WHEN HIGH-PERFORMING MODELS BEHAVE POORLY IN PRACTICE: PERIODIC SAMPLING CAN HELP

Anonymous authors
Paper under double-blind review

ABSTRACT

Training a deep neural network (DNN) for breast cancer detection from medical images suffers from the (hopefully) low prevalence of the pathology. For a sensible amount of positive cases, images must be collected from numerous places resulting in large heterogeneous datasets with different acquisition devices, populations, cancer incidences. Without precaution, this heterogeneity may result in a DNN biased by latent variables a priori independent of the pathology. This may be dramatic if this DNN is used inside a software to help radiologists to detect cancers. This work mitigates this issue by acting on how mini-batches for Stochastic Gradient Descent (SGD) algorithms are constructed. The dataset is divided into homogeneous subsets sharing some attributes (*e.g.* acquisition device, source) called Data Segments (DSs). Batches are built by sampling each DS periodically with a frequency proportional to the rarest label in the DS and by simultaneously preserving an overall balance between positive and negative labels within the batch. Periodic sampling is compared to balanced sampling (equal amount of labels within a batch, independently of DS) and to balanced sampling within DS (equal amount of labels within a batch and each DS). We show, on breast cancer prediction from mammography images of various devices and origins, that periodic sampling leads to better generalization than other sampling strategies.

1 INTRODUCTION

Breast cancer screening leads to an earlier detection of breast cancer resulting in an improved prognosis and a reduced mortality (Marmot et al., 2013). During a screening exam, Full-Field Digital Mammography (FFDM) images of the breasts of a patient are acquired. They are then analyzed by 1 or 2 radiologist(s) who look(s) for signs of malignancy. If any sign of malignancy is observed, the patient is recalled to assert the final diagnosis with complementary exams. Despite constant technological progress, cancers may still be missed while the recall rate remains high with a limited fraction of recalled patients actually diagnosed with cancer (Rawashdeh et al., 2013).

Machine Learning (ML) based screening systems may help radiologists detecting cancers and improve their sensitivity, specificity and reading time, (Pacilè et al., 2020; McKinney et al., 2020; Schaffter et al., 2020; Shen et al., 2019; Kooi et al., 2017). Indeed, Deep Neural Networks (DNNs) can help to solve this problem initially formalized as a binary classification problem. Recent approaches build upon breasts symmetry (Brhane Hagos et al., 2018; Kooi & Karssemeijer, 2017), the patient history (Kooi & Karssemeijer, 2017) or the whole set of exam images (Geras et al., 2017) to enhance classification performances. Other approaches further aim at localizing the malignant lesion(s) in the breast (Shen et al., 2021; Pedemonte et al., 2020) or improving the generalization of DNNs trained on FFDM images but used on Digital Breast Tomosynthesis (DBT) images, $3D$ images of the breast (Matthews et al., 2021; Singh et al., 2020).

It is hard to train a DNN on FFDM images that aims to be used in a software delivering nearly real time recommendations for a radiologist investigating screening exams. First, it requires a *large* quantity of labeled training data but medical data sources are typically small and partially labeled. Transfer Learning (TL) may help to partially cope with this problem (Morid et al., 2021; Karimi et al., 2021) while Domain Adaptation (DA) may help to reduce the potential distribution mismatch between multiple source data distributions and a test data distributions (Crammer et al., 2008; Ben-David et al., 2010; Ganin et al., 2016; Guan & Liu, 2021). Yet, TL may deliver little performance gain (Raghu et al., 2019), DA may not be appropriate for the setup considered here due to the potentially large number of training sources and the infinite number of testing sources and the ”nearly real time” prediction constraint may be incompatible with the common offline setup most DA solutions embrace as they often build upon source and target domains available at the same time (Ben-David et al., 2007; 2010; Ganin et al., 2016). Second, each medical institution exhibits its own characteristics (patients, radiologists experiences, sets of acquisition devices, disease incidences, ...), thus the DNN trained on data from such sources may still be biased: its predictions being modulated by latent uncontroled variables. If not addressed, this may result in dramatic consequences once the model is used in production (Barocas & Selbst, 2014). Finally, training a DNN on FFDM implies fighting class imbalance (the disease incidence being hopefully small) to avoid technical issues for ML models (He & Garcia, 2009) that could amplify the potential bias issue highlighted above.

The following experiment exhibits the complexity of training a DNN from only 2 sources of data. The goal was to train 2 DNNs to classify benign *vs* malignant FFDM images on 2 sets of data, DS_1 (7258 malignant / 13341 benign images) and DS_2 (160 malignant / 21871 benign) and to evaluate them on a left out set of data DS_3 (1106 malignant / 14192 benign) from another medical institution. A train / validation split was used patient-wise and source-wise: 80% (resp. 20%) of patients of a source were used for training (resp. validation). The proportion of patients with cancers was kept constant in both train / val sets. The first DNN, dnn_1, was trained on DS_1, the second one, $dnn_{1\cup 2}$ on $DS_1 \cup DS_2$ with a Stochastic Gradient Descent (SGD) like algorithm. Balancing benign and malignant images within mini-batches was used during training to cope with class imbalance. The Weighted Logloss per image (WLL / image), log loss with class weights, was monitored on a validation set during the training, see Figure 1 - left. The model $dnn_{1\cup 2}$ exhibits a low global validation WLL / image on $DS_1 \cup DS_2$, but a higher WLL / image on DS_1 compared to dnn_1. The validation WLL / image on DS_2 further indicates that it was over-fitted. Plus, $dnn_{1\cup 2}$ shows a lower Area Under the ROC Curve (AUC) on DS_3 than dnn_1, though it was trained on more data, see Figure 1 - center. Finally, the distributions of models predictions on DS_3 show that $dnn_{1\cup 2}$ predictions are shifted towards low values compared to dnn_1 predictions and that even for the malignant images, see Figure 1 - right. Hardly, does $dnn_{1\cup 2}$ benefit from data brought by DS_2.

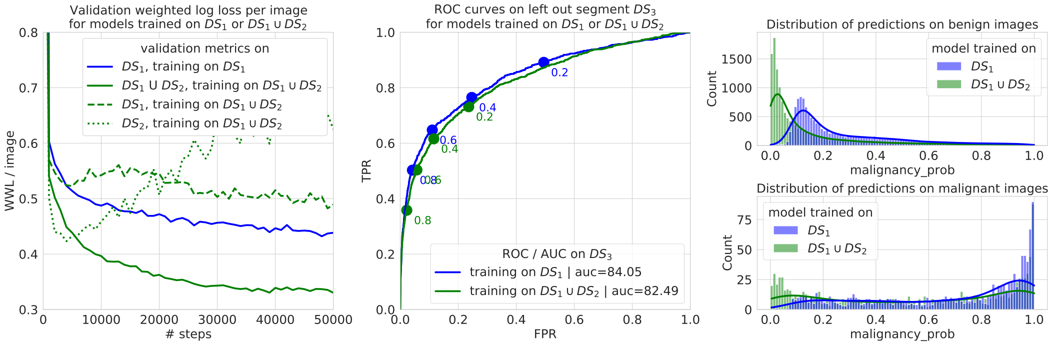

Figure 1: The model trained on $DS_1 \cup DS_2$ (green) exhibits poor validation metrics on the validation sets of DS_1 and DS_2 taken individually. It does not outperform the baseline trained on DS_1 (blue) on test set DS_3. Its predictions are shifted towards low values on benign and even malignant images.

This paper identifies 2 main locks to train efficiently a DNN for breast cancer prediction from mammography images: class imbalance and a non homogeneous learning set. Class imbalance can be addressed via an appropriate sampling strategy (He et al., 2008; Chawla et al., 2002), sample weighting (Byrd & Lipton, 2018) or a learnable sampling (Hu et al., 2019; Ren et al., 2018). The sampling strategy may add further benefits such as stratified sampling which may help to efficiently select samples from low variance clusters to speed up the convergence of mini-batch optimization algorithms (Csiba & Richtárik, 2018; Zhao & Zhang, 2014).

This paper focuses on training a DNN with a mini-batch SGD like optimization algorithm on a training set, divided into a finite ensemble of independent and homogeneous sets of samples, sharing an explicit and unique set of properties, called Data Segments (DSs). A *Periodic Sampling* strategy is proposed to build mini-batches of data by sampling each DS periodically with a frequency proportional to its rarest label and by balancing the labels within a batch and a DS to fight class imbalance. This paper, first introduces formally the proposed *Periodic Sampling*, then provides numerical evidence of its benefits and finally discusses its main limitations and directions for improvement.

