xAILAB
EXPLAINABLE MACHINE LEARNING

CNN-based Classification of I-123 ioflupane dopamine transporter SPECT brain images to support the diagnosis of Parkinson's disease with Decision Confidence Estimation

Master Thesis

Master of Science in Applied Computer Science

Aleksej Kucerenko

November 4, 2023

Supervisor:

1st: Prof. Dr. Christian Ledig
2nd: Dr. Ralph Buchert, Universitätsklinikum Hamburg-Eppendorf

Chair of Explainable Machine Learning
Faculty of Information Systems and Applied Computer Sciences
Otto-Friedrich-University Bamberg

Abstract

Short summary of your thesis (max. 1 page) ...

Abstract

Kurze Zusammenfassung Ihrer Abschlussarbeit (max. 1 Seite) ...

Acknowledgements

If you want to thank anyone (optional) ...

Contents

List of Figures

List of Tables

List of Acronyms

AI Artificial Intelligence

Notation

This section provides a concise reference describing notation as used in the book by **?**. If you are unfamiliar with any of the corresponding mathematical concepts, **?** describe most of these ideas in chapters 2–4.

Numbers and Arrays

a	A scalar (integer or real)
$\boldsymbol{a}$	A vector
$\boldsymbol{A}$	A matrix
$\mathbf{A}$	A tensor
$\boldsymbol{I}_n$	Identity matrix with n rows and n columns
$\boldsymbol{I}$	Identity matrix with dimensionality implied by context
$\boldsymbol{e}^{(i)}$	Standard basis vector $[0, \dots, 0, 1, 0, \dots, 0]$ with a 1 at position i
$\text{diag}(\boldsymbol{a})$	A square, diagonal matrix with diagonal entries given by $\boldsymbol{a}$
a	A scalar random variable
$\mathbf{a}$	A vector-valued random variable
$\mathbf{A}$	A matrix-valued random variable

Sets and Graphs

$\mathbb{A}$	A set
$\mathbb{R}$	The set of real numbers
$\{0, 1\}$	The set containing 0 and 1
$\{0, 1, \dots, n\}$	The set of all integers between 0 and n
$[a, b]$	The real interval including a and b
$(a, b]$	The real interval excluding a but including b
$\mathbb{A} \backslash \mathbb{B}$	Set subtraction, i.e., the set containing the elements of $\mathbb{A}$ that are not in $\mathbb{B}$
$\mathcal{G}$	A graph
$Pa_{\mathcal{G}}(\mathrm{x}_i)$	The parents of x_i in $\mathcal{G}$

Indexing

a_i	Element i of vector $\boldsymbol{a}$, with indexing starting at 1
a_{-i}	All elements of vector $\boldsymbol{a}$ except for element i
$A_{i,j}$	Element i, j of matrix $\boldsymbol{A}$
$\boldsymbol{A}_{i,:}$	Row i of matrix $\boldsymbol{A}$
$\boldsymbol{A}_{:,i}$	Column i of matrix $\boldsymbol{A}$
$\mathsf{A}_{i,j,k}$	Element (i, j, k) of a 3-D tensor $\mathbf{A}$
$\mathbf{A}_{:,:,i}$	2-D slice of a 3-D tensor
a_i	Element i of the random vector $\mathbf{a}$

Linear Algebra Operations

$\boldsymbol{A}^\top$	Transpose of matrix $\boldsymbol{A}$
$\boldsymbol{A}^+$	Moore-Penrose pseudoinverse of $\boldsymbol{A}$
$\boldsymbol{A} \odot \boldsymbol{B}$	Element-wise (Hadamard) product of $\boldsymbol{A}$ and $\boldsymbol{B}$
$\det(\boldsymbol{A})$	Determinant of $\boldsymbol{A}$

Calculus

$\frac{dy}{dx}$	Derivative of y with respect to x
$\frac{\partial y}{\partial x}$	Partial derivative of y with respect to x
$\nabla_{\boldsymbol{x}} y$	Gradient of y with respect to $\boldsymbol{x}$
$\nabla_{\boldsymbol{X}} y$	Matrix derivatives of y with respect to $\boldsymbol{X}$
$\nabla_{\mathbf{X}} y$	Tensor containing derivatives of y with respect to $\mathbf{X}$
$\frac{\partial f}{\partial \boldsymbol{x}}$	Jacobian matrix $\boldsymbol{J} \in \mathbb{R}^{m \times n}$ of $f : \mathbb{R}^n \to \mathbb{R}^m$
$\nabla^2_{\boldsymbol{x}} f(\boldsymbol{x})$ or $\boldsymbol{H}(f)(\boldsymbol{x})$	The Hessian matrix of f at input point $\boldsymbol{x}$
$\int f(\boldsymbol{x}) d\boldsymbol{x}$	Definite integral over the entire domain of $\boldsymbol{x}$
$\int_{\mathbb{S}} f(\boldsymbol{x}) d\boldsymbol{x}$	Definite integral with respect to $\boldsymbol{x}$ over the set $\mathbb{S}$

Probability and Information Theory

$\mathrm{a} \perp \mathrm{b}$	The random variables a and b are independent
$\mathrm{a} \perp \mathrm{b} \mid \mathrm{c}$	They are conditionally independent given c
$P(\mathrm{a})$	A probability distribution over a discrete variable
$p(\mathrm{a})$	A probability distribution over a continuous variable, or over a variable whose type has not been specified
$\mathrm{a} \sim P$	Random variable a has distribution P
$\mathbb{E}_{\mathrm{x} \sim P}[f(x)]$ or $\mathbb{E}f(x)$	Expectation of $f(x)$ with respect to $P(\mathrm{x})$
$\mathrm{Var}(f(x))$	Variance of $f(x)$ under $P(\mathrm{x})$
$\mathrm{Cov}(f(x), g(x))$	Covariance of $f(x)$ and $g(x)$ under $P(\mathrm{x})$
$H(\mathrm{x})$	Shannon entropy of the random variable x
$D_{\mathrm{KL}}(P \Vert Q)$	Kullback-Leibler divergence of P and Q
$\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$	Gaussian distribution over $\boldsymbol{x}$ with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$

Functions

$f : \mathbb{A} \to \mathbb{B}$	The function f with domain $\mathbb{A}$ and range $\mathbb{B}$
$f \circ g$	Composition of the functions f and g
$f(\boldsymbol{x}; \boldsymbol{\theta})$	A function of $\boldsymbol{x}$ parametrized by $\boldsymbol{\theta}$. (Sometimes we write $f(\boldsymbol{x})$ and omit the argument $\boldsymbol{\theta}$ to lighten notation)
$\log x$	Natural logarithm of x
$\sigma(x)$	Logistic sigmoid, $\dfrac{1}{1 + \exp(-x)}$
$\zeta(x)$	Softplus, $\log(1 + \exp(x))$
$\Vert \boldsymbol{x} \Vert_p$	L^p norm of $\boldsymbol{x}$
$\Vert \boldsymbol{x} \Vert$	L^2 norm of $\boldsymbol{x}$
x^+	Positive part of x, i.e., $\max(0, x)$
$\mathbf{1}_{\mathrm{condition}}$	is 1 if the condition is true, 0 otherwise

Sometimes we use a function f whose argument is a scalar but apply it to a vector, matrix, or tensor: $f(\boldsymbol{x})$, $f(\boldsymbol{X})$, or $f(\mathbf{X})$. This denotes the application of f to the array element-wise. For example, if $\mathbf{C} = \sigma(\mathbf{X})$, then $C_{i,j,k} = \sigma(X_{i,j,k})$ for all valid values of i, j and k.

Datasets and Distributions

p_{data}	The data generating distribution
$\hat{p}_{\text{data}}$	The empirical distribution defined by the training set
$\mathbb{X}$	A set of training examples
$\boldsymbol{x}^{(i)}$	The i-th example (input) from a dataset
$y^{(i)}$ or $\boldsymbol{y}^{(i)}$	The target associated with $\boldsymbol{x}^{(i)}$ for supervised learning
$\boldsymbol{X}$	The $m \times n$ matrix with input example $\boldsymbol{x}^{(i)}$ in row $\boldsymbol{X}_{i,:}$

1 Introduction

Parkinson's disease (PD) is the second most common neurodegenerative disease after Alzheimer's disease [1]. It is expected to impose an increasing social and economic burden on societies as populations age [2]. The prevalence of PD in industrialized countries is about 1% in people over 60 years of age [2]. The standardized incidence rate of PD is estimated to range between about 10 and about 20 per 100,000 person-years [2]. Thus, this results in the diagnosis of up to 100,000 new PD cases annually in the EU and up to 50,000 cases in the US.

PD is characterized by bradykinesia and variable expression of cardinal symptoms: resting tremor, rigidity, and postural instability [3, 4]. However, this combination of symptoms, often referred to as 'parkinsonism' or 'parkinsonian syndrome' (PS), occurs not only in PD (and some rare 'atypical' neurodegenerative PS such as multiple system atrophy, progressive supranuclear palsy and corticobasal degeneration). It also occurs in so-called 'secondary' (non-neurodegenerative) PS that can be induced by drugs, head trauma, inflammatory or metabolic disorder, as well as other diseases such as essential tremor, dystonic tremor, or normal pressure hydrocephalus [3, 5]. A particularly frequent cause of secondary PS is cerebrovascular disease [6]. The differentiation between PD and secondary PS is highly relevant, because secondary PS might be treated more effectively than PD and some secondary PS may be fully cured. Yet, the clinical, that is, symptom-based differentiation between PD and secondary PS is challenging in a significant fraction of patients, particularly at early disease stages with mild symptoms and in patients with atypical presentation [7, 8]. These cases are often referred to as 'clinical uncertain parkinsonian syndromes' (CUPS) [9].

DAT-SPECT with [^{123}I]FP-CIT is an established nuclear medicine brain imaging procedure for Parkinson's disease diagnosis. The wide usage of the procedure is due to its high accuracy, its relevant impact on patient management, and the strong guideline recommendations. In Europe about 70,000 patients are referred to DAT-SPECT per year, in Germany alone about 10,000, at UKE currently about 400 per year [22]. The demographical change in industrial countries is expected to result in a further increase in the number of DAT-SPECT examinations, because age is the major risk factor for PD [23]. Furthermore, there are early signs of PD such as smell loss and *idiophatic* rapid eye movement sleep and behavioral disorder that can precede movement problems by several years, but are not particularly specific for PD [24-26]. It becomes increasingly important to detect PD at these early premotor stages, because the earlier the treatment is initiated the better the chances of moderating the course of PD with disease-modifying drugs[27].

In clinical practice, the interpretation of DAT-SPECT is binary, that is, the nuclear medicine physician has to decide whether the SPECT images indicate degeneration of the dopaminergic neurotransmitter system (Parkinson's disease) or not (secondary PS). This decision can be challenging by visual inspection of the tomographic SPECT images, particularly for less experienced readers [28]. Thus, DAT-SPECT would benefit from methods for the automatic classification of the images

that achieve similar (or better) performance as experienced readers. Convolutional neural networks (CNNs) appear particularly promising for this purpose [29-47].

Yet, there are also 'true' borderline cases that cannot be classified with high certainty even by expert readers. In DAT-SPECT of CUPS, the proportion of visually inconclusive borderline cases ranges between 5 and 10% [48, 49]. Automatic binary classification of these cases by a CNN might pretend a certainty of the diagnosis that is not actually given. It is important, therefore, to identify these cases in order to make sure that the user visually inspects these SPECT images in order to check the automatic categorization particulary carefully. The user will accept the CNN's decision in some case, overrule the CNN in other cases, and will categorize the remaining cases as actually inconclusive (and might recommend follow-up DAT-SPECT after 6-12 months [50]).

The most obvious approach to identify borderline cases in CNN-based classification would be based on the distance of the CNN's sigmoid output from a predefined decision threshold (e.g., 0.5). However, empirically, sigmoid outputs of CNN for classification of DAT-SPECT tend to cluster at the extreme values so that their utility for the identification of borderline cases seems limited. As a consequence, this approach is not recommended among practitioners, as it tends to overestimate the certainty of CNN-based classification [51-53].

Against this background, the current work aimed to propose and validate a CNN-based approach for the automatic classification of DAT-SPECT that allows reliable identification of inconclusive cases that might be misclassified by the CNN when the decision threshold is strictly applied. The 'decision confidence' of the classifier is evaluated on a metric, proposed in the following, that aims to maximize the performance of the classifier on between-reader consensus cases while minimizing the potential effort of manual inspection originating from inconclusive cases.

Starting from the assumption that between-readers discrepancy in the binary visual interpretation of DAT-SPECT is much more likely in inconclusive cases than in conclusive cases, a standard CNN structure was trained for automatic classification of DAT-SPECT using a large training dataset in which each SPECT image had been visually classified by three independent readers. During the model training phase, the standard-of-truth label was selected randomly from the three independent available reads. This way, the same inconclusive image could be presented to the network with different standard-of-truth labels. The rationale was that this could allow the network to learn about the uncertainty of these cases, and that this would result in sigmoid outputs close to the decision threshold.

This "random label" training (RLT) approach was compared with the conventional majority vote training (MVT) approach. In the latter, the majority vote across the three readers was consistently used as standard-of-truth during the training phase. The MVT obviously "hides" the uncertainty associated with between-readers discrepancy from the network.

To be able to better assess the performance of the CNN-based approaches, univariate and multivariate conventional methods were employed as benchmark methods.

In addition, the performance of the approaches is also evaluated on independent external datasets.

The primary hypothesis put to test in this work was that the sigmoid output of the CNN is more appropriate for the identification of inconclusive cases (by an 'inconclusive' range around the decision threshold) when the network is trained with the RLT approach compared to MVT.

To test this hypothesis, the proportion of inconclusive cases required to achieve a given balanced accuracy in the conclusive cases was proposed and used as a performance metric. More precisely, the area under the curve (AUC) of balanced accuracy in conclusive cases versus the proportion of inconclusive cases (observed in the test set) was used as a model-agnostic quality metric. The AUC does not depend on a specific working point (target balanced accuracy). The rationale for this performance metric is that more inconclusive cases would require more attention and manual inspection by the attending physician which is considered 'expensive' ("90% inconclusive cases to achieve the required accuracy in the remaining 10% of cases is clearly useless"). Therefore the utility of the classifier for widespread use in clinical practice depends on its 'decision confidence', e.g. the proportion of inconclusive cases to be accepted to achieve a predefined balanced accuracy in the remaining conclusive cases.

The following secondary hypotheses were put to test. First, CNN-based classification outperforms conventional methods in terms of balanced accuracy, both univariate and multivariate conventional methods. The specific binding ratio (SBR) of the tracer uptake in the putamen was used for the univariate analyses. Current procedure guidelines recommend the putaminal SBR to support the visual interpretation of DAT-SPECT in everyday clinical patient care [54]. The putaminal SBR characterizes the contrast of the tracer uptake (= intensity) in the putamen relative to the mean tracer uptake in a reference region void of DAT [55]. The putaminal SBR is assumed to be proportional to the density of DAT in the putamen [55]. As a multivariate benchmark method, a random forest approach was implemented using the expression profile of a set of covariance patterns as input. The covariance patterns were identified by principal component analysis in the training dataset.

Second, CNN-based classification demonstrates enhanced generalizability, such as being more robust regarding varying image characteristics (e.g., spatial resolution) associated with the use of different acquisition hardware (different SPECT cameras, different collimators...) and different reconstruction and correction methods (application of resolution recovery, application of attenuation correction...). To test this hypothesis, the classification methods were compared in two test datasets fully independent of the training dataset.

The following research questions are addressed:

- When comparing the CNN-based approaches, how does the RLT approach perform compared to the MVT approach? Is the performance metric proposed in this work practically suitable for the comparison of different approaches?

- How do the CNN-based approaches perform on diverse testing data compared to conventional approaches? What conclusions can be made regarding the generalizability of the approaches under test?

Include thesis structure paragraph.

2 Background

2.1 DAT-SPECT for diagnosing PD

PD, as well as the 'atypical' neurodegenerative PS, is associated with progressive loss of substantia nigra pars compacta (SNpc) dopaminergic neurons projecting to the striatum [10]. Reduced availability of dopamine transporters (DAT) in the striatum is well-validated as a biomarker for nigrostriatal degeneration in PD [11-13]. It can be detected by single photon emission computed tomography (SPECT) with dopamine transporter (DAT) ligands [14, 15]. Reduction of striatal DAT availability is strongly advanced already at the earliest symptomatic (motor) stages of PD, because the degeneration of dopaminergic nerve endings in the striatum is an early step in the pathological PD cascade [11-13]. Compensatory downregulation of the DAT expression in the remaining nerve endings results in even more pronounced striatal DAT loss [16-18]. Secondary PS are as a rule not associated with nigrostriatal degeneration and loss of striatal DAT. To differentiate PD from secondary PS based on striatal DAT availability, the radioactively labeled DAT ligand [^{123}I]FP-CIT (trade name: DaTscan©) has been licensed as SPECT tracer in both, the US and Europe [19].

A recent review, including a non-systematic meta-analysis, of DAT-SPECT with [^{123}I]FP-CIT in PS confirmed high sensitivity (median 93%) and high specificity (median 89%) of DAT-SPECT for the differentiation of PD from secondary PS in patients with CUPS [20]. The review further revealed that DAT-SPECT leads to a change of diagnosis in about 40% and to a change of treatment in about the same proportion of patients with CUPS [20]. Thus, DAT-SPECT with [^{123}I]FP-CIT is highly diagnostically accurate and has a relevant impact on the diagnosis and treatment of CUPS patients. Guidelines from professional neurological societies therefore strongly strengthened the role of DAT-SPECT with [^{123}I]FP-CIT in the last years [21]. For example, the current version of the S3 guideline "Idiopathic Parkinson syndrome" of the German Society of Neurology states that DAT-SPECT *should* be performed at an early disease stage in CUPS.

2.2 Methods for classification

3 Methods

It concludes with an examination of the significant performance metrics utilized for the evaluation of the research outcomes.

3.1 Software Tools and Libraries

The project was built on *Python 3.10*. A variety of widely adopted open-source libraries were used in the project. *NumPy* was utilized to perform efficient array operations and numerical calculations. The *NIBabel* library was used for reading and writing of medical image data stored in the Neuroimaging Informatics Technology Initiative (NIfTI) file format. *PyTorch*, a widely adopted deep learning framework, was employed for building and training the neural networks. The *Torchvision* package provided the machine learning models and image transformation capabilities utilized in this project. *Pandas* was used for efficient structured data manipulation and analysis. *Matplotlib* and *Seaborn* were employed for the creation of customized data visualizations. The *Scikit-Learn* library provided machine learning models and model evaluation tools utilized in this project, whereas Scipy was used for data interpolation.

The seeds of the random number generators in each package were initialized to ensure reproducibility.

3.2 Development Data Preparation

In the following, the data preparation techniques applied to the development dataset are explained in detail.

3.2.1 Data Preprocessing

Individual DAT-SPECT images were stereotactically normalized to the anatomical space of the Montreal Neurological Institute (MNI) using the Normalize tool of the Statistical Parametric Mapping software package (version SPM12) and a set of custom DAT-SPECT templates representative of normal and different levels of Parkinson-typical reduction of striatal uptake as target [73]. The voxel size of the stereotactically normalized images was 2x2x2 mm^3. Intensity normalization was achieved by voxelwise scaling to the individual 75th percentile of the voxel intensity in a reference region comprising the whole brain without striata, thalamus, brainstem, cerebellum, and ventricles [74]. The resulting images are distribution volume (DVR) images. A 2-dimensional transversal DVR slab of 12mm thickness and 91x109 pixels with 2 mm edge length was obtained by averaging 6 transversal slices through the striatum [75].

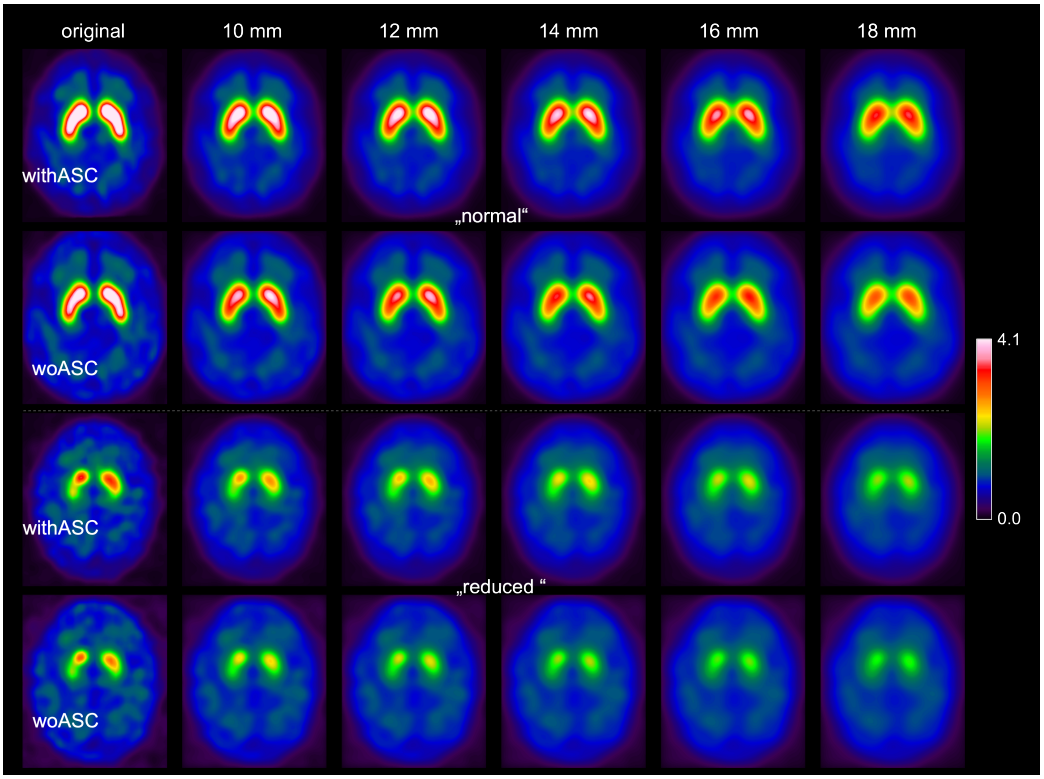


Figure 1: Images obtained through augmentation of two sample cases from the development dataset, a healthy case (above) and a PD case with reduced availability of DAT in the striatum (below).

3.2.2 Data Augmentation

Data augmentation was applied to the development dataset to increase the heterogeneity of the data. To enhance robustness across various attenuation correction and scatter correction methods, each image was generated in a version with and without attenuation and scatter corrections applied. Also 3D-smoothing was employed for augmentation using an isotropic Gaussian kernel with various Full Width at Half Maximum (FWHM) values (FWHM = 10, 12, 14, 16, 18mm). Thereby an augmented dataset of 20,880 images in total was constructed based on 1,740 cases. An example of two cases augmented using the described techniques is depicted in Figure 1.

3.2.3 Dataset Splitting

Ten distinct random splits were created from the augmented development dataset, resulting in ten unique combinations of training, validation, and test sets for the conducted experiments. In each random split, the data distribution was as follows: 60% for the training set, 20% for the validation set, and 20% for the test set. While splitting the data it was ensured that the augmented images associated with a

concrete patient were put only into one subset. Thereby inter-subset data leakage was prohibited.

3.3 Univariate benchmark: Specific Binding Ratio

The unilateral [^{123}I]FP-CIT specific binding ratio (SBR) was used as a benchmark classification method. Here, the SBR in left and right putamen was obtained by hottest voxels (HV) analysis of the stereotactically normalized DVR image using large unilateral putamen masks predefined in MNI space [46]. It can be calculated as

$$\text{HV-SBR}_{unilateral} = \frac{1}{K} \sum_{k} \hat{I}_{k,ROI} \ , \tag{1}$$

where $\hat{I}_{k,ROI}$ are the *normalized* voxel intensities of the K-hottest voxels of the unilateral ROI. The voxel intensities of the hottest voxels are normalized to the 75th percentile of the voxel intensities in the reference region associated with non-specific binding [46]. The minimum of the HV-SBR values from the left and right hemispheres was used for the analysis. An in-depth elaboration on SBR analysis can be found in [46].

