an approach is computationally expensive and not feasible for large datasets. A sparse graph can be constructed using KNN such that for each node the weights of k nearest neighbours are assigned a value of one and all other nodes are assigned a weight of zero (Ebert et al., 2011; P. Zhang et al., 2015). The label of each unlabelled nodes can be obtained by using weighted mean of all labelled neighbours, with weights obtained from the weight matrix (Zhu & Ghahramani, 2002). Graph based semi-supervised learning methods are transductive learning algorithms, where the model cannot be used to label new observation that are not present during model training. As a result, graph-based algorithms are computationally expensive and unsuitable for many real life applications.

2.4.4 Self-supervised deep learning

The flexibility of deep learning has seen a wide array of semi-supervised learning methods in recent years (Ouali et al., 2020; Vanyan & Khachatryan, 2021; X. Yang et al., 2021). Some approaches of semi-supervised deep learning are based on pseudo labels and extend self-training, co-training, and label propagation on a graph (Arazo et al., 2020; Iscen et al., 2019; Qiao et al., 2018). In teacher-student algorithms, a teacher neural network is trained on the labelled data only and a student neural network is trained on both labelled and unlabelled data, with predictions of the teacher neural network acting as pseudo labels for both labelled and unlabelled observations (Xie et al., 2020). The teacher neural network is often smaller than the student network to avoid over-fitting on small labelled datasets. Others have fitted teacher-student algorithms in iterative manner where the student network becomes the teacher network in the next iteration, and the size of the student neural network is increased at each step (M. Tan & Le, 2020). Ke (2019) proposed using two neural networks of same size instead of having teacher and student networks. In what they termed as dual students, the student network that had stable predictions at each training iteration provided pseudo labels for the other network. Stability was defined as root mean square between model prediction of perturbed and unperturbed inputs. Self-training has also been extended by simultaneously optimizing the values of pseudo-labels alongside model parameters (Shi et al., 2018). The labels of unlabeled observations are treated as variables/parameters to be estimated during model training.

Other approaches of semi-supervised learning using deep learning rely on the cluster assumption, where small perturbations of the inputs are not expected to change the labels (Zhu, 2005). The

cluster assumption is enforced by having a regularization term that forces an unlabelled input and its perturbed version to have the same label. For instance, given a neural network classifier $f_\theta(\cdot)$, we can compute the distance between the neural network output for unlabelled inputs X and their perturbations \hat{X} , $d(f_\theta(X), f_\theta(\hat{X}))$. Some of the commonly used distance functions are mean square error (MSE), Kullback–Leibler (KL) divergence and Jensen-Shannon divergence. A regularization term is then added to the supervised loss computed on the labelled data (e.g. cross-entropy loss). Ladder networks enforce the clustering assumption via regularization by ensuring that small perturbations of the inputs of hidden layers do not change their outputs (Rasmus et al., 2015). A network is trained to classify the labelled data. The trained neural network model is then extended to accommodate unlabelled data by adding a decoder network that reconstructs the outputs of the hidden layers that have been perturbed using Gaussian noise. The mean squared error between the the decoded outputs of perturbed and unperturbed hidden layer activations is added to the cost function as a regularization term. Pi-Model is a simplification of ladder network where instead of have a decoder network, the regularization term is obtained by computing the mean square error between the output of hidden layers of perturbed and unperturbed inputs (Laine & Aila, 2017). Temporal ensembling of model predictions are used instead of the model prediction at each training iteration to reduce model optimization difficulties arising from noisy predictions. Tarvainen (2018) proposed applying exponential moving average to the neural network weights instead of applying temporal ensembling to the predictions. The model derived from exponential averaging of the weights is treated as a teacher network, and was shown to be robust to confirmation bias observed in self-training.

Another variation to semi-supervised deep learning is based on Generative Adversarial Networks (GANs) (X. Liu & Xiang, 2020). GANs are generative models that learn the distribution of the inputs using two neural networks, where one network known as a generator G is trained to generate fake inputs and the other network known as a discriminator D is trained to distinguish between real and fake inputs (I. J. Goodfellow et al., 2014). The generator and the discriminator are trained simultaneously until the fake inputs generated by the generator are similar enough to real inputs for the discriminator to tell them apart. Once trained, the generator network G should be able to generate fake observations sampled from the distribution of the inputs given a vector of random noise z . Some GANs for semi-supervised learning follow extensions of GAN that have discriminators that predict the class of the input alongside whether the input was generated (Salimans et al., 2016). Given a classification task with k classes, the discriminator network is

trained to classify inputs into $k+1$ classes, where the additional class is for whether or not the input was generated or real. Li (2017) observed that the discriminator network had challenges performing both classification and discriminating real observations from fake and instead added a separate neural network for classification. A variation of GAN for semi-supervised learning is based on the manifold assumption, where the decision boundary of the classifier is expected to be smooth in the low dimension manifold. GANs have been shown to learn the low dimension manifold of the inputs (Radford et al., 2016), and thus can be used to enforce smoothness of the decision boundary of classifiers by providing manifold regularization approximations (L. Chen, 2020; Lecouat et al., 2018). The regularization ensures that the fake observations generated by the generator given a vector of random noise z will be classified the same as observations generated from slightly perturbed values of z .

Sometimes it is possible to obtain labels for some of the unlabelled observations, but cost and other constraints limit the number of observations that can be labelled. Active learning can be used to identify unlabelled observations that if labelled would improve the performance of the classification or regression models the most (Budd et al., 2021; Cohn et al., 1996; P. Ren et al., 2021). A model is trained on the labelled data and ranks the unlabelled data to determine which observations should be labelled next. The highest ranked observations are labelled (e.g., by having a radiologist annotate the selected medical images) and added to the labelled dataset. The model is then refitted or fine-tuned using the now larger training dataset. The procedure of identifying which observations should be labelled and retraining the model is repeated until the labelling budget is exhausted. Two main ways of ranking observations for labelling are used: uncertainty sampling and diversity sampling. Uncertainty sampling selects unlabelled observations whose predictions are most uncertain for labelling (Tong & Koller, 2000; Y. Yang et al., 2015). Therefore, the model has to be able to represent the uncertainty of predictions in unlabelled data, which limits the type of models that can be used. For instance, bayesian neural network have been used to estimate uncertainty in the predictions of neural networks (Gal et al., 2017; Tong & Koller, 2000; Tsymbalov et al., 2019). Diversity sampling aims at selecting observations that are as dissimilar as possible by exploiting the distribution of the data (Brinker, 2003; Xu et al., 2007). Diversity sampling identifies observations that would increase the representativeness of the labelled data in relation to the distribution of the entire data-set (B. Du et al., 2019). For instance, unsupervised clustering methods can be used to identify observations in clusters that do not have any labelled data (Ienco et al., 2013). Yang (2015) observed that unlabelled observations obtained using uncertainty sampling are often very similar

limiting the additional information that can be extracted. Therefore, some active learning approaches incorporate both uncertainty and diversity sampling (Shao et al., 2019; G. Wang et al., 2017). Uncertainty sampling is first used to identify candidate observations and a second algorithm is used to select a subset of identified observations that are as diverse as possible.

The key assumption in semi-supervised and active learning is that the labelled and unlabelled datasets have the same distribution, and thus the unlabelled data can be considered as missing completely at random (MCAR) (Wasserman & Lafferty, 2008). Therefore, the large datasets available in high income settings (or different contexts) cannot be used to improve performance on models fitted using small datasets in low income settings (or dissimilar contexts). The limitations of datasets originating from different settings may be overcome using transfer learning.

2.5 Transfer learning

Transfer learning, sometimes referred to as domain adaptation, aims at improving a task with insufficient data using data originating from a different setting (Farahani et al., 2021; Weiss et al., 2016; Zhuang et al., 2020). Transfer learning can also use model trained to solve one or more tasks that are not of interest to improve performance of models for similar tasks which are of interest. Transfer learning is inspired by human learning where learning a task can be easier if one has already learned a similar task. In transfer learning, we hope to use knowledge from a source domain to improve performance on a task in the target domain.

Formally, given a domain D consisting of feature space X having marginal distribution $p(X)$, i.e. $D = \{X, p(X)\}$. A task T is defined on a domain D and consists of a label space Y and a decision function $f(\cdot)$ that maps the inputs to the labels .i.e $T = \{Y, f(\cdot)\}$. The source and target domains can thus be defined as $D_s = \{X_s, p(X_s)\}$ and $D_t = \{X_t, p(X_t)\}$ respectively. Likewise, the source and target tasks can be defined as $T_s = \{Y_s, f_s(\cdot)\}$ and $T_t = \{Y_t, f_t(\cdot)\}$, respectively. Transfer learning aims at improving the target task T_t in the target domain D_t using knowledge obtained from the source domain D_s and the source task T_s . Where the source domain is not the same as the target domain or/and the source task is not the same as the target task. Data-sets originating from different contexts/settings are expected to have different distributions and can therefore be considered to originate from different domains.

Pan (2010) categorized transfer learning into inductive, transductive, and unsupervised transfer learning. Inductive transfer learning has different source and target tasks $T_s \neq T_t$, but the domain can be the same or different. The source and the target domains in transductive transfer learning are different $D_s \neq D_t$, but the source and target tasks are the same. The source domain can have a lot of labelled data while the target domain has little or no labelled data. On the other hand, unsupervised transfer learning has no labelled data in either the source or target domains. Transfer learning can also be categorized into heterogeneous and homogeneous transfer learning depending on whether or not feature/input spaces are the same in source and target domains (Farahani et al., 2021; Weiss et al., 2016).

Heterogeneous transfer learning is used when the feature space in the source and target domains are not the same $X_s \neq X_t$, and involves transforming the feature space of the source and/or target domain to a common feature space (Day & Khoshgoftaar, 2017; Iqbal et al., 2018). Heterogeneous transfer learning can be categorized into symmetric or asymmetric depending on how the input features are transformed. In symmetric transfer learning, both the source and target features are transformed into a common feature space. Asymmetric transfer learning transforms either the features in the source or target domain to match features in the other domain. For instance, Sukhija (2016) illustrated the used random forest to derive the relationship between features in the target and source domains by leveraging the common labels in the source and target domains.

In homogeneous transfer learning, the input space is the same in both source and target domains $X_s = X_t$, and its either the marginal distribution of the features that differ $D_s \neq D_t$ or the source and target tasks $T_s \neq T_t$. Algorithms that implement homogeneous and heterogeneous transfer learning can be categorized into instance-based, feature-based, or parameter-based transfer learning Figure 1.

2.5.1 Instance-based transfer learning

Instance-based transfer learning is a homogeneous transfer learning method where the source and target tasks are the same but the marginal distributions of the source and target features are different. Instance-based transfer learning is suitable for problems where there is large labelled

datasets from one or more source domains and large unlabelled data in the target domain. Instance-based transfer learning overcomes differences in marginal distributions by careful sampling of observations in the source domain or finding optimal weights to assign to observations in the source domain (Farahani et al., 2021). Each observation in the source domain can be weighted using ratio of probability distribution functions $\beta = P_t(x, y) / P_s(x, y)$, where $P_t(\cdot)$ and $P_s(\cdot)$ are the joint distributions of observations from the target and source domains, respectively. Estimating the probability density functions is challenging and thus β is often approximated during model training by minimizing the discrepancy between the source and target distributions. Discrepancy between the source and target distributions can be quantified using Maximum Mean Discrepancy (MMD) (Huang et al., 2006; Müller, 1997). Chattopadhyay (2012) introduced a technique for instance-based transfer learning from multiple source domains whereby a classifier is first fitted in each source domain and weight calculated for each source domain by computing the discrepancy between the distribution of each source domain and the target domain. All the classifiers are then used to make predictions on unlabelled data in the target domain and pseudo-labels are obtained by computing a weighted average of the predictions. Finally, a classifier is fitted on the target domain using both the labelled data and the unlabelled data (using pseudo labels for unlabelled data).

2.5.2 Parameter-based transfer learning

Parameter-based transfer learning algorithms are homogeneous transfer methods where parameters of models trained on the source domains are incorporated into models trained on the target task. Tommasi (2010) transferred information on support vector machine (SVM) hyper-planes of multiple source domains to the target task. Weights obtained using “leave one out” cross-validation were applied to parameters from each domain such that source domains that were more closely related to the target domain had larger weights. Other approaches use Bayesian methods where the parameters of both the source and target domains have shared prior distribution (Sultan et al., 2016; Xuan et al., 2021). Parameter-based transfer learning methods for deep learning are discussed in details later in this chapter.

2.5.3 Feature-based transfer learning

Feature-based transfer learning algorithms are used for both homogeneous and heterogeneous transfer learning. Feature-based transfer learning aims at transforming the inputs in source and/or target domains such that the extracted features are similar in both target and source domains. A classifier is then fitted using datasets from both source and target domains, with the extracted

features acting as inputs to the machine learning model. Homogeneous feature-based transfer learning can be achieved by identifying features that have the same distribution in both source and target domains and using the identified features as predictors for classifiers fitted using data from both the target and source domains. For instance, Uguroglu (2011) used convex optimization to identify features that produce the least maximum mean discrepancy between the source and target domain.

2.5.4 Transfer learning using deep learning

Parameter-based and feature-based transfer learning methods have been applied widely in deep learning (C. Tan et al., 2018). A neural network is trained to solve the source task and the parameters or predictions of the network can be used for either feature-based or parameter-based transfer learning. In feature-based transfer learning, the activations of hidden layers of the neural network trained on the source task/domain are used as inputs for models performing the target task. The inputs from the target domain are passed through the network trained on the source task and outputs of hidden layers are extracted and used as inputs for the target task. On the other hand, parameter-based transfer learning is carried out by using the parameters on the network trained on the source task to initialize the neural network trained to solve the target task. The architecture of the neural networks in the source and target domains have to be identical for parameter-based transfer learning. In many cases, the source and target tasks are different, but the features necessary for successfully solving the source task are expected to be relevant in solving the target task.

The source tasks can be classified into supervised or unsupervised/self-supervised depending on whether the dataset from the source domain is labelled (C. Tan et al., 2018). Initializing the weights of neural networks using parameters from a neural network trained on a labelled source domain is referred to as supervised pre-training. For instance, convolutional neural networks that are trained to classify Imagenet dataset have been used to initialize neural networks for various image classification tasks including classification of CXR, satellite images, and micro-structures in material science (Rajpurkar et al., 2017; D. Wang et al., 2022; White et al., 2022). On the other hand, self-supervised pre-training involves using unlabelled data to train the neural network for the source domain. The source task is often referred to as the pretext task, while the target task is referred to as the downstream task. The pretext tasks can be classified into generative or discriminative tasks.

In generative self-supervised models such as auto-encoders, a deep learning model is trained to predict the inputs by learning the identity function $f(X)=\hat{X}$. Auto-encoders comprise of two neural networks, where one neural network (the encoder) transforms the inputs into latent representation $g(X)=z$, and the second neural network (the decoder) transforms the latent representation back to the input space $h(z)=\hat{X}$. Therefore $f(X)=h(g(X))$. Regularization is used to ensure that the function f learns useful representation/features from the inputs. Denoising auto-encoders apply regularization by adding noise to the inputs and having the model reconstruct the inputs without noise (Vincent et al., 2010). Sparse auto-encoders apply regularization by forcing the latent representation to be sparse (Le et al., 2012). Other generative models such as generative adversarial networks (GANs) and variational auto-encoders (VAEs) are trained to sample from the distribution of the inputs (I. J. Goodfellow et al., 2014; Kingma & Welling, 2013).

Generative models such as auto-encoders are difficult to train when the input space is large and may not be necessary in learning useful representations of the inputs. Discriminative self-supervised algorithm avoid such difficulties by deriving pretext labels from unlabelled data and training a supervised learning model. For instance, each image in an unlabelled dataset can be split into tiles and a model trained to solve a jigsaw puzzle (Noroozi & Favaro, 2017). Labels can also be derived from images by treating each image in the dataset as a separate class. Multiple instances of each class/image can be derived by applying data augmentation. The common data augmentation techniques for images include rotating/flipping the image, random cropping, random brightness and contrast adjustment, and changing the aspect ratio (Shorten & Khoshgoftaar, 2019). Training a classification model where each image is a different class is not feasible if the number of images is large. Therefore, a contrastive learning models can be trained to map the latent representation of each input such that latent representations of two related inputs are closer compared to representations of unrelated inputs (Saunshi et al., 2022).

A major drawback of feature-based transfer learning is that aspects of the inputs that are relevant to the downstream task may be lost when training on the source task. Multi-task learning can overcome loss of information that is relevant to downstream tasks by training both the source and target tasks simultaneously (Crawshaw, 2020; Y. Zhang & Yang, 2017). Multi-task learning is achieved by having neural networks for solving multiple tasks share parameters/weights. There are

two major branches of multi-task learning for deep learning models: hard and soft parameter sharing (Caruana, 1993; Duong et al., 2015). Hard parameter sharing enforces weights for some of the hidden layers of neural networks for different tasks to be equal. The layers close to the inputs are shared while those close to the output are not. While it is difficult to over-fit with hard parameter sharing, the performance of the models for the various tasks can be dismal if all tasks are not closely related (Baxter, 1997). In soft parameter sharing, each task has a separate neural network. The shared hidden layers have identical architecture and corresponding layers are initialized to with the same values. A regularization term quantifying the differences in weights of shared layers is added to the cost function during model training (Ruder, 2017). Common regularization terms that have been used in literature include the L2 distance and the trace norm between weights in corresponding shared layers (Duong et al., 2015; Y. Yang & Hospedales, 2017).

The cost function of deep multi-task learning models is derived by summing the losses of all tasks. However, the training process can favor tasks whose losses have large magnitudes if the losses of different tasks have different magnitudes, thus exhibiting poor performance (Vandenhende et al., 2021). Chen (2018) proposed using gradient descent to learn optimal weights for different tasks, and Lin (2021) showed that assigning random weights to losses of different tasks at each training iteration can also perform well.

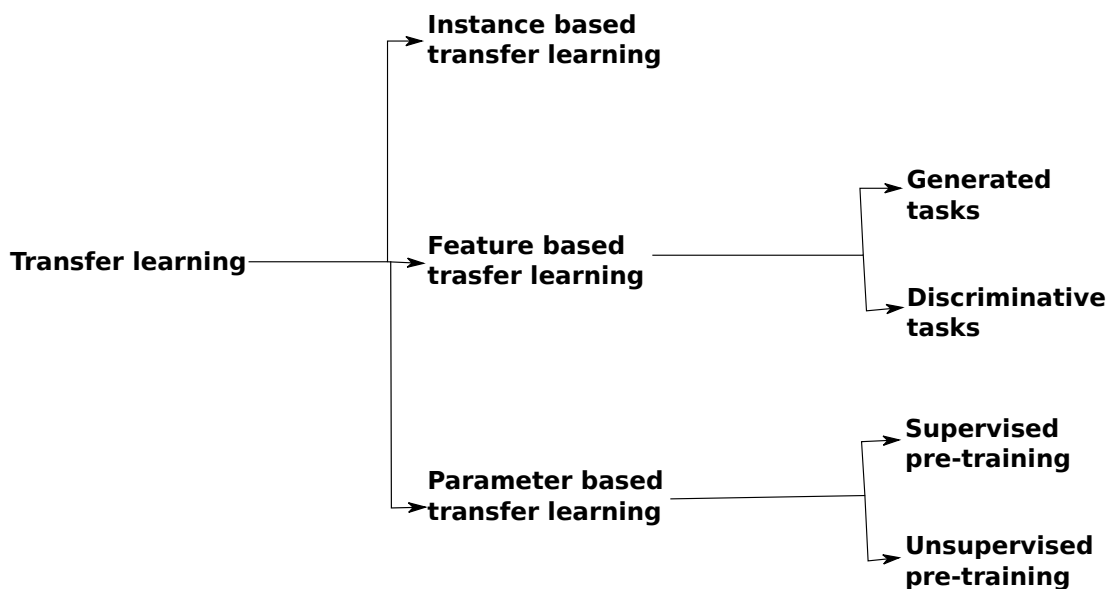


Figure 1: Transfer learning methods. Transfer learning can be categorized into three: instance based, feature based or parameter based transfer learning. Feature based transfer learning can further be divided into two depending on whether the source task learns the distribution of inputs (generative tasks) or solves a discriminative task using pseudo-labels generated from the inputs.

2.6 Transfer learning for medical images, bio-signals and clinical data

Parameter-based transfer learning using supervised pre-training has been applied to CXR images, whereby parameters of CNNs trained on the ImageNet dataset are used to initialize CNNs for classifying CXR images (Avola et al., 2021; Ponomaryov et al., 2021; Rajpurkar et al., 2017). Unsupervised pre-training, where weights of models trained using self-supervised learning have also been used to initialize deep learning models for classifying CXRs. For instance, Gazda (2021) used self-supervised model trained used contrastive learning to initialize a convolutional neural network for classifying CXR images. Self-supervised using contrastive learning has also been applied to computed tomography (CT) scans, magnetic resonance (MRI) imaging and ultrasonography (Ghesu et al., 2022).

Transfer learning has also been used in analysis of bio-signals. Weimann and Conrad (2021) used pre-trained convolutional neural networks for classification of heart arrhythmia using raw ECG signals. Others have applied transfer learning on bio-signals by first transforming the raw signal into an image/spectrogram using Fourier or wavelet transform and then use transfer learning

techniques commonly used for computer vision (Khan et al., 2021; Salem et al., 2018; Venton et al., 2020; Wu et al., 2018).

Transfer learning has also been applied to tabular medical data (Ebbehoj et al., 2022). Waldi (2021) used data from an electronic health records (EHR) database from one hospital to pre-train a neural network for classification of sepsis at a different hospital, outperforming a similar model without transfer learning in external validation. Supervised and unsupervised pre-training have been used to improve classification in subgroup of patients from minority ethnic groups that were under-represented in the training data (Gao & Cui, 2020).

In conclusion, while transfer learning has been applied successfully in natural images, similar applications in medical images are scarce. Pre-training machine learning models for medical images using natural images may be sub-optimal. Successful classification of natural images might depend on model's ability to detect edges, while classification of medical images might rely of changes in color texture or other latent features. Transfer learning from source domains that are not closely related to target domain might cause negative transfer, where models trained with transfer learning perform worse on the target domain compared to model trained without transfer learning (W. Zhang et al., 2021). In addition, augmentation techniques – which are at the heart of transfer learning algorithms such as contrastive learning - used for natural images might be inappropriate for medical images. For instance, image cropping could leave out parts of medical images that are relevant to making diagnosis, given that some medical conditions may occupy only a small region of the medical image. There is limited literature on use of transfer learning on tabular clinical data. In addition, diagnostic and prognostic models developed using of tabular have relied on logistic regression. Logistic regression models are linear models (that describe the log odds of the outcome as a linear combination of one or more explanatory variables) which are not expected to perform well if the dataset is not linearly separable. While a systematic review of the benefit of machine learning over logistic regression showed that machine learning models don't perform better, the reported models did not employ transfer learning and the effect of dataset size on model performance was not evaluated (Christodoulou et al., 2019). Analysis of bio-signals have relied on traditional signal processing techniques to extract features from raw signals. Such techniques require domain knowledge of the relationship between the raw signal and the outcome of interest, limiting the use of PPG signals in predicting novel outcomes. Deep learning has been shown to be

useful in classification of various signals without requiring extensive feature engineering. While end-to-end deep learning models require large labelled datasets, self-supervised learning (SSL) may improve the performance of end-to-end deep learning models by providing end-to-end models with optimal weight initialization. Classification of PPG signals may benefit from self-supervised learning methods such as contrastive learning. But it is also not clear which contrastive pre-text tasks would be relevant for bio-signals.

3 Methods

3.1 Analysis of Bio-signals

The aim of this analysis was to predict hospitalization of children seeking care at the outpatient department of a public hospital using bio-signals (PPG) obtained using a pulse oximeter. We sought to find out whether transfer learning incorporating an unlabelled PPG signals dataset could improve the performance of classification models fitted using a small labelled PPG dataset. We used Self-Supervised Learning (SSL) models trained using contrastive learning to perform transfer learning. We compared prediction performance of features extracted using SSL model fitted with labelled PPG signals only with features extracted using SSL model fitted using both labelled and unlabelled PPG signals. Furthermore, we compared performance of end-to-end deep learning models initialized randomly, with models initialized with weights from the SSL models. An overview of steps used to analyze the PPG signals are outlined in Figure 2.

3.1.1 Data sources

Labelled PPG signals

Labelled data was collected by a study nurse at the paediatric outpatient department of Mbagathi sub-county hospital between January and June of 2018. The study nurse used a pulse oximeter connected to an android tablet to collect PPG signals from recruited patients after obtaining informed consent from the caregivers. The study nurse also collected demographic information and clinical signs and symptoms of all recruited children. The facility clinician, who was not part of the study and without access to the study data made decision on whether children were hospitalized. The study nurse collected PPG signals from 1,031 children, 125(12.2%) of whom were hospitalized. We set aside 20% of the PPG signals for final model validation and used the rest for model training and validation.

Unlabelled PPG signals

We sourced additional PPG signals from the Smart Discharge (SD) study in Uganda (Wiens et al., 2021a, 2021b). The study nurses collected two PPG recordings from each patient at both admission and discharge. The models are meant to be used for triage and therefore require the training data to be collected at triage or as soon as a patient arrives at the hospital. Therefore, the data from SD

study was treated as unlabelled despite being collected from hospitalized children because the status of the child may have changed between the time they arrived at the hospital (triage) and the point of admission/discharge.

PPG signal acquisition

PPG signals were sampled at a rate of 80 Hertz (80 samples per second) for 60 seconds for both Mbagathi and SD study. The pulse oximeters were developed by Lionsgate Technologies medical (Vancouver, Canada) and utilized the red and infrared wavelengths.

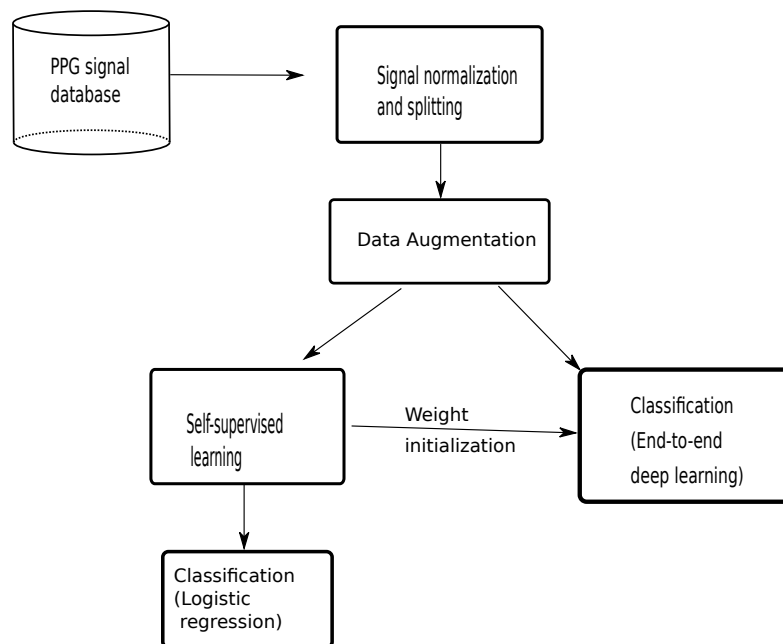


Figure 2: Schematic overview of the analysis of PPG signals. PPG signals were normalized using mean and standard deviation of the entire dataset and the signals from each patient were split into segments of 10 seconds with 2 second sliding window. Data augmentation was applied to PPG signals in the training dataset only before signals were analyzed using supervised and self-supervised deep learning models. Features extracted using SSL models were classified using logistic regression. Weights of SSL models were also used to initialize end-to-end deep learning models

3.1.2 Signal pre-processing

The red and infra-red PPG signals were normalized by subtracting the mean and dividing by the standard deviation computed from all PPG segments in Mbagathi study. The signals were then split into 10 second segments with two second sliding window (two second overlap between consecutive segments).

3.1.3 Data Augmentation

We applied data augmentation to normalized PPG signals in the training dataset to reduce overfitting as part of model training pipeline. Data augmentation is a technique of generating more data by applying random perturbations to existing data. We applied data augmentation to PPG signals by adding Gaussian noise and applying signal slicing and permutation (Um et al., 2017). Gaussian noise augmentation was applied by element-wise addition of noise sampled from the Gaussian distribution to the PPG signals. For each of the red and infrared channels of the PPG signal, random values numbering the length of samples in a PPG segment were drawn from a normal distribution with a mean of zero and standard deviation sampled from the log uniform distribution with a minimum and maximum values of 0.00001 and 0.01, respectively. Signal slicing and permutation augmentation was applied by splitting the PPG segment into n slices of equal length and concatenating the slices after permuting their order. Hyper-parameters for proportion of training PPG signals with augmentation and the number of slices were optimized during model training. An example of PPG segment data augmentation is shown in Figure 3.

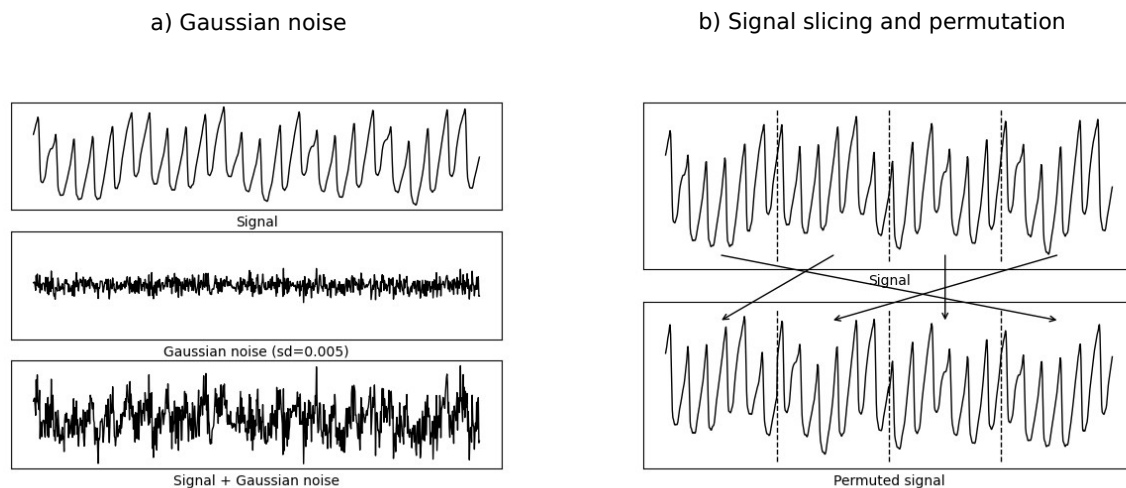


Figure 3: PPG data augmentation. a) Noise sampled from the Gaussian distribution is added to the original signal. b) Signal slicing and permutation was implemented by splitting the signals into multiple segments and combining the segments in random order.

3.1.4 Models

We fitted models to predict whether a child would be hospitalized given PPG signals collected during triage. We used deep learning models to either classify hospitalization given the raw PPG signals (end-to-end models) or to extract features from raw signals that could then be classified using logistic regression (self-supervised feature extraction). We used convolutional neural networks (CNNs) for all deep learning models. CNN is an architecture of artificial neural networks (ANNs) that widely used for datasets with a spatial or temporal structure (LeCun et al., 1999).

ResNet50 deep learning architecture was used for both end-to-end models and encoders with slight modification necessitated by differences between PPG signals and images. In particular, we used two channels in the first convolutional layer instead of three because PPG signals have two channels (red and infrared) as opposed to the red, green and blue channels in color images (He et al., 2015). In addition, all convolutional layers were modified to be one dimensional because the PPG signals are one dimensional (time) as opposed to the height and width dimensions of an image. Max pooling was applied after every convolutional block to reduce computation cost and increase the receptive field of the model.

SSL models based on contrastive learning were used to perform feature-based and parameter-based transfer learning. Feature-based transfer learning was implemented by using the trained SSL models to transform PPG signals into feature vectors that could be classified using logistic regression to predict hospitalization. On the other hand, parameter-base transfer learning was implemented by using the parameters of the SSL models to initialize the parameters of end-to-end deep learning models for predicting hospitalization.