2 METHOD

This paper focuses on binary classification and introduces a method to train a DNN, with a SGD like algorithm, from several segments of data while reducing the risk of using an identified bias. It aims to build appropriately the mini-batches of data with a dedicated *Periodic Sampling* strategy.

2.1 NOTATION AND PROBLEM STATEMENT

Let $\mathcal{X}$ stand for the input space, *i.e.* the space of FFDM images of a given shape. Let $\mathcal{Y} = \{0, 1\}$ stand for the space of labels, 0 standing for a benign image, 1 for a malignant image. $x, y \in \mathcal{X} \times \mathcal{Y}$ is a learning sample: an image with its label. Let $i, N_i \in \mathbb{N}$, then $DS_i = \{(x_j, y_j) \in \mathcal{X} \times \mathcal{Y} : j = 0, \cdots, N_i - 1\}$ refers to a Data Segment (DS) *i.e.* a subset of homogeneous learning samples. Let $M \in \mathbb{N}$ be the number of DSs at hand. The learning set is composed of M homogeneous and independent DSs: $\cup_{i=0}^{M-1} DS_i$. With a DNN: $f : \mathcal{X} \longrightarrow \mathcal{Y}$, and with ℓ standing for the binary cross-entropy, the empirical optimization problem to solve reads:

$$f^* \in \arg\min_f \frac{1}{\sum_{i=0}^{M-1} N_i} \sum_{i=0}^{M-1} \sum_{(x,y) \in DS_i} \ell(y, f(x)) \tag{1}$$

2.2 PERIODIC SAMPLING

General description First, the DNN should work on all the DSs of interest. It should learn at the same pace on the training DSs to avoid the issue seen in Figure 1 and not overfit to quickly a DS with few malignant images. Third, the DNN should not be biased toward a specific label on any DS.

Periodic Sampling is a two step process introduced to meet these requirements. First, it builds a *sampling pattern* that indicates how often samples from the DSs should be included in a mini-batch. Second, it iterates over the *sampling pattern* and selects a sample from the designated DS while alternating in a balanced way the labels to ensure label balance in a DS.

Sampling proportion With the above notation, we define the number of samples of the least represented label y_i^{min}, inside DS_i by $n_i^{min} = \sum_{j=0}^{N_i-1} \mathbb{1}_{y_j = y_i^{min}}$. Now, let $N^{min} = \sum_{i=0}^{M-1} n_i^{min}$ be the sum of the samples of the least represented label inside each DS. The main idea is to feed the DNN with the samples from DS_i with a proportion $p_i = n_i^{min}/N^{min}$ during the training process.

Sampling pattern Periodically selecting samples from DS_i with a period $1/p_i$ allows to feed the DNN with samples from DS_i with a proportion p_i. This way, even if p_i is dramatically small, samples from DS_i would still be selected on a regular basis. To do so, a sampling pattern S is built. This is a sequence of DS ids $i \in \{0, \cdots, M-1\}$ of length N^{min}, where the number of occurrences of DS_i is given by n_i^{min}. To build it, one first sorts the DS by their numbers n_i^{min}, in the ascending order. Then, one iterates over the sequence of i, n_i^{min}, to fill in the available slots of a list of length N^{min} and places the indices i at n_i^{min} locations, equally spaced while avoiding potential collisions with already filled slot. This is described in Algorithm 1.

Construction of training batches Then, one builds batches of $b \in \mathbb{N}$ samples by iterating sequentially (and infinitely) over the sampling pattern S to get, for each sample to be included in a batch, the index i of the DS from which it should be drawn. This way, for every nearly $N^{min}/b + 1$ seen samples by the DNN, the DNN has been trained on nearly $p_i \times (N^{min}/b + 1)$ samples from DS_i. To ensure that the labels are presented in equal proportions to the DNN, the mechanism alternates between 0 and 1 labels when selecting a sample from DS i. The batch construction pipeline is summarized in Algorithm 2. An example of the selection of samples according to a sampling pattern with the alternating label mechanism is provided in Figure 2

Algorithm 1: Build the sampling pattern

```
Data: $N$, $\{DS_i : i = 0, \cdots, M-1\}$
Result: Sampling pattern $S$
step 1: Compute the numbers of samples with the least represented label for every $DS_i$:
  $\{n_i^{min} : i = 0, \cdots, M-1\}$
step 2: Compute the length of the sampling pattern $L \leftarrow \sum_{i=0}^{M-1} n_i^{min}$
step 3: Build the sampling pattern:
$S \leftarrow$ array of size $L$ filled with -1 values
for $i \leftarrow 0$ to $M-1$ do
    $pos \leftarrow linspace(0, N^{min}, n_i^{min})$ ;
    for $j \leftarrow 0$ to $n_i^{min} - 1$ do
        $p \leftarrow pos[j]$
        while $S[p] \neq -1$ do
            $p \leftarrow p + 1$
        end
        $S[p] \leftarrow i$
    end
end
```

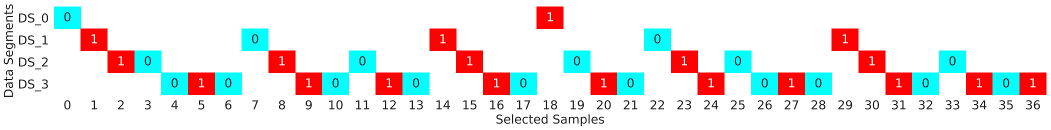

Figure 2: Selection of samples from 4 DSs, DS_0, DS_1, DS_2, DS_3, while alternating the labels $0, 1$ within a DS. A sampling pattern of length $N^{min} = 37$ was used to select the samples. It was built from the numbers of samples having the rarest labels (1 in this case) in DS_i: n_i^{min} for $i = 0, \cdots 3$. The following values were used: $n_0^{min} = 2$, $n_1^{min} = 5$, $n_2^{min} = 10$ and $n_3^{min} = 20$.

Algorithm 2: Build batches from sampling pattern

```
Data: $b$, $S$, $N^{min}$, $DS = \{DS_i : i = 0, \cdots, M-1\}$
Result: A collection of batches of size $b$
$Y \leftarrow \{0, 1\}^M$ randomly initialized
$idx \leftarrow 0$
while training do
    $B \leftarrow [\cdots]$ empty list
    for $j \leftarrow 0$ to $b-1$ do
        $i \leftarrow S[idx]$
        $y_i \leftarrow Y[i]$
        $(x, y) \leftarrow$ next samples $(x, y) \in DS_i$ such that $y = y_i$
        $B \leftarrow B + (x, y)$
        $idx = idx + 1 \pmod{N^{min}}$
        $Y[i] = Y[i] + 1 \pmod 2$
    end
    yield B
end
```

Practically, some Deep Learning (DL) libraries ensure that the order of the samples indices returned by a sampling function is maintained inside the mini-batches while others do not. This may not be an issue as long as the proportions of DSs are ensured at a scale of a few (hundreds) learning steps / batches.

3 EXPERIMENTS

3.1 EXPERIMENTAL SETUP

3.1.1 DATA

Image modality and labeling process The considered data are FFDM images *e.g.* gray-scale images, with pixel values in $[0, 4095]$, although other ranges can be used depending on the FFDM device manufacturer. Every full view (not zoom) FFDM image is considered for training and evaluating the models: Cranio-Caudal (CC), Medio-Lateral Oblique (MLO), ...

Images labels (0 and 1) are assigned as follows. A positive image (label 1) is an image where a suspicious lesion was localized and for which the malignant status was confirmed by a biopsy shortly after the mammogram was done (within 3 months). Each malignant lesion is localized as precisely as possible with a bounding box drawn by trained radiologists. A positive breast is a breast with at least one positive image. When the malignant lesion is not visible inside a specific view of a malignant breast the image is discarded from the dataset *e.g.* a lesion not visible in CC view while visible in MLO view. Images anterior and posterior to a positive breast are discarded as well. On the other hand, a negative image (label 0) is an image of a negative patient *i.e.* a patient without a positive breast. In addition, a one-year negative follow-up is required to confirm absence of signs of malignancy in the image. Additional data information are provided in Appendix.