The SBR-based classifier was obtained as follows. First the SBR was calculated for each case in the training set. Then the optimal cutoff on the SBR was determined using ROC analysis and the Youden criterion (Youden, 1950). The determined optimal cutoff was then used as the decision boundary between normal cases (NC) and Parkinson's disease (PD) and evaluated on the test split of the development dataset for each of the random splits. Also the determined cutoff was evaluated on the PPMI and MPH datasets described in Section 4.2.

3.4 Multivariate benchmark: PCA-enhanced Random Forest

As a further benchmark, a random forest classifier was trained on PCA-transformed features of the training set of the development dataset.

To be comparable with CNN-based approaches, first, a 91x91 pixel square-shaped region of interest was defined within the 91x109 pixel DVR slab, and each development data image (of each subset) was cropped to this region. The square-shaped region was determined by cropping an equal number of pixels from the top and bottom of the DVR slab along its height dimension.

Then a PCA model with 10 principle components was initialized and fit to the training set features to obtain the principle components of the training set. The determined principle components were used to transform the training set to the lower-dimensional space. An example of the principle components of the training set for one of the random splits is depicted in Figure 2.

The training data transformed by the principle components was then used to train a random forest classifier with 100 decision trees. As hyperparameters, the Gini impurity was used to assess split quality, with a minimum of 2 samples required to split an internal node and 1 sample needed at a leaf node. The trained random forest classifier was evaluated on the test split of the development dataset for each of the 10 random splits. In addition, the trained model was tested on the PPMI and MPH datasets described in Section 4.2.

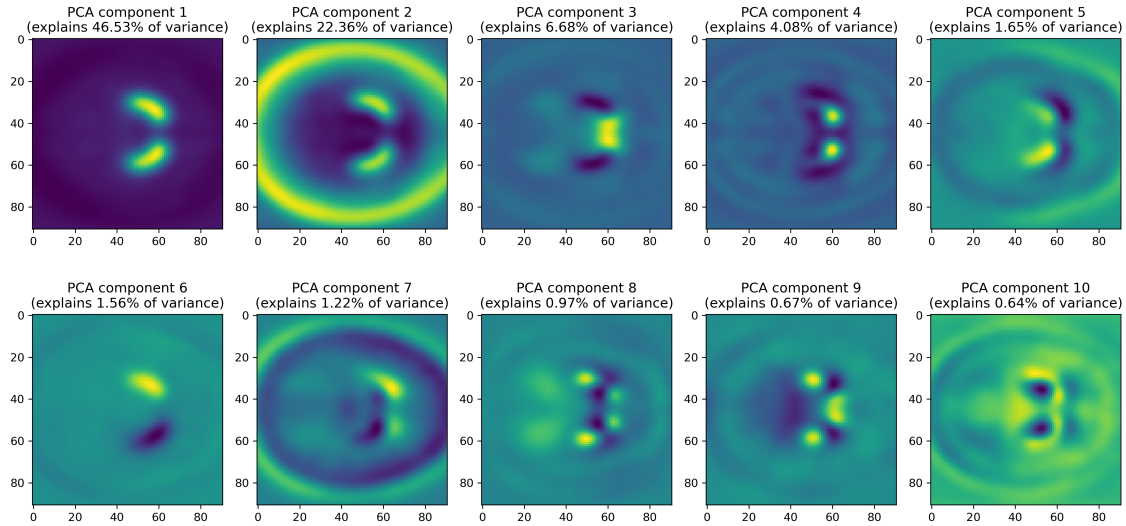


Figure 2: Principle components of the training set (development dataset) for one of the random splits.

3.5 CNN-based classification

The models of CNN-based classifiers were based on a Residual Network (ResNet) architecture. More precisely, the *ResNet-18* (He et al., 2015) model architecture consisting of 18 layers was used as basis. The non-pretrained weights of the ResNet-18 were used as initial weights. The ResNet-18 architecture expects input tensors of size (3, 224, 224), denoting images with 3 channels and spatial dimensions of 224 by 224 pixels. Since the development data has one color channel, the architecture was modified to expect one input channel at its first convolutional layer. Also the dimensions of the last fully-connected layer of the architecture were modified to produce one output node in the output layer. The modified ResNet-18 model is depicted in Figure 3. To obtain a probabilistic model output the sigmoid function was applied to the output layer.

Further development data preprocessing was performed to comply with the spatial input dimensions required by the model architecture. First, a 91x91 pixel square-shaped region of interest was defined within the 91x109 pixel DVR slab, and each development data image (of each subset) was cropped to this region. The cropping to

```
==========================================================================================
Layer (type:depth-idx)                        Output Shape              Param #
==========================================================================================
ResNet18                                      [64, 1]                   --
├─ResNet: 1-1                                 [64, 1]                   --
│    └─Conv2d: 2-1                            [64, 64, 112, 112]        3,136
│    └─BatchNorm2d: 2-2                       [64, 64, 112, 112]        128
│    └─ReLU: 2-3                              [64, 64, 112, 112]        --
│    └─MaxPool2d: 2-4                         [64, 64, 56, 56]          --
│    └─Sequential: 2-5                        [64, 64, 56, 56]          --
│    │    └─BasicBlock: 3-1                   [64, 64, 56, 56]          73,984
│    │    └─BasicBlock: 3-2                   [64, 64, 56, 56]          73,984
│    └─Sequential: 2-6                        [64, 128, 28, 28]         --
│    │    └─BasicBlock: 3-3                   [64, 128, 28, 28]         230,144
│    │    └─BasicBlock: 3-4                   [64, 128, 28, 28]         295,424
│    └─Sequential: 2-7                        [64, 256, 14, 14]         --
│    │    └─BasicBlock: 3-5                   [64, 256, 14, 14]         919,040
│    │    └─BasicBlock: 3-6                   [64, 256, 14, 14]         1,180,672
│    └─Sequential: 2-8                        [64, 512, 7, 7]           --
│    │    └─BasicBlock: 3-7                   [64, 512, 7, 7]           3,673,088
│    │    └─BasicBlock: 3-8                   [64, 512, 7, 7]           4,720,640
│    └─AdaptiveAvgPool2d: 2-9                 [64, 512, 1, 1]           --
│    └─Linear: 2-10                           [64, 1]                   513
==========================================================================================
Total params: 11,170,753
Trainable params: 11,170,753
Non-trainable params: 0
Total mult-adds (G): 111.03
==========================================================================================
Input size (MB): 12.85
Forward/backward pass size (MB): 2543.32
Params size (MB): 44.68
Estimated Total Size (MB): 2600.85
==========================================================================================
```

Figure 3: Overview of the architecture of CNN-based approaches.

a square shape was performed to preserve the aspect ratio while doing the subsequent upscaling. The square-shaped region was determined by cropping an equal number of pixels from the top and bottom of the DVR slab along its height dimension. Then the square-shaped images were resized to the target image size of 224x224 pixels using bicubic interpolation.

The CNN-based approaches were trained for 20 epochs using a batch size of 64. For the MVT and RLT approaches (described in Section 3.5.1) the Binary Cross Entropy (BCE) loss was employed for optimization, whereas for the Regression approach (described in Section 3.5.2) the Mean Squared Error (MSE) loss function was used. The Adam optimization algorithm was utilized with an initial learning rate of 0.0001. During the training of the model, the weights of the best epoch are saved for future evaluation. Each CNN-based approach was trained and evaluated using identical 10 random splits of the development data. Additionally, the trained models were tested on the PPMI and MPH datasets described in Section 4.2.

3.5.1 MVT-based and RLT-based methods

When training a CNN using the BCE loss function, one has to provide the ground truth label of each instance to the optimization algorithm. Given that each instance in the development data is labeled by three independent readers, a selection strategy must be determined. The following two label selection strategies are used for training the CNNs: Majority Vote training (MVT) and "Random Label" training (RLT). The labels chosen using one of the two strategies are then used, together with the model predictions, to compute the BCE loss.

Majority vote training involved selecting the label that received the majority of votes from the readers as the ground truth label. Since there are three available labels, a majority is reached when two out of the three readers agree on a particular label (e.g., the normal case (NC)). During the model training phase, the majority vote strategy was employed to select the labels for both the training and validation data instances.

In contrast to MVT, random label training involved choosing a random label from the three available options as the ground truth label. The seed of the random number generator (responsible for the random selection) is set only once at the start of the algorithm and is not reset between the model training epochs. Thereby a different label could be chosen as the ground truth label for each distinct training epoch. Here the random label selection strategy is applied both to the training and validation data.

3.5.2 Regression-based method

The regression-based approach aimed to incorporate the uncertainty regarding the ground truth label into the training algorithm. Therefore, the ground-truth label was derived from the combination of the three available labels, resulting in a floating-point number. Each of the following states of certainty about the label was mapped to a distinct floating-point valued ground-truth label: *all readers agree on 'normal'* (ground-truth label: 0.0), *majority of readers (two out of three) agree on 'normal'* (ground-truth label: 1.0/3.0), *majority of readers (two out of three) agree on 'reduced'* (ground-truth label: 2.0/3.0) and *all readers agree on 'reduced'* (ground-truth label: 1.0). This mapping of available labels to the ground-truth label was used for both the training and validation data during the model training phase.

During model training the loss was computed using the Mean Square Error loss function which aims to minimize the mean of the squared differences between the model predictions and the ground-truth labels. Thereby the optimization algorithm aimed to separate cases where no consensus was reached from those where consensus was achieved.

3.6 Evaluation Metrics and Procedure

In the following the performance metrics used for the evaluation of the different classification methods are explained in more detail.

First the mean ± SD (standard deviation) of the following measures were calculated across the different random splits for each classification approach and subset (training, validation and testing) given a cutoff: Area Under Curve (AUC) for ROC curve, Balanced accuracy, accuracy, sensitivity, specificity, Positive Predictive Value (PPV) and Negative Predictive Value (NPV). The natural cutoff of 0.5 was used for each classification approach except the SBR method. For the SBR method the optimal cutoff was determined using the Youden criterion (Youden, 1950) and was used for calculating the measures. Majority vote was used as strategy to assign labels for cases in which no between-reader consensus could be achieved.

Second for each element within a set of considered percentages of inconclusive cases in the validation set (PIncVal) the corresponding inconclusive interval was determined. Inconclusive cases were defined as cases predicted within an inconclusive interval (bounded by lower and upper bound), while conclusive cases were those predicted outside this interval. The determination of the inconclusive interval was exclusively performed using the validation set for each random split and classification approach independently. The set of PIncVal values considered ranged from 0.2% to 20.0%, increasing in increments of 0.2%. For each target PIncVal value the lower and upper bounds of the inconclusive interval were independently determined in such a way that there was a similar number of inconclusive cases both below and above the pre-defined cutoff. For the CNN-based classification approaches (described in Section 3.5) and the multivariate benchmark (described in Section 3.4) the natural cutoff of 0.5 was used. For the SBR-based univariate benchmark (described in Section 3.3), the optimal cutoff on the SBR obtained by applying the Youden criterion (Youden, 1950) using ROC analysis was used.

To assess the stability of the determined inconclusive interval over the proportion of inconclusive cases the determined upper and lower bounds (mean±SD) of the inconclusive interval were plotted against the corresponding PIncVal (%). The mean±SD of determined upper and lower bounds was calculated across the measures for different random splits. The rate at which the lower (upper) bound decreases (increases) over the PIncVal reflects the density of inconclusive cases within a certain region of PIncVal. Specifically, higher function gradients indicate lower concentration of predictions, and vice versa. Also a higher standard deviation from the mean indicates that a stable inconclusive interval determination is harder within a certain region of PIncVal. The measurement was conducted separately for each classification approach.

The main performance metric used in this work to evaluate and compare the classification approaches was the area under the curve (AUC) of mean balanced accuracy (%) on conclusive test cases as a function of the mean percentage of inconclusive test cases (mean PIncObs, %). More precisely the relative AUC (%) normalized to the maximum achievable area was used for the comparison. To obtain the relative AUC,

first, the mean balanced accuracy function was interpolated using cubic spline interpolation. Then the area under the mean balanced accuracy curve was computed using the trapezoidal rule and then normalized to the maximum achievable area. The evaluation of each classification method with respect to this metric was conducted on the test set of the development dataset as well as on the independent datasets PPMI and MPH.

As a further metric, the mean ± SD percentage of observed inconclusive cases in the test set (PIncObs, %) was plotted against the PIncVal (%). A mean of PIncObs(PIncVal) near the identity line is an indicator for a similar prediction distribution for validation set and test set on average. In case the mean of PIncObs(PIncVal) consistently lies over (under) the identity line the supposed prediction certainty on the test set, on average, is lower (higher) than on the validation set. Also a lower standard deviation of PIncObs over PIncVal indicates that PIncObs is less sensitive to the randomness of the inconclusive intervals across random splits. Therefore a lower standard deviation of PIncObs allows for a more reliable main performance metric calculation.

4 Data Sources

The study retrospectively included 3 different datasets with a total of 3025 DAT-SPECT images. The primary dataset ("development dataset") was used for both training and testing the models associated with the respective method, whereas the other two datasets, the *PPMI* dataset and the *MPH* dataset were used for testing only, not for training.

4.1 Development dataset

The development dataset comprised 1740 consecutive DAT-SPECT from clinical routine at our site as described in [56]. In brief, DAT-SPECT with [^{123}I]FP-CIT had been performed according to common procedures guidelines [57, 58] with different double-head cameras equipped with low-energy-high-resolution or fan-beam collimators. The projection data were reconstructed using the iterative ordered-subsets-expectation-maximization [59] with attenuation and simulation-based scatter correction as well as collimator-detector response modeling as implemented in the Hybrid Recon-Neurology tool of the Hermes SMART workstation v1.6 (Hermes Medical Solutions, Stockholm, Sweden) [60-63]. All parameter settings were as recommended by Hermes [60] for the EANM / EANM Research Ltd (EARL) EANM-DAT project (European Normal Control Database of DaTSCAN) [64-68]. More precisely, ordered-subsets-expectation-maximization was performed with 5 iterations and 15/16 subsets for 120/128 views. For noise suppression, reconstructed images were postfiltered by convolution with a 3-dimensional Gaussian kernel of 7 mm full-width-at-half-maximum. The development dataset was used for both, training and testing. For this purpose, the dataset was randomly split into ??? training cases

and ??? test cases. The gold standard label as either "normal" or Parkinson-typical reduction ("reduced") of the striatal signal had been obtained by visual interpretation of the DAT-SPECT images by 3 independent readers [56]. The between-reader consensus on the label could not be achieved for around 5% of dataset cases.

4.2 Independent testing datasets

The second dataset comprised 645 DAT-SPECT with [^{123}I]FP-CIT from the Parkinson's Progression Markers Initiative (PPMI) (www.ppmi-info.org/data) [69]. The dataset included 438 patients with Parkinson's disease and 207 healthy controls as described in [46]. Details of the PPMI DAT-SPECT protocol are given at http://www.ppmi-info.org/study-design/research-documents-and-sops/ [69]. Raw projection data has been transferred to the PPMI imaging core lab for central image reconstruction using an iterative (HOSEM) algorithm on a HERMES workstation. The clinical diagnosis was used as gold standard label (Parkinson's disease = "reduced", healthy control = "normal").

The third dataset ("MPH dataset") comprised 640 consecutive DAT-SPECT with [^{123}I]FP-CIT from clinical routine at UKE that had been acquired with a triple-head camera equipped with brain-specific multiple pinhole collimators. Multiple pinhole SPECT concurrently improves count sensitivity and spatial resolution compared to SPECT with parallel-hole and fan-beam collimators [70, 71]. The projection data were reconstructed with the Monte Carlo photon simulation engine and iterative one-step-late maximum-a-posteriori expectation-maximization implemented in the camera software (24 iterations, 2 subsets) [71, 72]. Neither attenuation nor scatter correction was applied. The gold standard label ("normal" or "reduced") was obtained by visual interpretation by an experienced reader (about 20 years of experience in clinical DAT-SPECT reading, $\geq$3,000 cases). All SPECT images were interpreted twice (with different randomization) by the same reader. The delay between the reading sessions was 14 days. Cases with discrepant interpretations between the two reading sessions were read a third time by the same reader to obtain an intra-reader consensus as the gold standard label. The MPH test dataset has not been described previously.

Image characteristics were quite different between the datasets (Figure ???). Compared to the development dataset, the internal test dataset was characterized by better spatial resolution (resulting in higher striatum-to-background contrast) and less statistical noise. The external test dataset showed lower spatial resolution than the development dataset (lower striatum-to-background contrast).

5 Evaluation

The preceding chapters have detailed the research methodology, data collection and sources, and the application of classification techniques to address the research questions posed in this study.

This chapter embarks on the evaluation of the research results, focusing on the performance and effectiveness of the methods employed, and the attainment of the research objectives.

The structure of this chapter has been designed to systematically lead readers through the assessment process. The chapter commences with the examination of the performance results obtained for the baseline methods. The core of this chapter subsequently unveils the results for the experimental methods evaluated using various test datasets and compared to the baseline performance. These findings are presented using performance summary tables for statistical measures and graphical representations. The chapter culminates with a comparative analysis, which seeks to assess and contrast the effectiveness and limitations of the research methods employed.

5.1 Baseline Performance

In this section, the performance of the SBR method, a widely recognized technique in the field, is thoroughly evaluated. Furthermore the outcomes for the multivariate PCA-RFC method are also provided as additional baseline. The objective of this evaluation is twofold: to comprehend the inherent capabilities of the baseline methods, SBR and PCA-RFC, and to establish a clear point of reference for the CNN-based methodologies.

5.1.1 SBR Method Results

Test set of development dataset

Binary classification performance Table 1 presents the quantitative performance (Balanced accuracy, accuracy, sensitivity, specificity, PPV, NPV and AUC-ROC) of the SBR-based classification on the particular subset of the Development dataset. In the evaluation process, the optimal SBR cutoff value of 0.703 (with a variation of ± 0.009 across random splits) was employed. The SBR method consistently achieves around 93% in balanced accuracy, accuracy, sensitivity, specificity, PPV and NPV on the training set, with a variance around 0.5% across random splits. The performance on validation and test set is also around 93% with respect to all the metrics with a slightly higher variance across random splits (0.5-2%) compared to training set. The comparable sensitivity and specificity imply a well-balanced SBR model which identifies both positive and negative cases similarly well. The SBR model achieves a stable AUC-ROC of 0.983 ± 0.002.

Determined inconclusive intervals Figure 4 illustrates the determined lower and upper bounds on the SBR as a function of the percentages of inconclusive cases in the validation set (development dataset), along with the mean$\pm$SD of the optimal

Table 1: Evaluation of the SBR method on Development dataset (SBR cutoff mean±SD: 0.703±0.009).

	train set	validation set	test set
Balanced Accuracy	0.936±0.003	0.929±0.008	0.935±0.007
Accuracy	0.936±0.003	0.930±0.008	0.935±0.007
Sensitivity	0.934±0.006	0.924±0.005	0.930±0.014
Specificity	0.937±0.003	0.935±0.015	0.939±0.012
PPV	0.933±0.005	0.929±0.014	0.930±0.015
NPV	0.938±0.005	0.930±0.004	0.938±0.018
AUC-ROC		0.983±0.002	

cutoff. Corroborating the intuitive expectation, the width of the inconclusive interval expands as the percentage of inconclusive cases increases. The close resemblance in slopes between the upper and lower bound functions indicates a nearly identical distribution of predictions both below and above the cutoff.

Transferability of inconclusive intervals In Figure 5 the correspondence between the percentage of inconclusive cases in the validation set (development dataset) and the mean±SD percentage of observed inconclusive cases (PIncObs) in the test set (development dataset) is demonstrated. The plot illustrates that the deviation of the mean PIncObs in the test set from the identity line is negligibly small. This can be attributed to the nearly identical distribution of data in both the test and validation sets (due to random splitting) which results in a similar distribution of SBR model predictions.

AUC for balanced accuracy over PIncObs Figure 6a shows the balanced accuracy (mean±SD across random splits) on both conclusive and inconclusive cases as a function of the mean PIncObs in the test set (development dataset). The balanced accuracy on inconclusive cases is not part of further performance analysis and comparison due to the emphasis on the balanced accuracy on conclusive cases as the basis for the main metric of this work. The balanced accuracy (mean±SD) on conclusive cases over the mean PIncObs is depicted with enhanced clarity and precision in Figure 6b. The mean of the balanced accuracy rises from approximately 94% when there are around 1% of inconclusive cases in the test set to about 98% when there are around 20% of inconclusive cases in the test set. The SBR baseline method attains a relative AUC of 96.38% for the mean balanced accuracy on conclusive cases over the mean PIncObs in the test set of the development dataset. For better illustration the area under the mean of the balanced accuracy is highlighted.

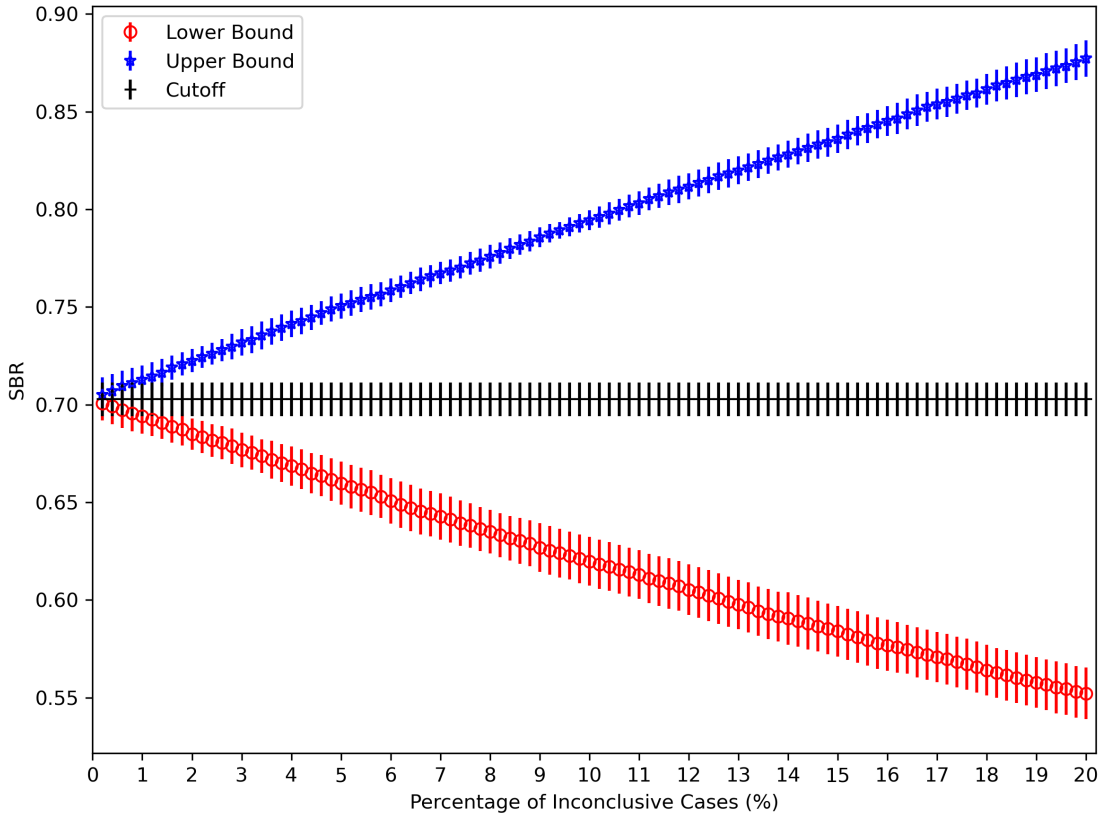


Figure 4: Evaluation of the SBR method on Test Set of Development dataset. Determined upper and lower bounds of the inconclusive interval as a function of the percentage of inconclusive cases.