3.1.5 Feature extraction using SSL

Encoder architecture and training

The encoder model consisted of two functions. The first function $f: \mathbb{R}^{2 \times 800} \rightarrow \mathbb{R}^{32}$, was a ResNet model that transformed a PPG segment into a latent representation h_i . The second function $g: \mathbb{R}^{32} \rightarrow \mathbb{R}^{32}$, was a multi-layer perceptron (MLP) with a single hidden layer that transformed the output of the first function into a compressed representation k_i . The encoder was trained such that given two PPG segments X_i and X_j , the distance between compressed representations k_i and k_j was closer if the PPG segments were sampled from the same patient compared to PPG segments

sampled from different patients (Figure 4). We trained the model using noise contrastive estimation (NCE) loss shown in equation (1) and used dot product to characterize the distance between two compressed representation (Gutmann & Hyvärinen, 2010).

$$l_j = -\log \frac{\exp(q \cdot k_+ / \tau)}{\sum_{i=0}^{2(N-1)} \exp(q \cdot k_i / \tau)} \quad (1)$$

where N is the mini-batch size, q is a query embedding, k_+ is the embedding from the positive pair, k_i are the embeddings of $2(N-1)$ negative and one positive segments, and τ is a temperature hyper-parameter.

We trained two SSL models: one was trained on labelled PPG signals only and the other on all PPG signals (both labelled and unlabelled)

Model training and Hyper-parameter optimization

Hyper-parameter optimization was done using Asynchronous Successive Halving Algorithm (ASHA) using the **ray-tune** library in Python (L. Li et al., 2020; Liaw et al., 2018). ASHA was implemented by randomly sampling 300 hyper-parameter configurations and stopping poorly performing configurations after 20, 40, 80, 160, 320, and 640 epochs. A list of all hyper-parameters and the search space is shown on Table 1. The models were trained for a maximum of 700 epochs using Adam optimizer (Kingma & Ba, 2017).

Table 1: Hyper parameter search space

Hyper-parameter	Search space
Dropout	$\sim \text{loguniform}(0.01, 0.5)$
Batch size	{8,16,32,64}
NCE temperature	$\sim \text{loguniform}(0.0001, 0.5)$
Learning rate of convolutional layers	$\sim \text{loguniform}(0.00001, 0.5)$
Weight decay parameter for convolutional layers	$\sim \text{loguniform}(0.000001, 1.0)$
Learning rate for fully connected layers	$\sim \text{loguniform}(0.00001, 0.5)$
Weight decay for fully connected layers	$\sim \text{loguniform}(0.000001, 1.0)$
Proportion of signals with Gaussian noise augmentation	{0.0,0.2,0.5,0.8,1.0}
Number of slices for with signal slicing and permutation augmentation	{2,5,8,10}
Proportion of signals with signal slicing and permutation augmentation	{0.0,0.2,0.5,0.8,1.0}

Feature extraction

The hidden layer and the output layers of the MLP had 32 units each. We discarded the last layer of the MLP after training the encoder because previous studies have shown that features extracted using the last layer perform poorly on downstream tasks because they are heavily tuned to solving the pretext task (T. Chen et al., 2020; He et al., 2020). Given a 10 second PPG segment, the model extracted 32 features, which were later used as predictors of hospitalization.

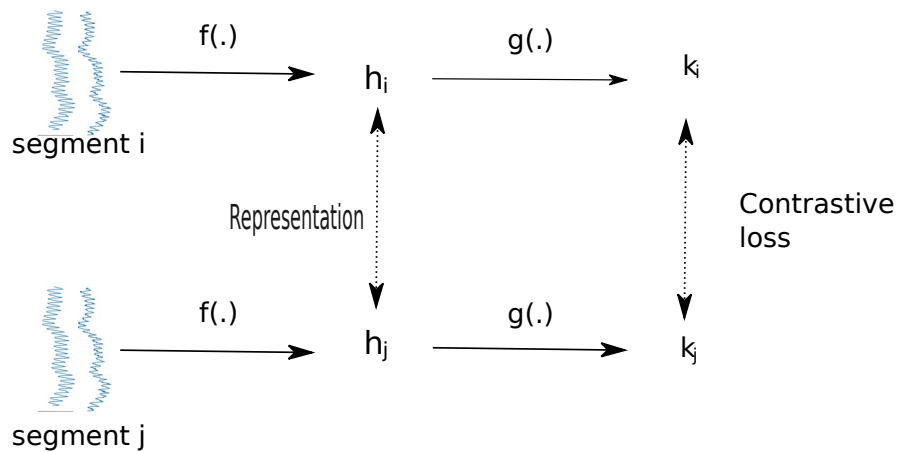


Figure 4: Contrastive learning. The function $f(\cdot)$ is a ResNet model that transforms raw PPG signals into an intermediate representation h_i , while the $g(\cdot)$ is a MLP that transforms the intermediate representation to the final representation k_i .

For comparison purposes, we also extracted features using principal component analysis (PCA), a traditional dimensionality reduction technique. PCA was used to reduce the dimensionality of raw PPG signals from 2×800 to 32. We qualitatively assessed the quality of extracted features using a scatter plot of the first and second PCA components and coloured the points using values of heart rate, respiratory rate and SpO₂.

3.1.6 Classification and regression using Linear models

We tested correlation of extracted features with physiological parameters known to be present in PPG signals (heart rate, respiratory rate, and SpO₂) using linear regression, and hospitalization using logistic regression. These physiological parameters were not of clinical interest but served as a measure of quality of features extracted using SSL.

We used Bayesian linear and logistic regression fitted using the Pyro library in python (Bingham et al., 2018). Horseshoe priors were used to induce sparsity in the coefficients of the linear/logistic regression models to prevent over-fitting, given that the dimensionality of the extracted features was large when second order polynomials (including all pairwise interactions) of the predictors were included (Carvalho et al., 2009). Hamiltonian Monte-Carlo algorithm was used to obtain samples of regression parameters from the posterior distribution (Neal, 2011). The algorithm was run to produce 5000 samples from the posterior distribution with a burn-in of 1000 samples. The 5000 samples were used to generate 5000 predictions for each observation in the test data-set, and the final prediction for each observation was obtained using unweighted mean.

We also evaluated the utility of the extracted features in improving logistic regression models for predicting hospitalization using clinical features identified from previous analyses (Mawji et al., 2021). Features extracted using SSL models were concatenated with eight clinical features: weight, mid-upper arm circumference (MUAC), restlessness, inability to drink/breastfeed, temperature, heart rate, SpO2 and difficulty breathing.

3.1.7 End to end deep learning

The model architecture used for the end-to-end deep learning model was similar to the encoder in section 3.1.5. The ResNet function $f(\cdot)$ was identical but the MLP function $g(\cdot)$ had one output at the last layer instead of 32. The model was trained to predict hospitalization using Adam optimizer and binary cross-entropy loss. The binary cross entropy loss between the target y_i and predicted probability of admission \hat{y}_i can be describe as:

$$l_i = -y_i \cdot \log \hat{y}_i + (1 - y_i) \cdot \log(1 - \hat{y}_i)$$

and the loss of a mini batch of n observations was obtained by computing the mean:

$$L = 1/n \cdot \sum_{i=1}^n l_i$$

During training, observations from the minority class (hospitalized) were up-sampled to address class imbalance. That is, the probability of an observation x_i with a given outcome being included in a mini-batch was equal to the reciprocal of number of observations with that outcome. Label smoothing was applied to reduce the negative effect of noise in the outcome. Given an outcome label y_i and a smoothing parameter α , the smoothed label was computed as:

$$y_{smooth} = (1 - \alpha) * y_i + \alpha / K$$

where K is the number of classes (2 for binary classification).

We compared three methods of initializing weights in the $f(\cdot)$ function of the end-to-end deep learning model: Random initialization, initializing using weights of the SSL model trained on labelled PPG signals only, and initialization using weights of SSL model trained using all PPG signals (both labelled and unlabelled). We used the same hyper-parameter optimization procedure as the SSL models in section 3.1.5.

3.2 Analysis of medical images (Chest radiographs)

We used transfer learning to improve classification accuracy of machine learning models for classifying paediatric CXR images from the PERCH study. We used parameter based transfer learning where models for classifying PERCH CXR images were initialized using weights from models trained to perform other computer vision tasks. We compared transfer learning using model trained on natural images (ImageNet dataset) with model trained on CXR images (Chestray -14 CXR images). We also explored transfer learning using multi-task learning where models for classifying PERCH and Chestray-14 CXRs were trained simultaneously. Finally, we explore a novel multi-task learning approach where a model is trained to simultaneously classify how multiple human readers would classify a given CXR. Such models can be trained on data-sets where multiple readers annotated the training data. The predictions of all readers for a given CXR are aggregated to obtain the final prediction. The performance of models classifying PERCH CXRs were evaluated using multi-label accuracy and AUC (one verses the rest).

3.2.1 Data

The PERCH dataset contains CXR images collected in nine sites from seven low and middle income countries: Kilifi, Kenya; Basse, The Gambia; Nakhon Phanom and Sa Kaeo, Thailand; Bamako, Mali; Soweto, South Africa; Lusaka, Zambia; and Dhaka and Matlab, Bangladesh. The CXRs were obtained from children aged 2-59 months hospitalized with severe or very severe pneumonia. PERCH CXR images are classified into five categories according to WHO classification of CXR for diagnosis on pneumonia: consolidation; other infiltrate; both consolidation and other infiltrate; normal or uninterpretable (Cherian et al., 2005). Normal CXR category accounted for almost half of images in all sites except for South Africa and Zambia (28% and 31%, respectively). The proportion of images in each site that were classified as uninterpretable ranged between 4% and 20% (Figure 5).

There were 18 readers (annotators), 14 initial readers (nine paediatricians and five radiologists) and four arbitrators (all radiologists). The initial readers consisted of two readers from each country who received training on the WHO methodology from the arbitrators. Whenever the two initial readers gave conflicting interpretations, two arbitrator with extensive WHO methodology experience were randomly chosen to review the image. If the two arbitrators still came to conflicting interpretations, the two arbitrators held a consensus discussion to make a final decision. Finally, the arbitrators reviewed 10% of images with initial concordance for quality control (Fancourt et al., 2017a).

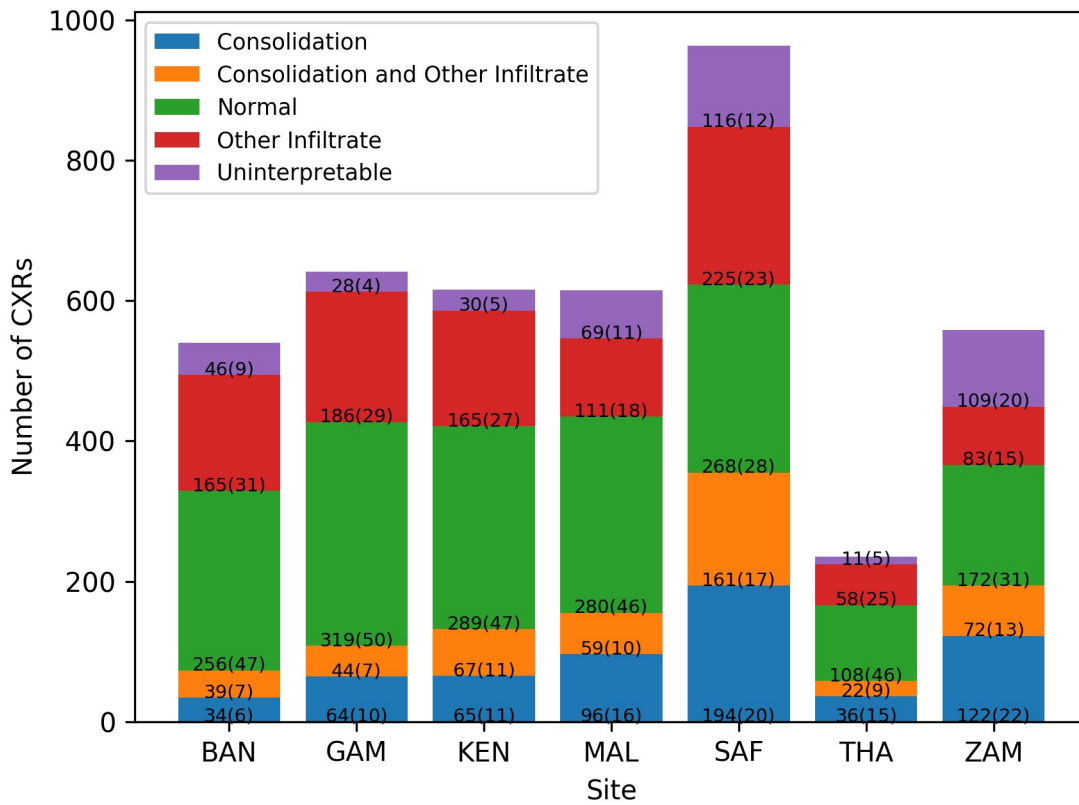


Figure 5: Number (percentage) of chest radiographs (CXRs) from each country by WHO category (label). Images were from Bangladesh (BAN), South Africa (SAF), Mali (MAL), Zambia (ZAM), Kenya (KEN), Thailand (THA), and Gambia (GAM).

The Chestray-14 data-set comprises of 112,120 CXRs obtained from 30,805 patients. The data-set was obtained from the Picture Archiving and Communication Systems (PACS) database of the National Institutes of Health (NIH). The data-set is classified into 14 common chest pathologies one of which is pneumonia. The 14 classes were obtained from radiological reports using machine learning methods for natural language processing. All patients in the data-set originate from one high income country and less than 1% of CXRs are obtained from children below five years.

CXRs from PERCH and Chestray-14 have significant differences. These include age of patients, ailment and technology used to acquire images. PERCH images from Zambia and the Matlab site in Bangladesh were obtained using analog means and the films scanned into digital format (Ominde et

al., 2018). Some CXRs in PERCH data-set do not occupy the entire image, while other include body parts beside the chest such skull, limbs, and stomach (Figure 6).

Both PERCH and Chestray-14 data-set were split once to obtain the training and test data-set as opposed to K-fold cross-validation. The PERCH dataset was split such that 20% (802/4008) of patients were set aside for final model evaluation (test set), while the official train-test split was used for Chestray-14 dataset (X. Wang et al., 2017). The training data-set was further split into training and validation datasets as part of model training for optimal hyper-parameter search. The final performance of all models was evaluated on the same test dataset so that comparison of different modeling approaches was not affected by differences in test images.

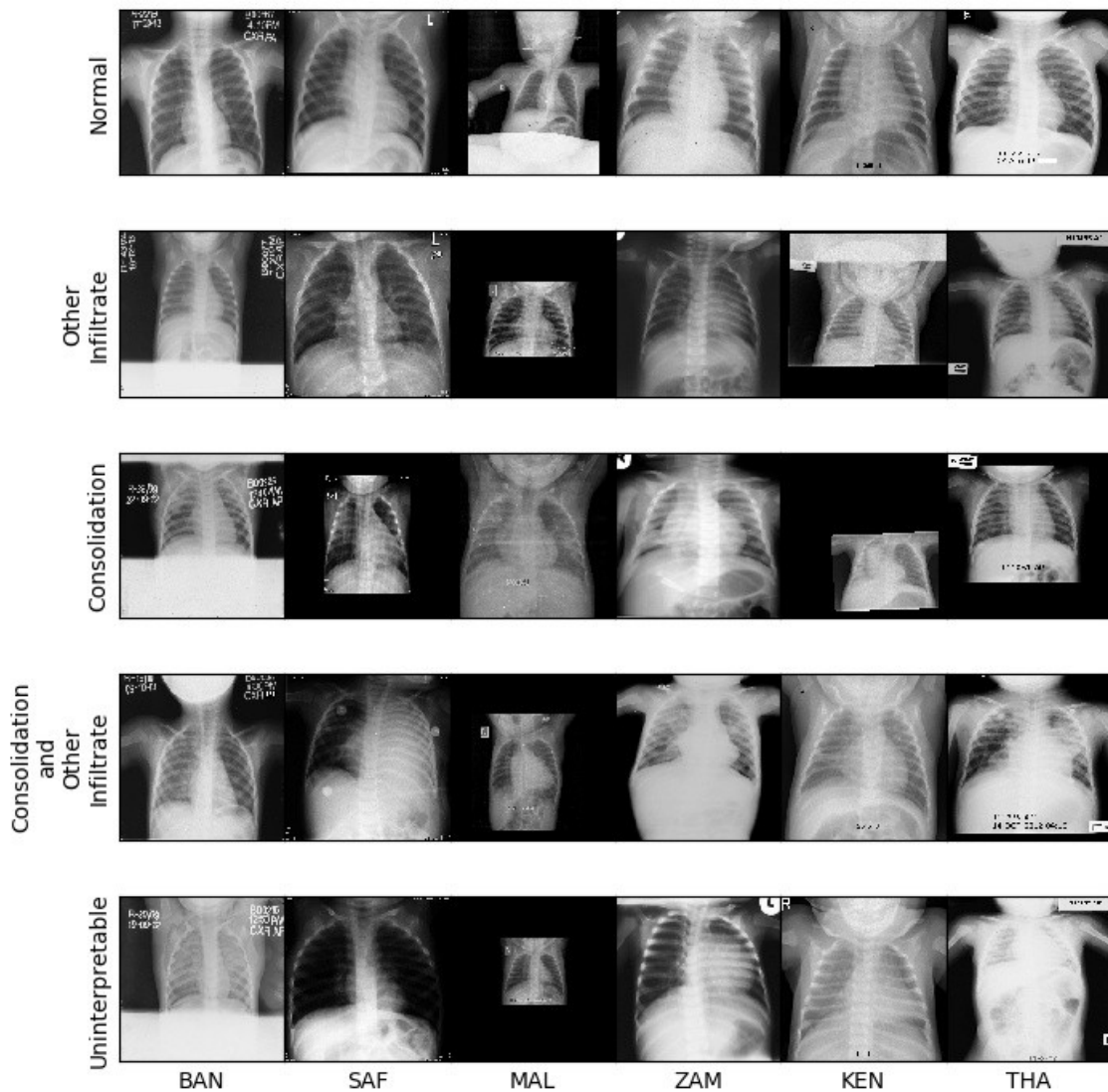


Figure 6: A random sample of PERCH CXR images. Some images do not occupy the entire film and images of very small children contain other body parts besides the chest.

3.2.2 Baseline models

For simplicity, we used ResNet18, ResNet34 and ResNet50 model architectures from the torchvision version 0.8.2 library for all our experiments (He et al., 2015; Marcel & Rodriguez, 2010; Paszke et al., 2019). The ResNet models' last fully connected layer was replaced with a fully

connected layer with five output units – one for each WHO category. The ResNet models in torchvision library are either initialized randomly or using weights from models trained on the ImageNet dataset (J. Deng et al., 2009). We used models initialized using ImageNet weights as a baseline because they have been shown to perform better on CXR datasets than randomly initialized models (Rajpurkar et al., 2017).

We used ASHA algorithm (described in details in section 3.1.5) to select optimal hyper-parameters for dropout, batch size, learning rate, weight decay and proportion of images with augmentation. The models were trained for 150 epochs, with learning rate halving after 50 and 100 epochs. We trained the models using Adam optimizer and cross entropy loss. For a classification problem with C classes, the cross entropy loss of an observation i with correct class k is defined as:

$$l_i = -\log\left(\frac{\exp(x[k])}{\sum_{j=1}^C \exp(x[j])}\right)$$

where x are the unnormalized scores for each class. The loss for a mini-batch of n observations was obtained using unweighted mean:

$$L = \sum_{i=1}^n l_i$$

Weighting the loss function by class frequencies to address class imbalance did not improve model performance and was not implemented in all subsequent analyses.

3.2.3 Pre-trained models

We initialized the weights of models for classifying PERCH CXRs using weights of deep learning models with the same architecture but trained on different tasks. We tested initializations using weights from supervised model trained to classify Chestray-14 CXRs (supervised pre-training) and unsupervised/self-supervised model trained using both PERCH and Chestray-14 CXRs (unsupervised pre-training).

Supervised pre-training

We trained models for classify Chestray-14 CXRs using ResNet models with varying sizes (ResNet18, ResNet34 and ResNet50). The models were identical to the baseline model for

classifying PERCH images except for the final fully connected layer, which classified each image into 15 binary classes. Given that Chestray-14 dataset contains 15 binary labels, the loss function was a weighted sum of 15 binary cross entropy losses. We weighted the losses of each of the 15 binary classification tasks by the homoscedastic uncertainty of each loss (Kendall et al., 2018). The aggregated loss function for the 15 tasks was defined as:

$$l_i = \sum_{j=1}^{15} \frac{1}{\sigma_j^2} l_{ij} + \log(\sigma_j^2)$$

where σ_j^2 were the weights of the 15 losses and were parameters optimized during model training.

ASHA algorithm was used for hyper-parameter optimization and Adam optimizer was used to train the models.

After training the model to classify Chestray-14 images, the weights of the best model was used to initialize models for classifying PERCH images. The training procedure for PERCH image was identical to the baseline model described in section 3.2.2.

Self-supervised pre-training

We trained a SSL models using contrastive learning which is described in details in section 3.1.5. We used instance classification pretext task, where the encoder was trained such that embeddings of two views of the same CXR were closer than embeddings of two views of different CXRs (Figure 7a). Multiple views of a single CXR were created by apply random augmentations to the CXR. The augmentation pipeline was composed of random resized crop, random color jitters (random contrast and brightness adjustments), random horizontal and vertical flips, and random affine transformation (Figure 7b).

We used Population Based Training (PBT) to find optimal hyper-parameters for augmentation and model training (Jaderberg et al., 2017). PBT was carried out as follows: Eight models were initialized with eight random hyper-parameter configurations. The models were then trained in parallel for five epochs after which four of the poorest performing models were discarded and replaced with clones of the four best performing models and hyper-parameters. The hyper-parameters were then perturbed and the models trained for another five epochs. The process was repeated until none of the model improved after fifty epochs. The best performing models was then taken as the final model.

The weights of the best model were used to initialize CNN models for classifying PERCH images. The model was then trained in a manner similar to the baseline model.

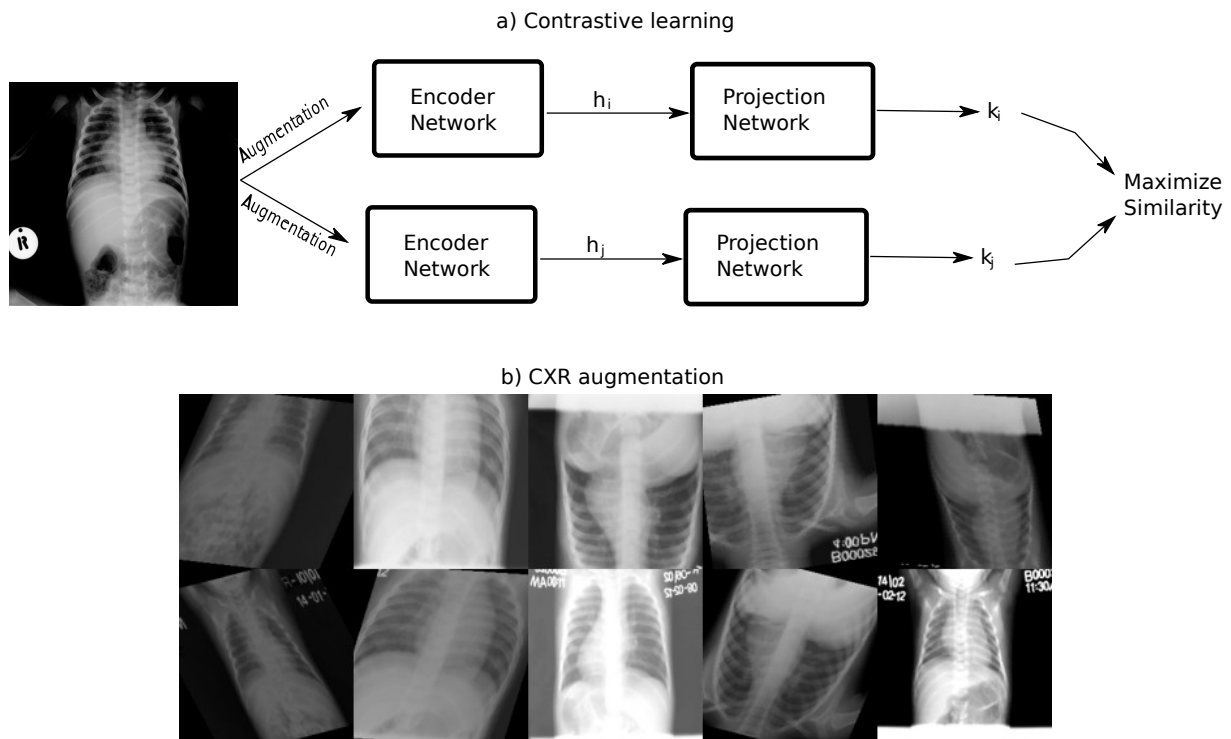


Figure 7: SSL using contrastive learning. a) The contrastive learning model was trained such that embeddings of different view of the same image were closer in distance compared to embeddings of views of two different images. b) Example of augmentation of 5 CXR images. Each column shows two views of the same CXR obtained by applying multiple augmentations.

3.2.4 Multi-task learning

Multi-task learning involve training models that can perform two or more tasks simultaneously. Multi-task models have multiple outcomes with each outcome corresponding to a task. Two ResNet models were trained simultaneously to classify PERCH and Chestray CXRs. The two models had the same architecture for all layers except the final fully connected layer, which had five and fifteen output units for PERCH and Chestray models, respectively. Constraints were placed on the weights

of all layers except the final fully connected layer during training such that corresponding weights of the PERCH and Chestray models would be equal or have small mean absolute differences (Figure 8).

Two multi-task learning approaches were compared. The first multi-task approach forced the weights of the shared layers to be equal but varied the number of shared layers (hard parameter sharing). The extent of weight sharing (Resnet block1, ResNet block2) was a hyper-parameter to be optimized during model training. The model for classifying Chestray-14 CXR images had 15 binary outcomes and the loss was computed as the mean of 15 binary cross-entropy losses. On the other hand, the outputs of the model for classifying PERCH images were the probabilities of a CXR belonging to each of the 5 WHO categories and the loss was the cross entropy loss. The losses for models classifying PERCH and Chestray-14 images were combined using a weighted sum where the weights were parameters to be estimated during model training. That is:

$$L = \frac{L_{PERCH}}{\sigma_1^2} + \frac{L_{Chestray-14}}{\sigma_2^2} + \log(\sigma_1^2) + \log(\sigma_2^2)$$

where σ_1^2 and σ_2^2 are the weights of PERCH and Chestray-14 models losses respectively and the term $\log(\sigma_1^2) + \log(\sigma_2^2)$ prevented the values of weights from becoming arbitrary large during model training, given that the optimizer can take a shortcut and reduce the overall loss by making the values of σ_1^2 and σ_2^2 arbitrarily large.

The second multi-task learning approach constrained the weight of the shared layers by adding a penalty to the loss function during model training (soft parameter sharing). The penalty was the mean absolute difference (MAD) between weights in corresponding layers of the two networks. The loss function including the MAD term was:

$$L = \frac{L_{PERCH}}{\sigma_1^2} + \frac{L_{Chestray-14}}{\sigma_2^2} + \log(\sigma_1^2) + \log(\sigma_2^2) + \frac{\lambda}{P} \sum_{i=1}^P |W_{1i} - W_{2i}|$$

where λ is a hyper-parameter controlling the strength of regularization, W_1 and W_2 are the weights of the models for classifying PERCH and Chestray-14 CXR images respectively, and P is the total number of weights shared by PERCH and Chestray models.

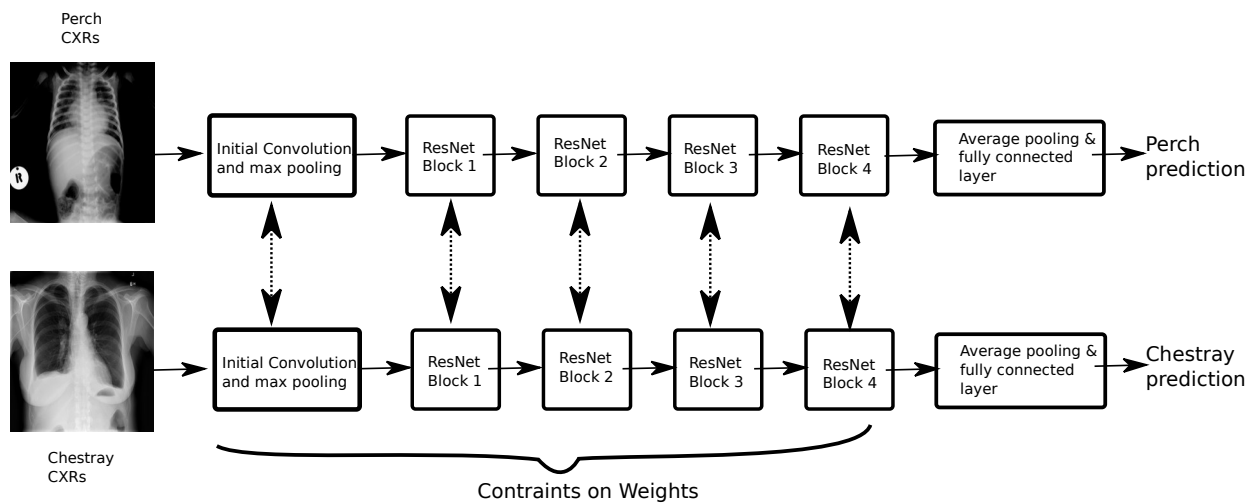


Figure 8: Multi-task learning for classifying PERCH and Chestray-14 images. Constraints were placed on the weights during training such that the weights would either be the same or similar

3.2.5 Incorporating individual reader annotations

We extended the PERCH model by adding an embedding layer for reader identifiers. An embedding layer maps a categorical variable into a vector of fixed size (Guo & Berkahn, 2016). Once trained, the vector/embeddings should contain semantic information about the category it embeds. That is, categories that are similar should produce embeddings that are closer as measured using distance metrics such as cosine similarity compared to embeddings of categories that are unrelated. Embeddings are widely used in language models where each word is embedded into a vector of fixed size (Mikolov et al., 2013). In our case, readers who classify CXR in the same way should have embeddings that are closer than embeddings of readers who don't. The addition of a reader embedding layer meant that the modified PERCH model could classify a given CXR image conditional on reader identifier.

The ResNet models have a global average pooling (GAP) after the final convolutional layer. The outputs of the GAP are mapped to model predictions using a single fully connected layer. Therefore, the neural network layers from the input layer to the GAP can be considered as a feature extractor whose outputs act as inputs to a linear classifier (the fully connected layer). We extended the ResNet models by combining reader embeddings with image embeddings using element wise multiplication. Given that the reader embeddings had dimensions of 32 units, a fully connected

layer was used to project reader embeddings to have the same dimensions as the image embeddings. We applied hyperbolic tangent (tanh), sigmoid or identity activation function to the projected reader embeddings, with the choice of activation function was tuned as a hyper-parameter during model training. A fully connected layer with 5 units was appended to the network to classify the CXRs into one of the five WHO categories (Figure 9).

The resulting model was equivalent to the PERCH model without reader embeddings if all the values of reader embedding have value one. If we consider image embeddings as features extracted from a given image, then the learned reader embedding allowed different readers to assign different weights to each of the extracted image features. We used the same model training procedures as supervised pre-training (we used the same image augmentation techniques, learning rate schedule, regularization and optimization algorithms).

While the training CXR images were the same for models with and without reader embeddings, each CXR appeared multiple times in each training epoch depending on how many readers classified it. We used each reader's classification as labels during training, unlike in models without reader embeddings where the final classification was used. There were 18 readers in total. Thus, 18 predictions could be made for every CXR image. The 18 predictions were aggregated to obtain the final prediction using unweighted mean.