Data segments The learning set is composed of 2 sources of data, S_1 and S_2. Each source contains images obtained with acquisition devices of 2 manufacturers M_a and M_b. Thus, 4 DSs can naturally be defined: $(S_1, M_a), (S_1, M_b), (S_2, M_a), (S_2, M_b)$. The testing sets are composed of up to 3 sources of data, S_3, S_4 and S_5, containing images obtained with acquisition devices from 4 manufacturers, M_a, M_b, M_c and M_d, leading to 5 DSs. The numbers of malignant and benign images from all train and test DSs are given in table 1.

Table 1: Numbers of benign (ben.) and malignant (mal.) images by DS and incidence rate in DSs

DS	# ben. images	# mal. images	p_{mal} in DS
(S_1, M_a)	13341	7258	0.3523
(S_1, M_b)	527	408	0.4364
(S_2, M_a)	21871	160	0.0073
(S_2, M_b)	19040	146	0.0076
(S_3, M_a)	14192	1106	0.0723
(S_3, M_b)	46006	2059	0.0428
(S_4, M_c)	4387	28	0.0063
(S_4, M_d)	19740	133	0.0067
(S_5, M_c)	38951	131	0.0034

Cross-validation scheme For each experiment, the learning set is divided into training and validation data using a source-wise and patient-wise split: 80% (resp. 20%) of the patients of each source are used for training (resp. validation). Furthermore, the split is made such that the ratio of malignant / benign patients is similar for the validation and the training sets.

Image pre-processing and data augmentation Each image is extracted from its native DICOM storing file (Pianykh, 2010) and resized to a shape of $(1152, w)$, the width w being adjusted accordingly to keep the same image proportions. The mask of the breast is extracted from the reshaped image using automatic thresholding and morphological operations.

Normalization and preprocessing techniques are used to standardize the FFDM images. First, the images are cropped / padded to fit a shape of $1152 * 832$. Then, the pixels inside the breast mask are re-normalized to have values in $[0, 1]$. The pixels outside mask are set to a constant value such that the average pixel value of the image is always the same. A standard augmentation pipeline (random rotation, zoom, shearing...) is used to augment images used for training while ensuring the malignant lesion remains visible in the augmented image.

3.1.2 Model training and evaluation

Metrics As mammography data usually comes with strong class imbalance, the evaluation metrics need to be independent of labels proportions. The Weighted Logloss per image (WLL / image), the standard logloss with samples weights estimated on the evaluation (here validation) set, the Receiver Operating Characteristic curve (ROC curve) and the AUC are considered. Metrics are computed *globally i.e.* on the whole (validation or test) set, and also *locally i.e.* on the DSs taken individually. To make sure a model exhibits a similar behavior on every DS, one also considers the Operating Points (OPs) at threshold in $\{0.2, 0.4, 0.6, 0.8\}$ on the ROC curve of this DS. The AUC of a DNN and the difference in AUC between 2 DNNs, Δ AUC, are evaluated measured using 2000 Efron's bootstraps with uniform sampling with replacement over the test set (or test data-segment). Then 95% Confidence Interval (CI), given in brackets, and P-value are provided as well for the ΔAUC, (Altman & Bland, 2011). A negative CI upper bound of ΔAUC, with p-value < 0.05 is regarded as a significant decrease in performance. A non negative CI lower bound of ΔAUC, with p-value < 0.05 is regarded as a significant increase in performance.

Neural network The considered DNN processes every image independently, it makes image wise predictions and assigns for each image a probability of malignancy. It follows a VGG-like architecture pattern (Schaffter et al., 2020) and is composed of 2 parts: a backbone and a classification head, the backbone being first pre-trained to classify lesions annotated on a subset of images from (S_1, M_a), see Appendix for more details. The full model is fine-tuned to perform binary classification on 2 Tesla P100, for 150000 steps with Adam optimizer (Kingma & Ba, 2014) and a learning rate equal to 10^{-4}. Exponential moving average with decay 0.99 of all trainable parameters are tracked, their averaged versions are then used for inference on validation and test data. A batch size of 24 samples is used (12 per GPU). A validation step is made every 1000 training steps. The checkpoint achieving the best WLL / image on the validation set is kept for inference. Tensorflow 2.3 (Abadi et al., 2015) was used for the experiments.

3.1.3 Baselines and compared methods

Baselines Two baselines are considered and are obtained by training the model on a single DS: (S_1, M_a) for baseline (1), (S_1, M_b) for baseline (2). Training is done with standard balanced sampling to cope with class imbalance. Baseline 1 (resp. 2) is used as a reference in experiments where models are evaluated on (S_3, M_a), (S_4, M_c), (S_4, M_d), (S_5, M_c) (resp. (S_3, M_b)) to compute the differences of AUC with the other methods (*c.f.* Δ AUC).

Compared methods Three strategies are benchmarked. They all aim at training the considered DNN on 2 or more DSs. The first one is called ”balance labels only” and is referred to by ”Bal. labels”. It aims at sampling equally the labels $0, 1$ independently of the DSs. This boils down to consider that all the training samples come from a unique DS as often assumed in ML. The second one is called ”balance every data-segments equally” and is referred to by ”Bal. segments”. It aims at sampling equally the DSs and the labels $0, 1$. In other words, it can be seen as a sampling pattern S built as if the proportion p_i of every DS DS_i was given by $1/M$, with M the number of DSs. This way, negative and positive samples from DS_i are sampled as often as negative and positive samples from DS_j, with $i \neq j$. The last one is the proposed *Period Sampling*, referred to by ”P. sampling”.

3.2 Experiment 1: training on 2 Data-Segments (2 sources, 1 manufacturer)

In this experiment, the training / validation set is divided into 2 DSs: (S_1, M_a) and (S_2, M_a): images have been obtained with acquisition devices of the same manufacturer but come from 2 different institutions. Performances on the test set (S_3, M_a) are reported in Table 2.

The DNNs trained with ”Bal. labels” and ”Bal. segments” do not beat the baseline on DS (S_3, M_a) whereas the DNN trained with ”P. sampling” brings a small gain of performance. Such a small gain may be due to the limited amount of malignant images of the DS (S_2, M_a) compared to the large number of malignant images in the DS (S_1, M_a).

Table 2: Performances of DNNs on (S_3, M_a).

AUC baseline	Method	AUC	Δ AUC	p-value
84.05 (82.64, 85.41)	Bal. labels	82.50 (81.04, 83.95)	-1.55 (-2.61, -0.45)	0.0048
84.05 (82.64, 85.41)	Bal. segments	82.87 (81.43, 84.27)	-1.19 (-2.26, -0.06)	0.0340
84.05 (82.64, 85.41)	P. sampling	**84.72** (83.38, 86.04)	**0.66** (-0.24, 1.57)	0.1500

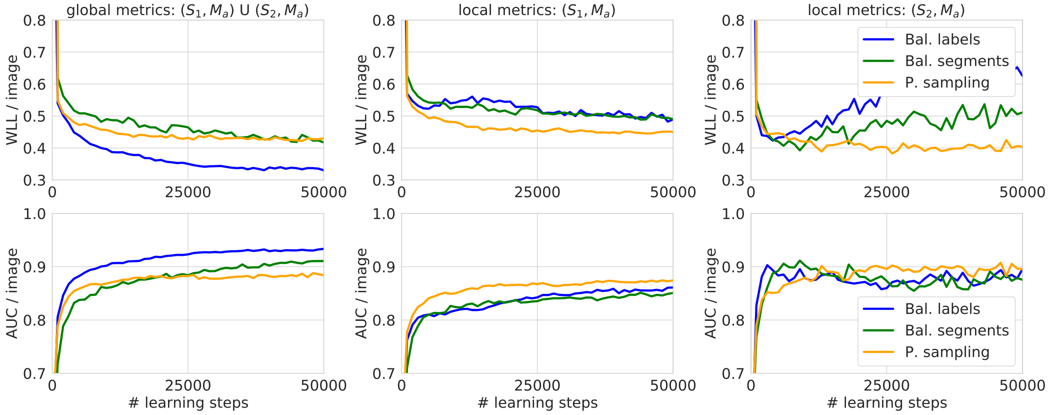

Figure 3: Validation metrics of DNNs trained on 2 DSs (S_1, M_a) and (S_2, M_a). First column: *global* metrics. Second column: *local* metrics on DS (S_1, M_a). Third column: *local* metrics on DS (S_2, M_a). First row: WLL / image. Second row: AUC per image.