PPMI dataset The results obtained from evaluating the SBR method on the PPMI dataset are depicted in Figure 7. The mean±SD percentage of inconclusive cases observed (PIncObs) in the PPMI test dataset over the percentage of inconclusive cases in the validation set (development dataset) is consistently below the identity line, which can be seen in Figure 7a. That implies that, on average, the supposed prediction certainty on PPMI dataset is higher than on validation set (development dataset), regardless of the prediction accuracy. The balanced accuracy on conclusive cases over the mean PIncObs is shown in Figure 7b. The mean of the balanced accuracy rises from approximately 96% when there are around 1% of inconclusive cases in the PPMI test set to about 99% when there are around 20% of inconclusive cases in the PPMI test set. The SBR baseline method achieves a relative AUC of 97.51% for the mean balanced accuracy on conclusive cases over the mean PIncObs in the PPMI test dataset. For better illustration the area under the mean of the balanced accuracy is highlighted.

MPH dataset The evaluation of the SBR method on the MPH dataset is shown in Figure 8. Figure 8a demonstrates the mean±SD percentage of inconclusive cases

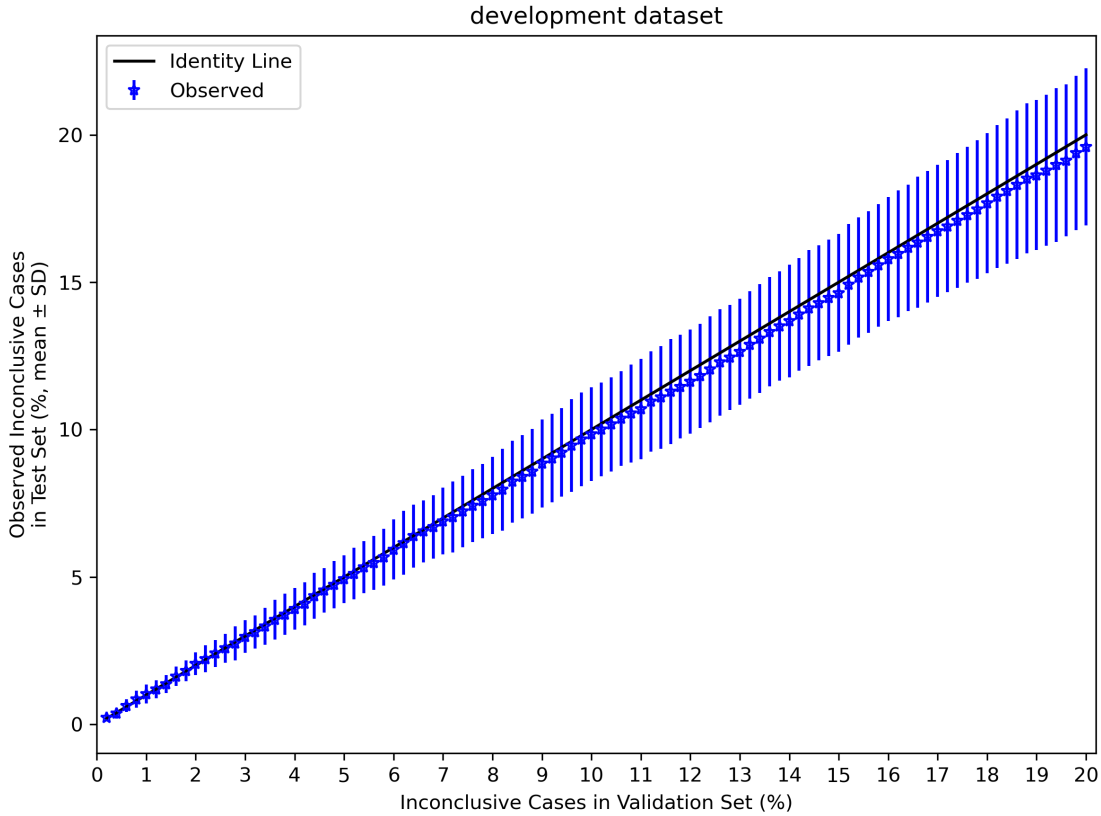


Figure 5: Evaluation of the SBR method on Test Set of Development dataset. Observed percentage of inconclusive cases in the test set for a given set of percentages of inconclusive cases in the validation set. Each of the percentages of inconclusive cases in the validation set is associated with an inconclusive range (determined in the validation set).

observed (PIncObs) in the MPH test dataset over the percentage of inconclusive cases in the validation set (development dataset). Similar as in case of the PPMI dataset, here the PIncObs in the MPH test dataset is also consistently below the identity line and thus the supposed prediction certainty on MPH dataset is higher than on validation set. The balanced accuracy on conclusive cases over the mean percentage of observed inconclusive cases (PIncObs) is shown in Figure 8b. The mean of the balanced accuracy rises from approximately 91.5% when there are around 1% of inconclusive cases in the MPH test set to about 95% when there are around 20% of inconclusive cases in the MPH test set. The SBR baseline method achieves a relative AUC of 93.46% for the mean balanced accuracy on conclusive cases over the mean PIncObs in the MPH test dataset. For better illustration the area under the mean of the balanced accuracy is highlighted.

5.1.2 PCA-RFC Method Results

Test set of development dataset

Binary classification performance Table 2 presents the quantitative performance (Balanced accuracy, accuracy, sensitivity, specificity, PPV, NPV and AUC-ROC) of the PCA-RFC classification on the particular subset of the Development dataset. In the evaluation process, the natural sigmoid cutoff value of 0.5 was employed. The PCA-RFC method achieves around 96% in balanced accuracy, accuracy, sensitivity, specificity, PPV and NPV on the validation and test set, with a variance around 1% across random splits. The SBR model achieves a stable AUC-ROC of 0.994±0.002.

Table 2: Evaluation of the PCA-RFC method on Development dataset. For evaluation, the natural sigmoid cutoff 0.5 was used.

	train set	validation set	test set
Balanced Accuracy	1.000±0.000	0.963±0.010	0.966±0.006
Accuracy	1.000±0.000	0.963±0.010	0.966±0.006
Sensitivity	1.000±0.000	0.957±0.012	0.962±0.010
Specificity	1.000±0.000	0.969±0.011	0.969±0.009
PPV	1.000±0.000	0.966±0.012	0.965±0.010
NPV	1.000±0.000	0.961±0.012	0.966±0.011
AUC-ROC		0.994±0.002	

Determined inconclusive intervals Figure 9 illustrates the determined lower and upper bounds on the probabilistic output as a function of the percentages of inconclusive cases in the validation set (development dataset), along with the natural cutoff of 0.5. The width of the inconclusive interval expands as the percentage of inconclusive cases increases and the visual resemblance in shape and slope between the curve a similar distribution of predictions both below and above the cutoff.

Transferability of inconclusive intervals The correspondence between the percentage of inconclusive cases in the validation set (development dataset) and the mean±SD percentage of observed inconclusive cases (PIncObs) in the test set (development dataset) is demonstrated in Figure 10. The deviation of the mean PIncObs in the test set from the identity line is small which can be attributed to the nearly identical distribution of data in both the test and validation sets (due to random splitting).

AUC for balanced accuracy over PIncObs Figure 11a shows the balanced accuracy (mean±SD across random splits) on both conclusive and inconclusive cases as a function of the mean PIncObs in the test set (development dataset). The balanced accuracy on inconclusive cases is not part of performance analysis and comparison due to the emphasis on the balanced accuracy on conclusive cases as the basis for the main metric of this work. The balanced accuracy (mean±SD) on conclusive cases over the mean PIncObs is depicted with enhanced clarity and precision in Figure 11b. The mean of the balanced accuracy rises from approximately 97% when there are around 1% of inconclusive cases in the test set to about 99.5% when there are around 20% of inconclusive cases in the test set. As a result, the PCA-RFC baseline method achieves a relative AUC of 98.71% for the mean balanced accuracy on conclusive cases over the mean PIncObs in the test set of the development dataset. The area under the mean of the balanced accuracy is highlighted for better illustration.

PPMI dataset The following results were obtained when evaluating the PCA-RFC method on the PPMI dataset. Figure 12a shows the mean±SD percentage of inconclusive cases observed (PIncObs) in the PPMI test dataset over the percentage of inconclusive cases in the validation set (development dataset). The function is consistently above the identity line. Therefore, on average, the supposed prediction certainty of the PCA-RFC method on PPMI dataset is lower than on validation set, regardless of the prediction accuracy. The balanced accuracy on conclusive cases over the mean PIncObs is presented in Figure 12b. The mean of the balanced accuracy rises from approximately 98% when there are around 1% of inconclusive cases in the PPMI test set to about 99.5% when there are around 20% of inconclusive cases in the PPMI test set. The PCA-RFC baseline method achieves a relative AUC of 99.12% for the mean balanced accuracy on conclusive cases over the mean PIncObs in the PPMI test dataset. For better illustration the area under the mean of the balanced accuracy is highlighted.

MPH dataset The evaluation of the PCA-RFC method on the MPH dataset shows the following results. In Figure 13a the mean±SD percentage of inconclusive cases observed (PIncObs) in the MPH test dataset over the percentage of inconclusive cases in the validation set (PIncVal) is illustrated. Here the mean of PIncObs in the MPH test dataset is also consistently above the identity line and its deviation from the identity line increases over PIncVal. Therefore the supposed prediction certainty on MPH dataset is lower than on validation set (development data). The balanced accuracy on conclusive cases over the mean PIncObs is shown in Figure 13b. The mean of the balanced accuracy rises from approximately 90.5% when there are around 1% of inconclusive cases in the MPH test set to about 94% when there are around 19% of inconclusive cases in the MPH test set. As a result, the PCA-RFC baseline method achieves a relative AUC of 92.42% for the mean balanced accuracy on conclusive cases over the mean PIncObs in the MPH test dataset. For better illustration the area under the mean of the balanced accuracy is highlighted.

5.2 Experimental Methods Performance

This section presents the performance results for the CNN-based classification approaches separately and compares them to the results obtained by the baseline approaches. First the results for the CNN-MVT method are presented, whereafter the CNN-RLT method is evaluated. Finally the findings for the CNN-Regression method are showcased.

5.2.1 CNN-MVT Method Results

Test set of development dataset

Binary classification performance The quantitative performance results of the CNN-MVT classification on the particular subset of the Development dataset are presented in Table 3. In the evaluation process, the natural sigmoid cutoff value of 0.5 was employed. The CNN-MVT method achieves around 96.4% in sensitivity, 97.6% in specificity and a balanced accuracy of 97.0%, with a variance between 1-2% across random splits, on the test set. The performance results on the validation set are very similar. The method achieves a stable AUC-ROC of 0.996±0.002.

Table 3: Evaluation of the CNN-MVT method on Development dataset. For evaluation, the natural sigmoid cutoff 0.5 was used.

	train set	validation set	test set
Balanced Accuracy	0.999±0.003	0.970±0.014	0.970±0.008
Accuracy	0.999±0.003	0.970±0.014	0.970±0.008
Sensitivity	1.000±0.000	0.963±0.010	0.964±0.015
Specificity	0.997±0.006	0.976±0.023	0.976±0.013
PPV	0.997±0.006	0.975±0.024	0.972±0.018
NPV	1.000±0.000	0.966±0.010	0.968±0.014
AUC-ROC		0.996±0.002	

Determined inconclusive intervals In Figure 14 the determined lower and upper bounds on the probabilistic sigmoid output are plotted as a function of the percentages of inconclusive cases in the validation set (development dataset), along with the natural cutoff 0.5. The visual resemblance in shape and slope between the upper and lower bound curves indicates a similar distribution of predictions both below and above the cutoff. The width of the inconclusive interval increases more

rapidly as the percentage of inconclusive cases increases when compared to the PCA-RFC baseline method. That implies that the CNN-MVT method produces relatively less inconclusive cases than the PCA-RFC baseline.

Transferability of inconclusive intervals The correspondence between the percentage of inconclusive cases in the validation set (development dataset) and the mean±SD percentage of observed inconclusive cases (PIncObs) in the test set (development dataset) is illustrated in Figure 15. As for the baseline cases, the deviation of the mean PIncObs in the test set from the identity line is small which can be attributed to the nearly identical distribution of data in both the test and validation sets (due to random splitting).

AUC for balanced accuracy over PIncObs The balanced accuracy (mean±SD) on conclusive cases over the mean PIncObs in the test set (development dataset) is depicted in Figure 16b. The mean of the balanced accuracy rises from about 97% when there are around 1% of inconclusive cases in the test set to about 99.5% when there are around 20% of inconclusive cases in the test set. As a result, the CNN-MVT method achieves a relative AUC of 98.95% for the mean balanced accuracy on conclusive cases over the mean PIncObs in the test set of the development dataset. The achieved relative AUC is approximately 2.5% higher than that of the SBR baseline method and around 0.2% higher than the PCA-RFC baseline. The area under the mean of the balanced accuracy is highlighted for better illustration.

PPMI dataset The following results were obtained when evaluating the CNN-MVT method on the PPMI dataset. Figure 17a depicts the mean±SD percentage of inconclusive cases observed (PIncObs) in the PPMI test dataset over the percentage of inconclusive cases in the validation set (PIncVal) of development dataset. For lower PIncVal the corresponding PIncObs in the PPMI test dataset are similar. However as PIncVal increases (corresponding to increasing inconclusive intervals) the supposed prediction certainty on PPMI dataset decreases when compared to the certainty on validation set, on average. The balanced accuracy on conclusive cases over the mean PIncObs is illustrated in Figure 17b. The mean of the balanced accuracy rises from approximately 98% when there are around 1% of inconclusive cases in the PPMI test set to about 99.5% when there are around 20% of inconclusive cases in the PPMI test set. The CNN-MVT method achieves a relative AUC of 99.23% for the mean balanced accuracy on conclusive cases over the mean PIncObs in the PPMI test dataset. The achieved relative AUC is approximately 1.7% higher than that of the SBR baseline method and around 0.1% higher than the PCA-RFC baseline. For better illustration the area under the mean of the balanced accuracy is highlighted.

MPH dataset The evaluation of the CNN-MVT method on the MPH dataset produced the following results. Figure 18a presents the mean±SD percentage of

inconclusive cases observed (PIncObs) in the MPH test dataset over the percentage of inconclusive cases in the validation set (PIncVal) of development dataset. The mean of PIncObs in the MPH test dataset is consisently above the identity line and the deviation from the identity line increases over PIncVal. The standard deviation of PIncObs also increases over PIncVal. When compared to the mean PIncObs of the SBR baseline the mean PIncObs of the CNN-MVT method is higher which indicates that CNN-MVT is supposedly less certain about the MPH set predictions than the SBR method. Also the PIncObs of the CNN-MVT has a much higher standard deviation compared to the PIncObs of the SBR method. The balanced accuracy on conclusive cases over the mean percentage of observed inconclusive cases (PIncObs) is depicted in Figure 18b. The mean of the balanced accuracy increases from approximately 95% when there are around 1% of inconclusive cases in the MPH test set to about 96% when there are around 20% of inconclusive cases in the MPH test set. As a result, the CNN-MVT method achieves a relative AUC of 95.73% for the mean balanced accuracy on conclusive cases over the mean PIncObs in the MPH test dataset. The achieved relative AUC is approximately 2.3% higher than that of the SBR baseline method and around 3.3% higher than the PCA-RFC baseline. For better illustration the area under the mean of the balanced accuracy is highlighted.

5.2.2 CNN-RLT Method Results

Test set of development dataset

Binary classification performance The quantitative performance results of the CNN-RLT classification on the particular subset of the Development dataset are presented in Table 4. In the evaluation process, the natural sigmoid cutoff value of 0.5 was employed. The CNN-RLT method achieves around 96.1% in sensitivity, 98.5% in specificity and a balanced accuracy of 97.3%, with a variance between 0.5-1.5% across random splits, on the test set. The performance results on the validation set are similar. The method achieves a stable AUC-ROC of 0.994±0.002.

Determined inconclusive intervals Figure 19 shows the determined lower and upper bounds on the probabilistic sigmoid output as a function of the percentages of inconclusive cases in the validation set (development dataset), along with the natural cutoff 0.5. The upper bound curve increases and saturates faster than the lower bound curve with a lower variance across the random splits. First this suggests a disparity in the distribution of predictions below and above the cutoff point. Also the determination of stable lower bounds across the random splits is more difficult than the determination of stable upper bounds. When compared to the PCA-RFC baseline method the width of the inconclusive interval increases more rapidly as the percentage of inconclusive cases increases. That implies that the CNN-RLT method tends to produce relatively less inconclusive cases than the PCA-RFC baseline.

Table 4: Evaluation of the CNN-RLT method on Development dataset. For evaluation, the natural sigmoid cutoff 0.5 was used.

	train set	validation set	test set
Balanced Accuracy	0.982±0.003	0.967±0.008	0.973±0.005
Accuracy	0.982±0.003	0.968±0.008	0.973±0.005
Sensitivity	0.980±0.008	0.951±0.013	0.961±0.014
Specificity	0.983±0.008	0.984±0.005	0.985±0.010
PPV	0.983±0.009	0.982±0.006	0.982±0.012
NPV	0.981±0.008	0.956±0.012	0.966±0.013
AUC-ROC		0.994±0.002	

Transferability of inconclusive intervals The correspondence between the percentage of inconclusive cases in the validation set (development dataset) and the mean±SD percentage of observed inconclusive cases (PIncObs) in the test set (development dataset) is depicted in Figure 20. As for the baseline cases, the deviation of the mean PIncObs in the test set from the identity line is small which can be attributed to the nearly identical distribution of data in both the test and validation sets (due to random splitting).

AUC for balanced accuracy over PIncObs The balanced accuracy (mean±SD) on conclusive cases over the mean PIncObs in the test set (development dataset) is depicted in Figure 21b. The mean of the balanced accuracy rises from about 97.5% when there are around 1% of inconclusive cases in the test set to about 99.5% when there are around 20% of inconclusive cases in the test set. As a result, the CNN-RLT method achieves a relative AUC of 99.02% for the mean balanced accuracy on conclusive cases over the mean PIncObs in the test set of the development dataset. The achieved relative AUC is approximately 2.6% higher than that of the SBR baseline method and around 0.3% higher than the PCA-RFC baseline. The area under the mean of the balanced accuracy is highlighted for better illustration.

PPMI dataset The following results were obtained when evaluating the CNN-RLT method on the PPMI dataset. Figure 22a shows the mean±SD percentage of inconclusive cases observed (PIncObs) in the PPMI test dataset over the percentage of inconclusive cases in the validation set (PIncVal) of the development dataset.

Here the mean PIncObs in the PPMI test dataset deviates only slightly from the identity line. For PIncVal less than 6% the mean PIncObs is slightly below the identity line. Subsequently the mean PIncObs rises slightly above the identity line with an increasing standard deviation of PIncObs. The balanced accuracy on conclusive cases over the mean percentage of observed inconclusive cases (PIncObs) is

presented in Figure 22b. The mean of the balanced accuracy rises from approximately 98.5% when there are around 1% of inconclusive cases in the PPMI test set to about 99.5% when there are around 20% of inconclusive cases in the PPMI test set. The CNN-RLT method achieves a relative AUC of 99.31% for the mean balanced accuracy on conclusive cases over the mean PIncObs in the PPMI test dataset. The achieved relative AUC is approximately 1.8% higher than that of the SBR baseline method and around 0.2% higher than the PCA-RFC baseline. For better illustration the area under the mean of the balanced accuracy is highlighted.

MPH dataset The evaluation of the CNN-RLT method on the MPH dataset produced the following results. Figure 23a illustrates the mean±SD percentage of inconclusive cases observed (PIncObs) in the MPH test dataset over the percentage of inconclusive cases in the validation set (PIncVal) (development dataset). Here the mean of the PIncObs in the MPH test dataset is slightly above the identity line and the standard deviation increases over the PIncVal. The balanced accuracy on conclusive cases over the mean PIncObs is depicted in Figure 23b. The mean of the balanced accuracy slightly increases from approximately 95% when there are around 1% of inconclusive cases in the MPH test set to about 96% when there are around 20% of inconclusive cases in the MPH test set. As a result, the CNN-RLT method achieves a relative AUC of 96.12% for the mean balanced accuracy on conclusive cases over the mean PIncObs in the MPH test dataset. The achieved relative AUC is approximately 2.7% higher than that of the SBR baseline method and around 3.7% higher than the PCA-RFC baseline. For better illustration the area under the mean of the balanced accuracy is highlighted.

5.2.3 CNN-Regression Method Results

Test set of development dataset

Binary classification performance The quantitative performance results of the CNN-Regression classification on the particular subset of the Development dataset are presented in Table 5. In the evaluation process, the natural sigmoid cutoff value of 0.5 was employed. The CNN-Regression method achieves around 96.1% in sensitivity, 98.5% in specificity and a balanced accuracy of 97.5%, with a standard deviation between 0.6-1.1% across random splits, on the test set. The performance results on the validation set are a balanced accuracy of 97.7%, a sensitivity of 98.3% and a specificity of 97.2%. The method achieves a stable AUC-ROC of 0.998±0.001.

Determined inconclusive intervals Figure 24 presents the determined lower and upper bounds on the probabilistic sigmoid output as a function of the percentages of inconclusive cases in the validation set (PIncVal) of the development dataset, along with the natural cutoff 0.5. Similar to the CNN-RLT method, here the upper bound curve increases and saturates slightly faster than the lower bound curve with

Table 5: Evaluation of the CNN-Regression method on Development dataset. For evaluation, the natural sigmoid cutoff 0.5 was used.

	train set	validation set	test set
Balanced Accuracy	0.982+/-0.003	0.977+/-0.006	0.975+/-0.006
Accuracy	0.980+/-0.003	0.977+/-0.007	0.976+/-0.006
Sensitivity	1.000+/-0.000	0.983+/-0.009	0.961+/-0.011
Specificity	0.963+/-0.005	0.972+/-0.009	0.988+/-0.008
PPV	0.960+/-0.005	0.967+/-0.011	0.986+/-0.009
NPV	1.000+/-0.000	0.985+/-0.008	0.967+/-0.010
AUC-ROC		0.998+/-0.001	

a lower variance across the random splits. This suggests a slight disparity in the distribution of predictions below and above the cutoff point. Since both the upper and lower bound functions exhibit a significant standard deviation across the random splits the determination of stable lower and upper bounds is difficult. When compared to the PCA-RFC baseline method the width of the inconclusive interval increases more rapidly over the PIncVal. Therefore the CNN-Regression method also tends to produce relatively less inconclusive cases than the PCA-RFC baseline.

Transferability of inconclusive intervals The correspondence between the PIncVal of the development dataset and the mean±SD percentage of observed inconclusive cases (PIncObs) in the test set of the development dataset is depicted in Figure 25. As for the baseline cases, the deviation of the mean PIncObs in the test set from the identity line is small which can be attributed to the nearly identical distribution of data in both the test and validation sets (due to random splitting).

AUC for balanced accuracy over PIncObs The balanced accuracy (mean±SD) on conclusive cases over the mean PIncObs in the test set (development dataset) is depicted in Figure 26b. The mean of the balanced accuracy rises from about 98% when there is a PIncObs of 1% in the test set to about 99.5% when there is a PIncObs around 20% in the test set. As a result, the CNN-Regression method achieves a relative AUC of 99.23% for the mean balanced accuracy on conclusive cases over the mean PIncObs in the test set of the development dataset. The achieved relative AUC is approximately 2.8% higher than that of the SBR baseline method and around 0.5% higher than the PCA-RFC baseline. The area under the mean of the balanced accuracy is highlighted for better illustration.