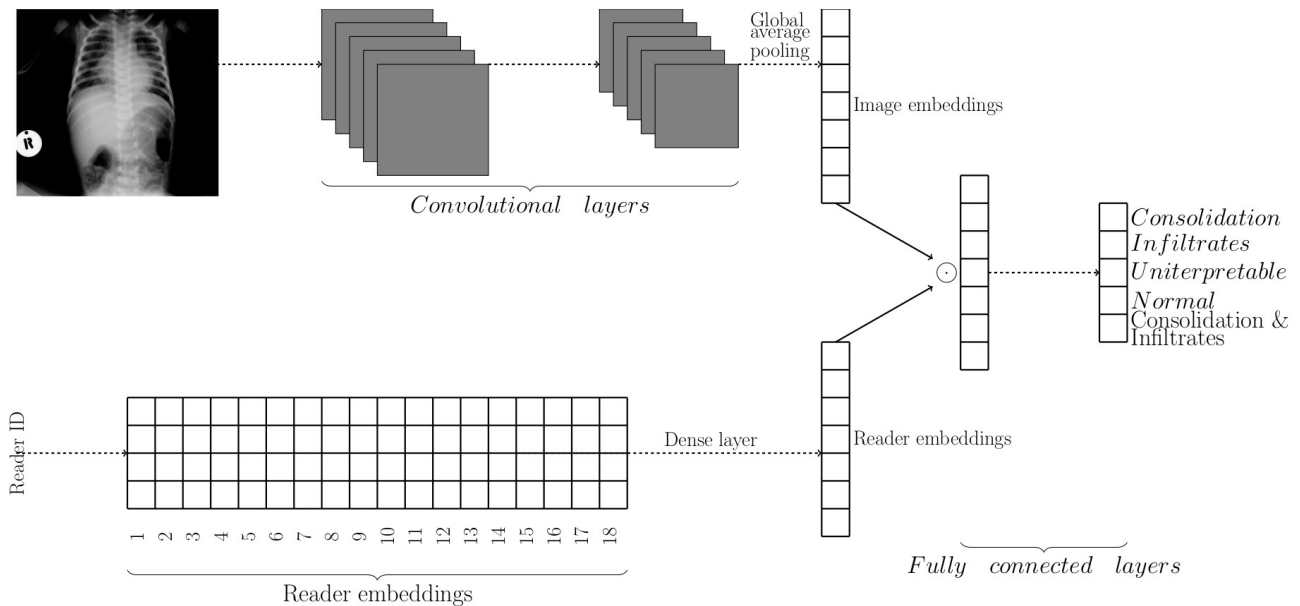


Figure 9: Model classifying PERCH CXRs conditional on reader. The model had two inputs: A CXR image and an identifier for the reader who classified the image. The output of the global average pooling of the ResNet models acted as image embeddings while the reader identifier were embedded into a vector of 32 units. The reader and imaged embeddings were combined using element wise multiplication and a single fully connected layer classified the combined embeddings into one of the five WHO categories.

3.3 Analysis of clinical data (Prediction of positive blood cultures)

The aim of this analysis was to predict blood culture results of hospitalized children using clinical signs and symptoms. We sought to evaluate the utility of transfer learning and semi-supervised techniques that leveraged an unlabelled dataset from 14 public hospitals in improving prediction accuracy on a labelled data from a single hospital. We compared a baseline models trained using logistic regression with parameter based transfer learning models based on deep learning and semi-supervised learning models based on self-training. Model performance was primarily evaluated using AUC, but we also report performance based on recall (sensitivity), specificity, precision (positive predictive value), F1 (geometric mean of recall and precision) and accuracy. We compared models fitted using labelled data only with models that incorporated the unlabelled data using transfer learning. The labelled data was collected for over 15 years and can be considered relatively large. Therefore, we tested the performance of models fitted on subsets of different sizes ranging from 5% to 100% of the labelled dataset.

3.3.1 Data sources

The study utilized two data-sets from two studies described in the sections below. We restricted our analysis to dataset of children aged 2-59 months. The following predictor variables were included: acidotic breathing, bulging fontanelle, ability to drink, capillary refill, convulsions, number of convulsions, cough, duration of cough, crackles, central cyanosis, decreased skin turgor, diarrhoea, diarrhoea duration, presence of blood in diarrhoea, difficulty breathing, fever, fever duration, indrawing, jaundice, lymphadenopathy, oedema, oxygen saturation, pallor, partial fits, weak pulse, pulse rate, respiratory rate, stiff neck, stridor, sunken eyes, temperature, temperature gradient, thrush, vomiting, vomiting everything, wasting, and wheeze. The following variables were excluded because they were not available in both datasets: AVPU (alert, voice, pain, or unresponsive) scale, irritability, flaring, grunting, body condition score (BCS), and head nodding. The following variables were excluded due to high proportion of missingness: blood pressure, and mid upper arm circumference (MUAC).

Labelled data

Labelled data was sourced from Kilifi county hospital which is located at the Kenyan coast. Kilifi county hospital is part of the Kilifi Health and Demographic Surveillance System (KHDSS), a surveillance system established in 1989 and embedded within Kenya Medical Research Institute (KEMRI) – Wellcome Trust Research Programme (Scott et al., 2012). Given the central role the hospital plays in various research studies, the laboratory facilities are likely to be better compared to typical public hospitals in low resource settings. The laboratory facilities are fully accredited by Qualogy UK limited for Good Clinical Laboratory Practice (GCLP) (Gumba et al., 2019). As part of the surveillance, all paediatric patients admitted to the hospital except those admitted with minor trauma or those undergoing elective surgery are investigated with blood cultures. In addition, the dataset consists of demographic and clinical variables (clinical signs and symptoms). The data-set spans 15 years and was collected between 2002 and 2017. The dataset consists of 44,493 observations of which 40,840 have known blood culture results.

Unlabelled data

The unlabelled data was sourced from the Clinical Information Network (CIN), a network of 14 public tertiary hospitals located in the western and central regions of Kenya (Tuti et al., 2016). The study aims at improving the quality of data collected on quality of care provided to children admitted in public hospitals. The study does not provide additional resources to the participating public hospitals above funding for data entry clerks and computers for data entry. The data was

abstracted from medical records of children admitted in paediatric wards after discharged. The hospitals receive quarterly reports on performance of selected documentation and quality of care indicators. The dataset contains information on demographic, clinical signs and symptoms, and outcome at discharge, but no reliable data on blood cultures. The lack of reliable blood culture results data is indicative of lack of reliable laboratory facilities in low resource settings. The dataset consist of 58,723 observations and was collected between 2013 and 2017. The location of of hospitals in CIN and KHDSS are shown in appendix A.

3.3.2 Data imputation and pre-processing

Predictor variables with more than 50% missingness were dropped from further analysis. For simplicity, we performed a single imputation using K-nearest neighbors. Predictor variables for each imputed variable were selected by picking variables with the highest mutual information with the variable being imputed. The number of neighbors, k , and the number of predictors to keep, were selected using K-fold cross-validation. Performance of imputation models was evaluated using F1 score and root mean square error for categorical and continuous variables, respectively.

Data pre-processing

Categorical predictor variables were converted into numeric variables using one-hot encoding with dummy variables for the first category in each variable dropped. The data was then re-scaled using min-max scaling such that all values ranged between -1 and 1.

3.3.3 Baseline model for predicting blood culture results

We fitted a logistic regression model with L2 weight decay as baseline. Nested five-fold cross validation was used to obtain confidence intervals for AUC (Figure 10). The inner loop was used to find optimal hyper-parameter value for L2 regularization using randomized search, while the outer loop was used to estimate model performance. We sampled values of L2 regularization hyper-parameter from a log uniform distribution with a minimum and maximum values of 0.0001 and 100 respectively. There was high imbalance in the outcome due to positive blood culture result being rare. Therefore, the loss of the logistic regression was weighted by the inverse of class frequencies during model training.

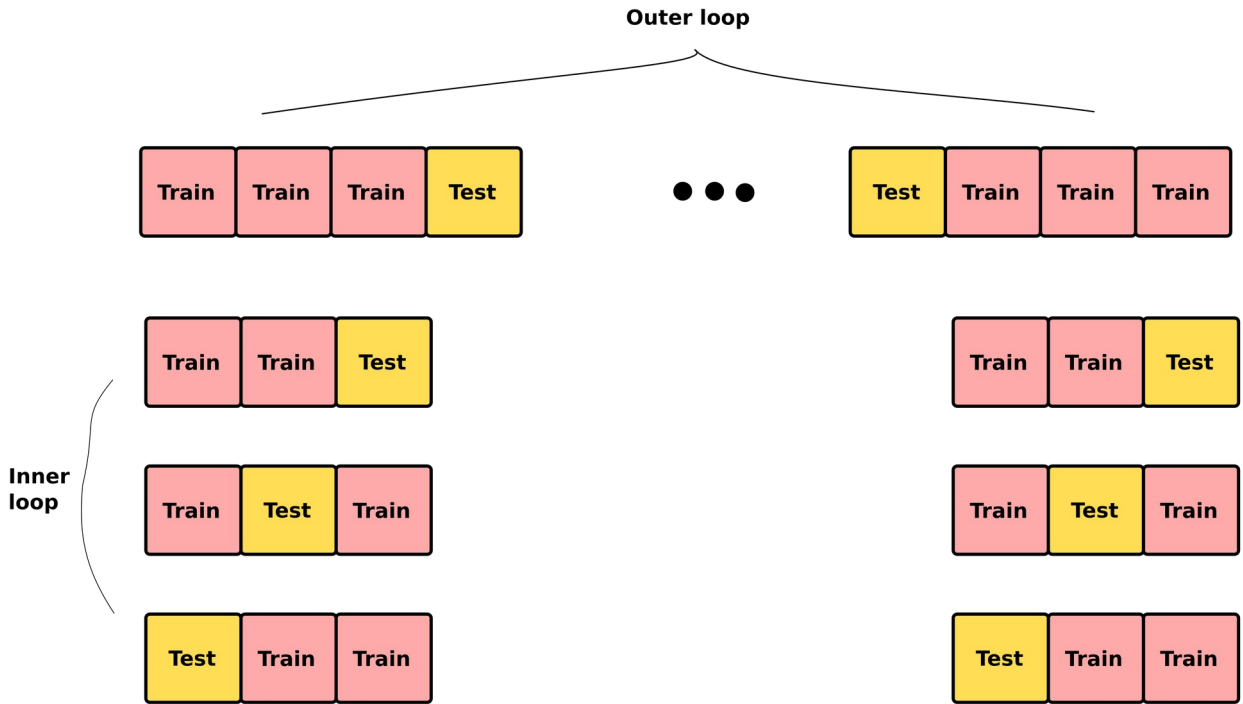


Figure 10: Nested K-fold cross-validation with a four fold outer loop and three fold inner loop. The inner loop is used to choose optimal hyper-parameters while the outer loop is used evaluate model performance.

3.3.4 Multi-layer perceptron (MLP)

We used multi-layer perceptrons (MLPs) with three hidden layers for all deep learning models. Each layer of the MLP consisted of a fully connected layer, batch normalization and LeakyRELU activation (Figure 11a). A fully connected layer is a linear function f with weight matrix parameters W and a bias parameters b such that for an input x , $f(x) = Wx + b$. Batch normalization was used to center and scale the outputs of the fully connected layer to fasten model convergence (Ioffe & Szegedy, 2015). For a vector of inputs x with mean $E[x]$ and variance $var[x]$, batch normalization was computed as:

$$y = \frac{x - E[x]}{\sqrt{Var[x] + \epsilon}} * \gamma + \beta,$$

where γ and β are parameters estimated during model training and ϵ is a very small positive number to prevent dividing by zero. LeakyRELU activation function allowed the MLP to learn a non-linear function that can be useful in classification on non-linearly separable data. The LeakyRELU function is defined as:

$$\text{LeakyRELU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ \alpha * x & \text{otherwise} \end{cases}$$

where α is the negative slope. We applied negative slope of 0.2 in all our experiments. Multiple forms of regularization were applied during model training to reduce over-fitting. We applied L1 and L2 regularization on model weights and dropout on the output of each MLP block.

We used binary cross-entropy loss and ADAM optimizer, a first order gradient based optimization algorithm to train the models (Kingma & Ba, 2017). Gradient based optimization algorithms iteratively update the parameters of the model by taking a step in a direction opposite to the partial derivatives of the loss with respect to model parameters. The loss is a measure of how well the model predictions agree with the observed values. The loss function including L1 and L2 regularization was defined as:

$$L(y, \hat{y}) = \frac{-1}{N} \times \sum_{i=1}^N w_i [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] + \lambda_1 |W| + \lambda_2 \|W\|$$

where y is the observed outcome, \hat{y} is the predicted outcome, N is the number of observations in a mini-batch, w_i is the weight of each observations in the mini-batch, and λ_1 and λ_2 control the strength of L1 and L2 regularization, respectively, on a model parameters W . we applied weights w_i to each observation to address class imbalance in blood culture results. Observations with negative blood cultures had a weight of 1.0 while those with a positive results were assigned a weight larger than 1.0 so that the models was penalized more for false negative predictions as compared to false positives.

Optimal hyper-parameters for weight assigned to observations with positive blood culture results, regularization, and learning rate for ADAM optimizer were tuned using the ASHA algorithm. ASHA hyper-parameter optimization was carried out by sampling 500 hyper-parameter configurations from the hyper-parameter search space resulting in 500 trials. The trials were then run in parallel with the number of trials halved after 50, 100 and 200 epochs by stopping trials with

low AUC on the validation dataset. The trial with the highest validation AUC after 250 epochs was chosen to be the final model. The procedure was repeated 5 times with different splits of the the training data (5 fold cross-validation).

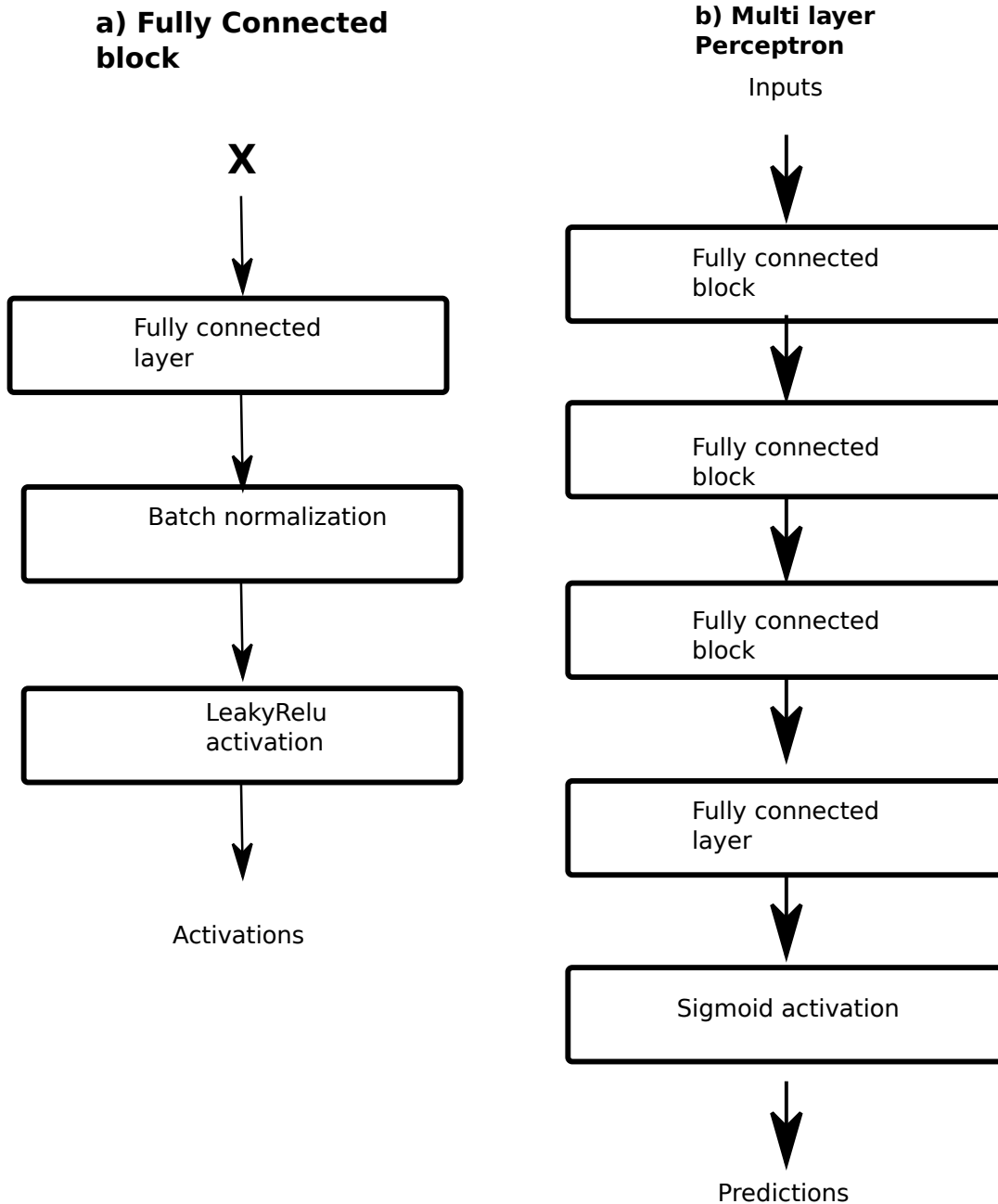


Figure 11: Multi-layer perceptron (MLP) used as backbone for classifiers and auto-encoders. The MLP model was made up of multiple fully connected blocks shown in (a). Figure (b) shows an example of an MLP model consisting of 4 fully connected blocks and a sigmoid activation layer for converting the outputs of last fully connected block into probabilities.

3.3.5 Self-supervised pre-training

Parameter based transfer learning was implemented using self-supervised pre-training. Self supervised pre-training involved fitting a self-supervised model and using the weights of the self-supervised models to initialize MLPs models for predicting positive blood cultures. Therefore, part of the self-supervised model had the same architecture as the final model for predicting positive blood culture. We used sparse and denoising auto-encoders to train self-supervised models and then used the weights of the auto-encoders to initialize MLPs for predicting blood culture results.

A bottleneck is applied to the networks to prevent the auto-encoder network from simply copying the inputs to the outputs, hence failing to learn useful representations of the inputs. Sparse auto-encoders enforce such a bottleneck by having the dimension of the latent representation z being smaller than X or applying regularization on the latent representation layer Figure 12b.

We fitted three types of sparse auto-encoders. The first sparse auto-encoder enforced a bottleneck by having the dimensions of the latent representation being much smaller than the dimensions of the inputs (16 units compared to input dimension of 39). The second and third sparse auto-encoders had latent representation with much larger dimensions than the inputs and enforced a bottleneck by adding regularization term to the cost function that encouraged the latent representation to be sparse. We explored two types of regularization terms: L1 and KL. L1 regularization was implemented by adding the L1-norm of the latent representation z to the mean square error loss $MSE(X, \hat{X}) + \lambda|z|$. Where λ is a hyper-parameter controlling the strength of regularization. On the other hand, KL regularization term were computed by calculating the KL divergence between the latent representation and a Bernoulli distribution with probability ρ . Sigmoid activation was first applied to the latent representation to restrict the values of the latent representation to between 0 and 1.

$$KL(\rho || \hat{\rho}_j) = \rho \log \frac{\rho}{\hat{\rho}_j} + (1 - \rho) \log \frac{1 - \rho}{1 - \hat{\rho}_j}$$

where $\hat{\rho}_j = 1 / (1 + \exp(-z_j))$. The hyper-parameter ρ controls the level of regularization with values closer to zero corresponding to sparse latent representation and was optimized during model

training. Therefore, the total loss for auto-encoder with KL regularization was $MSE(X, \hat{X}) + KL(\rho || \hat{\rho})$.

Denoising auto-encoders learn latent representation of the inputs by training the model to predict the inputs given the same inputs but corrupted in some way Figure 12a. We corrupted the inputs by adding random noise sampled from Gaussian or Bernoulli distributions depending on whether the input variable was categorical or continuous. For categorical variables, noise was added to the inputs by randomly flipping the sign of the dummy variables (the dummy variables had values 1 or -1 after min-max scaling). Noise was added to continuous variables by adding Gaussian noise with a mean of zero. The standard deviation of Gaussian noise and the proportion of categorical values with flipped sign were hyper-parameter optimized during model training.

Hyper-parameters for both sparse and denoising auto-encoders were optimized using ASHA algorithm. Sparse auto-encoders had the following hyper-parameters: mini-batch size, learning rate and momentum for the stochastic gradient descent optimizer, L2 regularization parameter for model weights, and dropout proportion. Denoising auto-encoders had additional hyper-parameters for level of noise added to input variables (proportion of flipped sign and Gaussian standard deviation for categorical and continuous variables respectively).

3.3.6 Self-Training

We explored semi-supervised learning using self-training. Semi-supervised learning assumes that the labelled and unlabelled data originate from the same distribution. The assumption that both labelled and unlabelled datasets originate from the same distribution was not unreasonable given that both datasets were obtained from public hospitals in Kenya. Self-training was performed in two steps. In the first step, a classification model was fitted using the labelled data. Pseudo labels for the unlabelled data were then generated by making predictions on both labelled and unlabelled data using the model fitted in the first step. In the second step, a linear regression model was fitted to predict the pseudo-labels using the unlabelled data.

We used the baseline logistic regression model in section 3.3.3 in the first step of self-training. The logistic regression model was used to predict the log odds of positive blood culture results on the

unlabelled data. The predicted log odds were then used as pseudo-labels for a linear regression model. The predictions of linear regression models were converted into probabilities using the sigmoid function:

$$\sigma(x) = \frac{1}{1 + \exp(-x)}$$

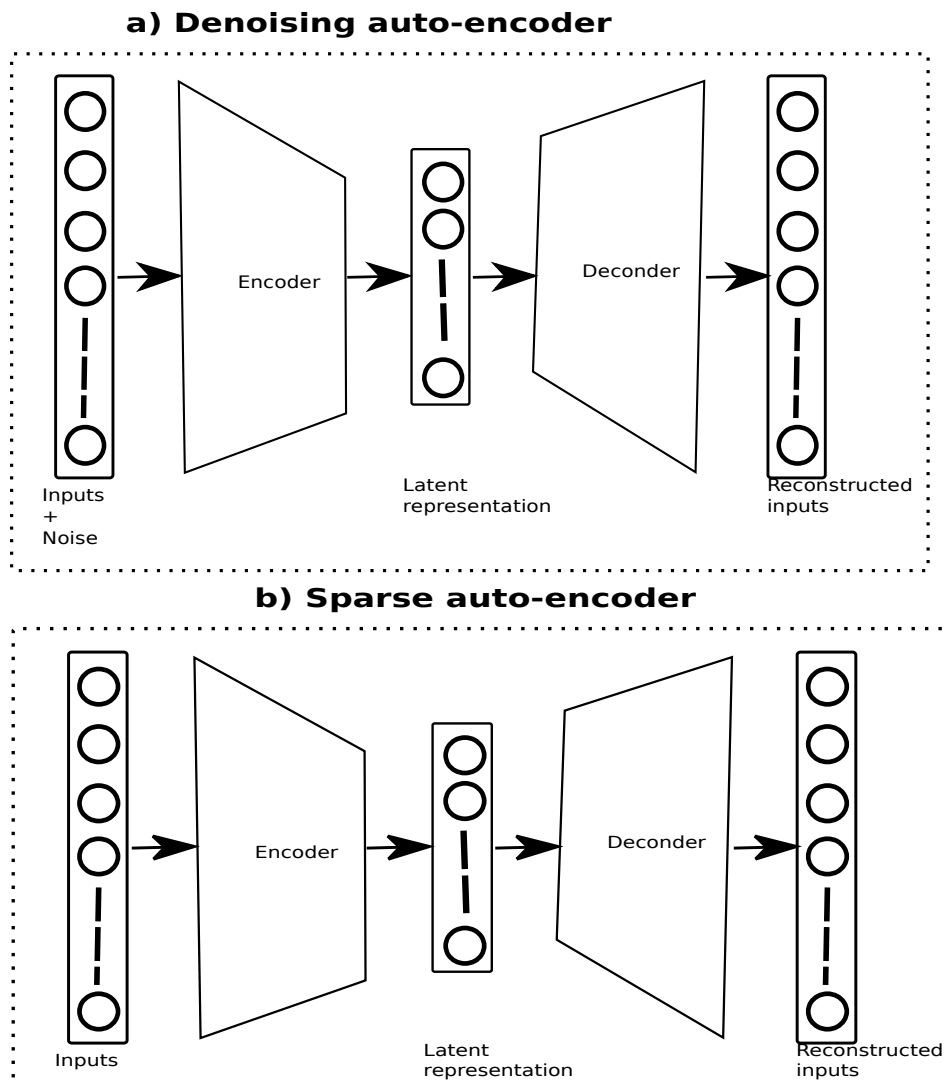


Figure 12: Self-supervised models. a) A denoising auto-encoder predicts the inputs given corrupted inputs. b) A sparse auto-encoder learns useful representations of the inputs by applying regularization on the latent representation to encourage sparse latent representation

4 Results

4.1 Analysis of bio-signals

4.1.1 Feature learning using contrastive learning

Contrastive learning model trained on both labelled and unlabelled PPG signals had lower noise contrastive estimation (NCE) loss and higher accuracy on the validation dataset compared to model trained on labelled data only. Contrastive learning model trained on labelled data only had an accuracy (classifying whether two PPG signals originated from the same patient) of 0.73 compared to an accuracy of 0.91 for model trained using both labelled and unlabelled dataset (Figure 13). Table 2 shows optimal hyper-parameters for SSL models identified using the ASHA algorithm.

Table 2: Optimal hyper-parameters for SSL models

Hyper-parameter	Self-supervised: Labelled and Unlabelled	Self-supervised: Labelled only
Proportion of signals with Gaussian noise augmentation	1	0.8
Number of slices for with signal slicing and permutation augmentation	10	2
Proportion of signals with signal slicing and permutation augmentation	0.2	0.8
Batch size	32	16
Dropout proportion	0.02	0.01
NCE temperature	0	0
Weight decay parameter for convolutional layers	0	0
Weight decay parameter for fully connected layers	0	0
Learning rate for convolutional layers	0	0
Learning rate for fully connected layers	0	0.01

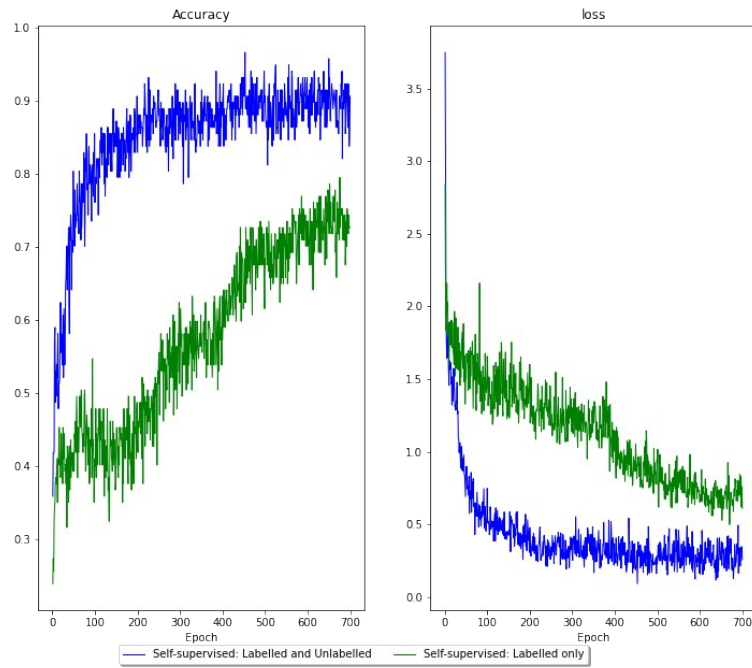
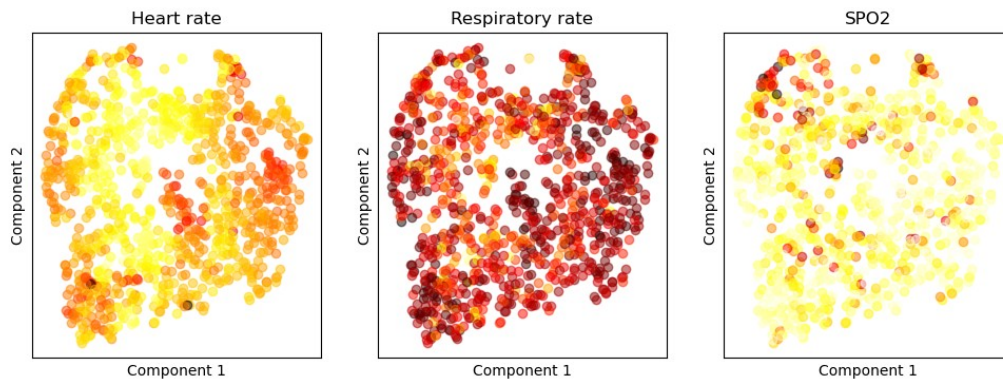


Figure 13: Validation accuracy and NCE loss of SSL model trained using contrastive learning. The models are trained to predict whether two PPG segments originate from the same patient.

Scatter plot of first and second components of t-SNE dimensionality reduction show that the extracted features are highly correlated with the physiological parameters (heart rate, respiratory rate and SpO2). In addition, features extracted using SSL model trained using all PPG signals were better as separating patients according to the values of the physiological parameters compared to features extracted using SSL model trained on labelled PPG signals only (Figure 14). For instance, patients with high values of heart rate appear on the left side of tSNE plot of SSL model trained using all PPG signals. A plot of first and second components of the PCA shows that features extracted using PCA were not discriminative of any of the physiological parameters (Figure 15).

(a) Labelled PPG signals



(b) Labelled and unlabelled PPG signals

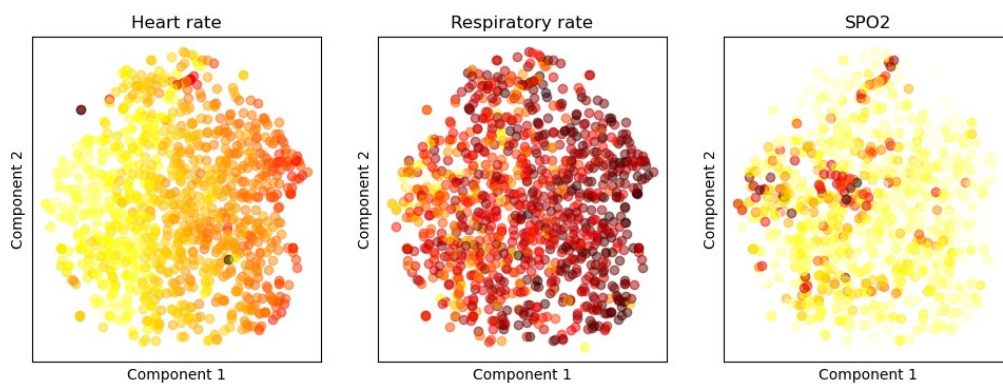


Figure 14: Dimensionality reduction of featured extracted using contrastive learning using t-SNE. Points are shaded by values of heart rate, respiratory rate and SPO2. Contrastive model is trained using labelled data only in (a) and both labelled and unlabelled data in (b). Lighter color represent larger values.