The *global* metrics and the *local* metrics on (S_1, M_a) and (S_2, M_a) exhibit a significantly large discrepancy for the DNN trained with ”Bal. labels”, see Figure 3 - blue curves. Indeed, this DNN presents a very low *global* WLL / image and a very (suspiciously) high *global* AUC / image while it also has much higher WLL / images and much lower AUCs on the DSs (S_1, M_a) and (S_2, M_a). It is as if the DNN trained with ”Bal labels” learned a wrong task: classify the malignant images from (S_1, M_a) *vs* the benign images of (S_2, M_a). Moreover, the WLL / image (resp. the AUC) on (S_2, M_a) indicates that the DNN has over-fitted very quickly this DS and does not learn from it after 5000 learning steps. Everything appears as if this second DS (S_2, M_a) is never used in the end. The DNN trained with ”Bal. segments” seems to over-fitting more slowly the DS (S_2, M_a) as suggested by the *local* metrics on this DS, Figure 3 - green curves. On the contrary, the DNN trained with ”P. sampling” demonstrates much more coherent and reliable *global* and *local* metrics, Figure 3 - yellow curves. The *global* metrics look like an average of *local* metrics both for the WLL / image and the AUC / image. Plus, the DNN does not seem to quickly over-fit the segment (S_2, M_a): it seems to learn at the same pace on both DSs. Last but not least, as a (small) gain of AUC is observed on the test set, this implies that, the DNN trained with ”P. sampling” is able to learn from both DSs and to leverage the second DS (S_2, M_a) to improve the test performances.

The ROC curves obtained on the validation sets of DSs (S_1, M_a) and (S_2, M_a) are represented for each model as well as the OPs at thresholds $0.2, 0.4, 0.6, 0.8$, see Fig 4. On the whole validation set, the DNNs trained with ”Bal. labels” and ”Bal. segments” seem *globally* better than the DNN trained with ”P. sampling”. Indeed, they reach higher *global* AUC, see black curves in Fig 4. Yet, on the DSs taken individually the DNN trained with ”P. sampling” performs better than the others and reduces the discrepancy between OPs of the ROC curves of both DSs. This likely indicates that the DNN trained with ”P. sampling” works similarly on both DSs. On the contrary, the DNN trained with ”Bal. labels” leads to un-synchronized OPs between the ROC curves of the 2 DSs. Plus the OPs of (S_1, M_a) look like opposite to the ones of (S_2, M_a) suggesting that this DNN may actually classify the malignant images from (S_1, M_a) *vs* the benign images from (S_2, M_a).

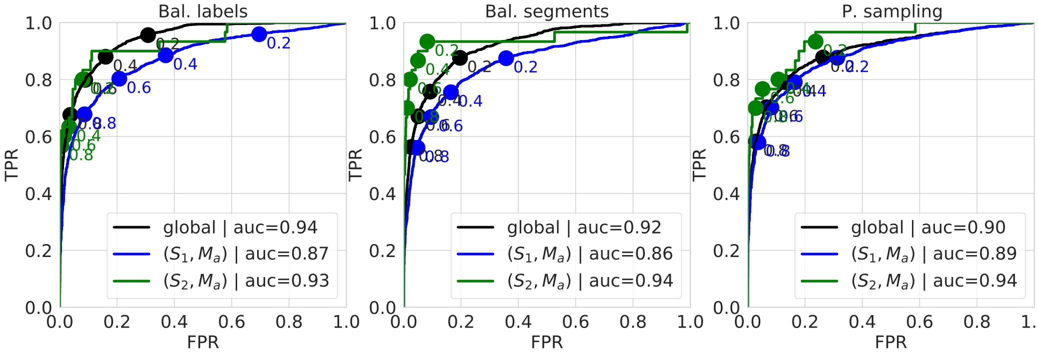

Figure 4: *Global* and *local* ROC curves on the validation sets DSs (S_1, M_a) and (S_2, M_a) for DNNs trained with: ”Bal. labels”, ”Bal. Segments” and ”P. sampling” (left to right). In black the *global* ROC curve is presented. In blue (resp. green) the ROC curves obtained on the validation sets of DSs (S_1, M_a) (resp. (S_2, M_a)). OPs at threshold $0.2, 0.4, 0.6, 0.8$ are added on each ROC curve.

3.3 Experiment 2: training on 4 Data-Segments

In this experiment, the training / validation set is composed of 4 DSs: (S_1, M_a), (S_1, M_b), (S_2, M_a) and (S_2, M_b). 2 baselines are considered: a DNN trained on (S_1, M_a) (resp. (S_1, M_b)) which is used when evaluating / comparing the methods on (S_3, M_a) (resp. (S_3, M_b)). The models performances on the test sets (S_3, M_a) and (S_3, M_b) are reported in Table 3.

Table 3: Performances on the data-segment (S_3, M_a) and (S_3, M_b).

AUC baseline	Method	AUC	Δ AUC	p-value
		(S_3, M_a)		
84.05 (82.64, 85.41)	Bal. labels	82.88 (81.39, 84.28)	-1.18 (-2.21, -0.11)	0.0280
84.05 (82.64, 85.41)	Bal. segments	82.17 (80.72, 83.68)	-1.89 (-3.02, -0.70)	0.0015
84.05 (82.64, 85.41)	P. sampling	**84.98** (83.58, 86.31)	**0.92** (0.15, 1.69)	0.0190
		(S_3, M_b)		
80.20 (79.03, 81.35)	Bal. labels	88.59 (87.75, 89.40)	8.39 (7.33, 9.54)	< 0.0001
80.20 (79.03, 81.35)	Bal. segments	86.89 (85.99, 87.73)	6.69 (5.61, 7.75)	< 0.0001
80.20 (79.03, 81.35)	P. sampling	**90.03** (89.28, 90.80)	**9.84** (8.79, 10.88)	< 0.0001

The DNNs trained with ”Bal. labels” and ”Bal. segments” do not beat the baseline on (S_3, M_a) and exhibit a negative Δ AUC which is statistically significant. On the other hand, the DNN trained with ”P. sampling” brings a small gain of performance which is is statistically significant. On (S_3, M_b), all DNNs beat the baseline, but the one trained with ”P. sampling” gives a larger gain of AUC. Synchronization of OPs on the ROC curves is again impacted, *c.f.* Figure 6 in Appendix.

Last but not least, the DNN trained with the proposed ”P. sampling” seems to better generalize on new manufacturers than the DNNs trained with other compared approaches and seems slightly better than the baseline trained on DS (S_1, M_a) although the gain is not significant, see Table 4.

4 Discussion

Periodic sampling exhibits some drawbacks that are now discussed.

The main drawback of the proposed method is related to the segmentation of the learning set into DSs as we have little information on how building DSs. Two questions arise: (1) ”when does one have to segment the learning set ?” and (2) ”how ?”.

Table 4: Benchmark on 2 unknown manufacturers from the 3 DSs: (S_4, M_c), (S_4, M_d), (S_5, M_c).

AUC Baseline	Method	AUC	Δ AUC	p-value
(S_4, M_c)				
$85.20\ (75.16, 92.90)$	Bal. labels	83.02 (73.13, 91.38)	-2.18 (-9.37, 6.24)	0.6000
$85.20\ (75.16, 92.90)$	Bal. segments	83.76 (74.19, 92.30)	-1.44 (-10.58, 7.37)	0.7700
$85.20\ (75.16, 92.90)$	P. sampling	**88.26** (78.80, 95.25)	**3.06** (-3.14, 10.94)	0.4000
(S_4, M_d)				
$89.01\ (85.24, 92.29)$	Bal. labels	87.67 (83.74, 91.31)	-1.34 (-4.99, 2.40)	0.4900
$89.01\ (85.24, 92.29)$	Bal. segments	85.46 (81.39, 89.26)	-3.55 (-6.79, -0.46)	0.0280
$89.01\ (85.24, 92.29)$	P. sampling	88.57 (84.79, 91.88)	-0.45 (-2.19, 1.64)	0.6600
(S_5, M_c)				
$85.97\ (82.14, 89.43)$	Bal. labels	82.70 (77.94, 87.20)	-3.27 (-7.43, 0.62)	0.1100
$85.97\ (82.14, 89.43)$	Bal. segments	85.94 (82.16, 89.42)	-0.03 (-4.05, 3.80)	0.9900
$85.97\ (82.14, 89.43)$	P. sampling	**87.93** (84.02, 91.66)	**1.96** (-1.20, 5.11)	0.2300

By experimenting, we came up with 3 empirical ways to know whether a set of data should be divided into DSs. First, a poor generalization performance on a left out source of data is likely to indicate that the learning set should be split into DSs as in experiments 1 and 2. Second, a very surprisingly nice validation metrics is also a good indicator that the learning set should be divided into DSs, see Figure 3. Finally, good knowledge of the learning set and a careful look at positive and negative samples numbers in natural subsets of data may indicate a required division of the learning set into DSs, see Table 1. An unbalance between negative samples from a subset and positive samples from another one is likely to indicate a needed segmentation.