PPMI dataset The following results were obtained when evaluating the CNN-Regression method on the PPMI dataset. Figure 27a illustrates the mean±SD

percentage of inconclusive cases observed (PIncObs) in the PPMI test dataset over the percentage of inconclusive cases in the validation set (PIncVal) of the development dataset. Here for lower PIncVal values (less than 5%) the corresponding mean PIncObs in the PPMI test dataset is near the identity line. However for higher PIncVAl values the mean of PIncObs increasingly rises above the identity line and the standard deviation of PIncObs increases strongly. The balanced accuracy on conclusive cases over the mean PIncObs is presented in Figure 27b. The mean of the balanced accuracy rises from approximately 98.5% when there are around 1% of inconclusive cases in the PPMI test set to about 99.5% when there are around 20% of inconclusive cases in the PPMI test set. The CNN-Regression method achieves a relative AUC of 99.38% for the mean balanced accuracy on conclusive cases over the mean PIncObs in the PPMI test dataset. The achieved relative AUC is approximately 1.9% higher than that of the SBR baseline method and around 0.3% higher than the PCA-RFC baseline. For better illustration the area under the mean of the balanced accuracy is highlighted.

MPH dataset The evaluation of the CNN-Regression method on the MPH dataset produced the following results. Figure 28a demonstrates the mean±SD percentage of inconclusive cases observed (PIncObs) in the MPH test dataset over the percentage of inconclusive cases in the validation set (PIncVal) (development dataset). Here the mean of the PIncObs in the MPH test dataset is above the identity line and deviates stronger from the identity line as the PIncVal increases. Also the standard deviation of the PIncObs is high and increases over the increasing PIncVal. The balanced accuracy on conclusive cases over the mean PIncObs is depicted in Figure 28b. The mean of the balanced accuracy slightly increases from approximately 95% when there are around 1% of inconclusive cases in the MPH test set to about 96.5% when there are around 20% of inconclusive cases in the MPH test set. As a result, the CNN-Regression method achieves a relative AUC of 96.24% for the mean balanced accuracy on conclusive cases over the mean PIncObs in the MPH test dataset. The achieved relative AUC is approximately 2.8% higher than that of the SBR baseline method and around 3.8% higher than the PCA-RFC baseline. For better illustration the area under the mean of the balanced accuracy is highlighted.

5.3 Comparative Performance Analysis

In this section, a summary comparison of the performance between the baseline and experimental methods is presented. The comparison focuses on two aspects: transferability of inconclusive intervals (in both validation and test sets) and the AUC of balanced accuracy on conclusive cases across varying percentages of observed inconclusive cases (PIncObs). To support the analysis visually one comparison figure is used for each aspect tested on a specific dataset. The comparison is carried out for the test set of the development data, the PPMI dataset and the MPH dataset, respectively.

5.3.1 Performance on test set of development dataset

Figure 29 provides a comparison of the transferability of the inconclusive intervals from the validation set to the test set (development data) along the baseline and experimental methods. For each considered method the mean of the percentage of observed inconclusive cases (PIncObs) hardly deviates from the identity line. The similarity in data distribution of the validation and test set due to random splitting is an explanation for that. The standard deviation of PIncObs is also similarly low across the methods. Thus the PIncObs is hardly affected by the randomness of the inconclusive intervals across random splits for each method. The mean of PIncObs can be reliably used for the calculation of the main metric of this work compared in the following.

In Figure 30, the performance comparison of the methods on the test set (development dataset) concerning the main metric of this work, the relative AUC for the mean balanced accuracy on conclusive cases over the mean PIncObs in the test set, is shown. In general, the CNN-based methods outperform both baseline methods, the SBR method and PCA-RFC. The highest performance is achieved by the CNN-Regression (relative AUC: 99.23%) method whereas the lowest AUC is that for the SBR-based method (relative AUC: 96.38%). The CNN-RLT method achieves slightly higher performance than the CNN-MVT.

5.3.2 Performance on PPMI dataset

Figure 31 shows a comparison of the transferability of the inconclusive intervals from the validation set to the PPMI test dataset along the baseline and experimental methods. The percentage of observed inconclusive cases (PIncObs) of CNN-based methods shows a higher standard deviation compared to the baseline methods. A possible explanation for that is the higher sensitivity of the CNN-based methods to the variability in inconclusive intervals across random splits. Also the mean of PIncObs deviates stronger from the identity line (above identity line) for the CNN-MVT and CNN-Regression methods compared to the baseline methods and CNN-RLT. That indicates that, on average, the CNN-MVT and CNN-Regression methods are supposedly less certain about the PPMI dataset predictions compared to the other methods.

The performance comparison of the methods concerning the relative AUC for the mean balanced accuracy on conclusive cases over the PIncObs in the PPMI dataset is depicted in Figure 32. The CNN-based methods outperform both baseline methods, the SBR method and PCA-RFC. The highest performance is achieved by the CNN-Regression (relative AUC: 99.38%) method whereas the lowest AUC is that for the SBR-based method (relative AUC: 97.51%). The CNN-RLT method achieves slightly higher performance (relative AUC: 99.31%) than the CNN-MVT (relative AUC: 99.23%).

5.3.3 Performance on MPH dataset

Figure 33 illustrates a comparison of the transferability of the inconclusive intervals from the validation set to the MPH test dataset along the baseline and experimental methods. As for the PPMI dataset, on the MPH dataset the percentage of observed inconclusive cases (PIncObs) of CNN-based methods shows a higher standard deviation compared to the baseline methods. The higher sensitivity of the CNN-based methods to the variability in inconclusive intervals across random splits is a possible explanation. Here also the mean of PIncObs deviates stronger from the identity line (above identity line) for the CNN-MVT and CNN-Regression methods compared to the CNN-RLT. That indicates that, on average, the CNN-MVT and CNN-Regression methods are supposedly less certain about the MPH dataset predictions compared to the CNN-RLT method. However the highest deviation of the mean PIncObs from the identity line shows the baseline PCA-RFC method and thus shows the lowest supposed certainty about the MPH dataset predictions, on average.

In Figure 34 the performance comparison of the methods concerning the relative AUC for the mean balanced accuracy on conclusive cases over the PIncObs in the MPH dataset is presented. As for the other test datasets, CNN-based methods outperform both baseline methods, the SBR method and PCA-RFC. The highest performance is achieved by the CNN-Regression (relative AUC: 96.24%) method whereas the lowest AUC is that for the baseline PCA-RFC method (relative AUC: 92.42%). The CNN-RLT method achieves slightly higher performance (relative AUC: 96.12%) than the CNN-MVT (relative AUC: 95.73%).

5.4 Conclusion

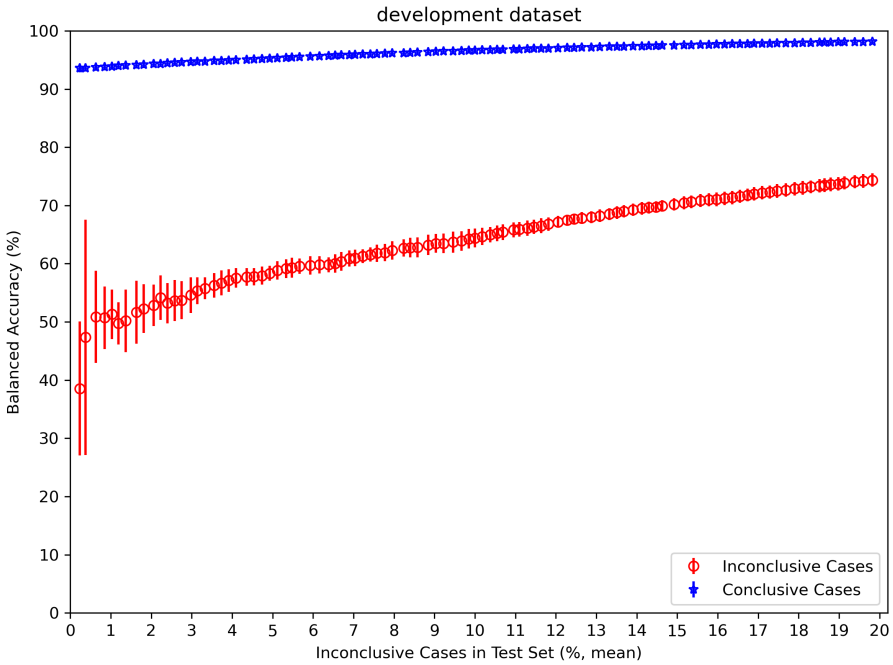
A concluding overview of the performance of the methods on different test datasets concerning the relative AUC for the mean balanced accuracy on conclusive cases over the PIncObs is summarized in Figure 35. The CNN-based methods outperform the baseline, especially on the MPH dataset. The best performance is achieved by the CNN-Regression method, followed by the CNN-RLT and CNN-MVT methods.

6 Discussion

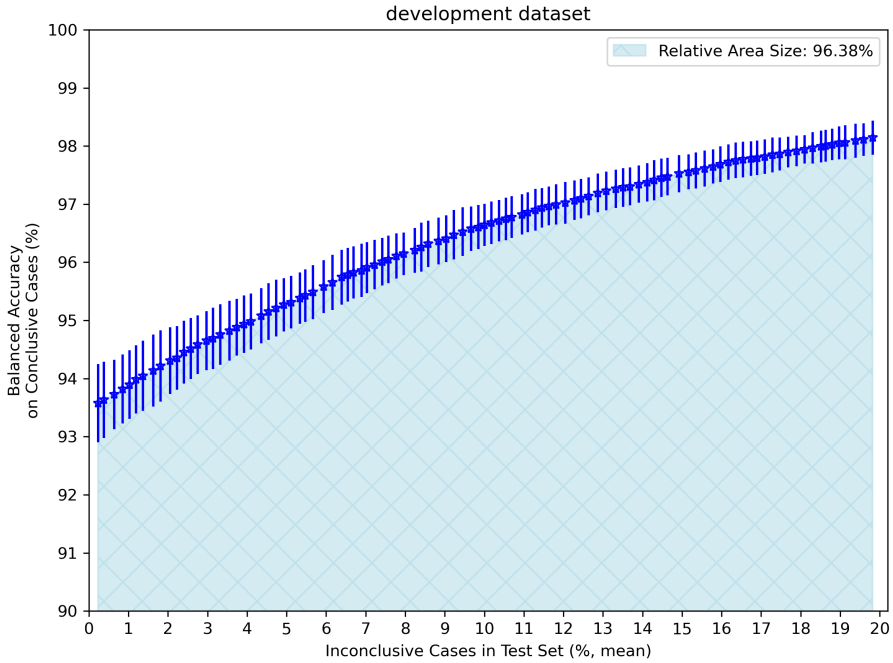
Interpretation: - PPMI is easier to classify than test set of dev data and MPH

Limitations: - bACC-AUC metric may be less human interpretable compared to the standard AUC, however when usefull to compare performance of classification models - Higher variance across random splits is a weak point of the bAcc-AUC metric..

7 Conclusion

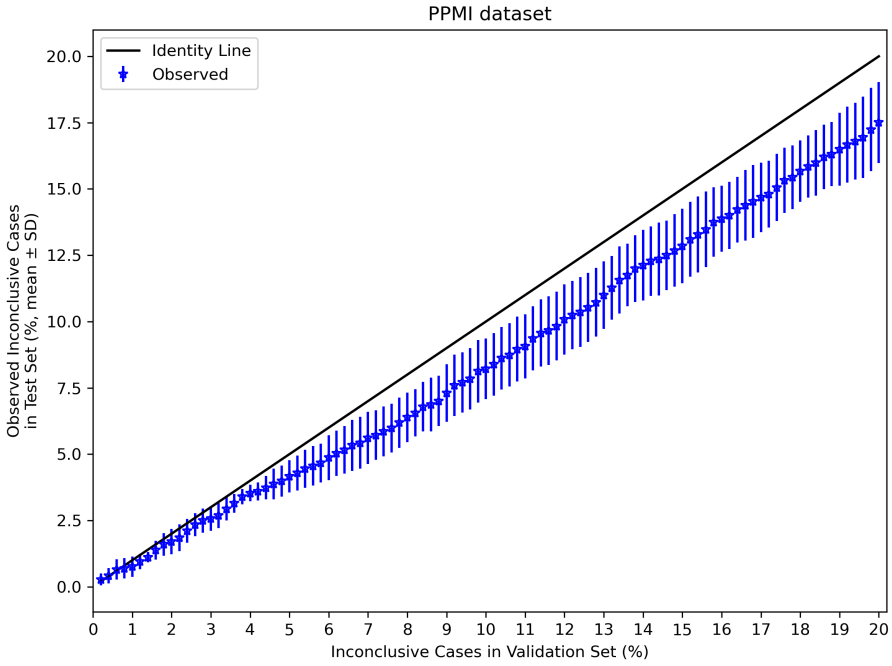


(a)

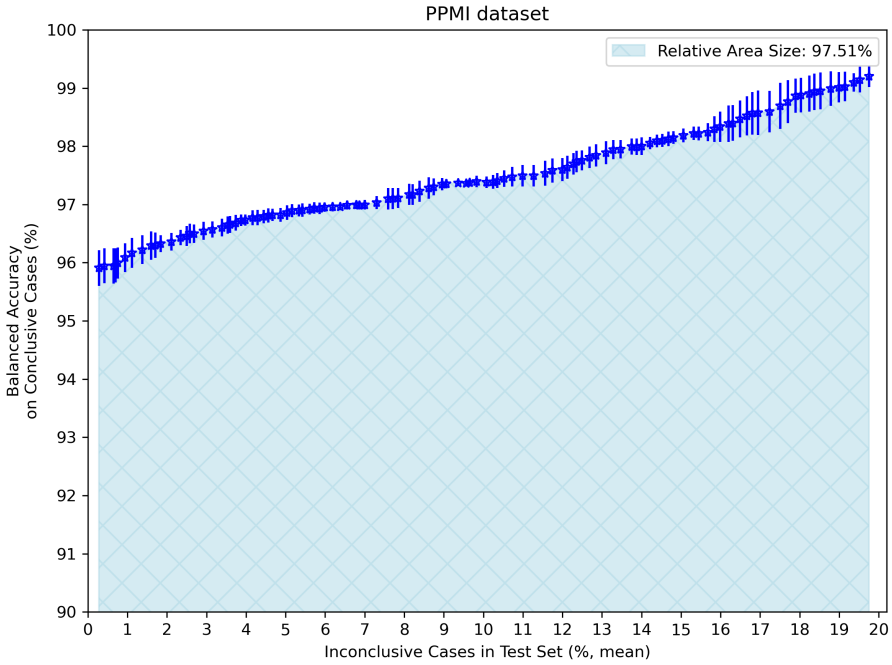


(b)

Figure 6: Evaluation of the SBR method on Test Set of Development dataset. Balanced accuracy for a given mean percentage of observed inconclusive cases in the test set on (a) both conclusive and inconclusive cases and (b) only conclusive cases. Each of the mean percentages of observed inconclusive cases is associated with an inconclusive range (determined in the validation set).

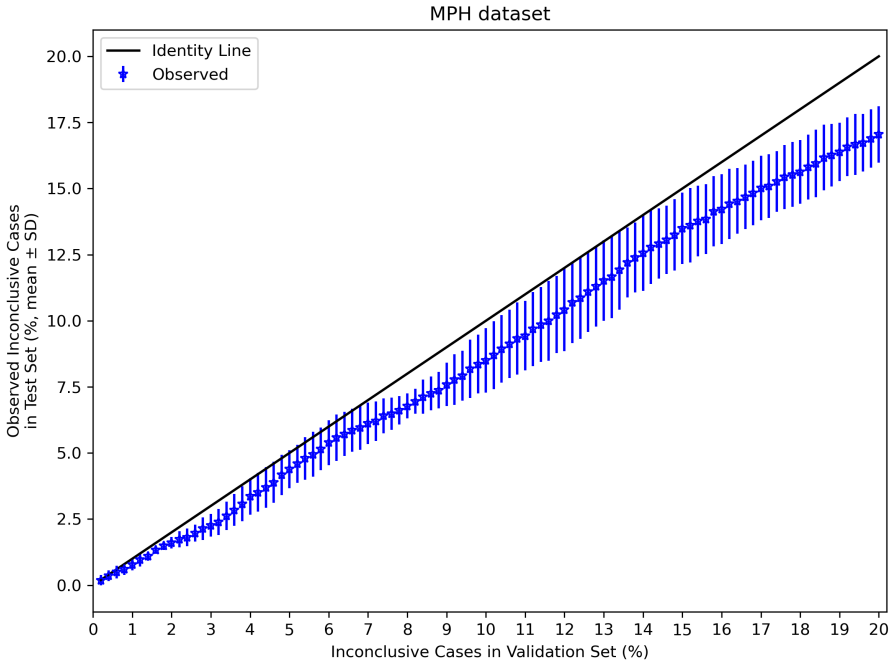


(a) Observed percentage of inconclusive cases in the PPMI dataset for a given set of percentages of inconclusive cases in the validation set (Development dataset).

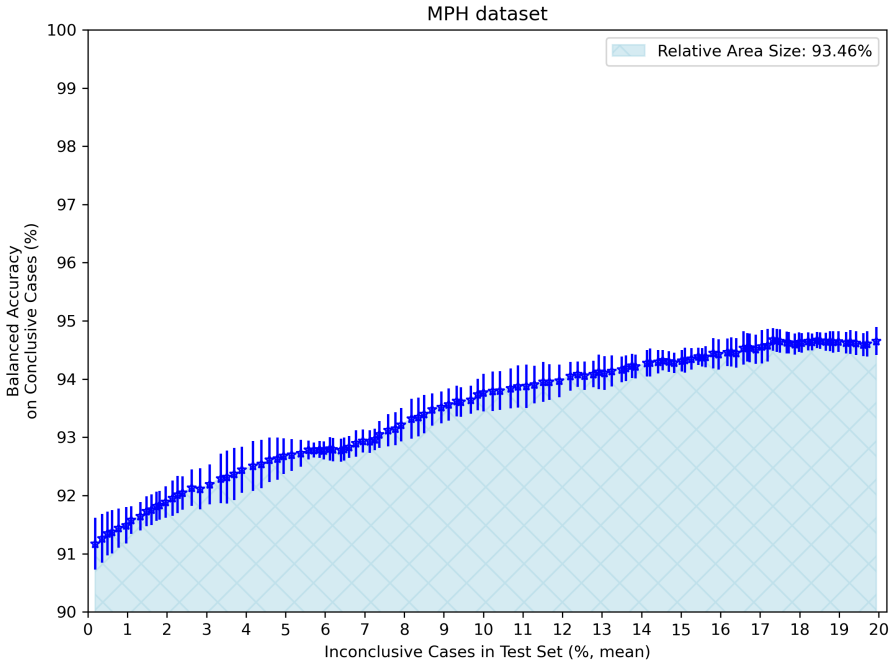


(b) Balanced accuracy on conclusive cases for a given mean percentage of inconclusive cases observed in the PPMI dataset.

Figure 7: Evaluation of the SBR method on PPMI dataset.



(a) Observed percentage of inconclusive cases in the MPH dataset for a given set of percentages of inconclusive cases in the validation set (Development dataset).



(b) Balanced accuracy on conclusive cases for a given mean percentage of inconclusive cases observed in the MPH dataset.

Figure 8: Evaluation of the SBR method on MPH dataset.

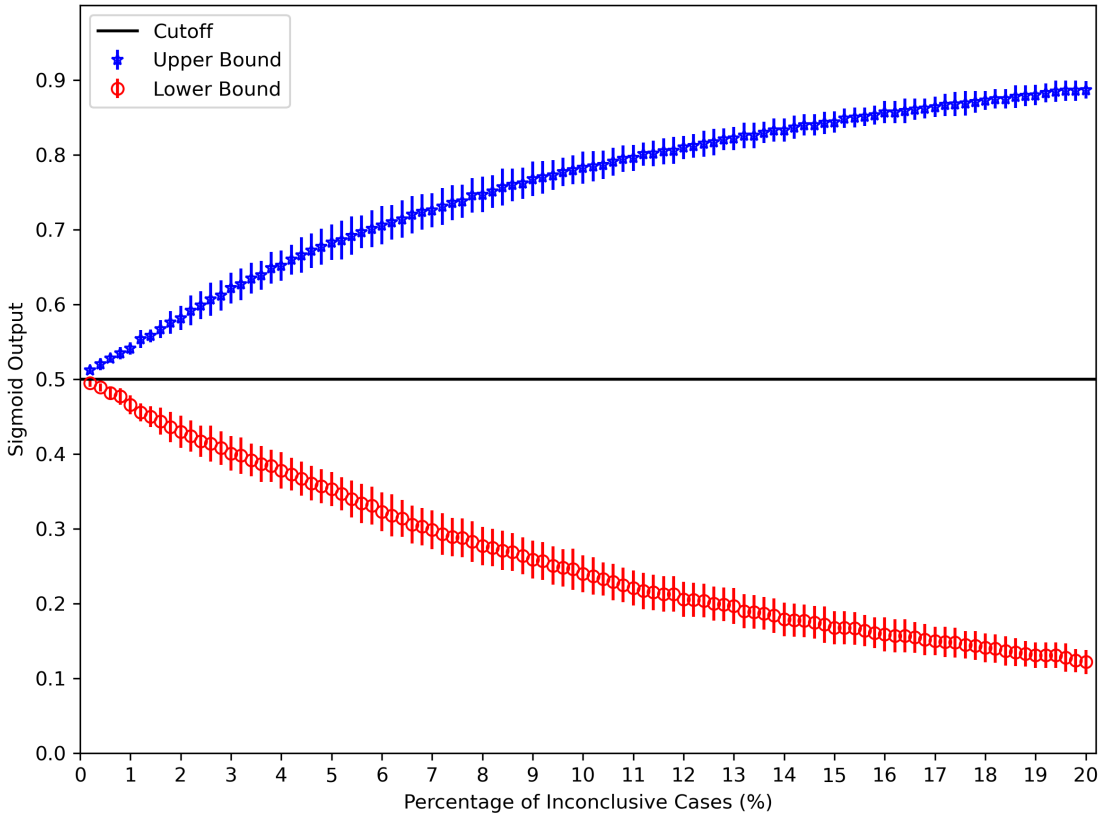


Figure 9: Evaluation of the PCA-RFC method on Test Set of Development dataset. Determined upper and lower bounds of the inconclusive interval as a function of the percentage of inconclusive cases.