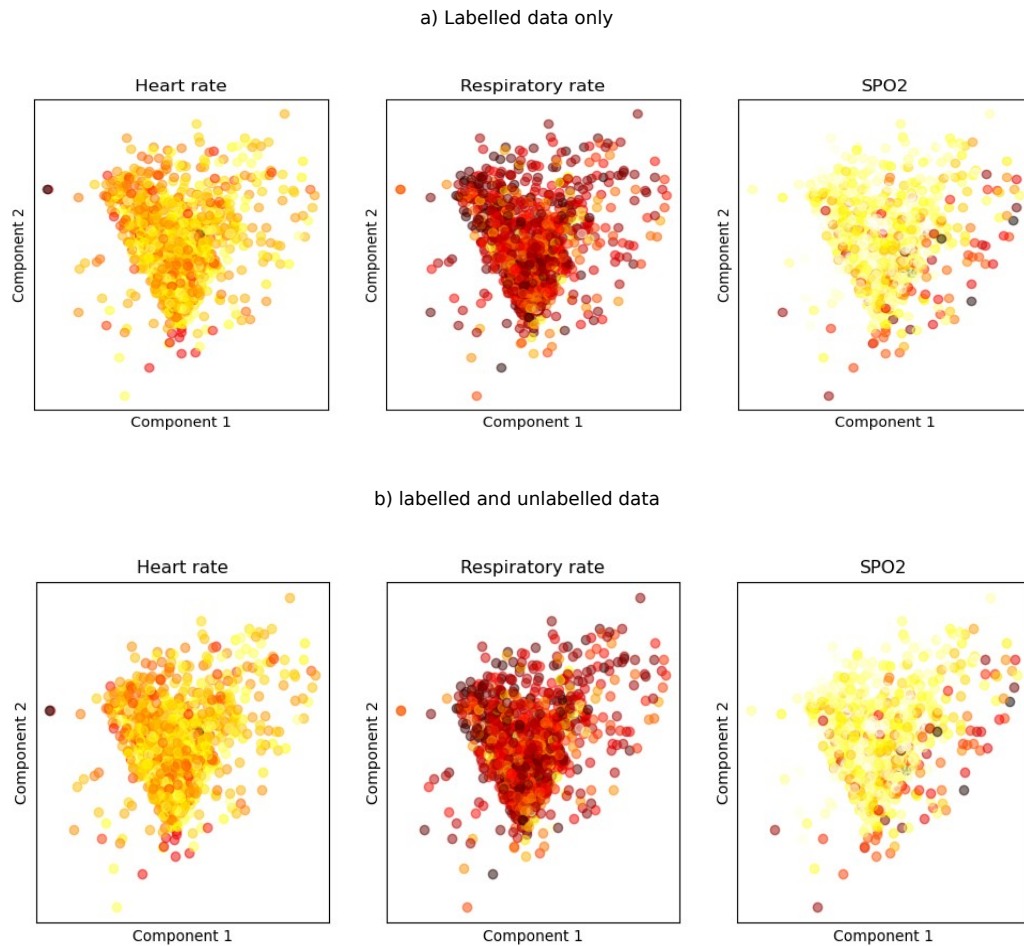


Figure 15: Dimensionality reduction of raw PPG signals using PCA. Points are shaded by values of heart rate, respiratory rate and SPO2. PCA model is fitted using labelled data only in (a) and both labelled and unlabelled data in (b).

4.1.2 Classification and regression models using extracted features

Features extracted using SSL models trained using both labelled and unlabelled PPG signals were better at solving classification and regression tasks compared to features extracted using SSL models trained on labelled PPG signals only. Table 3 show that features extracted using SSL model trained on both labelled and unlabelled data were better at predicting heart rate (R^2 0.82 vs 0.71), respiratory rate (R^2 0.36 vs 0.24), and SpO2 (R^2 0.70 vs 0.18). Features extracted using PCA were not predictive of heart rate, respiratory rate or SpO2. Table 4 shows that logistic regression models for predicting hospitalization had the highest prediction performance when trained using clinical features and features from SSL models trained using both labelled and unlabelled PPG signals (AUC 0.89). Models trained on clinical feature only had an AUC of 0.86, while models trained on heart-rate and SpO2 only (obtained using a pulse oximeter) had an AUC of 0.72.

Table 3: Root mean square error (RMSE) and coefficient of determination (R2) of regression models fitted using features extracted using contrastive learning and PCA.

	Metric	Contrastive learning		PCA	
		Labelled	Labelled & unlabelled	Labelled	Labelled & unlabelled
SpO2	R2	0.18	0.70	-0.01	-0.02
	RMSE	4.0	2.4	4.4	4.4
Heart rate	R2	0.71	0.82	-0.01	-0.01
	RMSE	14.3	11.4	26.7	26.7
Respiratory rate	R2	0.24	0.36	0	0
	RMSE	13.2	12.1	15	15.1

.....

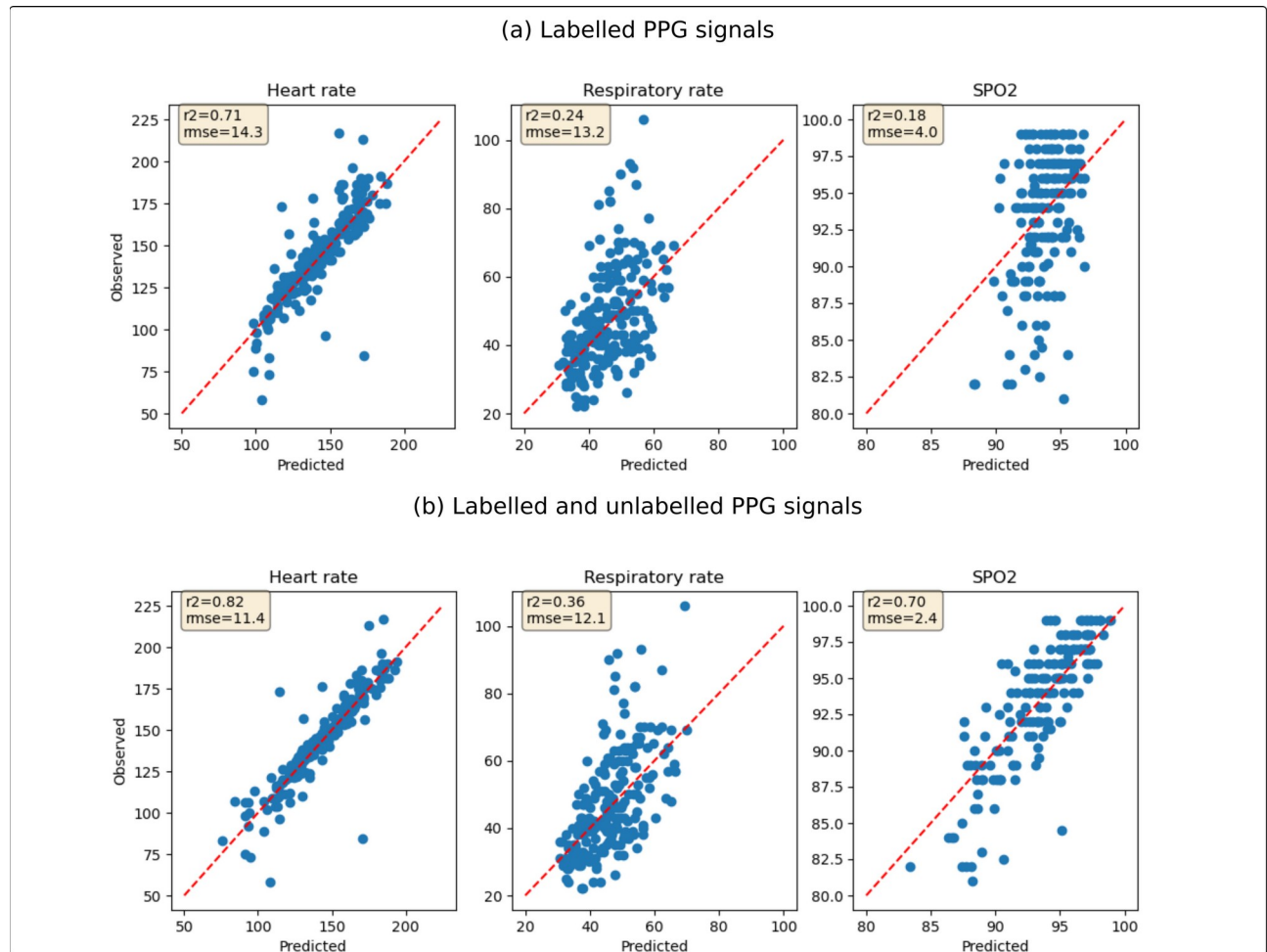


Figure 16: RMSE and R2 of linear regression models for predicting heart rate, respiratory rate and SpO2 using features extracted from SSL models. SSL models are fitted using labelled PPG signals only in (a) and all PPG signals in (b).

4.1.3 End to end Deep Learning

We compared different initialization schemes for end-to-end deep learning models for predicting hospitalization. End-to-end model initialized randomly had an AUC of 0.73, while end-to-end models initialized using weights of SSL models had AUCs of 0.80 and 0.77 for SSL models trained using all PPG signals and labelled PPG signals, respectively Table 4.

Table 4: Performance of logistic regression and end-to-end models for predicting hospitalization. Logistic regression models were trained on clinical features, SSL features, or both. Deep learning models were initialized either randomly, using weights of SSL models trained on labelled data only, or using weights of SSL models trained using both labelled and unlabelled PPG signals.

Model	Initialization/Features	Precision	Sensitivity	Specificity	AUC
Deep learning	Random	0.20	0.73	0.58	0.73
	SSL(Labelled & Unlabelled)	0.27	0.85	0.67	0.80
	SSL(Labelled)	0.21	0.81	0.56	0.77
Logistic regression	Clinical	0.34	0.81	0.77	0.86
	Clinical & SSL(Labelled & Unlabelled)	0.33	0.85	0.75	0.89
	Clinical & SSL(Labelled)	0.36	0.81	0.79	0.87
	SPO2 & heart rate	0.22	0.69	0.65	0.72
	SPO2, heart rate & SSL(Labelled & Unlabelled)	0.24	0.77	0.64	0.80
	SPO2, heart rate & SSL(Labelled)	0.20	0.88	0.50	0.83
	SSL(Labelled & Unlabelled)	0.23	0.73	0.64	0.80
SSL(Labelled)	0.20	0.88	0.50	0.83	

Table 5 show optimal hyper-parameters for all end-to-end models. End-to-end models initialized using weights of the SSL model reached convergence in fewer epochs

Table 5: Optimal hyper-parameters for end-to-end models with different initialization schemes

Hyper-parameter	SSL: Labelled & Unlabelled	SSL: Labelled	Random
Dropout proportion	1.00E-2	1.35E-5	5.00E-3
Batch size	64	128	16
Label smothering parameter	0.01	0.01	0
Learning rate for convolutional layers	2.08E-3	1.92E-4	2.99E-4
Weight decay parameter for convolutional layers	5.92E-6	3.35E-2	2.79E-6
Learning rate for fully connected layer	1.28E-4	8.25E-3	1.48E-3
Weight decay for fully connected layer	2.75E-05	4.45E-3	5.18E-4
Proportion of segments with Gaussian augmentation	0.8	0.5	0
Number of slices for signal slicing and permutation augmentation	10	20	2
Proportion of signals with slicing and permutation augmentation	0.5	0.3	0.9
Number of training iterations	100	50	150

4.2 Analysis of chest radiographs

4.2.1 Supervised Pre-training models

Table 6 shows AUCs of ResNet18, ResNet34 and ResNet50 models trained to classify Chestray 14 dataset. Each model could make 15 binary predictions. Model architecture did not have any effect on performance of models trained to classify Chestray-14 dataset. The average AUC (over the 14 conditions) was 0.79 for ResNet18, and 0.8 for both ResNet34 and ResNet50. The AUC for different conditions ranged from 0.69 for infiltrates to 0.90 for hernia. ResNet34 model architecture had the best performance for 9 of the 14 conditions.

Table 6: AUC of models fitted on Chestray 14 dataset. AUCs are for binary classification comparing each class against all the rest (one vs rest).

condition	ResNet18	ResNet34	ResNet50
Atelectasis	0.76	0.76	0.77
Cardiomegaly	0.88	0.87	0.87
Effusion	0.81	0.82	0.82
Infiltration	0.69	0.68	0.7
Mass	0.81	0.83	0.81
Nodule	0.74	0.74	0.76
Pneumonia	0.71	0.71	0.7

Pneumothorax	0.83	0.84	0.84
Consolidation	0.75	0.75	0.74
Edema	0.83	0.84	0.83
Emphysema	0.84	0.88	0.88
Fibrosis	0.81	0.81	0.8
Pleural Thickening	0.75	0.77	0.77
Hernia	0.9	0.88	0.87
Average	0.79	0.8	0.8

4.2.2 Unsupervised pre-training

Unsupervised pre-trained models were trained using contrastive learning to classify whether two CXR images were obtained by applying augmentation on the same or different CXR images. The SSL model embedded each CXR into a compressed representation of 32 units. For visualization purposes, the dimensions of the compressed representation was reduced to 2 using t-SNE dimensionality reduction technique. The first and second components of t-SNE were plotted in a scatter plot and the points colored by values of children's age, WHO categories (labels), and site. The t-SNE plots show that features extracted using self-supervised models had information about the age of patients but could not distinguish between images with different WHO categories or from different sites (Figure 17). Clustering of embedding by age was evident for both SSL model trained using PERCH dataset and model trained using all dataset (PERCH and Chestray-14).

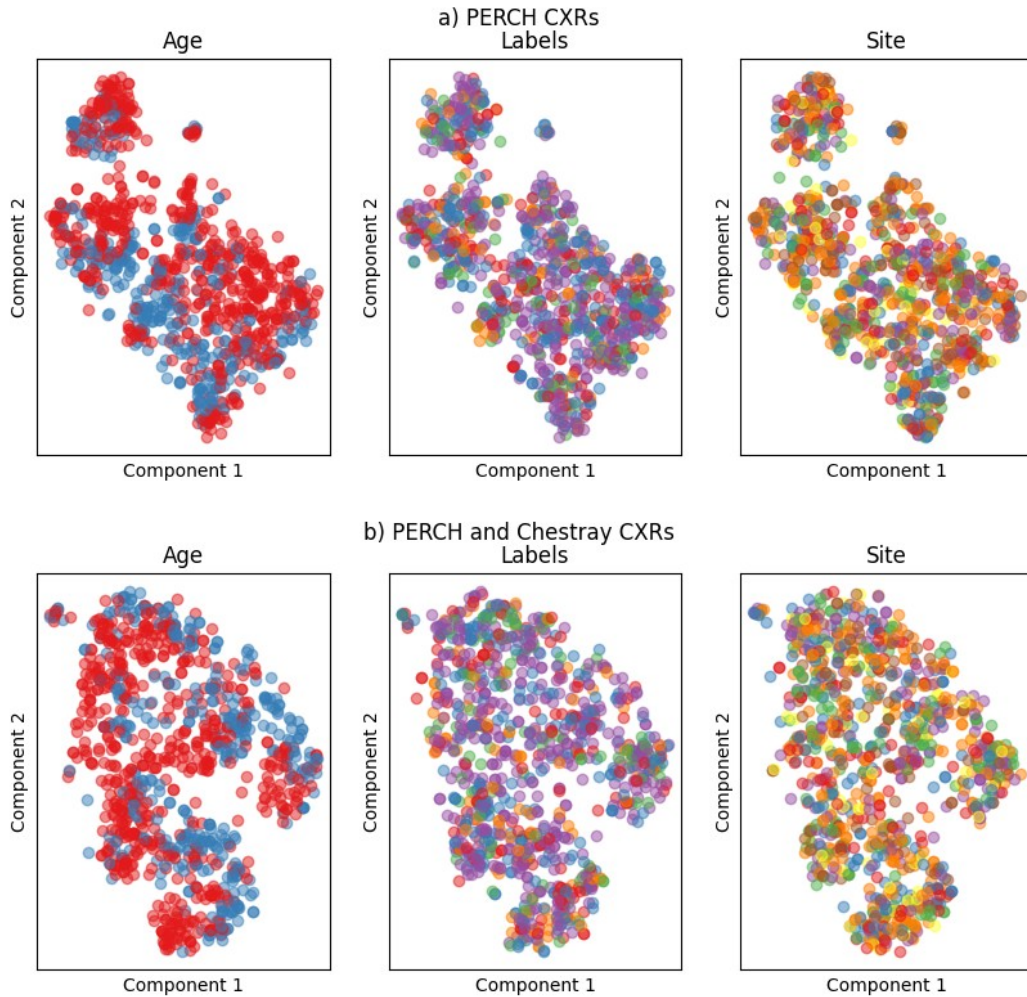


Figure 17: t-SNE plot of hidden representation learned using self-supervised learning (ResNet50 encoder). The points are shaded by age (1-11 months vs 12-59 months), labels (one of the five WHO categories), and site (one of seven countries). Legends are excluded for clarity.

Table 7: Accuracy and AUC of models trained to classify PERCH CXRs. The baseline model is initialized using ImageNet weights, while the supervised and unsupervised models are initialized using weights of models trained on Chest-ray14 dataset.

			ResNet18	ResNet34	ResNet50			
	Includes Chestray 14 dataset	Reader embedding	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC
Baseline	No	No	0.59	0.84	0.57	0.82	0.59	0.84
	No	Yes	0.61	0.86	0.6	0.86	0.6	0.86
Supervised Pre-training	Yes	No	0.61	0.84	0.6	0.84	0.61	0.84
	Yes	Yes	0.62	0.86	0.6	0.86	0.62	0.87
Unsupervised Pre-training	No	No	0.58	0.83	0.59	0.81	0.6	0.84
	No	Yes	0.59	0.84	0.57	0.83	0.59	0.85
	Yes	No	0.6	0.84	0.6	0.85	0.59	0.83
	Yes	Yes	0.6	0.86	0.62	0.86	0.59	0.84

Models with reader embeddings (ensemble) had higher overall AUCs for all initialization schemes (Figure 18). All models had lower AUCs for infiltrates compared to other categories. The average AUC was 0.832 for consolidation, 0.791 for other infiltrates, 0.874 for both consolidation and other infiltrates, 0.869 for normal, and 0.856 for uninterpretable.

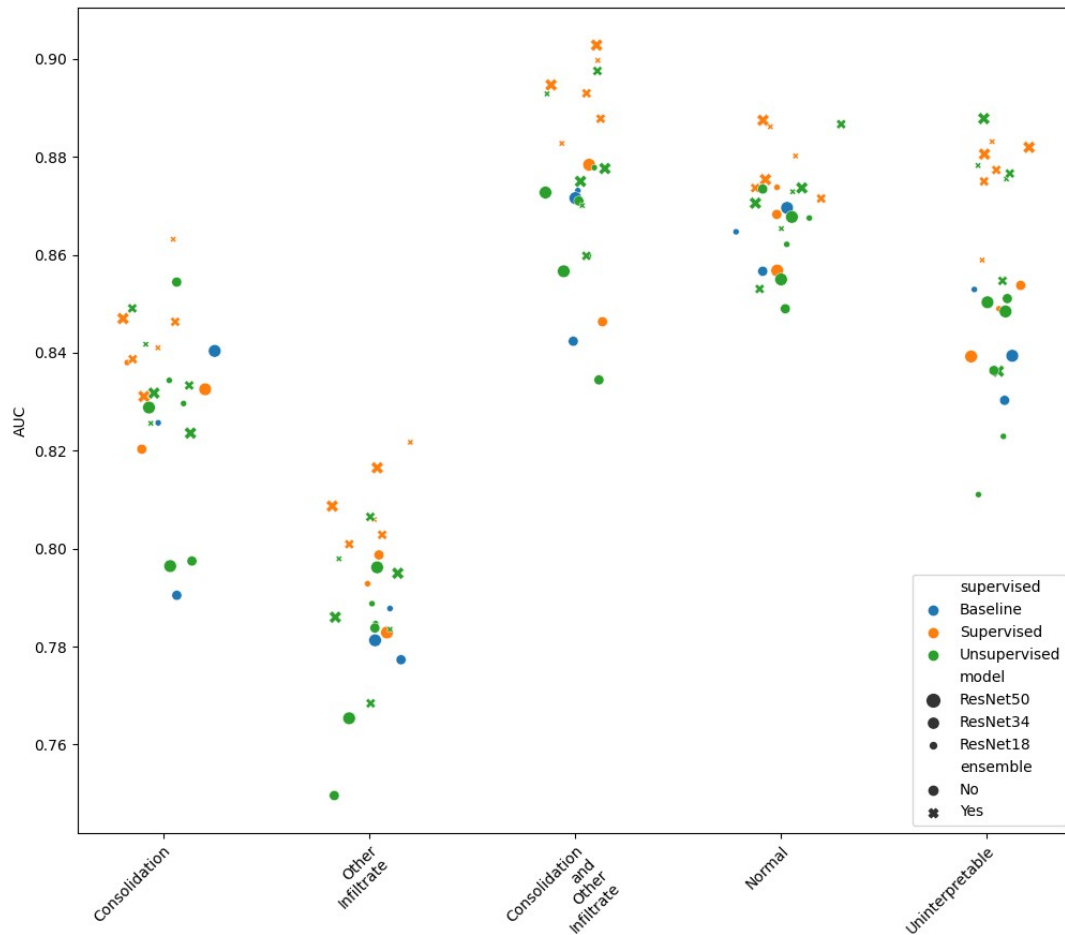


Figure 18: Dis-aggregated AUCs of models trained to classify PERCH CXRs. The baseline models are initialized using Imagenet weights. Models are categorized by initialization scheme (supervised vs unsupervised), model size (ResNet50, ResNet34, or ResNet18), and whether reader embeddings were included (ensemble).

Boxplots of AUCs showed that models fitted using ResNet18 model architecture had slight advantage over the larger models fitted using ResNet34 and ResNet50 architectures but the difference was not statistically significant (Figure 19). Incorporating reader embeddings had statistically significant improvement on AUCs (mean 0.854 vs 0.834, p-value = 0.002). The mean AUC increased from 0.834 for models initialized using weights from the torchvision library

(Imagenet) to 0.84 for models initialized using weights from the self-supervised models trained on both PERCH and Chestray-14 datasets and 0.853 for models initialized using supervised models trained on Chestray-14 ($p=0.07$). The inter-quantile range for all boxplots were too wide given the small number of points used to construct each boxplot.

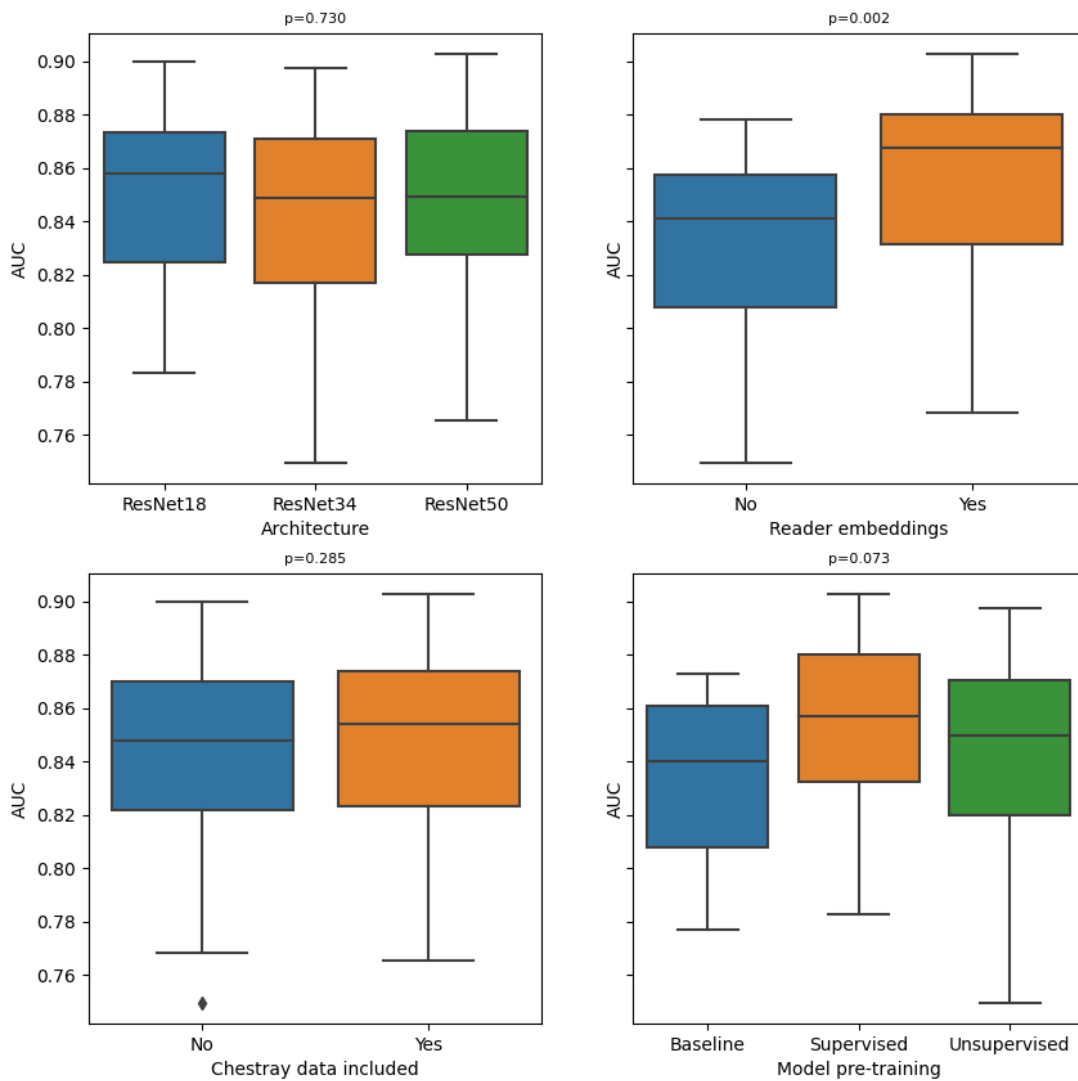


Figure 19: Boxplots of model performance on PERCH dataset categorized by model architectures and initialization schemes. The p -value (p) is used to test for difference in means (t -test for variables with two categories and analysis of variance for variables with more than two categories).

4.2.3 Multi-task Analysis

Table 8 shows performance of multi-task learning models trained to simultaneously classify PERCH and Chestray-14 CXR images. Two approaches of multi-task learning were compared. The first approach had shared weights for models classifying PERCH and Chestray-14 datasets (hard parameter sharing). The number of layers with shared weights was a hyper-parameter. The second approach used soft parameter sharing where the mean absolute difference between corresponding layers of model classifying PERCH and Chestray-14 images was added to the loss function, encouraging the two networks to have similar parameter values. The accuracy on PERCH CXRs for both multi-task models ranged between 0.57 and 0.59, while the mean AUCs ranged between 0.82 and 0.84. For multi-task learning model with soft parameter sharing, the optimal number of shared blocks was 2 out of 4 for ResNet18 and 3 out of 4 for both ResNet34 and ResNet50.

For multitask model with hard parameter sharing, the mean AUCs for Chestray-14 dataset were 0.71, 0.68, and 0.73 for ResNet18, ResNet43 and ResNet50 respectively. The AUCs of multi-task learning models with soft parameter sharing were 0.73, 0.75, and 0.66 for ResNet18, ResNet34 and ResNet50 respectively.

Table 8: PERCH dataset classification accuracy and AUCs for models trained using multi-task learning

Model	Hard parameter sharing		Soft parameter sharing	
	Accuracy	AUC	Accuracy	AUC
ResNet18	0.59	0.83	0.58	0.83
ResNet34	0.59	0.82	0.59	0.84
ResNet50	0.57	0.83	0.59	0.84

4.2.4 Model with highest performance on PERCH CXRs

The best performing model in classification of PERCH CXR images had an accuracy of 0.62 and average AUC of 0.87. The model had ResNet50 architecture, was initialized using weights of CNN model classifying Chestray-14 CXR (supervised pre-training) and had reader embeddings. The model had an AUC of 0.85 for consolidation, 0.82 for infiltrated, 0.90 for both consolidation and infiltrates, 0.87 for normal and 0.88 for un-interpretable. The classification accuracy for any consolidation (consolidation or consolidation and other infiltrates) was 0.87 while that of any infiltrates was 0.76. There was high variability in model performance across sites. The model

accuracy was 0.65 in Bangladesh, 0.70 in Gambia, 0.66 in Kenya, 0.60 in Mali, 0.56 in South Africa, 0.67 in Thailand and 0.52 in Zambia. Figure 20b shows that model accuracy increased by age. Model accuracy increased sharply from below 50% for children below 8 months of age to above 65% and then stagnated for older children. The model had an accuracy of 0.60 for children below 12 months of age and 0.65 for older children.

A confusion matrix of the observed versus predicted WHO categories of the PERCH test CXR is shown on Figure 20a. The model had the best accuracy for CXR images that were normal. Eighty percent of normal CXR were correctly classified while the rest were miss-classified as infiltrates. For CXR with both consolidation and infiltrates, 40% were miss-classified as consolidation only and 30% as infiltrates only. Forty percent of CXR images that were un-interpretable were miss-classified as normal.

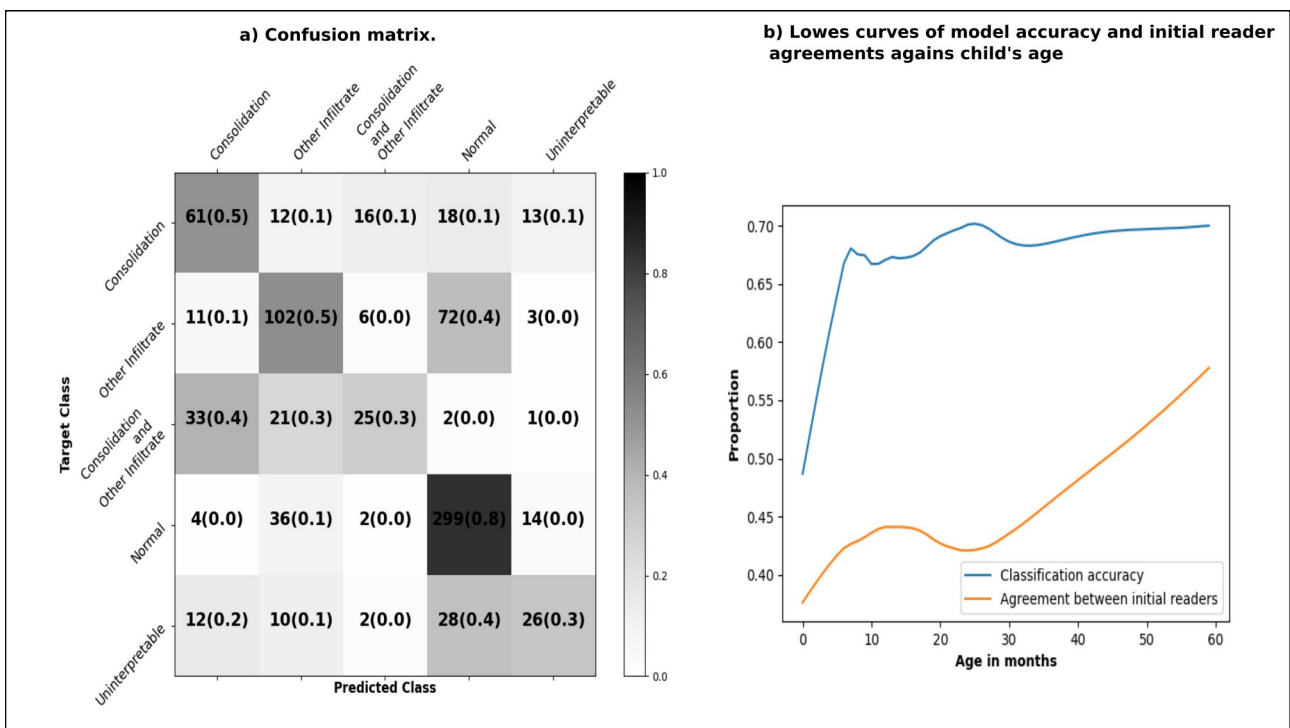


Figure 20: Confusion matrix and lowes plot of classification accuracy against age of model with highest accuracy and AUC.

A Grad-CAM visualization of the model showed that the model used the correct regions of the CXR images in making predictions Figure 21.