Through an iterative process, one can define the DSs relevant to the the learning set and the task to perform: (a) consider $N = 1$ DS (b) train a model (c) evaluate the models on the potential test DSs of interest, (d) consider partitioning the learning set into $N > 1$ DSs according to some known characteristics (e.g. source, manufacturer, acquisition device version, population characteristics). Only accumulated knowledge of the data led us to consider the sources and the manufacturers as a segmentation levels to split the learning set into DSs. A potential segmentation of the learning set according to the ethnicity (which may be already covered by the segmentation by source), the age, the biological sex of patients stand as different levels which may be worth investing to improve fairness. Still, addressing such segmentation levels implies dealing with sensible information raising concerns about data privacy and data security. The proposed approach still works when there are more than 2 segmentation levels (not shown) but the segmentation into DSs needs to be defined on some categorical criteria. Binning a continuous variable could allow one to tackle a continuous source of bias (e.g. age). Further investigation are needed to cover this aspect.

This work builds upon the assumption that 2 DSs DS_i and DS_j do not have some shared information and are independent. Yet, this may be wrong at some point. Indeed, some patients may have data in different DSs when for instance a hospital has used different acquisition devices over time or when a patient goes to another hospital. Adapting the sampling proportions used to generate the sampling pattern could help to cope with this issue. Future work should investigate this subject.

The proposed approach relies on a fixed sampling pattern that cannot change with the dynamics of the training process (Hu et al., 2019; Ren et al., 2018). A dynamic and learnable sampling could be an advantage and cope with issues when 2 DSs are not independent. Alternatively, the problem addressed in this work could be tackled with sample weighting (Shimodaira, 2000) and may lead to similar performances provided that one samples all labels and DSs in equal proportions to avoid having batches containing samples of a single label or a single DS.

This paper focuses on binary classification through the application of breast cancer prediction, a concrete but also restricted scenario. Yet, the formalism is general enough to be tried with other modalities of data and / or other learning tasks. This shall be covered in follow up studies

REFERENCES

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL `https://www.tensorflow.org/`. Software available from tensorflow.org.

Douglas G Altman and J Martin Bland. How to obtain the p value from a confidence interval. *BMJ*, 343, 2011. ISSN 0959-8138. doi: 10.1136/bmj.d2304. URL `https://www.bmj.com/content/343/bmj.d2304`.

Solon Barocas and Andrew D. Selbst. Big Data's Disparate Impact. *SSRN eLibrary*, 2014.

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. In B. Schölkopf, J. Platt, and T. Hoffman (eds.), *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2007. URL `https://proceedings.neurips.cc/paper/2006/file/b1b0432ceafb0ce714426e9114852ac7-Paper.pdf`.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1):151–175, May 2010. ISSN 1573-0565. doi: 10.1007/s10994-009-5152-4. URL `https://doi.org/10.1007/s10994-009-5152-4`.

Yeman Brhane Hagos, Albert Gubern Mérida, and Jonas Teuwen. Improving Breast Cancer Detection Using Symmetry Information with Deep Learning. In Danail Stoyanov, Zeike Taylor, Bernhard Kainz, Gabriel Maicas, Reinhard R. Beichel, Anne Martel, Lena Maier-Hein, Kanwal Bhatia, Tom Vercauteren, Ozan Oktay, Gustavo Carneiro, Andrew P. Bradley, Jacinto Nascimento, Hang Min, Matthew S. Brown, Colin Jacobs, Bianca Lassen-Schmidt, Kensaku Mori, Jens Petersen, Raúl San José Estépar, Alexander Schmidt-Richberg, and Catarina Veiga (eds.), *Image Analysis for Moving Organ, Breast, and Thoracic Images*, Lecture Notes in Computer Science, pp. 90–97, Cham, 2018. Springer International Publishing. ISBN 9783030009465. doi: 10.1007/978-3-030-00946-5_10.

Jonathon Byrd and Zachary C. Lipton. Weighted risk minimization & deep learning. *CoRR*, abs/1812.03372, 2018. URL `http://arxiv.org/abs/1812.03372`.

Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. Smote: Synthetic minority over-sampling technique. *J. Artif. Int. Res.*, 16(1):321–357, June 2002. ISSN 1076-9757.

Koby Crammer, Michael Kearns, and Jennifer Wortman. Learning from multiple sources. *Journal of Machine Learning Research*, 9(57):1757–1774, 2008. URL `http://jmlr.org/papers/v9/crammer08a.html`.

Dominik Csiba and Peter Richtárik. Importance sampling for minibatches. *Journal of Machine Learning Research*, 19(27):1–21, 2018. URL `http://jmlr.org/papers/v19/16-241.html`.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35, 2016. URL `http://jmlr.org/papers/v17/15-239.html`.

Krzysztof J. Geras, Stacey Wolfson, S. Gene Kim, Linda Moy, and Kyunghyun Cho. High-resolution breast cancer screening with multi-view deep convolutional neural networks. *CoRR*, abs/1703.07047, 2017. URL `http://arxiv.org/abs/1703.07047`.

Hao Guan and Mingxia Liu. Domain adaptation for medical image analysis: A survey, 2021.

Haibo He and Edwardo A. Garcia. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284, 2009. doi: 10.1109/TKDE.2008.239.

Haibo He, Yang Bai, Edwardo A. Garcia, and Shutao Li. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pp. 1322–1328, 2008. doi: 10.1109/IJCNN.2008.4633969.

Zhiting Hu, Bowen Tan, Russ R Salakhutdinov, Tom M Mitchell, and Eric P Xing. Learning data manipulation for augmentation and weighting. In *Advances in Neural Information Processing Systems 32*, pp. 15764–15775. Curran Associates, Inc., 2019. URL `http://papers.nips.cc/paper/9706-learning-data-manipulation-for-augmentation-and-weighting.pdf`.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 448–456, Lille, France, 07–09 Jul 2015. PMLR. URL `http://proceedings.mlr.press/v37/ioffe15.html`.

Davood Karimi, Simon K Warfield, and Ali Gholipour. Transfer learning in medical image segmentation: New insights from analysis of the dynamics of model parameters and learned representations. *Artif Intell Med*, 116:102078, 2021 Jun 2021. ISSN 1873-2860. doi: 10.1016/j.artmed.2021.102078.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014. URL `http://arxiv.org/abs/1412.6980`. cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.

Thijs Kooi and Nico Karssemeijer. Classifying symmetrical differences and temporal change for the detection of malignant masses in mammography using deep neural networks. *Journal of Medical Imaging*, 4(4), October 2017. ISSN 2329-4302. doi: 10.1117/1.JMI.4.4.044501. URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5633751/`.

Thijs Kooi, Geert Litjens, Bram van Ginneken, Albert Gubern-Mérida, Clara I. Sánchez, Ritse Mann, Ard den Heeten, and Nico Karssemeijer. Large scale deep learning for computer aided detection of mammographic lesions. *Medical Image Analysis*, 35:303–312, January 2017. ISSN 1361-8423. doi: 10.1016/j.media.2016.07.007.

M. G. Marmot, D. G. Altman, D. A. Cameron, J. A. Dewar, S. G. Thompson, and M. Wilcox. The benefits and harms of breast cancer screening: an independent review. *British Journal of Cancer*, 108(11):2205–2240, June 2013. ISSN 1532-1827. doi: 10.1038/bjc.2013.177.

Thomas P. Matthews, Sadanand Singh, Brent Mombourquette, Jason Su, Meet P. Shah, Stefano Pedemonte, Aaron Long, David Maffit, Jenny Gurney, Rodrigo Morales Hoil, Nikita Ghare, Douglas Smith, Stephen M. Moore, Susan C. Marks, and Richard L. Wahl. A multisite study of a breast density deep learning model for full-field digital mammography and synthetic mammography. *Radiology: Artificial Intelligence*, 3(1):e200015, 2021. doi: 10.1148/ryai.2020200015. URL `https://doi.org/10.1148/ryai.2020200015`. PMID: 33937850.