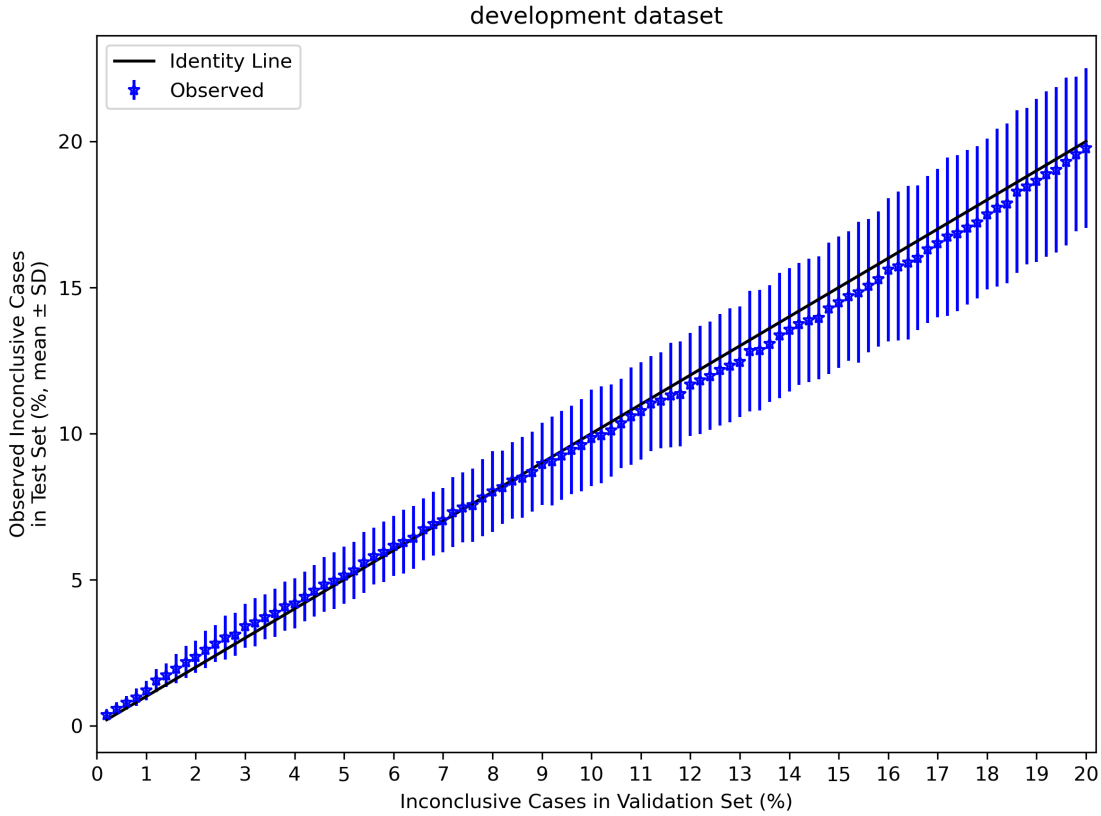
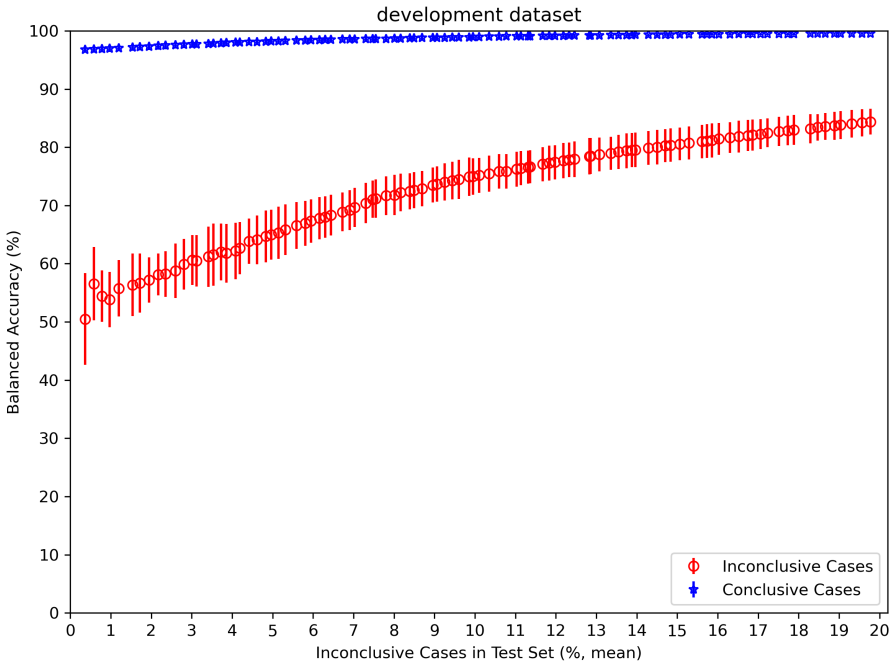
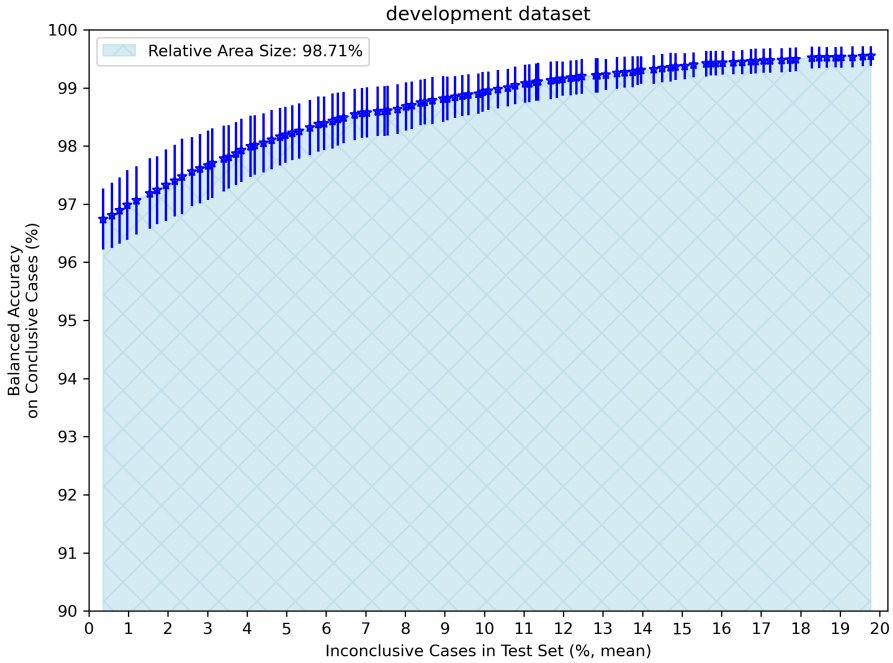


Figure 10: Evaluation of the PCA-RFC method on Test Set of Development dataset. Observed percentage of inconclusive cases in the test set for a given set of percentages of inconclusive cases in the validation set. Each of the percentages of inconclusive cases in the validation set is associated with an inconclusive range (determined in the validation set).

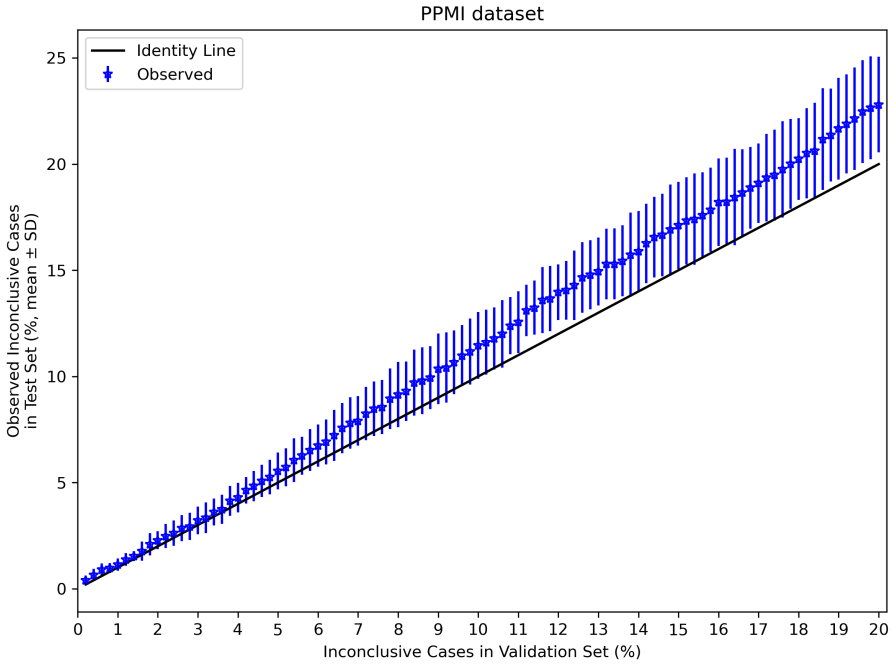


(a)

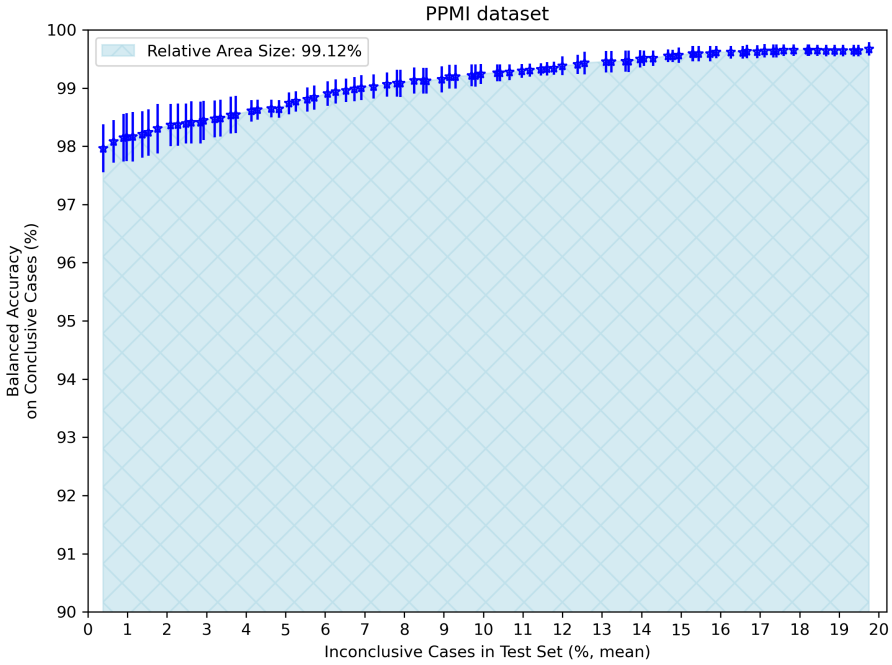


(b)

Figure 11: Evaluation of the PCA-RFC method on Test Set of Development dataset. Balanced accuracy for a given mean percentage of observed inconclusive cases in the test set on (a) both conclusive and inconclusive cases and (b) only conclusive cases. Each of the mean percentages of observed inconclusive cases is associated with an inconclusive range (determined in the validation set).

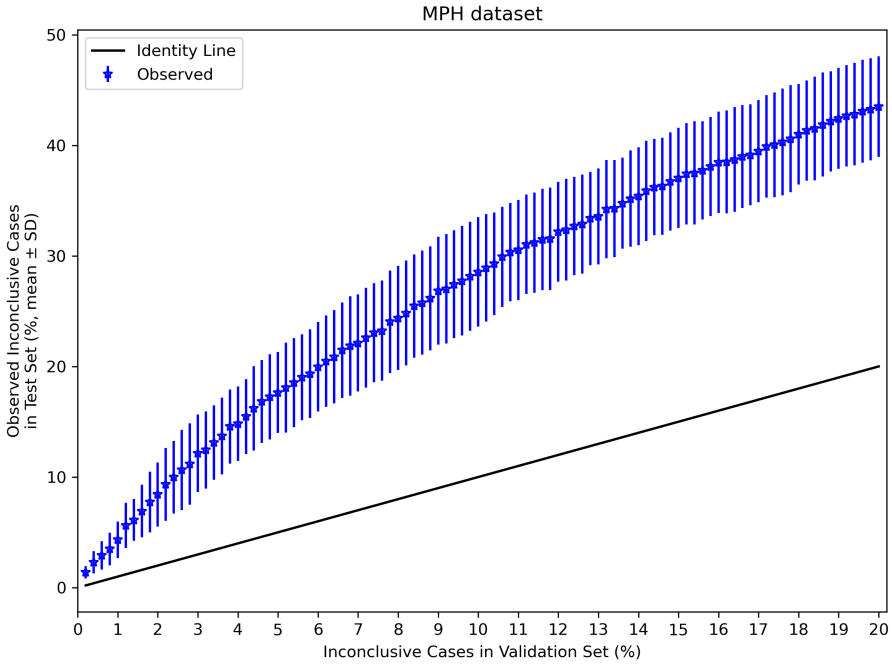


(a) Observed percentage of inconclusive cases in the PPMI dataset for a given set of percentages of inconclusive cases in the validation set (Development dataset).

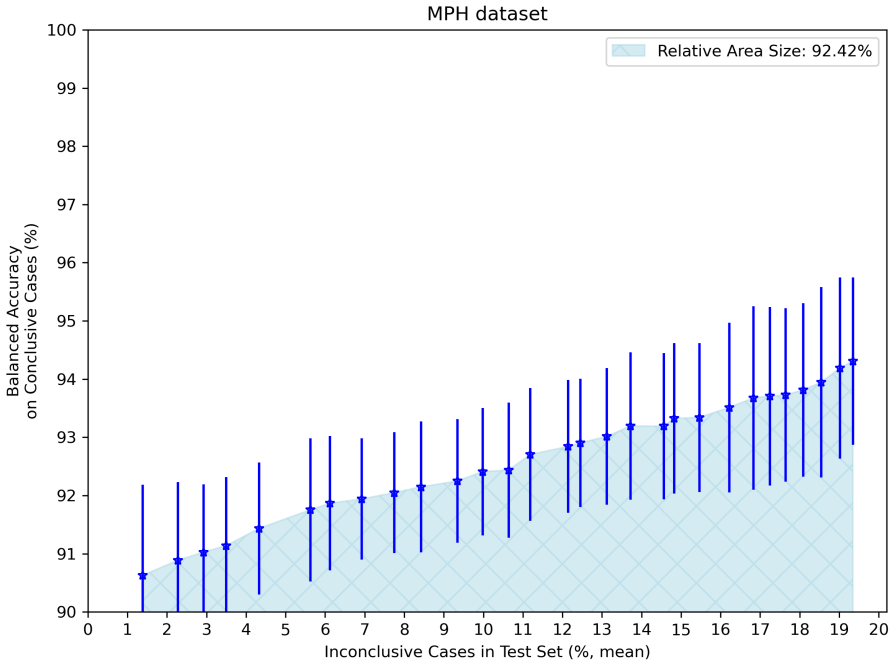


(b) Balanced accuracy on conclusive cases for a given mean percentage of inconclusive cases observed in the PPMI dataset.

Figure 12: Evaluation of the PCA-RFC method on PPMI dataset.



(a) Observed percentage of inconclusive cases in the MPH dataset for a given set of percentages of inconclusive cases in the validation set (Development dataset).



(b) Balanced accuracy on conclusive cases for a given mean percentage of inconclusive cases observed in the MPH dataset.

Figure 13: Evaluation of the PCA-RFC method on MPH dataset.

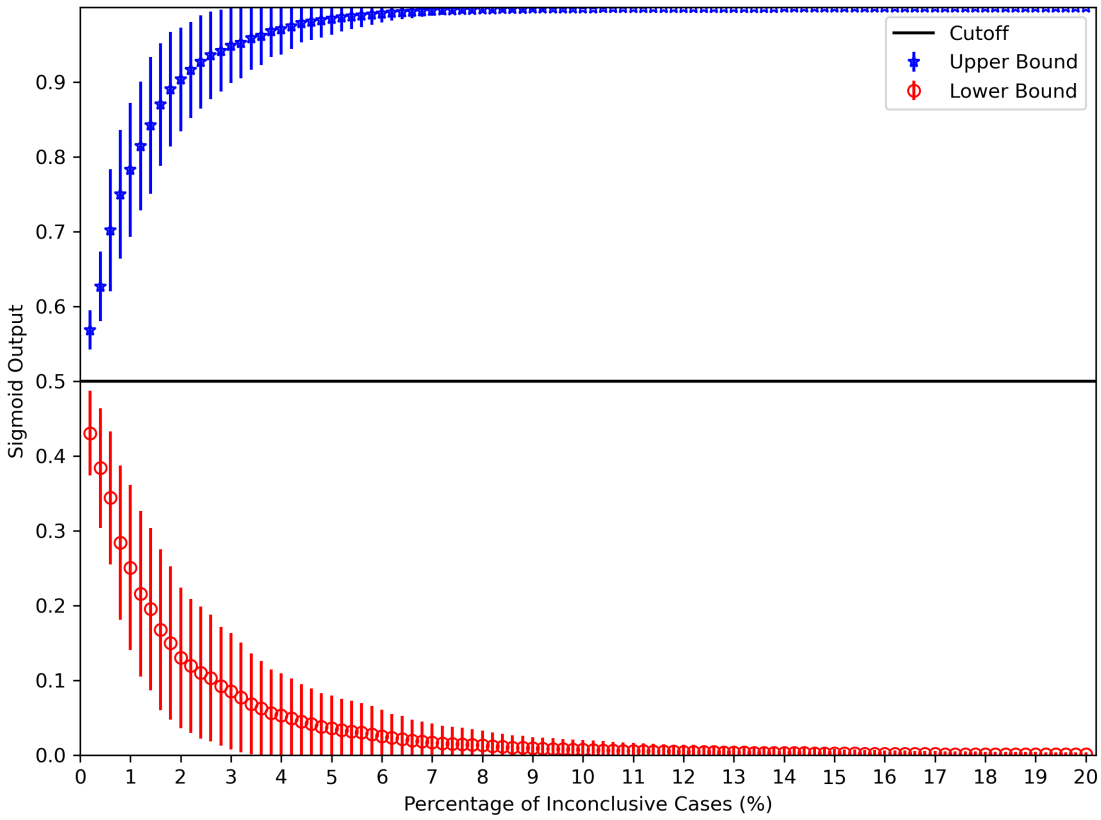


Figure 14: Evaluation of the CNN-MVT method on Test Set of Development dataset. Determined upper and lower bounds of the inconclusive interval as a function of the percentage of inconclusive cases.

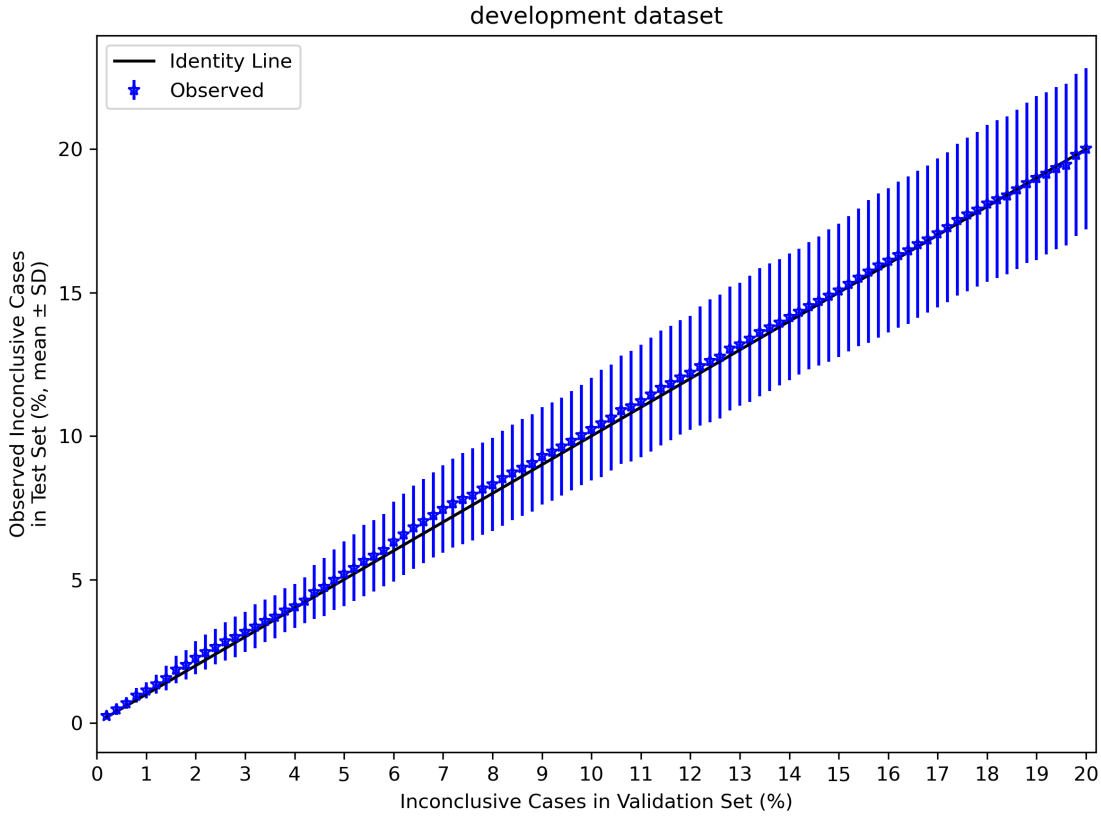
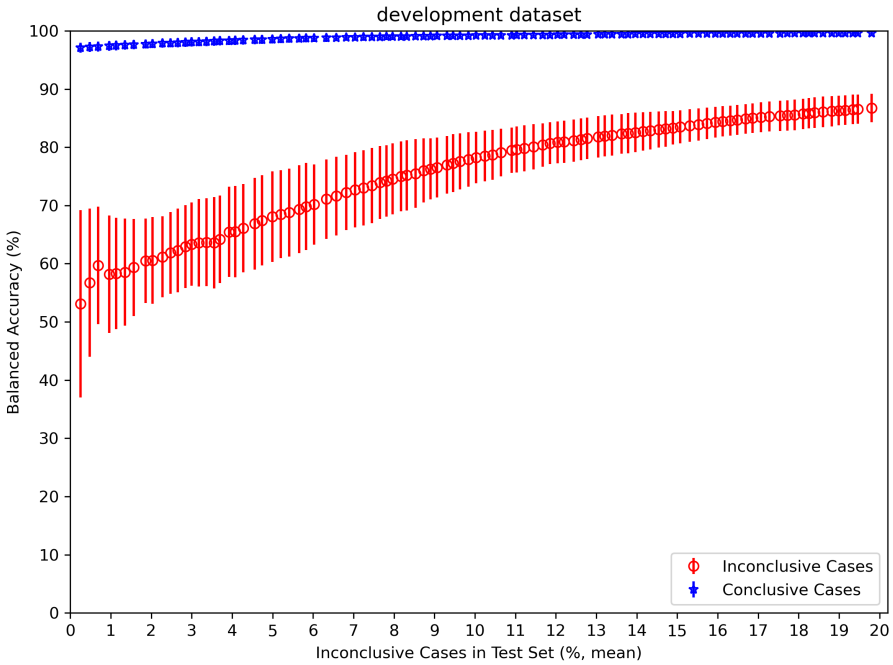
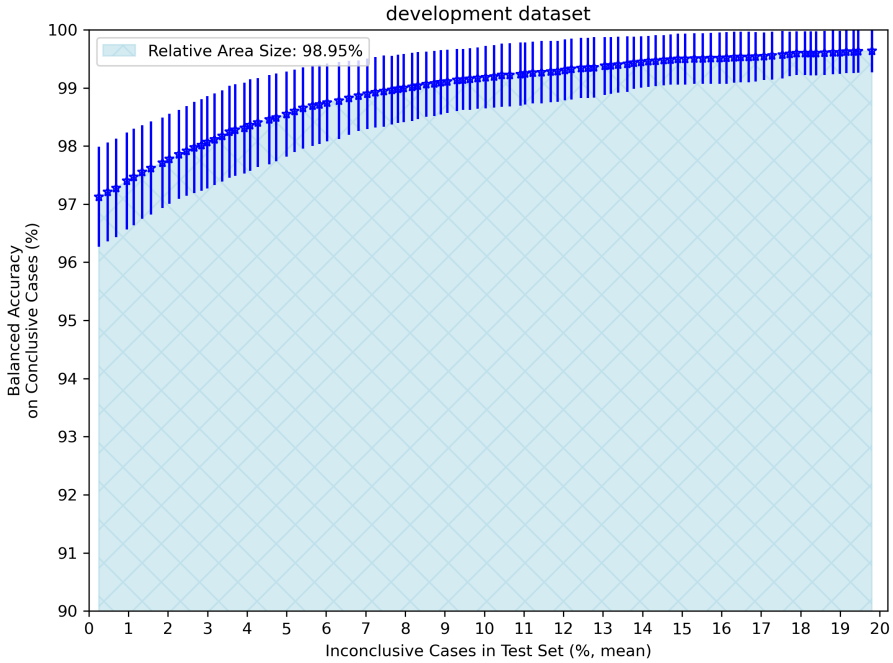


Figure 15: Evaluation of the CNN-MVT method on Test Set of Development dataset. Observed percentage of inconclusive cases in the test set for a given set of percentages of inconclusive cases in the validation set. Each of the percentages of inconclusive cases in the validation set is associated with an inconclusive range (determined in the validation set).