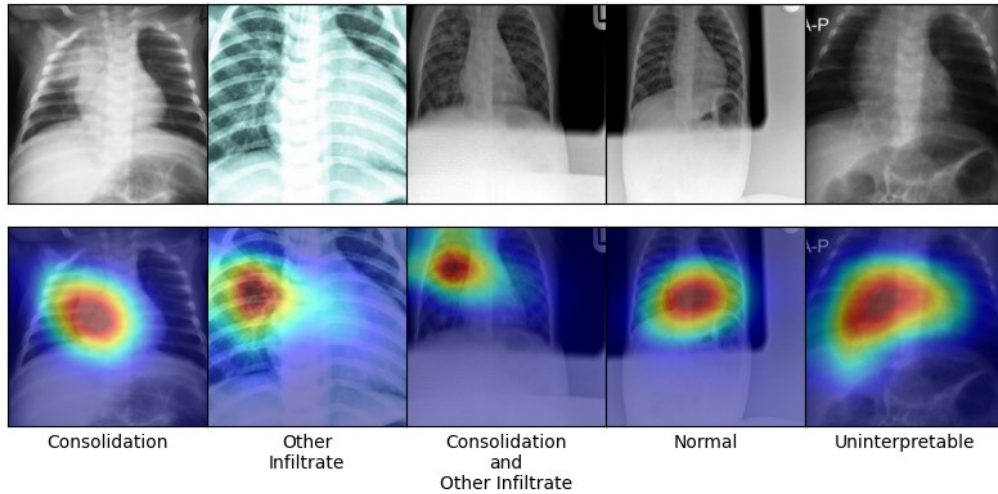


Figure 21: Grad-CAM visualization of randomly selected images that were correctly classified by the best model. The top row shows the original CXR images while the bottom row shows heat-maps of regions the model considered important in making the prediction. The model was able to identify the relevant regions of the CXRs when making predictions.

4.3 Analysis of clinical data

Distribution of predictor variables

Table 9 shows the distribution of predictor variables in the CIN and Kilifi datasets. Hypothesis tests for differences in distribution of predictors between CIN and Kilifi dataset are computed using independent sample t-test for continuous variables and chi-square test for categorical variables. There was strong evidence of differences in CIN and Kilifi datasets for all variables except presence of convulsion, cough greater than two weeks, and difficulty breathing. There was high class imbalance with majority class accounting for more than 95% of observations in variables for bulging fontanelle, capillary refill greater than 3 seconds, cough duration greater than two weeks, cyanosis, bloody diarrhoea, diarrhoea duration greater than two weeks, jaundice, lymphadenopathy, oedema, partial fits, weak pulse, stiff neck, stridor, and thrust.

Table 9: Distribution of predictor variables. Percentages are reported for the presence of categorical sign/symptom and mean for continuous variables. P-values for differences in distribution between CIN and Kilifi datasets were computed using Chi-square test for categorical variables and independent sample t-test for continuous variables.

Predictor	Category	All	CIN	Kilifi	p-value
Acidotic breathing	Yes	6307(6.11%)	1511(2.57%)	4796(10.78%)	<0.001
Bulging fontanelle	Yes	934(0.90%)	604(1.03%)	330(0.74%)	<0.001
Can drink	Yes	71123(68.91%)	45998(78.33%)	25125(56.47%)	<0.001

Cap refill	>3 Sec	1030(1.00%)	672(1.14%)	358(0.80%)	<0.001
Convulsions	No	78611(76.16%)	44050(75.01%)	34561(77.68%)	0.052
Convulsions number	mean(sd)	0.3(1.1)	0.4(1.1)	0.2(1.0)	<0.001
Cough	Yes	55696(53.96%)	33760(57.49%)	21936(49.30%)	<0.001
Cough duration	mean(sd)	2.7(6.5)	2.8(6.3)	2.6(6.6)	<0.001
Cough duration > 2wks	Yes	4056(3.93%)	2298(3.91%)	1758(3.95%)	0.847
Crackles	Yes	19712(19.10%)	12379(21.08%)	7333(16.48%)	<0.001
Cyanosis	Yes	638(0.62%)	429(0.73%)	209(0.47%)	<0.001
Decreased skin turgor	Yes	13673(13.25%)	9830(16.74%)	3843(8.64%)	<0.001
Diarrhoea	Yes	29380(28.46%)	18646(31.75%)	10734(24.13%)	<0.001
Diarrhoea bloody	Yes	916(0.89%)	742(1.26%)	174(0.39%)	<0.001
Diarrhoea duration	mean(sd)	0.8(2.6)	1.1(2.7)	0.3(2.2)	<0.001
Diarrhoea duration > 14 days	Yes	727(0.70%)	644(1.10%)	83(0.19%)	<0.001
Difficulty breathing	Yes	36893(35.74%)	20718(35.28%)	16175(36.35%)	0.555
Fever	Yes	76686(74.30%)	43013(73.25%)	33673(75.68%)	<0.001
Fever duration	mean(sd)	2.7(5.1)	2.5(4.6)	3.0(5.6)	<0.001
Indrawing	Yes	29859(28.93%)	16573(28.22%)	13286(29.86%)	0.003
Jaundice	Yes	2027(1.96%)	1091(1.86%)	936(2.10%)	0.025
Lymphadenopathy	Yes	1247(1.21%)	461(0.79%)	786(1.77%)	<0.001
Oedema	Yes	3129(3.03%)	1101(1.87%)	2028(4.56%)	<0.001
Oxygen saturation	mean(sd)	96.3(4.2)	94.4(4.7)	97.4(3.4)	<0.001
Pallor	Yes	18768(18.18%)	8544(14.55%)	10224(22.98%)	<0.001
Partial fits	Yes	3028(2.93%)	1507(2.57%)	1521(3.42%)	<0.001
Pulse	Weak	4314(4.18%)	2617(4.46%)	1697(3.81%)	<0.001
Pulse rate	mean(sd)	142.7(30.7)	129.8(29.3)	154.5(26.9)	<0.001
Respiratory rate	mean(sd)	39.1(9.7)	39.8(9.8)	38.4(9.5)	<0.001
Stiff neck	Yes	1724(1.67%)	1332(2.27%)	392(0.88%)	<0.001
Stridor	Yes	1693(1.64%)	1431(2.44%)	262(0.59%)	<0.001
Sunken eyes	No	87022(84.31%)	48542(82.66%)	38480(86.49%)	<0.001
Temperature	mean(sd)	37.7(1.2)	37.6(1.2)	37.8(1.2)	<0.001
Temperature gradient	Yes	7022(6.80%)	2254(3.84%)	4768(10.72%)	<0.001
Thrust	Yes	2038(1.97%)	1296(2.21%)	742(1.67%)	<0.001
Vomiting	No	62775(60.82%)	31192(53.12%)	31583(70.98%)	<0.001
Vomiting everything	No	46832(45.37%)	13089(22.29%)	33743(75.84%)	<0.001
Wasting	Yes	5726(5.55%)	1505(2.56%)	4221(9.49%)	<0.001
Wheeze	Yes	5897(5.71%)	3855(6.56%)	2042(4.59%)	<0.001

4.3.1 Missing data and imputation

The proportion of missing data in the predictor variables ranged between 56% and less than 1%. Variables for vomiting everything, oxygen saturation, and severe wasting had more than 50% missing in the CIN dataset. Variable for vomiting everything had the highest level of missingness in the combined dataset (CIN and Kilifi) with 41% missingness.

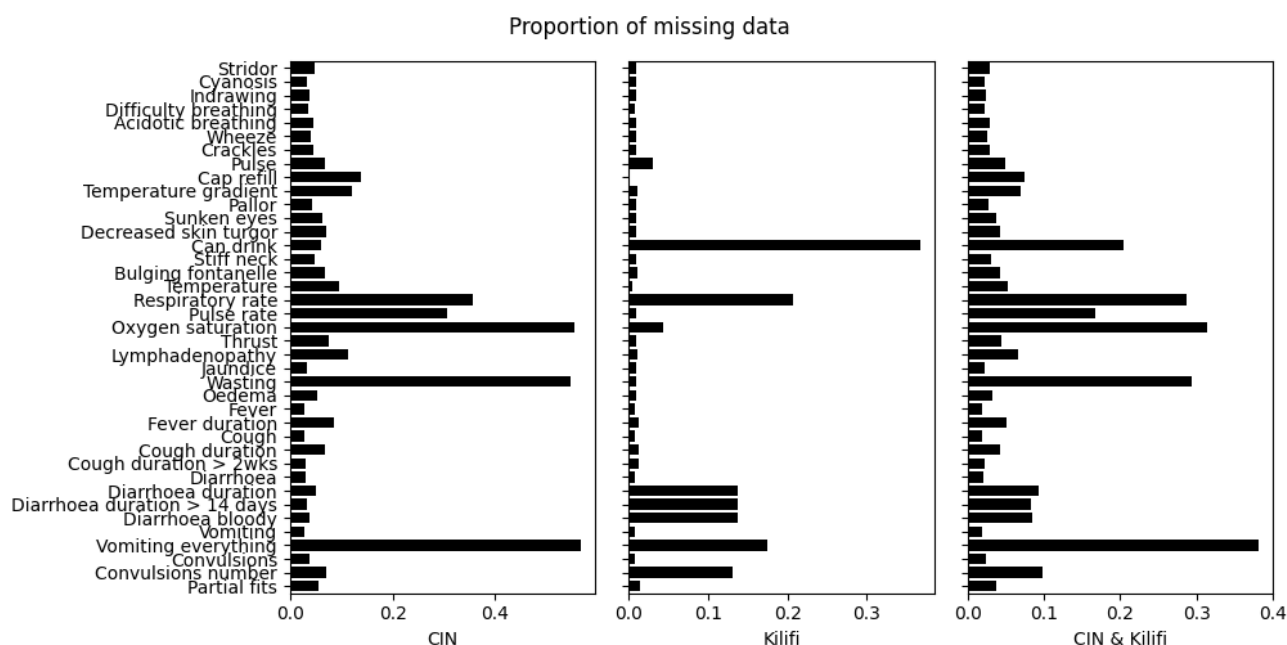


Figure 22: Proportion of missing data in the predictor variables

Table 10 show the performance of KNN imputation models for each predictor variable. The performance of the imputations were assessed using F1 score for categorical variables and root mean squared error (RMSE) for continuous variables. Low F1 scores were observed for categorical variables with high class imbalance because missing values in those variables were predicted to belong to the majority class. Therefore, imputation for predictors with high class imbalance was similar to univariate imputation using the majority class.

Table 10: Performance of KNN imputation model. Imputation model was assessed using F1 score for categorical variables and root mean square error (RMSE) for continuous variables.

Predictor	Metric	CIN	Kilifi
Convulsions number		0.33	0.23
Cough duration		0.33	0.32
Diarrhoea duration		0.35	0.21
Fever duration		0.29	0.28
Oxygen saturation		0.28	0.35
Pulse rate		0.27	0.25
Respiratory rate		0.26	0.25
Temperature	RMSE	0.27	0.24
Acidotic breathing		0.03	0.31
Bulging fontanelle		0.01	0.05
Can drink	F1	0.91	0.98

Cap refill	0.08	0.05
Convulsions	0.41	0.5
Cough	0.74	0.74
Cough duration > 2wks	0.2	0.34
Crackles	0.43	0.49
Cyanosis	0.02	0.03
Decreased skin turgor	0.43	0.65
Diarrhoea	0.57	0.63
Diarrhoea bloody	<0.01	0.01
Diarrhoea duration > 14 days	0.05	0.13
Difficulty breathing	0.67	0.79
Fever	0.86	0.89
Indrawing	0.65	0.8
Jaundice	0.05	0.03
Lymphadenopathy	<0.01	0.03
Oedema	0.04	0.22
Pallor	0.23	0.3
Partial fits	0.02	0.03
Pulse	0.22	0.33
Stiff neck	0.04	0.02
Stridor	0.03	0.01
Sunken eyes	0.45	0.66
Temperature gradient	0.21	0.25
Thrust	0.02	0.05
Vomiting	0.6	0.62
Vomiting everything	0.54	0.27
Wasting	0.12	0.31
Wheeze	0.15	0.21

4.3.2 Auto-encoders

Figure 23 shows performance of sparse and denoising auto-encoders on the validation data. A comparison is made between auto-encoders trained on Kilifi data only with auto-encoders trained on both CIN and Kilifi datasets. Auto-encoders fitted with both CIN and Kilifi data-sets had lower reconstruction loss compared to those fitted using Kilifi data alone. The lowest reconstruction loss was attained by a sparse auto-encoders with latent representation of dimension 128 and L1 regularization.

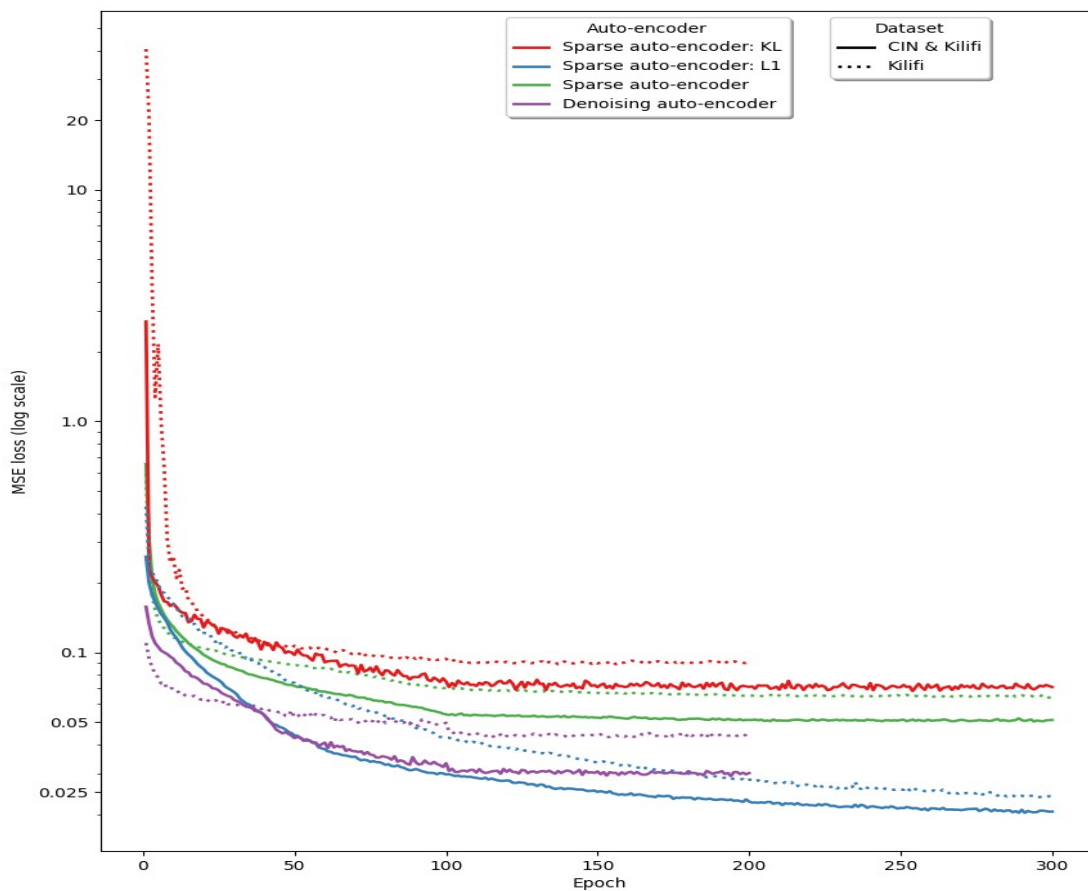


Figure 23: Performance of sparse and denoising auto-encoders on the validation set. The models were trained for a maximum of 300 epochs. Sparse auto-encoders had the lowest reconstruction loss.

4.3.3 Prediction of blood culture results

Table 11 shows the AUCs of models trained to predict blood cultures results. Standard deviation of the performance measures were derived using 5-fold cross validation. Six models were fitted for each experiment with varying amount of data (5%, 10%, 30%, 50%, 70%, and 100%) corresponding to training dataset of sizes (1,633, 3,267, 9,800, 16,336, 22,870, and 32,672 respectively). The test dataset had 8,168 observations regardless of the size of training dataset.

The AUCs of the baseline logistic regression model ranged between 0.666 for model trained using 5% of the training data and 0.705 for model trained using full training dataset. Self-training model based on logistic and linear regression had slightly higher AUCs than baseline logistic regression for models trained using smaller datasets but identical results for larger datasets. However, the differences in performance for small dataset sizes were within the margin of error. Models based on MLPs had significantly lower AUCs when trained on smaller datasets compared to models based on logistic/linear regression. On the other hand, all model based on MLPs had higher AUCs compared to models based on logistic/linear regression when the full training dataset was used but the difference was within the margin of error. Results of accuracy, F1, precision, recall and specificity and presented in appendix B.

Table 11: AUCs of all models predicting blood culture results

		Proportion of labelled data used					
		0.05	0.1	0.3	0.5	0.7	1
Model	datasets						
Logistic (Baseline)	Kilifi	0.666±	0.679±	0.698±	0.701±	0.704±	0.705±
		0.019	0.018	0.014	0.016	0.015	0.015
Logistic: Self Training	Kilifi	0.663±	0.676±	0.694±	0.699±	0.704±	0.705±
		0.019	0.020	0.017	0.017	0.016	0.015
	Kilifi & CIN	0.673±	0.685±	0.697±	0.702±	0.704±	0.705±
		0.024	0.022	0.016	0.015	0.016	0.015
MLP:Denoising auto-encoder	Kilifi	0.621±	0.674±	0.693±	0.700±	0.709±	0.710±
		0.017	0.014	0.018	0.017	0.016	0.015
	Kilifi & CIN	0.623±	0.669±	0.690±	0.702±	0.704±	0.706±
		0.040	0.020	0.014	0.012	0.013	0.014
MLP:Random initialization	Kilifi	0.622±	0.664±	0.690±	0.705±	0.705±	0.710±
		0.036	0.015	0.017	0.011	0.016	0.012
MLP: Sparse auto-encoder	Kilifi	0.631±	0.670±	0.692±	0.699±	0.701±	0.710±

		0.026	0.013	0.014	0.011	0.015	0.013
	Kilifi & CIN	0.594± 0.037	0.655± 0.017	0.696± 0.008	0.702± 0.013	0.703± 0.012	0.709± 0.012
MLP:Sparse auto-encoder (KL)	Kilifi	0.602± 0.019	0.655± 0.021	0.690± 0.009	0.703± 0.017	0.703± 0.016	0.712± 0.012
	Kilifi & CIN	0.605± 0.035	0.660± 0.009	0.692± 0.011	0.697± 0.015	0.705± 0.012	0.711± 0.012
MLP:Sparse auto-encoder (L1)	Kilifi	0.597± 0.036	0.664± 0.023	0.689± 0.016	0.697± 0.018	0.706± 0.015	0.708± 0.012
	Kilifi & CIN	0.636± 0.018	0.656± 0.011	0.688± 0.007	0.699± 0.016	0.704± 0.013	0.707± 0.014

There were large disparities in performance of MLPs on the test data compared to validation data for models trained using small datasets suggesting that the models were over-fitting to the validation set. Figure 24 shows performance of MLPs for predicting blood culture results models on the validation data. The validation AUC ranged between 0.57 and 0.7 for models fitted using 5% of training data while the validation AUCs of models trained using the full training data-set were almost identical. Furthermore, the performance of MLPs trained on all training data was similar for both validation and test data.

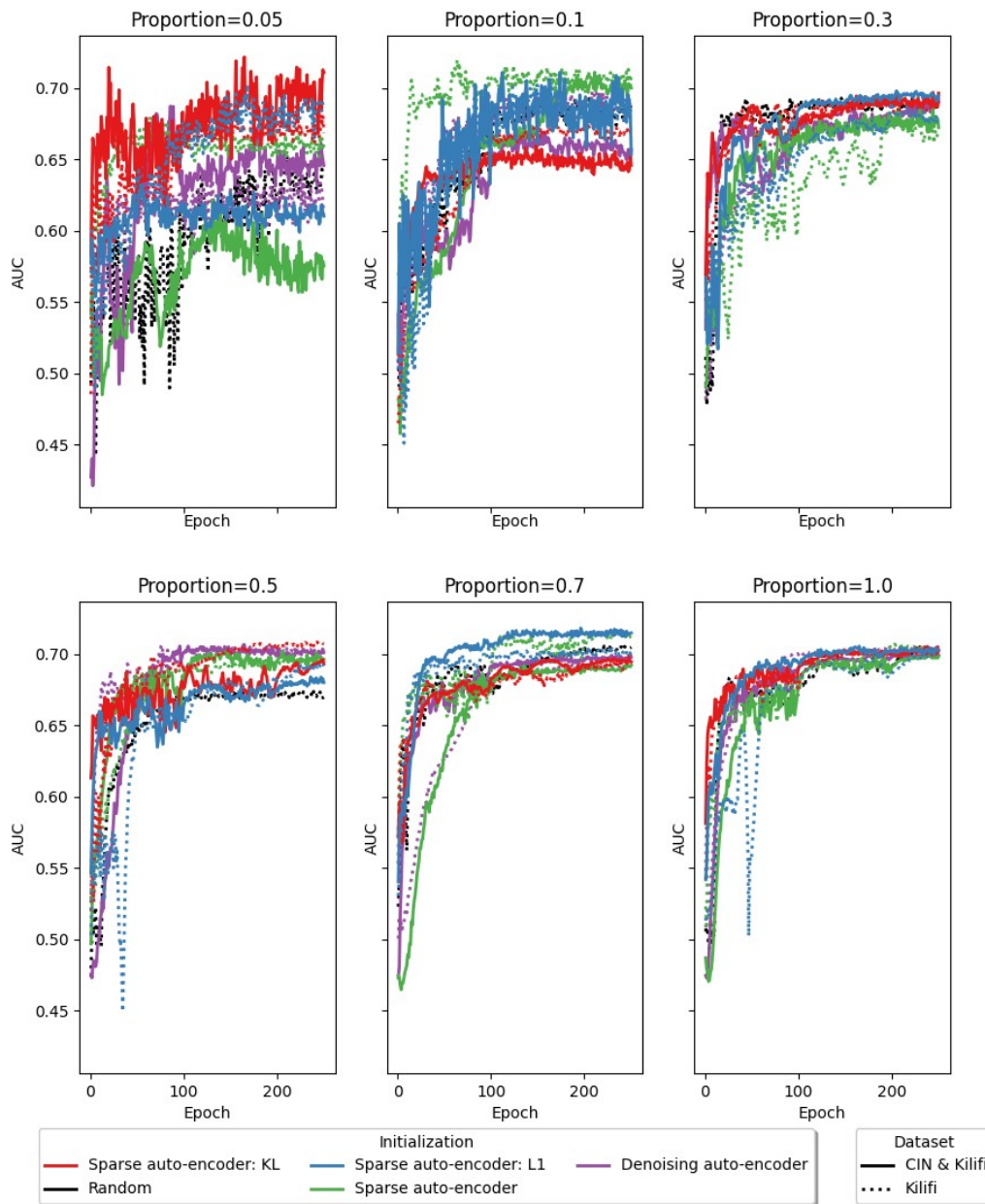


Figure 24: Performance of MLPs with different initialization schemes on the validation data during model training by proportion of labelled data used. There was large variation in performance of MLPs trained using smaller labelled data.

Table 12 shows tests of association between model performance and both data size and whether the unlabelled CIN data was leveraged using transfer learning. Two sets of linear regression models were fitted. One set of models regressed whether or not CIN data was used against each performance metric while the second set of models regressed log of proportion of labelled data used against each performance metric. There was no evidence of association between whether CIN data was used with any of the metrics. On the other hand, data size was positively correlated with AUC and F1 (p-value <0.001 and 0.011 respectively). A unit increase in log of proportion of training data increased AUC and F1 score by 0.025 and 0.006 respectively. Further inspection of effect of data size on AUC showed evidence of effect modification by whether the classification model for predicting bacteremia was based on MLP or linear model (interaction p-value <0.001). A unit increase in log of proportion of training data used increased AUC by 0.013 for linear models and 0.03 for MLPs.

Table 12: Effect of data size and sources of data used on model performance. Coefficients and p-values are obtained using linear regression

Metric	CIN data-set included using transfer learning		Data size	
	Coefficient	p-value	Coefficient	p-value
AUC	-0.001	0.860	0.025	<0.001
Accuracy	-0.085	0.078	0.007	0.748
F1	-0.008	0.127	0.006	0.011
Precision	-0.005	0.083	0.002	0.099
Recall	0.060	0.094	0.028	0.083
Specificity	-0.091	0.078	0.006	0.792

5 Discussion

5.1 Analysis of bio-signals

Parameter based transfer learning improved classification of raw PPG signals using end-to-end deep learning models. The end-to-end deep learning model had auc of 0.80 when initialized using weights from SSL model trained with the entire PPG dataset, 0.77 when initialized using SSL model trained on labelled data only, and 0.73 when initialized randomly. Therefore, initializing end-to-end models using weights from SSL models may still be beneficial when all PPG signals are labelled. Moreover, features extracted using the SSL models could be used as predictors of hospitalization using logistic regression. Features extracted using the SSL model trained on both labelled and unlabelled PPG signals were less predictive of hospitalization compared to features extracted using SSL model trained on labelled PPG signals only (AUC 0.80 vs 0.83). However, features extracted using SSL models trained on both labelled and unlabelled data had higher performance when combine with clinical features (AUC 0.89 vs 0.87). Furthermore, SSL models trained with larger PPG dataset (both labelled and unlabelled) learn better representations producing features that are more predictive of heart rate, respiratory rate and SpO₂, suggesting that unlabelled data is beneficial to transfer learning.

The results of this study agree with a previous study that showed that PPG signal are predictive of hospitalization and may therefore be used to classify patients according to severity of illness (Garde et al., 2016). The study used logistic regression model trained on hand-crafted features extracted from PPG signals using signal decomposition techniques, and achieved an AUC of 0.75. Such hand-crafted features required extensive domain knowledge in signal processing hindering their application on novel tasks. In this study, we were able to achieve better model performance using automatic feature learning and a smaller labelled dataset (1,031 vs 3,374). Features extracted from PPG signals using SSL performed poorer than clinical features (AUC 0.86 vs 0.83). However, most clinical signs and symptoms are subject and required effort to collect compared to the objective measurements of a pulse oximeter.

SSL models were trained to classify whether two PPG segments are obtained from the same patient as a pre-text task. Such a pretext task could learn useful representations of PPG signals if the underlying physiological parameters contained in the PPG signals do not change much within a

short period of time, and the differences between segments from the same patient are due to motion artifacts and other sources of noise. Dimensionality reduction of PPG signals using PCA did not extract features that were predictive of the outcomes. Poor performance of features extracted using PCA might be explained by the model being a linear transformation of the inputs and therefore unable to learn the complex structure of PPG signals. In contrast, deep learning models like the SSL models are universal function approximators capable of learning non-linear functions of the inputs (Hornik et al., 1989).

The manufacturer of the pulse oximeter does not provide information on any pre-processing applied to the PPG signals (closed source software). Such pre-processing may be beneficial for measurements of heart rate and SPO₂ but may lose information beneficial to prediction of hospitalization. In addition, PPG signals collected using pulse oximeters from a different manufacturer may have different pre-processing steps hindering generalization of fitted models. However, the models fitted here do not require knowledge on signal pre-processing to be applicable.

This research demonstrates that transfer learning using self-supervised learning can be successfully applied to bio-signals. While, self-supervised learning has been applied widely for natural images, the type of pre-text tasks used in images may not be suitable for bio-signals. A novel pre-text task involving classifying whether two segments of PPG signals belong to the same patient is proposed and evaluated. We show that the proposed pre-text task can extract features from raw PPG signals, eliminating the need for feature engineering using traditional signal processing techniques. The deep learning models proposed here may also be more advantageous than traditional signal processing techniques if the signals are noisy (Yoon et al., 2019). PPG sensors are cheap and miniaturized, allowing development of cheap and non-invasive wearable sensors. The methods proposed in this research could enhance development of novel applications using such sensors for continuous and non-invasive monitors for various physiological parameters involving respiratory and cardiovascular systems.

In conclusion, we have shown that SSL models can extract features from raw PPG signals and provide better initializations for end-to-end deep learning models. In addition, features extracted from PPG signals using SSL models can be used for various classification and regression tasks

using linear models. Furthermore, SSL models can incorporate unlabelled bio-signals to improve extracted features thus enhancing models fitted using small labelled datasets.

5.2 Analysis of chest radiographs

We evaluated transfer learning using two parameter based and two multi-task learning techniques. Parameter-based transfer learning involved initializing CNNs with weights from a CNN model trained to classify Chest-ray14 dataset (supervised pre-training) and weights from a self-supervised learning (SSL) models (unsupervised pre-training). We also evaluated the utility of multitask learning where models were trained to classify PERCH and Chestray 14 dataset simultaneously. Finally, we proposed a new multi-task learning technique where CNN models are trained to classify CXR images conditional on reader identifier.

The baseline CNN models had an accuracy of 0.59 for models with ResNet18 architecture, 0.57 for ResNet34 and 0.59 for ResNet50. Initializing the CNN models using weights from CNNs trained to classify Chestray-14 CXRs (supervise pre-training) improved the model performance marginally. The accuracy increased to 0.61 for ResNet18, 0.6 for ResNet34 and 0.61 for ResNet50. CNN models initialized using weights from SSL models (unsupervised pre-training) had better performance for SSL model trained on both PERCH and Chestray 14 dataset for ResNet18 and RestNet34 but worse performance for ResNet50. Self-supervised weights of models trained using both PERCH and Chestray-14 improved performance of ResNet18 (Accuracy 0.6 vs 0.58) and ResNet34 (0.6 vs 0.59), but had detrimental effect for ResNet50 (0.59 vs 0.6).

The SSL models were trained to predict whether two CXR images were obtained by applying data augmentation on two different CXR images or applying data augmentation on the same CXR image. A t-SNE visualization of embeddings extracted from CXR images using the SSL models showed that the embeddings were clustered by children's age and not by WHO category or site. Therefore, the SSL model did not learn representations that were useful in predicting WHO categories. It is possible that images of children of different ages had different aspect ratios which the SSL model used as a shortcut whenever the model was presented with CXR images of children with different ages. Effective data augmentation can prevent SSL models from taking such shortcuts

and improve the learned representation, but it is not clear what are the appropriate data augmentation techniques for medical images. We used random resizing, color jitters, image flipping, and affine transformation augmentation procedures which have been shown to work well for natural images (Shorten & Khoshgoftaar, 2019). However, such augmentation may be counter-productive for medical images. For instance, random cropping may exclude the part of the image that contains the pathology of interest. Therefore, more research on data augmentation for medical images is required. Another approach that may prevent the SSL models from taking shortcuts might be mining of hard negative examples (Kalantidis et al., 2020). For a given query image, mining negative examples involves computing the distance between embeddings of the query image and all other images and selecting the image with the smallest distance. The limitation of mining negative examples is that computing the distance between each query example and the rest of the dataset is computationally expensive.

Multi-task learning models trained to simultaneously classify PERCH and Chestray-14 datasets performed either at par or worse than models trained to classify PERCH only and consistently worse than models trained to classify Chestray-14 dataset only. Multi-task learning is thought to improve performance across tasks by exploiting commonality between tasks using shared representation (Caruana, 1997). However, training multi-task learning models successfully is challenging and multi-task models that perform worse than single tasks have been reported elsewhere (Parisotto et al., 2016; Rusu et al., 2015). First, one has to decide how much weight should be placed on each task (Lin et al., 2021). Choosing appropriate weights is especially challenging when the magnitude/scale of the losses for different tasks differ significantly. When the scale of one task is significantly higher than that of the others, the combined loss of all tasks may be dominated by the task with the large loss. In our case, the loss for the model classifying PERCH images accounted for most of the total loss hindering improvement of model classifying Chestray-14 dataset. Second, multi-task learning is beneficial when there is similarity among tasks. Both Chestray-14 and PERCH had classes for consolidation and infiltrates which makes a case for multi-task learning. However, there were 12 other tasks in Chestray-14 that might not be similar to the tasks in PERCH dataset. Yu (2020) hypothesized that the difficulty in optimizing multi-task learning stems from gradients of different tasks conflicting. Gradients are considered to be conflicting if the cosine similarity between gradients of two tasks is negative.