Scott Mayer McKinney, Marcin Sieniek, Varun Godbole, Jonathan Godwin, Natasha Antropova, Hutan Ashrafian, Trevor Back, Mary Chesus, Greg S. Corrado, Ara Darzi, Mozziyar Etemadi, Florencia Garcia-Vicente, Fiona J. Gilbert, Mark Halling-Brown, Demis Hassabis, Sunny Jansen, Alan Karthikesalingam, Christopher J. Kelly, Dominic King, Joseph R. Ledsam, David Melnick, Hormuz Mostofi, Lily Peng, Joshua Jay Reicher, Bernardino Romera-Paredes, Richard Sidebottom, Mustafa Suleyman, Daniel Tse, Kenneth C. Young, Jeffrey De Fauw, and Shravya Shetty. International evaluation of an AI system for breast cancer screening. *Nature*, 577 (7788):89–94, January 2020. ISSN 1476-4687. doi: 10.1038/s41586-019-1799-6. URL `https://www.nature.com/articles/s41586-019-1799-6`.

Mohammad Amin Morid, Alireza Borjali, and Guilherme Del Fiol. A scoping review of transfer learning research on medical image analysis using imagenet. *Computers in Biology and Medicine*, 128:104115, 2021. ISSN 0010-4825. doi: https://doi.org/10.1016/j.compbiomed.2020.104115. URL https://www.sciencedirect.com/science/article/pii/S0010482520304467.

Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, pp. 807–814, Madison, WI, USA, 2010. Omnipress. ISBN 9781605589077.

Serena Pacilè, January Lopez, Pauline Chone, Thomas Bertinotti, Jean Marie Grouin, and Pierre Fillard. Improving Breast Cancer Detection Accuracy of Mammography with the Concurrent Use of an Artificial Intelligence Tool. *Radiology: Artificial Intelligence*, 2(6):e190208, November 2020. doi: 10.1148/ryai.2020190208. URL https://pubs.rsna.org/doi/abs/10.1148/ryai.2020190208.

Stefano Pedemonte, Brent Mombourquette, Alexis Goh, Trevor Tsue, Aaron Long, Sadanand Singh, Thomas Paul Matthews, Meet Shah, and Jason Su. A hypersensitive breast cancer detector. *CoRR*, abs/2001.08382, 2020. URL https://arxiv.org/abs/2001.08382.

Oleg S. Pianykh. *Digital Imaging and Communications in Medicine (DICOM): A Practical Introduction and Survival Guide*. Springer Publishing Company, Incorporated, 1st edition, 2010. ISBN 3642094007.

Maithra Raghu, Chiyuan Zhang, Jon Kleinberg, and Samy Bengio. Transfusion: Understanding transfer learning for medical imaging. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/eb1e78328c46506b46a4ac4a1e378b91-Paper.pdf.

Mohammad A. Rawashdeh, Warwick B. Lee, Roger M. Bourne, Elaine A. Ryan, Mariusz W. Pietrzyk, Warren M. Reed, Robert C. Heard, Deborah A. Black, and Patrick C. Brennan. Markers of good performance in mammography depend on number of annual readings. *Radiology*, 269 (1):61–67, October 2013. ISSN 1527-1315. doi: 10.1148/radiol.13122581.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In Jennifer G. Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4331–4340. PMLR, 2018. URL http://proceedings.mlr.press/v80/ren18a.html.

Thomas Schaffter, Diana S. M. Buist, Christoph I. Lee, Yaroslav Nikulin, Dezső Ribli, Yuanfang Guan, William Lotter, Zequn Jie, Hao Du, Sijia Wang, Jiashi Feng, Mengling Feng, Hyo-Eun Kim, Francisco Albiol, Alberto Albiol, Stephen Morrell, Zbigniew Wojna, Mehmet Eren Ahsen, Umar Asif, Antonio Jimeno Yepes, Shivanthan Yohanandan, Simona Rabinovici-Cohen, Darvin Yi, Bruce Hoff, Thomas Yu, Elias Chaibub Neto, Daniel L. Rubin, Peter Lindholm, Laurie R. Margolies, Russell Bailey McBride, Joseph H. Rothstein, Weiva Sieh, Rami Ben-Ari, Stefan Harrer, Andrew Trister, Stephen Friend, Thea Norman, Berkman Sahiner, Fredrik Strand, Justin Guinney, Gustavo Stolovitzky, , and the DM DREAM Consortium. Evaluation of Combined Artificial Intelligence and Radiologist Assessment to Interpret Screening Mammograms. *JAMA Network Open*, 3(3):e200265–e200265, 03 2020. ISSN 2574-3805. doi: 10.1001/jamanetworkopen.2020.0265. URL https://doi.org/10.1001/jamanetworkopen.2020.0265.

Li Shen, Laurie R. Margolies, Joseph H. Rothstein, Eugene Fluder, Russell McBride, and Weiva Sieh. Deep Learning to Improve Breast Cancer Detection on Screening Mammography. *Scientific Reports*, 9(1):12495, August 2019. ISSN 2045-2322. doi: 10.1038/s41598-019-48995-4. URL https://www.nature.com/articles/s41598-019-48995-4.

Yiqiu Shen, Nan Wu, Jason Phang, Jungkyu Park, Kangning Liu, Sudarshini Tyagi, Laura Heacock, S. Gene Kim, Linda Moy, Kyunghyun Cho, and Krzysztof J. Geras. An interpretable classifier for high-resolution breast cancer screening images utilizing weakly supervised localization. *Medical Image Analysis*, 68:101908, 2021. ISSN 1361-8415. doi: https://doi.org/10.1016/j.media.2020.101908. URL https://www.sciencedirect.com/science/article/pii/S1361841520302723.

Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000. ISSN 0378-3758. doi: https://doi.org/10.1016/S0378-3758(00)00115-4. URL https://www.sciencedirect.com/science/article/pii/S0378375800001154.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

Sadanand Singh, Thomas Paul Matthews, Meet Shah, Brent Mombourquette, Trevor Tsue, Aaron Long, Ranya Almohsen, Stefano Pedemonte, and Jason Su. Adaptation of a deep learning malignancy model from full-field digital mammography to digital breast tomosynthesis, 2020.

Peilin Zhao and Tong Zhang. Accelerating minibatch stochastic gradient descent using stratified sampling, 2014.

A Appendix

A.1 Additional information about data

This section contains additional information about the training and testing data. Table 5 provides information about the countries of origin of the sources and the number of images per status and type of view. Table 6 provides information about the types of the malignant lesions in each DS. Table 7 provides information about the age of patients in the different DSs.

A.2 Model architecture

A.2.1 General architecture

The considered DNN is a fully convolutional neural network predicting image wise probability of malignancy on FFDM images. The DNN is thus trained on whole FFDM images of shape $(1152, 832)$ with image-wise labels as explained in the main body of the paper.

The DNN exhibits a VGG-like architecture (Simonyan & Zisserman, 2015) and adopts a 2-level architecture as depicted in Figure 5. The first part of the DNN is referred to as *the Backbone*. The Backbone aims at creating an intermediate representation of the full image which facilitates prediction of the final image-wise probability of malignancy. The second part of the model is the classifier and it is in charge of predicting the final image wise probability of malignancy. The Backbone and the classifier are detailed below.

A.2.2 Backbone

The Backbone is built to take as input a patch of size $(224, 224)$ and to predict its associated label. As 5 labels are possible, the Backbone outputs a vector of size 5. It is designed in a fully convolutionnal way and thus can take as input, images of arbitrary size to form an output feature map. In the case of images at resolution $(1152, 832)$, the Backbone outputs a feature map of size $(16, 11, 5)$. This returned feature map plays the role of the intermediate representation mentioned above. This can be seen as a score vector of size 5 for each spatial position in the grid $(16, 11)$.

The backbone architecture is given in Figure 5. It builds upon a succession of convolution blocks of the form: (a) batch normalization Ioffe & Szegedy (2015), (b) 2D convolution with $(3, 3)$ kernels and ReLU activation Nair & Hinton (2010), (c) 2D max pooling with $(2, 2)$ pool size and stride of $(2, 2)$. The numbers of kernels in each convolution block are given in Figure 5. Finally, 3 2D convolution layers with resp. 1024, 512 and 5 kernels of shapes resp. $(2, 2)$, $(1, 1)$ and $(1, 1)$ and ReLU activations play the role of the patch classifier to output a vector of size 5.