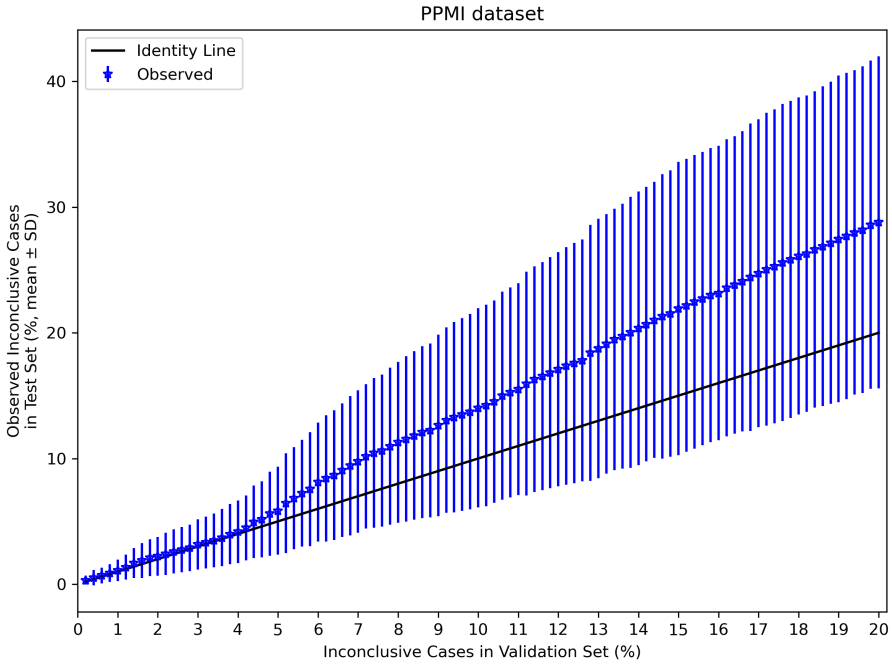


(a)

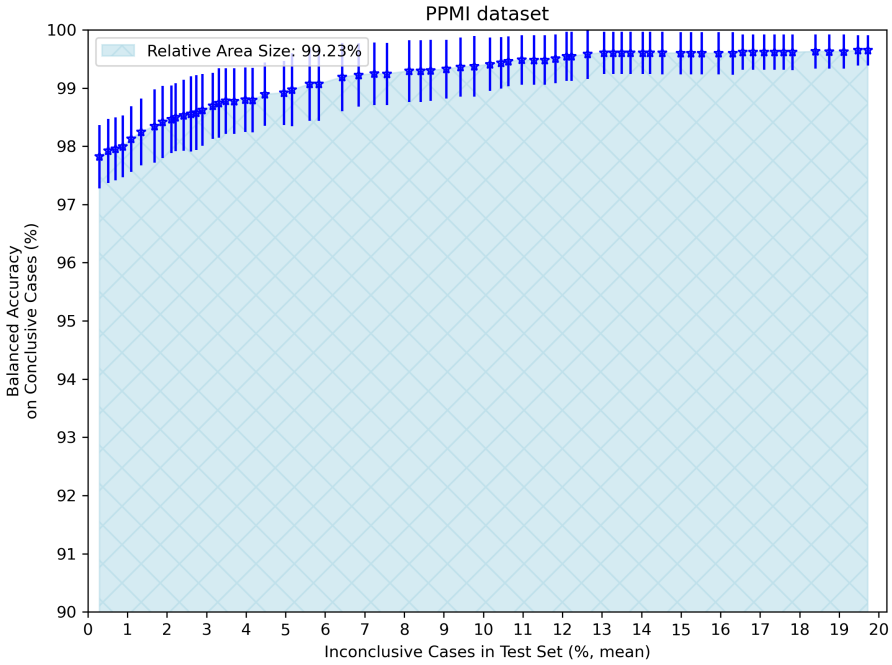


(b)

Figure 16: Evaluation of the CNN-MVT method on Test Set of Development dataset. Balanced accuracy for a given mean percentage of observed inconclusive cases in the test set on (a) both conclusive and inconclusive cases and (b) only conclusive cases. Each of the mean percentages of observed inconclusive cases is associated with an inconclusive range (determined in the validation set).

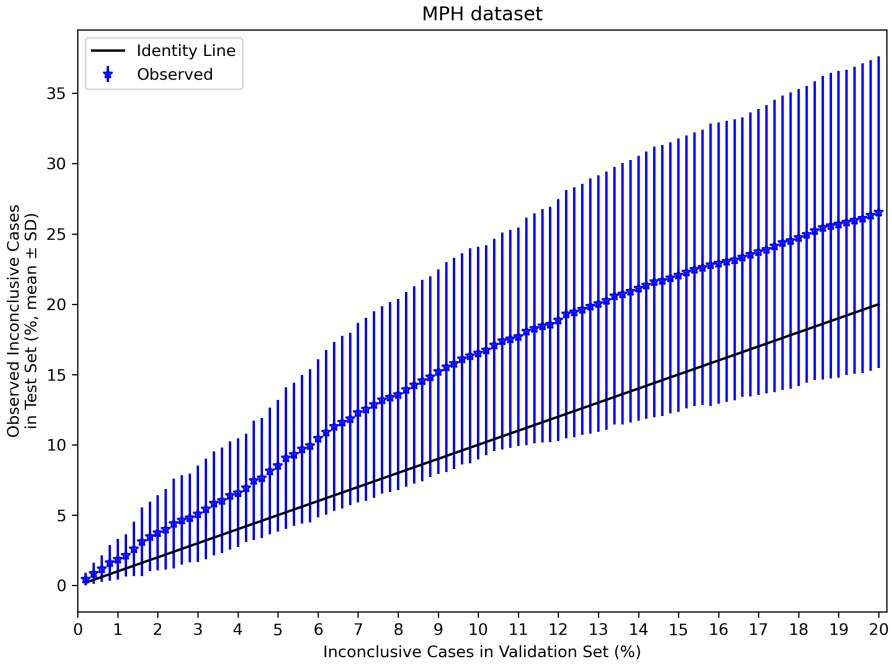


(a) Observed percentage of inconclusive cases in the PPMI dataset for a given set of percentages of inconclusive cases in the validation set (Development dataset).

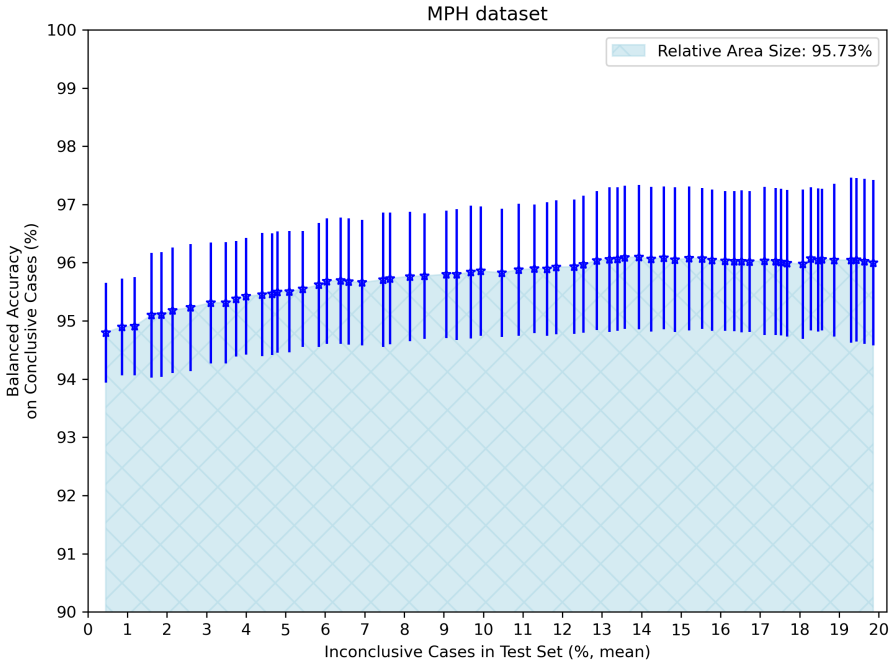


(b) Balanced accuracy on conclusive cases for a given mean percentage of inconclusive cases observed in the PPMI dataset.

Figure 17: Evaluation of the CNN-MVT method on PPMI dataset.



(a) Observed percentage of inconclusive cases in the MPH dataset for a given set of percentages of inconclusive cases in the validation set (Development dataset).



(b) Balanced accuracy on conclusive cases for a given mean percentage of inconclusive cases observed in the MPH dataset.

Figure 18: Evaluation of the CNN-MVT method on MPH dataset.

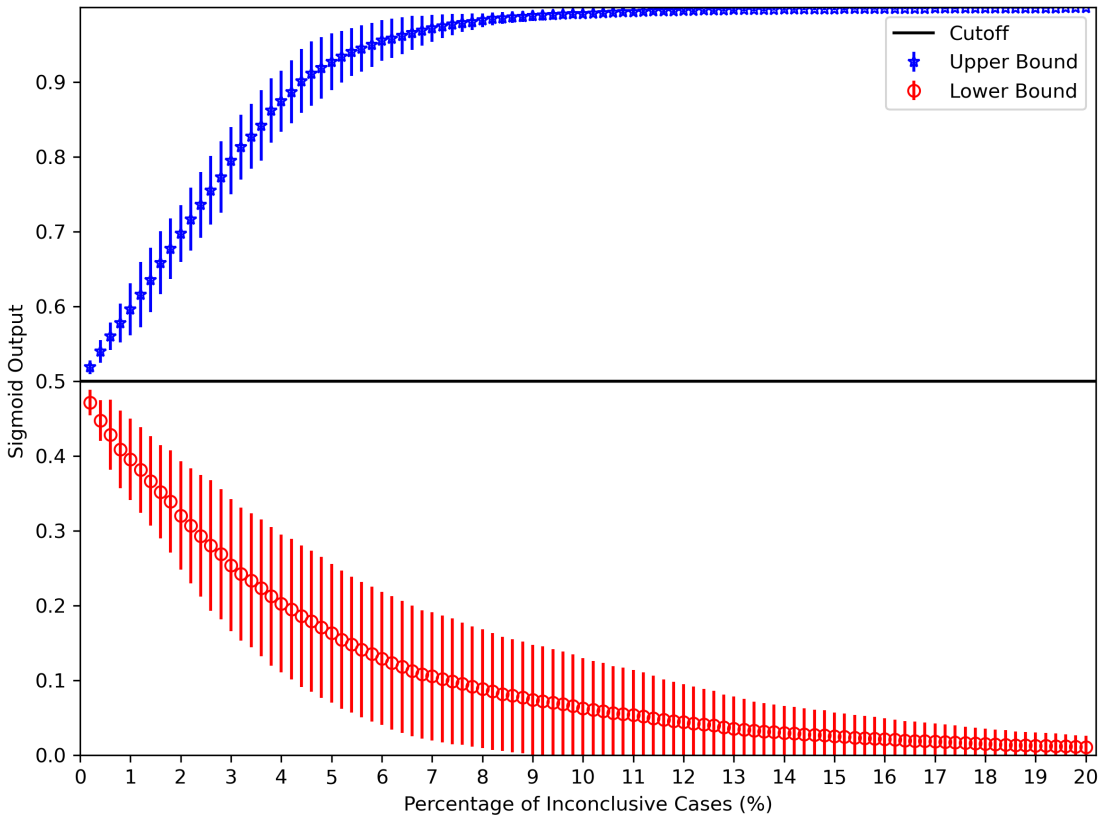


Figure 19: Evaluation of the CNN-RLT method on Test Set of Development dataset. Determined upper and lower bounds of the inconclusive interval as a function of the percentage of inconclusive cases.

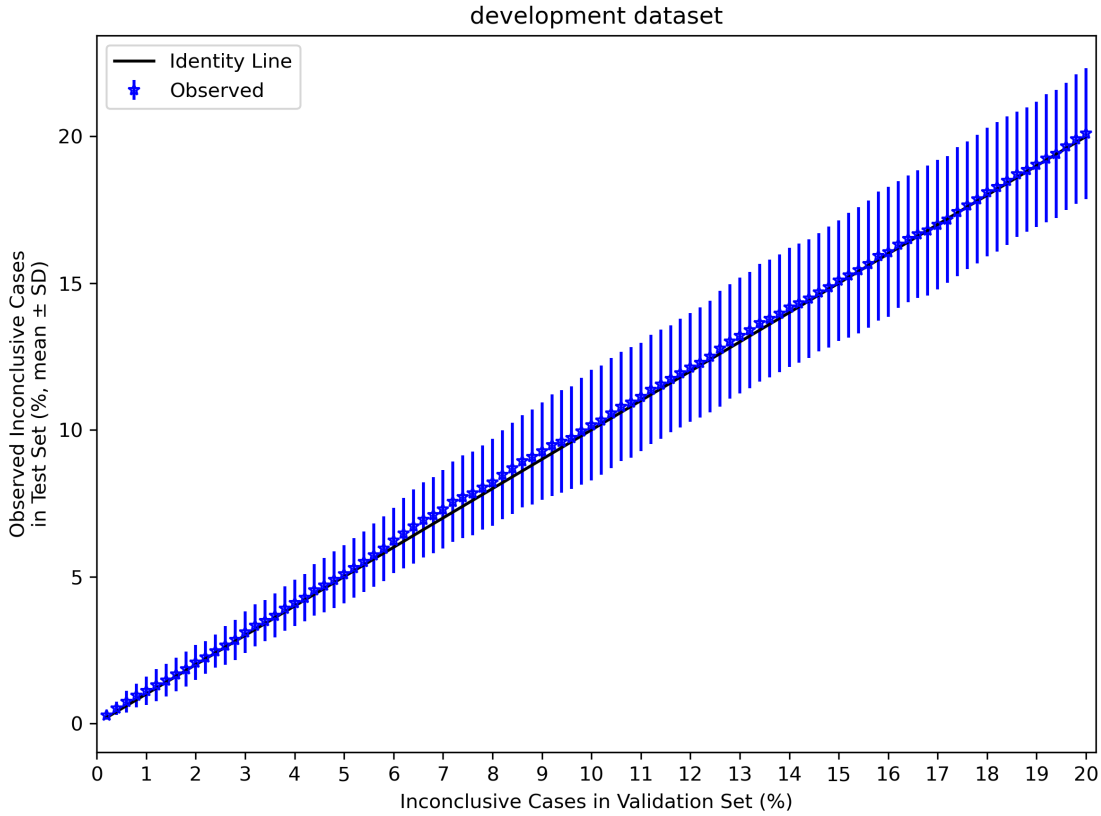
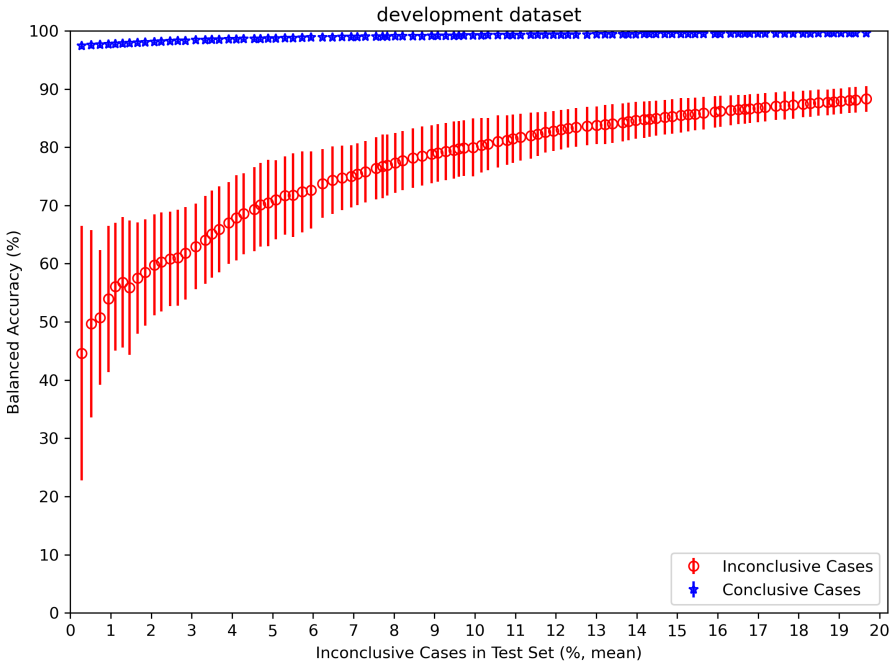
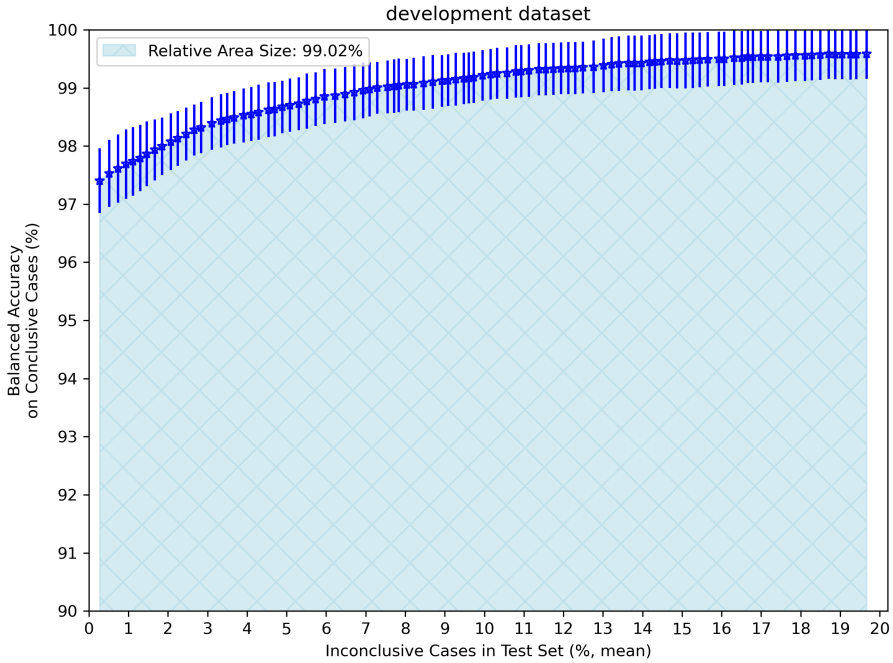


Figure 20: Evaluation of the CNN-RLT method on Test Set of Development dataset. Observed percentage of inconclusive cases in the test set for a given set of percentages of inconclusive cases in the validation set. Each of the percentages of inconclusive cases in the validation set is associated with an inconclusive range (determined in the validation set).

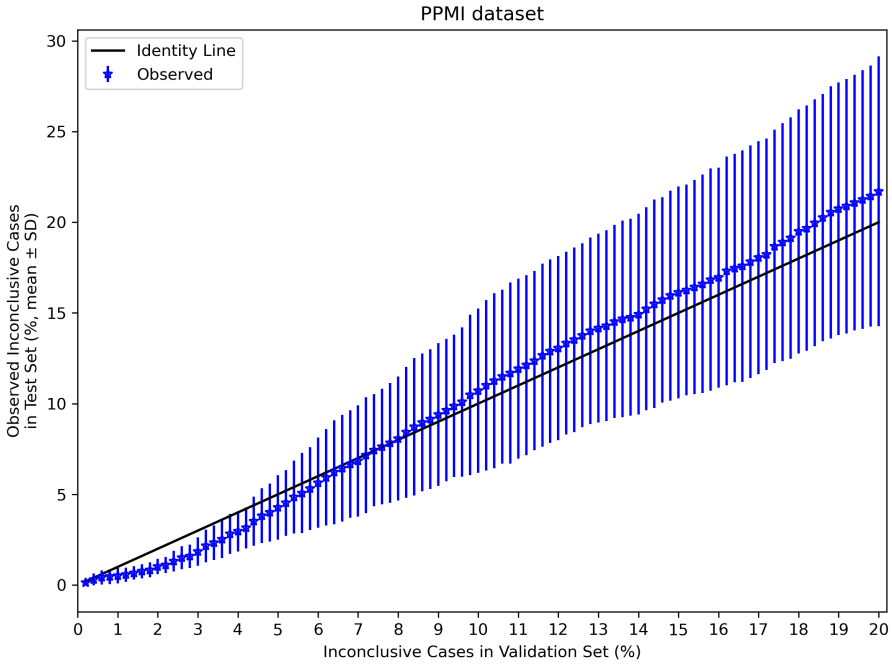


(a)

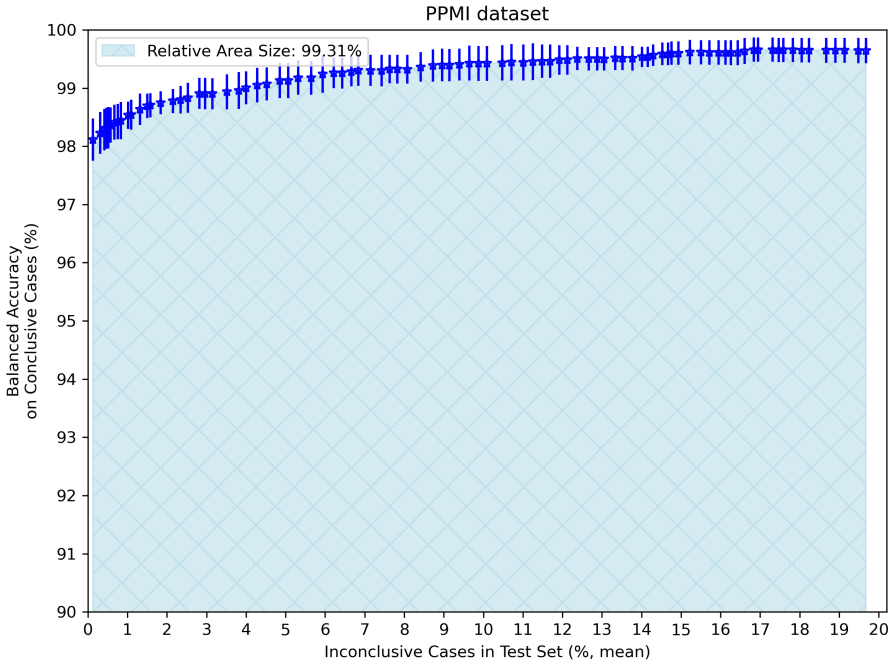


(b)

Figure 21: Evaluation of the CNN-RLT method on Test Set of Development dataset. Balanced accuracy for a given mean percentage of observed inconclusive cases in the test set on (a) both conclusive and inconclusive cases and (b) only conclusive cases. Each of the mean percentages of observed inconclusive cases is associated with an inconclusive range (determined in the validation set).