Models with reader embeddings had consistently higher performance compared to corresponding models without reader embeddings. The higher performance was observed regardless of model initialization techniques or architecture. Models with reader embeddings aggregated predictions (one for each reader) to get the final prediction for each CXR image which might explain the increase in performance. The aggregation technique is similar to staking, an ensemble method where predictions from multiple models are aggregated. Model ensembles are on average expected to perform as well as the best model (I. Goodfellow et al., 2016). Unlike stacking, our approach does not require multiple models to be fitted reducing the cost of training the models. However, the model had extra parameters for the embedding layer and the linear layer that projected the reader embeddings to have the same dimensions as image embedding. The increase in parameters was minimal given the large size of the models. ResNet50 had 67,416 additional parameters while ResNet18 and ResNet34 had 16,928 additional parameters each. Given that the models have tens of millions of parameters, the increase in parameter was less than 1%.

The best performing model for classifying PERCH CXR had an Accuracy and AUC of 0.62 and 0.87, respectively. The model had an embedding layer for readers and was initialized using weights from the supervised learning model for classification of Chestray-14 CXR images. The model had higher accuracy for children older than 12 months compared to children aged between 1 and 11 months (accuracy 0.65 vs 0.60). Model accuracy increased rapidly for children aged one to eight months from less 50% to more than 65% and then remained the same for older children suggesting that the model may only be suitable for children older than 8 months. The poor performance for younger children could be explained by lower agreement between readers for that age group and not a limitation of machine learning models. Both model accuracy and human reader agreement increased with age. The low agreement between human readers for younger children is likely due to difficulties in obtaining good quality CXRs in younger children. Low agreement may be indicative of high level of miss-classification noise in the outcome which has been shown to be detrimental to model performance (Nettleton et al., 2010; Pechenizkiy et al., 2006). Model performance ranged between 0.52 and 0.7 across sites. The large variation in model performance across sites could be explained by differences in distribution of labels. South Africa and Zambia, the two sites with accuracy below 0.6, had the lowest proportion of normal CXR images (28% and 31% respectively), and the model was better at classifying normal CXR compared to other categories. Sites that did not have digital CXR machines and scanned analogue images had high accuracy suggesting that machine learning models are useful in setting that lack modern CXR equipment. CXR with both

consolidation and infiltrates were more likely to be misclassified compared to other categories. Forty percent were misclassified as consolidation only while 30% were misclassified as infiltrates only. It is possible that re-framing the outcome as multiple binary outcomes (one outcome per condition) instead of a single outcome with five categories might reduce miss-classification of images with both consolidation and infiltrates.

The best model had an AUC of 0.85 for consolidation, 0.82 for infiltrates, 0.9 for both consolidation and infiltrates, 0.87 for normal and 0.88 for un-interpretable. The accuracy for any consolidation (consolidation or consolidation and infiltrates) was 0.87 while that of any infiltrates was 0.76. The lower predictive performance for infiltrates as compared to other categories has been reported for Chestray-14 dataset (Rajpurkar et al., 2017; X. Wang et al., 2017; Yao et al., 2018). The poor performance in classifying infiltrates is likely due to difficulty in identifying infiltrates from CXR images. Other medical imaging techniques such as CT scans and ultrasound are more accurate for diagnosis of pneumonia compared to CXR images. CT scans have higher accuracy compared to both CXRs and ultrasound but they are expensive and exposes patients to higher levels of radiation making them unsuitable for low income settings (Brenner & Hall, 2007; Syrjälä et al., 1998). On the other hand, ultrasound have been shown to be more accurate in diagnosis on pneumonia compared to CXR images in high-income settings but have not been widely evaluated in low income settings where there is high prevalence of tuberculosis and chronic obstructive pulmonary disease (Amatya et al., 2018; Sippel et al., 2011). Deep learning models have been shown to be highly sensitive and specific in classification of lung ultrasounds for diagnosis on pneumonia (Diaz-Escobar et al., 2021; La Salvia et al., 2021). Therefore, CNN models trained using ultrasound images may still be useful if CXRs are replaced with ultrasound.

The low sensitivity in classifying CXR by human readers may stem from some features of CXR images being too subtle for human readers rather than the information being absent from the CXRs. For instance, it has been shown that machine learning models can predict race using CXR image despite no known features of race on the images (Banerjee et al., 2021). Machine learning models for classifying CXR images have relied on labels derived from human readers interpreting the CXRs. Such models may be negatively affected by misclassified training CXRs if human readers are unable to detect all features relevant in the CXRs. An alternative means of obtaining labels for CXR images might be to use labels derived from CT scans to train models for classifying CXR images. In

such a study, images would be obtained using both CT scans and CXR. Machine learning models would then be trained to classify CXR images using the labels derived from CT scans. The CT scans would not be required during deployments of the models and the model can be used in low income settings where CXR equipment are widely available.

Our approach of including reader embeddings could be considered as a form of multi-task learning. However, unlike conventional multi-task learning where each forward pass would have produced multiple predictions depending on number of readers, our model was provided with reader identifiers along side CXR images. The advantage of our approach is that it can be used when not all readers annotated each image, which was the case in our dataset. In addition, the difficulty of finding optimal weights for different task that arise in conventional multi-task learning was not present in our approach.

Model size did not influence model performance. The total number of parameters for the baseline model was 11,179,077 for ResNet18, 21,287,237 for ResNet34 and 23,518,277 for ResNet50. Raghu (2019) showed that small models perform just as well as large one for retinal fundus photographs and CXR medical images but performed worse on natural images in ImageNet dataset. However, initializing the weights using weights from model classifying ImageNet dataset (supervised pretraining) improved performance of large models marginally but did not have an effect on small models. Moreover, the benefit of supervised pre-training for larger models was more pronounced in small data regiments (smaller dataset of medical images). The differences in effect of model size on performance observed between natural images and medical images may be suggestive of need to develop neural network architectures specific to medical images. Unlike natural images where edges are important in recognizing an object, classification of medical images may be more reliant on the image texture. Transformer based deep learning architectures have been suggested as alternative for CNNs, but such models require large dataset that are often lacking in medical settings (Matsoukas et al., 2021). Transformer based model may also be more advantageous for medical images because their attention maps offer a means of visualizing decision making mechanism of the model improving model explain-ability, which is essential in adoption of machine learning in medical applications.

The baseline models were initialized using weights from models trained to classify ImageNet, a database on natural images such as airplanes, wildlife, dogs, etc. Therefore, we expected models initialized using tasks derived from CXR images to perform better than models initialized using weights from tasks involving natural images. However, the improvement in models classifying PERCH CXR images as a result of supervised and unsupervised pre-training was modest at best. It is possible that the PERCH dataset was too large to benefit significantly from supervised and unsupervised pre-training.

5.3 Analysis of clinical data

We compared performance of models based on multi-layer perceptrons (MLPs) with linear models in predicting blood culture results given clinical signs and symptoms. We assessed the utility of incorporating unlabelled data using transfer learning in improving performance of MLPs and linear models fitted with varying amount of labelled data. We varied the amount of labelled data used to train the models to assess the effect of transfer learning on dataset size. We used training datasets of approximately 1,633, 3,267, 9,800, 16,336, 22,870, and 32,672 observations, accounting for 5%, 10%, 30%, 50%, 70% and 100% of labelled data respectively. For linear models, we compared a logistic regression model with self-training model based of logistic and linear regression. For MLPs, we compared MLPs initialized randomly with MLP initialized using weights of sparse and denoising auto-encoders.

Transfer learning and semi-supervised learning did not significantly improve prediction of bacteraemia using clinical signs and symptoms. Semi-supervised learning models using self-training and logistic regression had marginally higher AUCs compared to the baseline logistic regression for models trained using 5% and 10% of the data (0.673 vs 0.666 and 0.685 vs 0.679, respectively). The baseline logistic regression had slightly higher AUC than the self-training model based on logistic regression when the models were trained using 30% of the data (0.698 vs 0.697). Models based on self-training and logistic regression had identical performance when trained with 70% and 100% of the data. Models based on MLPs had higher performance compared to models based on logistic regression (baseline and self-training) for models trained with 50% of the data or more. MLP model initialized randomly had the best performance for models trained using 50% of

the dataset (AUC=0.705), while MLPs initialized using weights from the L1 and KL regularized sparse auto-encoders had the best performance for models trained using 70% and 100% of the training data (AUC = 0.706 and 0.712, respectively). In general, models based on logistic regression were more predictive of bacteraemia compared to models based on MLP when small training datasets were used. The AUCs of logistic regression models ranged between 0.663 and 0.673 for models trained on 5% of training data compared to AUCs ranging between 0.597 and 0.636 for MLPs trained using similar sized training dataset. However, models based on MLPs had marginally higher AUCs than model based on logistic regression when 100% of training data was used. The best MLP model had an AUC of 0.712 compared with AUC of 0.705 for logistic regression models. However, the differences in performance were within the margin of error.

Models based on MLPs had challenges classifying smaller datasets because the models were overfitting to the validation data. There was high disparity between test and validation AUCs for models trained using 5% of labelled data. On the other hand, the disparity between test and validation AUCs was minimal for models trained using 100% of labelled dataset. The validation AUCs for models trained using 5% of the training dataset ranged between 0.57 and 0.7 compared to test AUCs that ranged between 0.597 and 0.636. The disparity in AUCs between validation and test dataset could be explained by the validation data being much smaller than the test set. We used the same test data for all experiments regardless of size of training dataset while the validation set was set to 20% of the training dataset. The standard deviation of test AUCs obtained using 5-fold cross-validation had a range of 0.017 – 0.04 for models trained using 5% of the training dataset and 0.012-0.015 for models trained using 100% of training datasets. Therefore, the standard deviations for the test dataset were small for models trained using small and large dataset indicating that the test dataset was large enough to evaluate the performance of all models with high precision. It is possible that ASHA, the hyper-parameter selection technique used may not be suitable for small datasets. ASHA was implemented by running 500 trials in parallel with each trial consisting of a hyper-parameter configuration sampled from the hyper-parameter search space. Given the large number of trials, it is possible that one trial might perform well on the small validation data by chance and fail to generalize to the test dataset.

There was strong evidence that AUC increased by size of training data. A unit increase in log of proportion of training data used increased AUC by 0.025 (p-value <0.001). Moreover, the relationship between data size and AUC was different for MLP models compared to linear models. A unit increase in log of proportion of data used increased AUC by 0.013 for logistic regression

models and 0.03 for MLPs (p -value < 0.001). If the observed differences between logistic regression and MLP holds for datasets larger than the labelled dataset ($n=32,672$), then the performance of MLPs is expected to be significantly better than that of linear models for larger datasets.

The challenges of MLPs over-fitting on small datasets might be overcome by using Bayesian networks (Charnock et al., 2020; Jospin et al., 2022). Traditional neural networks tend to be over-confident on data points outside the distribution of the training dataset, and models fitted using smaller datasets are more likely to encounter out-of-distribution observations during testing (Guo et al., 2017; Nixon et al., 2020). Neural networks can be converted into Bayesian neural networks by having stochastic weights, such that weights are represented as distributions instead of point estimates. Consequently, Bayesian neural networks can give an estimate of the confidence of a given prediction. Having neural networks that provide precision for a given prediction is useful in medical diagnosis because the model can flag instances where human intervention is needed, enhancing safety (Amodei et al., 2016). Bayesian neural networks are also well suited for small datasets because they don't require cross-validation for hyper-parameter optimization, but can instead average out the hyper-parameters. During hyper-parameter optimization, 20% of training data was put aside for model validation, reducing the size of the training dataset. Test AUC increased approximately linearly with the log of the proportion of the labelled dataset, suggesting that the price of holding out the validation dataset on model performance was steeper for smaller datasets.

We used denoising and sparse auto-encoders for SSL models instead of contrastive learning models used for CXR and PPG signals. Auto-encoders model the joint distribution of the inputs which can be difficult when different variables have different units. Therefore, continuous variables such as temperature and oxygen saturation were scaled to have a range of -1 to 1 using min-max scaling. The self-supervised learning methods used relied on the ability of the model to reconstruct the inputs. Such methods may have challenges with datasets consisting of clinical signs and symptoms where the variables have high class imbalance. That is because the models can take shortcuts and learn to minimize the loss during training by simply predicting the majority class for each input variable, and fail to learn useful representations of the inputs. Data augmentation techniques such as adding random noise and randomly switching the class of an input variable might not be sufficient to prevent such shortcuts.

The overall performance of all models may be too low to have clinical utility. While an AUC of 0.7 is indicative of the models being able to rank patients according to risk of bacteremia, the AUC is too low to be useful in making decisions on clinical management of patients. The models might still be useful for determining the prevalence of bacteremia in various settings, but external validation using data from other hospitals would be required to assess generalizability. This study demonstrates that the low performance in models for classifying bacteremia is not due to use of simple linear models. MLP models which have been shown to work well in many machine learning applications performed either worse or at par with linear models.

Results of blood culture results were used as an indicator of bacteremia. Blood cultures have low sensitivity and specificity because of difficulties in culturing the micro-organism (Bloos et al., 2012; Westh et al., 2009). The poor sensitivity acts as noise in the outcome/label which has been shown to hinder training of machine learning models. Polymerase chain reaction (PCR) tests are highly sensitive alternative tests for bacteremia that could reduce noise in the labels. However, the high cost of PCR tests hinders its use in public hospitals.

The use of clinical signs and symptoms as predictors of bacteremia has several limitations. First, multiple diseases with different etiologies/causes have similar presentation. For instance, severe malaria which is caused by protozoa and bacterial meningitis have overlapping clinical signs and symptoms limiting the utility of the clinical signs. In addition, assessment of clinical symptoms is subjective and may vary from one clinician to another which introduces measurement noise in the predictors. Su (2011) was able to predict bacteremia using logistic regression with an AUC of 0.854 by incorporating laboratory measured biomarkers for lymphocyte counts (Lymphopenia), C-reactive proteins, and procalcitonin. Jaimes (2004) was able to predict bacteremia with an AUC of 0.718 using leukocyte count and other clinical signs and symptoms as predictors. Such models with predictors that require well equipped laboratories may not be feasible in low income settings. An alternative source of predictors for bacteremia might be near infrared (NIR) spectrometry (Ciurczak & Igne, 2014). Application of NIR spectrometry in medicine includes measurements of blood glucose, oxygen, and haemoglobin (Sakudo, 2016). Recent advances in NIR spectroscopy have seen the devices become portable and miniaturized, which has improved their usability (Alcalà et al., 2013). In addition, NIR spectrometers like pulse oximeters are non-invasive and can be used by the

bedside without requiring presence of well equipped laboratories. Deep learning model work well with high dimensional data, and hence are ideal for developing models using NIR signals. Furthermore, CNNs might exploit the spatial aspects of NIR to achieve higher performance.

Only 4 % of blood cultures were positive. Therefore, there was high imbalance in the outcome which made model training difficult. We weighted the loss using the inverse of class frequencies for logistic regression and up sampled the minority class for MLPs. Consequently, metrics for sensitivity and specificity in logistic regression and MLPs are not comparable. Having only a few observations with positive blood cultures also made model evaluation difficult. The class imbalance informed the choice of AUC - which measure of how well the classifier can rank any two observation according to the risk of the outcome - as the evaluation metric.

Deep learning methods don't perform as well as tree based models such as boosted trees on tabular data (Shwartz-Ziv & Armon, 2022). However, incorporating unlabelled data while training tree based models is difficult except by using self-training. Consequently, successful transfer learning methods based on deep learning may remain sub-optimal until significant progress has been made in deep learning architectures for tabular data.

6 Conclusion and Recommendations

This study addresses the application of transfer learning to medical data. Most of the previous efforts in transfer learning have concentrated on either natural images or text. We applied transfer learning to the three types of datasets available for diagnostic and prognostic models in low income setting: medical images, bio-signals and tabular data. There are significant differences between medical images and natural images that might affect suitability of transfer learning methods that have been applied to natural images. For instance, data augmentation methods such as cropping that used for natural images might be inappropriate for medical images.

The study makes significant contribution to transfer learning for medical data. First, it introduces a pretext task for bio-signals where a model is trained to classify whether two bio-signal segments originate from the same patient. Secondly, the study demonstrates improvement in classification of medical images when the pre-text task utilized another data-set of medical images instead of natural images. Lastly, the study confirmed previous research on the limited utility of clinical signs and symptoms in developing diagnostic models (Christodoulou et al., 2019). The study shows that the poor performance of models classifying bacteremia using signs and symptoms is not due to using linear models that have limited complexity.

Transfer learning techniques improved models fitted on pulse oximeter signals and CXRs but had little effect on models fitted using clinical signs and symptoms. The observed differences in improvement might be due use of deep learning, whose application on heterogeneous tabular data such as clinical signs and symptoms is often challenging (Borisov et al., 2022). The hidden layers of deep learning models learn latent representations of the inputs, making deep learning models ideal for transfer learning when combined with unsupervised/self-supervised learning. Furthermore, deep learning models are currently the state of art for developing machine learning models using highly structured data such as images and signals.

There were limitations in the study relating to the scope of models used due to limitations in computing resources. Deep learning models are computationally intensive and require computers

fitted with accelerators such as graphical processing units (GPUs) or tensor processing units (TPUs). The models reported in this study were fitted using three desktops, each having 24GBs random access memory (RAM) and an NVIDIA 1080Ti GPU (11GB memory). The size of memory in the GPUs limited the size of deep learning models and batch sizes that could be processed at a time. Consequently, we could not train models with large batch size, which has been shown to be beneficial for contrastive learning (He et al., 2020). Furthermore, limitations in computing resources hindered exploration of alternative deep learning architectures such as transformers for medical images and recurrent neural networks for bio-signals. Extensive hyper-parameter search requires fitting many models with varying hyper-parameter configurations which is computationally expensive. As a result, we used ASHA algorithm instead of grid search hyper-parameter selection algorithm for the deep learning models. Unlike grid search algorithm where all hyper-parameter configurations are tested, ASHA algorithm samples a fixed number of hyper-parameter configurations, and stops poorly performing hyper-parameter configurations early. While early stopping reduces the amount of computation required to fit the models, there is a risk of stopping well-performing hyper-parameter configurations prematurely. For instance, models trained using small learning rates might be stopped early despite the possibility of having lower loss at the end of training because their loss decreases too slowly and consequently have high loss during the initial epochs.

Each data type – medical images, bio-signals, and tabular data – was represented by a single dataset, and only one classification task was tested for each data type. Consequently, the findings of this study may not generalize to all tasks for any data type. There are a variety of other medical images such as MRI scans, CT scans, and ultra-sounds which might have different modalities. For instance, MRI and CT scans are 3-dimensional images while CXR images are 2-dimensional. In addition, transfer-ability may depend on how closely related the tasks are, and therefore be task dependent. It is also possible that there are other large unlabelled datasets that are better suited for tasks explored in this study. However, this study provides evidence for the benefits of transfer learning in the development of diagnostic and prognostic models and serves as a baseline for transfer learning for prognostic and diagnostic models.

Further research is needed in the development of appropriate data augmentation for medical datasets. Data augmentation is central to self-supervised learning using contrastive learning and denoising

auto-encoders. Data augmentation is also key to deep learning models in general, and is used widely to prevent over-fitting. Therefore, it is essential to develop data augmentation techniques specific for various medical data types. Such techniques might rely on domain knowledge about sources of noise for various data types. For instance, PPG signals contain motion artifacts and interference from ambient light. As such, data augmentation techniques that can add motion and ambient light noise to signals may improve transfer learning for PPG signal datasets. It might be worthwhile to explore generative adversarial networks for data augmentation given that developing a model for the noise generating process from domain knowledge might be difficult (Sandfort et al., 2019).

There is limited research on the impact of diagnostic and prognostic models on clinical outcomes. Fitting the models is only the first step and more research is needed for the implementation of the models in clinical practice. Interventions involving technology might be challenging in low resource settings which might render good performing models unusable (McCool et al., 2020).

In conclusion, performance of diagnostic and prognostic models can be improved using transfer learning. However, the improvement in performance of the models might depend on the task and data type used. For data types that deep learning models excel such as medical images and bio-signals, transfer learning models based on deep learning are beneficial. On the other hand, simpler linear models perform better than deep learning for smaller tabular datasets.

References

- Adamson, A. S., & Smith, A. (2018). Machine Learning and Health Care Disparities in Dermatology. *JAMA Dermatology*, *154*(11), 1247–1248.
<https://doi.org/10.1001/jamadermatol.2018.2348>
- Agrawala, A. (1970). Learning with a probabilistic teacher. *IEEE Transactions on Information Theory*, *16*(4), 373–379. <https://doi.org/10.1109/TIT.1970.1054472>
- Alcalà, M., Blanco, M., Moyano, D., Broad, N. W., O'Brien, N., Friedrich, D., Pfeifer, F., & Siesler, H. W. (2013). Qualitative and Quantitative Pharmaceutical Analysis with a Novel Hand-Held Miniature near Infrared Spectrometer. *Journal of Near Infrared Spectroscopy*, *21*(6), 445–457. <https://doi.org/10.1255/jnirs.1084>
- Althnian, A., AlSaeed, D., Al-Baity, H., Samha, A., Dris, A. B., Alzakari, N., Abou Elwafa, A., & Kurdi, H. (2021). Impact of Dataset Size on Classification Performance: An Empirical Evaluation in the Medical Domain. *Applied Sciences*, *11*(2), Article 2.
<https://doi.org/10.3390/app11020796>
- Amatya, Y., Rupp, J., Russell, F. M., Saunders, J., Bales, B., & House, D. R. (2018). Diagnostic use of lung ultrasound compared to chest radiograph for suspected pneumonia in a resource-limited setting. *International Journal of Emergency Medicine*, *11*, 8.
<https://doi.org/10.1186/s12245-018-0170-2>
- Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mané, D. (2016). *Concrete Problems in AI Safety*.
- Arazo, E., Ortego, D., Albert, P., O'Connor, N. E., & McGuinness, K. (2020). Pseudo-Labeling and Confirmation Bias in Deep Semi-Supervised Learning. *ArXiv:1908.02983 [Cs]*.
<http://arxiv.org/abs/1908.02983>
- Avola, D., Bacciu, A., Cinque, L., Fagioli, A., Marini, M. R., & Taiello, R. (2021). *Study on Transfer Learning Capabilities for Pneumonia Classification in Chest-X-Rays Image*.
<https://doi.org/10.48550/arXiv.2110.02780>
- Awad, M., & Khanna, R. (2015). Machine Learning. In: Efficient Learning Machines. In M. Awad & R. Khanna (Eds.), *Efficient Learning Machines: Theories, Concepts, and Applications for*

Engineers and System Designers (pp. 1–18). Apress. https://doi.org/10.1007/978-1-4302-5990-9_1

- Ayukekbong, J. A., Ntemgwa, M., & Atabe, A. N. (2017). The threat of antimicrobial resistance in developing countries: Causes and control strategies. *Antimicrobial Resistance & Infection Control*, 6(1), 47. <https://doi.org/10.1186/s13756-017-0208-x>
- Balestriero, R., Pesenti, J., & LeCun, Y. (2021). Learning in High Dimension Always Amounts to Extrapolation. *ArXiv:2110.09485 [Cs]*. <http://arxiv.org/abs/2110.09485>
- Banerjee, I., Bhimireddy, A. R., Burns, J. L., Celi, L. A., Chen, L.-C., Correa, R., Dullerud, N., Ghassemi, M., Huang, S.-C., Kuo, P.-C., Lungren, M. P., Palmer, L., Price, B. J., Purkayastha, S., Pyrros, A., Oakden-Rayner, L., Okechukwu, C., Seyyed-Kalantari, L., Trivedi, H., ... Gichoya, J. W. (2021). Reading Race: AI Recognises Patient's Racial Identity In Medical Images. *ArXiv:2107.10356 [Cs, Eess]*. <http://arxiv.org/abs/2107.10356>
- Barragán-Montero, A., Javaid, U., Valdés, G., Nguyen, D., Desbordes, P., Macq, B., Willems, S., Vandewinckele, L., Holmström, M., Löfman, F., Michiels, S., Souris, K., Sterpin, E., & Lee, J. A. (2021). Artificial intelligence and machine learning for medical imaging: A technology review. *Physica Medica*, 83, 242–256. <https://doi.org/10.1016/j.ejmp.2021.04.016>
- Baxter, J. (1997). A Bayesian/Information Theoretic Model of Learning to Learn via Multiple Task Sampling. *Machine Learning*, 28(1), 7–39. <https://doi.org/10.1023/A:1007327622663>
- Bellman, R. (1961). *Adaptive Control Processes*.
<https://press.princeton.edu/books/hardcover/9780691652214/adaptive-control-processes>
- Bengio, Y., Courville, A., & Vincent, P. (2014). Representation Learning: A Review and New Perspectives. *ArXiv:1206.5538 [Cs]*. <http://arxiv.org/abs/1206.5538>
- Bengio, Y., Delalleau, O., & Le Roux, N. (2006). Label Propagation and Quadratic Criterion. In *Semi-Supervised Learning* (Semi-Supervised Learning, pp. 193–216). MIT Press.
<https://www.microsoft.com/en-us/research/publication/label-propagation-and-quadratic-criterion/>
- Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P., Horsfall, P., & Goodman, N. D. (2018). Pyro: Deep Universal Probabilistic Programming. *Journal of Machine Learning Research*.

- Bloos, F., Sachse, S., Kortgen, A., Pletz, M. W., Lehmann, M., Straube, E., Riedemann, N. C., Reinhart, K., & Bauer, M. (2012). Evaluation of a Polymerase Chain Reaction Assay for Pathogen Detection in Septic Patients under Routine Condition: An Observational Study. *PLOS ONE*, 7(9), e46003. <https://doi.org/10.1371/journal.pone.0046003>
- Blum, A., & Mitchell, T. (1998). Combining Labeled and Unlabeled Data with Co-Training. *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, 92–100. <https://doi.org/10.1145/279943.279962>
- Borisov, V., Leemann, T., Seßler, K., Haug, J., Pawelczyk, M., & Kasneci, G. (2022). *Deep Neural Networks and Tabular Data: A Survey* (arXiv:2110.01889). arXiv. <https://doi.org/10.48550/arXiv.2110.01889>
- Brenner, D. J., & Hall, E. J. (2007). Computed tomography—An increasing source of radiation exposure. *The New England Journal of Medicine*, 357(22), 2277–2284. <https://doi.org/10.1056/NEJMra072149>
- Brinker, K. (2003). Incorporating Diversity in Active Learning with Support Vector Machines. *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, 59–66.
- Brownstein, J. S., Freifeld, C. C., & Madoff, L. C. (2009). Digital Disease Detection—Harnessing the Web for Public Health Surveillance. *New England Journal of Medicine*, 360(21), 2153–2157. <https://doi.org/10.1056/NEJMp0900702>
- Budd, S., Robinson, E. C., & Kainz, B. (2021). A Survey on Active Learning and Human-in-the-Loop Deep Learning for Medical Image Analysis. *Medical Image Analysis*, 71, 102062. <https://doi.org/10.1016/j.media.2021.102062>
- Carracedo-Reboredo, P., Liñares-Blanco, J., Rodríguez-Fernández, N., Cedrón, F., Novoa, F. J., Carballal, A., Maojo, V., Pazos, A., & Fernandez-Lozano, C. (2021). A review on machine learning approaches and trends in drug discovery. *Computational and Structural Biotechnology Journal*, 19, 4538–4558. <https://doi.org/10.1016/j.csbj.2021.08.011>
- Caruana, R. (1993). Multitask Learning: A Knowledge-Based Source of Inductive Bias. *Proceedings of the Tenth International Conference on Machine Learning*, 41–48.

- Caruana, R. (1997). Multitask Learning. *Machine Learning*, 28(1), 41–75.
<https://doi.org/10.1023/A:1007379606734>
- Carvalho, C. M., Polson, N. G., & Scott, J. G. (2009). Handling Sparsity via the Horseshoe. In D. van Dyk & M. Welling (Eds.), *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics* (Vol. 5, pp. 73–80). PMLR.
<https://proceedings.mlr.press/v5/carvalho09a.html>
- Castiglioni, I., Rundo, L., Codari, M., Di Leo, G., Salvatore, C., Interlenghi, M., Gallivanone, F., Cozzi, A., D’Amico, N. C., & Sardanelli, F. (2021). AI applications to medical images: From machine learning to deep learning. *Physica Medica*, 83, 9–24.
<https://doi.org/10.1016/j.ejmp.2021.02.006>
- Chapelle, O., Schölkopf, B., & Zien, A. (2006). *Semi-Supervised Learning. Adaptive Computation and Machine Learning series*. The MIT Press.
- Chapelle, O., Scholkopf, B., & Zien, A. (2009). Semi-supervised learning (chapelle, o. Et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3), 542–542.
- Charnock, T., Perreault-Levasseur, L., & Lanusse, F. (2020). *Bayesian Neural Networks*.
<https://arxiv.org/abs/2006.01490v2>
- Chattopadhyay, R., Sun, Q., Fan, W., Davidson, I., Panchanathan, S., & Ye, J. (2012). Multisource Domain Adaptation and Its Application to Early Detection of Fatigue. *ACM Trans. Knowl. Discov. Data*, 6(4). <https://doi.org/10.1145/2382577.2382582>
- Chen, L. (2020). Overview of clinical prediction models. *Annals of Translational Medicine*, 8(4), 71. <https://doi.org/10.21037/atm.2019.11.121>
- Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A Simple Framework for Contrastive Learning of Visual Representations. *ArXiv:2002.05709 [Cs, Stat]*.
<http://arxiv.org/abs/2002.05709>
- Chen, Z., Badrinarayanan, V., Lee, C.-Y., & Rabinovich, A. (2018). GradNorm: Gradient Normalization for Adaptive Loss Balancing in Deep Multitask Networks. *ArXiv:1711.02257 [Cs]*. <http://arxiv.org/abs/1711.02257>
- Cherian, T., Mulholland, E. K., Carlin, J. B., Ostensen, H., Amin, R., de Campo, M., Greenberg, D., Lagos, R., Lucero, M., Madhi, S. A., O’Brien, K. L., Obaro, S., & Steinhoff, M. C. (2005).

Standardized interpretation of paediatric chest radiographs for the diagnosis of pneumonia in epidemiological studies. *Bulletin of the World Health Organization*, 83(5), 353–359.

<https://doi.org/S0042-96862005000500011>

Christodoulou, E., Ma, J., Collins, G. S., Steyerberg, E. W., Verbakel, J. Y., & Van Calster, B. (2019). A systematic review shows no performance benefit of machine learning over logistic regression for clinical prediction models. *Journal of Clinical Epidemiology*, 110, 12–22.

<https://doi.org/10.1016/j.jclinepi.2019.02.004>

Ciurczak, E. W., & Igne, B. (2014). *Pharmaceutical and medical applications of near-infrared spectroscopy*. CRC Press.

Cohn, D. A., Ghahramani, Z., & Jordan, M. I. (1996). Active Learning with Statistical Models.

ArXiv:Cs/9603104. <http://arxiv.org/abs/cs/9603104>

Collins, F. S., & Varmus, H. (2015). A new initiative on precision medicine. *The New England Journal of Medicine*, 372(9), 793–795. <https://doi.org/10.1056/NEJMp1500523>

Collins, G. S., & Altman, D. G. (2009). An independent external validation and evaluation of QRISK cardiovascular risk prediction: A prospective open cohort study. *The BMJ*, 339, b2584. <https://doi.org/10.1136/bmj.b2584>

Crawshaw, M. (2020). Multi-Task Learning with Deep Neural Networks: A Survey.

ArXiv:2009.09796 [Cs, Stat]. <http://arxiv.org/abs/2009.09796>

Cutillo, C. M., Sharma, K. R., Foschini, L., Kundu, S., Mackintosh, M., & Mandl, K. D. (2020).

Machine intelligence in healthcare—Perspectives on trustworthiness, explainability, usability, and transparency. *Npj Digital Medicine*, 3(1), Article 1.

<https://doi.org/10.1038/s41746-020-0254-2>

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4), 303–314. <https://doi.org/10.1007/BF02551274>

Dara, S., Dhamecherla, S., Jadav, S. S., Babu, C. M., & Ahsan, M. J. (2021). Machine Learning in Drug Discovery: A Review. *Artificial Intelligence Review*, 1–53.