A.2.3 Image classifier

To predict the final image-wise probability of malignancy, the intermediate representation built by the Backbone is fed into the classifier that processes it with convolutional blocks to obtain the final image wise prediction.

Table 5: Additional statistics about DSs countries and number of images for each type of view. CC stands for Cranio-Caudal, MLO for Medio-Lateral Oblique, other for any other views.

Segment	Source	Manufacturer	Organization	Status	view	# images
(S_1, M_a)	S_1	M_a	U.K.	benign	CC	6626
(S_1, M_a)	S_1	M_a	U.K.	malignant	CC	3536
(S_1, M_a)	S_1	M_a	U.K.	benign	MLO	6705
(S_1, M_a)	S_1	M_a	U.K.	malignant	MLO	3713
(S_1, M_a)	S_1	M_a	U.K.	benign	other	10
(S_1, M_a)	S_1	M_a	U.K.	malignant	other	9
(S_1, M_b)	S_1	M_b	U.K.	benign	CC	262
(S_1, M_b)	S_1	M_b	U.K.	malignant	CC	195
(S_1, M_b)	S_1	M_b	U.K.	benign	MLO	264
(S_1, M_b)	S_1	M_b	U.K.	malignant	MLO	204
(S_1, M_b)	S_1	M_b	U.K.	benign	other	1
(S_1, M_b)	S_1	M_b	U.K.	malignant	other	9
(S_2, M_a)	S_2	M_a	France	benign	CC	10868
(S_2, M_a)	S_2	M_a	France	malignant	CC	66
(S_2, M_a)	S_2	M_a	France	benign	MLO	10692
(S_2, M_a)	S_2	M_a	France	malignant	MLO	70
(S_2, M_a)	S_2	M_a	France	benign	other	311
(S_2, M_a)	S_2	M_a	France	malignant	other	24
(S_2, M_b)	S_2	M_b	France	benign	CC	9121
(S_2, M_b)	S_2	M_b	France	malignant	CC	66
(S_2, M_b)	S_2	M_b	France	benign	MLO	9144
(S_2, M_b)	S_2	M_b	France	malignant	MLO	70
(S_2, M_b)	S_2	M_b	France	benign	other	775
(S_2, M_b)	S_2	M_b	France	malignant	other	10
(S_3, M_a)	S_3	M_a	U.S.A.	benign	CC	7046
(S_3, M_a)	S_3	M_a	U.S.A.	malignant	CC	547
(S_3, M_a)	S_3	M_a	U.S.A.	benign	MLO	7144
(S_3, M_a)	S_3	M_a	U.S.A.	malignant	MLO	558
(S_3, M_a)	S_3	M_a	U.S.A.	benign	other	2
(S_3, M_a)	S_3	M_a	U.S.A.	malignant	other	1
(S_3, M_b)	S_3	M_b	U.S.A.	benign	CC	22982
(S_3, M_b)	S_3	M_b	U.S.A.	malignant	CC	1011
(S_3, M_b)	S_3	M_b	U.S.A.	benign	MLO	22994
(S_3, M_b)	S_3	M_b	U.S.A.	malignant	MLO	1045
(S_3, M_b)	S_3	M_b	U.S.A.	benign	other	30
(S_3, M_b)	S_3	M_b	U.S.A.	malignant	other	3
(S_4, M_c)	S_4	M_c	France	benign	CC	1984
(S_4, M_c)	S_4	M_c	France	malignant	CC	11
(S_4, M_c)	S_4	M_c	France	benign	MLO	2347
(S_4, M_c)	S_4	M_c	France	malignant	MLO	12
(S_4, M_c)	S_4	M_c	France	benign	other	56
(S_4, M_c)	S_4	M_c	France	malignant	other	5
(S_4, M_d)	S_4	M_d	France	benign	CC	9938
(S_4, M_d)	S_4	M_d	France	malignant	CC	61
(S_4, M_d)	S_4	M_d	France	benign	MLO	9777
(S_4, M_d)	S_4	M_d	France	malignant	MLO	59
(S_4, M_d)	S_4	M_d	France	benign	other	25
(S_4, M_d)	S_4	M_d	France	malignant	other	13
(S_5, M_c)	S_5	M_c	France	benign	CC	20802
(S_5, M_c)	S_5	M_c	France	malignant	CC	64
(S_5, M_c)	S_5	M_c	France	benign	MLO	16955
(S_5, M_c)	S_5	M_c	France	malignant	MLO	57
(S_5, M_c)	S_5	M_c	France	benign	other	1194
(S_5, M_c)	S_5	M_c	France	malignant	other	10

Table 6: Malignant lesions types over the different DSs

Segment	Source	Manufacturer	Status	Lesion type	# images
(S_1, M_a)	S_1	M_a	malignant	calcification	2069
(S_1, M_a)	S_1	M_a	malignant	calcification & soft tissue lesion	185
(S_1, M_a)	S_1	M_a	malignant	soft tissue lesion	5004
(S_1, M_b)	S_1	M_b	malignant	calcification	117
(S_1, M_b)	S_1	M_b	malignant	calcification & soft tissue lesion	14
(S_1, M_b)	S_1	M_b	malignant	soft tissue lesion	277
(S_2, M_a)	S_2	M_a	malignant	calcification	44
(S_2, M_a)	S_2	M_a	malignant	calcification & soft tissue lesion	5
(S_2, M_a)	S_2	M_a	malignant	soft tissue lesion	111
(S_2, M_b)	S_2	M_b	malignant	calcification	41
(S_2, M_b)	S_2	M_b	malignant	calcification & soft tissue lesion	2
(S_2, M_b)	S_2	M_b	malignant	soft tissue lesion	103
(S_3, M_a)	S_3	M_a	malignant	calcification	289
(S_3, M_a)	S_3	M_a	malignant	calcification & soft tissue lesion	44
(S_3, M_a)	S_3	M_a	malignant	soft tissue lesion	773
(S_3, M_b)	S_3	M_b	malignant	calcification	670
(S_3, M_b)	S_3	M_b	malignant	calcification & soft tissue lesion	89
(S_3, M_b)	S_3	M_b	malignant	soft tissue lesion	1300
(S_4, M_c)	S_4	M_c	malignant	calcification	7
(S_4, M_c)	S_4	M_c	malignant	calcification & soft tissue lesion	2
(S_4, M_c)	S_4	M_c	malignant	soft tissue lesion	19
(S_4, M_d)	S_4	M_d	malignant	calcification	37
(S_4, M_d)	S_4	M_d	malignant	calcification & soft tissue lesion	5
(S_4, M_d)	S_4	M_d	malignant	soft tissue lesion	91
(S_5, M_c)	S_5	M_c	malignant	calcification	10
(S_5, M_c)	S_5	M_c	malignant	calcification & soft tissue lesion	13
(S_5, M_c)	S_5	M_c	malignant	soft tissue lesion	108

Table 7: Average age (mean ± std) for patients in the different training and testing DSs

Segment	Source	Manufacturer	Status	Age in years
(S_1, M_a)	S_1	M_a	benign	59 ± 7
(S_1, M_a)	S_1	M_a	malignant	60 ± 8
(S_1, M_b)	S_1	M_b	benign	59 ± 8
(S_1, M_b)	S_1	M_b	malignant	60 ± 11
(S_2, M_a)	S_2	M_a	benign	54 ± 10
(S_2, M_a)	S_2	M_a	malignant	57 ± 14
(S_2, M_b)	S_2	M_b	benign	54 ± 11
(S_2, M_b)	S_2	M_b	malignant	61 ± 16
(S_3, M_a)	S_3	M_a	benign	57 ± 10
(S_3, M_a)	S_3	M_a	malignant	61 ± 10
(S_3, M_b)	S_3	M_b	benign	54 ± 10
(S_3, M_b)	S_3	M_b	malignant	60 ± 10
(S_4, M_c)	S_4	M_c	benign	53 ± 11
(S_4, M_c)	S_4	M_c	malignant	57 ± 14
(S_4, M_d)	S_4	M_d	benign	55 ± 11
(S_4, M_d)	S_4	M_d	malignant	59 ± 14
(S_5, M_c)	S_5	M_c	benign	54 ± 11
(S_5, M_c)	S_5	M_c	malignant	61 ± 13