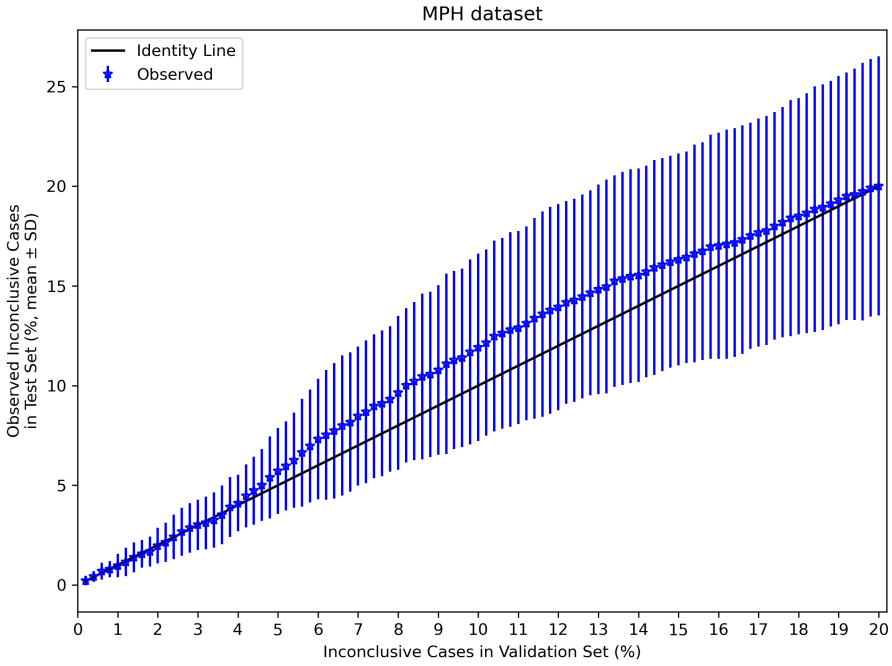


(a) Observed percentage of inconclusive cases in the PPMI dataset for a given set of percentages of inconclusive cases in the validation set (Development dataset).

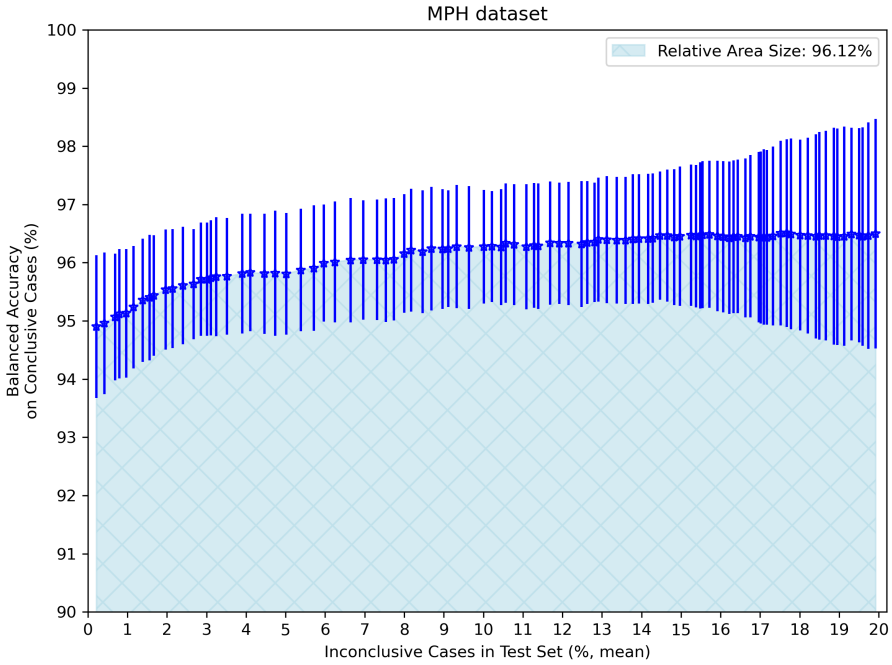


(b) Balanced accuracy on conclusive cases for a given mean percentage of inconclusive cases observed in the PPMI dataset.

Figure 22: Evaluation of the CNN-RLT method on PPMI dataset.



(a) Observed percentage of inconclusive cases in the MPH dataset for a given set of percentages of inconclusive cases in the validation set (Development dataset).



(b) Balanced accuracy on conclusive cases for a given mean percentage of inconclusive cases observed in the MPH dataset.

Figure 23: Evaluation of the CNN-RLT method on MPH dataset.

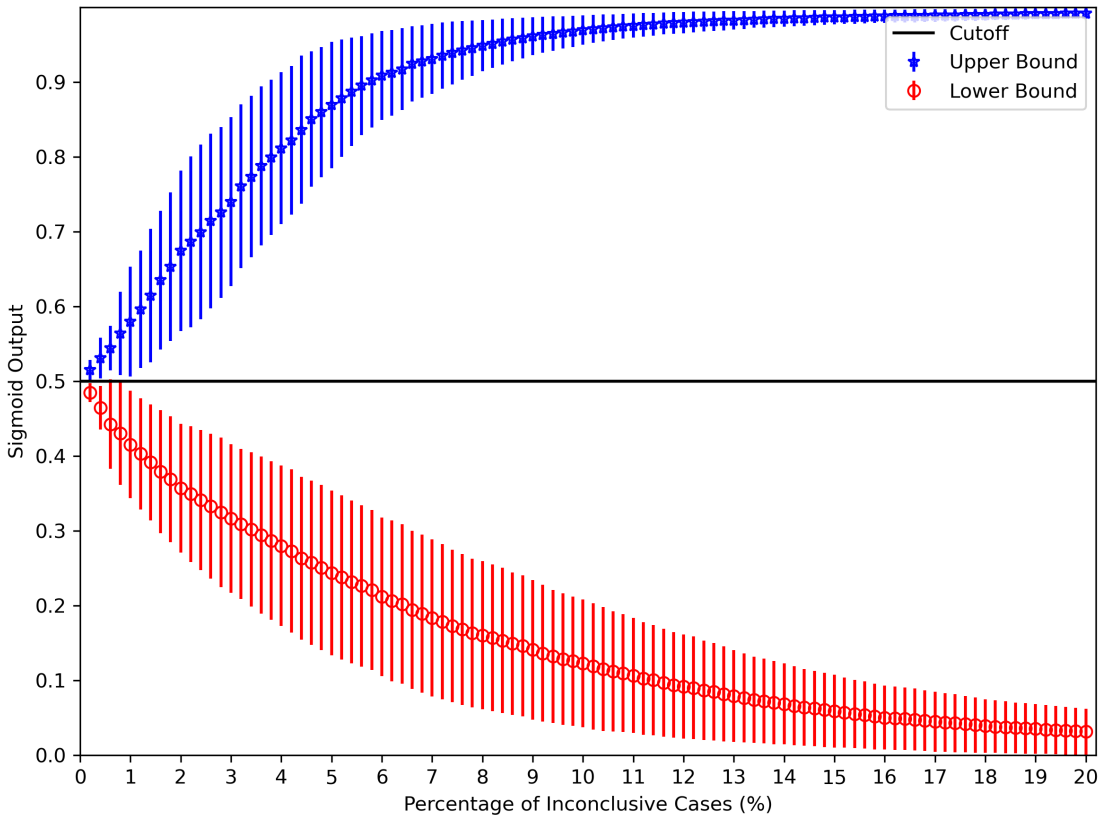


Figure 24: Evaluation of the CNN-Regression method on Test Set of Development dataset. Determined upper and lower bounds of the inconclusive interval as a function of the percentage of inconclusive cases.

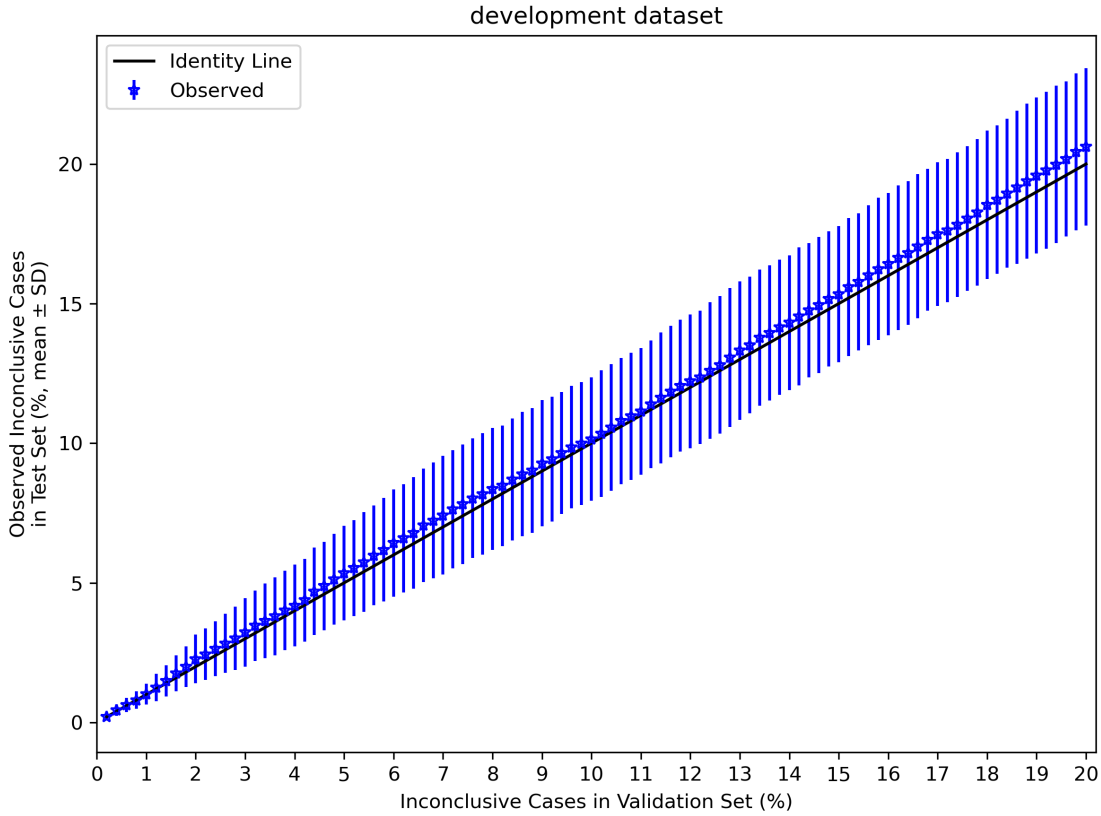
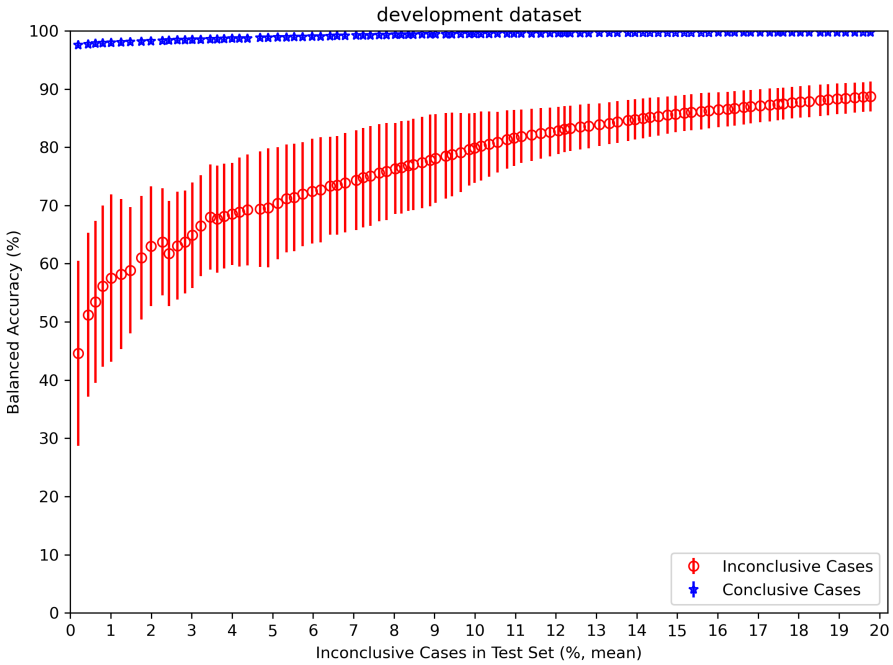
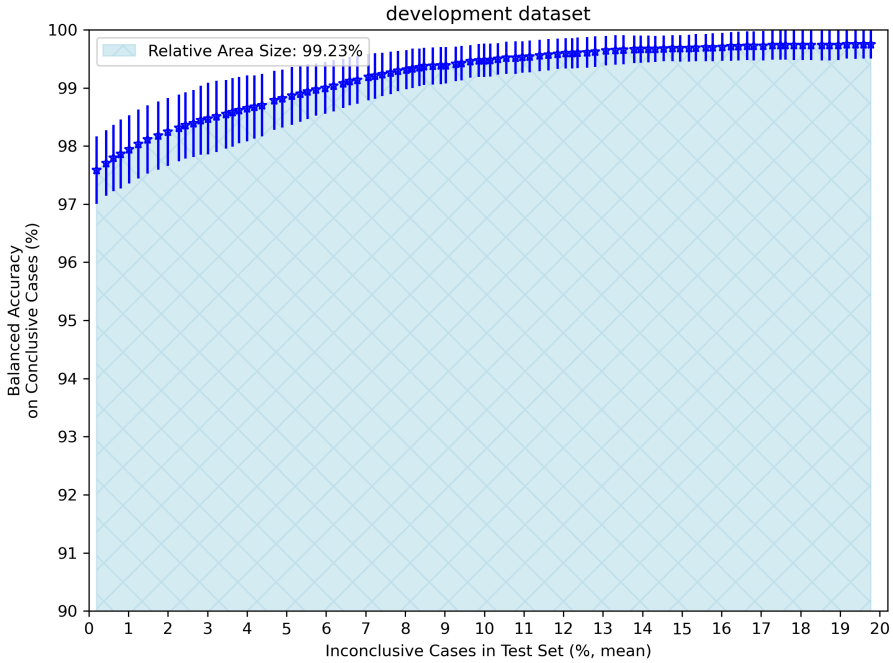


Figure 25: Evaluation of the CNN-Regression method on Test Set of Development dataset. Observed percentage of inconclusive cases in the test set for a given set of percentages of inconclusive cases in the validation set. Each of the percentages of inconclusive cases in the validation set is associated with an inconclusive range (determined in the validation set).

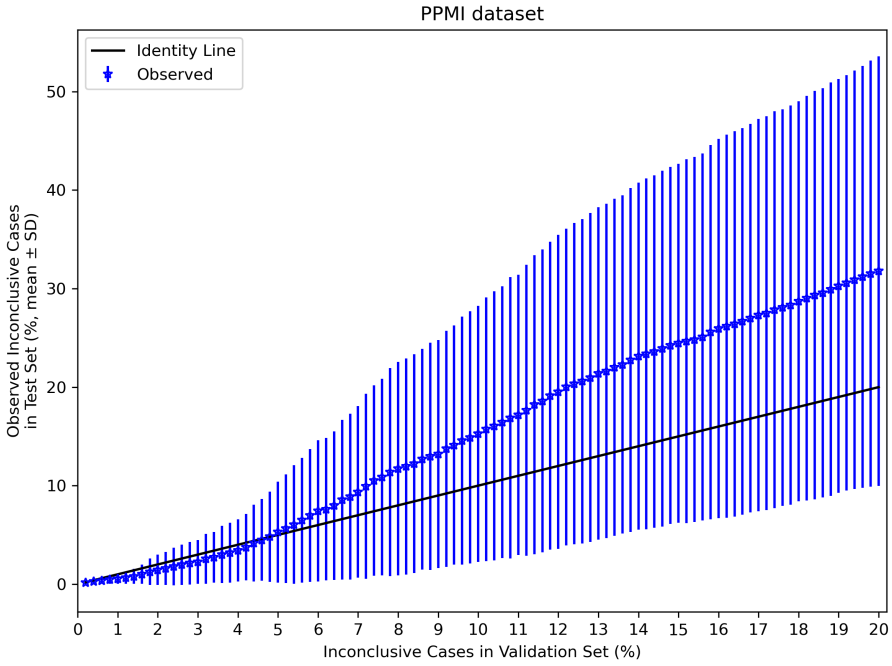


(a)

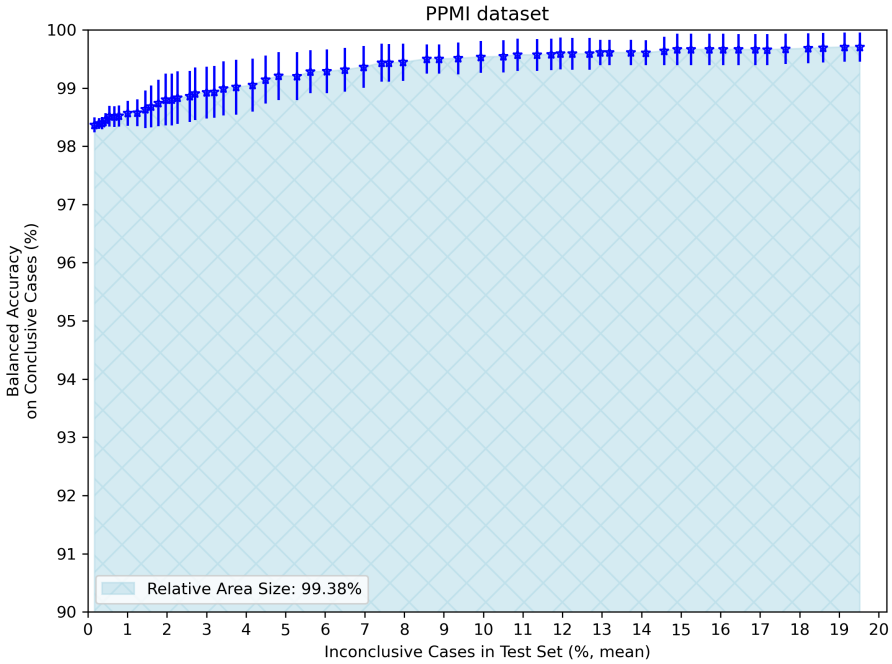


(b)

Figure 26: Evaluation of the CNN-Regression method on Test Set of Development dataset. Balanced accuracy for a given mean percentage of observed inconclusive cases in the test set on (a) both conclusive and inconclusive cases and (b) only conclusive cases. Each of the mean percentages of observed inconclusive cases is associated with an inconclusive range (determined in the validation set).

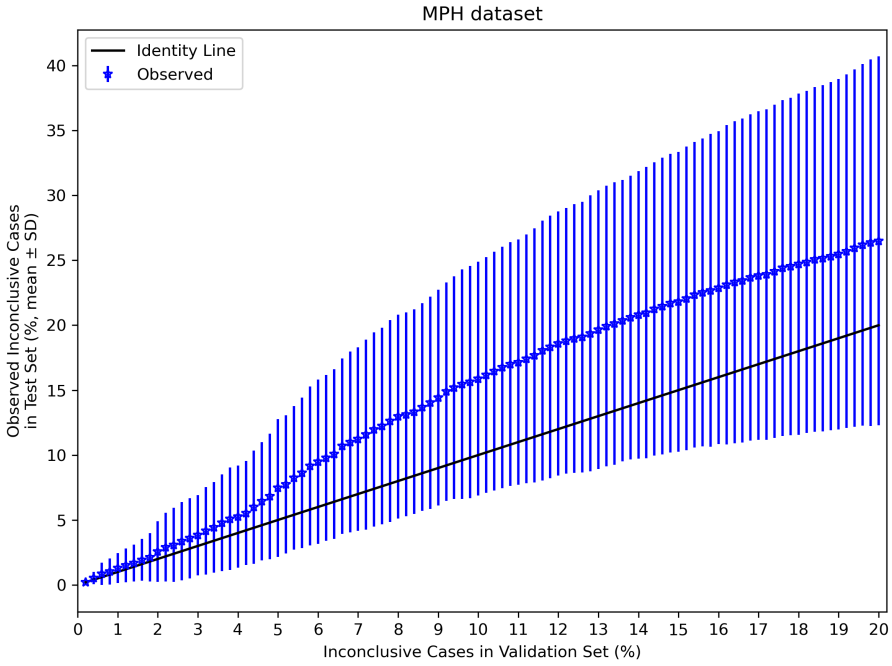


(a) Observed percentage of inconclusive cases in the PPMI dataset for a given set of percentages of inconclusive cases in the validation set (Development dataset).

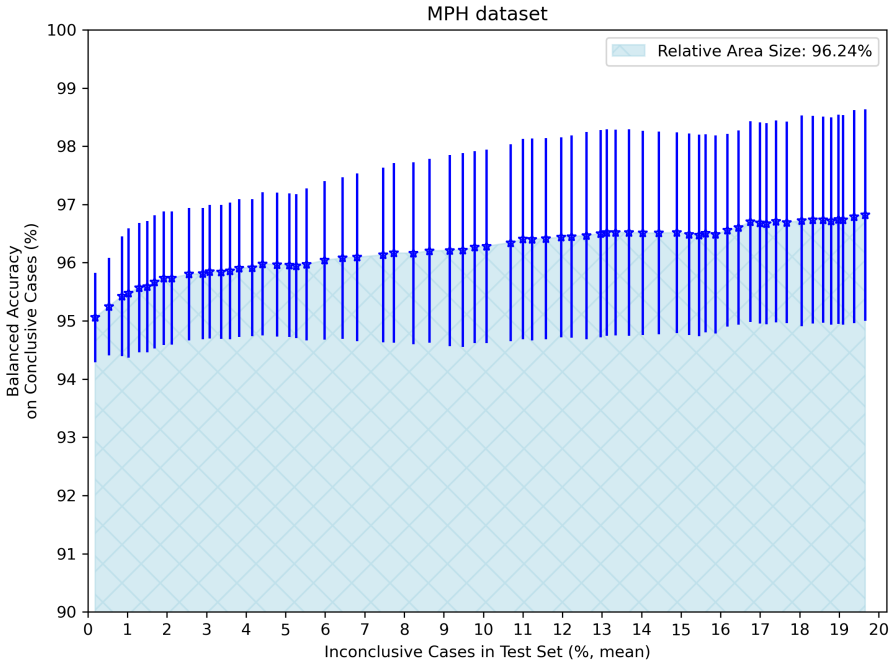


(b) Balanced accuracy on conclusive cases for a given mean percentage of inconclusive cases observed in the PPMI dataset.

Figure 27: Evaluation of the CNN-Regression method on PPMI dataset.

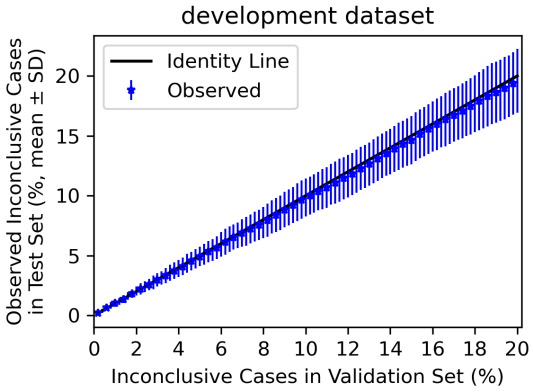


(a) Observed percentage of inconclusive cases in the MPH dataset for a given set of percentages of inconclusive cases in the validation set (Development dataset).

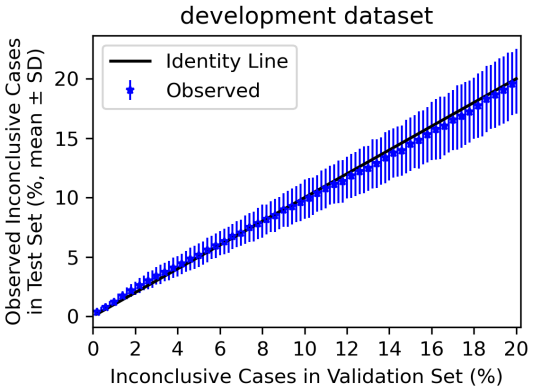


(b) Balanced accuracy on conclusive cases for a given mean percentage of inconclusive cases observed in the MPH dataset.

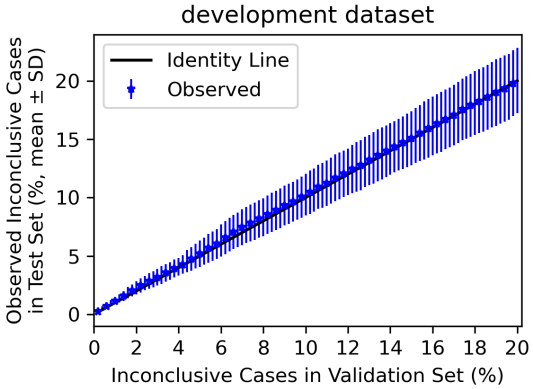
Figure 28: Evaluation of the CNN-Regression method on MPH dataset.