<https://doi.org/10.1007/s10462-021-10058-4>

Davenport, T., & Kalakota, R. (2019). The potential for artificial intelligence in healthcare. *Future Healthcare Journal*, 6(2), 94–98. <https://doi.org/10.7861/futurehosp.6-2-94>

- Day, O., & Khoshgoftaar, T. M. (2017). A survey on heterogeneous transfer learning. *Journal of Big Data*, 4(1), 29. <https://doi.org/10.1186/s40537-017-0089-0>
- Deng, J., Dong, W., Socher, R., Li, L., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.
- Deng, S., Sun, Y., Zhao, T., Hu, Y., & Zang, T. (2020). A Review of Drug Side Effect Identification Methods. *Current Pharmaceutical Design*, 26(26), 3096–3104. <https://doi.org/10.2174/1381612826666200612163819>
- Diaz-Escobar, J., Ordóñez-Guillén, N. E., Villarreal-Reyes, S., Galaviz-Mosqueda, A., Kober, V., Rivera-Rodriguez, R., & Rizk, J. E. L. (2021). Deep-learning based detection of COVID-19 using lung ultrasound imagery. *PLOS ONE*, 16(8), e0255886. <https://doi.org/10.1371/journal.pone.0255886>
- Ding, F., Fu, J., Jiang, D., Hao, M., & Lin, G. (2018). Mapping the spatial distribution of *Aedes aegypti* and *Aedes albopictus*. *Acta Tropica*, 178, 155–162. <https://doi.org/10.1016/j.actatropica.2017.11.020>
- Du, B., Wang, Z., Zhang, L., Zhang, L., Liu, W., Shen, J., & Tao, D. (2019). Exploring Representativeness and Informativeness for Active Learning. *ArXiv:1904.06685 [Cs, Stat]*. <http://arxiv.org/abs/1904.06685>
- Du, J., Ling, C., & Zhou, Z.-H. (2011). When Does Cotraining Work in Real Data? *IEEE Trans. Knowl. Data Eng.*, 23, 788–799. <https://doi.org/10.1109/TKDE.2010.158>
- Duong, L., Cohn, T., Bird, S., & Cook, P. (2015). Low Resource Dependency Parsing: Cross-lingual Parameter Sharing in a Neural Network Parser. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 845–850. <https://doi.org/10.3115/v1/P15-2139>
- Ebbehoj, A., Thunbo, M. Ø., Andersen, O. E., Glindtvad, M. V., & Hulman, A. (2022). Transfer learning for non-image data in clinical research: A scoping review. *PLOS Digital Health*, 1(2), e0000014. <https://doi.org/10.1371/journal.pdig.0000014>

- Ebert, S., Fritz, M., & Schiele, B. (2011). Pick Your Neighborhood—Improving Labels and Neighborhood Structure for Label Propagation. *DAGM-Symposium*.
- Erickson, B. J., Korfiatis, P., Akkus, Z., & Kline, T. L. (2017). Machine Learning for Medical Imaging. *RadioGraphics*, 37(2), 505–515. <https://doi.org/10.1148/rg.2017160130>
- Fancourt, N., Deloria Knoll, M., Barger-Kamate, B., de Campo, J., de Campo, M., Diallo, M., Ebruke, B. E., Feikin, D. R., Gleeson, F., Gong, W., Hammitt, L. L., Izadnegahdar, R., Kruatrachue, A., Madhi, S. A., Manduku, V., Matin, F. B., Mahomed, N., Moore, D. P., Mwenechanya, M., ... O'Brien, K. L. (2017a). Standardized Interpretation of Chest Radiographs in Cases of Pediatric Pneumonia From the PERCH Study. *Clinical Infectious Diseases*, 64(suppl_3), S253–S261. <https://doi.org/10.1093/cid/cix082>
- Farahani, A., Pourshojae, B., Rasheed, K., & Arabnia, H. R. (2021). A Concise Review of Transfer Learning. *ArXiv:2104.02144 [Cs]*. <http://arxiv.org/abs/2104.02144>
- Fralick, S. (1967). Learning to recognize patterns without a teacher. *IEEE Transactions on Information Theory*, 13(1), 57–64. <https://doi.org/10.1109/TIT.1967.1053952>
- Gal, Y., Islam, R., & Ghahramani, Z. (2017). Deep Bayesian Active Learning with Image Data. *ArXiv:1703.02910 [Cs, Stat]*. <http://arxiv.org/abs/1703.02910>
- Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., & Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM Computing Surveys*, 46(4), 44:1-44:37. <https://doi.org/10.1145/2523813>
- Gao, Y., & Cui, Y. (2020). Deep transfer learning for reducing health care disparities arising from biomedical data inequality. *Nature Communications*, 11(1), Article 1. <https://doi.org/10.1038/s41467-020-18918-3>
- Garde, A., Zhou, G., Raihana, S., Dunsmuir, D., Karlen, W., Dekhordi, P., Huda, T., Arifeen, S. E., Larson, C., Kissoon, N., Dumont, G. A., & Ansermino, J. M. (2016). Respiratory rate and pulse oximetry derived information as predictors of hospital admission in young children in Bangladesh: A prospective observational study. *BMJ Open*, 6(8), e011094. <https://doi.org/10.1136/bmjopen-2016-011094>

- Gazda, M., Gazda, J., Plavka, J., & Drotar, P. (2021). Self-supervised deep convolutional neural network for chest X-ray classification. *IEEE Access*, 9, 151972–151982.
<https://doi.org/10.1109/ACCESS.2021.3125324>
- Ghesu, F. C., Georgescu, B., Mansoor, A., Yoo, Y., Neumann, D., Patel, P., Vishwanath, R. S., Balter, J. M., Cao, Y., Grbic, S., & Comaniciu, D. (2022). Self-supervised Learning from 100 Million Medical Images. *ArXiv:2201.01283 [Cs]*. <http://arxiv.org/abs/2201.01283>
- Ghosh, S., Chakraborty, P., Nsoesie, E. O., Cohn, E., Mekaru, S. R., Brownstein, J. S., & Ramakrishnan, N. (2017). Temporal topic modeling to assess associations between news trends and infectious disease outbreaks. *Scientific Reports*, 7(1), 1–12.
- Gilad-Bachrach, R., Navot, A., & Tishby, N. (2003). An Information Theoretic Tradeoff between Complexity and Accuracy. In B. Schölkopf & M. K. Warmuth (Eds.), *Learning Theory and Kernel Machines* (pp. 595–609). Springer. https://doi.org/10.1007/978-3-540-45167-9_43
- Giordano, C., Brennan, M., Mohamed, B., Rashidi, P., Modave, F., & Tighe, P. (2021). Accessing Artificial Intelligence for Clinical Decision-Making. *Frontiers in Digital Health*, 3.
<https://www.frontiersin.org/article/10.3389/fdgth.2021.645232>
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative Adversarial Networks. *ArXiv E-Prints*.
- Gumba, H., Musyoki, J., Mosobo, M., & Lowe, B. (2019). Implementation of Good Clinical Laboratory Practice in an Immunology Basic Research Laboratory: The KEMRI-Wellcome Trust Research Laboratories Experience. *American Journal of Clinical Pathology*, 151(3), 270–274. <https://doi.org/10.1093/ajcp/aqy138>
- Guo, C., & Berkhahn, F. (2016). Entity Embeddings of Categorical Variables. *ArXiv:1604.06737 [Cs]*. <http://arxiv.org/abs/1604.06737>
- Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On Calibration of Modern Neural Networks. *ArXiv:1706.04599 [Cs]*. <http://arxiv.org/abs/1706.04599>
- Gutmann, M., & Hyvärinen, A. (2010). Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In Y. W. Teh & M. Titterton (Eds.), *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics* (Vol. 9, pp. 297–

304). JMLR Workshop and Conference Proceedings.

<http://proceedings.mlr.press/v9/gutmann10a.html>

Haneef, R., Kab, S., Hrzic, R., Fuentes, S., Fosse-Edorh, S., Cosson, E., & Gallay, A. (2021). Use of artificial intelligence for public health surveillance: A case study to develop a machine Learning-algorithm to estimate the incidence of diabetes mellitus in France. *Archives of Public Health*, 79(1), 168. <https://doi.org/10.1186/s13690-021-00687-0>

He, K., Fan, H., Wu, Y., Xie, S., & Girshick, R. (2020). Momentum Contrast for Unsupervised Visual Representation Learning. *ArXiv:1911.05722 [Cs]*. <http://arxiv.org/abs/1911.05722>

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. *CoRR*, *abs/1512.03385*. <http://arxiv.org/abs/1512.03385>

Hendriksen, J. M. T., Geersing, G. J., Moons, K. G. M., & de Groot, J. a. H. (2013). Diagnostic and prognostic prediction models. *Journal of Thrombosis and Haemostasis*, 11(s1), 129–141. <https://doi.org/10.1111/jth.12262>

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5), 359–366. [https://doi.org/10.1016/0893-6080\(89\)90020-8](https://doi.org/10.1016/0893-6080(89)90020-8)

Huang, J., Gretton, A., Borgwardt, K., Schölkopf, B., & Smola, A. (2006). Correcting Sample Selection Bias by Unlabeled Data. In B. Schölkopf, J. Platt, & T. Hoffman (Eds.), *Advances in Neural Information Processing Systems* (Vol. 19). MIT Press. <https://proceedings.neurips.cc/paper/2006/file/a2186aa7c086b46ad4e8bf81e2a3a19b-Paper.pdf>

Ienco, D., Bifet, A., Zliobait', I., & Pfahringer, B. (2013). *Clustering Based Active Learning for Evolving Data Streams*. 8140. https://doi.org/10.1007/978-3-642-40897-7_6

Ioffe, S., & Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *ArXiv:1502.03167 [Cs]*. <http://arxiv.org/abs/1502.03167>

Iqbal, M. S., Luo, B., Khan, T., Mehmood, R., & Sadiq, M. (2018). Heterogeneous transfer learning techniques for machine learning. *Iran Journal of Computer Science*, 1(1), 31–46. <https://doi.org/10.1007/s42044-017-0004-z>

- Iscen, A., Toliyas, G., Avrithis, Y., & Chum, O. (2019). Label Propagation for Deep Semi-supervised Learning. *ArXiv:1904.04717 [Cs]*. <http://arxiv.org/abs/1904.04717>
- Jaderberg, M., Dalibard, V., Osindero, S., Czarnecki, W. M., Donahue, J., Razavi, A., Vinyals, O., Green, T., Dunning, I., Simonyan, K., Fernando, C., & Kavukcuoglu, K. (2017). Population Based Training of Neural Networks. *ArXiv:1711.09846 [Cs]*. <http://arxiv.org/abs/1711.09846>
- Jaimes, F., Arango, C., Ruiz, G., Cuervo, J., Botero, J., Velez, G., Upegui, N., & Machado, F. (2004). Predicting Bacteremia at the Bedside. *Clinical Infectious Diseases*, 38(3), 357–362. <https://doi.org/10.1086/380967>
- Jospin, L. V., Buntine, W., Boussaid, F., Laga, H., & Bennamoun, M. (2022). Hands-on Bayesian Neural Networks—A Tutorial for Deep Learning Users. *ArXiv:2007.06823 [Cs, Stat]*. <http://arxiv.org/abs/2007.06823>
- Kalantidis, Y., Sariyildiz, M. B., Pion, N., Weinzaepfel, P., & Larlus, D. (2020). *Hard Negative Mixing for Contrastive Learning* (arXiv:2010.01028). arXiv. <https://doi.org/10.48550/arXiv.2010.01028>
- Kaufman, D. R., Sheehan, B., Stetson, P., Bhatt, A. R., Field, A. I., Patel, C., & Maisel, J. M. (2016). Natural Language Processing–Enabled and Conventional Data Capture Methods for Input to Electronic Health Records: A Comparative Usability Study. *JMIR Medical Informatics*, 4(4), e35. <https://doi.org/10.2196/medinform.5544>
- Ke, Z., Wang, D., Yan, Q., Ren, J., & Lau, R. (2019). Dual Student: Breaking the Limits of the Teacher in Semi-Supervised Learning. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 6727–6735. <https://doi.org/10.1109/ICCV.2019.00683>
- Kendall, A., Gal, Y., & Cipolla, R. (2018). Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. *ArXiv:1705.07115 [Cs]*. <http://arxiv.org/abs/1705.07115>
- Khan, M. Z., Kumar, M., & Khattak, M. (2021, July). *ECG Classification using Deep Transfer Learning*. <https://doi.org/10.1109/ICICT52872.2021.00008>
- Kim, J., & Ahn, I. (2021). Infectious disease outbreak prediction using media articles with machine learning models. *Scientific Reports*, 11(1), Article 1. <https://doi.org/10.1038/s41598-021-83926-2>

- Kingma, D. P., & Ba, J. (2017). Adam: A Method for Stochastic Optimization. *ArXiv:1412.6980 [Cs]*. <http://arxiv.org/abs/1412.6980>
- Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. *ArXiv Preprint ArXiv:1312.6114*.
- Kohavi, R. (1996). Bias plus variance decomposition for zero-one loss functions. In *Machine Learning: Proceedings of the Thirteenth International Conference*, 275–283.
- Kondratovich, E., Baskin, I. I., & Varnek, A. (2013). Transductive Support Vector Machines: Promising Approach to Model Small and Unbalanced Datasets. *Molecular Informatics*, 32(3), 261–266. <https://doi.org/10.1002/minf.201200135>
- Kormushev, P., Calinon, S., & Caldwell, D. G. (2013). Reinforcement Learning in Robotics: Applications and Real-World Challenges. *Robotics*, 2(3), Article 3. <https://doi.org/10.3390/robotics2030122>
- Krishnan, S., & Athavale, Y. (2018). Trends in biomedical signal feature extraction. *Biomedical Signal Processing and Control*, 43, 41–63. <https://doi.org/10.1016/j.bspc.2018.02.008>
- Kroegel, M.-A., & Scheffer, T. (2004). Multi-Relational Learning, Text Mining, and Semi-Supervised Learning for Functional Genomics. *Machine Learning*, 57(1), 61–81. <https://doi.org/10.1023/B:MACH.0000035472.73496.0c>
- Kull, M., Filho, T. M. S., & Flach, P. (2017). Beyond sigmoids: How to obtain well-calibrated probabilities from binary classifiers with beta calibration. *Electronic Journal of Statistics*, 11(2), 5052–5080. <https://doi.org/10.1214/17-EJS1338SI>
- La Salvia, M., Secco, G., Torti, E., Florimbi, G., Guido, L., Lago, P., Salinaro, F., Perlini, S., & Leporati, F. (2021). Deep learning and lung ultrasound for Covid-19 pneumonia detection and severity classification. *Computers in Biology and Medicine*, 136, 104742. <https://doi.org/10.1016/j.compbiomed.2021.104742>
- Laine, S., & Aila, T. (2017). Temporal Ensembling for Semi-Supervised Learning. *ArXiv:1610.02242 [Cs]*. <http://arxiv.org/abs/1610.02242>
- Le, Q. V., Ranzato, M., Monga, R., Devin, M., Chen, K., Corrado, G. S., Dean, J., & Ng, A. Y. (2012). Building high-level features using large scale unsupervised learning. *ArXiv:1112.6209 [Cs]*. <http://arxiv.org/abs/1112.6209>

- Lecouat, B., Foo, C.-S., Zenati, H., & Chandrasekhar, V. (2018). Manifold regularization with GANs for semi-supervised learning. *ArXiv:1807.04307 [Cs, Stat]*.
<http://arxiv.org/abs/1807.04307>
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444.
<https://doi.org/10.1038/nature14539>
- LeCun, Y., Haffner, P., Bottou, L., & Bengio, Y. (1999). Object Recognition with Gradient-Based Learning. In D. A. Forsyth, J. L. Mundy, V. di Gesú, & R. Cipolla (Eds.), *Shape, Contour and Grouping in Computer Vision* (pp. 319–345). Springer. https://doi.org/10.1007/3-540-46805-6_19
- Li, C., Xu, K., Zhu, J., & Zhang, B. (2017). Triple Generative Adversarial Nets. *ArXiv:1703.02291 [Cs]*. <http://arxiv.org/abs/1703.02291>
- Li, L., Jamieson, K., Rostamizadeh, A., Gonina, E., Ben-tzur, J., Hardt, M., Recht, B., & Talwalkar, A. (2020). A System for Massively Parallel Hyperparameter Tuning. *Proceedings of Machine Learning and Systems*, 2, 230–246.
- Liaw, R., Liang, E., Nishihara, R., Moritz, P., Gonzalez, J. E., & Stoica, I. (2018). Tune: A Research Platform for Distributed Model Selection and Training. *ArXiv:1807.05118 [Cs, Stat]*.
<http://arxiv.org/abs/1807.05118>
- Lin, B., Ye, F., & Zhang, Y. (2021). A Closer Look at Loss Weighting in Multi-Task Learning. *ArXiv:2111.10603 [Cs]*. <http://arxiv.org/abs/2111.10603>
- Liu, X., & Xiang, X. (2020). How Does GAN-based Semi-supervised Learning Work? *ArXiv:2007.05692 [Cs, Stat]*. <http://arxiv.org/abs/2007.05692>
- Liu, Y., Gadepalli, K., Norouzi, M., Dahl, G. E., Kohlberger, T., Boyko, A., Venugopalan, S., Timofeev, A., Nelson, P. Q., Corrado, G. S., Hipp, J. D., Peng, L., & Stumpe, M. C. (2017). Detecting Cancer Metastases on Gigapixel Pathology Images. *CoRR*, *abs/1703.02442*.
<http://arxiv.org/abs/1703.02442>
- Lujic, S., Watson, D. E., Randall, D. A., Simpson, J. M., & Jorm, L. R. (2014). Variation in the recording of common health conditions in routine hospital data: Study using linked survey and administrative data in New South Wales, Australia. *BMJ Open*, 4(9), e005768.
<https://doi.org/10.1136/bmjopen-2014-005768>

- Luxburg, U., & Schölkopf, B. (2008). Statistical Learning Theory: Models, Concepts, and Results. *Handbook of the History of Logic*, 10. <https://doi.org/10.1016/B978-0-444-52936-7.50016-1>
- Lynn, L. A. (2019). Artificial intelligence systems for complex decision-making in acute care medicine: A review. *Patient Safety in Surgery*, 13(1), 6. <https://doi.org/10.1186/s13037-019-0188-2>
- Mahony, N. O., Campbell, S., Carvalho, A., Harapanahalli, S., Velasco-Hernandez, G., Krpalkova, L., Riordan, D., & Walsh, J. (2020). Deep Learning vs. Traditional Computer Vision. *ArXiv:1910.13796 [Cs]*, 943. <https://doi.org/10.1007/978-3-030-17795-9>
- Marcel, S., & Rodriguez, Y. (2010). Torchvision the Machine-Vision Package of Torch. *Proceedings of the 18th ACM International Conference on Multimedia*, 1485–1488. <https://doi.org/10.1145/1873951.1874254>
- Matsoukas, C., Haslum, J. F., Söderberg, M., & Smith, K. (2021). Is it Time to Replace CNNs with Transformers for Medical Images? *ArXiv:2108.09038 [Cs]*. <http://arxiv.org/abs/2108.09038>
- Mawji, A., Akech, S., Mwaniki, P., Dunsmuir, D., Bone, J., Wiens, M., G♦rges, M., Kimutai, D., Kissoon, N., English, M., & Ansermino, M. (2021). Derivation and internal validation of a data-driven prediction model to guide frontline health workers in triaging children under-five in Nairobi, Kenya [version 3; peer review: 2 approved]. *Wellcome Open Research*, 4(121). <https://doi.org/10.12688/wellcomeopenres.15387.3>
- McComb, M., Bies, R., & Ramanathan, M. (2021). Machine learning in pharmacometrics: Opportunities and challenges. *British Journal of Clinical Pharmacology*, n/a(n/a). <https://doi.org/10.1111/bcp.14801>
- McCool, J., Dobson, R., Muinga, N., Paton, C., Pagliari, C., Agawal, S., Labrique, A., Tanielu, H., & Whittaker, R. (2020). Factors influencing the sustainability of digital health interventions in low-resource settings: Lessons from five countries. *Journal of Global Health*, 10(2), 020396. <https://doi.org/10.7189/jogh.10.020396>
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. *ArXiv:1310.4546 [Cs, Stat]*. <http://arxiv.org/abs/1310.4546>

- Miljković, F., Martinsson, A., Obrezanova, O., Williamson, B., Johnson, M., Sykes, A., Bender, A., & Greene, N. (2021). Machine Learning Models for Human In Vivo Pharmacokinetic Parameters with In-House Validation. *Molecular Pharmaceutics*, *18*(12), 4520–4530. <https://doi.org/10.1021/acs.molpharmaceut.1c00718>
- Mooney, S. J., & Pejaver, V. (2018). Big Data in Public Health: Terminology, Machine Learning, and Privacy. *Annual Review of Public Health*, *39*(1), 95–112. <https://doi.org/10.1146/annurev-publhealth-040617-014208>
- Müller, A. (1997). Integral Probability Metrics and Their Generating Classes of Functions. *Advances in Applied Probability*, *29*(2), 429–443. <https://doi.org/10.2307/1428011>
- National Research Council, N. R. (2004). *Computer Science: Reflections on the Field, Reflections from the Field*. National Academies Press.
- Navarrete-Dechent, C., Dusza, S. W., Liopyris, K., Marghoob, A. A., Halpern, A. C., & Marchetti, M. A. (2018). Automated Dermatological Diagnosis: Hype or Reality? *The Journal of Investigative Dermatology*, *138*(10), 2277–2279. <https://doi.org/10.1016/j.jid.2018.04.040>
- Neal, R. M. (2011). *MCMC using Hamiltonian dynamics*. <https://doi.org/10.1201/b10905>
- Nettleton, D. F., Orriols-Puig, A., & Fornells, A. (2010). A study of the effect of different types of noise on the precision of supervised learning techniques. *Artificial Intelligence Review*, *33*(4), 275–306. <https://doi.org/10.1007/s10462-010-9156-z>
- Ng, A., & Jordan, M. (2002). On Discriminative vs. Generative Classifiers: A comparison of logistic regression and naive Bayes. In T. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in Neural Information Processing Systems* (Vol. 14). MIT Press. <https://proceedings.neurips.cc/paper/2001/file/7b7a53e239400a13bd6be6c91c4f6c4e-Paper.pdf>
- Nixon, J., Dusenberry, M., Jerfel, G., Nguyen, T., Liu, J., Zhang, L., & Tran, D. (2020). Measuring Calibration in Deep Learning. *ArXiv:1904.01685 [Cs, Stat]*. <http://arxiv.org/abs/1904.01685>
- Noroozi, M., & Favaro, P. (2017). Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles. *ArXiv:1603.09246 [Cs]*. <http://arxiv.org/abs/1603.09246>
- Ogero, M., Ndiritu, J., Sarguta, R., Tuti, T., Aluvaala, J., & Akech, S. (2023). Recalibrating prognostic models to improve predictions of in-hospital child mortality in resource-limited

settings. *Paediatric and Perinatal Epidemiology*, 37(4), 313–321.

<https://doi.org/10.1111/ppe.12948>

Ominde, M., Sande, J., Ooko, M., Bottomley, C., Benamore, R., Park, K., Ignas, J., Maitland, K., Bwanaali, T., Gleeson, F., & Scott, A. (2018). Reliability and validity of the World Health Organization reading standards for paediatric chest radiographs used in the field in an impact study of Pneumococcal Conjugate Vaccine in Kilifi, Kenya. *PLOS ONE*, 13(7), e0200715. <https://doi.org/10.1371/journal.pone.0200715>

Ouali, Y., Hudelot, C., & Tami, M. (2020). An Overview of Deep Semi-Supervised Learning. *ArXiv:2006.05278 [Cs, Stat]*. <http://arxiv.org/abs/2006.05278>

Pan, S. J., & Yang, Q. (2010). A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10), 1345–1359. <https://doi.org/10.1109/TKDE.2009.191>

Parisotto, E., Ba, J., & Salakhutdinov, R. (2016). Actor-Mimic: Deep Multitask and Transfer Reinforcement Learning. *International Conference on Learning Representations*.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., ... Chintala, S. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d\textquotesingle Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 32* (pp. 8024–8035). Curran Associates, Inc. <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>

Pechenizkiy, M., Tsymbal, A., Puuronen, S., & Pechenizkiy, O. (2006). Class Noise and Supervised Learning in Medical Domains: The Effect of Feature Extraction. *19th IEEE Symposium on Computer-Based Medical Systems (CBMS'06)*, 708–713.

<https://doi.org/10.1109/CBMS.2006.65>

Petti, C. A., Polage, C. R., Quinn, T. C., Ronald, A. R., & Sande, M. A. (2006). Laboratory medicine in Africa: A barrier to effective health care. *Clinical Infectious Diseases: An Official Publication of the Infectious Diseases Society of America*, 42(3), 377–382.

<https://doi.org/10.1086/499363>

- Pham, T.-H., Qiu, Y., Zeng, J., Xie, L., & Zhang, P. (2021). A deep learning framework for high-throughput mechanism-driven phenotype compound screening and its application to COVID-19 drug repurposing. *Nature Machine Intelligence*, 3(3), Article 3.
<https://doi.org/10.1038/s42256-020-00285-9>
- Ponomaryov, V. I., Almaraz-Damian, J. A., Reyes-Reyes, R., & Cruz-Ramos, C. (2021). Chest x-ray classification using transfer learning on multi-GPU. In N. Kehtarnavaz & M. F. Carlsohn (Eds.), *Real-Time Image Processing and Deep Learning 2021* (Vol. 11736, pp. 111–120). SPIE. <https://doi.org/10.1117/12.2587537>
- Price, W. N., & Cohen, I. G. (2019). Privacy in the Age of Medical Big Data. *Nature Medicine*, 25(1), 37–43. <https://doi.org/10.1038/s41591-018-0272-7>
- Qiao, S., Shen, W., Zhang, Z., Wang, B., & Yuille, A. (2018). Deep Co-Training for Semi-Supervised Image Recognition. *ArXiv:1803.05984 [Cs]*. <http://arxiv.org/abs/1803.05984>
- Radford, A., Metz, L., & Chintala, S. (2016). Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *ArXiv:1511.06434 [Cs]*.
<http://arxiv.org/abs/1511.06434>
- Raghu, M., Zhang, C., Kleinberg, J., & Bengio, S. (2019). Transfusion: Understanding Transfer Learning for Medical Imaging. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (Vol. 32). Curran Associates, Inc.
<https://proceedings.neurips.cc/paper/2019/file/eb1e78328c46506b46a4ac4a1e378b91-Paper.pdf>
- Rajkomar, A., Dean, J., & Kohane, I. (2019). Machine Learning in Medicine. *New England Journal of Medicine*, 380(14), 1347–1358. <https://doi.org/10.1056/NEJMra1814259>
- Rajoub, B. (2020). *Characterization of biomedical signals: Feature engineering and extraction* (pp. 29–50). <https://doi.org/10.1016/B978-0-12-818946-7.00002-0>
- Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz, C., Shpanskaya, K., & others. (2017). Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *ArXiv Preprint ArXiv:1711.05225*.

- Rasmus, A., Valpola, H., Honkala, M., Berglund, M., & Raiko, T. (2015). Semi-Supervised Learning with Ladder Networks. *ArXiv:1507.02672 [Cs, Stat]*.
<http://arxiv.org/abs/1507.02672>
- Redfield, C., Tlimat, A., Halpern, Y., Schoenfeld, D. W., Ullman, E., Sontag, D. A., Nathanson, L. A., & Horng, S. (2020). Derivation and validation of a machine learning record linkage algorithm between emergency medical services and the emergency department. *Journal of the American Medical Informatics Association: JAMIA*, 27(1), 147–153.
<https://doi.org/10.1093/jamia/ocz176>
- Ren, G., & Wang, X. (2014). Epidemic spreading in time-varying community networks. *Chaos (Woodbury, N.Y.)*, 24(2), 023116. <https://doi.org/10.1063/1.4876436>
- Ren, P., Xiao, Y., Chang, X., Huang, P.-Y., Li, Z., Gupta, B. B., Chen, X., & Wang, X. (2021). A Survey of Deep Active Learning. *ACM Computing Surveys*, 54(9), 180:1-180:40.
<https://doi.org/10.1145/3472291>
- Rodriguez, S., Hug, C., Todorov, P., Moret, N., Boswell, S. A., Evans, K., Zhou, G., Johnson, N. T., Hyman, B. T., Sorger, P. K., Albers, M. W., & Sokolov, A. (2021). Machine learning identifies candidates for drug repurposing in Alzheimer’s disease. *Nature Communications*, 12(1), 1033. <https://doi.org/10.1038/s41467-021-21330-0>
- Roe, K. D., Jawa, V., Zhang, X., Chute, C. G., Epstein, J. A., Matelsky, J., Shpitser, I., & Taylor, C. O. (2020). Feature engineering with clinical expert knowledge: A case study assessment of machine learning model complexity and performance. *PLOS ONE*, 15(4), e0231300.
<https://doi.org/10.1371/journal.pone.0231300>
- Ruder, S. (2017). An Overview of Multi-Task Learning in Deep Neural Networks. *ArXiv:1706.05098 [Cs, Stat]*. <http://arxiv.org/abs/1706.05098>
- Rumisha, S. F., Lyimo, E. P., Mremi, I. R., Tungu, P. K., Mwingira, V. S., Mbata, D., Malekia, S. E., Joachim, C., & Mboera, L. E. G. (2020). Data quality of the routine health management information system at the primary healthcare facility and district levels in Tanzania. *BMC Medical Informatics and Decision Making*, 20(1), 340. <https://doi.org/10.1186/s12911-020-01366-w>

- Rusu, A. A., Colmenarejo, S. G., Gulcehre, C., Desjardins, G., Kirkpatrick, J., Pascanu, R., Mnih, V., Kavukcuoglu, K., & Hadsell, R. (2015). *Policy Distillation*.
<https://arxiv.org/abs/1511.06295v2>
- Sakudo, A. (2016). Near-infrared spectroscopy for medical applications: Current status and future perspectives. *Clinica Chimica Acta*, 455, 181–188.
- Salem, M., Taheri, S., & Shiun-Yuan, J. (2018). ECG Arrhythmia Classification Using Transfer Learning from 2-Dimensional Deep CNN Features. *2018 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, 1–4. <https://doi.org/10.1109/BIOCAS.2018.8584808>
- Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., & Chen, X. (2016). Improved Techniques for Training GANs. *ArXiv:1606.03498 [Cs]*. <http://arxiv.org/abs/1606.03498>
- Samuel, A. L. (1959). Some Studies in Machine Learning Using the Game of Checkers. *IBM Journal of Research and Development*, 3(3), 210–229. <https://doi.org/10.1147/rd.33.0210>
- Sandfort, V., Yan, K., Pickhardt, P. J., & Summers, R. M. (2019). Data augmentation using generative adversarial networks (CycleGAN) to improve generalizability in CT segmentation tasks. *Scientific Reports*, 9(1), Article 1. <https://doi.org/10.1038/s41598-019-52737-x>
- Sauleau, E. A., Paumier, J.-P., & Buemi, A. (2005). Medical record linkage in health information systems by approximate string matching and clustering. *BMC Medical Informatics and Decision Making*, 5(1), 32. <https://doi.org/10.1186/1472-6947-5-32>
- Saunshi, N., Ash, J., Goel, S., Misra, D., Zhang, C., Arora, S., Kakade, S., & Krishnamurthy, A. (2022). Understanding Contrastive Learning Requires Incorporating Inductive Biases. *ArXiv:2202.14037 [Cs]*. <http://arxiv.org/abs/2202.14037>
- Schwaller, P., Gaudin, T., Lanyi, D., Bekas, C., & Laino, T. (2018). “Found in Translation”: Predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models. *Chemical Science*, 9(28), 6091–6098.
- Schwaller, P., Vaucher, A. C., Laino, T., & Reymond, J.-L. (2021). Prediction of chemical reaction yields using deep learning. *Machine Learning: Science and Technology*, 2(1), 015016. <https://doi.org/10.1088/2632-2153/abc81d>