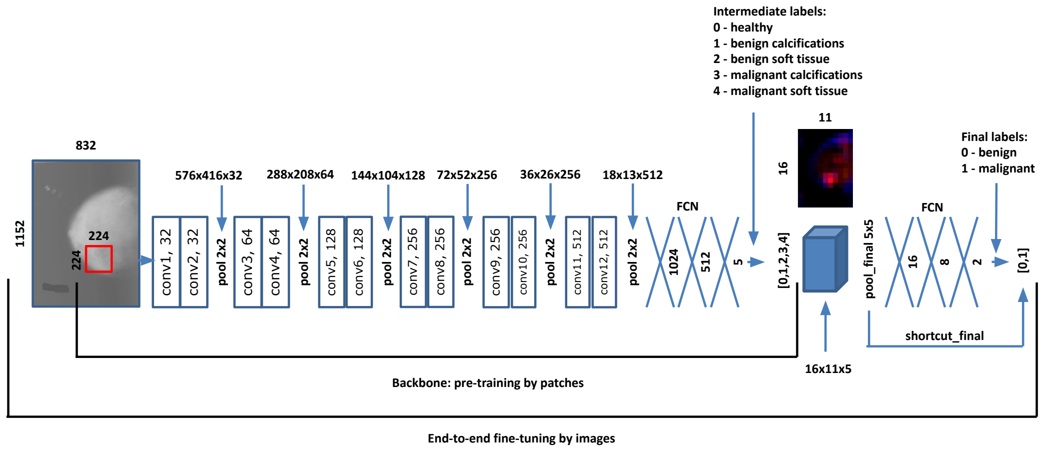

Figure 5: Full model architecture composed of a Backbone and a classifier. The Backbone is in charge of building a meaning full feature representation. The classifier is in charge of predicting image wise probability of malignancy. The Backbone is first pre-trained to classify (224, 224) patches into 5 classes: healthy tissue, benign / malignant soft tissue lesion, benign / malignant calcification. The whole model is then fine-tuned to classify (1152, 832) mammography images into 2 classes: benign and malignant. The number of convolution kernels of each convolution block are given as well as the shapes of the intermediate feature maps.

The classifier builds upon a MaxPooling layer with poolsize of $(5, 5)$ and stride of $(5, 5)$. Then, it is composed of 2 sub-parts which process the max-pooled feature maps. The first sub-part builds on 3 successive blocks of convolution of the form: (a) batch normalization layer, (b) ReLU activation (c) 2D convolution with resp. 16, 8 and 2 kernels of size resp. $(3, 2)$, $(1, 1)$ and $(1, 1)$. This sub-part returns a vector of size 2. The second sub-part is composed of a batch normalization layer and a convolution layer with 2 kernels of size $(3, 2)$. This sub-part returns a vector of size 2 as well. The outputs of the sub-parts are finally summed up to form the final image wise prediction.

A.2.4 Remark on training

The 2-level architecture is also reflected in the training procedure which is a two-step process. First, the Backbone is pre-trained on patches to predict their labels. Then, the Backbone weights are used to initialized the full DNN weights and the DNN is then fine-tuned on whole FFDM images to make image wise predictions of malignancy.

A.3 Backbone training

The training process of the Backbone is detailed below, the training / fine-tuning process of the DNN being described in the main body of the paper. Practically, the Backbone is fed with patches centered on annotated lesions and is trained to predict which label among 5 possible labels should be assigned to the patch.

A.3.1 Data

Lesions have been annotated on some images from the DS (S_1, M_a) by trained radiologists. Lesions positions and lesions labels have been assigned. Lesions have been categorized into 5 classes: healthy tissue, benign soft tissue, benign calcification, malignant soft tissue, malignant calcification. The numbers of annotated lesions and images are provided in Table 8.

Cross-validation scheme Similarly as described in the main paper, the learning set is divided into training and validation data using a source-wise and patient-wise split: 80% (resp. 20%) of the

Table 8: Numbers of annotated lesions of each label. The corresponding numbers of annotated images are provided as well.

Segment	Label	# patches	# images
(S_1, M_a)	benign calcification	1611	1162
(S_1, M_a)	benign soft tissue lesion	4287	2252
(S_1, M_a)	healthy tissue	2597	1773
(S_1, M_a)	malignant calcification	2629	2254
(S_1, M_a)	malignant soft tissue lesion	5747	5187

patients of each source are used for training (resp. validation). Furthermore, the split is made such that the ratio of malignant / benign patients is similar for the validation and the training sets.

The same train-validation split as described in the paper is used to train the Backbone. This means that the learning samples of a patient used for training (resp. validation of) the BackBone, are also used for training (resp. validation of) the DNN.

Image pre-processing and data augmentation Similarly as in the paper, normalization and pre-processing techniques are used to standardize the FFDM images. A patch of size nearly equal to $(224, 224)$ (see below why) is first cropped from a FFDM image. The pixels inside the breast mask are re-normalized to have values in $[0, 1]$. The pixels outside mask are set to a constant value such that the average pixel value of the image is always the same. A standard augmentation pipeline (random rotation, zoom, shearing...) is used to augment images used for training and to return an augmented patch of size $(224, 224)$. No augmentation technique is used for validation patches.

The augmentation transform is always pre-computed, to know exactly the dimension of the patch to crop. This way, every pixel in the augmented patch is an output, through the augmentation transform, of a pixel from the original patch: the transform is surjective. This is why the cropping step, for training patches, does not return patches with an exact shape of $(224, 224)$ but with a shape of nearly $(224, 224)$.

A.3.2 Training setup

The Backbone is trained to minimize the categorical cross-entropy defined for 5 classes. The training process is done on 20000 learning steps, a sufficiently large number of steps to see convergence. Samples labels are balanced within a batch to fight potential issues related to class imbalance. The training relies on Adam optimizer (Kingma & Ba, 2014) and a learning rate equal to 10^{-3}. Exponential moving average with decay 0.99 of all trainable parameters are tracked, their averaged versions are then used for inference on validation data. The training is done on 2 Tesla P100 and a batch size of 64 samples is used (32 per GPU).

A validation step is made every 500 training steps. The Weighted Logloss per lesion (WLL / lesion), the standard logloss with samples weights estimated on the validation set was used to monitor the training progress. The checkpoint achieving the best Weighted Logloss per lesion (WLL / lesion) on the validation set is kept to initialize the DNN weights.

The models and the experiments were carried out with Tensorflow 2.3 (Abadi et al., 2015).

A.4 Additional figure for experiment 2

On top of AUC gain - loss training a DNN with ”Bal. labels” leads to a poor synchronization of OPs on the ROC curves obtained on (S_3, M_a) and (S_3, M_b), see Figure 6 - left. This suggests that the DNN works in 2 different regimes on the 2 test DSs: a non desired property which stresses the manufacturer bias the DNN has integrated. On the other hand, the DNN trained with ”P. sampling” gives visually better synchronized OPs on the ROC curves of test DSs. This suggests that this DNN behaves quite similarly on them and that it is not influenced by the manufacturer to take its decisions.

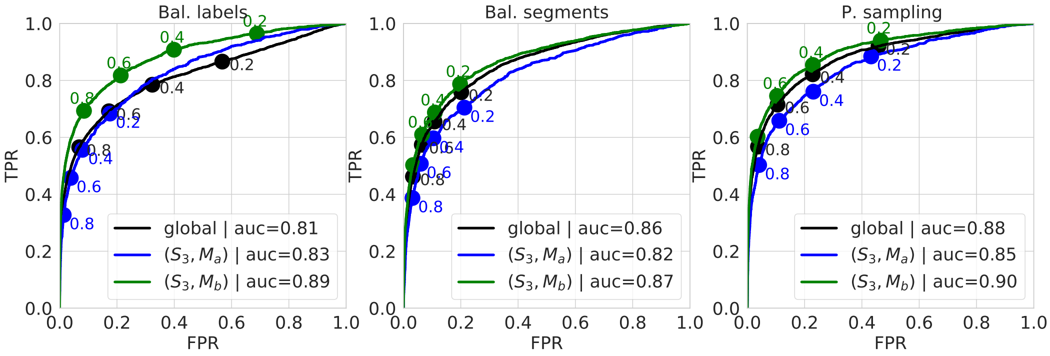

Figure 6: *Global* and *local* ROC curves on: (S_3, M_a) and (S_3, M_b). Black: *global* ROC curve on $(S_3, M_a) \cup (S_3, M_b)$. Blue (resp green): ROC curve on (S_3, M_a) (resp. (S_3, M_b)).