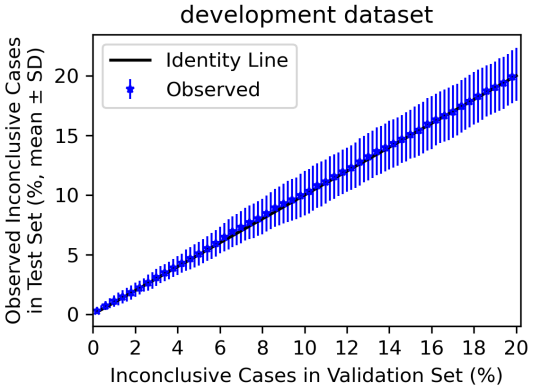
(a) SBR method.



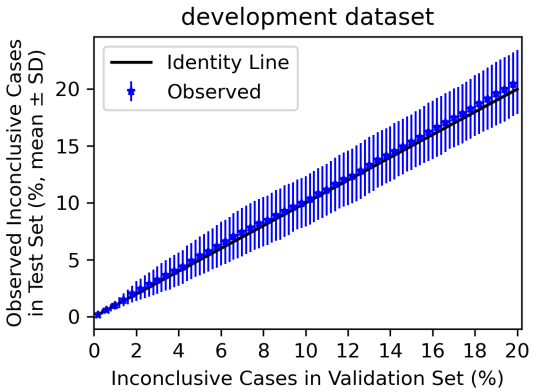
(b) PCA-RFC method.



(c) CNN method - MVT

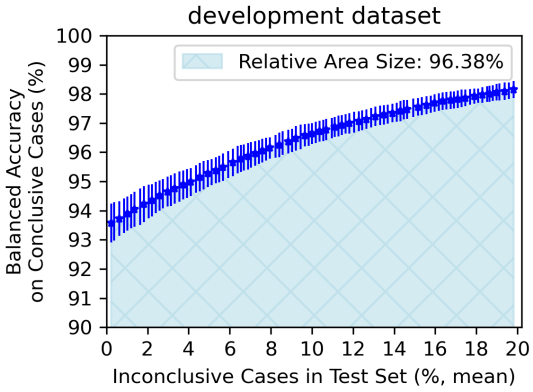


(d) CNN method - RLT

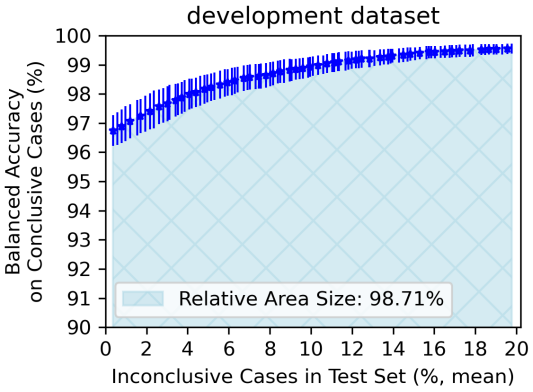


(e) CNN method - Regression

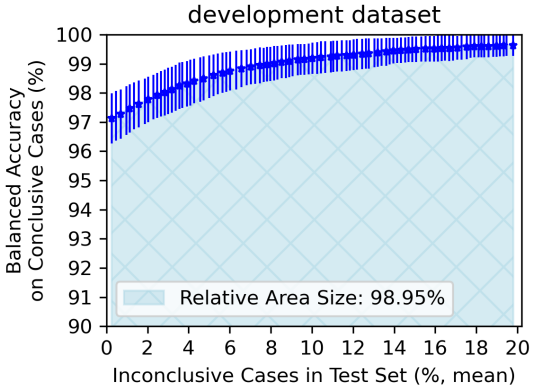
Figure 29: Comparison of different methods on test set of development data. Transferability of inconclusive intervals.



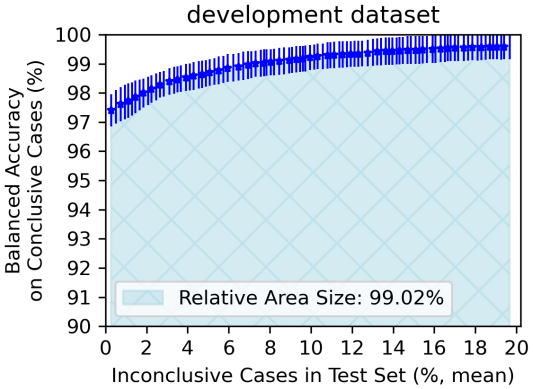
(a) SBR method.



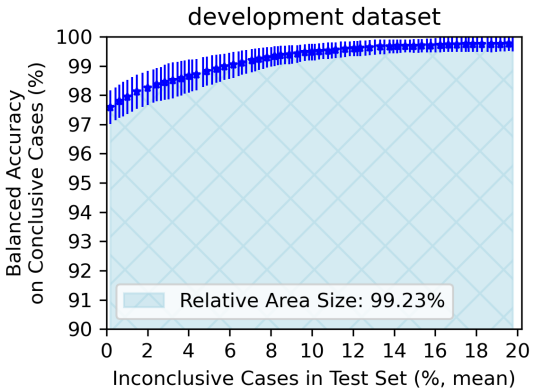
(b) PCA-RFC method.



(c) CNN method - MVT

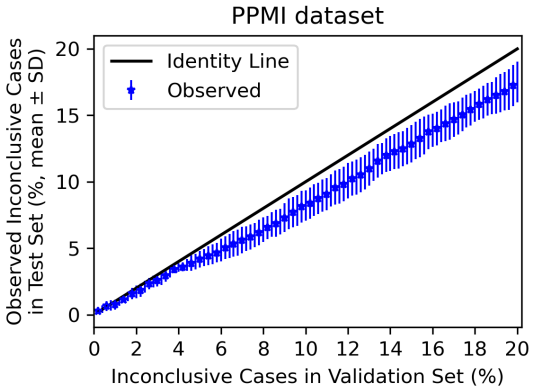


(d) CNN method - RLT

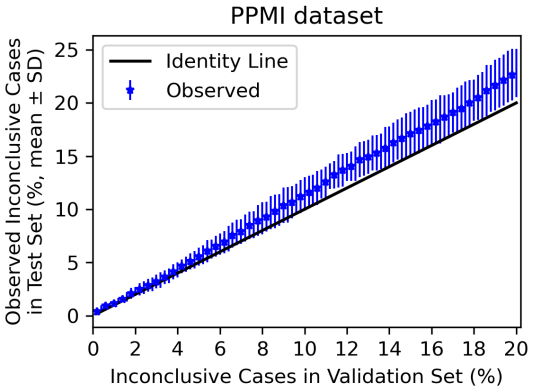


(e) CNN method - Regression

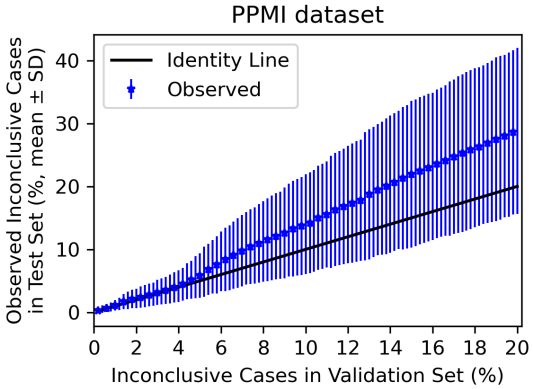
Figure 30: Comparison of different methods on test set of development data. Balanced accuracy over the percentage of observed inconclusive cases.



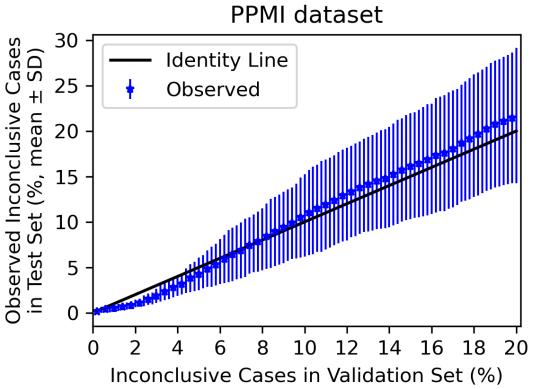
(a) SBR method.



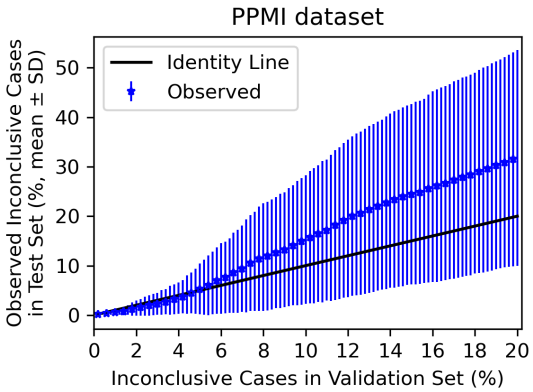
(b) PCA-RFC method.



(c) CNN method - MVT

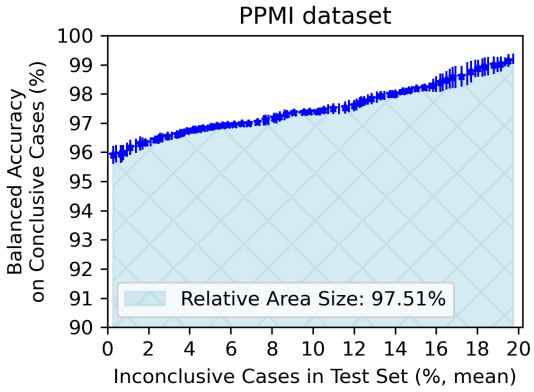


(d) CNN method - RLT

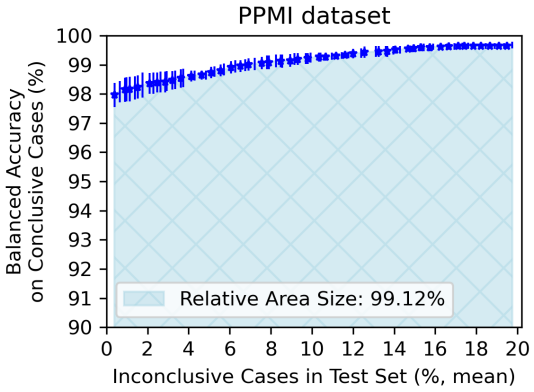


(e) CNN method - Regression

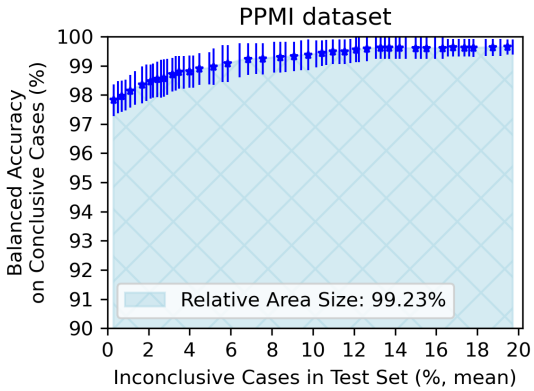
Figure 31: Comparison of different methods on PPMI dataset. Transferability of inconclusive intervals.



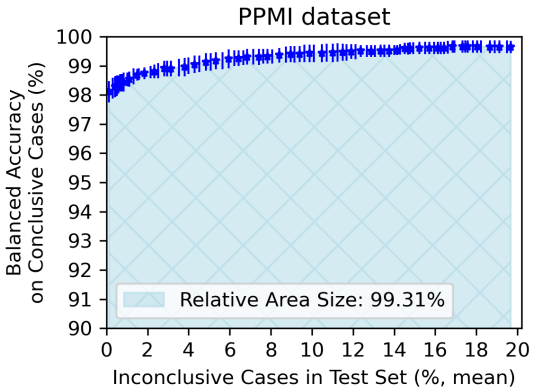
(a) SBR method.



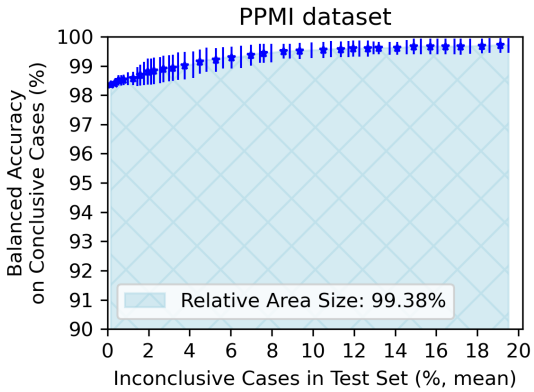
(b) PCA-RFC method.



(c) CNN method - MVT

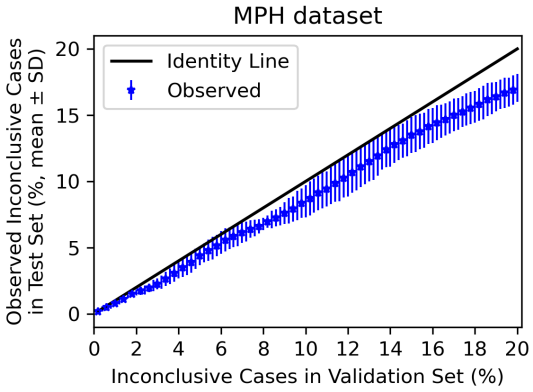


(d) CNN method - RLT

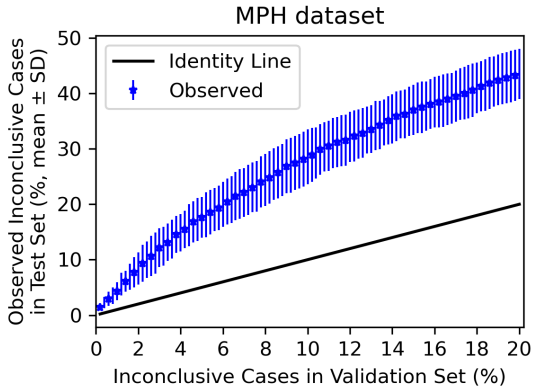


(e) CNN method - Regression

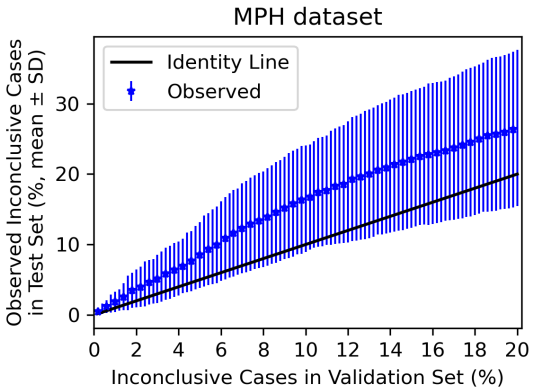
Figure 32: Comparison of different methods on PPMI dataset. Balanced accuracy over the percentage of observed inconclusive cases.



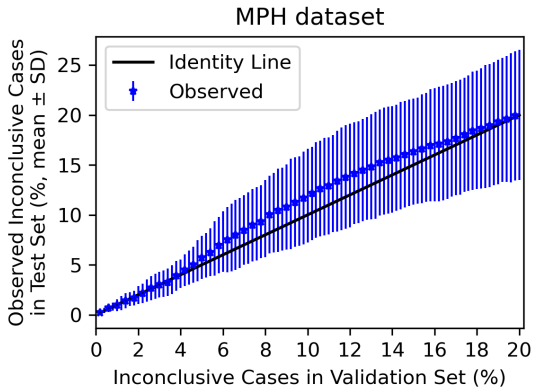
(a) SBR method.



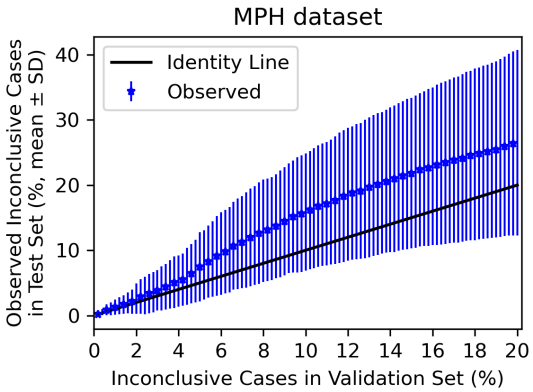
(b) PCA-RFC method.



(c) CNN method - MVT

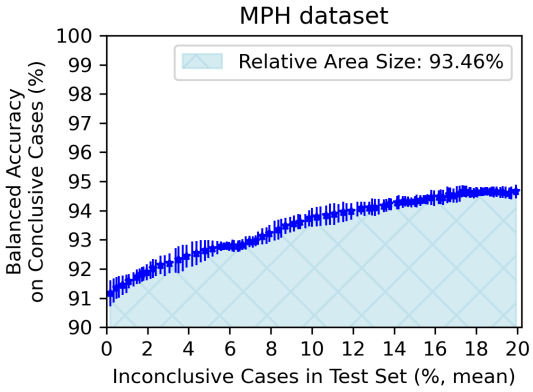


(d) CNN method - RLT

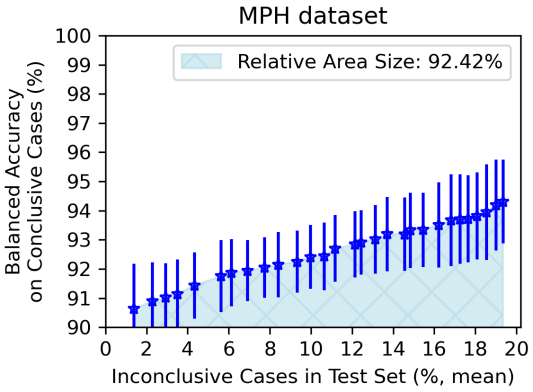


(e) CNN method - Regression

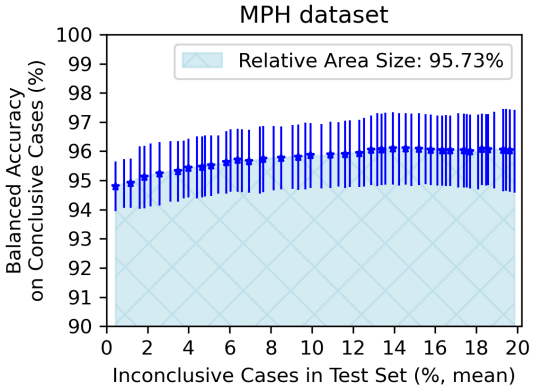
Figure 33: Comparison of different methods on MPH dataset. Transferability of inconclusive intervals.



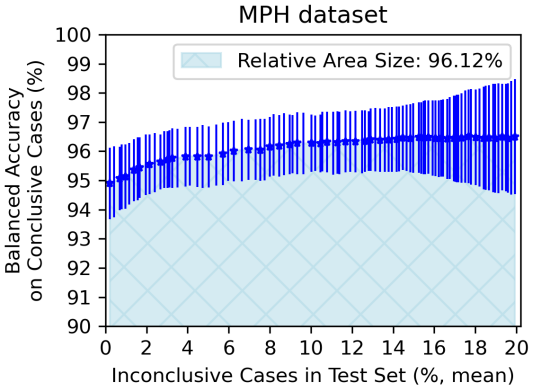
(a) SBR method.



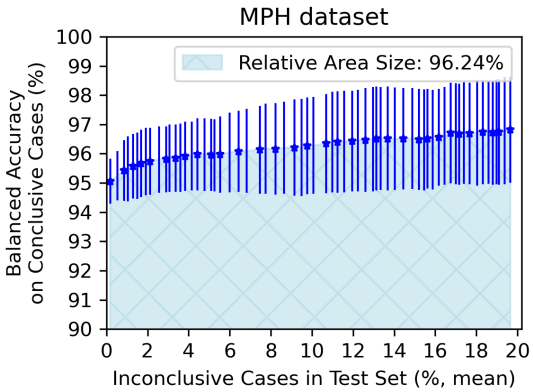
(b) PCA-RFC method.



(c) CNN method - MVT



(d) CNN method - RLT



(e) CNN method - Regression

Figure 34: Comparison of different methods on MPH dataset. Balanced accuracy over the percentage of observed inconclusive cases.

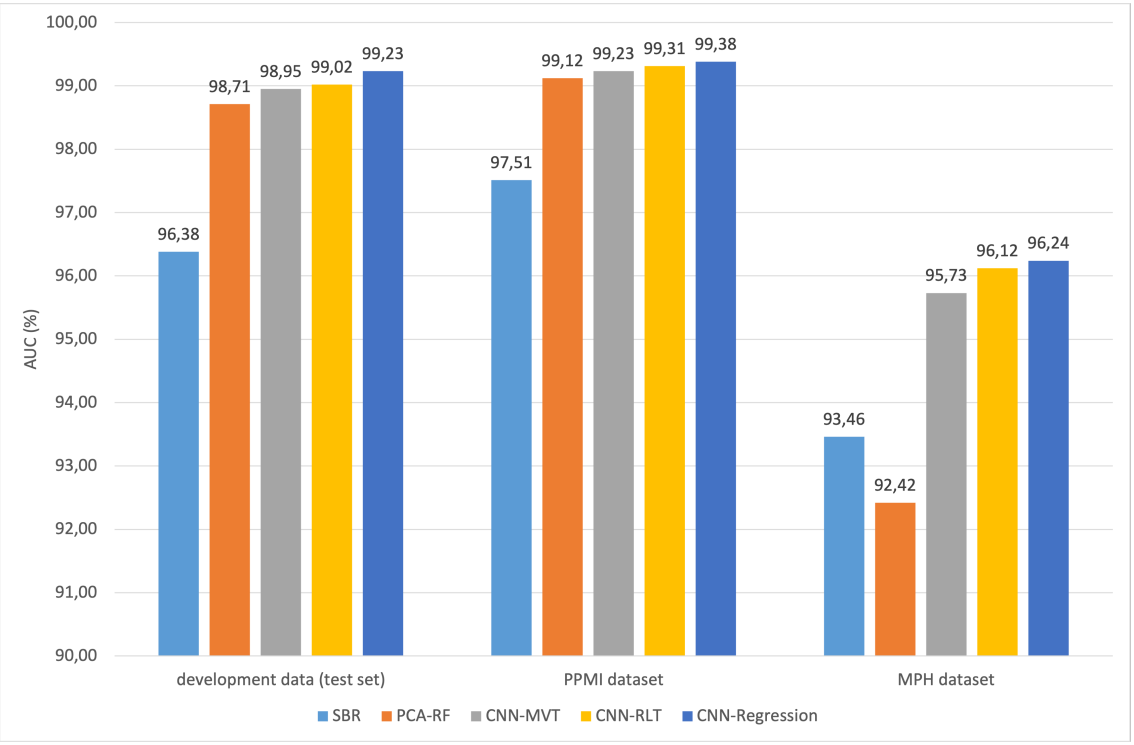


Figure 35: AUC achieved by baseline and experimental methods on different test data. The AUC was calculated for the mean balanced accuracy over the percentage of inconclusive cases in the considered test set.

A Appendix

If needed for supplementary material, such as detailed description of data collection, tables, or figures.

Bibliography

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

William J. Youden. Index for rating diagnostic tests. *Cancer*, 3(1):32–35, 1950.

Declaration of Authorship

Ich erkläre hiermit gemäß §9 Abs. 12 APO, dass ich die vorstehende Abschlussarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Des Weiteren erkläre ich, dass die digitale Fassung der gedruckten Ausfertigung der Abschlussarbeit ausnahmslos in Inhalt und Wortlaut entspricht und zur Kenntnis genommen wurde, dass diese digitale Fassung einer durch Software unterstützten, anonymisierten Prüfung auf Plagiate unterzogen werden kann.

Place, Date	Signature