- Scott, J. A. G., Bauni, E., Moisi, J. C., Ojal, J., Gatakaa, H., Nyundo, C., Molyneux, C. S., Kombe, F., Tsofa, B., Marsh, K., Peshu, N., & Williams, T. N. (2012). Profile: The Kilifi Health and Demographic Surveillance System (KHDSS). *International Journal of Epidemiology*, 41(3), 650–657. <https://doi.org/10.1093/ije/dys062>
- Scudder, H. (1965). Probability of error of some adaptive pattern-recognition machines. *IEEE Transactions on Information Theory*, 11(3), 363–371. <https://doi.org/10.1109/TIT.1965.1053799>
- Şerban, O., Thapen, N. A., Maginnis, B., Hankin, C., & Foot, V. (2019). Real-time processing of social media with SENTINEL: A syndromic surveillance system incorporating deep learning for health classification. *Inf. Process. Manag.* <https://doi.org/10.1016/J.IPM.2018.04.011>
- Seyyed-Kalantari, L., Liu, G., McDermott, M., Chen, I. Y., & Ghassemi, M. (2021). CheXclusion: Fairness gaps in deep chest X-ray classifiers. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, 26, 232–243.
- Shao, J., Wang, Q., & Liu, F. (2019). Learning to Sample: An Active Learning Framework. *ArXiv:1909.03585 [Cs, Stat]*. <http://arxiv.org/abs/1909.03585>
- Shi, W., Gong, Y., Ding, C., Ma, Z., Tao, X., & Zheng, N. (2018). Transductive Semi-Supervised Deep Learning Using Min-Max Features. *ECCV*.
- Shimodaira, H. (2000). Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2), 227–244.
- Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, 6(1), 60. <https://doi.org/10.1186/s40537-019-0197-0>
- Shwartz-Ziv, R., & Armon, A. (2022). Tabular data: Deep learning is not all you need. *Information Fusion*, 81, 84–90. <https://doi.org/10.1016/j.inffus.2021.11.011>
- Singh, J., & Urolagin, S. (2021). *Use of Artificial Intelligence for Health Insurance Claims Automation* (pp. 381–392). https://doi.org/10.1007/978-981-15-5243-4_35
- Sippel, S., Muruganandan, K., Levine, A., & Shah, S. (2011). Review article: Use of ultrasound in the developing world. *International Journal of Emergency Medicine*, 4, 72. <https://doi.org/10.1186/1865-1380-4-72>

- Sjoding, M. W., Dickson, R. P., Iwashyna, T. J., Gay, S. E., & Valley, T. S. (2020). Racial Bias in Pulse Oximetry Measurement. *New England Journal of Medicine*.
<https://doi.org/10.1056/NEJMc2029240>
- Sordo, M., & Zeng, Q. (2005). On sample size and classification accuracy: A performance comparison. *International Symposium on Biological and Medical Data Analysis*, 193–201.
- Su, C.-P., Chen, T. H.-H., Chen, S.-Y., Ghiang, W.-C., Wu, G. H.-M., Sun, H.-Y., Lee, C.-C., Wang, J.-L., Chang, S.-C., Chen, Y.-C., Yen, A. M.-F., Chen, W.-J., & Hsueh, P.-R. (2011). Predictive model for bacteremia in adult patients with blood cultures performed at the emergency department: A preliminary report. *Journal of Microbiology, Immunology and Infection*, 44(6), 449–455. <https://doi.org/10.1016/j.jmii.2011.04.006>
- Sukhija, S., Krishnan, N. C., & Singh, G. (2016). Supervised Heterogeneous Domain Adaptation via Random Forests. *IJCAI*.
- Sultan, M. A., Boyd-Graber, J., & Sumner, T. (2016). Bayesian Supervised Domain Adaptation for Short Text Similarity. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 927–936.
<https://doi.org/10.18653/v1/N16-1107>
- Supratak, A., Wu, C., Dong, H., Sun, K., & Guo, Y. (2016). *Survey on Feature Extraction and Applications of Biosignals* (Vol. 9605, pp. 161–182). https://doi.org/10.1007/978-3-319-50478-0_8
- Sutton, R. S., Barto, Andrew G. . (1998). *Reinforcement learning: An introduction*. MIT Press; /z-wcorg/.
- Syrjälä, H., Broas, M., Suramo, I., Ojala, A., & Lähde, S. (1998). High-resolution computed tomography for the diagnosis of community-acquired pneumonia. *Clinical Infectious Diseases: An Official Publication of the Infectious Diseases Society of America*, 27(2), 358–363. <https://doi.org/10.1086/514675>
- Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., & Liu, C. (2018). A Survey on Deep Transfer Learning. *ArXiv:1808.01974 [Cs, Stat]*. <http://arxiv.org/abs/1808.01974>
- Tan, M., & Le, Q. V. (2020). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *ArXiv:1905.11946 [Cs, Stat]*. <http://arxiv.org/abs/1905.11946>

- Tarvainen, A., & Valpola, H. (2018). Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *ArXiv:1703.01780 [Cs, Stat]*. <http://arxiv.org/abs/1703.01780>
- Tomé, A., Hidalgo-Muñoz, A., López, M., Teixeira, A., Santos, I. M., Pereira, A. T., Vázquez-Marrufo, M., & Lang, E. (2013, February). Feature extraction and classification of biosignals: Emotion valence detection from EEG signals. *Proceedings of BIOSIGNALS 2013-International Conference on Bio-Inspired Systems and Signal Processing*.
- Tommasi, T., Orabona, F., & Caputo, B. (2010). Safety in numbers: Learning categories from few examples with multi model knowledge transfer. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. <https://doi.org/10.1109/CVPR.2010.5540064>
- Tong, S., & Koller, D. (2000). Active Learning for Parameter Estimation in Bayesian Networks. In T. Leen, T. Dietterich, & V. Tresp (Eds.), *Advances in Neural Information Processing Systems* (Vol. 13). MIT Press.
<https://proceedings.neurips.cc/paper/2000/file/0731460a8a5ce1626210cbf4385ae0ef-Paper.pdf>
- Tripathi, R., Reza, A., & Garg, D. (2020). Prediction of the disease controllability in a complex network using machine learning algorithms. *ArXiv:1902.10224 [Physics]*.
<http://arxiv.org/abs/1902.10224>
- Tsybalov, E., Makarychev, S., Shapeev, A., & Panov, M. (2019). Deeper Connections between Neural Networks and Gaussian Processes Speed-up Active Learning. *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, 3599–3605.
<https://doi.org/10.24963/ijcai.2019/499>
- Tuti, T., Bitok, M., Malla, L., Paton, C., Muinga, N., Gathara, D., Gachau, S., Mbevi, G., Nyachiro, W., Ogero, M., Julius, T., Irimu, G., & English, M. (2016). Improving documentation of clinical care within a clinical information network: An essential initial step in efforts to understand and improve care in Kenyan hospitals. *BMJ Global Health*, 1(1).
<https://doi.org/10.1136/bmjgh-2016-000028>
- Uguroglu, S., & Carbonell, J. (2011). Feature Selection for Transfer Learning. In D. Gunopulos, T. Hofmann, D. Malerba, & M. Vazirgiannis (Eds.), *Machine Learning and Knowledge*

Discovery in Databases (pp. 430–442). Springer. https://doi.org/10.1007/978-3-642-23808-6_28

- Um, T. T., Pfister, F. M. J., Pichler, D., Endo, S., Lang, M., Hirche, S., Fietzek, U., & Kulić, D. (2017). Data augmentation of wearable sensor data for parkinson's disease monitoring using convolutional neural networks. *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, 216–220. <https://doi.org/10.1145/3136755.3136817>
- Urbina, F., Puhl, A. C., & Ekins, S. (2021). Recent advances in drug repurposing using machine learning. *Current Opinion in Chemical Biology*, 65, 74–84. <https://doi.org/10.1016/j.cbpa.2021.06.001>
- Vandenhende, S., Georgoulis, S., Van Gansbeke, W., Proesmans, M., Dai, D., & Van Gool, L. (2021). Multi-Task Learning for Dense Prediction Tasks: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–1. <https://doi.org/10.1109/TPAMI.2021.3054719>
- Vanyan, A., & Khachatryan, H. (2021). Deep Semi-Supervised Image Classification Algorithms: A Survey. *JUCS - Journal of Universal Computer Science*, 27(12), 1390–1407. <https://doi.org/10.3897/jucs.77029>
- Vapnik, V. (1998). *The nature of statistical learning theory* (1st ed.). Wiley-Interscience.
- Venton, J., Aston, P. J., Smith, N. A. S., & Harris, P. M. (2020). Signal to Image to Classification: Transfer Learning for ECG. *2020 11th Conference of the European Study Group on Cardiovascular Oscillations (ESGCO)*, 1–2. <https://doi.org/10.1109/ESGCO49734.2020.9158037>
- Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Manzagol, P.-A. (2010). Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion. *J. Mach. Learn. Res.*, 11, 3371–3408.
- Wald, C., Allen, B., Agarwal, S., & Gichoya, J. (2021). *AI Central*. Data Science Institute, American College of Radiology. <https://aicentral.acrdsi.org/>
- Wang, D., Zhang, J., Du, B., Xia, G.-S., & Tao, D. (2022). An Empirical Study of Remote Sensing Pretraining. *ArXiv:2204.02825 [Cs]*. <http://arxiv.org/abs/2204.02825>

- Wang, G., Hwang, J.-N., Rose, C., & Wallace, F. (2017). Uncertainty sampling based active learning with diversity constraint by sparse selection. *2017 IEEE 19th International Workshop on Multimedia Signal Processing (MMSP)*, 1–6.
<https://doi.org/10.1109/MMSP.2017.8122269>
- Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., & Summers, R. M. (2017). Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference On*, 3462–3471.
- Wardi, G., Carlile, M., Holder, A., Shashikumar, S., Hayden, S. R., & Nemati, S. (2021). Predicting Progression to Septic Shock in the Emergency Department Using an Externally Generalizable Machine-Learning Algorithm. *Annals of Emergency Medicine*, 77(4), 395–406. <https://doi.org/10.1016/j.annemergmed.2020.11.007>
- Wasserman, L., & Lafferty, J. (2008). Statistical Analysis of Semi-Supervised Regression. In J. Platt, D. Koller, Y. Singer, & S. Roweis (Eds.), *Advances in Neural Information Processing Systems* (Vol. 20). Curran Associates, Inc.
<https://proceedings.neurips.cc/paper/2007/file/53c3bce66e43be4f209556518c2fcb54-Paper.pdf>
- Weimann, K., & Conrad, T. O. F. (2021). Transfer learning for ECG classification. *Scientific Reports*, 11(1), 5251. <https://doi.org/10.1038/s41598-021-84374-8>
- Weiss, K., Khoshgoftaar, T. M., & Wang, D. (2016). A survey of transfer learning. *Journal of Big Data*, 3(1), 9.
- Westh, H., Lisby, G., Breysse, F., Böddinghaus, B., Chomarat, M., Gant, V., Goglio, A., Raglio, A., Schuster, H., Stuber, F., Wissing, H., & Hoefl, A. (2009). Multiplex real-time PCR and blood culture for identification of bloodstream pathogens in patients with suspected sepsis. *Clinical Microbiology and Infection*, 15(6), 544–551. <https://doi.org/10.1111/j.1469-0691.2009.02736.x>
- White, M. D., Tarakanov, A., Race, C. P., Withers, P. J., & Law, K. J. H. (2022). Digital Fingerprinting of Microstructures. *ArXiv:2203.13718 [Cond-Mat, Physics:Physics]*.
<http://arxiv.org/abs/2203.13718>

- Wiens, M., Kabakyenga, J., Kumbakumba, E., Businge, S., Tagoola, A., Kenya Mugisha, N., Lavoie, P., Ansermino, J. M., & Kissoon, N. (Tex). (2021b). <6m Observation—Pulse Oximetry (dataset) Smart Discharge (Version V1). Scholars Portal Dataverse. <https://doi.org/10.5683/SP2/ZDDFZL>
- Wiens, M., Kabakyenga, J., Kumbakumba, E., Businge, S., Tagoola, A., Kenya-Mugisha, N., Lavoie, P., Ansermino, J. M., & Kissoon, N. (Tex). (2021a). 6-60m Observation—Pulse Oximetry (dataset) Smart Discharge (Version V1). Scholars Portal Dataverse. <https://doi.org/10.5683/SP2/PHX4C5>
- Willyard, C. (2019). Can AI Fix Medical Records? *Nature*, 576(7787), S59–S62. <https://doi.org/10.1038/d41586-019-03848-y>
- Wilson, M. L., Fleming, K. A., Kuti, M. A., Looi, L. M., Lago, N., & Ru, K. (2018). Access to pathology and laboratory medicine services: A crucial gap. *Lancet (London, England)*, 391(10133), 1927–1938. [https://doi.org/10.1016/S0140-6736\(18\)30458-6](https://doi.org/10.1016/S0140-6736(18)30458-6)
- Wu, Y., Yang, F., Liu, Y., Zha, X., & Yuan, S. (2018). A Comparison of 1-D and 2-D Deep Convolutional Neural Networks in ECG Classification. *ArXiv:1810.07088 [Cs]*. <http://arxiv.org/abs/1810.07088>
- Xie, Q., Luong, M.-T., Hovy, E., & Le, Q. V. (2020). Self-training with Noisy Student improves ImageNet classification. *ArXiv:1911.04252 [Cs, Stat]*. <http://arxiv.org/abs/1911.04252>
- Xu, Z., Akella, R., & Zhang, Y. (2007). Incorporating Diversity and Density in Active Learning for Relevance Feedback. *Proceedings of the 29th European Conference on IR Research*, 246–257.
- Xuan, J., Lu, J., & Zhang, G. (2021). Bayesian Transfer Learning: An Overview of Probabilistic Graphical Models for Transfer Learning. *ArXiv:2109.13233 [Cs]*. <http://arxiv.org/abs/2109.13233>
- Yadav, H., Shah, D., Sayed, S., Horton, S., & Schroeder, L. F. (2021). Availability of essential diagnostics in ten low-income and middle-income countries: Results from national health facility surveys. *The Lancet Global Health*, 9(11), e1553–e1560. [https://doi.org/10.1016/S2214-109X\(21\)00442-3](https://doi.org/10.1016/S2214-109X(21)00442-3)

- Yang, F., Zhang, Q., Ji, X., Zhang, Y., Li, W., Peng, S., & Xue, F. (2022). Machine Learning Applications in Drug Repurposing. *Interdisciplinary Sciences: Computational Life Sciences*. <https://doi.org/10.1007/s12539-021-00487-8>
- Yang, X., Song, Z., King, I., & Xu, Z. (2021). A Survey on Deep Semi-supervised Learning. *ArXiv:2103.00550 [Cs]*. <http://arxiv.org/abs/2103.00550>
- Yang, Y., & Hospedales, T. M. (2017). Trace Norm Regularised Deep Multi-Task Learning. *ArXiv:1606.04038 [Cs]*. <http://arxiv.org/abs/1606.04038>
- Yang, Y., Ma, Z., Nie, F., Chang, X., & Hauptmann, A. G. (2015). Multi-Class Active Learning by Uncertainty Sampling with Diversity Maximization. *International Journal of Computer Vision*, 113(2), 113–127. <https://doi.org/10.1007/s11263-014-0781-x>
- Yao, L., Prosky, J., Poblenz, E., Covington, B., & Lyman, K. (2018). Weakly Supervised Medical Diagnosis and Localization from Multiple Resolutions. *ArXiv:1803.07703 [Cs]*. <http://arxiv.org/abs/1803.07703>
- Yoon, D., Lim, H. S., Jung, K., Kim, T. Y., & Lee, S. (2019). Deep Learning-Based Electrocardiogram Signal Noise Detection and Screening Model. *Healthcare Informatics Research*, 25(3), 201–211. <https://doi.org/10.4258/hir.2019.25.3.201>
- Yu, T., Kumar, S., Gupta, A., Levine, S., Hausman, K., & Finn, C. (2020). *Gradient Surgery for Multi-Task Learning*.
- Zhang, H. (2004). The Optimality of Naive Bayes. *Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference, FLAIRS 2004*, 2.
- Zhang, P., Wang, F., Hu, J., & Sorrentino, R. (2015). Label Propagation Prediction of Drug-Drug Interactions Based on Clinical Side Effects. *Scientific Reports*, 5. <https://doi.org/10.1038/srep12339>
- Zhang, W., Deng, L., Zhang, L., & Wu, D. (2021). A Survey on Negative Transfer. *ArXiv:2009.00909 [Cs, Stat]*. <http://arxiv.org/abs/2009.00909>
- Zhang, Y., Ibaraki, M., & Schwartz, F. W. (2020). Disease surveillance using online news: Dengue and zika in tropical countries. *Journal of Biomedical Informatics*, 102, 103374.
- Zhang, Y., & Yang, Q. (2017). A Survey on Multi-Task Learning. *CoRR*, *abs/1707.08114*. <http://arxiv.org/abs/1707.08114>

- Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., & Sun, M. (2020). Graph neural networks: A review of methods and applications. *AI Open*, 1, 57–81.
<https://doi.org/10.1016/j.aiopen.2021.01.001>
- Zhou, S. K., Greenspan, H., Davatzikos, C., Duncan, J. S., van Ginneken, B., Madabhushi, A., Prince, J. L., Rueckert, D., & Summers, R. M. (2021). A review of deep learning in medical imaging: Imaging traits, technology trends, case studies with progress highlights, and future promises. *ArXiv:2008.09104 [Cs, Eess]*. <https://doi.org/10.1109/JPROC.2021.3054390>
- Zhu, X. (2005). *Semi-Supervised Learning Literature Survey* [Technical Report]. University of Wisconsin-Madison Department of Computer Sciences.
<https://minds.wisconsin.edu/handle/1793/60444>
- Zhu, X., & Ghahramani, Z. (2002). *Learning from Labeled and Unlabeled Data with Label Propagation*. <http://www.scholar.google.com/url?sa=U\&q=http://www.cs.cmu.edu/~zhuxj/pub/propagate.ps.gz>
- Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., & He, Q. (2020). A Comprehensive Survey on Transfer Learning. *ArXiv:1911.02685 [Cs, Stat]*.
<http://arxiv.org/abs/1911.02685>

Appendix A: Map of CIN and KHDSS hospitals

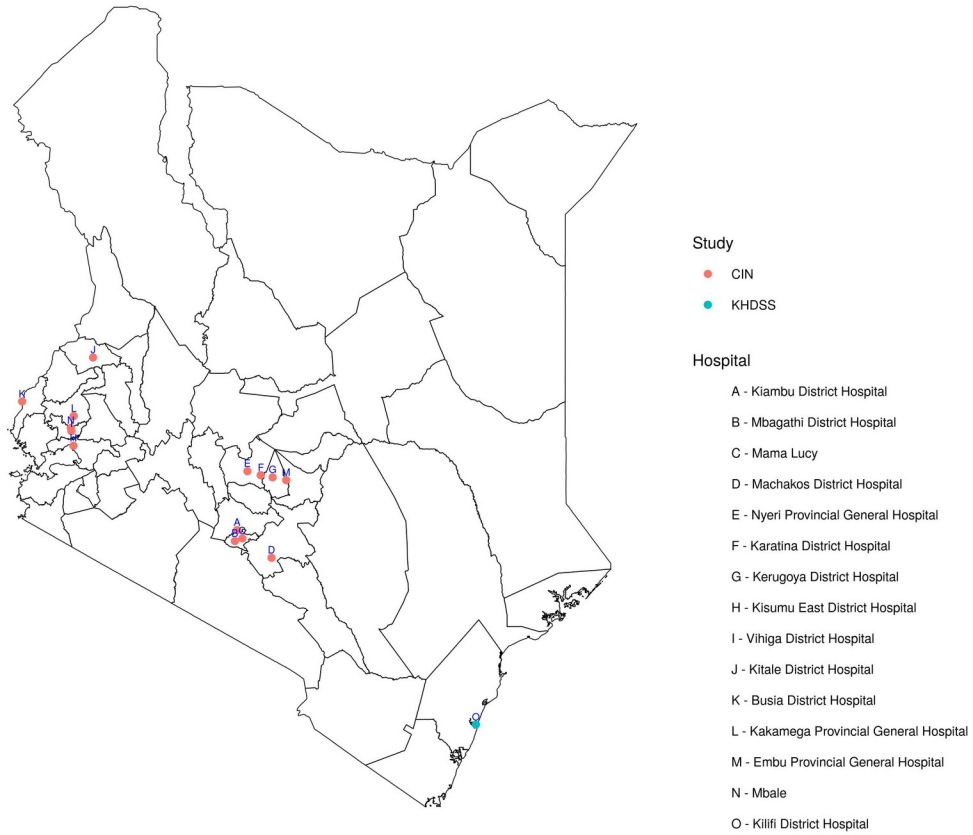


Figure 25: Location of 14 CIN hospitals and Kilifi county hospital. Labelled data for blood culture was only available in Kilifi hospital

Appendix B: Performance of models predicting blood culture results using clinical signs and symptoms

Table 13: Accuracy, F1, precision, recall, and specificity of models predicting blood culture results using clinical signs and symptoms

			Proportion of labelled data used					
			0.05	0.1	0.3	0.5	0.7	1
Metric	Model	datasets						
Accuracy	Logistic (Baseline)	Kilifi	0.765± 0.021	0.745± 0.033	0.728± 0.011	0.721± 0.009	0.717± 0.006	0.710± 0.005
		Kilifi & CIN	0.780± 0.038	0.755± 0.012	0.718± 0.011	0.726± 0.013	0.716± 0.008	0.711± 0.005
	Logistic: Self Training	Kilifi	0.777± 0.024	0.750± 0.041	0.733± 0.015	0.707± 0.011	0.720± 0.006	0.712± 0.007
		Kilifi & CIN	0.777± 0.024	0.750± 0.041	0.733± 0.015	0.707± 0.011	0.720± 0.006	0.712± 0.007
	MLP:Denoi sing auto-encoder	Kilifi	0.170± 0.186	0.052± 0.015	0.438± 0.303	0.147± 0.145	0.370± 0.292	0.514± 0.161
		Kilifi & CIN	0.300± 0.313	0.181± 0.259	0.394± 0.312	0.394± 0.292	0.268± 0.267	0.245± 0.265
	MLP:Rand om initalizatio n	Kilifi	0.499± 0.362	0.178± 0.131	0.554± 0.262	0.426± 0.252	0.367± 0.292	0.714± 0.009
		Kilifi & CIN	0.499± 0.362	0.178± 0.131	0.554± 0.262	0.426± 0.252	0.367± 0.292	0.714± 0.009
	MLP:Spars e auto-encoder	Kilifi	0.747± 0.128	0.520± 0.308	0.380± 0.331	0.352± 0.226	0.604± 0.169	0.490± 0.275
		Kilifi & CIN	0.480± 0.367	0.340± 0.248	0.233± 0.257	0.435± 0.326	0.321± 0.345	0.376± 0.298
	MLP:Spars e auto-encoder (KL)	Kilifi	0.446± 0.370	0.392± 0.265	0.569± 0.283	0.519± 0.274	0.398± 0.256	0.596± 0.215
		Kilifi & CIN	0.377± 0.310	0.137± 0.157	0.296± 0.149	0.274± 0.239	0.267± 0.221	0.581± 0.160
	MLP:Spars e auto-encoder (L1)	Kilifi	0.433± 0.095	0.166± 0.166	0.457± 0.254	0.045± 0.008	0.296± 0.231	0.400± 0.327
		Kilifi & CIN	0.436± 0.292	0.328± 0.262	0.367± 0.289	0.302± 0.265	0.388± 0.291	0.345± 0.215
F1	Logistic (Baseline)	Kilifi	0.135± 0.011	0.140± 0.010	0.142± 0.006	0.141± 0.006	0.141± 0.005	0.138± 0.006
		Kilifi & CIN	0.140± 0.013	0.138± 0.014	0.140± 0.007	0.139± 0.007	0.141± 0.005	0.139± 0.007
	Logistic: Self Training	Kilifi	0.139± 0.018	0.141± 0.008	0.139± 0.006	0.141± 0.006	0.140± 0.006	0.139± 0.005
		Kilifi & CIN	0.140± 0.013	0.138± 0.014	0.140± 0.007	0.139± 0.007	0.141± 0.005	0.139± 0.007
	MLP:Denoi sing auto-encoder	Kilifi	0.081± 0.006	0.078± 0.001	0.111± 0.028	0.083± 0.009	0.109± 0.032	0.117± 0.022
		Kilifi & CIN	0.089± 0.021	0.088± 0.020	0.109± 0.031	0.108± 0.027	0.095± 0.023	0.094± 0.025
	MLP:Rand om initalizatio n	Kilifi	0.076± 0.014	0.084± 0.007	0.121± 0.023	0.113± 0.033	0.107± 0.030	0.144± 0.004
		Kilifi & CIN	0.076± 0.014	0.084± 0.007	0.121± 0.023	0.113± 0.033	0.107± 0.030	0.144± 0.004

	n							
	MLP:Spars e auto- encoder	Kilifi	0.116± 0.022	0.114± 0.027	0.107± 0.031	0.101± 0.023	0.127± 0.021	0.117± 0.027
		Kilifi & CIN	0.089± 0.022	0.097± 0.025	0.092± 0.020	0.115± 0.033	0.106± 0.036	0.108± 0.030
	MLP:Spars e auto- encoder (KL)	Kilifi	0.091± 0.020	0.100± 0.020	0.124± 0.027	0.120± 0.026	0.106± 0.023	0.129± 0.022
		Kilifi & CIN	0.094± 0.016	0.082± 0.008	0.094± 0.011	0.094± 0.019	0.096± 0.021	0.126± 0.022
	MLP:Spars e auto- encoder (L1)	Kilifi	0.089± 0.003	0.085± 0.010	0.114± 0.029	0.077± 0.000	0.096± 0.019	0.112± 0.035
		Kilifi & CIN	0.099± 0.018	0.094± 0.018	0.103± 0.025	0.102± 0.032	0.107± 0.029	0.101± 0.021
Precision	Logistic (Baseline)	Kilifi	0.079± 0.007	0.081± 0.008	0.081± 0.004	0.080± 0.004	0.080± 0.003	0.079± 0.004
	Logistic: Self Training	Kilifi	0.083± 0.012	0.082± 0.005	0.079± 0.003	0.081± 0.004	0.079± 0.003	0.079± 0.003
		Kilifi & CIN	0.083± 0.009	0.081± 0.010	0.080± 0.004	0.079± 0.004	0.081± 0.003	0.079± 0.004
	MLP:Denoi sing auto- encoder	Kilifi	0.043± 0.004	0.041± 0.001	0.061± 0.018	0.044± 0.005	0.060± 0.020	0.064± 0.014
		Kilifi & CIN	0.049± 0.016	0.047± 0.012	0.060± 0.020	0.059± 0.016	0.051± 0.014	0.050± 0.015
	MLP:Rand om inititalizatio n	Kilifi	0.054± 0.012	0.044± 0.004	0.068± 0.014	0.062± 0.022	0.059± 0.019	0.082± 0.003
	MLP:Spars e auto- encoder	Kilifi	0.070± 0.013	0.066± 0.020	0.060± 0.021	0.055± 0.015	0.071± 0.014	0.065± 0.017
		Kilifi & CIN	0.052± 0.015	0.053± 0.017	0.049± 0.013	0.064± 0.021	0.059± 0.023	0.059± 0.019
	MLP:Spars e auto- encoder (KL)	Kilifi	0.057± 0.017	0.055± 0.014	0.072± 0.020	0.067± 0.017	0.058± 0.015	0.072± 0.014
		Kilifi & CIN	0.052± 0.012	0.043± 0.005	0.049± 0.006	0.050± 0.012	0.051± 0.013	0.070± 0.014
	MLP:Spars e auto- encoder (L1)	Kilifi	0.048± 0.001	0.044± 0.006	0.063± 0.019	0.040± 0.000	0.051± 0.011	0.063± 0.023
		Kilifi & CIN	0.055± 0.013	0.051± 0.012	0.056± 0.016	0.055± 0.020	0.059± 0.018	0.054± 0.013
Recall	Logistic (Baseline)	Kilifi	0.457± 0.019	0.512± 0.018	0.562± 0.014	0.570± 0.016	0.580± 0.015	0.579± 0.015
	Logistic: Self Training	Kilifi	0.433± 0.019	0.501± 0.020	0.565± 0.017	0.561± 0.017	0.576± 0.016	0.582± 0.015
		Kilifi & CIN	0.451± 0.024	0.493± 0.022	0.543± 0.016	0.589± 0.015	0.576± 0.016	0.581± 0.015
	MLP:Denoi sing auto- encoder	Kilifi	0.896± 0.017	0.998± 0.014	0.753± 0.018	0.949± 0.017	0.822± 0.016	0.753± 0.015
		Kilifi & CIN	0.772± 0.040	0.906± 0.020	0.788± 0.014	0.800± 0.012	0.875± 0.013	0.885± 0.014
	MLP:Rand	Kilifi	0.571±	0.925±	0.680±	0.795±	0.823±	0.600±

	om inititalizatio n		0.036	0.015	0.017	0.011	0.016	0.012
	MLP:Spars e auto- encoder	Kilifi	0.410± 0.026	0.649± 0.013	0.777± 0.014	0.837± 0.011	0.663± 0.015	0.735± 0.013
		Kilifi & CIN	0.587± 0.037	0.796± 0.017	0.890± 0.008	0.761± 0.013	0.825± 0.012	0.811± 0.012
	MLP:Spars e auto- encoder (KL)	Kilifi	0.638± 0.019	0.764± 0.021	0.648± 0.009	0.712± 0.017	0.802± 0.016	0.673± 0.012
		Kilifi & CIN	0.736± 0.035	0.943± 0.009	0.884± 0.011	0.873± 0.015	0.905± 0.012	0.701± 0.012
	MLP:Spars e auto- encoder (L1)	Kilifi	0.690± 0.036	0.938± 0.023	0.764± 0.016	0.999± 0.018	0.872± 0.015	0.780± 0.012
		Kilifi & CIN	0.695± 0.018	0.803± 0.011	0.800± 0.007	0.867± 0.016	0.795± 0.013	0.852± 0.014
Specificity	Logistic (Baseline)	Kilifi	0.777± 0.022	0.754± 0.035	0.735± 0.013	0.727± 0.009	0.723± 0.007	0.716± 0.005
	Logistic: Self Training	Kilifi	0.795± 0.040	0.765± 0.014	0.724± 0.012	0.733± 0.015	0.721± 0.009	0.717± 0.005
		Kilifi & CIN	0.790± 0.026	0.760± 0.045	0.741± 0.017	0.712± 0.012	0.726± 0.007	0.717± 0.007
	MLP:Denoi sing auto- encoder	Kilifi	0.140± 0.200	0.013± 0.016	0.425± 0.324	0.114± 0.154	0.351± 0.312	0.504± 0.173
		Kilifi & CIN	0.281± 0.337	0.150± 0.277	0.378± 0.333	0.377± 0.311	0.243± 0.285	0.218± 0.282
	MLP:Rand om inititalizatio n	Kilifi	0.496± 0.395	0.147± 0.140	0.549± 0.280	0.411± 0.268	0.348± 0.311	0.719± 0.009
	MLP:Spars e auto- encoder	Kilifi	0.761± 0.140	0.515± 0.332	0.363± 0.355	0.331± 0.241	0.601± 0.181	0.480± 0.294
		Kilifi & CIN	0.475± 0.397	0.320± 0.266	0.206± 0.275	0.421± 0.347	0.300± 0.368	0.357± 0.318
	MLP:Spars e auto- encoder (KL)	Kilifi	0.438± 0.400	0.376± 0.284	0.566± 0.304	0.511± 0.294	0.381± 0.274	0.592± 0.230
		Kilifi & CIN	0.362± 0.334	0.103± 0.168	0.271± 0.158	0.249± 0.255	0.241± 0.234	0.576± 0.172
	MLP:Spars e auto- encoder (L1)	Kilifi	0.423± 0.104	0.133± 0.177	0.444± 0.271	0.005± 0.008	0.272± 0.246	0.384± 0.349
		Kilifi & CIN	0.425± 0.315	0.308± 0.281	0.349± 0.309	0.278± 0.282	0.371± 0.311	0.324± 0.230